

5
010



This is to certify that the
dissertation entitled


**ANALYTICAL AND QUANTITATIVE CHARACTERIZATION
OF WIRELESS SENSOR NETWORKS**

presented by

MUHAMMAD USMAN ILYAS

has been accepted towards fulfillment
of the requirements for the

Doctoral degree in Electrical Engineering


Major Professor's Signature

12/16/2009

Date

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.
MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

**ANALYTICAL AND QUANTITATIVE CHARACTERIZATION OF
WIRELESS SENSOR NETWORKS**

By

Muhammad Usman Ilyas

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Electrical Engineering

2009

ABSTRACT

ANALYTICAL AND QUANTITATIVE CHARACTERIZATION OF WIRELESS SENSOR NETWORKS

By

Muhammad Usman Ilyas

In this thesis we characterize key properties of wireless sensor networks (WSN) by analytical and quantitative methods. These include the link-layer bit error rate (BER) process, network lifetime and topology.

For the analysis of the BER process, we collected a large set of packet traces over IEEE 802.15.4 links. Our packet traces distinguish themselves from other data sets in that they record channel state information (CSI) as well as full and partial packet erasures. A channel model, which is conditioned on observed CSI, is developed. This conditional model reduces the variance of the BER's distribution by one order of magnitude. Packet traces are also analyzed to determine memory length of bit errors. Correlation analysis of bit and symbol level traces reveals that memory length of errors in all traces is 2 bits and 2 symbols, respectively. For packet-level traces consisting of BER measurements of individual packets the traditional correlogram based analysis fails and so we introduce relative mutual information (RMI) as a more robust method for measuring channel memory. RMI based analysis of packet traces shows that memory length of the BER ranges from 0 to 2sec.

The research on the network lifetime problem proposes joint minimization of mean and variance of sensor power consumption rates as an alternative to the minimax formulation of the lifetime problem in WSN. This proposed statistical optimization objective better fits the vision of WSNs consisting of large numbers of inexpensive, redundant, disposable sensors than the minimax formulation which focuses on the top power consuming node. We formulate this problem in quadratic program (QP) form. To avoid scalability issues of using a QP, an approximate dynamic program

(DP) formulation of lower complexity rooted in operational rate-distortion theory is developed. For a randomly generated WSN of 100 nodes DP exhibits upto 44% reduction in variance at the cost of 19% increase in its mean, with many intermediate operating points of higher benefit/cost ratios to choose from.

The research on topological characteristics of WSNs explores the possibility of building WSNs with small-world topologies that combine desirable properties of Euclidean/lattice graphs with those of random graphs. An analytical model is developed to explain the phase difference in characteristic path length and clustering coefficient in lattice graphs when shortcut links of limited range are used. We test and implement a software based system for commercial-off-the-shelf motes that increases communication range of links in WSNs using cooperative communication and diversity combining. A trace based implementation demonstrates proof-of-concept of its ability to reduce the fraction of packets with errors on a channel from 20% down to 1% and reduce the BER of packets that cannot be corrected. This is followed by an implementation on the Crossbow Imote2 sensor mote. Results from the mote based implementation show an increase in packet reception rate from 22 – 30% to 73 – 76%.

Finally, we develop a centrality measure to identify well connected clusters of central nodes for the placement of network resources. For mesh network topologies that are characteristic of WSNs, eigenvector centrality (EVC) consistently fails to identify more than a single, arbitrarily located cluster of nodes as the most central. We introduce principal component centrality (PCC), a node centrality inspired by the Karhunen Loève transform/principal component analysis. We demonstrate PCC's ability to identify a larger number of central hub nodes than EVC, depending on the number of features used in its computation.

COPYRIGHT

Copyright by
Muhammad Usman Ilyas
2009

DEDICATION

To my wife, *Ayesha*
and my daughter, *Haadiya*.

ACKNOWLEDGMENT

I would like to acknowledge my advisor Professor Hayder Radha for his years of support through the Ph.D. program, for his professional mentoring and counsel. I would also like to thank Professor Subir K. Biswas, Professor Tongtong Li and Professor Philip K. McKinley for the guidance and valuable feedback they provided as members of my Ph.D. program committee.

I would like to acknowledge the Higher Education Commission of the government of Pakistan, the National Science Foundation and Michigan State University for generously providing funding at different stages during my Ph.D. program.

Many thanks to my colleagues at the WAVES lab Khayam, Shirish, Kiran, Sauleh, Sohraab, Nima, Rami, Yongju, Moonseong, Ahmed and Aqeel for letting me bounce off ideas, many, many technical discussions and their company on our coffee rounds & all-nighters. I am also grateful to Khawar, Awais, Aparna, Keyur and Zubair for their friendships.

None of this would have been possible without the constant support and encouragement of my wife Ayesha. I would like to acknowledge Ayesha's parents without whose active support and assistance I would not be at MSU. I am grateful to my parents whose years of investment in my education allowed me to pursue graduate studies. Finally, I want to acknowledge my lovely daughter Haadiya for being a reality check and bring balance merely by her presence.

Muhammad Usman Ilyas

As the area of your knowledge grows, so does the periphery of your ignorance.

Neil deGrasse Tyson

TABLE OF CONTENTS

List of Tables	xii
List of Figures	xiii
1 Introduction	1
1.1 Organization	2
1.2 Overview of Contributions	2
2 Network Channel Capacity of IEEE 802.15.4 Wireless Sensor Networks Under Reachback	6
2.1 Introduction	7
2.2 System Model	9
2.2.1 Wireless Networking Standards for WSNs	9
2.2.2 Source & Channel Coding	15
2.2.3 Power Consumption Model	17
2.3 Channel Model	18
2.3.1 1-Hop Communication	19
2.3.2 Overlay Network Communication	24
2.3.3 Sensor-To-Base Station Capacity	27
2.4 Results and Analysis	28
3 CSI Driven Model of BER Process on IEEE 802.15.4 Links	33
3.1 Introduction	34
3.1.1 Previous Work	35
3.2 Trace Collection Setup	37
3.2.1 Experimental Setup	37
3.2.2 Packet Payload	38
3.2.3 Trace Generation	39
3.2.4 Channel State Information	39
3.2.5 Spectral and Environmental Diversity	44
3.3 Correlation Analysis of CSI Measures	45
3.4 CSI Driven BER Model	50
3.5 Model Evaluation	52
3.5.1 Variance Reduction	52
3.5.2 Dependence On Deployment Environment	53
3.6 Conclusions	56

4	Memory Properties of the Link-level BER Process in IEEE 802.15.4 Links	58
4.1	Introduction	59
4.2	Memory Length Measurement By Correlation Analysis	61
4.2.1	Correlation Function	61
4.2.2	Correlograms of Bit and Symbol-level Traces	64
4.2.3	Correlograms of Packet-level Traces	64
4.3	Hurst Analysis of Packet-level BER Process	67
4.3.1	Observations For MC Trace Set	69
4.3.2	Observations For ME Trace Set	71
4.4	Memory Length Measurement By Relative Mutual Information	71
4.4.1	Shannon Information Measures	71
4.4.2	Description: Relative Mutual Information	72
4.4.3	Discussion	76
4.5	Conclusions	78
5	A Statistical Measure Of Network Lifetime For Wireless Sensor Networks	80
5.1	Introduction	81
5.2	Previous Work	82
5.3	Novelty Of Approach	83
5.4	Network Model	85
5.5	Quadratic Program Formulation	86
5.6	Results	91
5.7	Conclusions	92
6	A Dynamic Programming Approach to Maximizing Lifetime of Sensor Networks	95
6.1	Introduction	96
6.2	Previous Work	98
6.3	Network Model	99
6.3.1	Device Model	99
6.3.2	Link Model	100
6.4	Problem Formulation	102
6.5	Route Discovery	104
6.5.1	Bottleneck Edge Disjoint Paths	106
6.5.2	Bottleneck Node Disjoint Paths	106
6.5.3	Edge Disjoint Paths	106
6.5.4	Node Disjoint Paths	108
6.6	Dynamic Programming	108
6.6.1	Theoretical Background	109
6.6.2	Dynamic Programming Algorithm	110
6.6.3	Computational Complexity of Finding Optimal Solution	114
6.7	Performance Analysis	115

6.7.1	Mean-Variance Trade-off	116
6.7.2	Spatial Redistribution of Energy	119
6.8	Conclusions	122
7	Mean-Field Solution of Small-World Wireless Sensor Network Models With Range Limited Shortcuts	127
7.1	Introduction	128
7.2	Background: Small-World Networks	129
7.2.1	Characteristic Path Length	131
7.2.2	Clustering Coefficient	131
7.2.3	Small World, Geometric and Random Graphs	132
7.3	Small-World Topology Construction Methods for Wireless Networks	133
7.3.1	Hybrid Sensor Network	133
7.3.2	Multi-radio Network	133
7.3.3	Receiver Side Cooperation	134
7.4	Mean Field Analysis of Small-World Wireless Networks	135
7.4.1	Clustering Coefficient	135
7.4.2	Characteristic Path Length	139
7.5	Observations	148
7.6	Conclusions	151
8	Enabling Cooperative Communication and Diversity Combination in IEEE 802.15.4 Wireless Networks Using Off-the-shelf Sensor Motes	157
8.1	Introduction	158
8.2	Related Work	161
8.3	SIMO Diversity Combining Techniques	163
8.3.1	Selection Diversity	164
8.3.2	Equal Gain Diversity	166
8.3.3	Maximal Ratio Diversity	167
8.4	gPMSS Protocol	168
8.4.1	gPMSS Cluster Creation	169
8.4.2	Error-free Reception by at Least One Recipient	171
8.4.3	Erroneous Reception by All Recipients	172
8.5	Trace Based Proof of Concept	173
8.5.1	Experimental Setup	173
	Packet Payload	174
	Trace Generation	175
8.5.2	Channel State Information	176
8.5.3	Implementation Results	176
	PER and PLR Analysis	178
	BER Analysis	180
8.6	gPMSS Protocol Implementation	181
8.7	Results and Analysis	183
8.7.1	Packet Reception Rate	183

8.7.2	Energy Per Packet	184
8.7.3	Packet Transmission Attempts	188
8.8	Conclusions	189
9	Principal Component Centrality as a Measure of Node Centrality in Communication Networks	191
9.1	Introduction	192
9.2	Background	194
9.2.1	Degree Centrality	195
9.2.2	Closeness Centrality	195
9.2.3	Betweenness Centrality	196
9.2.4	Eigenvector Centrality	196
9.2.5	The Need for a New Centrality Measure	198
9.3	Principal Component Centrality	200
9.4	Evaluation	203
9.4.1	Interpretation of Eigenvalues	203
9.4.2	Interpretation of Eigenvectors	205
9.4.3	Graphical Interpretation of PCC	208
9.4.4	Effect of Number of Features on PCC	211
9.5	Conclusions	212
10	Conclusions	214
10.1	Channel Modeling	215
10.2	Network Lifetime	216
10.3	WSN Topology	218
	Bibliography	221

LIST OF TABLES

2.1	Tabular listing of relevant features of the three wireless networking technologies under consideration for use in WSNs.	11
3.1	Traffic information and statistics collected in various studies.	36
3.2	State space variables, symbols and values.	43
3.3	Collection environments of various traces in the ME trace set.	46
3.4	Error rates in trace sets.	46
3.5	Cross correlation of different random processes.	49
3.6	Expected value of variance when using different combinations of RSSI and LQI as CSI.	54
6.1	Symbols and notation.	101
6.2	MLE parameters of marginal histograms generated from applying DPA with BED, BND, ED and ND route discovery algorithms to 100 randomly generated network topologies.	118
8.1	Packet counts.	184
8.2	PRR of individual nodes without gPMSS, PRR with gPMSS protocol, PRR gain for individual receivers R_1 , R_2 and R_3 due to selection diversity, and the PRR gain due to diversity combining.	185
8.3	Energy consumed by transmissions at transmitter and receiver side per error-free received packet. Columns (1) and (2) correspond to the baseline scenario using only retransmissions. Columns (3) and (4) correspond to the case when only selection diversity is used by receivers. Columns (5) and (6) corresponds to the case where a full implementation of gPMSS is used that employs diversity combining (equal gain or maximal-ratio) in addition with selection diversity. . .	186

LIST OF FIGURES

2.1	Physical layout & network topology of WSN.	8
2.2	Relationship between δ_{DEC-CL} , δ_{DEC-ON} , $\delta_{DEC-ON-E2E}$ and $\delta_{DEC-ON-S2BS}$	14
2.3	Slepian-Wolf coding in cluster-level communication.	14
2.4	End-to-End channel between a transmitter and receiver on a multihop wireless network.	19
2.5	<i>EPS</i> spent across the network for varying degrees of spatial correlation and number of CLHs.	29
2.6	<i>EPS</i> spent across the network for varying degrees of spatial correlation and number of CLHs.	30
2.7	<i>EPS</i> spent across the network for varying degrees of spatial correlation and number of CLHs.	31
3.1	Equipment setup for trace collection.	40
3.2	CC2420 MAC frame format used for experiments.	40
3.3	Office deployment environment.	41
3.4	Residential deployment environment.	41
3.5	IEEE 802.11b and IEEE 802.15.4 channels in the ISM band.	47
3.6	PDF of B conditioned on Λ and $\Phi = 1, 2$	50
3.7	PDF of B conditioned on P and $\Phi = 1, 2$	51
3.8	Values of b for various (λ, ρ)	52

3.9	The PDFs of the BER obtained from the actual traces for various LQI measurements at an RSSI of $88dBm$, $p_B(\beta \lambda, \rho = 88dBm, \phi = 1, 2)$	53
3.10	The PDFs of the BER for the same range of LQI measurements at RSSI of $88dBm$ as modeled by a discretized exponential PDF, $\tilde{p}_B(\beta \lambda, \rho = 88dBm, \phi = 1, 2)$	54
3.11	Histogram of KDD values for a) Model from ‘Residential’ trace set and ‘Lab’ traces and b) Model from ‘Hallway’ trace set and ‘Outdoor’ traces.	57
4.1	Auto-correlation functions for bit level traces of the MC and ME trace sets.	62
4.2	Auto-correlation functions for symbol level traces of the MC and ME trace sets.	63
4.3	Auto-correlation functions for traces of the MC trace sets.	66
4.4	Auto-correlation functions for traces of the ME trace sets.	67
4.5	BER process of trace MC-25 after filtering by 600 point averaging filter.	68
4.6	Plots of estimates of the Hurst parameter obtained using various techniques along with their average BERs, PERs and PLRs.	70
4.7	For traces MC-11, MC-12, MC-13, MC-14 and MC-15 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$	73
4.8	For traces MC-16, MC-17, MC-18, MC-19 and MC-20 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$	74
4.9	For traces MC-21, MC-22, MC-23, MC-24, MC-25 and MC-26 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$	75

4.10	For traces ME-1, ME-2, ME-3, ME-4, ME-5 and ME-6 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$	77
4.11	For traces ME-7, ME-8, ME-9, ME-10, ME-11 and ME-12 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$	78
5.1	The law of conservation of flow requires that the sum of incoming flows $q_{j,i}$ and data Q_i generated at node n_i must equal the sum of all outgoing flows q_{ik}	87
5.2	Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 10$	92
5.3	Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 15$	93
5.4	Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 20$	94
6.1	Paths from n_{99} to n_0	105
6.2	N lists of routes sorted in ascending order of path energies.	109
6.3	Selection of next optimal point in the MV-plane by DPA.	111
6.4	Mean-Variance tradeoffs offered by BED, BND, ED and ND paths.	114
6.5	Plots of percent decrease in variance against percent increase in mean.	120
6.6	Marginal histograms of percent increase $\mu_{\Delta\mu_E}$ for the scatter plots in figures 6.5a through 6.5d.	121
6.7	Marginal histograms of percent decrease $\mu_{\Delta\sigma_E^2}$ for the scatter plots in figures 6.5a through 6.5d.	122
6.8	Diffusion plot of energy consumption rates averaged over all 100 networks under SPF routing.	123
6.9	Differential diffusion plots of energy consumption rates averaged over 100 networks using BED paths and DPA.	124

6.10	Differential diffusion plots of energy consumption rates averaged over 100 networks using BND paths and DPA.	124
6.11	Differential diffusion plots of energy consumption rates averaged over 100 networks using ED paths and DPA.	125
6.12	Differential diffusion plots of energy consumption rates averaged over 100 networks using ND paths and DPA.	125
7.1	Illustrated examples for three different classes of graphs; a) Geometric graph, b) Random graph, and c) Small world graph.	131
7.2	Overlapping communication regions of two communicating sensor nodes in a WSN.	137
7.3	Geometry of WSN deployment and regions within it with respect to an individual sensor v_j	141
7.4	Plots of clustering coefficient C [left] and characteristic path length L , [right] as functions of μ for different values of R . Network parameters that remain fixed are $A = 10000$, $k_{global} = 4$, $\rho = 10$, $\xi = 3$ and $n_c = 1$.	148
7.5	Plots clustering coefficient C [left] and characteristic path length L , [right] as a function of μ for different ratios of global scale link to local scale link communication range ξ . Network parameters remain fixed at $A = 10000$, $k_{global} = 4$, $\rho = 10$, $n_c = 1$ and $R = 4$	149
7.6	Plots clustering coefficient C [left] and characteristic path length L , [right] as a function of μ for different values of k_{global} . Network parameters remain fixed at $A = 10000$, $\rho = 10$, $\xi = 3$, $R = 3$ and $n_c = 3$.	150
7.7	Graphical representation of integration term of $d(\Gamma(v))$	155
8.1	Application of generalized gPMSS in a wireless sensor network with mesh topology. Path from transmitter T to receiver $R1$ marks the multihop path that would be taken in a network without gPMSS. Dashed line links between T and receivers $R1$, $R2$ and $R3$ denote the longer range but high loss links that are used under Generalized gPMSS. . .	160
8.2	Illustration of logical functioning of selection diversity.	168
8.3	Illustration of logical functioning of equal gain diversity.	168
8.4	Illustration of logical functioning of maximal-ratio gain diversity. . . .	169
8.5	gPMSS protocol operations.	170

8.6	Equipment setup for trace collection.	173
8.7	CC2420 MAC frame format used for experiments.	174
8.8	PDF of BER experienced by receivers $R1$, $R2$ and $R3$ ($p_B(\beta = 0)$ is cropped out for better view of non-zero range.	177
8.9	PDF of LQI experienced by receivers $R1$, $R2$ and $R3$	178
8.10	PDF of RSSI experienced by receivers $R1$, $R2$ and $R3$	179
8.11	PER, PLR and PER+PLR experienced by receivers $R1$, $R2$ and $R3$ without gPMSS diversity combining and with selection, equal gain, and maximal ratio diversity combining.	180
8.12	Histogram of BERs observed by receivers $R1$, $R2$ and $R3$ without gPMSS diversity combining and with selection, equal gain, and maximal ratio diversity combining.	181
8.13	The energy in μJ consumed by transmitter and receivers per successfully delivered packet.	185
8.14	Maximum number of transmission attempts m versus delivery guarantee $g(\%)$	188
9.1	This figure shows a graph on the lower plane, overlaid with another plane of the interpolated surface plot of node centrality scores. The centrality planes typically exhibit a number of peaks or local maxima.	193
9.2	A spatial graph of 200 nodes. Node colors are indicative of the range in which their EVC falls.	197
9.3	[Top] Histogram of eigenvalues of adjacency matrix and Laplacian matrix \mathbf{A} of network in figure 9.2; [Bottom] Cumulative sum of the sequence of eigenvalues of adjacency matrix and Laplacian matrix of network in figure 9.2 when sorted in descending order of magnitudes. In both figures the lines plotted in red color are averages of 50 networks generated randomly with the same parameters.	200
9.4	Reconstructed topologies of the graph from figure 9.2 using only the first 1, 2, 3, 5, 10, 15, 50 and all 200 eigenvectors.	201
9.5	Spectral drawing of graph in three dimensions using entries of \mathbf{x}_1 , \mathbf{x}_2 , and \mathbf{x}_3 for the three coordinate axes. Nodes are colored according to their \mathbf{C}_{15} PCC.	205

9.6	PCC of nodes in network of figure 9.2 when computed using first (a) 1 and (b) 2 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.	207
9.7	PCC of nodes in network of figure 9.2 when computed using first (a) 3 and (b) 5 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.	208
9.8	PCC of nodes in network of figure 9.2 when computed using first (a) 10 and (b) 15 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.	209
9.9	PCC of nodes in network of figure 9.2 when computed using first (a) 50 and (b) all 200 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes.	210
9.10	Plot of phase angles ϕ (in radians) of PCC vectors with the EVC vector for the graph in figures 9.6, 9.7, 9.8 and 9.9.	212

Chapter 1

Introduction

1.1 Organization

This research characterizes, analytically and quantitatively, the bit error rate (BER) process, the network lifetime and the topological properties of wireless sensor networks (WSN) [2], [59]. The following section describes the contributions made in this dissertation.

1.2 Overview of Contributions

Chapter 2 provides an analytical model for the end-to-end capacity between nodes in a WSN and the base station. This study distinguishes itself from previous studies of wireless network capacity in that it assumes a many-to-one data flow (in place of an all-to-all flow) producing what is called the funneling or reachback effect. The model also takes into account in-network data fusion/ compression. We consider scenarios in which there is a) no compression, b) opportunistic compression and c) perfect compression. The model shows that the distribution of power consumption rates of nodes is skewed due to the uneven traffic load flowing through nodes. Power consumption is lessened in systems in which sensor measurements are highly correlated and take advantage of it by employing aggressive data compression like Slepian-Wolf coding [114].

Chapters 3 and 4 relate to modeling of the BER process in IEEE 802.15.4 channels low-rate wireless personal area networks (LR-WPAN). Chapter 3 describes the experimental setup used to collect a large set of residual packet traces. These traces distinguish themselves from traces collected by other studies in that they also capture channel state information (CSI) measurements alongside every packet and events such as packet losses and truncated packets. The traces are used to establish the degree of correlation between CSI measurements and the BER process that packets

are subjected to. After establishing the usefulness of CSI for estimating BER, we develop CSI-driven models of the BER process for sets of traces collected in different environments. Divergence measures between the models derived from different traces show that there is very small average divergence between different models leading to a CSI-driven BER model that is independent of physical environment. Thus CSI measurements of traffic received over a particular link can be used to estimate the BER and assign it a link cost. Chapter 4 analyzes the memory properties of errors in the traces. Correlation based analysis on bit and symbol level traces reveals a constant memory length of 2 bits and 2 symbols, respectively. Furthermore, we highlight the inability of correlation analysis to provide a clear measure of memory length of the BER process when a channel is subjected to periodic interference and arbitrary selection of a significance threshold for the correlation function. To address this shortcoming of the correlation function based analysis we introduce relative mutual information (RMI) [56], a normalized form of Shannon's mutual information (MI) [32] that provides a more robust means of measuring memory length.

Chapters 5 and 6 contain our work on the network lifetime problem in WSNs. The lifetime of a WSN is typically defined as the time until the first node runs out of power. The corresponding optimization problem can be formulated as a minimax linear program (LP) ([18]). However, the vision of the future of WSN calls for a network consisting of a very large number of devices that are inexpensive and redundant. The fact that WSN applications by their very nature impose a many-to-one data flow that produces a funneling effect (see Milgram [82]) leads to large variations in the power consumption rates in nodes. Chapter 5 proposes the redistribution of traffic routed to a WSN's base station by the joint minimization of the mean and variance of power consumption rates. Such a statistical objective function cannot be held hostage by the life of an individual sensor but takes a more global view of network lifetime. This chapter provides a formulation of the optimization problem in the

form of a quadratic program (QP) ([18]). The quadratic nature of the optimization problem puts restrictions on the scalability of the QP approach. Chapter 6 provides an approximate formulation of the lifetime problem in the form of a dynamic program (DP) ([18]). Experiments show that in WSN as large as 100 nodes there is a visible difference in the rate of reduction of variance and rate of increase in mean of power consumption rates as shorter routes are traded-off in favor of routes that are less efficient in the greedy, shortest-path-first (SPF) sense.

Chapters 7, 8 and 9 describe topological properties of WSNs, particularly the small-world effect. Chapter 7 describes the place small-worlds networks [128] occupy on the spectrum of networks with node connections varying between order and randomness. In this chapter we describe the potential benefits of having wireless networks with small-world topologies and previous approaches to achieve this goal. Small-world networks are characterized by a phase difference between the drops in characteristic path length and clustering coefficient of networks as the fraction of links in the network that are shortcuts is increased. Prior models of characteristic path length and clustering coefficient in various types of graphs did not consider the addition of range limited links. This chapter presents analytical models for both these quantities for graph topologies that are subject to the constraints of wireless networks. The model developed in this chapter can be parameterized to accommodate any of the existing approaches to building small-worlds in WSNs. Chapter 8 describes a method based on single-input multiple-output (SIMO) principles, called the generalized ‘Poor-Man’s-SIMO-System’ (gPMSS), a practical implementation of shortcut links in WSNs by leveraging cooperative communication and diversity combination principles. The diversity combining methods explored include selection diversity, equal gain and maximal-ratio gain diversity combining, where the weight factors used in maximal-ratio combining are provided by the CSI-driven channel model of the BER developed earlier in chapter 3. gPMSS enables the construction of small-

world networks in wireless networks by alternative means. Its chief advantage over existing approaches lies in the fact that it is not based on any modifications to the hardware of sensor motes. Proof of concept is provided by performing detailed analysis on a set of wireless channel packet error traces and also implemented on the Crossbow Imote2 sensor mote. The work on gPMSS is also relevant to the research on the network lifetime problem. Chapter 9 attempts to identify the most central nodes/regions in a wireless network. It highlights the problems demonstrates the shortcoming of pre-existing centrality measures (degree, closeness, betweenness and eigenvector centrality). A new measure of node centrality called principal component centrality (PCC) is proposed which addresses the shortcoming of eigenvector centrality in identifying well connected hubs within graphs characteristic of wireless mobile ad-hoc networks and WSNs. The hubs in a network identified by PCC can be used for the placement of shared network resources, e.g. endpoints of shortcut links.

Finally, chapter 10 concludes this dissertation by summing up our finding.

Chapter 2

Network Channel Capacity of IEEE 802.15.4 Wireless Sensor Networks Under Reachback

2.1 Introduction

Over the past several years different assumptions have been made about the structure and capabilities of wireless sensor networks (WSN) and the devices they are constituted of. In [43] Gupta and Kumar studied the scalability of wireless networks with randomly chosen source destination pairs. Their conclusions offer two solutions to the scalability problem; 1) Design smaller networks, or 2) localize communication by clustering nodes. The idea of a WSN consisting of homogeneous devices gradually gave way to that of a network consisting of a homogeneous mix of nodes with non-uniform device capabilities. In this newly emerged view of WSNs, sensors are grouped into clusters. Each cluster of sensor nodes elects from among its members a clusterhead node (CLH) that acts as a gateway for all incoming and outgoing communications.

Consequently, WSNs can be thought of as hierarchical networks with s levels. However, since most previous work on WSNs distinguishes only between two roles for devices, sensors and CLHs, we assume a two-level hierarchy with $s = 2$. This assumption is also supported by the IEEE 802.15.4 low rate-wireless personal area networks (LR-WPAN) draft standard. The standard refers to a hardware device with more resources with a more complex implementation as a full function device (FFD) and a device with fewer resources and a simpler, less expensive implementation as a reduced function device (RFD). For the remainder of this chapter we will assume a WSN built out of devices compliant with the IEEE 802.15.4 physical layer (PHY)/medium access control (MAC) specifications.

Level 1, the lower level, refers to the network formed by a CLH and its associated sensors. Typically, all devices within a cluster are capable of communicating with each other directly. Therefore, communication between sensors and their CLH are assumed to take place over a single hop. We will use the terms cluster, intra-cluster or cluster-level communication interchangeably to refer to the exchange of messages

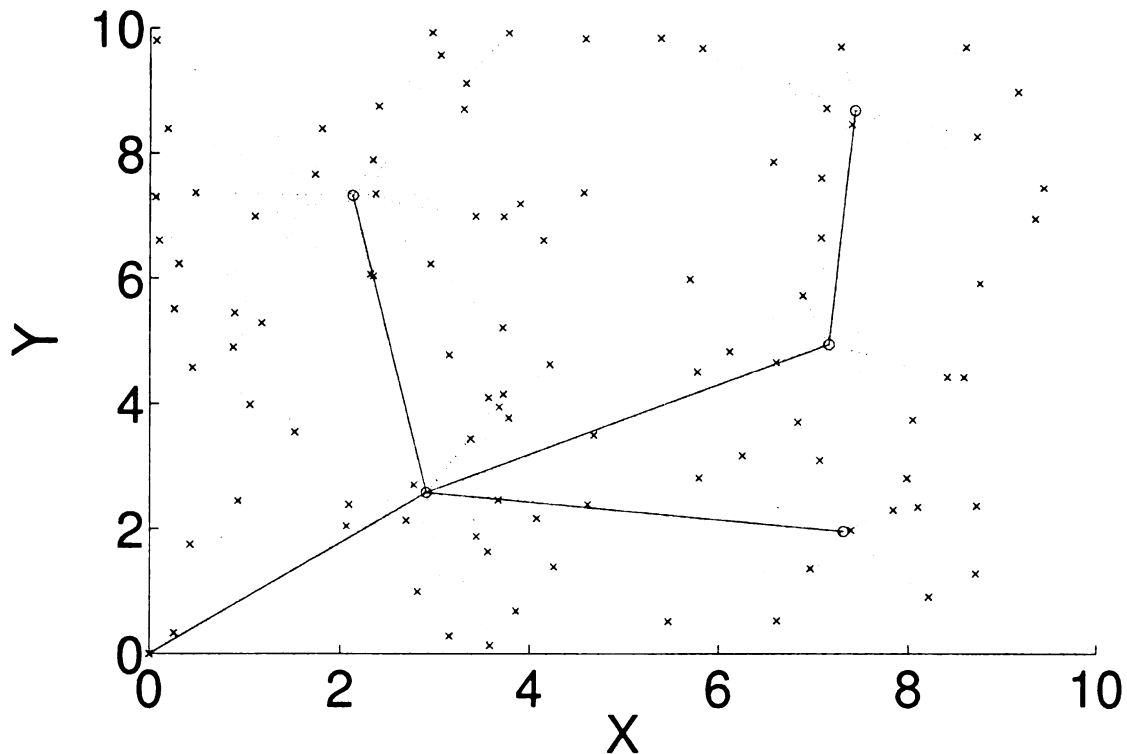


Figure 2.1: Physical layout & network topology of WSN.

between nodes belonging to the same cluster. Communication within a particular cluster proceeds at a common frequency, i.e. there is a potential for interference between transmissions of different sensors of a cluster. Different clusters may or may not use differing frequencies. Moreover, the network topology within a cluster is restricted to a star topology with the CLH at the center.

Level 2, the upper layer, refers to the network formed by the CLHs of all clusters in the WSN and the base station. We will also refer to this network as the overlay network (ON). We assume that the CLHs participating in the ON are capable of routing and relaying their own and other clusters' packets towards the base station. Moreover, since the most widely used WSN routing algorithms like destination-sequenced distance vector (DSDV) [99], directed source routing (DSR) [61], ad-hoc on-demand distance vector (AODV) [100], directed diffusion [58] etc., are different forms of short-

est path routing algorithms, we assume that at any given time the routes from CLHs to base station in the ON form a tree rooted at the base station. We are discounting the possibility of using bifurcated routing, i.e. multiple paths from source to destination. We will use the terms overlay, overlay network or CLH-level communication interchangeably to refer to the exchange of messages between nodes belonging to the ON. Moreover, it is assumed that message exchanges between CLHs in the ON take place at one frequency, i.e. like sensors in a cluster, CLHs in the ON have the potential to produce interference for each other. However, this frequency channel is assumed to be free of interference from cluster-level communication. Moreover, the topology of the ON is a mesh topology.

As 2.1 shows, as traffic generated by the CLHs situated farther away approaches the base station, the expected volume of traffic carried by a link increases. This leads to a capacity bottleneck around the base station that subsequently limits the rate at which CLHs, and ultimately sensors, can inject data into the network. This is called the reachback problem [9]. Thus source coding is used to alleviate the effects of the reachback channel. Besides entropy coding methods, a very popular compression method in WSNs is Slepian-Wolf coding [114] which assumes prior knowledge of the degree of correlation between sensors.

2.2 System Model

2.2.1 Wireless Networking Standards for WSNs

In our attempt to formulate a generalized expression for the end-to-end capacity of a channel between an arbitrary sensor and the base station we remain open to the possibility of a number of different wireless networking technologies. Besides proprietary radio interfaces, the most commonly encountered standardized wireless networking

technologies found in early implementations of WSNs are the IEEE 802.11x wireless local area network (WLAN) standard [51], the IEEE 802.15.1/ Bluetooth standard [4] and the IEEE 802.15.4 standard [5]. The distinguishing features of these three networking standards are many. Table 2.1 lists only the features that we were concerned with in our work, i.e. the operating frequencies and the types of MAC protocols.

As we will show in a later section, our interpretation of how the end-to-end capacity of a wireless channel can be computed requires us to assume a pathloss model to model the physical channel. Pathlosses depend on numerous environmental factors whose effects are generally too complicated to predict. Two parameters that influence pathlosses in a major way are the frequency and transmitter-receiver separation. We assume that all transmissions are taking place in the $2.4 - 2.4835GHz$ industrial scientific and medical (ISM) band which is used in all three wireless networking standards under consideration. This will allow us to use a single pathloss model that will be applicable to all three networking standards under consideration. Therefore, our end-to-end channel capacity expression will not be applicable to WSNs using IEEE 802.15.4 networks operating in the $868 - 868.6MHz$ or $902 - 928MHz$ bands. In addition, we are allowing for both collision avoiding carrier sense medium access/collision avoidance (CSMA/CA) and collision-free time division multiple access (TDMA), type MAC protocols. The choice of the MAC protocol mode will affect parameters in the model.

Let N be the total number of sensors in a WSN of M clusters. Let $C_i \forall i \in \{1, 2, 3, \dots, M\}$ denote an individual cluster consisting of one CLH and N_i sensors (therefore; $N = \sum_{i=1}^M N_i$). Sensors are addressed using a 2-dimensional address scheme in which the i^{th} cluster's j^{th} sensor node is labeled $n_i(j) \forall i \in \{1, 2, 3, \dots, M\}$, $\forall j \in \{1, 2, 3, \dots, N_i\}$, with $n_i(0)$ denoting its CLH, and $n_0(0)$ denoting the base station. We also define a frequency function $f(n_i(j))$ that returns the frequency at which the device passed in the argument communicates within a cluster. This way

Table 2.1: Tabular listing of relevant features of the three wireless networking technologies under consideration for use in WSNs.

<i>Feature</i>	IEEE 802.15.4	IEEE 802.11b	IEEE 802.15.1/ Bluetooth
<i>Frequencies</i>	1. 868-868.6MHz 2. 902-928MHz 3. 2.4-2.4835GHz	2.4-2.4835 GHz	2.4-2.4835 GHz
<i>MACtype</i>	1. TDMA in GTS 2. CSMA/CA in CTS	1. CSMA/CA in DCF 2. Polling in PCF	Polling
<i>Topologies</i>	Meshed Star	Star (DCF)/Meshed Star (PCF)	Star/Meshed Star (Piconet)

$f(n_i(0))$ denotes the intra-cluster communication frequency of cluster C_i . We use $f_i \forall i \in \{1, 2, 3, \dots, M\}$ as a shortened form to denote the cluster frequencies of all M clusters. Similarly, we abbreviate $f(n_0(0))$ by f_0 to denote the frequency used by CLHs for communicating in the ON. The probability of $n_k(l)$ making an interfering transmission at the same time as $n_i(j)$ is making its transmission is denoted by $q_{n_k(l)}(n_i(j))$. We also define a *frequency indicator function* in equation 2.1;

$$I_f(n_i(j), n_k(l)) = \begin{cases} 0; & f(n_i(j)) \neq f(n_k(l)) \\ 1; & f(n_i(j)) = f(n_k(l)) \end{cases} \quad (2.1)$$

With the definitions of symbols describing the network in place we now come to the terms that will quantize the channel conditions, i.e. capacity and probability of error terms. First off we define the probability of a bit error or the bit error rate (BER) denoted by p . The 802.15.4 standard uses binary phase shift keying (BPSK) modulation at 20 and 40kbps and offset-quaternary phase shift keying (O-QPSK) modulation at 250kbps. Both BPSK and QPSK modulators make hard decisions based on the received symbol and output either a '0' or a '1'. Therefore, assuming an additive white Gaussian noise (AWGN) channel both BPSK and QPSK receivers can be modeled by binary symmetric channel (BSC) with probability of error p . p is defined as the fraction of transmitted bits that are received without errors. Hence, this term models the channel between the Physical layers of the OSI model transmitter and receiver.

$$p = \frac{\# \text{ of bits received without errors}}{\# \text{ of bits transmitted}} \quad (2.2)$$

Frames in 802.15.4 are protected only by a 16 bit frame check sequence (FCS) with no error correction capability. This implies that whenever even a single bit in an entire frame is received in error the receiver discards the entire frame. Packetization

affects net throughput. We denote the fraction of transmitted bits that are received without errors by δ . δ will be referred to as the packet error rate (PER). Since packets are either received correctly and forwarded to the next higher layer or discarded due to FCS failure, the PER is equivalent to the probability of error δ of a binary erasure channel (BEC). A BEC is used to model the channel between the MAC Layers of the OSI model transmitter and receiver.

$$\delta = \frac{\# \text{ of packets received without errors}}{\# \text{ of packets transmitted}}. \quad (2.3)$$

Finally, since we are assuming the use of Slepian-Wolf coding we have take into account the dependencies that introduced in data streams. As a result of these dependencies packets that will be received error free will at times not be decodable due to losses in transmissions from other sources. As a result, packets that are received free of errors can end up being undecodable due errors in another transmission it depends on. Therefore, we define the PER with Slepian-Wolf coding as the fraction of transmitted packets that are decodable by the receiver denoted by δ_{DEC} .

$$\delta_{DEC} = \frac{\# \text{ of decodable packets received}}{\# \text{ of packets transmitted}}. \quad (2.4)$$

p , δ and δ_{DEC} for links within a cluster are denoted by p_{CL} , δ_{CL} and δ_{DEC-CL} . Similarly, the corresponding quantities for links on the ON are abbreviated by p_{ON} , δ_{ON} and δ_{DEC-ON} . Since, the ON channel between an arbitrary CLH and the base station may comprise of multiple hops we need to define additional quantities for the end-to-end channel in the ON p_{ON-E2E} , δ_{ON-E2E} and $\delta_{DEC-ON-E2E}$. Finally, the channel formed between each individual sensor and the base station is characterized by a combination of the above computed values and denoted by $p_{ON-S2BS}$, $\delta_{ON-S2BS}$ and $\delta_{DEC-ON-S2BS}$. To clarify, figure 2.2 shows the relationship between δ_{DEC-CL} , δ_{DEC-ON} , $\delta_{DEC-ON-E2E}$

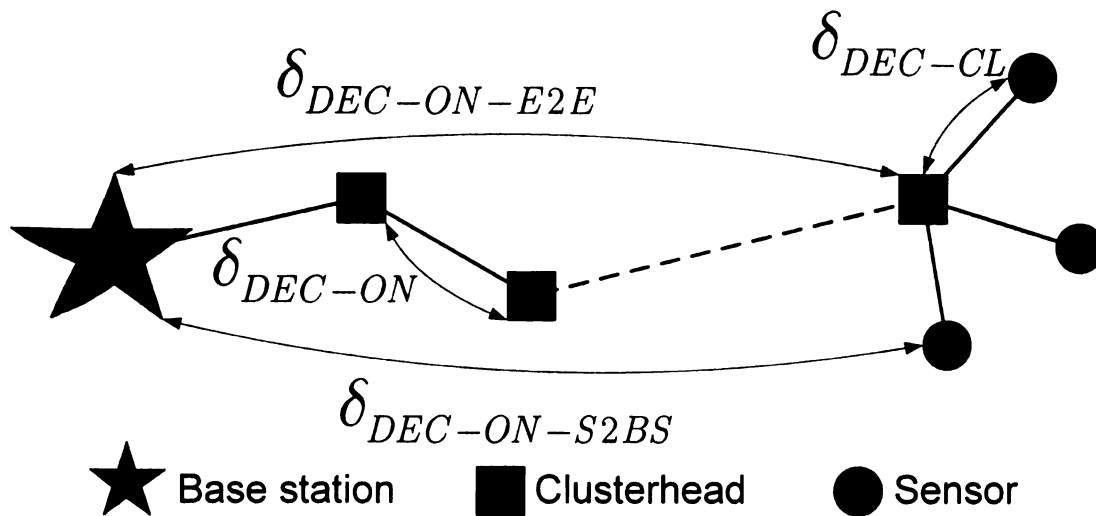


Figure 2.2: Relationship between δ_{DEC-CL} , δ_{DEC-ON} , $\delta_{DEC-ON-E2E}$ and $\delta_{DEC-ON-S2BS}$.

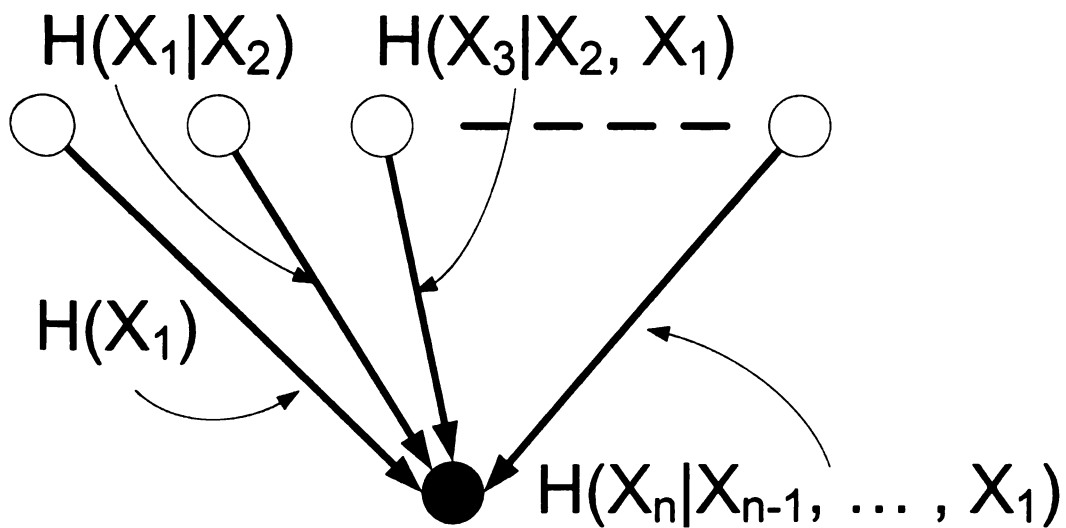


Figure 2.3: Slepian-Wolf coding in cluster-level communication.

and $\delta_{DEC-ON-S2BS}$.

2.2.2 Source & Channel Coding

We consider three options for source coding; 1) No source coding, 2) *opportunistic compression* [71], and 3) *Slepian-Wolf coding*. The first option does not perform any compression at all. Data is merely collected at each CLH, concatenated, packetized and routed to the base station. Opportunistic Compression assumes correlation between different sensor readings. Each CLH compresses data it receives from downstream based on other data available to it from other sources. Slepian-Wolf coding was first proposed by Slepian and Wolf in [114]. However, we make some simplifying assumptions about Slepian-Wolf coding as it applies to WSNs. These assumptions have been taken from Marco and Neuhoff in [80] and [81]. Slepian-Wolf coding also exploits the spatial correlation of sensor readings in neighboring sensors and is employed as a means to compress data before transmitting it to the base station. Slepian-Wolf coding is based on equation 2.5. However, Slepian-Wolf assumes that the degree of correlation is known prior to transmission, whereas opportunistic compression assumes no prior knowledge. Consider a cluster i as in figure 2.3 consisting of N_i sensors and a CLH in which the first sensor $n_i(1)$ produces reading $X_{n_i(1)}$, the second sensor $n_i(2)$ produces reading $X_{n_i(2)}$ and so on.

$$\begin{aligned} & \left[H(X_{n_i(1)}) \right] + \left[H(X_{n_i(2)} | X_{n_i(1)}) \right] + \dots + \left[H(X_{n_i(N_i)} | X_{n_i(N_i-1)} \dots X_{n_i(1)}) \right] \\ & \geq \left[H(X_{n_i(N_i)} \dots X_{n_i(1)}) \right] \end{aligned} \quad (2.5)$$

The size of all N_i sensor readings after lossless compression is lower-bounded by their joint entropy. As an example, Slepian-Wolf coding within a cluster proceeds as follows.

1. Sensors transmit their readings to their CLH. For the purpose of simplifying

analytical expressions, let us assume that the transmissions to the CLH are scheduled in the ascending order of sensor nodes' ID numbers.

2. The first transmission $X_{n_i(1)}$ by $n_i(1)$ is not compressed (see first term on left-hand side of (1)).
3. The second sensor $n_i(2)$ compresses its sensor reading to $H(X_{n_i(2)}|X_{n_i(1)})$ based on the side information of $n_i(1)$'s transmission of size $H(X_{n_i(1)})$. Therefore, the j^{th} node in cluster i transmits its data as $H(X_{n_i(j)}|X_{n_i(j-1)} \dots X_{n_i(1)})$ bits. This way the total volume of all transmissions approaches the joint entropy as shown in (1) and also depicted in figure 2.3.

However, this coding scheme has one major disadvantage. While the savings in transmissions can be substantial, depending on the spatial distribution of the phenomena being sensed, the failure of the CLH to receive the k^{th} transmission results in its inability to reconstruct all subsequent transmissions $k + 1$ through N_i for that round. The high sensitivity of a receiver's ability to decode packets to packet losses, and the fact that bit and packet error rates in single hop wireless networks are orders of magnitude higher than those in multi-hop wired networks makes the use of MAC layer channel codes imperative. Even without any dependencies between data streams the BERs of end-to-end channels representative of multi-hop paths in wireless networks can often times be large enough to impose unrealistically large overheads on end-to-end channel codes.

The PERs of various links will depend on the size of packets. When Opportunistic Compression or Slepian-Wolf coding are employed that will depend on the degree of correlation in data readings. To model the lossless compression of data and determine the message sizes $L_m(n_i(j))$ we use the model proposed by Patten, Krishnamachari and Govindan in [97]. The use of Slepian-Wolf coding forces an order on the devices in the WSN which is also described in detail in [97]. The number assigned to a particular

device $n_i(j)$ in this order is returned by the order function $O(\cdot)$ which returns a value in the range $1 \leq O(n_i(j)) \leq N$.

2.2.3 Power Consumption Model

The total energy $E(n_i(j))$ consumed in mote $n_i(j)$ in equation 2.6 is modeled as the sum of energies consumed in communication $E_{TX}(n_i(j))$ and processing $E_{PR}(n_i(j))$.

$$E(n_i(j)) = E_{COMM}(n_i(j)) + E_{PR}(n_i(j)). \quad (2.6)$$

$E_{COMM}(n_i(j))$ in equation 2.7 can be further broken down into energy consumed in transmission $E_{TX}(n_i(j))$ and reception $E_{RX}(n_i(j))$.

$$E_{COMM}(n_i(j)) = E_{TX}(n_i(j)) + E_{RX}(n_i(j)). \quad (2.7)$$

The total energy spent by all devices in a network divided by the uncompressed number of bits of data communicated to the base station yields the *energy per symbol* (EPS) of the network configuration in equation 2.8.

$$EPS = \frac{\sum_{i=1}^M \sum_{j=1}^{N_i} E_{n_i(j)}}{\sum_{i=1}^M \sum_{j=1}^{N_i} H(X_{n_i(j)})}. \quad (2.8)$$

The processing cost incorporates the cost of performing source and channel coding and decoding operations encoding and decoding. Since several algorithms of varying degrees of complexity are available for both source and channel coding we model the complexity of these operations at the transmitter and receiver by terms $c_1 L_m^{\alpha_1}$ and $c_2 L_m^{\alpha_2}$, respectively. The cost is dependent on the message size L_m on which the operations are performed.

2.3 Channel Model

In this section we derive a channel model for the end-to-end channel between a transmitter and receiver communicating in a multi-hop 802.15.4 wireless network. The first step consists of identifying a suitable pathloss model for the $2.4 - 2.4835GHz$ ISM band. We use the model put forward by the physical channel modeling subgroup of the IEEE 802.15 taskgroup 4 in [86] for this purpose. Figure 2.4 depicts a transmitter and receiver communicating over a multi-hop wireless channel. The transmitter and receiver are depicted by the two protocol stacks of the *Open Systems Interconnect* (OSI) model. At the Physical layer of the OSI model we assume a *discrete memoryless channel* (DMC). The pathloss model is an abstraction of the physical channel between a single transmitter and receiver. In terms of the OSI model, the pathloss model represents an abstraction of everything that falls under the Physical layer. From the pathloss model we can determine a bit error model that will be representative of everything below the data-link layer. The ultimate goal here is to determine a model that is capable of abstracting everything down from layer 5 of the OSI model. In the next step, we add another layer of abstraction to the bit error model by using it to obtain a packet error model. The packet error model is an abstraction of the network stack from the Network layer down. Note that the packet error model can serve as an end-to-end model in single-hop networks. However, for multi-hop networks such as the ones we are considering we will have to modify the packet error model to obtain the desirable end-to-end packet error model that will abstract everything from the session layer down. This successive abstraction of the channel between transmitter and receiver based on the next lower model is depicted in figure 2.4. The layers of the OSI model inside a dashed box represent the layers encapsulated in the corresponding model.

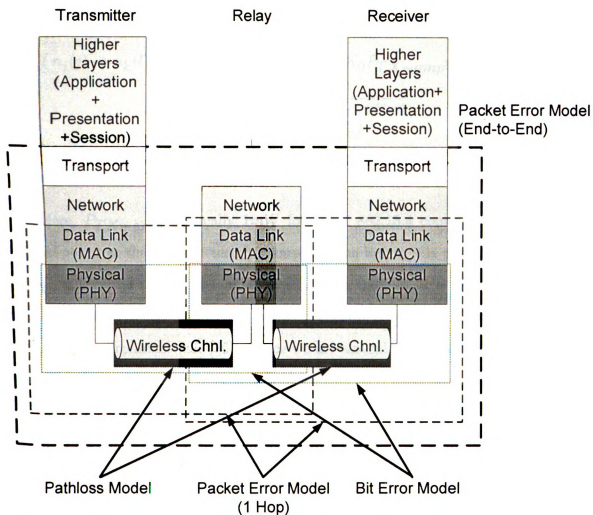


Figure 2.4: End-to-End channel between a transmitter and receiver on a multihop wireless network.

2.3.1 1-Hop Communication

The expressions we derive in this section are equally applicable to all 1-hop communication links in the WSN, irrespective of whether they are in the ON or cluster. Therefore, we will refrain from using the CL and ON at the end of the subscripts of the sought quantities, i.e. received signal power, SINR, BER and PER. From the Pathloss model in [86] we can obtain the expression in equation 2.9 to obtain the Power $P(n_i(j) \rightarrow n_k(l))$ of a signal transmitted by $n_i(j)$ at receiver $n_k(l)$.

$$\begin{aligned}
P(n_i(j) \rightarrow n_k(l)) = & I_f(f(n_i(j)), f(n_k(l))) K_0 P_{TX-amp} \eta_{TX-ant} \eta_{RX-ant} \\
& \times \frac{PL_0}{\left(\frac{d(n_i(j), n_k(l))}{d_0}\right)^2 \left(\frac{f}{f_c}\right)^{2K+2}}
\end{aligned} \tag{2.9}$$

Here, P_{TX-amp} (typically $1mW$ for IEEE 802.15.4 compliant devices) is the signal power at the transmitter after amplification before it is passed to the antenna, η_{TX-ant} is the transmitter antenna efficiency, η_{RX-ant} is the receiver antenna efficiency and K , PL_0 , and K_0 are environmental parameters that depends on the operating environment (see [86]). Since we are assuming that a WSN consists of devices with identical radio interfaces transmitting at a fixed power we consider these terms to be constant across the entire network. We also define two reference parameters f_c and d_0 for this pathloss model. For the sets of parameters provided in [86], f_c is set to $5GHz$ and d_0 is set to $1m$. This leaves the expression dependent on the transmission frequency f and the distance between devices $n_i(j)$ and $n_k(l)$ that is returned by the distance function $d(n_i(j), n_k(l))$.

In order to obtain the BER for a DMC we need an expression for the *signal-to-interference & noise ratio* (SINR). The general expression for the SINR is given in equation 2.10. Using the expression for the pathloss model described above we can arrive at an expression for the SINR in the terms of known quantities.

$$SINR = \frac{P_{TX}}{P_A + \sum P_{int}}, \tag{2.10}$$

where P_{TX} is the power of the received signal for which the SINR is being computed and P_{int} is the power of interfering signals caused by undesired concurrent transmissions elsewhere in the network. P_A is the ambient noise power of interfer-

ence produced by sources that are not part of the WSN and are not modeled by any terms in P_{int} . Sources of ambient noise power may include but are not limited to other networks that are co-located with the WSN under consideration [136] or devices or appliances (microwave ovens, cordless phones) that operate in the same frequency band. Since at the cluster-level all transmissions are directed from sensors to their respective CLH, the term $SINR(n_i(j) \rightarrow n_k(l))$ represents the SINR of the transmission from $n_i(j)$ to $n_k(l)$. In the general case, all sensors and CLHs can be considered potential sources of interference. Obviously, we assign $q_{n_i(j)}(n_i(j)) = 0$. The interference power $\sum P_{int}$ in equation 2.10 is the sum of all other signals that are received originating from sources other than $n_i(j)$. This term is computed by summing the interference power of all sensors. In order for $n_k(l)$ to contribute to the power of the interference signal for a transmission originating from $n_i(j)$, $n_k(l)$ must be 1) operating in the same frequency band and 2) transmitting at the same time as $n_i(j)$. Transmitters operating at different frequencies make effectively no contribution to interference. Therefore, the indicator function defined earlier is used. The probability that a sensor will produce an interfering signal at the same time as $n_i(j)$ transmits is accounted for by multiplying the term further by $q_{n_k(l)}(n_i(j))$.

$$\begin{aligned}
 & SINR(n_i(j) \rightarrow n_k(l)) \\
 &= \frac{I_f(n_i(j), n_k(l))P(n_i(j) \rightarrow n_k(l))}{P_A + \sum_{m=1}^M \sum_{n=0}^{N_k} I_f(n_i(j), f(n_m(n)))q_{n_m(n)}(n_i(j))P(n_m(n) \rightarrow n_i(0))}.
 \end{aligned} \tag{2.11}$$

Then equation 2.11 is the final expression for $SINR(n_i(j) \rightarrow n_k(l))$. In the next step we use the SINR and *Q-function* to obtain the probability of error for a BSC. The use of the Q-function to arrive at the BER using the SINR is described

by Rappaport in [104]. If $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u}{2}} du$, then the probability of receiving a bit error at $n_k(l)$ in a transmission originating at $n_i(j)$ is called the *bit error rate* (BER) $p(n_i(j) \rightarrow n_k(l))$. IEEE 802.15.4 uses two different modulation techniques. 802.15.4 uses Binary Phase-Shift Keying (BPSK) at the 20 and 40kbps data rates in the 868 – 868.6MHz and 902 – 928MHz bands, respectively. At 250kbps in the 2.400 – 2.4835GHz band it uses offset-quadrature phase-shift keying (O-QPSK). In either case the BER is obtained by using the Q-function. Equation 2.12 shows the relationship between SINR and BER for a QPSK receiver. Equation 2.13 is the corresponding equation for a BPSK receiver. This can be thought of as the probability of error in a binary symmetric channel (BSC). The corresponding channel capacity in terms of the BER is obtained from equation 2.14, where $H_b(\cdot)$ is a function that returns the entropy of a Bernoulli random variable with the parameter provided in the argument.

$$\begin{aligned} p(n_i(j) \rightarrow n_k(l)) &= Q\left(\sqrt{2SINR(n_i(j) \rightarrow n_k(l))}\right) \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2SINR(n_i(j) \rightarrow n_k(l))}}^{\infty} e^{-\frac{u}{2}} du. \end{aligned} \quad (2.12)$$

$$\begin{aligned} p(n_i(j) \rightarrow n_k(l)) &= Q\left(\sqrt{SINR(n_i(j) \rightarrow n_k(l))}\right) \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{SINR(n_i(j) \rightarrow n_k(l))}}^{\infty} e^{-\frac{u}{2}} du. \end{aligned} \quad (2.13)$$

$$C_{BER}(n_i(j) \rightarrow n_k(l)) = 1 - H_b(n_i(j) \rightarrow n_k(l)) \quad (2.14)$$

From the BER we now determine expression 2.17 for obtaining the probability of a packet loss or the *packet error rate* (PER) $\delta(n_i(j) \rightarrow n_k(l))$ for a transmission from $n_i(j)$ to $n_k(l)$. The PER corresponds to the probability of error of a binary erasure channel (BEC). We assume that without channel coding a received packet is discarded if a single bit is in error, a valid assumption considering our choice of wireless

standards. L_h is the per sample header size in bits. The size of the sample making up the message depends, of course, on the entropy reduction method employed. When no source coding is used $L_m(n_s(t))$ is simply $\left[H\left(X_{n_s(t)}\right) \right]$, where $H(\cdot)$ is the entropy function. When opportunistic compression is used the size of a message originating at $n_s(t)$ is $L_m(n_s(t))$ is $H\left[H\left(X_{n_s(t)}|X_{n_s(1)}, \dots, X_{n_s(t-1)}\right) \right]$. From 2.15 we trivially obtain expression 2.16 for the channel capacity $C_{PER}(n_i(j) \rightarrow n_k(l))$ in terms of $\delta(n_i(j) \rightarrow n_k(l))$. Note that for the case of Slepian-Wolf coding, 2.15 and 2.16 do not yet take into account the additional losses due to a receiver's inability to decode a received message caused by losses in transmissions of other messages the received message is dependent on.

$$\delta(n_i(j) \rightarrow n_k(l)) = 1 - [1 - p(n_i(j) \rightarrow n_k(l))]^{L_h + [L_m(n_s(t))]} \quad (2.15)$$

$$C_{PER}(n_i(j) \rightarrow n_k(l)) = 1 - p(n_i(j) \rightarrow n_k(l)) \quad (2.16)$$

Now we consider the case where Slepian-Wolf coding is employed and a further expression for the PER of a message can be obtained which takes into account losses in streams that the receiver depends on to decode the message. Since cluster-level communication is single hop only and the ON is essentially a multi-hop network the expressions for δ_{DEC-CL} and δ_{DEC-ON} are significantly different and we will depart from our practice of finding general expressions that we followed to this point.

$$\delta_{DEC-CL}(n_i(j) \rightarrow n_i(0)) = 1 - [1 - p(n_i(j) \rightarrow n_i(0))]^{L_h + L_m} \quad (2.17)$$

where,

$$Lm(n_i(j)) = \left[H(X_{n_i(j)}|A) \right]$$

$$A = X_A, \forall a : 1 \leq a \leq O(n_i(j)) \quad (2.18)$$

$$C_{PER-SW-CL}(n_i(j) \rightarrow n_i(0)) = 1 - \delta_{DEC-CL}(n_i(j) \rightarrow n_i(0))$$

2.3.2 Overlay Network Communication

We now turn our attention to the end-to-end capacity of the channel between an arbitrary CLH and the base station communicating over a multi-hop wireless network. The case in which all CLHs are directly communicating with the base station in the ON becomes a special case of the more general case of a multi-hop ON. Like for the 1-hop case, we start from the expression for SINR, this time for a signal transmitted by a CLH $n_i(0)$ to its upstream neighbor. This is given in 2.19.

Before proceeding further we define a set of new functions that will subsequently be used in this section. $R^{1\uparrow}(n_i(0))$ returns the immediate upstream neighbor of CLH $n_i(0)$, where upstream denotes the direction towards the base station in the network topology. $R^{1\downarrow}(n_i(0))$ returns the set of CLHs that is 1 hop downstream from $n_i(0)$, where the term downstream refers to the direction away from the base station in the network topology. $R^\uparrow(n_i(0))$ returns the set of all CLHs that are upstream from $n_i(0)$. Similarly, $R^\downarrow(n_i(0))$ returns the set of all CLHs that are downstream from $n_i(0)$.

$$SINR_{ON}(n_i(0)) = \frac{P(n_i(0) \rightarrow R^{1\uparrow}(n_k(0)))}{P_{ON-A} + \sum_{\substack{k=1 \\ k \neq 1}}^M q_{n_k(0)}(n_i(0)) P(n_k(0) \rightarrow R^{1\uparrow}(n_k(0)))}. \quad (2.19)$$

Note that for a TDMA protocol in the ON $q_{n_k(0)} = 0 \forall k$, and hence 2.19 simplifies to 2.20.

$$SINR_{ON}(n_i(0)) = \frac{P(n_i(0) \rightarrow R^{1\uparrow}(n_k(0)))}{P_{ON-A}}. \quad (2.20)$$

Applying the Q-function leads us to similar expressions 2.21 and 2.22 for the BER as before.

$$\begin{aligned} p_{ON}(n_i(0)) &= Q\left(\sqrt{2SINR_{ON}(n_i(0))}\right) \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2SINR_{ON}(n_i(0))}}^{\infty} e^{-\frac{u}{2}} du \end{aligned} \quad (2.21)$$

$$\begin{aligned} p_{ON}(n_i(0)) &= Q\left(\sqrt{SINR_{ON}(n_i(0))}\right) \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{SINR_{ON}(n_i(0))}}^{\infty} e^{-\frac{u}{2}} du \end{aligned} \quad (2.22)$$

From 2.21 and 2.22 we can obtain equation 2.23 a recursive definition for the BER of the multi-hop, end-to-end channel between a CLH and the base station.

$$\begin{aligned} p_{ON-E2E}(n_i(0)) &= p_{ON}(n_i(0)) \left[1 - p_{ON-E2E}(R^{1\uparrow}(n_i(0)))\right] \\ &\quad + p_{ON-E2E}(R^{1\uparrow}(n_i(0))) \left[1 - p_{ON}(n_i(0))\right]. \end{aligned} \quad (2.23)$$

The expression for the PER for a packet originating at $n_i(0)$ on a link between $n_k(0)$ and its upstream neighbor $R^{1\uparrow}(n_k(0))$ is then provided by equation 2.24.

$$\delta_{ON-n_k(0)}(n_i(0)) = 1 - (1 - p_{ON}(n_k(0)))^{L_h + L_m(n_i(0))}, \quad (2.24)$$

where $\delta_{ON-n_k(0)}(n_j(0)) = 1$ if $n_k(0) \notin R^\uparrow(n_j(0))$ and the message size originating from $n_i(0)$ is,

$$L_m(n_i(0)) = \sum_{i=0}^{N_i} H(X_i). \quad (2.25)$$

This leads us to the final expression 2.26 for the end-to-end PER from an arbitrary CLH $n_i(0)$ to the base station. Equation 2.27 gives the corresponding expression for the end-to-end capacity.

$$\delta_{ON-E2E}(n_i(0)) = 1 - \prod_{n_k(0) \in R^\uparrow(n_i(0))} \left(1 - \delta_{ON-n_k(0)}(n_i(0)) \right) \quad (2.26)$$

$$C_{PER-ON-E2E}(n_i(0)) = 1 - \delta_{ON-E2E}(n_i(0)) \quad (2.27)$$

For Opportunistic compression and Slepian-Wolf coding the derivations for expressions in equations 2.19 through 2.27 proceed in exactly the same manner, equation 2.25 being the exception. Opportunistic compression and Slepian-Wolf coding replace equation 2.25 by equation 2.28 and 2.30, respectively.

$$L_m(n_i(0)) = \left[H(X_{n_i(0)} \cdots X_{n_i(N_i)} | A) \right] \quad (2.28)$$

$$A = \{\forall X_a : a \in R^\downarrow(n_i(0))\} \quad (2.29)$$

$$L_m(n_i(0)) = \sum_{a \in R^\downarrow(n_i(0))} L_m(a) \quad (2.30)$$

2.3.3 Sensor-To-Base Station Capacity

Using the above results the sensor-to-base station BER/ PER for any arbitrary sensor $n_i(j)$ can be obtained by multiplying the cluster-level BER $p(n_i(j))$ in 2.12/ PER $\delta(n_i(j))$ in 2.17 with the end-to-end BER $p_{ON-E2E}(n_i(0))$ in 2.23/ PER $\delta_{ON-E2E}(n_i(0))$ in equation 2.26, respectively. This yields an expression for the sensor-to-base station BER in equation 2.31 and a corresponding sensor-to-base station PER expression in equation 2.32. From 2.31 and 2.32 we can obtain the corresponding capacity expressions in equations 2.33 and 2.34.

$$\begin{aligned} p_{S2BS}(n_i(j)) \\ = p_{ON-E2E}(n_i(0)) [1 - p(n_i(j))] + p(n_i(j)) [1 - p_{ON-E2E}(n_i(0))] \end{aligned} \quad (2.31)$$

$$\delta_{S2BS}(n_i(j)) = 1 - [1 - \delta(n_i(j))] [1 - \delta_{ON-E2E}(n_i(0))] \quad (2.32)$$

$$C_{BER-S2BS}(n_i(j)) = 1 - H_b(p_{S2BS}(n_i(j))) \quad (2.33)$$

$$\begin{aligned} C_{PER-S2BS}(n_i(j)) &= 1 - \delta(n_i(j)) \\ &= [1 - \delta(n_i(j))] [1 - \delta_{ON-E2E}(n_i(0))] \end{aligned} \quad (2.34)$$

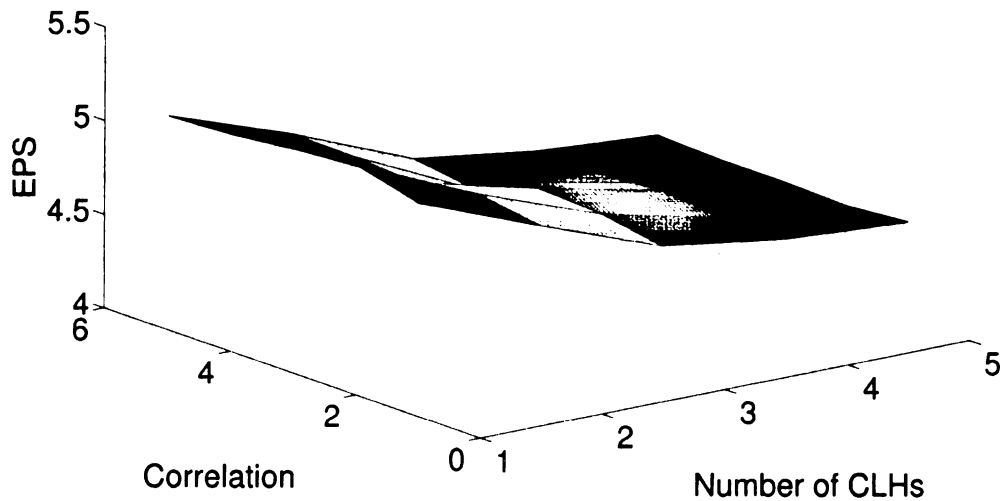
2.4 Results and Analysis

For the following experiments we use an 802.15.4 network with its MAC operating in TDMA/ GTS enabled mode. We use the PHY channel model for residential environments in [86]. The WSN consists of $N = 150$ sensor nodes randomly placed over a square region of dimensions 10×10 according to a uniform random distribution. To create longer multi-hop routes in the ON we place the base station at coordinates $(0,0)$. We assume a set of 15 available frequencies for cluster-level communication in addition to one frequency reserved for communication between CLHs in the ON. Figure 2.1 depicts a sample WSN. Circles denote the positions of CLHs while the x-marks denote the positions of sensors. Sensors in closest proximity of the means obtained by the k-means clustering algorithm [44] are assigned the role of the CLH for that cluster. Solid lines depict the topology of the ON while broken lines indicate a sensor's association with its respective CLH.

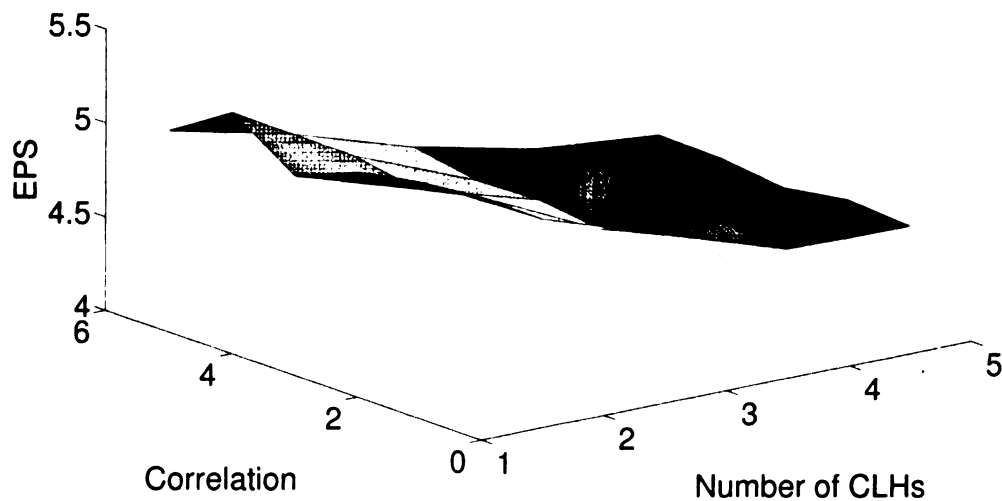
Values for the correlation factor ρ are varied through $\rho = 0.1, 1, 10, 20, 30$. Increase in ρ can alternatively be viewed as an increase in the node density. The number of CLHs M is also varied through $M = 5, 10, 20, 30, 40$. We assume the use of near capacity achieving channel codes at the MAC layer and evaluate the EPS for the cases when,

1. No source coding,
2. Opportunistic compression, and
3. Slepian-Wolf coding are employed.

For each of these cases we assume two sets of complexity vectors $c_1, \alpha_1, c_2, \alpha_2$, 1) $\{5, 1, 10, 1\}$, and 2) $\{5, 2, 10, 2\}$. The constant coefficient at the receiver side c_2 is assigned a higher value than the corresponding transmitter side coefficient c_1 because channel decoding is typically more complex than encoding. This produces six options



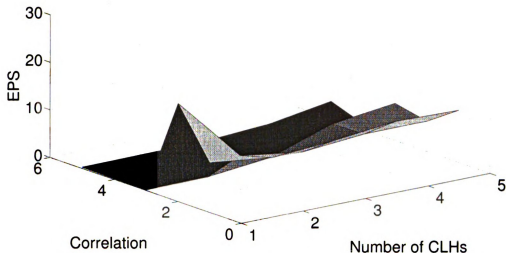
(a) No source coding $\alpha_1 = 1$, $\alpha_2 = 1$, $c_1 = 5$, and $c_2 = 10$.



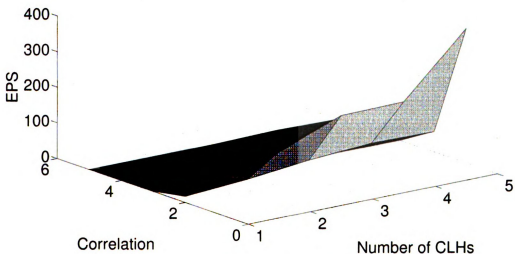
(b) No source coding $\alpha_1 = 2$, $\alpha_2 = 2$, $c_1 = 5$, and $c_2 = 10$.

Figure 2.5: *EPS* spent across the network for varying degrees of spatial correlation and number of CLHs.

to evaluate. Obviously, a low is desirable. Furthermore, based on the measurements reported by Polastre, Szewczyk and Culler in [101] and Madden, Franklin, Hellerstein and Hong in [79] we take the ratio of per bit energy consumption for transmission,



(a) Opportunistic compression $\alpha_1 = 1$, $\alpha_2 = 1$, $c_1 = 5$, and $c_2 = 10$.

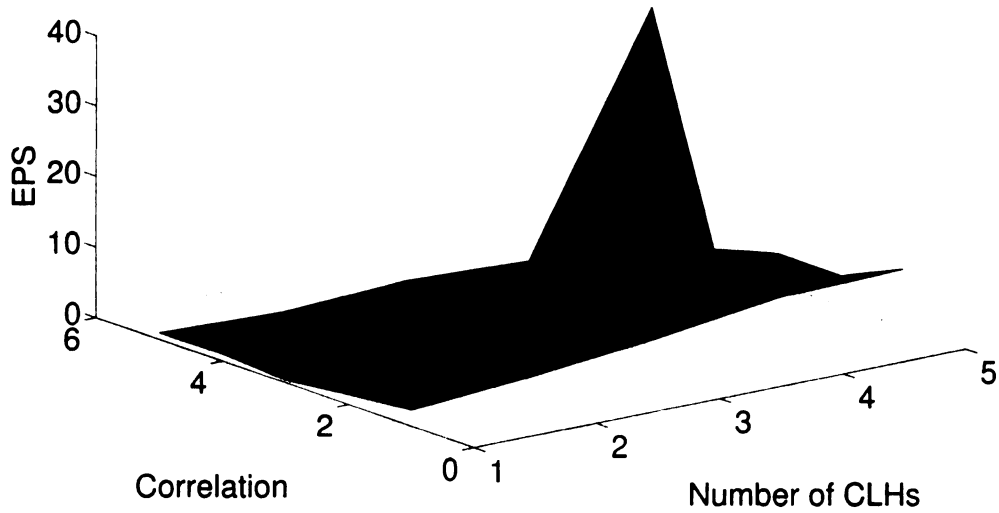


(b) Opportunistic compression $\alpha_1 = 2$, $\alpha_2 = 2$, $c_1 = 5$, and $c_2 = 10$.

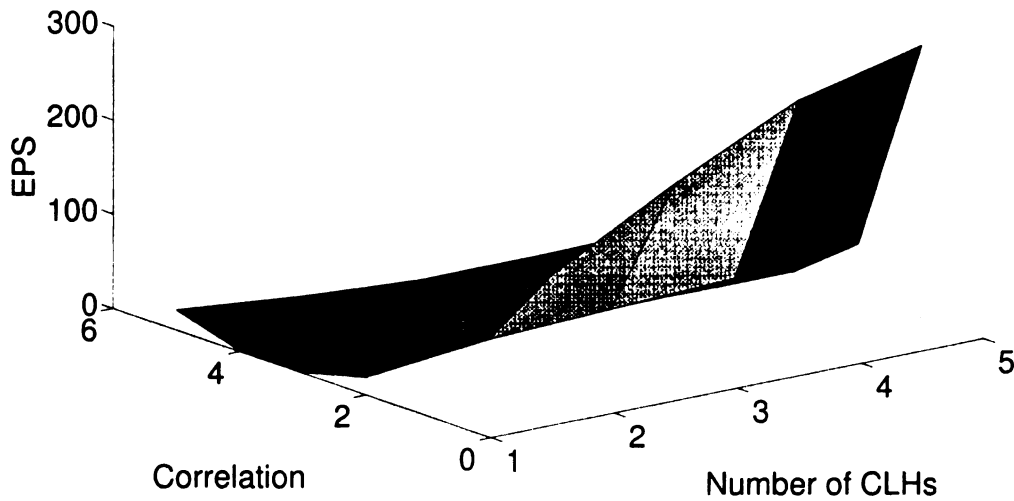
Figure 2.6: *EPS* spent across the network for varying degrees of spatial correlation and number of CLHs.

reception, and instruction execution to be $1 : 0.6 : 1/800$.

Figures 2.5a, 2.5b, 2.6a, 2.6b, 2.7a and 2.7b plot *EPS* against the degree of correlation and the number of CLHs. Figures 2.5a and 2.5b show a decrease in the *EPS*



(a) Slepian-Wolf coding $\alpha_1 = 1$, $\alpha_2 = 1$, $c_1 = 5$, and $c_2 = 10$.



(b) Slepian-Wolf coding $\alpha_1 = 2$, $\alpha_2 = 2$, $c_1 = 5$, and $c_2 = 10$.

Figure 2.7: *EPS* spent across the network for varying degrees of spatial correlation and number of CLHs.

with increasing number of CLHs and no variation with increasing values of ρ . The second observation is according to expectations because we are not using any source coding to leverage the correlation in the data. Figure 2.6a, 2.6b, 2.7a and 2.7b ex-

hibit similar trends but on very different scales. *EPS* increases with increasing M and decreases with increasing ρ . The greater exploitation of the inherent correlation in the data means a smaller scale of the EPS for figure 2.7a and 2.7b relative to figures 2.6a and 2.6b. The sudden increase at high M and ρ can be attributed to outlier values. These results lead us to conclude that weakly correlated data favors the omission of any source coding scheme and a higher number of M . For highly correlated data on the other hand Opportunistic Compression and Slepian-Wolf coding are the better choice. At high values of ρ the value of M plays an increasingly negligible role. The choice between Opportunistic compression and Slepian-Wolf will depend on the complexity of the available implementations of the two, i.e. at similar complexities Slepian-Wolf outperforms Opportunistic compression but in a comparison between Slepian-Wolf with complexity vector $\{5, 2, 10, 2\}$ with Opportunistic compression with complexity $\{5, 1, 10, 1\}$ the latter outperforms the former.

Chapter 3

CSI Driven Model of BER Process on IEEE 802.15.4 Links

3.1 Introduction

The bit errors and packet losses that are observed at the wireless receiver's medium access control (MAC) layer are modeled by a random process that is commonly referred to as the *error process*. An understanding of the error process is of fundamental importance for a wide variety of reasons, e.g. design of high level (network layer and above) protocols, retransmission strategies, error correction and concealment strategies etc.

The IEEE 802.15.4 low rate-wireless personal area network (LR-WPAN) standard [5] is of particular interest to the wireless sensor network (WSN) research community because it is the first wireless communication standard built around devices with severe constraints on power consumption rates. Thus it is widely anticipated that IEEE 802.15.4 will play a major role in WSN applications. This chapter analyzes the performance and contributes to the understanding of IEEE 802.15.4 based LR-WPANs.

The objective of this empirical study is it to gain better insight into the time varying error process. We begin our analysis with a correlation analysis of the bit error rate (BER), link quality indication (LQI) and received signal strength indication (RSSI) processes and establish that if a packet's BER is known to be non-zero, i.e. it has failed the cyclic redundancy check (CRC) test, it is correlated with its LQI and RSSI measurements. Based on the knowledge and the empirical data set we come up with a model for the BER's probability density function (PDF) driven by channel state information (CSI) measurements. We evaluate the utility of this model in different environments by dividing the data set along lines of different collections environments, generating models for each of them and, using traces from other environments as test data, measuring divergence between models and test data.

We further analyze the amount of memory in IEEE 802.15.4 LR-WPAN links at the

packet, symbol and bit level.

3.1.1 Previous Work

To develop a better understanding of wireless channels' error and loss processes, several recent data trace collection efforts have targeted a variety of wireless networks, including 3G networks [70], WaveLAN and 802.11x WLANs [93],[67],[66],[133],[105], CC1100 based MICA2 networks [137] and 802.15.4 LR-WPANs [77],[116],[115],[64]. All of these efforts involve the collection of received data while offering different levels of insight and resolution (e.g., bit-, byte-, and/or packet-level) into the error process on wireless channels. These studies usually focus on what is referred to as the *residual error* process. In general, residual errors are bit-level or packet-level errors that are not corrected by the PHY-layer and hence appear at the MAC-layer. Such errors (usually) cause packet drops in traditional wireless MAC protocols such as IEEE 802.11 and IEEE 802.15.4. Most the error trace collection efforts restrict the error process resolution to packet-level information. The most that can be extracted from packet-level traces are statistics such as

- Packet reception rates (PRR), the fraction of transmitted packets that are received error-free and pass the CRC.
- Packet error rates (PER), the fraction of transmitted packets that are received with at least one bit in error and fail the CRC test.
- Packet loss rates (PLR), the fraction of transmitted packets that are not received at all.

By our definition of these terms.

$$PRR = 1 - PER - PLR \quad (3.1)$$

Table 3.1: Traffic information and statistics collected in various studies.

	Network	PRR	PLR/ PER	Bit Errors	Truncated Packets	CSI Available
[93]	WaveLan	✓				
[66],[67]	IEEE 802.11b	✓		✓		
[77]	IEEE 802.15.4	✓				✓
[116],[115]	IEEE 802.15.4	✓				✓
[133]	RFM/MICA	✓				
[137]	CC1100/ MICA2	✓				✓
[105]	IEEE 802.11x	✓				✓
[64]	IEEE 802.11g	✓		✓		✓
This Work	IEEE 802.15.4	✓	✓	✓	✓	✓

Khayam, et al. work in [67] was arguably the first bit-level residual error trace collection efforts for 802.11b/g WLANs. Meanwhile, there have been other trace collection efforts for the more recent IEEE 802.15.4 LR-WPAN, but like most 802.11 trace collections these too are limited to the observation of packets that pass the CRC test. Table 3.1 tabulates the statistics and observable parameters in various works.

The collection of data packets for traces is often augmented by recording of additional information with each packet. Various wireless networking standards require measurement of physical layer channel conditions. For example, 802.11b/g requires the measurements of signal-to-noise ratio (SNR) and background traffic noise level. Similarly, 802.15.4 mandates the measurement of RSSI and LQI of each received packet. We refer to all such measurements by the umbrella term CSI. As the side-by-side comparison in Table 3.1 shows, our error traces are by far the most detailed in terms of data collection and the recording of CSI. Our traces distinguish themselves in that they log are he only ones to provide bit-level residual error traces for IEEE 802.15.4, the positions of lost packets, as well as partially lost packets, a phenomenon,

which is probably unique to 802.15.4, and whose cause is explained a little later in this chapter. Further details regarding the meaning and definition of these parameters will also be provided.

The remainder of this chapter is organized as follows. Section 3.2 provides a detailed description of the setup used for trace data collection. Section 3.3 establishes the degree to which CSI measures provide information about the BER process. Section 3.4 uses maximum likelihood estimation (MLE) to arrive at a CSI driven model of the BER process. Section 3.5 evaluates the CSI driven BER model. Section 3.6 concludes this chapter.

3.2 Trace Collection Setup

To our knowledge, this is the first detailed trace collection effort for IEEE 802.15.4. These traces differ from previously collected ones in quantitative and qualitative aspects. We collected error traces of approximately 10 million packets in a way that provides, to the authors' best knowledge, an unprecedented level of insight into the effects of the wireless channel state on the level of corruption of packets.

3.2.1 Experimental Setup

The trace-collection setup is depicted in Figure 3.1 and consists of a Crossbow MPR2400 MICAz mote [34] transmitter and another MICAz mote mounted on a Crossbow MIB600 Ethernet gateway [33] as receiver. The gateway is connected to a host PC running an application that continuously retrieves data from the receiver and logs it.

3.2.2 Packet Payload

TinyOS [75] is one of the most widely used open source operating system in WSN devices. TinyOS v1.1 allows various packet formats to be transmitted. We suitably modified code to enable the standard 802.15.4 frame format which TinyOS v1.1 labels *CC2420 Frame Format* (after the Chipcon CC2420 chipset [120] used in MICAz devices). Strictly speaking, the term packet refers to the protocol data unit (PDU) exchanged between network layers of the transmitter and receiver while the term frame is used for PDU's exchanged between MAC layers. However, since our analysis is restricted to the MAC layer alone there is little cause for confusion and we will be using these two terms interchangeably to refer to MAC layer PDUs. The exact MAC frame format used is shown in Figure 3.2. The size of the frame is 41 bytes and comprises of a 1 byte Length Field, 2 byte frame control field (FCF), 1 byte sequence number, 2 byte destination PAN ID, 2 byte destination address, 1 byte type field, 1 byte group field, 29 bytes of data/payload followed by a 2 byte frame check sequence (FCS) containing a CRC. The contents of the payload field are of our own choosing and consist of 3 unused bytes, the Source Address, the Destination Address and 6 copies of a 32 bit sequence number. The sequence number in the data/payload is used to keep track of lost packets. If the sequence number between two consecutively received packets skips one or more numbers that is indicative of a packet loss. The sequence number field alone proves insufficient for this task in the face of long fades. Also, a single bit error in the 1 byte counter could easily become a source of ambiguity (did we just lose two long sequences of packets or receive bit errors in the sequence number field?). Note that transmitted packets differ only in the 1 byte sequence number in the header and the six 32 bit sequence numbers in the payload, and the CRC. For a particular trace all remaining bits remain unchanged. However, since the wireless channel will introduce bit errors the copies of the sequence number used

to track packet losses in the received packet may differ. For this purpose we use a majority vote of the received sequence numbers to determine the transmitted sequence number. From this we reconstruct the contents of the Data/Payload field and hence the transmitted packet.

3.2.3 Trace Generation

Bit-level error traces can be generated by comparing a transmitted packet with its received version. A simple bit-wise XOR operation on the transmitted and received packets yields a bit pattern in which a zero ('0') signifies a bit that is received without error while a one ('1') represents an inverted bit. We observe that in some cases the length of the received packet is shorter than the transmitted packets. This constitutes a partial loss and we use the term *truncated packets* to refer to such packets. An erased bit in a received packet will be denoted by a two ('2') in the error trace. Truncated packets are logged when bits in the MAC header's length field are inverted and the receiver stops listening to the wireless channel prematurely. It has also been observed that if bits in the length field are inverted in such a way that the length of the incoming packet appears longer than actual the length of the logged packet still equals that of the transmission. Although the Length field in the received packet may falsely indicate a longer packet, the absence of a carrier signal allows the receiver to detect the end of the transmission.

3.2.4 Channel State Information

Each received packet's logged entry is accompanied with three pieces of packet level CSI parameters. The first is the FCS status of the packet modeled by random variable Φ with the n^{th} packet's FCS status is represented by $\phi[n]$.

Ordinarily receivers only distinguish between two states, i.e. *FCS Pass* (denoted

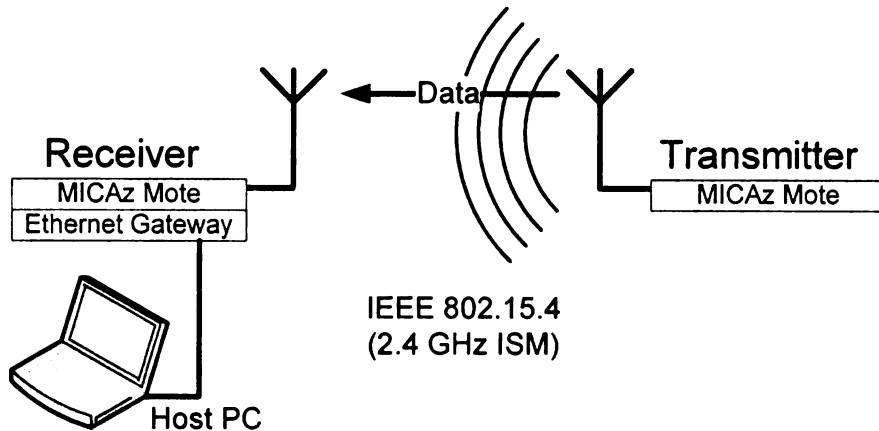


Figure 3.1: Equipment setup for trace collection.

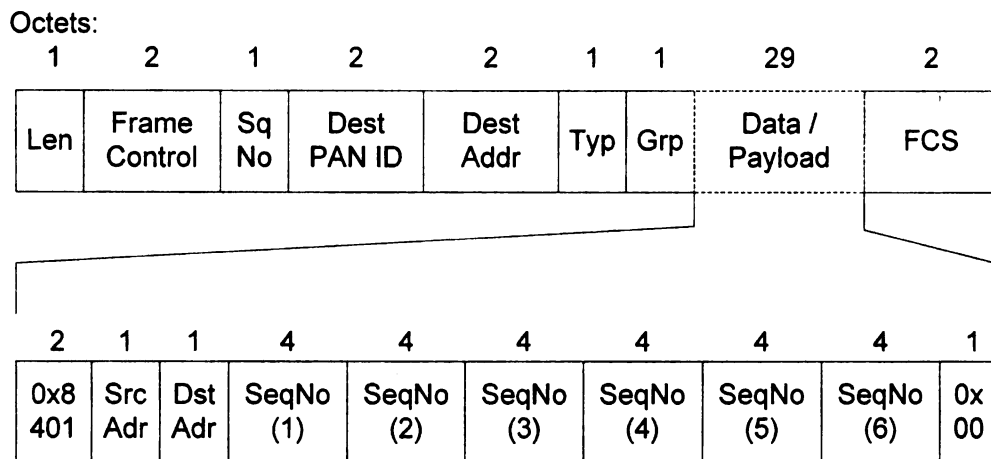


Figure 3.2: CC2420 MAC frame format used for experiments.

by $\phi[n] = 0$) if the CRC value in the FCS field matches the CRC of the received packet, and FCS Fail if does not. Since we have knowledge of packet erasures and size of transmitted packets we extend the definition of FCS status to accommodate the reason for failure. We restrict the definition of *FCS Fail BE* (denoted by $\phi[n] = 1$) to mean that the size of a received packet matches the size of the transmitted packet and the CRC failure is due to bit errors (BE). We further define two additional states, *FCS Fail PL* (denoted by $\phi[n] = 2$) and *FCS Fail CL* (denoted by $\phi[n] = 3$), where PL and CL are abbreviations for *partial loss* and *complete loss* respectively. Packets

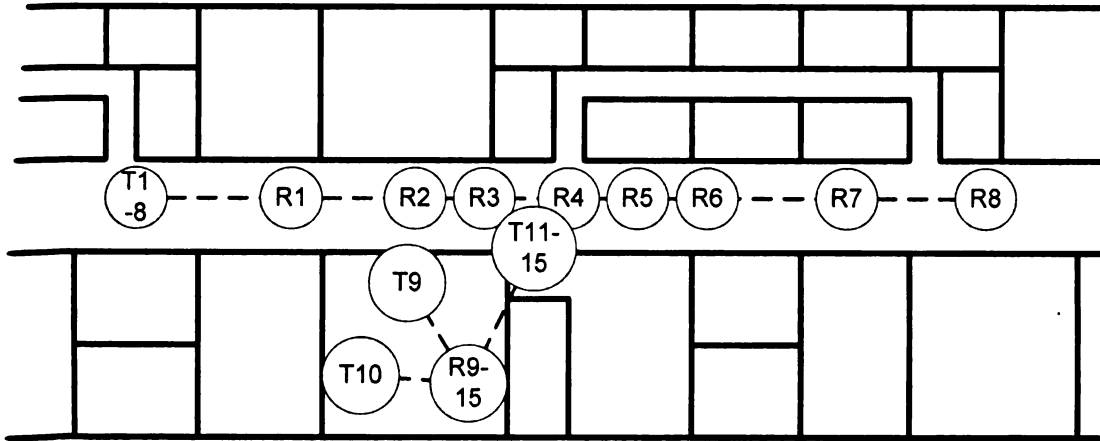


Figure 3.3: Office deployment environment.

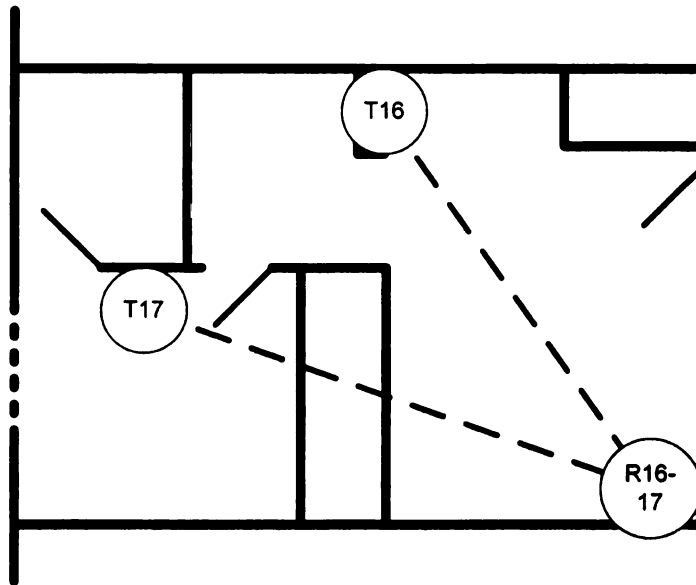


Figure 3.4: Residential deployment environment.

that are partially lost cannot pass the CRC test and are marked FCS Fail PL. Packets that are not received at all, i.e. when the decoded Sequence Number at receiver skips, are marked FCS Fail CL.

The Crossbow MICAz sensor mote uses a TI Chipcon CC2420 transceiver chip [120] for its communication subsystem. The CC2420 is an IEEE 802.15.4 compliant radio interface. In accordance with the standard, the receiver measures RSSI. The

RSSI constitutes the second piece of our CSI parameters and is modeled by random variable P and the n^{th} packet's RSSI denoted by $\rho[n]$. The RSSI is recorded as an 8 bit, signed 2's complement value.

Technically, the CC2420 does not measure the LQI directly. Instead, it measures the correlation $CORR$ between the first 8 received symbols (of the PHY header) and a corresponding 8 known symbols (preamble). IEEE 802.15.4 uses 16-ary Offset-Quadrature Phase Shift Keying (O-QPSK) modulation which encodes 4 bits in one symbol. The first 8 symbols, 4 bytes, of the PHY header comprise of the Preamble sequence consisting of 32 binary zeros. The LQI is then defined as

$$LQI = (CORR - c_1) \cdot c_2. \quad (3.2)$$

where the two constants c_1 and c_2 are functions of the Packet Error Rate (PER) measured over an extended period of time and are determined experimentally. c_1 and c_2 serve to scale the 7 bit value of the correlation to the full range of an 8 bit number. Since 8.2 is merely a linear transformation of the measured $CORR$ we simply take $c_1 = 0$ and $c_2 = 1$. The LQI is modeled by the random variable Λ and the n^{th} packet's LQI denoted by $\lambda[n]$. Although the BER process is not considered a CSI measure we are including it here nevertheless. The BER process is modeled by random variable B and the n^{th} packet's BER is denoted by $\beta[n]$. It must be mentioned here that the term BER is not used in its strict traditional sense where it denotes the long term average probability of a bit error, such as in a binary symmetric channel (BSC). Instead the BER is computed over each individually received packet. Thus, for the n^{th} received packet it is defined as,

$$BER = \beta[n] = \frac{\text{Number of inverted bits in } n^{th} \text{ received packet}}{\text{Length in bits of } n^{th} \text{ received packet}}. \quad (3.3)$$

Table 3.2: State space variables, symbols and values.

<i>Quantity</i>	Symbol of Packet Random Process	Symbol of Realization of Random Process	Valid Values Assigned to Random Processes
<i>FCSStatus</i>	Φ	$\phi[n]$	$\phi = \begin{cases} 0, & \text{FCS Pass} \\ 1, & \text{FCS Fail BE} \\ 2, & \text{FCS Fail PE} \\ 3, & \text{FCS Fail CL} \end{cases}$
<i>RSSI</i>	P	$\rho[n]$	$\rho \in Z \wedge -128 \leq \rho \leq 127$
<i>LQI</i>	Λ	$\lambda[n]$	$\lambda \in Z^+ \wedge 0 \leq \lambda \leq 127$
<i>BER</i>	B	$\beta[n]$	$\beta \in R^+ \wedge 0 \leq \beta \leq 1$

The value of $\beta[n]$ associated with the n^{th} packet is the instantaneous measure of the BER over that packet. Note that at this point we do not make any assumptions about the distribution of the inverted bits within a packet. Completely lost packets, with $\phi = 3$, are assigned $\rho = -128$, $\lambda = 0$, and $\beta = 1$.

Readers might argue that transmitter receiver separation could have been included as another dimension of the phase space. However, numerous previous works like [101] and ones listed in Table 3.1 have already established the tenuous nature of the relationship between CSI and distance. Hence, transmitter receiver separation is excluded from the phase space. Thus each received packet is characterized by its FCS Status, LQI, RSSI and BER processes. Together they constitute four state space variables of our system. Table 3.2 summarizes this notation and lists the range of possible values each may assume.

3.2.5 Spectral and Environmental Diversity

The traces collected in this study are divided into two sets, a multi-channel (MC) trace set, and a multi-environment (ME) trace set. In order to obtain sets of traces rich in environmental diversity, traces in the ME trace set were collected over a period of months, at different times of the day, in office, residential and outdoor environments. All ME traces were collected while operating in channel 26 in the $2.480GHz$ band. The reason for choosing channel 26 was the fact that it is the channel in the frequency spectrum that is the farthest removed from all 802.11bg frequency channels.

Each trace collection is characterized by the locations of the transmitter and receiver, separation between them, packet transmission rate ω in packets per second (pps) and whether communication was line-of-sight (LOS) or non-line-of-sight (NLOS). For all our traces the transmission power was kept constant at the default $0dB$ which corresponds to $1mW$. Figure 3.3 and Figure 3.4 depict floor plans of the environments in which traces were collected. The circles labeled $T1$ through $T18$ denote transmitter locations. The corresponding locations of receivers are marked by $R1$ through $R18$. For the remainder of the chapter, we refer to individual traces $TR1$ through $TR19$. Traces collected in the same location are collectively referred to by the name of the trace collection environment. This way $TR1$ through $TR8$ are collectively referred to as the ‘Hallway’ trace set, $TR9$ through $TR15$ as the ‘Lab’ trace set, $TR16$ and $TR17$ as the ‘Residential’ trace set and $TR18$ and $TR19$ as the ‘Outdoor’ trace set. With the exception of the ‘Hallway’ traces none of the environments had any significant WLAN interference sources. Traces are subject to interference from cordless phones and microwave ovens. Table 3.4 briefly characterizes the various trace sets by providing PER, PLR and PRR of each.

The MC set consists of 16 traces, labeled MC-11 through MC-26, all collected in the same residential, non-line-of-sight (NLOS) setting with 15 feet transmitter-

receiver separation. The transmitter is operated at full transmission power. Each MC trace is collected at one of the 16 channels numbered 11 through 26 specified by IEEE 802.15.4. The residential environment used for the collection of the MC set is subject to interference from multiple IEEE 802.11x WLANs. At the time the MC set was collected there were two networks in WLAN channel 1, two networks in channel 6, one network in channel 10 and 2 networks in channel 11 with varying activity levels. Figure 4 of [56] depicts the population of the $2.4GHz$ ISM band and what IEEE 802.11b/g WLAN channels interfere with which IEEE 802.15.4 LR-WPAN channels.

Furthermore, the transmit power of IEEE 802.11b/g devices is $30mW$ which is significantly higher than the $1mW$ of 802.15.4. As Srinivasan, Dutta, Tavakoli and Levis reported in [115], to a co-located 802.11 WLAN sharing the same spectrum traffic from IEEE 802.15.4 devices appears as noise. One might argue that since interference scenarios are part of our evaluation, other interference sources occupying the ISM band (see figure 3.5) such as Bluetooth/ IEEE 802.15.1 [4], microwave ovens and cordless phones should also have been part of our traces. However, as Sikora and Groza have shown in their study [112] on the coexistence of 802.15.4 with other systems in the $2.4GHz$ ISM band, co-located Bluetooth networks and microwave ovens have no discernible effect on the operation of IEEE 802.15.4 networks. The effects of 802.11b/g networks on the other hand are significant and have been the subject of many studies ([112], [48], [132], [102]), which is why we have only included 802.11b/g networks as interference sources.

3.3 Correlation Analysis of CSI Measures

A primary objective of this study is to model the probability density function (PDF) of the BER process B conditioned on CSI. It is important to highlight that although we were able to observe the BER process in our trace-collection study, in an actual

Table 3.3: Collection environments of various traces in the ME trace set.

Trace	Environment	Interference Sources	Distance (feet)	LOS/NLOS
ME-1	Office bldg corridor	802.11x (strong)	40	LOS
ME-2	Office bldg corridor	802.11x (strong)	60	LOS
ME-3	Office bldg corridor	802.11x (strong)	70	LOS
ME-4	Office	802.11x	20	NLOS
ME-5	Office	802.11x	20	NLOS
ME-6	Residence	802.11x (multiple networks)	15	NLOS
ME-7	Office bldg corridor	802.11.x (low)	20	LOS
ME-8	Office bldg corridor	802.11.x (low)	50	LOS
ME-9	Office bldg corridor	802.11.x (low)	100	LOS
ME-10	Office bldg corridor	802.11.x (low)	80	LOS
ME-11	Outdoors	-	100	LOS
ME-12	Office bldg corridor	802.11x (low)	120	LOS

Table 3.4: Error rates in trace sets.

<i>TraceSets</i>	PER	PLR	PRR
<i>Hallway</i>	0.1877	0.0762	0.7361
<i>Lab</i>	0.0275	0.0161	0.9564
<i>Residential</i>	0.0993	0.0420	0.8587
<i>Outdoors</i>	0.0005	0.0030	0.9965

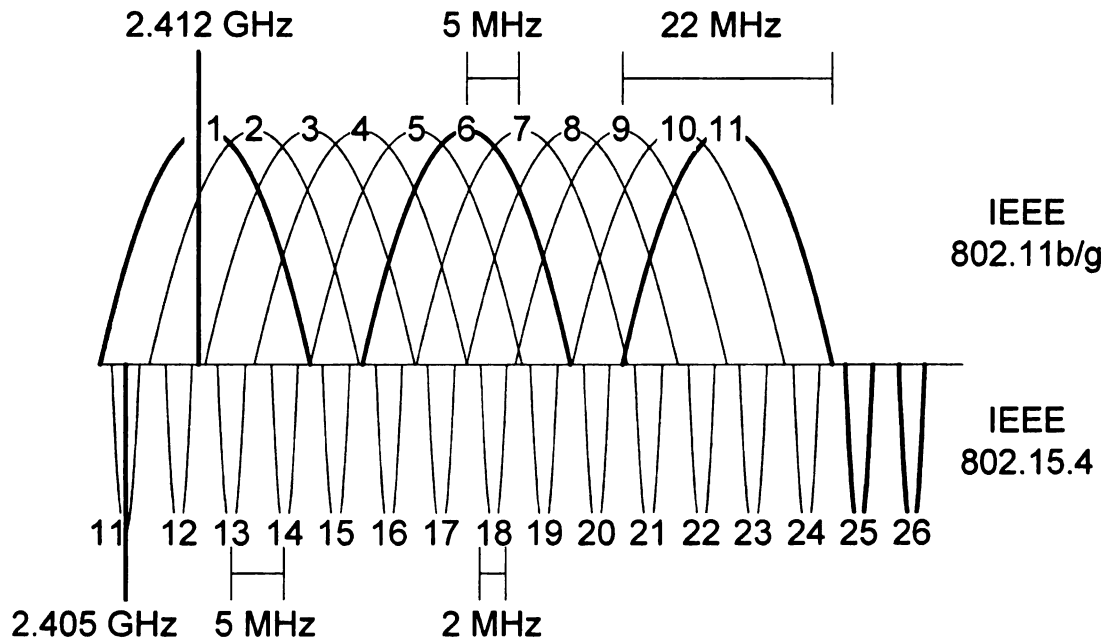


Figure 3.5: IEEE 802.11b and IEEE 802.15.4 channels in the ISM band.

network application the error process can only be estimated; and hence $\beta[n]$ is not observable in actual networks. However, all other processes ($\phi[n]$, $\rho[n]$, and $\lambda[n]$) are observable. Hence, the problem of determining $p(\beta[n])$, the PDF of the error process $\beta[n]$, for the n^{th} received packet based on observable CSI measures $\lambda[n]$, $\rho[n]$ and $\phi[n]$. Note that this is different from prior uses of CSI such as in [116] where it is shown that the average PLR/PER of a link is correlated with the average RSSI. When $\phi[n]$ this indicates that the received packet passed the CRC and can be considered free of errors, i.e. $\beta[n] = 0$ with certainty. When $\phi = 3$ the packet is erased completely and there is no data and no CSI available. This leaves us to focus on the cases when $\phi = 1, 2$, i.e. the received packet contains errors and/or is partially lost, i.e.

$$\lambda[n], \rho[n] | \phi[n] = 1, 2 \rightarrow p(\beta[n]). \quad (3.4)$$

Reducing the uncertainty surrounding the PDF of $\beta[n]$ of a packet received with some bits in error on the basis of its LQI and RSSI implies that these CSI measurements are somehow correlated with $p(\beta[n])$. We begin by establishing that CSI parameters are indeed highly correlated with the actual BERs of packets by means of correlation based analysis. The uncertainty in the knowledge about $p(\beta[n])$ is measured in terms of its variance.

Packets with low BER are expected to have high RSSI and LQI, and vice versa. Thus, to qualify as ‘useful’ RSSI and LQI must both be negatively correlated with BER. Let $Ti(\beta)$ denote a signal consisting of the BERs observed for packets captured in trace TRi . If X and Y are two random variables with means $E[X]$ and $E[Y]$, then the correlation coefficient $R_{XY}(t)$ is defined as,

$$\begin{aligned} R_{XY}(t) &= \frac{Cov[X, Y_t]}{\sqrt{Var[X] Var[Y_t]}} \\ &= \frac{E[(X - E[X])(Y_t - E[Y_t])]}{\sqrt{E[X^2] - E^2[X]}\sqrt{E[Y_t^2] - E^2[Y_t]}} \end{aligned} \quad (3.5)$$

Where Y_t is the t time delayed version of Y . When correlation is computed for a range of time delays the set of correlation coefficients so obtained will be referred to as a correlation function. For every trace we treat the observed values of state space variables β , λ and ρ as time series signals and compute the correlation coefficients between them. Ideally we would like to see $R_{BP}(0)$, the cross-correlation coefficient of B and P , and $R_{B\Lambda}(t)$, the cross-correlation coefficient of B and Λ , to be very close to -1 . For sake of completeness we also computed $R_{P\Lambda}(t)$, the cross-correlation coefficient of P and Λ . These values are computed separately for different trace sets by concatenating signals of state space variables from all traces contained in a trace set. Strictly speaking though, correlation coefficients should be computed separately

Table 3.5: Cross correlation of different random processes.

	$R_{B\Lambda}(t)$	$R_{BP}(t)$	$R_{\Lambda P}(t)$
<i>Hallway</i>	-0.4662	-0.2938	0.8414
<i>Lab</i>	-0.2745	-0.0919	0.5978
<i>Residential</i>	-0.2071	-0.0091	0.5346
<i>Outdoors</i>	-0.3262	-0.1613	0.8443

for each trace. However, the results shown in this subsection change very little across traces in the same trace set. Therefore, for brevity reasons we are presenting results for the set of concatenated traces. As Table 3.5 shows, for packets observed at the receiver (packets with $\phi = 0, 1, 2$), both RSSI and LQI have exhibit a moderate to strong negative correlation with the BER. Furthermore, RSSI and LQI are also strongly correlated.

This trend is also observable in a visual inspection of the data. We are able to obtain the joint distribution $p_{B\Lambda P\Phi}(\beta, \lambda, \rho, \phi)$ and from it any conditional distribution, such as $p_B(\beta|\lambda, \phi = 1, 2)$ and $p_B(\beta|\rho, \phi = 1, 2)$, by means of the error traces we collected. Figures 3.6 and 3.7 show two conditional distributions $p_B(\beta|\lambda, \phi = 1, 2)$ and $p_B(\beta|\rho, \phi = 1, 2)$, respectively. The axis along the left hand side of the horizontal plane is the BER axis. The axis along the right hand side of the horizontal plane is the CSI measure (LQI in figure 3.6 and RSSI in dBm scale in figure 3.7). Thus for each value of LQI and RSSI the figures depict a valid PDF of BER. Figure 3.6 depicts how variance and average BER of packets in the low LQI range increases which is in accordance with our previous observations. The same trend is visible for RSSI in figure 3.7. Thus, the figures depict inverse relationships of the BER with both CSI measures. Our analysis so far leads us to conclude that both RSSI as well as LQI measurements made with each received transmission have the potential to serve as good indicators of the BER for that particular packet.

PDF of BER conditioned on LQI

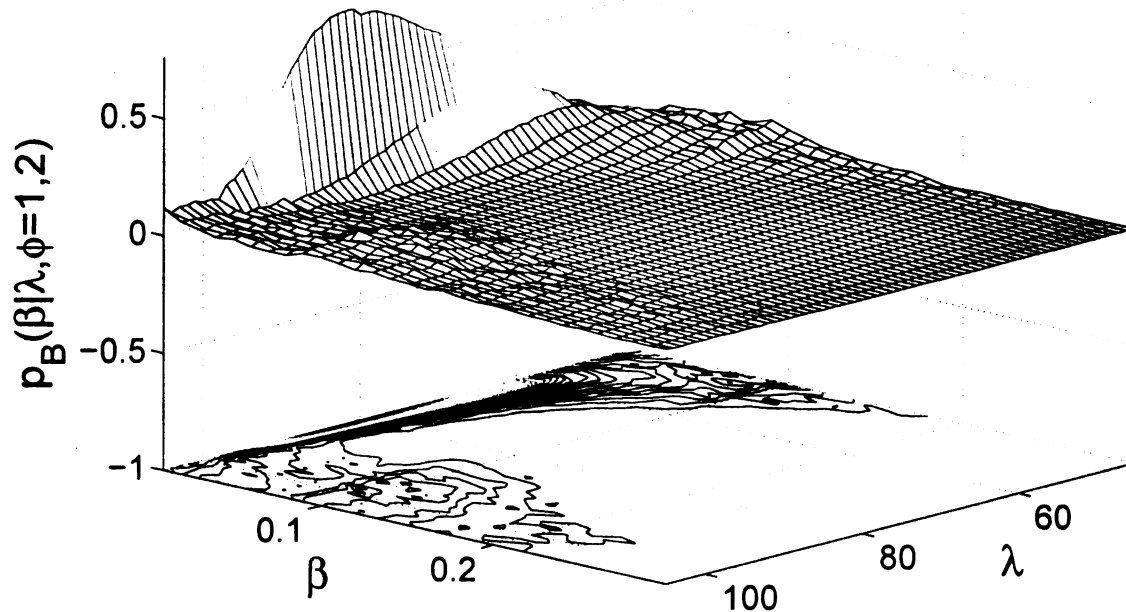


Figure 3.6: PDF of B conditioned on Λ and $\Phi = 1, 2$.

3.4 CSI Driven BER Model

In this section we employ maximum likelihood estimation (MLE) to determine the parameters of $\tilde{p}_B(\beta|\lambda, \rho, \phi = 1, 2)$, an exponential PDF modeling $p_B(\beta|\lambda, \rho, \phi = 1, 2)$. From collected traces we observe that the BER PDFs are always decreasing functions. We will be modeling PDFs of BER by a discretized exponential PDF. The shape of the exponential PDF is determined by a single parameter b as shown in 3.6,

$$f(x; b) = \frac{1}{b} e^{-\frac{x}{b}}; x \geq 0. \quad (3.6)$$

The MLE estimates of parameter b can be obtained from a data set by,

$$b = E[X]. \quad (3.7)$$

PDF of BER conditioned on RSSI

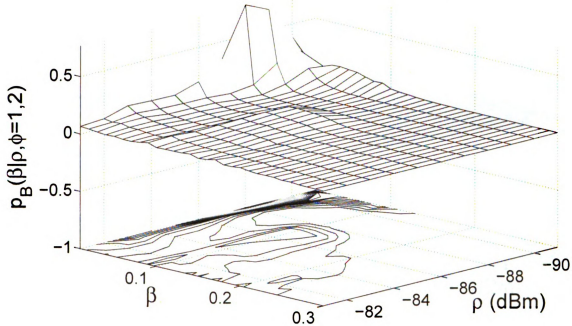


Figure 3.7: PDF of B conditioned on P and $\Phi = 1, 2$.

The MLE of b for all observed combinations of values of (λ, ρ) constitutes a model of the BER process. $b(\lambda, \rho)$ are depicted in Figure 3.8. Depending on the objective of the application seeking the channel state estimate, $\hat{p}_B(\beta|\lambda, \rho, \phi = 1, 2)$ can now either be used as is or mapped to a single numerical value representative of the channel state. For lack of a better graphical representation, figure 3.9 shows $p_B(\beta|\lambda, \rho = 88dBm, \phi = 1, 2)$, the PDFs of the BER process observed over the range of LQI measurements with RSSI fixed at $88dBm$ (chosen arbitrarily for illustration purposes). Figure 3.10 displays the same PDFs modeled by MLE exponentials. The above model leaves us with two very important questions.

Q1: How useful is this model in terms of reducing uncertainty or variance of the BER's PDF?

Parameters of exponential PDF at different LQI and RSSI

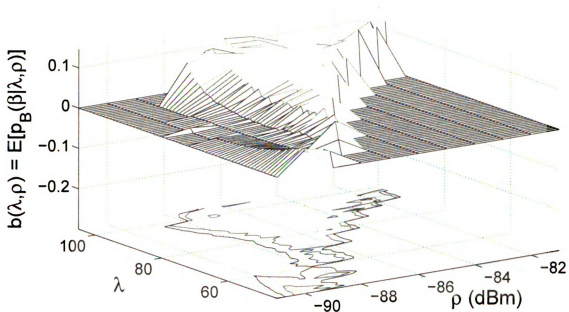


Figure 3.8: Values of b for various (λ, ρ) .

Q2: How universally applicable is this CSI driven BER model especially when we consider that wireless channels behave very differently in different environments?

We will address these two questions in the following section where we evaluate this model.

3.5 Model Evaluation

3.5.1 Variance Reduction

Taking variance as a measure of uncertainty in stochastic systems we evaluate the expected value of variance in estimates of the BER. In other words, we contrast the performance of LQI and RSSI used individually and when used in conjunction by evaluating the expected value of the variance of estimates of the BER's PDFs, i.e. $E[VAR[\hat{p}_B(\beta|\lambda, \rho, \phi = 1, 2)]] = E[VAR_B|\Lambda P], E[VAR[\hat{p}_B(\beta|\lambda, \phi = 1, 2)]] =$

Actual $p_B(\beta|\lambda, \rho=88dBm)$ from collected traces

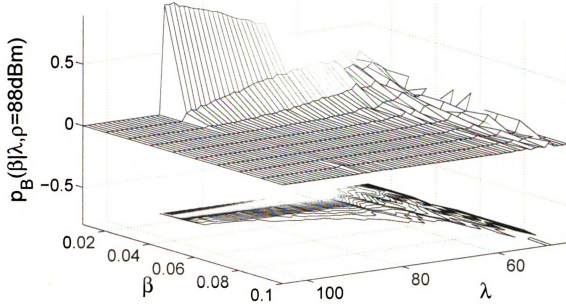


Figure 3.9: The PDFs of the BER obtained from the actual traces for various LQI measurements at an RSSI of $88dBm$, $p_B(\beta|\lambda, \rho = 88dBm, \phi = 1, 2)$.

$E[VAR_{B|\Lambda}]$, and $E[VAR[\hat{p}_B(\beta|\rho, \phi = 1, 2)]] = E[VAR_{B|P}]$. These expected values are computed over the joint PDF $p_{\Lambda P}(\lambda, \rho|\phi = 1, 2)$ of LQIs and RSSIs with which packets with errors are received, also obtained from our traces. Table 3.6 tabulates the expected value of variance with which we estimate the BER process. Clearly, once again we see that using LQI and RSSI together is more beneficial and estimates the error process with lesser uncertainty.

3.5.2 Dependence On Deployment Environment

So far we have established that for every error-prone packet with $\phi = 1, 2$ it is possible to obtain a better estimate of the PDF of its BER based on measurements of RSSI and LQI. Ideally, we would like to make this estimate dependent solely on the RSSI and LQI and would like to see little or no dependence on the physical environment.

Model of $p_B(\beta|\lambda, \rho=88dBm)$ based on MLE of exponential PDF

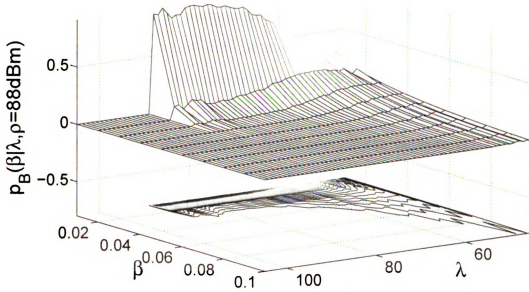


Figure 3.10: The PDFs of the BER for the same range of LQI measurements at RSSI of $88dBm$ as modeled by a discretized exponential PDF, $\tilde{p}_B(\beta|\lambda, \rho = 88dBm, \phi = 1, 2)$.

To evaluate the application of the CSI driven BER model in different environment we partition the data set according to collection environments, i.e. 'Hallway', 'Lab', 'Residential', and 'Outdoor', and use each to arrive at a CSI driven BER model independently. Ideally the four models so obtained should be identical. As we already know, each model is a set of PDFs. If λ and ρ vary in the ranges $\lambda_{min} \leq \lambda \leq \lambda_{max}$ and $\rho_{min} \leq \rho \leq \rho_{max}$, respectively, that means each model will consist of (at most)

Table 3.6: Expected value of variance when using different combinations of RSSI and LQI as CSI.

	<i>Expected Variance</i>
$E[VAR_B]; CSI = None$	1.7000×10^{-3}
$E[VAR_{B \Lambda}]; CSI = LQI$	8.0863×10^{-4}
$E[VAR_{B P}]; CSI = RSSI$	8.1242×10^{-4}
$E[VAR_{B \Lambda P}]; CSI = LQI + RSSI$	5.7524×10^{-4}

$(\lambda_{max} - \lambda_{min} + 1) \times (\rho_{max} - \rho_{min} + 1)$ number of PDFs $\tilde{p}_B(\beta|\lambda, \rho, \phi = 1, 2)$, one PDF of B for every combination of possible (λ, ρ) tuples, we show that the PDFs $\tilde{p}_B(\beta|\lambda, \rho, \phi = 1, 2)$ obtained from one trace set closely approximate the PDFs $p_B(\beta|\lambda, \rho, \phi = 1, 2)$ of a different trace set for the same values of (λ, ρ) . This way traces the CSI driven BER model derived using data from one environment will be verified against test data collected in a different environment. At this point we require a measure that quantifies the degree of similarity between two PDFs. A well established divergence measure is the *Kullback-Leibler Divergence* (KLD), also known as *relative entropy*. The KLD of two PDFs p_X and p_Y is denoted by $D(p_X \parallel p_Y)$ and defined as,

$$D(p_X \parallel p_Y) = \sum_{x \in S_{p_X}} p_X(x) \log \left(\frac{p_X(x)}{p_Y(y)} \right). \quad (3.8)$$

S_{p_X} denotes the *region of support* of PDF p_X , or the elements in the domain of p_X for which $p_X(x) > 0$. Note that according to the definition of $D(p_X \parallel p_Y)$, if $S_{p_X} \not\subseteq S_{p_Y}$ then for any $x \notin S_{p_Y}$ the KLD is ∞ . Although this should not happen in a large data set consisting of a comprehensive set of traces from various locations, trace packets in individual partitions of the data set might not exhibit sufficient diversity across the entire spectrum of possible values of (λ, ρ) to avoid such a result. To circumnavigate this pitfall in our analysis we use a modified form of the KLD called the K-directed divergence (KDD) introduced by Lin in [76] which is defined,

$$K(p_X \parallel p_Y) = \sum_{x \in S_{p_X}} p_X(x) \log \left(\frac{p_X(x)}{\frac{1}{2}p_X(x) + \frac{1}{2}p_Y(y)} \right). \quad (3.9)$$

The KDD's most significant feature relevant to our application is that the denominator of its fractional term will never be zero, and hence the KDD cannot evaluate to ∞ .

Thus, to measure the similarity between PDF $\tilde{p}_{B-X}(\beta|\lambda, \rho, \phi = 1, 2)$ derived from

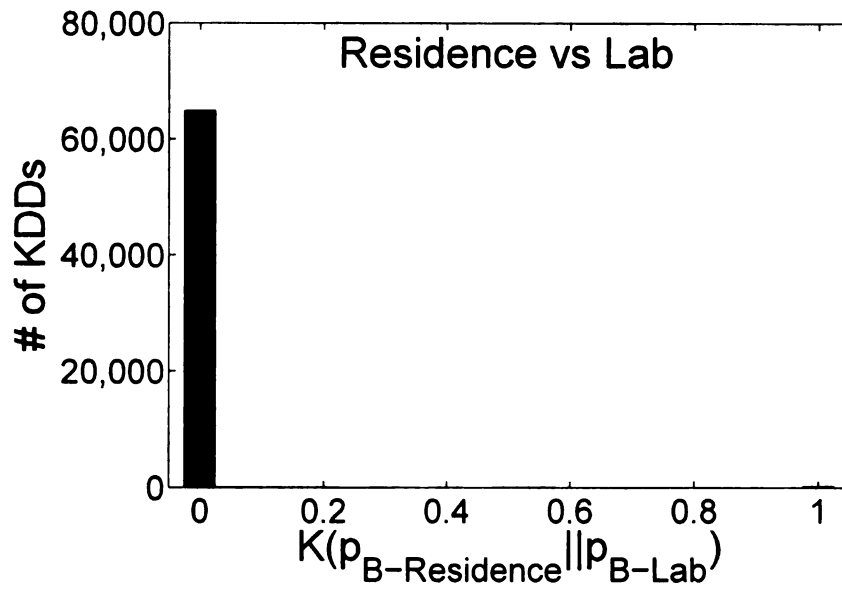
traces collected in an environment X and a PDF $p_{B-Y}(\beta|\lambda, \rho, \phi = 1, 2)$ of traces collected in an environment Y we employ the KDD. A complete comparison of two environments will require the computation of KDDs for each (λ, ρ) pair. This yields an entire set of KDD measures. Ideally, all KDDs in this set should be close to zero. In figure 3.11a we plot the histogram of the set of KDDs obtained from comparing the CSI driven BER model derived from the 'Residential' set with the PDF of the data collected from the 'Office' trace set. As the histogram shows, the vast majority of KDDs are zero, with very few that are non-zero but nevertheless close to zero.

Similarly, figure 3.11b is the same histogram plot for PDFs from 'Hallway' and 'Outdoor' traces. A particular reason why we chose this example for illustration is that it uses the 'Outdoor' trace set which exhibits and significantly different from that of the other three sets (see table 3.1). Those differences are manifesting themselves in a less obvious way in the histogram of KDD values in figure 3.11b. The histogram rises to a non-zero entry at the bin centered at 1 which, however, remains insignificant in comparison to the number of entries in the bin centered at 0. For the vast majority of valid (λ, ρ) pairs the KDD between different trace sets is zero. We conclude that the CSI driven model for the PDF of the BER process is applicable in a variety of environments.

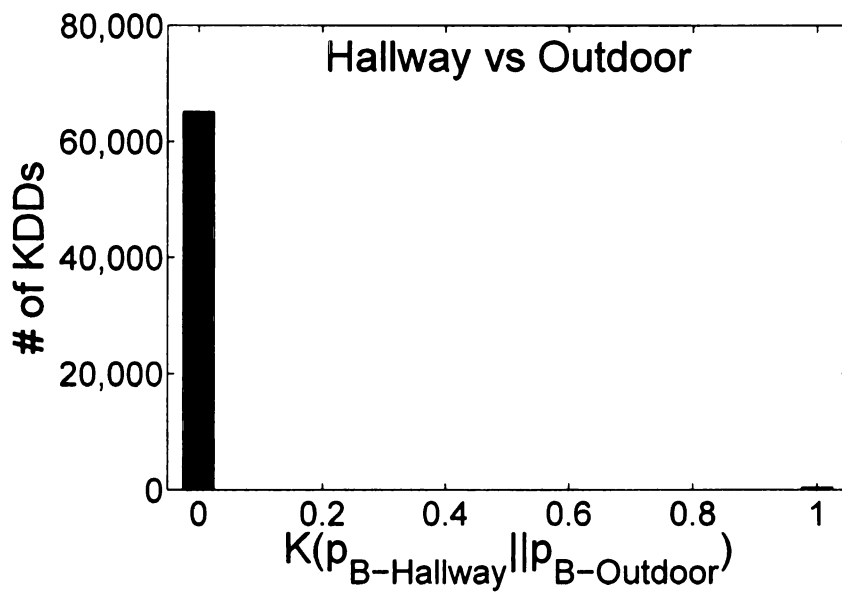
3.6 Conclusions

We collected and analyzed an extensive and diverse set of residual error traces from 802.15.4 links. Listed below are our conclusions.

1. LQI and RSSI exhibit moderate negative correlation with the BER process (and strong positive correlation with each other).
2. LQI and RSSI can be used to reduce uncertainty regarding in the BER individ-



(a)



(b)

Figure 3.11: Histogram of KDD values for a) Model from ‘Residential’ trace set and ‘Lab’ traces and b) Model from ‘Hallway’ trace set and ‘Outdoor’ traces.

ual packets are subjected to.

3. The CSI driven BER model remains valid across a variety of physical environments.

Chapter 4

Memory Properties of the Link-level BER Process in IEEE 802.15.4 Links

4.1 Introduction

Communication channels in the real world are not perfect and are prone to introduce errors into transmissions. Errors are orders of magnitude more frequent in wireless channels than in wired channels. Design of network protocols and other architectural components of wirelessly networked communication systems entail a better understanding of the error process that affects transmissions. The memory length of the error process is an important parameter of interest that has to be taken into consideration in the formulation of channel models. In particular, channel models that take into account the persistence of errors in wireless channels, such as those based on Markov chains, require information about the memory length of the error process.

Previous works ([68], [66], [67]) measuring memory length in 802.11b channels were restricted to the bit-level error process and relied on the correlation function based analysis, which sometimes worked well enough at that scale. We were the first to perform bit-level analysis for IEEE 802.15.4 low rate-wireless personal area network (LR-WPAN) [5] channels and extend it to symbol and packet-level error processes in [56]. In this chapter we are extending our previous analysis by using a much larger data set that spans not only the set of all 16 channels in which 802.15.4 operates, but also a variety of physical environments. The failure of traditional correlation analysis is explained using results from [54]. At the bit and symbol level the error processes are modeled by binomial processes (a bit/symbol is either received correctly or incorrectly). The packet-level process is called the bit error rate (BER) process and is denoted by B . The BER process B is defined as a series of measurements of the rate at which each packet has been subjected to bit-errors. The n^{th} measurement in a realization of B is denoted by $\beta[n]$ and is computed as shown in equation 8.1.

$$\beta[n] = \frac{\# \text{ of bits in } n^{th} \text{ packet received with error}}{\# \text{ of bits received in } n^{th} \text{ packet}}. \quad (4.1)$$

We are concerned with determining the memory length of errors due to the effects of slow and fast fading. As we will demonstrate, the determination of this memory length is complicated by several factors which include slow fading, non-stationarity due to long stretches of interference, and periodic interference. A way around the effects of slow fading and interference is to preprocess the BER process by de-trending and normalization [131]. However, this is complicated by their non-periodic, unpredictable nature.

Like some prior works that measured memory length of errors, this research too depends on bit-level, binary signals, called *residual bit-error traces* (short: *error traces* or *traces*), representing the positions of bit-errors in received transmissions. The basic concept of error traces is very simple and is explained easily enough. An error trace is a map of bit positions of all packets collected in the course of a trace collection session that reached the receiver with errors. Conceptually, for a single packet such a bitmap of errors is obtained by comparing a transmitted packet (free of errors) with its received version (may contain errors and failed the cyclic redundancy check). This way the sequence of BERs computed by equation 8.1 for each packet in a trace constitutes the BER process. The complications involved in working with traces arise due to lack of access to low level drivers and/or firmware that need to be modified to gain access to packets received with errors, otherwise discarded by receivers.

The remaining chapter is organized as follows. Section 4.2 demonstrates the use of correlation coefficient in measuring the memory length of bit, symbol and packet-level error processes. Section 4.3 presents the results of Hurst analysis on the packet-level BER process. Section 4.4 describes *relative mutual information* (RMI), an alternative means of measuring memory of in measuring the memory length of the BER process observed of IEEE 802.15.4 LR-WPANs channels. Section 4.5 concludes the chapter.

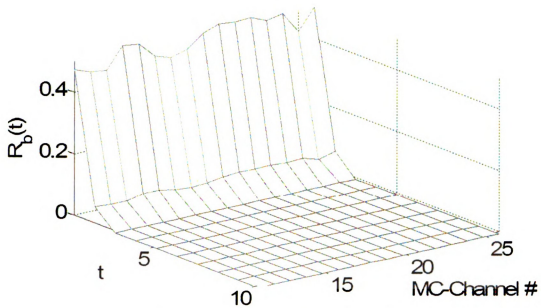
4.2 Memory Length Measurement By Correlation Analysis

4.2.1 Correlation Function

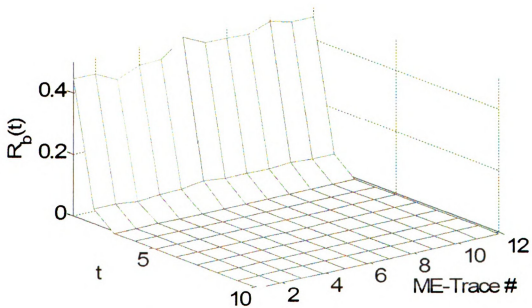
Let X be a random process consisting of random variables $[X_1, X_2, \dots, X_N, \dots]$. The corresponding measurements in a random process are denoted by $[x_1, x_2, \dots, x_N, \dots]$. $X^{(m)}$ is an m time unit delayed (right-shifted) version of X . To measure memory length we compute the correlation coefficient of X and $X^{(m)}$ for a range of values of m . Thus, the correlation coefficients denoted by $R_X(m)$ are a function of m and are defined as,

$$R_X(m) = \frac{Cov(X, X^{(m)})}{\sigma_X \sigma_{X^{(m)}}} = \frac{E[(X - \mu_X)(X^{(m)} - \mu_{X^{(m)}})]}{\sigma_X \sigma_{X^{(m)}}}. \quad (4.2)$$

Here $Cov(X, X^{(m)})$ denotes the covariance of X and $X^{(m)}$, σ_X the standard deviation of X , μ_X the mean of X , and $E[\cdot]$ the expectation function. This function of m is also known as the correlation function or *correlogram*. The value of m after which the correlation function $R_X(m)$ becomes "insignificant" and drops below threshold R_t is the memory length M_X of process X . There is no clear consensus on what the value of R_t should be. In the simplest of cases the smallest value of m for which the correlation function is 0 or close to it is taken as a measure of M_X . There is a 95% significance level that is frequently used as a rule of thumb to determine whether M_X is zero (successive measurements in X are independent) or non-zero (successive measurements in X are not independent). For a signal of N consecutive measurements the 95% significance range is defined as $\pm \frac{2}{\sqrt{N}}$. If 95% of correlation coefficients for m in the range $1 \leq m \leq \frac{N}{4}$ lie within this specified range, then



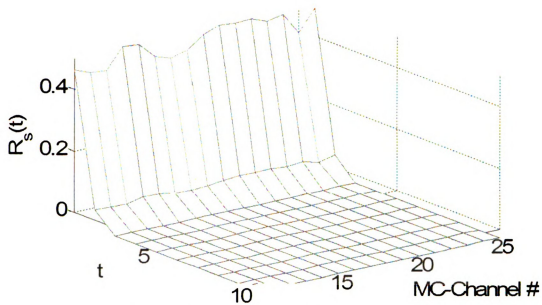
(a) All bit-level traces of the MC trace set.



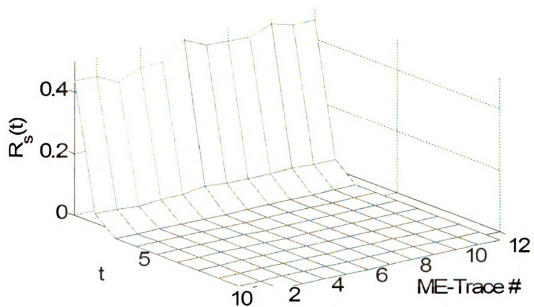
(b) All bit-level traces of the ME trace set.

Figure 4.1: Auto-correlation functions for bit level traces of the MC and ME trace sets.

consecutive measurements are deemed independent, leaving $M_X = 0$.



(a) All symbol-level traces of the MC trace set.



(b) All symbol-level traces of the ME trace set.

Figure 4.2: Auto-correlation functions for symbol level traces of the MC and ME trace sets.

$$M_X = \min_m (R_X(m) < R_t). \quad (4.3)$$

4.2.2 Correlograms of Bit and Symbol-level Traces

For analysis of bit and symbol-level error processes we compute $R_b(k)$ and $R_s(k)$ defined as,

$$R_b(k) = E [R_b(k, i)], R_s(k) = E [R_s(k, i)]. \quad (4.4)$$

$R_b(k, i)$ and $R_s(k, i)$ are the correlation functions of the i^{th} packet's bit and symbol traces with its k^{th} following packet, respectively. The correlation functions $R_b(k)$ and $R_s(k)$ of a particular trace are computed as the expectation function over i of $R_b(k, i)$ and $R_s(k, i)$, respectively. If for a $k > m_b, m_s$ the correlation functions $R_b(k)$ and $R_s(k)$ drop to a value very close to zero, then m_b and m_s are the bit and symbol-level channel memory of the channel. Figures 4.1a and 4.1b plot the auto-correlation functions or correlograms of all bit-level traces in the MC and ME trace sets, respectively. What is apparent from this figure is that in spite of the wide variation in trace collection parameters, we consistently observe a bit-level channel memory m_b of at most 2 bits across all traces regardless of channel frequency and environmental differences. Similarly, figures 4.2a and 4.2b are the correlograms of the symbol-level traces for the MC and ME trace sets, respectively. From both correlograms it is clearly visible that symbol level memory m_s is also (at most) 2. Therefore, there is no further need to check for long range dependence (LRD) and we conclude that the bit and symbol-level error processes in 802.15.4 wireless channels have constant memory of (at most) 2 bits and 2 symbols, respectively.

4.2.3 Correlograms of Packet-level Traces

Figure 4.3 depicts the correlogram functions for all 16 traces in the MC trace set, and figure 4.4 for the 12 traces in the ME trace set. For the computation of the

autocorrelation function, the length of each of the traces is truncated to a uniform $N = 100,000$ points, making the 95% significance range $[-0.0063, +0.0063]$. By this measure it appears as if all channel traces exhibit memory to some degree. To determine just how much memory several rules of thumb have been used in literature. These include;

$R_t = \min_i (R_B(i) = 0)$: Correlation coefficient falls close to insignificance/zero. In all our analyzed traces this has rarely been the case.

$R_t = 0.1 \times R_B(1)$: Correlation coefficient drops to less than 10% of the coefficient at lag 1. By this standard, the memory length of the BER process will range in the tens to thousands of seconds.

$R_t = \min_i (|R_B(i) - R_B(j)| < \delta) : \forall j > i, \delta \rightarrow 0$: Correlation coefficient becomes steady and subsequent changes between consecutive values are within a very small value δ . Periodic interference, most likely from IEEE 802.11b beacon frames, causes periodic spikes in the correlogram functions rendering this criterion useless. Furthermore, there is no clear interpretation of the value of correlation coefficients and the degree of predictability of one measurement on another.

According to all these selection criteria for R_t , memory lengths of all these traces lie in the range of tens to hundreds of seconds. Clearly, that rules out correlogram analysis to measure memory length of fast fades. The failure of the correlogram analysis is explained by the non-stationary nature of the BER processes captured by the traces. Figure 4.5 plots the BER process observed in MC-25, the trace of the MC trace set that was collected with devices operating in channel 25. For clearer visibility, the BER process was pre-processed by passing it through a 600 point averaging filter. There is an initial period of about 1400sec in which the channel experiences

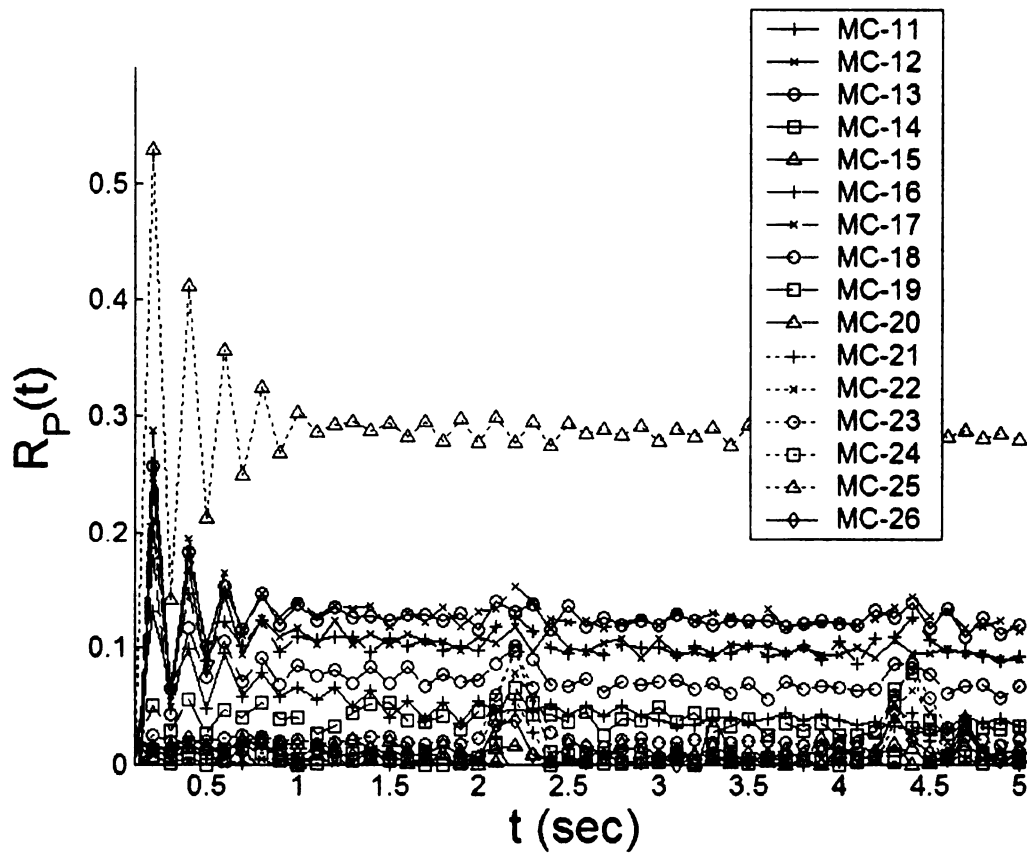


Figure 4.3: Auto-correlation functions for traces of the MC trace sets.

a moderate BER which then increases to a much higher value for about 900sec. It then reduces to a very low value and remains so for the remainder of the trace. We attribute these changes in channel conditions to interference from co-located devices sharing the spectrum in the 2.4GHz ISM band. When the correlation function is computed over a trace which experiences a change in channel conditions like the one in figure 4.5, the correlation function is expected to maintain a significant value for an extended period of time.

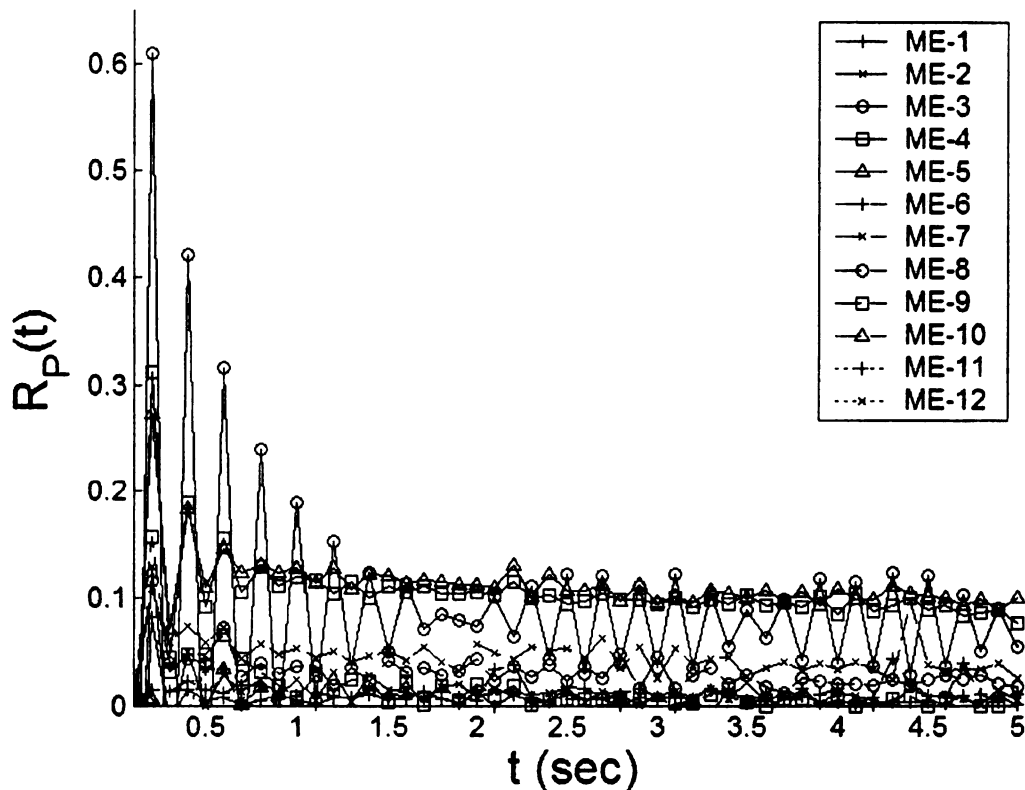


Figure 4.4: Auto-correlation functions for traces of the ME trace sets.

4.3 Hurst Analysis of Packet-level BER Process

The correlation function alone did not allow us to make a definitive conclusion about memory length. For this reason we employ the Hurst parameter H for which several estimators exist in literature [10]. We use the Hurst parameter, (or Hurst exponent) as a means of determining whether a process is LRD or not. A stationary process is said to be LRD if there exists a real number $\alpha \in (0, 1)$ and constant $c_p > 0$ for which,

$$\lim_{k \rightarrow \infty} \frac{R(k)}{c_p k^{-\alpha}} = 1. \quad (4.5)$$

The Hurst parameter is defined as $H = 1 - \frac{\alpha}{2}$. A process is determined to be LRD if $0.5 < \alpha < 1$. Several methods exist for estimating the Hurst parameter. In

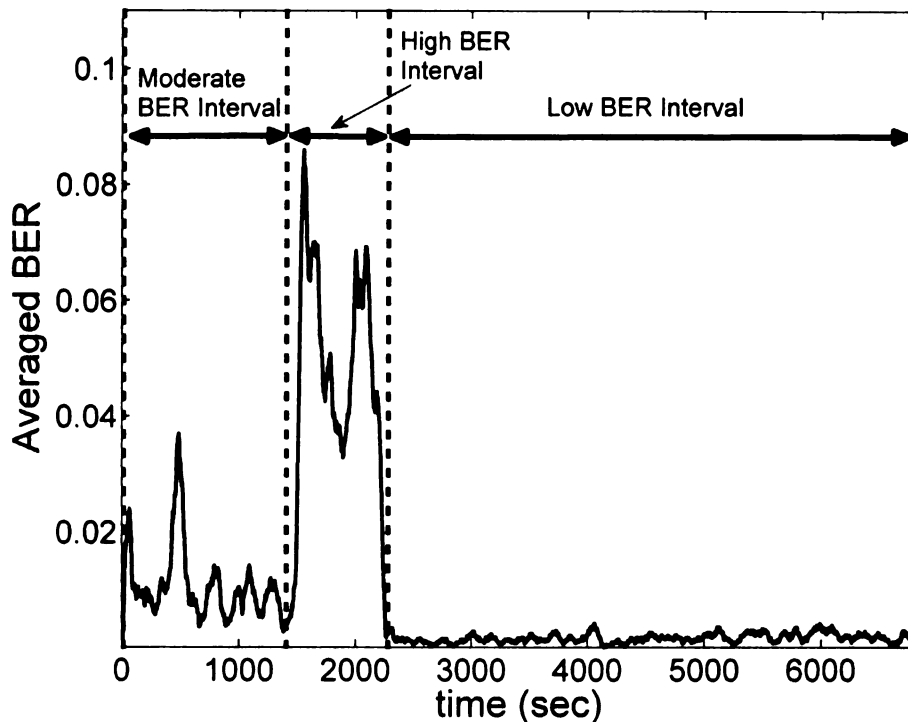


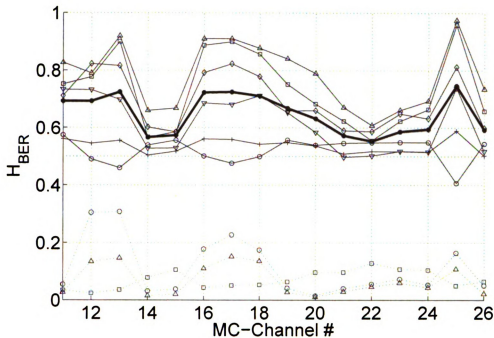
Figure 4.5: BER process of trace MC-25 after filtering by 600 point averaging filter.

this work we use the Aggregate Variance, the R/S method, the Periodogram method, the Absolute Value method, the Abry-Veitch estimator and the Whittle estimator, details for all of which can be looked up in [10]. We use the SELFIS tool developed by Karagiannis, Faloutsos and Riedi as part of their work [63]. Figures 4.6a and 4.6b plot H_{BER} , the Hurst parameter of the BER process for the MC and ME trace sets using all the above listed estimators. In addition, for each trace we plot the average Hurst parameter over all estimators (thick line). The plots of the Hurst parameter are accompanied by those of three other quantities, i.e. the average packet error rate (PER), the average packet loss rate (PLR) and the average conditional bit error rate (CBER). The average PER for a trace is the ratio of number of packets received with failed CRC to total number of transmitted packets. The average PLR is the ratio of the number of packets never received by the receiver to the total number

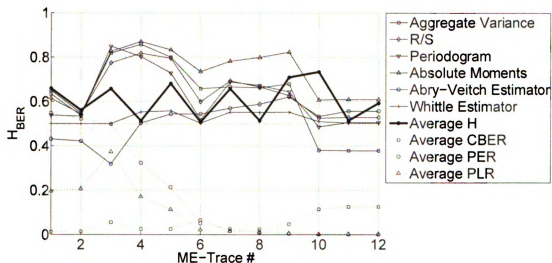
of packets transmitted. The average CBER is the expected value of the BER of all received packets whose CRC failed. As all plots show, there are significant variations in the estimates of the Hurst parameter computed by different methods. Estimates of obtained by the absolute moments and aggregate variance methods consistently provide the highest estimates, whereas the ones obtained by Abry-Veitch and Whittle estimators are consistently the lowest. The ones provided by the Periodogram and R/S methods fall approximately in the middle of this range. This observed ordering holds true for H_{BER} across MC and ME trace sets.

4.3.1 Observations For MC Trace Set

For the MC traces we observe that most estimates of H_{BER} rise in three distinct places, peaking for channels 13, 17 and 25. Channels 13, 17 and 25 occupy the centers of the spectrum bandwidth occupied by interfering channels 1, 6 and 11 of IEEE 802.11b/g. The degree of interference experienced by channels is measured by the average PER and average PLR. Thus H_{BER} bear some correlation with the average PER and average PLR and, hence, the interference from nearby 802.11b/g WLANs. The Abry-Veitch estimator is the only estimator that defies this observation and in fact exhibits inverse correlation with average PER and average PLR. Estimates of H_{BER} for channels experiencing strong interference are high enough to conclude LRD in the packet-level error process. But at the same time we observe that for 802.15.4 channels that do not experience interference from 802.11b/g WLANs estimates of H are close to or below the critical threshold of 0.5. It appears that inherently the packet-level error process in 802.15.4 channels is not LRD, except when it is subjected to 802.11b/g interference. The explanation for this behavior comes from Karagiannis, Faloutsos and Riedi [63] who concluded that periodic interference in an otherwise memoryless channel can give the appearance of LRD (when LRD is



(a) MC trace set.



(b) ME trace set.

Figure 4.6: Plots of estimates of the Hurst parameter obtained using various techniques along with their average BERs, PERs and PLRs.

determined by estimating H). Moreover, LRD is also unlikely to exist in a process if estimates of H_{BER} by various methods do not converge, which is surely the case in our analysis. WLANs periodically transmit a beacon frame every $100msec$ that is used to synchronize the network. It is the periodic interference from these beacons

frames that is giving the appearance of LRD in the packet-level error process.

4.3.2 Observations For ME Trace Set

Recall that all these traces in the ME set were collected in channel 26 to reduce effects of interference. Most traces in this set experience lower average PER and average PLRs. For many traces the average Hurst parameter often remains close to 0.5. Nevertheless, as in the case of the MC trace set, the estimates of H_{BER} still do not clearly converge to one value, and when they do it is in a range very close to 0.5. The lack of consensus among estimates, and the close proximity to 0.5 otherwise leads us to believe that the channels in the ME trace set are also not LRD.

4.4 Memory Length Measurement By Relative Mutual Information

4.4.1 Shannon Information Measures

For random variables X and Y , with probability density functions (PDF) $p_X(x)$ and $p_Y(y)$ and joint PDF $p_{XY}(x, y)$, the mutual information $I(X; Y)$ is defined as,

$$I(X; Y) = \sum_{\forall x, y; p(x, y) > 0} p_{XY}(x, y) \log_2 \left(\frac{p_{XY}(x, y)}{p_X(x)p_Y(y)} \right). \quad (4.6)$$

Mutual information [32] can also be understood more intuitively in terms of entropy as,

$$I(X; Y) = H(X) + H(Y) - H(X, Y). \quad (4.7)$$

More generally, mutual information is defined between two sets of random vari-

ables. If random variables X and Y in equation 4.6 are replaced by sets of random variables X_1, X_2, \dots, X_n and Y_1, Y_2, \dots, Y_m , respectively, then $p_X(x)$, $p_Y(y)$ and $p_{XY}(x, y)$ are replaced by respective joint PDFs $p_{X_1 X_2 \dots X_n}(x_1, x_2, \dots, x_n)$ and $p_{Y_1 Y_2 \dots Y_m}(y_1, y_2, \dots, y_m)$, and the joint PDF of all random variables $p_{X_1 X_2 \dots X_n Y_1 Y_2 \dots Y_m}(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m)$.

4.4.2 Description: Relative Mutual Information

We are essentially faced with the challenge of evaluating;

- How much information random variable Y can provide about another random variable X ;
- While at the same time providing a measure of the remaining uncertainty about X .

The Shannon mutual information $I(X; Y)$ achieves the first goal. However, mutual information is not restricted to a fixed range. Thus, an evaluation of $I(X; Y)$ does not give us a sense of how much information about X or Y remains unknown. We use relative mutual information (RMI), denoted $RMI(X; Y)$, previously described by [29] and adopted by us for the purpose of memory length measurement in [57]. RMI is defined as;

$$RMI(X; Y) = \frac{I(X; Y)}{H(X)}. \quad (4.8)$$

Note that while $I(X; Y)$ is a symmetric measure, by definition RMI is non-symmetric, i.e. $RMI(X; Y) \neq RMI(Y; X)$ because $H(X) \neq H(Y)$. Since $I(X; Y) \leq \min(H(X), H(Y))$, the RMI's value is limited to the interval $[0, 1]$. An RMI close to 1 implies Y contains most of the information contained in X , leaving little uncertainty, while an RMI close to 0 implies the opposite.

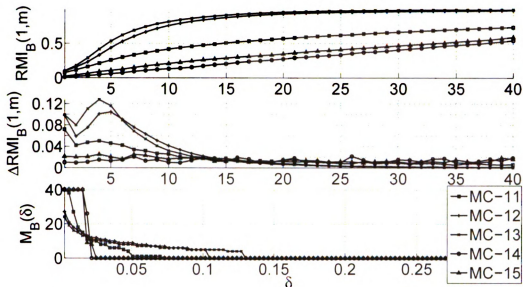


Figure 4.7: For traces MC-11, MC-12, MC-13, MC-14 and MC-15 each subfigure, (from top to bottom): [Top] $RMI_B(1,m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1,m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1,m)$.

In the current context we replace X by the BER process $B^{(0)}$, and replace Y by the BER process of preceding packets $B^{(1)}$, i.e. a one-right shifted version of $B^{(0)}$. Then $RMI(B^{(0)}; B^{(1)})$ measures the amount of information that a packet's BER $\beta[n]$ shares, on average, with the following packet's BER $\beta[n-1]$. A natural extension of this measure would be to include more than just the immediately following packet, but include any arbitrary number m such that,

$$RMI(B^{(0)}, B^{(1)}, B^{(2)}, \dots, B^{(m)}) = \frac{I(B^{(0)}; B^{(1)}, B^{(2)}, \dots, B^{(m)})}{H(B^{(0)})}. \quad (4.9)$$

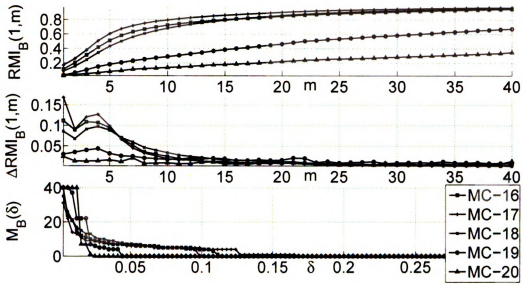


Figure 4.8: For traces MC-16, MC-17, MC-18, MC-19 and MC-20 each subfigure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$.

Since we are assuming the BER process to be wide sense stationary, the RMI becomes a function of m , the number of immediately following measurements. We use the abbreviated notation $RMI_B(1, m)$ for the RMI in equation 4.9. Recall that $RMI_B(1, m)$ is a function of the Shannon mutual information (and hence the joint PDF) of BER processes B through $B^{(m)}$. Ideally, once m exceeds the channel's memory length M_B , B become independent of $B^{(m)}$ and the Shannon mutual information equation 4.6 will drop to zero, producing a zero RMI in equation 4.8. But as in the case of correlogram based analysis, the presence of slow fading complicates interpretation of RMI. Therefore, for the duration of the slow fade a packet's BER will remain weakly correlated with its followers. Hence, after $m > M_B$, this translates into a slowly increasing $RMI_B(1, m)$ for successive values of m until it finally becomes 1. However, for the problem at hand we are not particularly interested in slow fades but the memory length of fast fades. Therefore, we define channel memory

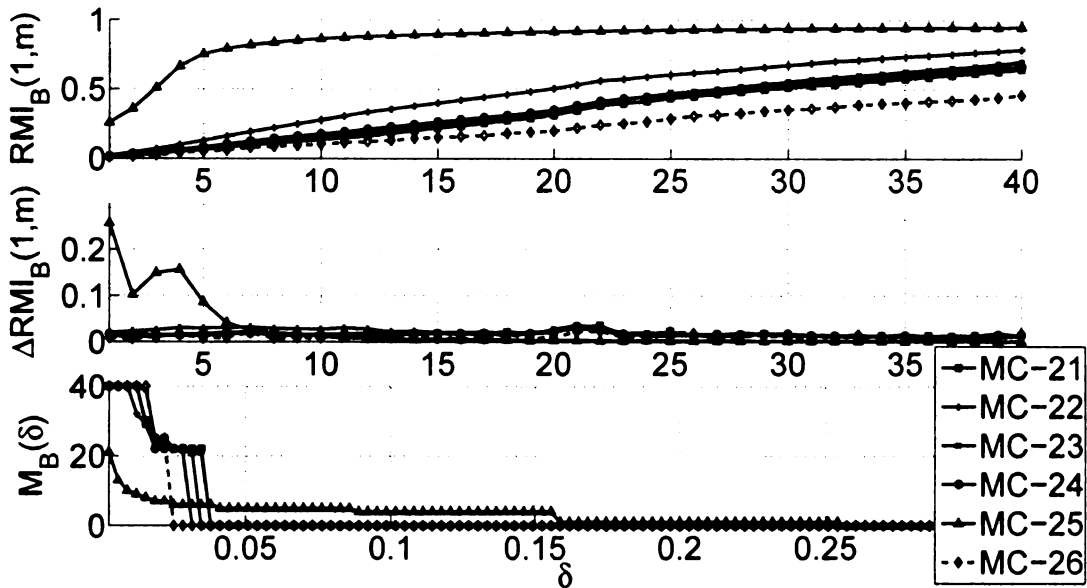


Figure 4.9: For traces MC-21, MC-22, MC-23, MC-24, MC-25 and MC-26 each sub-figure, (from top to bottom): [Top] $RMI_B(1, m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1, m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1, m)$.

length M_B as the time it takes for the $RMI_B(1, m)$ function to rise to a level after which it grows only slowly. Then channel memory M_B is determined by computing the RMI function for increasing values of m and setting M_B equal to the largest lag m for which the amount of additional RMI provided by including the $(m + 1)$ delayed BER becomes smaller than δ , i.e.

$$M_B(\delta) = \max_m (|RMI_B(1, m - 1) - RMI_B(1, m)| \geq \delta) \quad \forall m \in [1, \infty]. \quad (4.10)$$

Thus the memory length $M_B(\delta)$ is a function of δ , the significant RMI increment threshold. That leaves us with the choice an appropriate value of δ . RMI is understood to be the fraction of Shannon information of one random variable contained

collectively in another set of random variables. Thus, in terms of RMI the memory length of the BER process can then be understood as the lag m after which every subsequent BER measurement will contribute less than $100 \times \%$ of new information about $B^{(0)}$. The feature that makes RMI attractive for use as a tool for memory measurement is not the fact that provides an unequivocal measurement of the BER process' memory length. Like the correlation coefficient before it, the RMI depends on the subjective selection of a threshold that determines the cutoff between significance and insignificance. Rather, its strength lies in the fact that there is a clear interpretation of the threshold (in this case δ) in information theoretic terms, e.g. $\delta = 0.15$ implies that on average, an M_B delayed measurement contains at least 15% information about the current measurement. Another advantage of using RMI over the commonly used correlation coefficient is that the changing trends operating on the timescales of slow fades that cause non-stationary behavior of the BER process do not complicate the analysis.

4.4.3 Discussion

The plot at the top in figure 4.7 is the $RMI_B(1, m)$ of traces of IEEE 802.15.4 links operating channels 11, 12, 13, 14 and 15 for a range of lag values m . The middle figure plots the $\Delta RMI(1, m)$ functions of the same traces for the same range of lags. The bottom plot in figure 4.7 shows the memory length as a function of increments δ . Figures 4.8 and 4.9 plot the same quantities for the remaining channels in the MC trace set. Similarly, figures 4.10 and 4.11 plot $RMI_B(1, m)$, $\Delta RMI_B(1, m)$ and $M_B(\delta)$ for the traces in the ME trace set. For our reading of the plots let us take $\delta = 0.15$,i.e. last measurement within memory length contributes at least 15% information. Then the plots of $M_B(\delta)$ for all traces adequately demonstrate that all 802.15.4 traces exhibit a memory length within a narrow range of 0 to 10

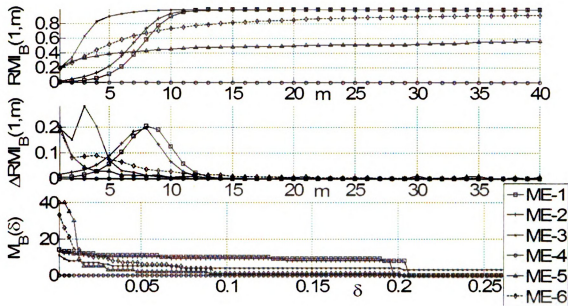


Figure 4.10: For traces ME-1, ME-2, ME-3, ME-4, ME-5 and ME-6 each subfigure, (from top to bottom): [Top] $RMI_B(1,m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1,m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1,m)$.

packets which, at the packet transmission rate of 10 packets per seconds used for trace collection, corresponds to a time period of 0 to 1sec. Thus, diversity in physical environments and channel selection do not appear to have any significant bearing on the duration of fast fades, as measured by RMI.

On a side note, there is a major practical challenge to the online and/ or real-time computation of RMI. The computation of $RMI_B(1,m)$ is based on an $m + 1$ dimensional joint PDF of all processes $B^{(0)}, B^{(1)}, B^{(2)}, \dots, B^{(m)}$. Populating such a high dimensional PDF requires large data set even for moderate values of m . Collecting data points to fill a modest 10 dimensional PDF takes significant time, especially when considering that IEEE 802.15.4 is a low rate communication standard. This makes this methodology especially unsuitable for applications in which this computation is to be performed online. We repeatedly performed this computation

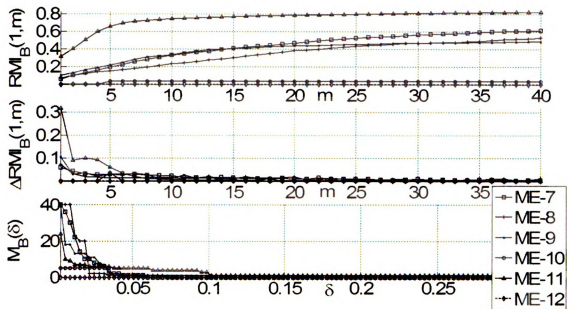


Figure 4.11: For traces ME-7, ME-8, ME-9, ME-10, ME-11 and ME-12 each subfigure, (from top to bottom): [Top] $RMI_B(1,m)$ of BER process observed in MC traces for lag m varying from 1 through 40. [Middle] $\Delta RMI_B(1,m)$ of BER process for the same channel traces. [Bottom] The memory length M_B plotted as a function of δ , the increments in $RMI_B(1,m)$.

offline on different data sets, and found that even on a mid-level server class machine the computation of the RMI function based on 50,000 data points for m ranging from 1 to 40 takes more than 30 minutes.

4.5 Conclusions

1. All IEEE 802.15.4 channels, regardless of channel selection or physical environment, exhibit a memory length of at most 2 bits and 2 symbols, respectively.
2. Based on the correlation function and LRD based analysis we conclude that various estimates of Hurst parameter may or may not detect packet level memory in 802.15.4 channels. The 'memory', however, is not due to the channel's inherent properties at those frequencies, but due to interference from 802.11b/g

traffic and beacon frames, i.e. if interference is periodic the channel appears to have memory, if it is not periodic there is no memory.

3. The Abry-Veitch and Whittle estimators' consistent relative insensitivity to changes in average PER, average PLR, average CBER and interference across different traces leads us to conclude that they are better measures of the 802.15.4 channel's inherent degree of LRD.
4. The aggregate variance, R/S, periodogram, and absolute moment estimators' strong dependence on average PER and average PLR leads us to conclude that these estimators are good detectors of (WLAN) interference.
5. The average CBER to which a packet is subjected by a channel is inversely related to the average PER/ average PLR. Thus, it appears that interference produces higher BERs in packets than do channel fading effects.
6. We present use of RMI as a standardized version of the mutual information and apply it to the BER process captured in bit traces. We observe that interference free 802.15.4 channels are memoryless, while channels experiencing significant interference from 802.11b/g networks sharing the 2.4GHz ISM band, a common source of interference, have "true" memory lengths varying in the narrow range of 0 to 1sec.

Chapter 5

A Statistical Measure Of Network Lifetime For Wireless Sensor Networks

5.1 Introduction

One of the most fundamental challenges in wireless sensor networks (WSN) is the short and often limited supply of energy. Due to the disposable nature of WSN nodes power sources are often non-replenishable or replenishable very slowly at best. In either case this forces prudent use of battery power for all operations. Since WSNs can be spread over large geographical areas multi-hop communication is employed in transmitting sensor measurements to the data collection point, also known as the base station. The problem is further compounded by the many-to-one traffic flow pattern that is imposed by the data collection process. It produces a traffic hot-spot or bottleneck around the base station and, depending on the positioning of nodes, in other regions of the network as well. This phenomenon is called the reachback and was investigated by Barros and Servetto in [9]. If the same shortest-path-first (SPF) routes [30] are maintained, as in the case of most present day mobile ad-hoc networks (MANET) and WSN routing algorithms, nodes start running out of energy. Nodes gradually start disappearing from the sensor network beginning with those handling the highest traffic volume, the ones communicating directly with the base station. Such nodes are referred to as critical nodes. Each node going offline will reduce coverage provided by the WSN. Eventually all nodes in communication range of the base station will run out of power and the base station will stand disconnected from all surviving nodes, effectively dropping situational awareness at the base station to zero. While the nature of the traffic flow makes the degradation of system capabilities over time inevitable, it is desirable to make it as graceful as possible. This leads us to consider a solution that will redistribute the volume of traffic handled by critical nodes more evenly. It is noted here that any deviation from routes selected using a routing algorithm based on SPF means selecting a route that is suboptimal in the traditional, greedy sense (of which there are many even for small sets of critical nodes).

The above discussion provides us with two objectives; 1) reducing the differences between energy consumption rates of nodes, 2) but at the same time keeping the average energy consumption rate low. This requires the joint minimization of both objectives. Since these two objectives run counter to each other the selection of an operating point is a trade-off between mean and variance of power consumption rates.

The remaining chapter is organized as follows. Section 5.2 reviews some recent efforts that attempt to increase longevity of WSNs and positions our work. Section 5.3 describes our interpretation of network lifetime and how the advantages of its use as an objective over previous definitions. Section 5.4 describes our network model and introduces terminology. Section 5.5 formulates the problem as a quadratic program. Section 5.6 describes some results for small scale examples and section 5.7 concludes the chapter.

5.2 Previous Work

The volume of works spanning energy efficient routing protocols for WSN is extensive. Early WSNs borrowed routing protocols from ad-hoc wireless networks and MANETs. The routes selected by dynamic destination-sequenced distance-vector (DSDV) [99], dynamic source routing (DSR) [61], ad-hoc on-demand distance vector (AODV) [100] and directed diffusion [58] protocols are "optimal" only in a greedy, SPF sense which worked well enough for networks without power constraints. The performance of a system using these protocols is as much subject to the reachback problem as one making use of naïve SPF routing.

In [25] Chang and Tassiulas formulate the lifetime problem as a minimax linear program (LP) that seeks to maximize the minimum sensor lifetime. However, while the approach is theoretically sound and provides a bound for any attempt at maximizing that particular notion of network lifetime, there are scalability problems which

are exacerbated by the very large number of optimization variables for which the LP is solved. This was followed by several other LP formulations of the lifetime problem [11],[107],[135], [78], all based on the same or similar meaning of network lifetime, of varying degrees of usability. In [6] Baek and de Veciana proposed a proactive multi-path routing scheme by introducing joint minimization of the "spreading factor" ω , and the probability of battery depletion of a sensor which is similar to Ilyas and Radha's parallel work in [53] that uses average and variance of power consumption rates in sensors. However, Baek and de Veciana's mechanism used for path discovery does not take into account several other available communication links. While the authors demonstrate the improvements offered by their energy balancing algorithm in networks with any-to-any data flow the proposed solution does not seem to offer an improvement when the traffic flow is many-to-one/ all-to-one. More recently Khanna, Liu and Chen [65] took an evolutionary approach. However, this was marked by a high complexity due to the inherent nature of Genetic algorithms and poses challenges to scalability.

5.3 Novelty Of Approach

The bulk of previous work on the lifetime problem defines network lifetime as the time until the first sensor runs out of power. The rigidity of this definition is of advantage because it provides a clear objective function for optimization. However, any set of routes that deviate from greedy SPF routes produce an increase in the power consumption rates of some nodes, decreasing their individual lifetimes. The prior LP approaches that maximize the minimum sensor lifetime are no exception. However, by solely focusing on one sensor's lifetime (the minimum lifetime sensor), it ignores the cost, the decrease in other sensors' lifetimes at which this maximization is achieved. This also implies a higher rate of failure of sensors as a network approaches the end of

its life, as defined under LP minimax problem formulations. This definition disregards the inherent redundancy in WSNs and their ability to cope with a limited device failure rate. In this chapter we use a notion of lifetime that takes these "shortcomings" into account. We propose the joint use of two statistics of P , the random variable modeling the energy consumption rates of sensors in a WSN, namely;

1. $E[P]$: the mean of P .
2. $Var[P]$: the variance of P .

The problem then becomes a joint-minimization problem. This notion of lifetime takes into account the lifetimes of the entire population of sensors making up the network. Some previous solutions such as Singh, Woo and Raghavendra [113] describe the independent minimization of only the variance of node power levels to extend the lifetime of a network. Minimization of $E[P]$ alone is achieved by SPF routing protocols based on energy as a cost metric. However, selecting routes based solely on the minimization of $E[P]$ will inevitably lead to the aforementioned reachback problem where sensor nodes closer to the base station transmit packets at significantly higher frequency compared to sensor nodes farther away. This problem can be formulated as a budget constrained allocation problem: The minimization of $Var[P]$ defined as equation 5.1 is subject to the constraint equation 5.2 that $E[P]$ be less than some maximum budget value $E[P]^*$, where P_i is the energy consumption rate of node $n_i, \forall i \leq N$, and N is the number of nodes.

$$\min Var[P] = \frac{\sum_{i=1}^N (P_i - E[P])^2}{N} \quad (5.1)$$

Subject to,

$$E[P] = \frac{\sum_{i=1}^N P_i}{N} \leq E[P]^* \quad (5.2)$$

Using the constraint formulation highlighted above, we can summarize our optimization framework as identifying the set of routes throughout a given WSN such that the following is satisfied: $\min_{E[P] \leq E[P]^*} \{Var[P]\}$.

5.4 Network Model

To model devices we are adhering to the IEEE 802.15.4 low-rate wireless personal area network (LR-WPAN) draft standard [5]. The standard defines two device classes, reduced-function devices (RFDs) and full-function devices (FFDs). Since RFDs are incapable of performing routing functions and since we are investigating a routing solution we use the term node to refer to FFDs only.

A WSN consists of $N_{FFD} + N_{RFD}$ sensors and one base station. Since we will only be dealing with FFDs in a routing solution we abbreviate N_{FFD} by N . FFD routers are numbered 1 through N and denoted by n_1 through n_N . The base station is assigned FFD ID 0 and denoted by n_0 . Furthermore we are assuming the nodes participating in the WSN to be capable of measuring received signal strength indication (RSSI) and link quality indication (LQI) for received packets and adjusting their transmission power as laid down in the standard. Nodes are capable of varying transmission power at run-time on a packet-by-packet basis. Transmission power is chosen as a function of the spatial separation between transmitter and receiver.

For the channel model we adhere to the model of the 802.15.4 Physical Channel Modeling Subgroup [86]. We are modeling the communication range of a sensor in each direction by a Gaussian random variable Z with mean μ_Z and variance σ_Z^2 . As a result, some links in the network are unidirectional. Each link from a node n_i to another node n_j is assigned a link cost C_{ij} obtained by,

$$c_{ij} = \frac{d^{\alpha}(n_i, n_j) + Z(\mu_Z, \sigma_Z^2)}{B_i} \geq 0. \quad (5.3)$$

Here α is an exponential decay factor which varies for wireless communication from 2 to 4, depending on the type of environment. An assignment of c_{ij} denotes the absence of a direct link from n_i to n_j . Here $d(a, b)$ is a function that returns the Euclidean distance between the nodes provided in the argument and B_i denotes the battery reserve of n_i . The cost c_{ij} is the cost of transmitting the packet from n_i at a power that ensures reception at n_j with a minimum signal-to-noise ratio (SNR) [104] of $SINR_{min}$ with required certainty.

For the MAC model we refer to the 802.15.4 LR-WPAN standard [5]. The standard is geared towards energy-conservation. Since the protocol supports a slotted collision sense/multiple access (sCSMA) mode based on time division multiple access (TDMA) we can make the reasonable assumption that there is little interference as long as link utilization remains under the maximum possible data throughput rate.

5.5 Quadratic Program Formulation

In view of the new understanding of network lifetime and the associated objective function we formulate the lifetime maximization program as a quadratic program (QP). Let Q_i denote the data produced by n_i for transmission to n_0 . This value can depend on the spatial distribution of the entropy [32] of the underlying event being sensed by the WSN. Let q_{ij} denote the flow from n_i to its neighbor n_j with which it communicates directly. Furthermore, $q_{ij} \geq 0$. If c_{ij} denotes the cost of communicating a unit of information directly from n_i to n_j , then the cost incurred by n_i in communicating q_{ij} to n_j is $c_{ij}q_{ij}$. We are not assuming the use of any in-network processing that might violate the conservation of flow in the network. This

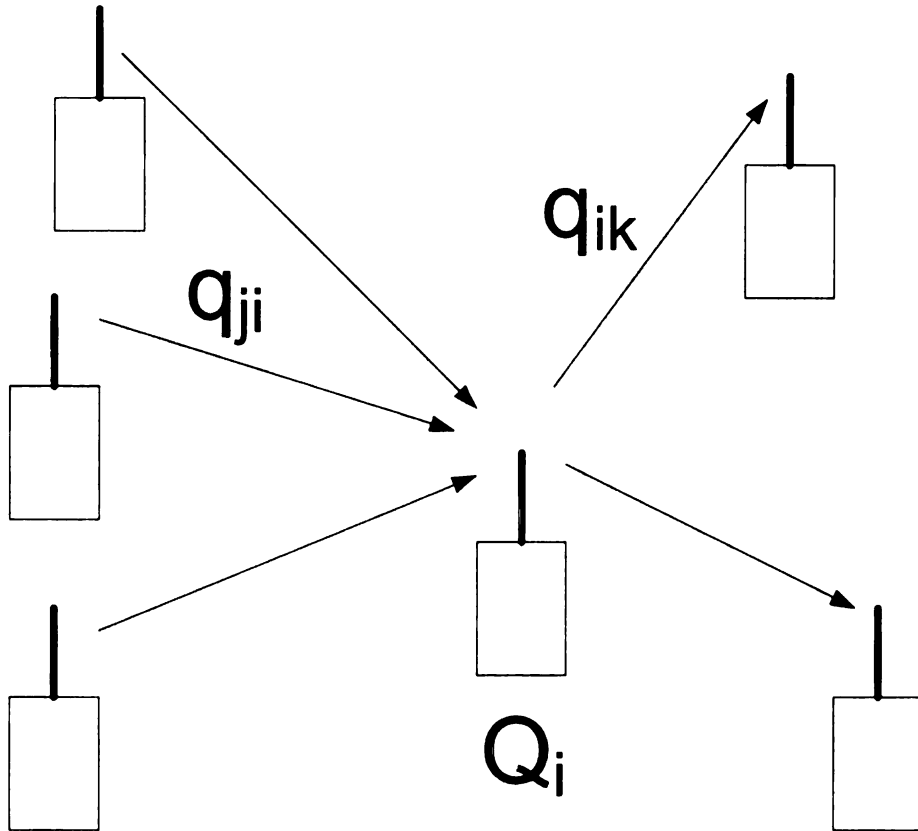


Figure 5.1: The law of conservation of flow requires that the sum of incoming flows $q_{j,i}$ and data Q_i generated at node n_i must equal the sum of all outgoing flows $q_{i,k}$.

condition is illustrated in figure 5.1. Then, if S_i denotes the set of nodes n_i can relay its flow to, the condition of flow conservation for any single node n_i can be expressed as,

$$Q_i + \sum_{i \in S_i} q_{ji} = \sum_{j \in S_i} q_{ij}, \forall 1 \leq i \leq N. \quad (5.4)$$

Note that the base station n_0 has been deliberately excluded from the condition of flow conservation in equation 5.4 since it is a consumer of flow. Since links between nodes are not necessarily bi-directional, hence generally $c_{ij} \neq c_{ji}$. If m_i denotes the cardinality of S_i , then individual elements of S_i are referred to by $S_i(j)$ where

$1 \leq j \leq m_i$. For the purpose of simplified notation elements within S_i are assumed to be sorted in ascending order of their node IDs. The total number of links in the network is denoted by $M = \sum_{i=1}^N m_i$. We formulate the problem as a quadratic program in matrix form. Let \underline{q} denote the flow vector that is to be optimized and \underline{Q} the flow generation vector as in.

$$\underline{q} = \left[q_{1,S_1(1)} \cdots q_{1,S_1(m_1)} q_{2,S_2(1)} \cdots q_{2,S_2(m_2)} q_{N,S_N(1)} \cdots q_{N,S_N(m_N)} \right]^T \quad (5.5)$$

$$\underline{Q} = [Q_1 Q_2 \cdots Q_N]^T \quad (5.6)$$

Let the separable cost matrix C be defined as,

$$C = \begin{bmatrix} c_{1,S_1(1)} & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ c_{1,S_1(m_1)} & 0 & \cdots & 0 \\ 0 & c_{2,S_2(1)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & c_{2,S_2(m_2)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & c_{N,S_N(1)} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & c_{N,S_N(m_N)} \end{bmatrix}^T \quad (5.7)$$

C contains the same information as the adjacency matrix of the network, but is organized in a sparser and separable fashion more suitable for later use. Since the bulk of power consumed in sensors is due to data transmission we are associating a cost

only to the data transmission process. Power consumption for reception is usually constant and significantly less than transmission power. Since $P_i = \sum_{j \in S_i} c_{ij} q_{ij}$ the objective function of minimizing the variance of power consumption in equation 5.1 can be expressed in the terms defined above as,

$$\begin{aligned}
& \min \frac{1}{N} \sum_{i=1}^N \left[\sum_{j \in S_i} c_{ij} q_{ij} - \frac{1}{N} \sum_{i=1}^N \sum_{j \in S_i} c_{ij} q_{ij} \right]^2 \\
& = \min \sum_{i=1}^N \left[\left(\sum_{j \in S_i} c_{ij} q_{ij} \right)^2 + \left(\frac{1}{N} \sum_{i=1}^N \sum_{j \in S_i} c_{ij} q_{ij} \right)^2 \right. \\
& \quad \left. - \frac{2}{N} \left(\sum_{j \in S_i} c_{ij} q_{ij} \right) \left(\sum_{i=1}^N \sum_{j \in S_i} c_{ij} q_{ij} \right) \right] \quad (5.8)
\end{aligned}$$

Note that we dropped the $\frac{1}{N}$ term. Equation 5.8 can be rewritten in matrix form as,

$$\begin{aligned}
& \min (\underline{C}\underline{q})^T \underline{C}\underline{q} + \frac{1}{N^2} (\underline{1}^T \underline{C}\underline{q})^2 - \frac{2}{N} \\
& = \min \underline{q}^T \underline{C}^T \underline{C}\underline{q} + \left(\frac{1-2N}{N^2} \right) (\underline{1}^T \underline{C}\underline{q})(\underline{1}^T \underline{C}\underline{q}) \\
& = \min \underline{q}^T \underline{C}^T \underline{C}\underline{q} + \left(\frac{1-2N}{N^2} \right) ((\underline{C}\underline{q})^T \underline{1})(\underline{1}^T \underline{C}\underline{q}) \\
& = \min \underline{q}^T \left(\underline{C}^T \underline{C} + \left(\frac{1-2N}{N^2} \right) \underline{C}^T \underline{1}_{N \times N} \underline{C} \right) \underline{q}. \quad (5.9)
\end{aligned}$$

Here $\underline{1}$ denotes a vector of 1s. Since the second order coefficient matrix in equation 5.9 is symmetric this QP has a solution, provided that the constraints are well defined. The budget constraint in equation 5.2 can be expressed in matrix form as,

$$\frac{\underline{1}^T \underline{C}\underline{q}}{N} \leq E[P]^* \quad (5.10)$$

$$\underline{q} \succeq 0 \quad (5.11)$$

Finally we need to define the constraint based on the conservation of flow. We define a flow matrix F of size $N \times M$ as below.

$$F^T = \begin{bmatrix} f_{1,S_1(1)}(1,1) & \cdots & f_{1,S_1(1)}(N,1) \\ \vdots & & \vdots \\ f_{1,S_1(m_1)}(1,m_1) & \cdots & f_{1,S_1(1)}(N,m_1) \\ f_{2,S_2(1)}(1,m_1+1) & \cdots & f_{2,S_2(1)}(N,m_1+1) \\ \vdots & & \vdots \\ f_{2,S_2(m_2)}\left(1,\sum_{i=1}^2 m_i\right) & \cdots & f_{2,S_2(m_2)}\left(N,\sum_{i=1}^2 m_i\right) \\ \vdots & & \vdots \\ f_{N,S_N(1)}\left(1,\sum_{i=1}^{N-1} m_i+1\right) & \cdots & f_{N,S_N(1)}\left(N,\sum_{i=1}^{N-1} m_i+1\right) \\ \vdots & & \vdots \\ f_{N,S_N(m_N)}\left(1,\sum_{i=1}^N m_i\right) & \cdots & f_{N,S_N(m_N)}\left(N,\sum_{i=1}^N m_i\right) \end{bmatrix} \quad (5.12)$$

Elements $f_{a,S_b(c)}(d,e)$ for which $d = b$ are set to 1. All elements for which $d \in S_a$ are set to -1 . All remaining elements are set to 0. That allows us to express the condition for the conservation of flow like in equation 5.13.

$$F\underline{q} = \underline{Q} \quad (5.13)$$

Thus equations 5.9,5.10,5.11 and 5.13 constitute the QP formulation. Note that the base station or data collection point n_0 is exempted from the condition of conservation of flow. The nature of this formulation is such that the solution \underline{q}^* provides

the optimal distribution of traffic with which outgoing links should be utilized.

Any routing strategy that seeks to redistribute the traffic load deviates from the SPF routing strategy that provides minimum global energy consumption, thereby raising $E[P]$. The increase in Chang and Tassiulas' [25] defined network lifetime comes at the cost of decreased individual lifetimes of some other nodes in the network. However, the LP formulation is such that there is no control over the cost at which this increase in lifetime is achieved. A benefit of the QP formulation over previous LP formulations is that the objective function provides a global view of node consumption rates.

Unfortunately, the complexity of solving this QP even for moderate values of M and N is too high to be of interest for practical use. Nevertheless it provides us with a bound on the best possible solution given a set of constraints.

5.6 Results

We applied the QP formulation to some example networks of varying, yet manageable sizes of 10, 15 and 20 nodes. The nodes are randomly scattered in a square region of 10×10 dimensions. All nodes are assumed to have equal initial battery reserves $B_1(0) = B_2(0) = \dots = B_N(0) = 1$ with maximum transmission range $r = 10$. The Gaussian noise source producing irregular link costs is set to $Z(10, 2)$. The decay factor is taken $\alpha = 3$, typical of omni-directional antennas in open spaces. The spatial distribution of the sensed event's entropy is assumed to be uniform. Hence, all sensors are generating data at a uniform rate as well, i.e. $Q_1 = Q_2 = \dots = Q_N$.

To illustrate the QPs ability to offer a gradual tradeoffs of $Var[P]$ for $E[P]$. Each operating point in figures 5.2, 5.3 and 5.4 is obtained by successively relaxing the constraint in equation 5.10 by incrementally raising $E[P]^*$ and solving each resulting QP. As expected, each successive solution offers decreasing $Var[P]$. However, after a

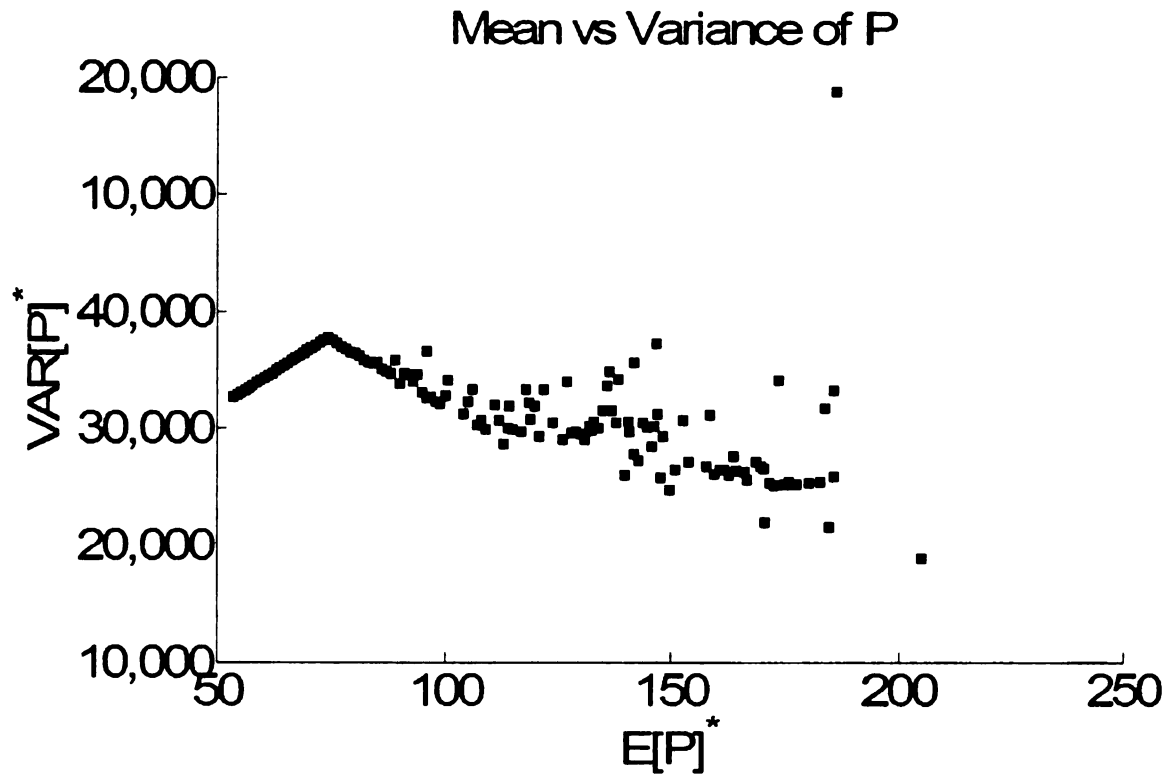


Figure 5.2: Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 10$.

certain point $Var[P]$ starts increasing again.

5.7 Conclusions

We propose a new definition of network lifetime consisting of $E[P]$ and $Var[P]$. This notion of network lifetime provides a more inclusive view of the power consumption of sensors across the network. The objective function offers an alternative view of network lifetime. We went on to formulate the optimization problem for the new objective function in the form of a QP and showed that a solution exists.

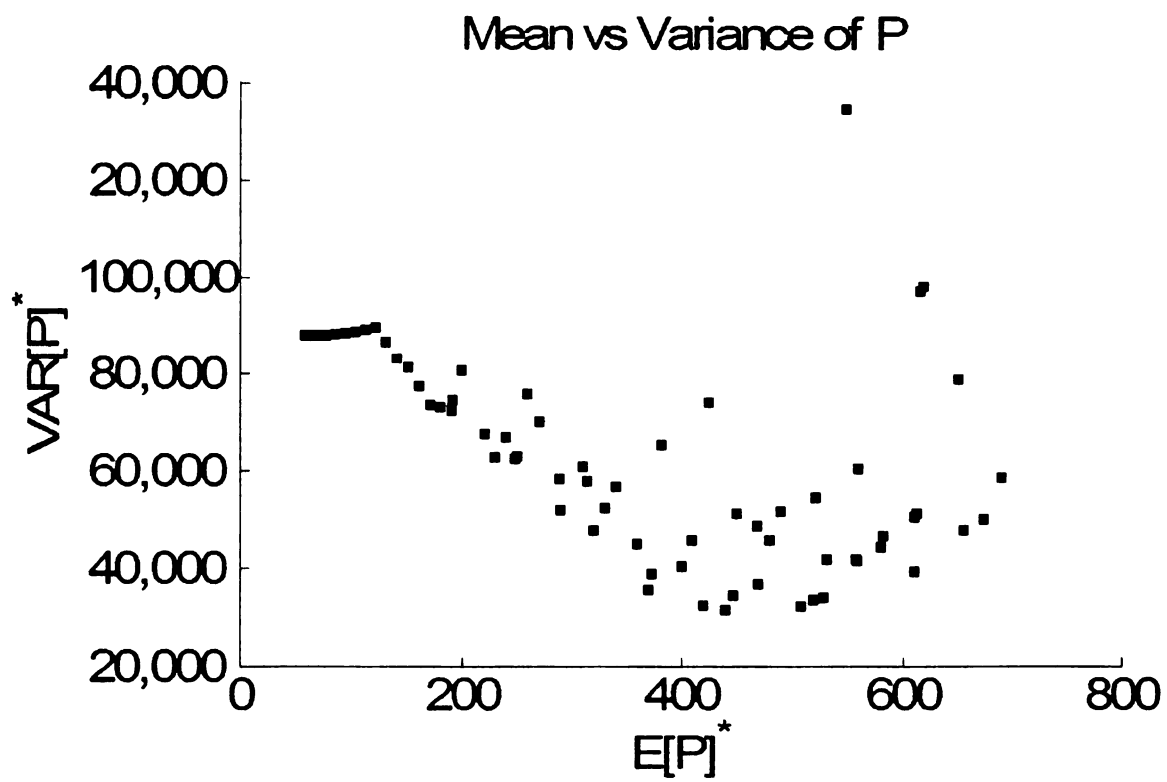


Figure 5.3: Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 15$.

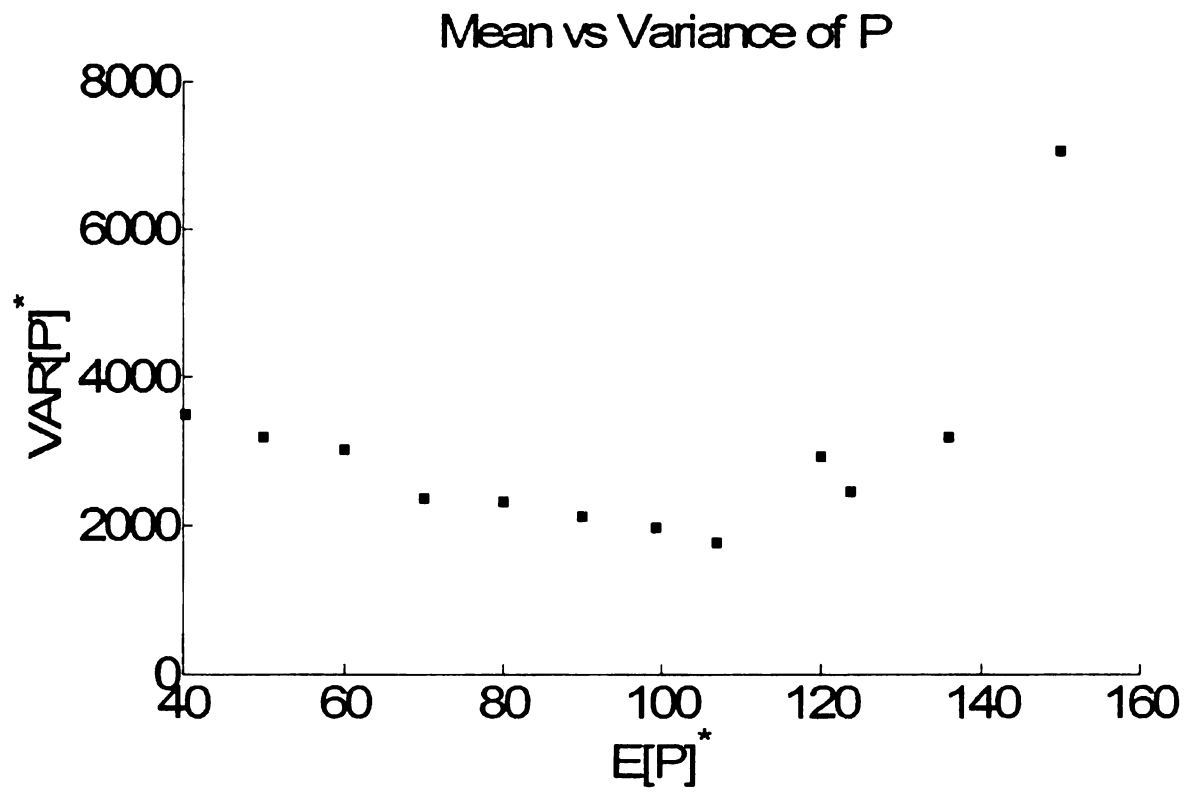


Figure 5.4: Tradeoff of $Var[P]$ versus $E[P]$ for a network with $N = 20$.

Chapter 6

A Dynamic Programming Approach to Maximizing Lifetime of Sensor Networks

6.1 Introduction

Wireless sensor networks (WSN) are the enabling technology for applications ranging from infrastructure protection and operation, emergency and crisis intervention, all the way to surveillance and environmental monitoring systems. One of the most fundamental constraints of WSN is the short and often limited supply of energy. Difficulties accessing sensors post-deployment, hostile deployment environments, and impracticality of performing maintenance operations on individual sensors requires making sensors disposable. Power sources are often non-replenishable or replenishable very slowly at best. In either case this forces prudent use of battery power for all operations. Since WSNs can be spread over large geographical areas, multi-hop communication is employed in transmitting sensor measurements from sensor nodes to the data collection point or base station. The problem is further compounded by the many-to-one traffic flow behavior witnessed in information gathering for in-situ and remote sensing applications. It produces a traffic hot-spot around the base station and other regions of the network with traffic bottlenecks. Barros and Servetto [9] called this phenomenon the *reachback effect*. Wan et al. [125] named it the *funneling effect*. If the same shortest-path-first (SPF) routes [30] are maintained, as in the case of most present day mobile ad-hoc networks (MANET) and WSN routing algorithms, some nodes will run out of energy sooner than others. Nodes will gradually start disappearing from the sensor network beginning with those handling the highest traffic volume, i.e. the ones communicating directly with the base station. These nodes are referred to as critical nodes. Eventually all nodes in communication range of the base station will run out of power and the base station will stand disconnected from all surviving nodes, effectively dropping available coverage to zero. It is desirable to stave off this event for as long as possible. This leads us to consider a solution that will redistribute the volume of traffic handled by critical nodes more evenly. It is

noted here that any deviation from routes selected using a routing algorithm based on SPF means selecting a suboptimal route (of which there are many, even for small sets of critical nodes). We discuss and compare several methods for obtaining ordered listings of suboptimal paths. A dynamic programming algorithm (DPA) then selects an optimal set of routes. The link cost metrics are derived from physical layer information, thus qualifying this route selection method as a cross-layer approach.

The above discussion provides us with the objective of reducing the differences between energy consumption rates of nodes while keeping the average energy consumption rate low. This requires the joint minimization of both objectives. Since these two objectives have conflicting requirements the selection of an operating point becomes a trade-off between mean and variance of power consumption rates. We evaluate a number of route discovery algorithms that produce paths other than the shortest paths between source and destination. This is followed by picking one path for each source-destination pair in a way that energy consumption is spread out, yet, kept low. The core contributions of this work are summarized below.

1. We propose mean and variance of the distribution of node power consumption rates as alternative optimization objectives for network lifetime.
2. We provide a low-complexity, dynamic programming formulation of the optimization problem rooted in operational rate-distortion theory.
3. We evaluate various algorithms for the discovery of paths deemed sub-optimal in the SPF-sense as inputs to the dynamic programming algorithm.

The rest of the chapter is organized as follows. Section 6.2 reviews some recent efforts that attempt to increase longevity of WSNs and positions our work. Section 6.3 describes our network model. Section 6.4 is the formal problem formulation. Section 6.5 describes four route discovery algorithms. Section 6.6 explains in detail

the working of the DPA inspired by operational rate-distortion (RD) theory. Section 6.7 presents an in-depth analysis of the performance of DPA and the route discovery algorithms used in conjunction with it. Finally, section 6.8 concludes this chapter.

6.2 Previous Work

The volume of works spanning energy efficient routing protocols for WSN is extensive. Early WSNs borrowed routing protocols from ad-hoc wireless networks and MANETs. The routes selected by dynamic destination-sequenced distance-vector (DSDV) [99], dynamic source routing (DSR) [61], ad-hoc on-demand distance vector (AODV) [100] and directed diffusion [58] protocols are "optimal" only in a greedy, SPF sense which worked well enough for networks without power constraints. These protocols are prone to the *reachback effect*. In [25] Chang and Tassiulas proposed an optimization approach based on linear programming. However, their formulation maximizes the minimum node lifetime because it defines the system lifetime as the time until the first node runs out of power. More recently, Baek and de Veciana proposed in [6] a proactive multi-path routing scheme which bears some similarity to our own interpretation of the problem by introducing joint minimization of a "spreading factor" ω , and the probability of battery depletion of a sensor which is very similar to Ilyas and Radha's parallel work in [53] that uses average and variance of power consumption rates in sensors. However, Baek and de Veciana's mechanism used for path discovery does not take into account several other available communication links. While the authors demonstrate the improvements offered by their energy balancing algorithm in networks with any-to-any data flow the proposed solution does not seem to offer an improvement when the traffic flow is many-to-one. More recently Khanna, Liu and Chen [65] took an evolutionary approach. However, this was marked by a high complexity due to the very nature of Genetic algorithms and poses challenges to scal-

ability. As we will show in the following sections the DPA proposed here resolves traffic hotspots in a WSN irrespective of the nature of traffic flow.

6.3 Network Model

6.3.1 Device Model

To model devices we are adhering to the IEEE 802.15.4 low-rate wireless personal area network (LR-WPAN) draft standard [5]. The standard defines two device classes, reduced-function devices (RFDs) and full-function devices (FFDs). RFDs feature only a limited implementation of the features defined by the standard. They are capable of associating and communicating with FFDs only and are incapable of performing routing functions. FFDs on the other hand have a full implementation of the standard and are capable of associating and communicating with both FFDs and RFDs. FFDs are also capable of performing routing functions. Since the DPA is a routing algorithm and only FFDs are capable of performing routing functions all nodes in our network model are assumed to be FFDs.

A WSN consists of N FFD sensors capable of performing routing functions, additional RFD sensors and one base station. FFD routers are numbered 1 through N . The base station is assigned FFD ID 0. Furthermore we are assuming the nodes participating in the WSN to be capable of measuring received signal strength indication (RSSI) and link quality indication (LQI) for received packets and adjusting their transmission power as laid down in the standard. Moreover, nodes are capable of varying transmission power at run-time on a packet-by-packet basis. Transmission power is chosen as a function of the spatial separation between transmitter and receiver and the signal decay factor described next.

6.3.2 Link Model

We are employing and adhering to the channel model proposed by the 802.15.4 Physical Channel Modeling Subgroup in [86]. Early exploratory work by Ilyas and Radha [53] assumed the simple disk model for a node's communication range. A consequence of this model was that it made all links bidirectional which is not necessarily true in real wireless networks. We are modeling the communication range of a sensor in each direction by a Gaussian random variable Ξ with mean μ_{Ξ} and variance σ_{Ξ}^2 . Each link from a node n_i to another node n_j is assigned a link cost $L_{i,j}$. The link cost is a function of the energy cost of receiving, processing and (re-)transmitting a packet (reliably) and the remaining battery level at the transmitter. Although the total energy consumed is the sum total expended in receiving, processing and transmitting a packet, this total is dominated by the energy of the transmission step. The cost of receiving a packet depends on the platform used for a sensor and is approximately constant for equally sized packets and the processing energy is negligible in comparison. The link cost $L_{i,j}$ in equation 6.1 is the cost to node n_i of transmitting a packet to node n_j , that is $d_{i,j}$ distance away. For an omni-directional antenna the decay factor α is usually around 3 [86]. $L_{i,j}$ also takes into account the remaining battery level in a node. The sum of reception, processing and transmission cost is divided by the fraction of remaining battery level B_i .

$$L_{i,j} = \begin{cases} \frac{d_{i,j}^{\alpha}}{B_i} & \text{where } d_{i,j} \leq c_i \\ \infty & \text{where } d_{i,j} > c_i \end{cases} \quad (6.1)$$

Table 6.1: Symbols and notation.

<i>Symbol</i>	Description
n_0	Base station
n_i	FFD sensor node with identifier i
$\Xi_{i,j}$	Maximum communication range of n_i in the direction of n_j .
$G(V, A)$	Graph consisting of vertex set V connected by directed edge set A .
$A_{i,j}$	Directed edge from n_i to n_j .
T_i	Rate of traffic generated at n_i .
$P_{i,j}^k$	k -th best path from n_i to n_j .
E_i	Energy consumption rate of n_i under the global traffic flow under consideration.
μ_E	Sample mean of energy consumption rates E_i for $1 \leq i \leq N$.
σ_E^2	Sample variance of energy consumption rates E_i for $1 \leq i \leq N$.
$SP(i \rightarrow j, G)$	Shortest path from n_i to n_j on graph G .
$L_{i,j}$	Link cost on n_i for link from n_i to n_j .
B_i	Fraction of total battery power remaining in n_i .

Then the power consumption rate of a node $n_i \forall 1 \leq i \leq N$ is denoted E_i and is defined in terms of link cost terms like in equation 6.2,

$$E_i = \sum_{n=1}^N \sum_{\substack{j=1 \\ A_{i,j} \in P_{n,0}^{k_i}}}^N L_{i,j} \text{ for all } 1 \leq i \leq N \quad (6.2)$$

6.4 Problem Formulation

WSNs serve to provide situational awareness and collect data from the entire region covered by them. An area is said to be covered as long as there is at least one sensor taking measurements from it. Data reporting of an area can stop due to two reasons, 1) The sensor providing coverage runs out of power, and 2) the sensor is unable to route its data to the base station due to a partition / fragmentation of the network. When a node stops communicating data to the base station there is a drop in coverage. Therefore, the utility of a WSN is related to the time period for which a WSN can maintain minimum coverage. The skewed distribution of energy consumption rates in sensors requires a redistribution of traffic load that,

1. Produces a more evened out traffic load across sensors.
2. Yet, at the same time, guarantees low total energy consumption.

In mathematical terms, if E is a random variable modeling the energy consumption rates of sensors in a WSN, we aim to minimize its sample variance σ_E^2 (defined in equation 6.3) while keeping its sample mean μ_E low (defined in equation 6.4).

$$\sigma_E^2 = \frac{\sum_{i=1}^N (E_i - \mu_E)^2}{N} \quad (6.3)$$

$$\mu_E = \frac{\sum_{i=1}^N E_i}{N}. \quad (6.4)$$

This leads to a joint-minimization problem. Minimization of μ_E alone is achieved by SPF routing protocols based on energy as a cost metric. However, selecting routes based solely on the minimization of μ_E will inevitably lead to a situation where sensor nodes closer to the base station transmit packets at significantly higher frequency

compared to sensor nodes farther away. In a multi-hop WSN this implies significant variations in individual nodes' energy consumption rates. Due to the optimality principle of subproblems, the SPF routes from sensors to the base station form a tree rooted at the destination. Over time the region of failing nodes will expand from the base station away. Once this region spreads to the maximum transmission range of a sensor node in all directions around the base station it is cut off from the rest of the network. The network will experience a partition [119] despite the fact that there will be a large number of nodes with significant battery life left. This problem can be formulated as a budget constrained allocation problem: The minimization of σ_E^2 in equation 6.5 is the objective function. Minimization of the objective is subject to the inequality constraint that μ_E be less than some (user defined) maximum budget value $\mu_E \text{ budget}$ in equation 6.6.

$$\min \sigma_E^2 \tag{6.5}$$

subject to constraints,

$$\sum_{n=1}^N \sum_{j=1}^N \frac{L_{i,j}}{N} \leq \mu_E \text{ budget}.$$

$$A_{i,j} \in P_{n,0}^{k_i} \tag{6.6}$$

In [113] Singh, Woo and Raghavendra describe the independent minimization of only the variance of node power levels to extend the lifetime of a network. We propose joint minimization of μ_E and σ_E^2 . Using the constraint formulation highlighted above, we can summarize our optimization framework as identifying the set of routes, one for each source destination pair in a given WSN such that σ_E^2 is minimized while

$\mu_E \leq \mu_E \text{ budget}$. This is analogous to minimizing a distortion measure given a rate budget constraint under an RD framework. Consequently, our proposed dynamic programming algorithm (DPA) is based on an operational RD-framework where we strive to select the mean-variance (MV) operating point that offers the lowest achievable σ_E^2 while maintaining $\mu_E \leq \mu_E \text{ budget}$. The solution takes the form of a set of routes, one from each node to destination (base station). In terms of the optimization problem formulated in equations 6.5 and 6.6, the solution is the set of values $k_1, k_2, k_3, \dots, k_N$ that select a corresponding set of paths $P_{1,0}^{k_1}, P_{2,0}^{k_2}, P_{3,0}^{k_3}, \dots, P_{N,0}^{k_N}$ that minimize the variance of the power consumption under the simultaneous flow of traffic from all nodes.

We have not, to this point, discussed the generation of sets of alternative routes $P_{i,0}^{k_j}$ from every node $1 \leq i \leq N$ to the base station. In the following section we describe four different algorithms to obtain alternative paths between sources and destination.

6.5 Route Discovery

Sub-sections 6.5.1 through 6.5.4 describe the four different route discovery algorithms used for finding alternative paths from nodes to the base station. The four types of routes are bottleneck edge disjoint (BED), bottleneck node disjoint (BND), edge disjoint (ED) and node disjoint (ND). To explain and illustrate the differences between the route discovery algorithms figure 6.1a through 6.1d show the different routes determined from sensor node n_{99} in a WSN consisting of a base station and $N = 99$ randomly placed FFD sensor nodes in a square shaped region of size 10×10 with each nodes transmission range chosen from the Gaussian distribution $\mathcal{N}(\mu_{\Xi}, \sigma_{\Xi}^2)$ (with mean μ_{Ξ} and variance σ_{Ξ}^2). For the following discussion we represent the WSN by graph a $G(V, A)$. Nodes are represented by the set of vertices V and the

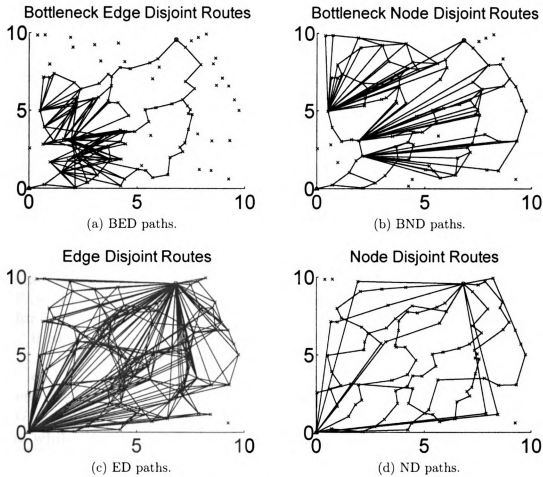


Figure 6.1: Paths from n_{gg} to n_0 .

communication links between elements of V is denoted by the set of edges A . A directed edge from vertex n_i to n_j is denoted by $A_{i,j}$. Furthermore, let Γ denote the minimum spanning tree rooted at the base station on the connected component of G . We also define the maximum power node function $\text{MaxPowNode}(\Gamma)$ which returns the node that experiences the highest power consumption when traffic flows to the base station along Γ . The predecessor function $\text{Pred}(\Gamma, i)$ returns the parent node of n_i in Γ . The indegree(n_i, Γ) function returns the in-degree of node n_i in graph Γ . The shortest path from vertex n_i to vertex n_j on a graph G is denoted by $\text{SP}(i \rightarrow j, G)$.

6.5.1 Bottleneck Edge Disjoint Paths

We define the bottleneck edge to be the outgoing edge from node $\text{MaxPowNode}(\Gamma)$. This method repeatedly determines and removes the bottleneck edge, and rediscovers the minimum spanning tree Γ and from it the shortest paths from each node to the base station. Figure 6.1a depicts all BED paths from n_N to n_0 .

Algorithm 1 Generate Bottleneck Edge Disjoint Paths

Require: $n \geq 0 \vee x \neq 0$

Ensure: $S(i) = \{\}, \forall 1 \leq i \leq N$

$G :=$ Topology of WSN with all available links

$\Gamma := G$

while $\text{indegree}(n_0, \Gamma) > 0$ **do**

$\Gamma :=$ Minimum spanning tree on G rooted at n_0

for all n_i **do**

if $\text{SP}(i \rightarrow 0, \Gamma) \neq S(\text{end})$ **then**

$S(i) := \{S(i), \text{SP}(i \rightarrow 0, \Gamma)\}$

end if

end for

$G := G(V, A - A_{(\text{MaxPowNode}(\Gamma), \text{Pred}(\text{MaxPowNode}(\Gamma), \Gamma)})}$

end while

6.5.2 Bottleneck Node Disjoint Paths

A bottleneck node is defined as the node returned by the $\text{MaxPowNode}(\Gamma)$ function. This method repeatedly determines the bottleneck node under traffic flow along Γ , removes all its incoming edges from the graph, and rediscovers the minimum spanning tree Γ and from it the shortest paths from each node to the base station. Figure 6.1b depicts all BND paths from n_9 to n_0 .

6.5.3 Edge Disjoint Paths

Edge disjoint paths between two vertices n_i and n_j in a graph are paths that do not have any edges in common between them. To determine an ordered set of edge disjoint

Algorithm 2 Generate Bottleneck Node Disjoint Paths

Require: $n \geq 0 \vee x \neq 0$
Ensure: $S(i) = \{\}, \forall 1 \leq i \leq N$
 $G :=$ graph of WSN with all available links
while $\text{indegree}(n_0, \Gamma) > 0$ **do**
 $\Gamma :=$ Minimum spanning tree on G rooted at n_0
 for all n_i **do**
 if $\text{SP}(i \rightarrow 0, \Gamma) \neq S[\text{end}]$ **then**
 $S(i) := \{S(i) \text{ SP}(i \rightarrow 0, \Gamma)\}$
 end if
 end for
 $G := G(V - \text{MaxPowNode}(\Gamma), A)$
end while

paths from a node n_i to the base station this algorithm repeatedly finds and saves $\text{SP}(i \rightarrow 0, G)$ and removes all edges it is composed of from G . The $\text{Edge}()$ function returns the edge set of the path provided in the argument. Figure 6.1c depicts all ED paths from n_N to n_0 .

Algorithm 3 Generate Edge Disjoint Paths

Require: $n \geq 0 \vee x \neq 0$
Ensure: $S(i) = \{\}, \forall 1 \leq i \leq N$
 $G :=$ graph of WSN with all available links
while $\text{indegree}(n_0, \Gamma) > 0$ **do**
 $\Gamma :=$ Minimum spanning tree on G rooted at n_0
 for all n_i **do**
 if $\text{SP}(i \rightarrow 0, \Gamma) \neq S[\text{end}]$ **then**
 $S(i) := \{S(i) \text{ SP}(i \rightarrow 0, \Gamma)\}$
 end if
 end for
 $\text{TempMaxNode} := \text{MaxPowNode}(\Gamma)$
 while $\text{TempMaxNode} \neq n_0$ **do**
 $G := G(V, A - A_{\text{TempMaxNode}, \text{Pred}(\text{TempMaxNode}, \Gamma)})$
 $\text{TempMaxNode} := \text{Pred}(\text{TempMaxNode}, \Gamma)$
 end while
end while

6.5.4 Node Disjoint Paths

Node disjoint paths between two vertices n_i and n_j in a graph are paths that do not have any nodes in common between them. To determine an ordered set of node disjoint paths from a node n_i to the base station this algorithm repeatedly finds and saves $SP(i \rightarrow 0, G)$ and removes all nodes it is composed of from G . The Vertices function returns the vertex set of intermediate nodes between source and destination nodes of the path provided in the argument. Figure 6.1d depicts all ND paths from n_N to n_0 . It is worth mentioning here that the ND paths discovered by this algorithm bear resemblance to the routes determined between source and destination nodes in the multipath routing method described in Baek and de Veciana [6].

Algorithm 4 Generate Node Disjoint Paths

Require: $n \geq 0 \vee x \neq 0$

Ensure: $S(i) = \{\}, \forall 1 \leq i \leq N$

$G :=$ graph of WSN with all available links

while $\text{indegree}(n_0, \Gamma) > 0$ **do**

$\Gamma :=$ Minimum spanning tree on G rooted at n_0

for all n_i **do**

if $SP(i \rightarrow 0, \Gamma) \neq S[\text{end}]$ **then**

$S(i) := \{S(i) \cup SP(i \rightarrow 0, \Gamma)\}$

end if

end for

$TempMaxNode := \text{MaxPowNode}(\Gamma)$

while $TempMaxNode \neq n_0$ **do**

$G := G(V - TempMaxNode, A)$

$TempMaxNode := \text{Pred}(TempMaxNode, \Gamma)$

end while

end while

6.6 Dynamic Programming

The algorithms in the preceding section described four different methods of generating a list of paths in increasing order of path cost from every node to base station. In

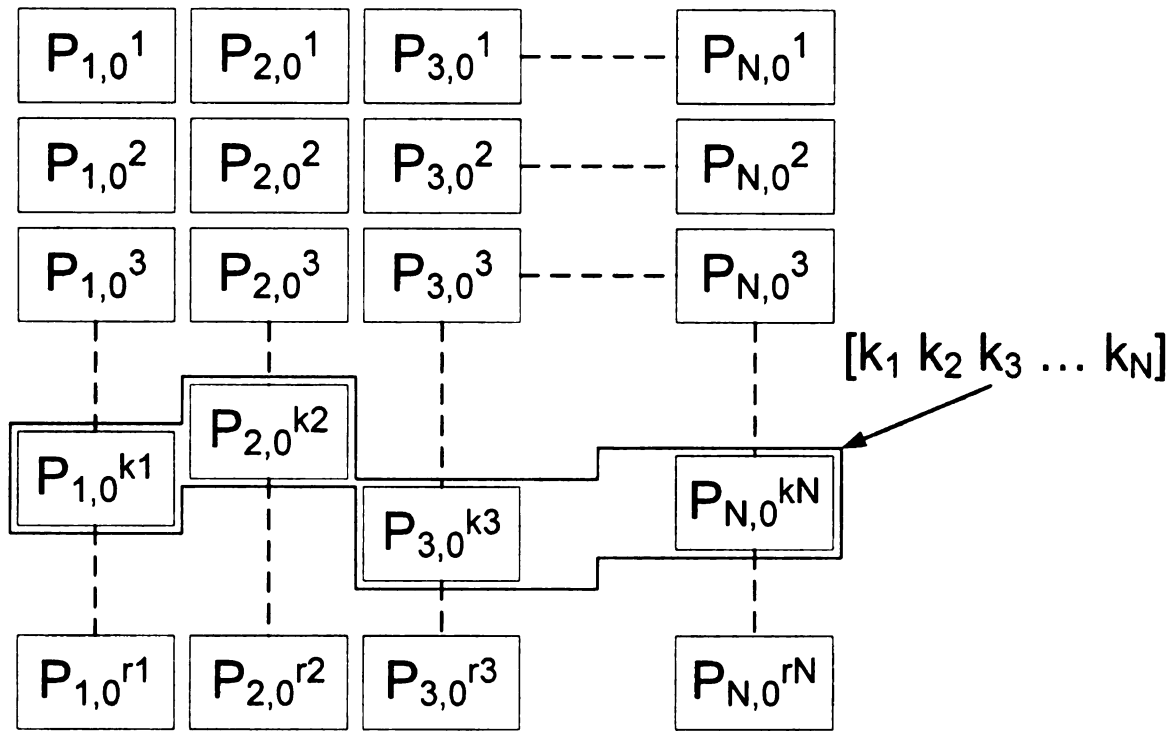


Figure 6.2: N lists of routes sorted in ascending order of path energies.

in this section we describe the DPA that selects one path from each list while achieving the optimization objective. Figure 6.2 illustrates this idea and depicts lists of paths obtained by one of the route discovery algorithms. Each column corresponds to a source node n_i , and every box in it represents one of the r_i paths from source node n_i to destination n_0 . The shaded boxes identified by path indices $[k_1, k_2, k_3, \dots, k_N]$ represents the solution vector to the optimization problem. The DPA described in this section traverses the space of possible solution vectors and attempts to find one that approximates the optimum solution, but at significantly lower complexity.

6.6.1 Theoretical Background

Similar to the operational rate-distortion [32] problem in source coding [94], this joint optimization can be mapped into a Lagrange optimization framework [39]. This formulation is feasible if there is an optimum tuple $[\mu_E, \sigma_E^2]$ such that $\mu_E \leq \mu_E \text{ budget}$

(i.e. an optimal/minimal overall energy can be found such that it exactly equals the budget constraint). In this case, one can view the problem as one that allocates the total energy budget $N \times \mu_E$ budget to all N nodes in the network such that the variance σ_E^2 is minimum. This is referred to as the budget constrained allocation problem [94].

In brief, we can formulate the lifetime maximization problem by considering the *Lagrangian cost* $J(\lambda) = \sigma_E^2 + \lambda\mu_E$, which depends on the *Lagrange multiplier* $\lambda \geq 0$. Identifying the optimum value for the Lagrange multiplier is a crucial aspect of this approach. In particular, the parameter λ represents the slope of the curve in the RD plane. However, the computational complexity of finding the true optimal solution is staggeringly high. The MV plane consists of μ_E on the horizontal axis and σ_E^2 on the vertical axis. The MV region consists of all MV operating points achievable by different combinations of possible paths, one from each node, to the base station. The desired optimal solution will lie on the hull of the MV region. This is very similar to the problem of finding the optimal quantizer in an operational RD sense from Information Theory [32]. We exploit some of the strategies used under RD optimization to develop a DPA for identifying the optimum solution points in the MV space.

6.6.2 Dynamic Programming Algorithm

In order to reduce the computational complexity of finding the hull of the MV region we use a DPA similar to the one used in the determining the hull of the RD region for optimal quantizer design. In the sensor network lifetime maximization problem under consideration it is desirable to minimize σ_E^2 , the variance of node power consumption rates, and μ_E , the mean of node power consumption rates. The reduction in computational complexity over an exhaustive search of all operating points comes

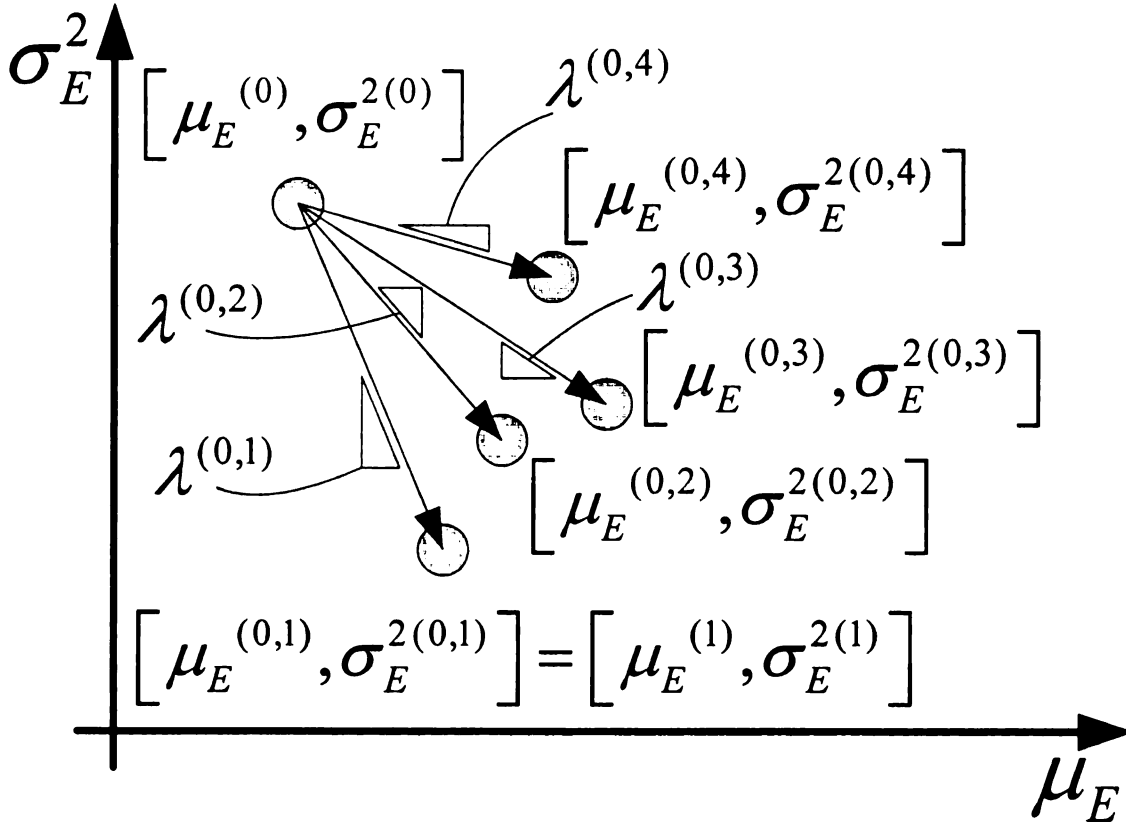


Figure 6.3: Selection of next optimal point in the MV-plane by DPA.

at the cost of DPA being only an operational optimization method approximating the exact optimal solution. The DPA operates on N sorted lists, one for each node, each containing $|P_{i,0}| = r_i$ paths to base station ordered in ascending order of their path costs. The algorithm described here is greedy in nature. It is possible that this algorithm does not find the optimum operational $[\mu_E, \sigma_E^2]$ point under certain scenarios. Nevertheless, the algorithm does provide the optimum operational solution under many practical scenarios. Furthermore, a set of RD variations of this algorithm are rather popular due to their simplicity and low-complexity ([103],[109],[118]).

The starting point $[\mu_E^{(0)}, \sigma_E^{2(0)}]$ of the DPA is the operating point that offers the lowest possible μ_E and is obtained by selecting all $P_{j,0}^1$ where $\forall j, 1 \leq j \leq N$ and is located closest to the σ_E^2 axis. The minimum mean power consumption $\mu_E^{(0)}$ is

associated with a relatively high variance $\sigma_E^2(0)$. The DPA uses $[\mu_E^{(0)}, \sigma_E^2(0)]$ as a starting point to search for the next achievable operating point $[\mu_E^{(1)}, \sigma_E^2(1)]$ closest to the hull of the MV region that adheres to the budget constraint \hat{E}_{budget} . The shortest routes that correspond to $[\mu_E^{(0)}, \sigma_E^2(0)]$ are the ones at the top of all columns in figure 6.2. If pair $[\mu_E^{(0)}, \sigma_E^2(0)]$ satisfies the specified constraint $\mu_E^{(0)} \leq \mu_E^{budget}$ and $[\mu_E^{(1)}, \sigma_E^2(1)]$ does not, $[\mu_E^{(0)}, \sigma_E^2(0)]$ is the optimum point and the algorithm terminates. However, in general, the above condition is not met except in special cases (such as where all nodes are at an equal distance and one hop away from the base station), and the algorithm proceeds to the next step. Rather than considering all possible route combinations, at any operating point $[\mu_E^{(i)}, \sigma_E^2(i)]$ during the $(i+1)$ th iteration, the algorithm introduces the smallest increase ΔE in energy to each route by considering replacement of $P_{j,0}^i$ in the set of routes used for $[\mu_E^{(i)}, \sigma_E^2(i)]$ by $P_{j,0}^{i+1}$ in turn. In a network of N nodes this produces up to N new possible operating points for $[\mu_E^{(i+1)}, \sigma_E^2(i+1)]$. The exact number of alternative operating points depends on the number of route lists that still have an alternate path left for consideration. Each of these possible alternatives is obtained by replacing one of the routes that produced $[\mu_E^{(i)}, \sigma_E^2(i)]$ by the next path and computing a new operating point. This way up to N different alternatives $[\mu_E^{(i+1,j)}, \sigma_E^2(i+1,j)]$ for $1 \leq j \leq N$ are obtained. For each of these N possible operating points we compute their slopes $\lambda^{(i,j)}$ with respect to previous operating point $[\mu_E^{(i)}, \sigma_E^2(i)]$ like in equation 6.7.

$$\lambda^{(i,j)} = \frac{\sigma_E^2(i+1,j) - \sigma_E^2(i)}{\mu_E^{(i+1,j)} - \mu_E^{(i)}} \quad (6.7)$$

Equation 6.8 shows that out of these N alternative operating points $\left[\mu_E^{(i+1,j)}, \right.$

$\sigma_E^2(i+1,j)$] we select the one which offers the lowest gradient $\lambda^{(i,j_{min})}$.

$$j_{min} = \arg \min(\lambda^{(i,j)}) \quad (6.8)$$

Equation 6.9 assigns the point obtained by changing node $n_{j_{min}}$'s path to be the next achievable operating point of the MV plane and advances the the route pointer in column j_{min} of the route list in figure 6.2 down by one position.

$$= [\mu_E^{(i,j_{min})}, \sigma_E^2(i,j_{min})] \quad (6.9)$$

This way the algorithm iteratively determines the next point on the MV region's hull. Figure 6.3 illustrates a simple example. $[\mu_E^{(0)}, \sigma_E^2(0)]$ is the starting point and there are four possible points in the MV plane one of which has to be chosen as the next operating point on the MV curve. In this example $\lambda^{(0,1)} < \lambda^{(0,2)} < \lambda^{(0,3)} < \lambda^{(0,4)}$. Therefore, the point $[\mu_E^{(0,1)}, \sigma_E^2(0,1)]$ corresponding to the smallest slope $\lambda^{(0,1)}$ will be chosen for $[\mu_E^{(1)}, \sigma_E^2(1)]$. In this manner a curve is obtained in the MV plane which estimates the boundary of the MV region. Figure 6.4 depicts MV curves for a WSN using different route discovery algorithms at our disposal. The differently sloped lines (λ' , λ'' and λ''') represent different benefit/cost (variance reduction/mean increase) ratios, that can be used to identify a suitable operating point on the hull of an MV curve. The curves shown in this figure are truncated at the right end when the mean value becomes greater than the mean at the initial starting point. If the DPA is allowed to run to completion the number of points returned by it is approximately 50 to 100 times the ones plotted here. However, the region of the curve that is of interest to us is the one between the starting point and the point offering minimum

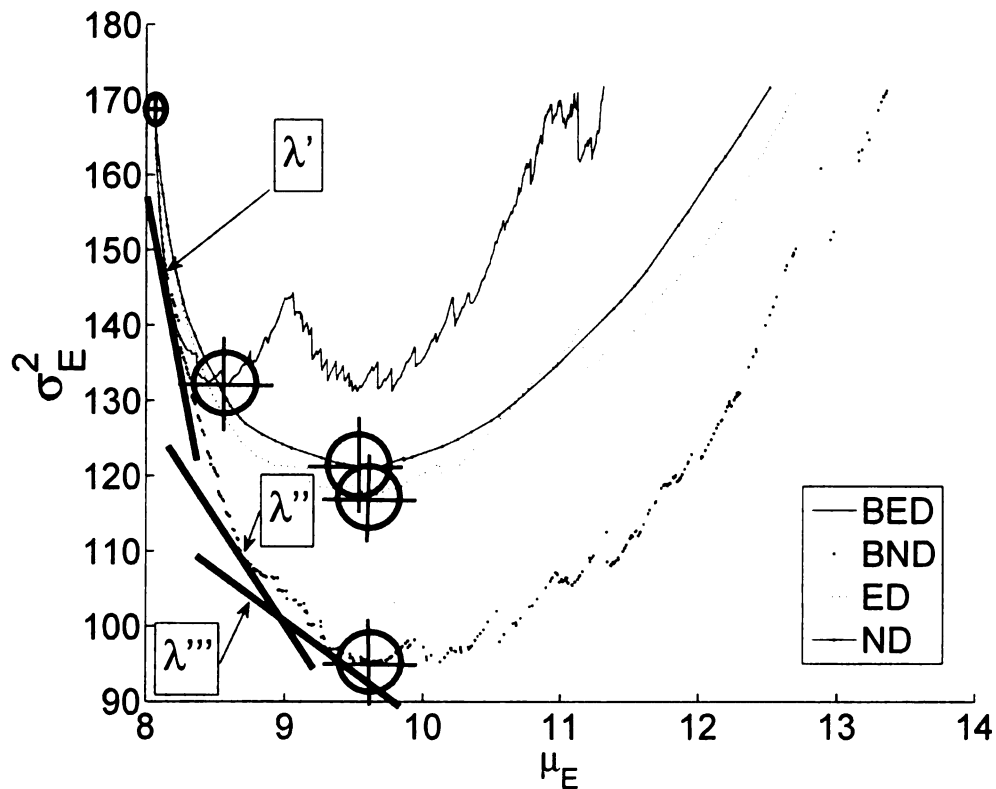


Figure 6.4: Mean-Variance tradeoffs offered by BED, BND, ED and ND paths.

variance.

6.6.3 Computational Complexity of Finding Optimal Solution

There are two sources of complexity in this problem; 1) The computation of achievable operating points, 2) the traversal of the search space for the best achievable operating point. The graph $G(V, A)$ that represents the WSN consists of a vertex set V of cardinality $N + 1$ and a directed edge set A of maximum cardinality $N^2 + N$ (we consider a worst case scenario in which the network topology forms a fully connected graph). First, we derive an upper bound for the order of complexity of the size of the

search space for a brute force method. The total number of paths from a node n_i to base station n_0 with q intermediate hops is $(N-1)P_q$. Then the total number of possible paths from n_i to n_0 is the number of paths with $0, 1, 2, \dots, N-1$ intermediate nodes, i.e. $\sum_0^{N-1} (N-1)P_q = \sum_0^{N-1} \frac{(N-1)!}{(N-q-1)!} = O(N!)$. Then the search space of solution vectors for routes from all N nodes to destination consists of $O(N!^N)$ points. A linear search of this space leaves the complexity unaffected, i.e. $O(N!^N)$.

Now we compute the complexity of the DP formulation proposed in this chapter. The size of the search space for in DPA is $\prod_{q=1}^N r_q$. The expression of r_i in terms of N depends on the route discovery algorithm used. For BED the upper bound on the number of paths r_i generated for a node n_i is $(N+1)N = O(N^2)$ bringing the total size of the search space to $N \cdot (N+1)N = O(N^3)$. The bound for BED can be used as an upper bound on the number of paths per node generated by ED.

For BND the upper bound on the number of paths r_i generated for a node n_i is $N = O(N)$ bringing the total size of the search space to $N \cdot N = O(N^2)$. The bound for BND can be used as an upper bound on the number of paths per node generated by ND. Since the removal of each entry from the search space requires the computation of N slopes, the complexity of the DPA for BED and ED is upper bound by $O(N \cdot N^3) = O(N^4)$ and for BND and ND it is upper bound by $N \cdot N^2 = O(N^3)$. Regardless of whether BED/ED or BND/ND are used for route discovery, their respective complexity terms $O(N^4)$ and $O(N^3)$, respectively, are still orders of magnitude lower than $O(N!^N)$, the order of the exhaustive brute force search method.

6.7 Performance Analysis

For the comparative evaluation of BED+DPA, BND+DPA, ED+DPA and ND+DPA algorithms we generate 100 randomly generated networks deployed in a square shaped

region of size 10×10 . In all networks the base station is located at coordinates $(0, 0)$ to obtain large variations in path lengths. $N = 99$ nodes are randomly and uniformly scattered in the plane. Transmission ranges Ξ of sensors are Gaussian distributed according to $\mathcal{N}(10, 2)$. The WSN is monitoring a process with an entropy uniform across the space spanned by the network. This implies that the traffic generation rates of all sensors are also equal, i.e. $T_1 = T_2 = T_3 = \dots = T_{99} = T$. Sensors are equipped with omni-directional antennas with decay factor $\alpha = 3$. Initial battery reserves B_i for all sensors are assumed to be 1.

6.7.1 Mean-Variance Trade-off

Figure 6.4 illustrates the variations in the estimate of the MV region boundary produced by the DPA when different route discovery algorithms are used for the same network. In this particular example the BND paths seem to offer the greatest decrease in σ_E^2 , ND the least and BND and ED fall in between the two extremes. However, similar plots of other networks may reveal that this ordering does not hold true for all cases.

We will attempt to determine if there is a route discovery algorithm which provides consistent superior performance over the other algorithms. The four scatter plots in figure 6.5a through figure 6.5d correspond to the four route discovery algorithms. Each plot is obtained by applying the DPA to the list of routes produced by one of the four route discovery algorithm. Each point in a plot corresponds to one of the 100 randomly generated networks. Its position on the horizontal axis indicates the percent increase in μ_E (denoted $\Delta\mu_E$) and its position on the vertical axis the percent decrease in σ_E^2 (denoted $\Delta\sigma_E^2$) offered by the minimum variance point (marked by red cross hair) on the MV curve with respect to its starting point (marked by black cross hair). The decrease in variance of power consumption rates $\Delta\sigma_E^2$ are savings S

that are achievable at a cost C that is the increase in mean power consumption rate $\Delta\mu_E$. In the remainder of this section we will use the terms savings and cost to refer to the two quantities.

As such, it is desirable that most data points be concentrated in the upper left corner of the scatter plots, indicating high savings S achievable at low cost C . Points located close to the origin indicate little change from the starting point. Data points located in the lower right corner indicate bad trade-offs, i.e. low savings in return for high cost. The great overlap in the regions covered by the points indicates that there is no clear winner among the route discovery algorithms.

To gain further insight we project data points in scatter plots in figures 6.5a through figure 6.5d on the horizontal/ C axis, and the vertical/ S axis. This process is similar to finding the marginal distributions of a joint distribution of two random variables. The four histograms in figure 6.6 are obtained by projection of data points in figure 6.5a through 6.5d on the horizontal/ C axis. Similarly, the histograms in figure 6.7 are obtained by projecting onto the vertical/ S axis. In most cases the shapes of the histograms in figure 6.6 and figure 6.7 all appear to indicate an underlying gaussian distribution. We estimate the values of the mean and variance parameters of gaussian distributions fitting the histogram using. We compute the maximum likelihood estimates (MLE) [122] of the parameters μ_C , σ_C^2 , μ_S and σ_S^2 based on 100 data points for each histogram. The MLEs for BED, BND ED and ND are given in table 6.2 and the resulting Gaussian distributions properly scaled for comparison are plotted together with the histograms. Since we were previously unable to clearly identify one route discovery method as being superior to others in terms of performance we now evaluate their performances in probabilistic terms. A clear winner will have a distribution for C with low mean and low variance and a distribution for S with a high mean and low variance. Based on MLE parameters in table 6.2 ND emerges as the least attractive option because it offers the smallest average savings $\mu_S = 24.37\%$

Table 6.2: MLE parameters of marginal histograms generated from applying DPA with BED, BND, ED and ND route discovery algorithms to 100 randomly generated network topologies.

	μ_C	σ_C^2	μ_S	σ_S^2
<i>BED</i>	15.25	328.76	28.09	353.20
<i>BND</i>	20.94	289.93	36.22	303.03
<i>ED</i>	32.25	504.43	35.37	307.67
<i>ND</i>	29.06	390.07	24.37	250.96

at nearly the highest average cost $\mu_C = 29.06\%$. We eliminate it from further consideration. Although with 35.37% ED offers on average one of the largest savings, it does so at a cost that is even higher than ND, i.e. an average savings of 35.37% in return for an average cost increase of 32.25%. In comparison, the BND+DPA option offers an average savings of 36.22% in return for a 20.94% increase in cost, which is significantly lower than from costs of ED and ND. Finally, BED offers average savings of 28.09% at an additional cost of 15.25%. In comparison with BND this means BED offers lower savings for a lower cost. However, the variances of cost and savings distributions of BED in table 6.2 show that there is considerably greater variation and uncertainty about BEDs costs and savings, relative to BND. The significant differences in variance terms σ_C^2 and σ_S^2 can be attributed to the fact that the Gaussian seems to be an ill-fit for BED's histograms in figures 6.6 and 6.7 and approximates an exponential more closely than a Gaussian. While multi-modal distributions could offer more accurate fits for the data they will complicate performance comparison.

We explain the poor performance of paths generated by ED and ND route discovery algorithms with respect to BED and BND by the fact that they exclude not just one, but all elements (edges or nodes) of a discovered path from consideration in subsequent path searches. While this cuts down the size of the search space it also removes too many alternative paths from consideration.

Furthermore, the high concentration of values in the bins centered close to 0 in ED's histogram in figure 6.7 indicates a high rate of failure of the DPA, i.e. instances where the DPA is not able to offer a significant trade-off. BED has the highest number of failures followed closely by ED. A closer look at the scatter plots of BED and BND in figures 6.5a and 6.5b respectively shows that BED offers better performance on tight budget constraints on μ_E budget in the low range of savings S whereas BND performs better when the constraint on μ_E budget is more relaxed and higher higher savings S are required. The higher rate failure rate of BED makes its use a more risky option. Also, it should be noted that the scatter plots only plot trade-offs offered by minimum variance point, i.e. when $\lambda = 0$. As the curve plots in figure 6.4 there is a large set of points that offer intermediate trade-offs for when lambda is in the range $-\infty < \lambda < 0$. This is illustrated in figure 6.4 for three different values of λ set to λ' , λ'' and λ'''

6.7.2 Spatial Redistribution of Energy

That leaves open the question of how effectively this method evens out the spatial distribution of energy consumption rates observed under SPF routing. The diffusion plot in figure 6.8 depicts the spatial distribution of energy consumption under SPF routing. This plot is obtained by averaging it over the same set of 100 networks used in the previous section. As expected, as one moves from the upper right corner, towards the lower left corner where the base station is located there is a gradual increase in the power consumption rate in nodes. The lower left corner, the region in immediate vicinity of the base station occupied by critical nodes, forms a traffic hotspot. To illustrate the differences in the spatial distribution of energy consumption rates produced by the BED, BND, ED and ND route discovery algorithms and DPA we first subtract the diffusion plot of the SPF routes in figure 6.8. The results are the

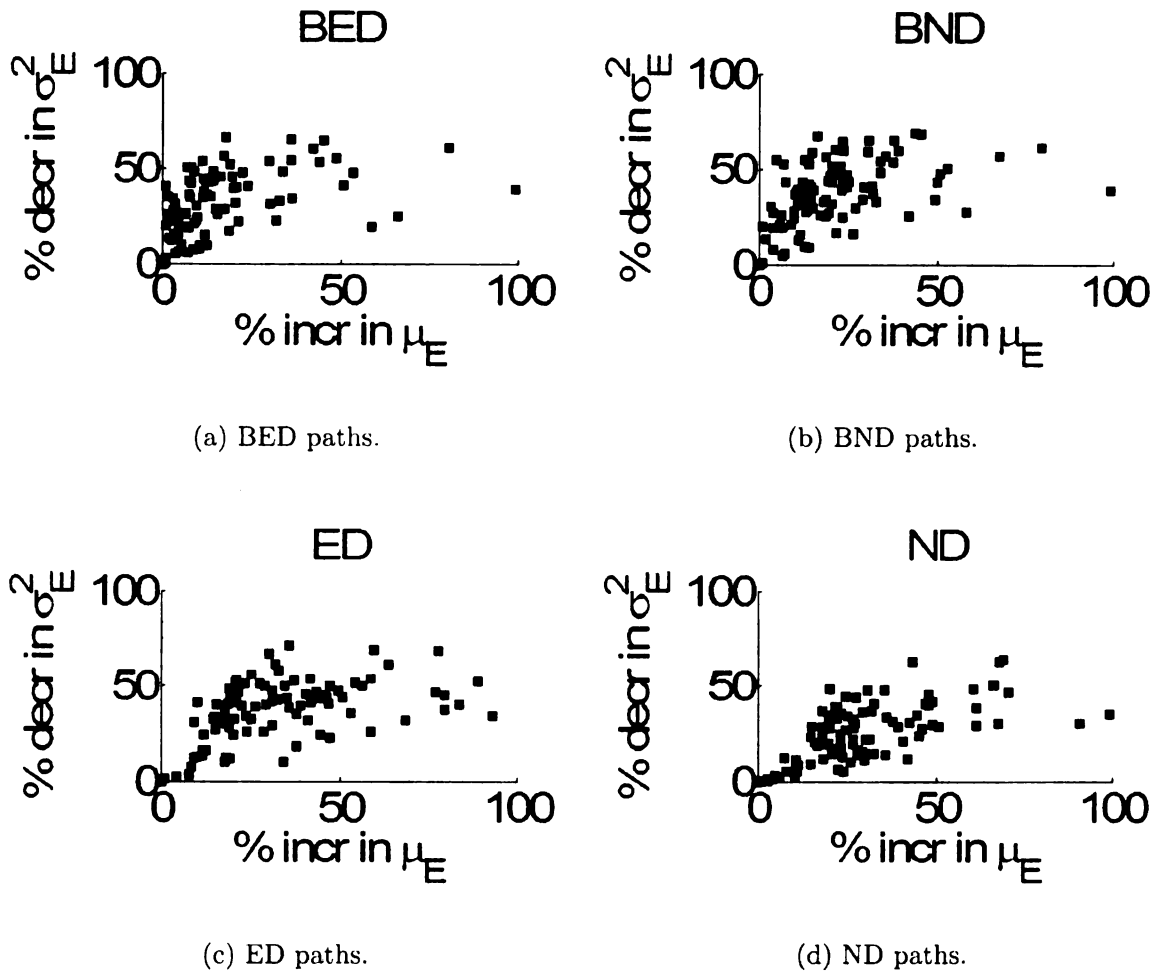


Figure 6.5: Plots of percent decrease in variance against percent increase in mean.

differential spatial diffusion plots in figures 6.9 through 6.12. Like the plot in figure 6.8, these too are averaged over all 100 networks. All route discovery algorithms produce a reduction in power consumption rates in the region occupied by a peak in the SPF's diffusion map in figure 6.8. This is shown by the blue negative region in the same place, representing a reduction in node power consumption. We also observe an increase in power consumption rates marked by a yellow-red-yellow band that envelops the blue region on its upper-right side. This signifies a partial shift of relay traffic from critical nodes to their immediate neighbors that are farther away from the base station. A closer look at figure 6.9 and 6.10 shows that the redistribution of traffic is

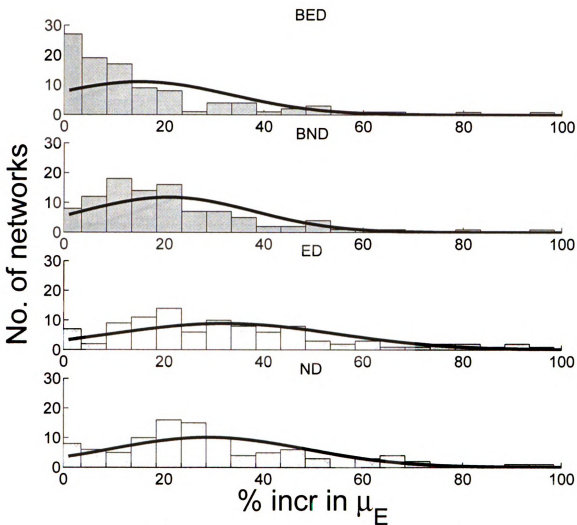


Figure 6.6: Marginal histograms of percent increase $\mu_{\Delta\mu_E}$ for the scatter plots in figures 6.5a through 6.5d.

similar under BED and BND, although BND spreads the energy out over a slightly larger region and that the reduction in power consumption in BND+DPA is around 8000 units, while BED+DPA produces a maximum reduction of only 7000 units. From our previous analysis we would expect the changes in the traffic distribution between SPF and ED to be less stark which is confirmed by figure 6.11. Finally, ND seems to be the worst performer because it produces the least change figure 6.12.

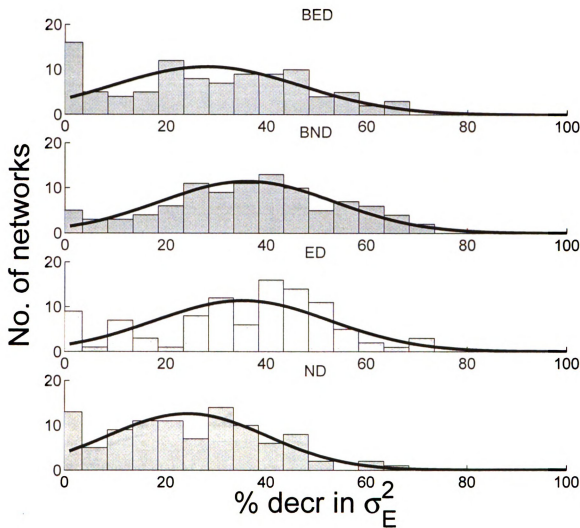


Figure 6.7: Marginal histograms of percent decrease $\mu_{\Delta\sigma_E^2}$ for the scatter plots in figures 6.5a through 6.5d.

6.8 Conclusions

In this chapter we presented an statistical interpretation of network lifetime that takes into account a limited degree of node redundancy in WSNs. The interpretation of network lifetime in terms of the mean and variance of network-wide node power consumption rates provides us an optimization objective that does not narrowly focus the on the lifetime of a single node as is the case of most prior work. We provide a dynamic program formulation of the problem that seeks to optimize variance of power

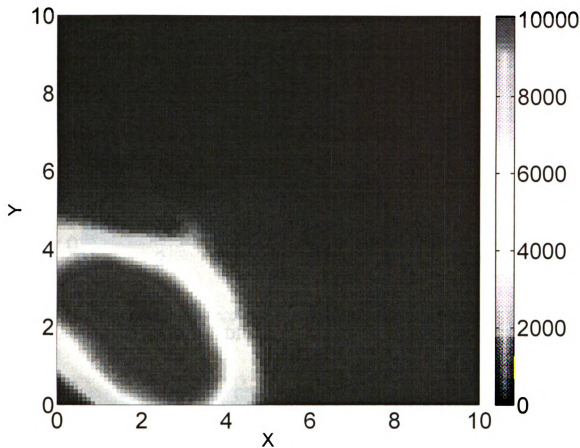


Figure 6.8: Diffusion plot of energy consumption rates averaged over all 100 networks under SPF routing.

consumption rates while constraining the average consumption rate. We develop the DPA that chooses routes in a way that optimizes for our objective from sets of paths that would be considered sub-optimal in the shortest path sense. Four variants are developed based on the BED, BND, ED and ND route discovery algorithms. We also observe that the routes discovered by the ND algorithm are very similar to those proposed in previously proposed load balancing techniques such as Baek and de Veciana's in [6]. Interestingly, under our understanding of network lifetime and a many-to-one traffic flow ND is the worst performing of all route discovery algorithms. A statistical performance comparison of these four techniques for the case when $\lambda = 0$ shows that on average the BND and BED in conjunction with the DPA yield the best

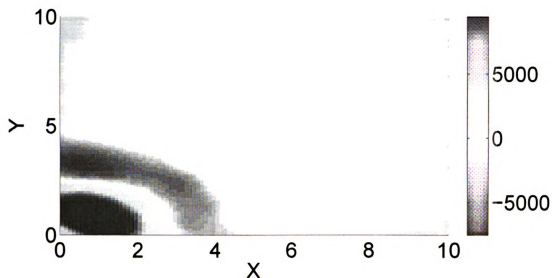


Figure 6.9: Differential diffusion plots of energy consumption rates averaged over 100 networks using BED paths and DPA.

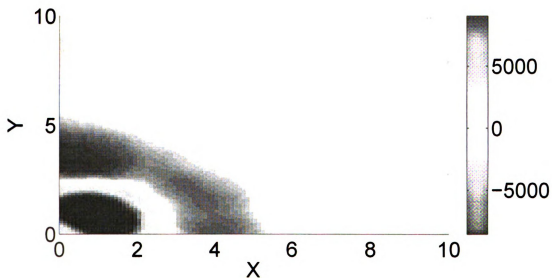


Figure 6.10: Differential diffusion plots of energy consumption rates averaged over 100 networks using BND paths and DPA.

performance. BND and BED yield reductions of up to 28% and 36% in variance of power consumption rates at the cost of raising average node power consumption by 15% and 21%, respectively. The computational complexity of variants of the DPA vary from $O(N^3)$ to $O(N^4)$ which is significantly lower than the full search of

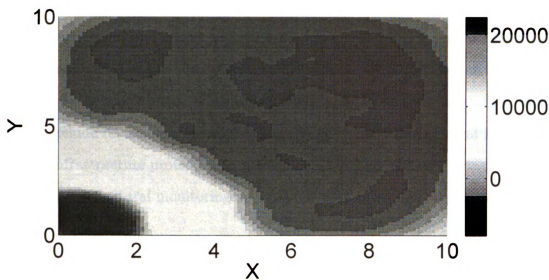


Figure 6.11: Differential diffusion plots of energy consumption rates averaged over 100 networks using ED paths and DPA.

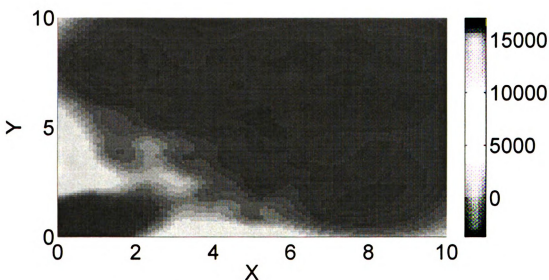


Figure 6.12: Differential diffusion plots of energy consumption rates averaged over 100 networks using ND paths and DPA.

the solution space which is of complexity $O(N!^N)$. Analysis by means of diffusion plots verifies that DPA indeed reduces power consumption of sensors that experience highest power consumption under shortest path routing algorithms. Diffusion plots also show that the reduction power consumption is highest under BND, followed

closely by BED.

The resulting route selection method is one that is suitable for applications with many-to-one traffic flows. Route discovery algorithms and DPA assume availability of global network connectivity which is very likely going to be the base station. While this may exclude its use in some applications we envision it having great application in critical infrastructure protection/control/monitoring, surveillance network and environmental/agricultural monitoring applications with infrequent topology changes.

Chapter 7

Mean-Field Solution of Small-World Wireless Sensor Network Models With Range Limited Shortcuts

7.1 Introduction

The limited range of wireless channels naturally imposes geometric or Euclidean graph topologies [98] on wireless networks [117]. Wireless sensor networks (WSN) [2] are a class of mobile or stationary, multi-hop ad-hoc wireless networks of power constrained nodes. WSNs are employed by sensing, detection and data gathering applications which impose a many-to-one data flow. On the spectrum of randomness of graphs the endpoints are occupied by lattice graphs on one end (no randomness) and random graphs on the other (complete randomness). In between these extremes lie small-world networks [128]. Lattice graphs are characterized by strongly connected neighborhoods that imply a high degree of connectivity and resilience, but rather large diameters. Random graphs [13], on the other hand, are characterized by small diameters (that imply an ability to retrieve and disseminate information quickly) but low connectivity between neighborhood nodes. Note that the same topological properties that enable the fast dissemination of information within a network also enable fast information retrieval. Clearly, in the context of WSNs both strong connectivity at the local level and small diameters are desirable, both properties of small-world networks. In [128] and [127] Watts provided analytical models for 1-dimensional lattice graphs, connected caveman graphs and Moore graphs. Several attempts have been made to leverage the small-world network effect in WSNs ([111],[27],[125],[52]). However, to the authors' best knowledge there are no analytical models of 2-dimensional Euclidean graphs with range limited shortcuts that are applicable to WSNs. This research derives analytical models for the clustering coefficient and characteristic path length of WSNs whose topologies are augmented by a small number of shortcut links whose range is limited by practical limitations. Results show that in spite of the fact that shortcuts are range limited and significant differences in construction method the phase difference between drop in clustering coefficient and characteristic path length

that is the hallmark of small-world networks still appears. Although shortcuts are range-limited only about 0.005 – 0.05% of nodes had to be equipped with the ability to communicate over long ranges.

The remaining chapter is organized as follows. Section 7.2 provides a brief background of small-world networks, their properties and their standing relative to Euclidean and random graphs. Section 7.3 describes different system models for WSNs with small-world topologies and provides a literature review of prior methods used to this end. Section 7.4 uses mean field analysis to derive generalized expressions of clustering coefficient and characteristic path length that can be used to model small WSNs based on any of the system models in section 7.3. Section 7.5 evaluates the model for different ranges of parameters. Section 8.8 concludes the chapter.

7.2 Background: Small-World Networks

Small world networks were first discovered by Milgram in social networks in [82]. Watts in [128] and [127] analyzed social networks which included a collaboration network of actors and a collaboration network of mathematicians, to verify Milgram's idea of six degrees of separation. More recently Horvitz and Leskovec verified the presence of six degrees of separation in social networks using a much larger data set from Microsoft's Windows Live Messenger instant messaging traffic [130]. Since Milgram's original experiment several other works have analyzed networks in natural and man-made systems to discover that their topologies are in fact small world graphs. Latora and Marchiori [73] and Watts [127] analyzed the neural network of the *C.elegans* worm. Montoya and Sole [87] studied food webs in nature. Moore and Newman [88],[89] studied the transmission of infectious diseases in populations with small world connectivity. Studies of transportation networks include Latora, Marchiori [73],[74] on the Boston subway network and Sen et al. [110] on the Indian

railway network. Small world analysis of communication networks include Adamic's [1] on the World Wide Web. i Cancho and Sole [50] studied the lexical networks of human languages and determined them to be small worlds. More relevant to our work are studies on creating small world topologies in wireless networks by various means. Wan et al. [125], Cavalcanti et al. [23] and Costa and Barros [31] showed that selectively equipping a small fraction of nodes in a WSN with two radios (one regular, short range IEEE 802.15.4 and one long range IEEE 802.11b) induces a small world topology. Hubaux et al. [49] and Dimitar et al. [36] propose a small world application layer for ad hoc networks similar to logical links in peer-to-peer networks. Dixit, Yanmaz and Tonguz [37] analyze the topology of cellular wireless networks. Helmy [46],[47] proposes mobility assisted wireless networks to create shortcuts that mimic random links. Sharma, Mazumdar [111] make selected use of wired links to create shortcuts.

Among prior models of small world networks are Watts [128] models for the clustering coefficient and characteristic path length of 1-dimensional (circular) lattice graph, connected caveman graphs and Moore graphs. In [91] Newman and Watts found a mean-field solution of the small world network model for 1-dimensional lattices. Amaral [3] studied the statistical properties of classes of small world networks using a taxonomy based on the form of degree distribution of nodes. Some recent advancements in WSNs have enabled the addition of a number of links with longer communication range, called *shortcuts* or *global scale links*, to a network by various means (at a cost). In this chapter we derive analytical expressions for two defining properties of small-world networks, the clustering coefficient C and characteristic path length L of geometric graphs in a 2-dimensional plane with a number of range limited shortcut links. Let $G(V, E)$ denote a graph consisting of a set of N vertices $V = \{v_1, v_2, v_3, \dots, v_N\}$ and a set E of M edges consisting of all edges, where $e_{i,j}$

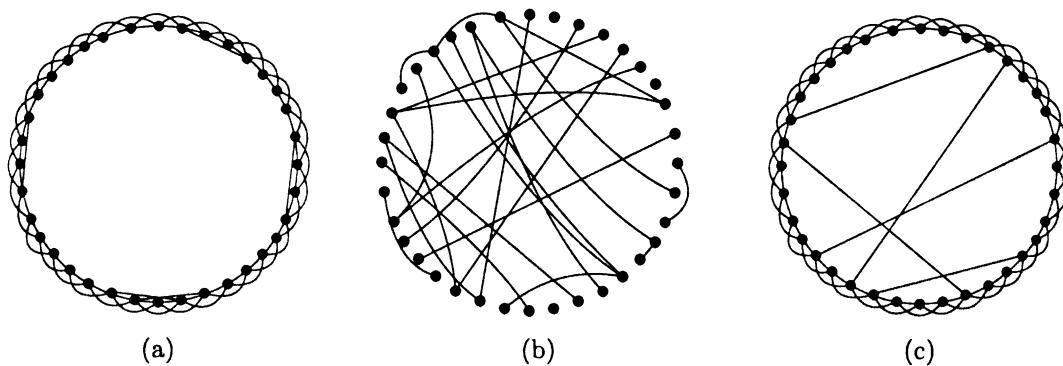


Figure 7.1: Illustrated examples for three different classes of graphs; a) Geometric graph, b) Random graph, and c) Small world graph.

denotes an undirected edge from v_i to v_j .

7.2.1 Characteristic Path Length

Characteristic path length is defined as the average length of geodesic paths between all pairs of nodes. For a graph G , the characteristic path length $L(G)$ is defined as the number of edges in the geodesic path between two vertices, averaged over all pairs of vertices. If $L_{i,j}$ is the number of edges on the geodesic path from v_i to v_j , then,

$$L = \frac{\sum_{i=1}^N \sum_{j=1}^N L_{i,j}}{N(N-1)} \quad (7.1)$$

7.2.2 Clustering Coefficient

The Clustering Coefficient is a measure of the cliquishness, the degree to which vertices in a graph coalesce into tight groups. For a graph G , the clustering coefficient $C(G)$ is defined as follows. Suppose a vertex v has k_v number of neighbors, then the maximum possible size of the set of undirected edges between v and all its neighbors is $\frac{k_v(k_v+1)}{2}$. Then C_v is the fraction of these edges that actually exist in E . The

clustering coefficient of the graph is then defined as the average C_v over all vertices V , as in equation 7.2.

$$C = \frac{\sum_{i=1}^N C_i}{N} \quad (7.2)$$

The above definitions of characteristic path length and clustering coefficient are easily extended to directed and weighted graphs.

7.2.3 Small World, Geometric and Random Graphs

We will be using the terms graph and network and the terms node and vertex interchangeably. Geometric graphs have (see figure 7.1a), both large L and C by virtue of their strong local connectivity. Random graphs (see figure 7.1b) are the other extreme and are characterized by both small L and C . Small world networks (see figure 7.1c) have small L but large C .

There are several construction methods for small world networks. Among the simplest to understand is the β -model described by Watts in [128]. A small world network can be constructed from a lattice or geometric graph by rewiring one end of every edge to a randomly selected node with probability β . The type of graph that is constructed by the β -model depends on the value of rewiring probability β . If $\beta = 0$ the graph remains a lattice/ geometric graph, while $\beta = 1$ yields a random graph. Watts and Strogatz [129], Newman and Watts [92], Pandit and Amritkar [96] and Watts [128] repeatedly demonstrated $\beta = 0.01 - 0.05 \approx \mathcal{O}\left(\frac{1}{N}\right)$ as a typical range of values for constructing small world networks, which means only a very small fraction of links has to be rewired. Hence, on the scale of randomness where lattice/ geometric graphs occupy one extreme and random graphs the other, small world graphs fall in between the two, but closer to the former than the latter.

In the context of WSNs the property of random graphs to propagate information quickly means that nodes can relay important data back to the base station quickly. The property of Euclidean graphs to have high C and strong local connectivity produces a stable network structure difficult to partition. It also enables the formation of clusters which facilitates collaboration among groups of nodes. Thus, the WSN application demands the best of Euclidean and random graphs.

7.3 Small-World Topology Construction Methods for Wireless Networks

There are several approaches to realize shortcuts in wireless networks with the intent of creating a small world topology. We conducted a survey of practical techniques.

7.3.1 Hybrid Sensor Network

Sharma, Mazumdar [111] and Chitradurga [27] proposed the judicious use of wired links between a small subset of nodes in a WSN. The resultant network is not a pure wireless network and is called a hybrid sensor network. The graph topology of hybrid sensor networks uses range limited links for shortcuts instead of truly random links. From a practical perspective the wired shortcuts limit the possible deployment scenarios.

7.3.2 Multi-radio Network

The multi-radio node as a means to building small worlds was first proposed by Wan in [125] with the objective of easing congestion in the primary network consisting of ubiquitous low-rate links. The multi-radio node architecture has the disadvantage that it requires a heterogeneous mix of nodes. The links of the long range radio

interface serve as small world shortcut links. The implementation used in [125] uses IEEE 802.15.4 [5] in conjunction with IEEE 802.11b [51] network interface. The significantly higher power consumption of IEEE 802.11b is a disadvantage in power constrained wireless sensor networks.

7.3.3 Receiver Side Cooperation

The third solution is based on single-input multiple-output (SIMO) principles and is dubbed by Ilyas and Radha as the 'Poor-Man's-SIMO-System' (PMSS) because it uses receiver side diversity combination techniques in a network of commercial-off-the-shelf components. Its principal strength over hybrid sensor networks and multi-radio networks is that it circumvents the need for customized or reconfigured hardware. Instead, it relies on receiver side cooperation in motes to form a SIMO system. This way cooperation reduces losses and retransmissions while increasing throughput and channel utilization. In [52] Ilyas and Radha describe the PMSS and how three well-known diversity combination techniques are adopted for use in a network of commercial IEEE 802.15.4 devices. The diversity combination techniques are derivatives of diversity combining methods for analog signals presented by Brennan in [19]. Simply put, the purposes of diversity combining are twofold; a) Select an error-free version of a received transmission from among multiple received versions or, b) If the first goal is not achievable, obtain another version of the transmission, with no errors or fewer errors than any of the individual received versions.

In conclusion of this survey it should be understood that as diverse as these different implementations of small-world networks in wireless networks may be, the analytical model can be parameterized accordingly.

7.4 Mean Field Analysis of Small-World Wireless Networks

7.4.1 Clustering Coefficient

Sensor nodes are distributed as a 2-dimensional Poisson point process with mean ρ points (sensor nodes) per unit area. Let there be two nodes v_1 and v_2 capable of communicating with nodes within range R_1 and R_2 , respectively. For simplicity's sake we will take $R_1 = R_2 = R$. If the two nodes are separated by distance $r < R$ then communication coverage regions of nodes v_1 and v_2 will overlap, as depicted by the shaded region in figure 7.2. Let Δ denote the size of the overlapping communication coverage regions. Then Δ is computed by adding the areas of sectors XOX' and $XO'X'$ and subtracting the areas of triangles XOX' and $XO'X'$ because they have been added twice during the addition of both sectors. The resulting expression for Δ is shown in equation 7.3.

$$\begin{aligned}\Delta(\theta, R) &= 2 \cdot \frac{2\theta}{2\pi} \pi R^2 - 2 \cdot R \sin \theta \cdot R \cos \theta \\ &= 2\theta R^2 - R^2 \sin 2\theta\end{aligned}\tag{7.3}$$

Here θ is the central angle $\angle XO'X'$ in figure 7.2. We can express θ as a function of r .

$$\theta(r, R) = \arccos \frac{r/2}{R}\tag{7.4}$$

Substituting the expression for θ in equation 7.4 back into equation 7.3 give us Δ as a function of r .

$$\begin{aligned}\Delta(r, R) &= 2R^2\theta(r, R) - R^2\sqrt{1 - \frac{r^2}{4R^2}} \cdot \frac{r}{2R} \\ &= 2R^2 \arccos\left(\frac{r}{2R}\right) - \frac{rR}{2}\sqrt{1 - \frac{r^2}{4R^2}}\end{aligned}\quad (7.5)$$

As we already described, for the network topology to be a small world some nodes have to provide *shortcuts* or *global scale links* to other nodes. These nodes will be referred to as *shortcut nodes* or *global scale nodes*. In literature, the graph formed by global scale nodes is also referred to as a network's substrate [128]. The remaining nodes are called *local scale nodes*. The ratio of the maximum communication range of global scale link to that of a local scale link is called the scaling factor ξ . Now, let ρ denote the average node density with which nodes occupy the covered region. The cooperative communication model for small world WSNs in section 7.3.3 for the implementation of shortcuts in a wireless sensor network requires multiple nodes to function as a single receiver. Depending on how many nodes are collapsed into a single node will reduce the effective node density ρ . Let n_c denote the average number of nodes that combine to function as a global scale node. This reduces the node density in the equivalent small world graph (spread over area A) to ρ' which relates to ρ according to equation 7.6.

$$\rho' = \rho - \frac{N_{global}(n_c - 1)}{A}\quad (7.6)$$

Note that when shortcut links are implemented in multi-radio networks (subsection 7.3.1) and hybrid sensor networks (subsection 7.3.2) the node density in the small world graph remains unchanged, i.e. $n_c = 1$ which implies $\rho' = \rho$. Then $\Delta(r, R)\rho'$

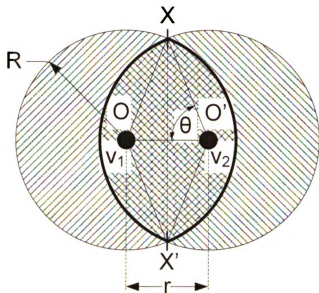


Figure 7.2: Overlapping communication regions of two communicating sensor nodes in a WSN.

denotes the number of edges a node (v_1) distance r away from a reference node (v_2) has with the set of nodes that can also communicate with the reference node. Let $\Gamma(v)$ denote the neighborhood of a vertex $v \in V$, where V denotes the set of all vertices in the graph G of the wireless sensor network's topology. We treat the entire region covered by sensors as a continuous, homogeneous sea of sensors populated with a uniform density of ρ' nodes per unit area. Then the total number of edges d in the neighborhood of vertex v can be computed by integrating $\Delta(r, R)\rho'$ with respect to r over the interval $[0, R]$, the radius of the communication range; The number of links in the neighborhood $\Gamma(v)$ of a node v is denoted by $d(\Gamma(v))$. The exact derivation of the expression for $d(\Gamma(v))$ in equation 7.7 is provided as lemma 1 in this chapter's appendix.

$$d(\Gamma(v)) = \pi \rho'^2 R^3 (2.9604R - \sqrt{3}) \quad (7.7)$$

PDF

links

links

links

links

links

links

links

links

links

Let $D(\Gamma(v))$ denote the number of edges in the completely connected graph of node v and its neighborhood $\Gamma(v)$.

$$D(\Gamma(v)) = \pi \rho' R^2 \times \pi \rho' R^2 = \pi^2 \rho'^2 R^4 \quad (7.8)$$

The clustering coefficient is defined as the expected value of the ratio of the number of links in a nodes neighborhood to the number of links in a complete graph of its neighborhood. This can also be defined in terms of previously computed values as;

$$C(G) = E_V \left[\frac{d(\Gamma(v))}{D(\Gamma(v))} \right] \quad (7.9)$$

For a node that is part of the sea of sensors communicating over local scale links only we define the clustering coefficient C_{local} as,

$$\begin{aligned} C_{local} &= \frac{d(\Gamma(v))}{D(\Gamma(v))} = \frac{2.9604R - \sqrt{3}}{\pi R} \\ &\approx \frac{2.9604R - \sqrt{3}}{\pi R} \end{aligned} \quad (7.10)$$

Hence, C_{local} grows $\mathcal{O}\left(\frac{1}{R}\right)$ with respect to R . For a node that is part of the substrate communicating over local as well as global scale links we define the clustering coefficient C_{global} as,

$$C_{global} = \frac{d(\Gamma(v)) + k_{global}}{D(\Gamma(v)) + k_{global}(\rho' \pi R^2) + (k_{global} - 1)!} \quad (7.11)$$

Year 1

Year 2

Year 3

Year 4

Year 5

Year 6

Year 7

7.4.

The

part

of

the

part

of

the

part

of

the

part

of

the

part

Note that in doing so we assumed that $\xi \geq 2$. Here k_{global} is the average degree of nodes in the global scale network or substrate with other nodes in the global scale network. The function for C_{global} grows as $\mathcal{O}\left(\frac{1}{k_{global}!}\right)$. It must be noted here that in the computation of C_{local} and C_{global} ignores border effects and assumes nodes are spread on a torus. To obtain the clustering coefficient of the network C_{local} and C_{global} are combined in the ratios in which local and global scale nodes appear. If we define $\mu = \frac{N_{global}}{N_{global}+N_{local}}$ then C is obtained by equation 7.12.

$$C = \mu C_{global} + (1 - \mu) C_{local} \quad (7.12)$$

7.4.2 Characteristic Path Length

The *characteristic path length* of a graph is defined as the average length of all geodesic paths (measured in hops). In a graph $G(V, E)$ the characteristic path length L is defined as the average of geodesic path lengths $l(v_i, v_j)$ between all connected pairs of vertices $v_i, v_j \in V$ and $v_i \neq v_j$. Like the clustering coefficient, the characteristic path length is also a function of the number of nodes with global scale links. Since the global scale links are range-limited, and since global scale nodes have to be placed within communication range of one other to be useful, the area covered by them is also limited. The region of the WSN that is serviced by global scale nodes may be fragmented into many different patches. Figure 7.3 shows a model of a WSN. For analytical ease the shape of the region covered by all nodes of the network is approximated by a circle of radius \sqrt{A} and center O . The region covered by global scale nodes is approximated by a circle of radius $\sqrt{A_g} < \sqrt{A}$ also with center O . The difference between coverage areas of the entire WSN and only the global scale nodes is a result of the constraint imposed by a) the limited range of shortcuts and b)

Fig. 1
Fig. 2
Fig. 3
Fig. 4
Fig. 5
Fig. 6
Fig. 7
Fig. 8
Fig. 9
Fig. 10
Fig. 11
Fig. 12
Fig. 13
Fig. 14
Fig. 15
Fig. 16
Fig. 17
Fig. 18
Fig. 19
Fig. 20
Fig. 21
Fig. 22
Fig. 23
Fig. 24
Fig. 25
Fig. 26
Fig. 27
Fig. 28
Fig. 29
Fig. 30
Fig. 31
Fig. 32
Fig. 33
Fig. 34
Fig. 35
Fig. 36
Fig. 37
Fig. 38
Fig. 39
Fig. 40
Fig. 41
Fig. 42
Fig. 43
Fig. 44
Fig. 45
Fig. 46
Fig. 47
Fig. 48
Fig. 49
Fig. 50
Fig. 51
Fig. 52
Fig. 53
Fig. 54
Fig. 55
Fig. 56
Fig. 57
Fig. 58
Fig. 59
Fig. 60
Fig. 61
Fig. 62
Fig. 63
Fig. 64
Fig. 65
Fig. 66
Fig. 67
Fig. 68
Fig. 69
Fig. 70
Fig. 71
Fig. 72
Fig. 73
Fig. 74
Fig. 75
Fig. 76
Fig. 77
Fig. 78
Fig. 79
Fig. 80
Fig. 81
Fig. 82
Fig. 83
Fig. 84
Fig. 85
Fig. 86
Fig. 87
Fig. 88
Fig. 89
Fig. 90
Fig. 91
Fig. 92
Fig. 93
Fig. 94
Fig. 95
Fig. 96
Fig. 97
Fig. 98
Fig. 99
Fig. 100

the *global vertex degree* (the number of shortcut links incident on a substrate node). Figure 7.3 also shows a node x distance removed from O . For the node in the figure $x > \sqrt{A_g}$ and the sensor region can be divided into three separate regions, numbered 1 through 3 in the figure. These regions are defined relative to every node v . Region 1 consists of all sensors that are reachable from v without having to traverse any global scale links. In figure 7.3, for node v at distance x from center O , region 1 corresponds to the uncolored white region. Region 2 is the area of size A_g occupied by global scale nodes and is colored solid gray. Region 3 is the region occupied exclusively by sensors with local scale links but which is reachable (by a shortest path) only after traversing region 2. In figure 7.3 region 3 is identified by shading and its area is denoted by $S(x)$. Assuming the deployment of sensors is sufficiently dense, we approximate the geodesic path from one node to another by a straight line between them. To proceed we define three central angles $\alpha = \frac{1}{2}\angle BOB'$, $\beta = \angle BOJ = \angle B'OJ'$ and $\gamma = \angle JOJ'$, where $2(\alpha + \beta) + \gamma = 2\pi$. Since α and γ are functions of x we will be denoting them by $\alpha(x)$ and $\gamma(x)$.

$$\alpha(x) = \arccos \frac{\sqrt{A_g}}{x} \quad (7.13)$$

$$\beta = \arccos \frac{\sqrt{A_g}}{\sqrt{A}} \quad (7.14)$$

$$\gamma(x) = 2\pi - 2\alpha(x) - 2\beta = 2(\pi - \alpha(x) - \beta) \quad (7.15)$$

VA

V

Figur
min

The

Sten

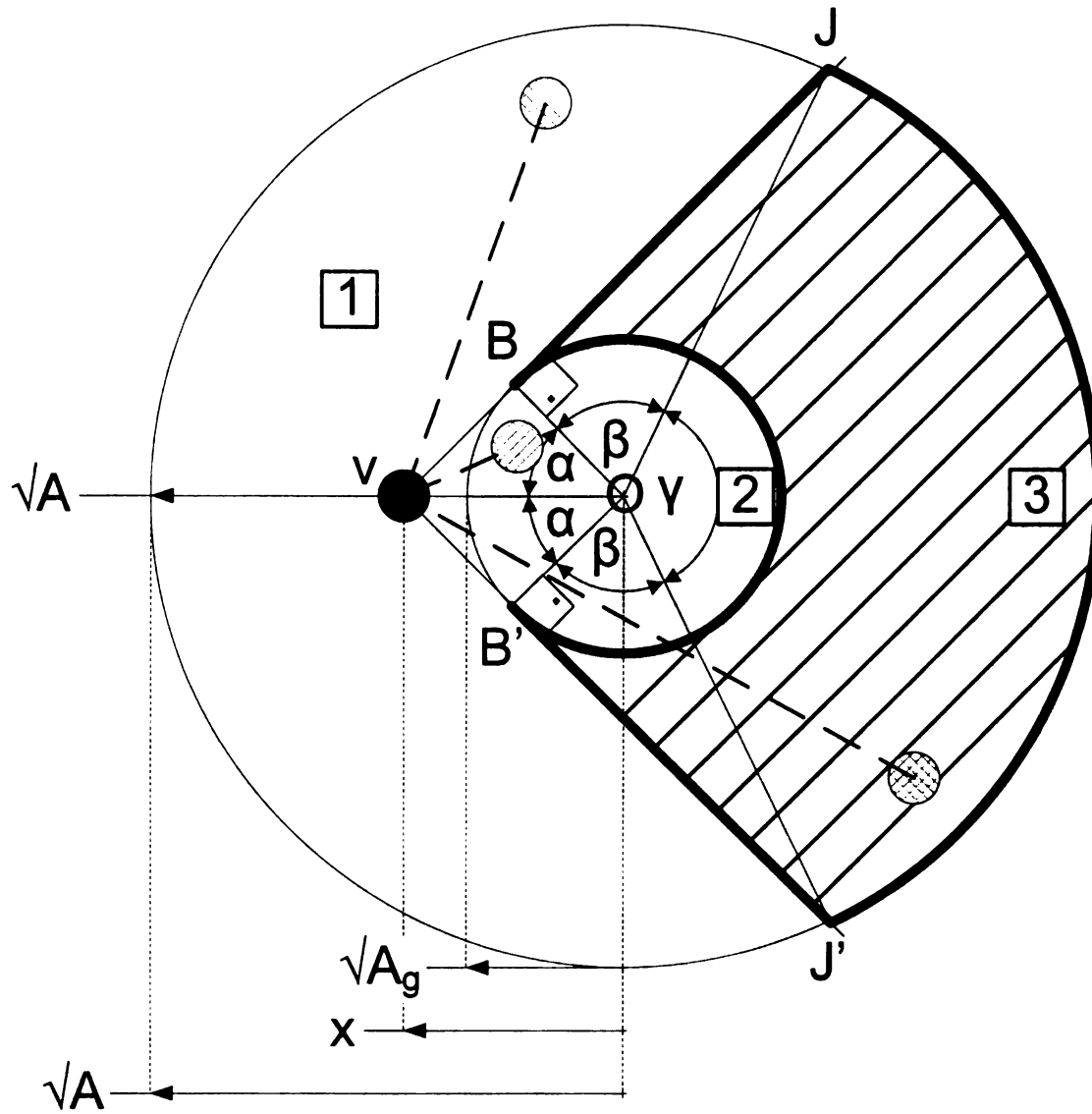


Figure 7.3: Geometry of WSN deployment and regions within it with respect to an individual sensor v_i .

The area of the region in figure 7.3 labeled region 3 is (see lemma 2 in the appendix) given in equation 7.16,

$$S(x) = 2\sqrt{AA_g} \sin \beta + \frac{A}{2}(\pi - \alpha(x) - \beta) - \frac{2\beta + \gamma}{2} A_g \quad (7.16)$$

Gen. Br.
By A.
Gen. Br.
L. Br.
L. Br.

Gen. Br.
L. Br.

•

•

•

•

Ne con

L. Br.

Characteristic path length is denoted by L . It is the average length (in hops) of all geodesic paths from every node $v_i \in V$ to every other node $v_j \in V$. L_{local} is the average number of hops contributed to geodesic paths by local scale links, while L_{global} is the average number of hops contributed to geodesic paths by global scale links. Then L can be expressed as equation 7.17.

$$L = L_{local} + L_{global} \quad (7.17)$$

Based on the 3 regions of the WSN in figure 7.3, geodesic paths are classified into one of four different types.

- **Type 1:** Geodesic paths that originate from and terminate at nodes in region 1 and (outside the region covered by the substrate). Type 1 paths consist of local scale links only.
- **Type 2:** Geodesic paths that originate from a node in region 1 and terminate at a node in region 2.
- **Type 3:** Geodesic paths that originate from a node in region 1 and terminate at a node in region 3. Such paths pass through region 2.
- **Type 4:** Geodesic paths that originate from and terminate at nodes in region 2.

We compute L_{local} (L_{global}) in equation 7.18 (equation 7.19) as the sum of $L_{local}^{(i)}$ ($L_{global}^{(i)}$), the contributions of local (global) scale links to type i paths, weighted

by the

For a

spatial

model

linear

space

of the

approx

And

1. To

of the

by the normalized frequency of occurrence of type i paths.

$$L_{local} = \frac{1}{R} \sum_{i=1}^4 L_{local}^{(i)} \quad (7.18)$$

$$L_{global} = \frac{1}{\xi R} \sum_{i=1}^4 L_{global}^{(i)} \quad (7.19)$$

For a node v in region 1 at $x > A_g$ distance from O , let $l_{local}^{(1)}(x)$ denote the average spatial distance on type 1 paths traversed over local scale links. Then for a particular node the size of region 1 is $A - A_g - S(x)$. Since the deployment of sensors is in 2 dimensions we approximate the size of the region traversed by these paths by the square root of its area, $\sqrt{A - A_g - S(x)}$. Although region 1 is irregularly shaped we approximate the spatial distance by one half of it, assuming the sensor is located approximately in the center of region 1. Then $l_{local}^{(1)}(x)$ is given by equation 7.20.

$$l_{local}^{(1)}(x) = \frac{1}{2} \sqrt{A - A_g - S(x)} \quad (7.20)$$

A node v at distance $x > \sqrt{A_g}$ from O has $\rho l_{local}^{(1)}(x)$ paths to other nodes in region 1. To obtain $L_{local}^{(1)}$ we integrate this term over the entire region of deployment of the WSN, i.e. all circumferences of radii ranging from $\sqrt{A_g}$ through \sqrt{A} . This is

shown in equation 7.21.

$$L_{local}^{(1)} = \frac{1}{(\rho' A)^2} \int_{\sqrt{A_g}}^{\sqrt{A}} \rho 2\pi x \cdot \rho (A - A_g - S(x)) l_{local}^{(1)}(x) dx \quad (7.21)$$

Since the paths from v to other sensors in region 1 do not traverse region 2 no global scale links are used. Hence $L_{global}^{(1)}$, the component of global scale links in paths to sensors in region 1 is 0, as in equation 7.22.

$$L_{global}^{(1)} = 0 \quad (7.22)$$

For a node v in region 1 at $x > A_g$ distance from O , let $l_{local}^{(2)}(x)$ denote the average spatial distance on type 2 paths traversed over local scale links. It is approximated by the mean of the shortest $(x - A_g)$ and longest distances (from node v to point B) from v to a node in region 2, i.e. $\frac{1}{2}(x - \sqrt{A_g} + x \sin \alpha(x))$. To include paths from nodes in region 2 back to node v we multiply this term by 2. This is approximated by equation 7.23. We roughly approximate $l_{global}^{(2)}(x)$ by the square root of the area of region 2 in equation 7.25.

$$l_{local}^{(2)}(x) = x - \sqrt{A_g} + x \sin \alpha(x) \quad (7.23)$$

$L_{local}^{(2)}$ (equation 7.18) is the component of L_{local} contributed by node v 's type 2 paths, i.e. to other nodes in region 2. A node v at distance $x > \sqrt{A_g}$ from O has $\rho'' A_g$ paths to other nodes in region 2. To obtain $L_{local}^{(2)}$ we integrate this term over all circumferences of radii ranging from $\sqrt{A_g}$ through \sqrt{A} . This is shown in

equation 7.24.

$$L_{local}^{(2)} = \frac{1}{(\rho' A)^2} \int_{\sqrt{A_g}}^{\sqrt{A}} \rho 2\pi x \cdot \rho'' A_g l_{local}^{(2)}(x) dx \quad (7.24)$$

The length of global scale paths from v to sensors in region 2 is approximated like in equation 7.25.

$$l_{global}^{(2)}(x) = \sqrt{A_g} \quad (7.25)$$

Equation 7.26 weighs this by the number of paths from all v outside of region 2 to all sensors inside region 2 (and vice versa) relative to the number of paths between all pairs of nodes, i.e. $(\rho' A)^2$.

$$\begin{aligned} L_{global}^{(2)} &= \frac{1}{(\rho' A)^2} \int_{\sqrt{A_g}}^{\sqrt{A}} \rho 2\pi x \rho'' A_g \sqrt{A_g} dx \\ &= \frac{\rho \rho'' \pi A_g^{3/2} (A - A_g)}{(\rho' A)^2} \end{aligned} \quad (7.26)$$

For a node v in region 1 at $x > A_g$ distance from O , let $l_{local}^{(3)}(x)$ denote the average spatial distance on type 3 paths traversed over local scale links. It is approximated by equation 7.27.

$$\begin{aligned} l_{local}^{(3)}(x) &= l_{local}^{(2)}(x) + \frac{\sqrt{A} - x}{2} \\ &= \frac{1}{2}(x + \sqrt{A} - \sqrt{A_g} + 2x \sin \alpha(x)) \end{aligned} \quad (7.27)$$

A node v at distance $x > \sqrt{A_g}$ from O has $\rho S(x)$ paths to other nodes in region 3. To obtain $L_{local}(3)$ we integrate this term over all circumferences of radii ranging from $\sqrt{A_g}$ through \sqrt{A} . This is shown in equation 7.28.

$$L_{local}(3) = \frac{1}{(\rho' A)^2} \int_{\sqrt{A_g}}^{\sqrt{A}} \rho 2\pi x \cdot \rho S(x) l_{local}^{(3)}(x) dx \quad (7.28)$$

For $L_{global}(3)$ in equation 7.29 we approximate the length of global scale links by the diameter of region 2, $2\sqrt{A_g}$, and weigh it by the total number of paths from all v to nodes in region 3, $\rho S_d(x)$.

$$L_{global}(3) = \frac{1}{(\rho' A)^2} \int_{\sqrt{A_g}}^{\sqrt{A}} \rho 2\pi x \cdot \rho S(x) 2\sqrt{A_g} dx \quad (7.29)$$

Finally, for a node v in region 2 at distance $x < A_g$ from O , let $l_{local}^{(4)}$ denote the average spatial distance on type 4 paths traversed over local scale links. We assume that on average, a node v in region 2 communicating with another node in region 2 will route (over local links) to a nearby node with global scale links. Once traffic has reached the vicinity of the target destination node traffic is routed over local scale links to the destination. Let h_{max} denote the maximum distance of a node from its closest global scale node, as expressed in equation 7.30.

$$h_{max} = \arg \max_h \sum_{h=1} \rho' \left[(hR)^2 - ((h-1)R)^2 \right] \leq \frac{\pi A_g}{N_{global}} \quad (7.30)$$

Assuming that equidistant nodes form concentric bands around global scale nodes the probability mass function $p_H(h)$ of the hop distance H is modeled by equation 7.31.

$$p_H(h) = \frac{\pi(hR)^2 - \pi((h-1)R)^2}{\pi(h_{max}R)^2} = \frac{2h-1}{h_{max}^2}. \quad (7.31)$$

Then the $l_{local}^{(4)}$ is approximated by the expected value of H as in equation 7.32.

$$l_{local}^{(4)} = E_H[h] = \sum_{h=1}^{h_{max}} h \frac{2h-1}{h_{max}^2} \quad (7.32)$$

A node v at distance $x < \sqrt{A_g}$ from O , within region 2, has $\rho'' A_g$ paths to other nodes in region 2. To obtain $L_{local}^{(4)}$ we integrate this term over all nodes in region 2. This is shown in equation 7.33.

$$L_{local}^{(4)} = \frac{2(\rho'' A_g)^2}{(\rho' A)^2} l_{local}^{(4)} \quad (7.33)$$

$$L_{global}^{(4)} = \frac{2(\rho'' A_g)^2}{(\rho' A)^2} \sqrt{A_g} \quad (7.34)$$

Now we can back substitute equations 7.21, 7.24, 7.28 and 7.33 into equation 7.18 to obtain L_{local} . Similarly, equations 7.22, 7.26, 7.29 and 7.34 are put back into equation 7.19 to obtain L_{global} . Equations 7.18 and 7.19 are substituted into 7.17 to obtain the characteristic path length L .

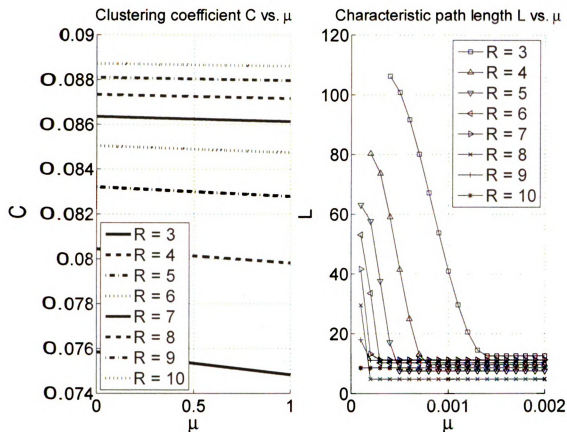


Figure 7.4: Plots of clustering coefficient C [left] and characteristic path length L , [right] as functions of μ for different values of R . Network parameters that remain fixed are $A = 10000$, $k_{global} = 4$, $\rho = 10$, $\xi = 3$ and $n_c = 1$.

7.5 Observations

In this section we explore the behavior of the analytical models of clustering coefficient C and characteristic path length L as we vary the length of local scale links R , the average degree of shortcut nodes in the substrate k_{global} and the scaling factor ξ . For the following evaluations we assumed a network of 100,000 nodes in an area of size $A = 10,000$ and $n_c = 1$. Unless stated otherwise, the default parameters of the network are $R = 3$, $\xi = 3$ and $k_{global} = 4$. Figure 7.4 plots the clustering coefficient and characteristic path length of networks as functions of μ , the fraction of nodes that are shortcut nodes, for different values of R . The clustering coefficient

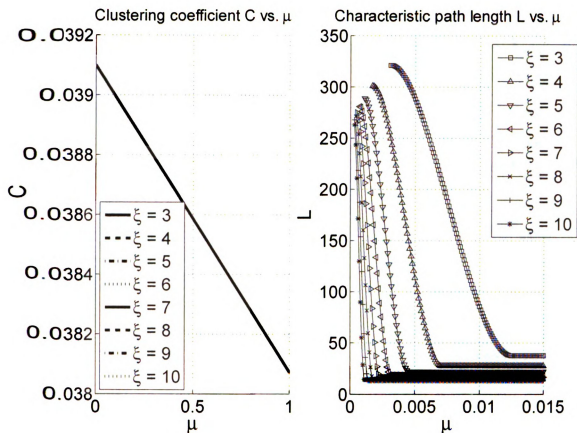


Figure 7.5: Plots clustering coefficient C [left] and characteristic path length L , [right] as a function of μ for different ratios of global scale link to local scale link communication range ξ . Network parameters remain fixed at $A = 10000$, $k_{global} = 4$, $\rho = 10$, $n_c = 1$ and $R = 4$.

C appears as a linearly decreasing function of μ with identical slopes. The clustering coefficient has higher values for longer range R of local scale links. The right pane of figure 7.4 plots the characteristic path length for various R . Higher values of R imply that destinations can be reached in fewer hops, thus reducing the characteristic path length. This is born out by the curves of L . Moreover, a side-by-side comparison clearly illustrates the phase difference in the reduction of C and L . This means the derived analytical model indeed captures the small world effect in the network. Unlike in Watts' β -model [128] of constructing small world networks, C lacks a sharp drop off as $\mu \rightarrow 1$. The reason for this is the range limitation on shortcuts which imposes a

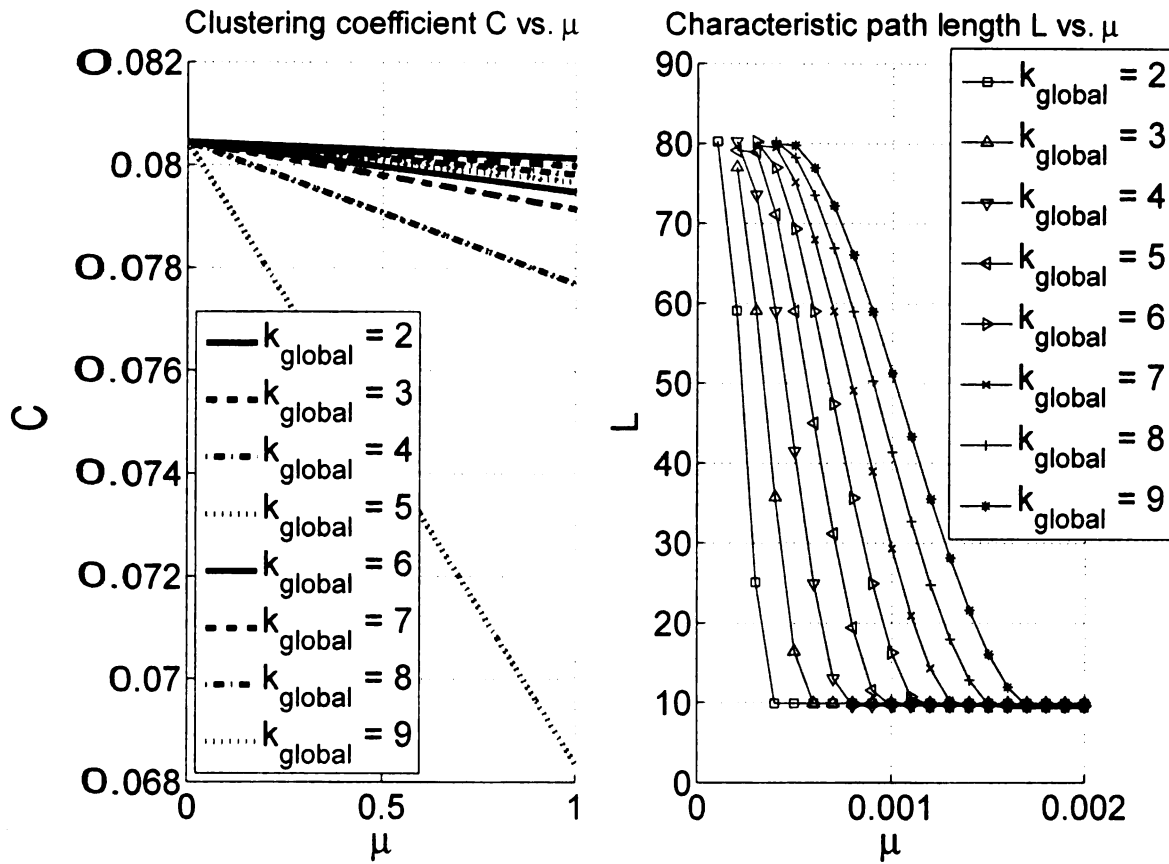


Figure 7.6: Plots clustering coefficient C [left] and characteristic path length L , [right] as a function of μ for different values of k_{global} . Network parameters remain fixed at $A = 10000$, $\rho = 10$, $\xi = 3$, $R = 3$ and $n_c = 3$.

degree of localization in a shortcut link's reach which leads to clustering. In addition, all construction methods of small world WSN in section 7.2 add shortcuts on top of the existing geometric graph instead of rewiring existing links as the β -model does where each global scale link is added at the cost of removing a local scale link.

Figure 7.5 plots clustering coefficient C and characteristic path length L as functions of μ for different values of global scale link scaling factors ξ . Recall that in deriving the analytical model for the clustering coefficient we made the assumption that $\xi \geq 2$, i.e. shortcut links have a communication range of at least twice that of local scale links. Note that in figure 7.5 all plotted function of C overlap, i.e. as long as the assumption $\xi \geq 2$ holds, the clustering coefficient as a function of μ is

independent of the global scaling factor ξ . The small world network effect is clearly **visible** by the large gap between clustering coefficient and characteristic path length. **As expected**, the drop in characteristic path length L drops off earlier as ξ increases, i.e. **fewer** shortcuts are necessary to achieve the same reduction in L as shortcuts get longer.

For the networks in figure 7.6 we varied the average number of shortcuts k_{global} incident on global scale node. Larger values of k_{global} imply a slower expansion of the area within a WSN that is serviced by shortcut nodes (region 2 in figure 7.3). **Figure 7.6** shows that as k_{global} increases, the rate at which the clustering coefficient **drops** increases. We also observe that the number of global scale nodes needed to achieve a significant reduction in L also increases as evidenced by higher value of μ to achieve the same characteristic path length L for higher k_{global} . That is because at higher k_{global} the range limited nature of shortcuts forces more global scale nodes to be deployed in closer proximity.

However, we observe consistently that the small world effect holds in 2-dimensional spatial graphs with range limited shortcuts such as those formed by WSNs with some shortcuts. What is also interesting is the fact that for a sufficiently dense WSN the number of shortcuts that is needed to achieve a significant reduction in L is very small.

7.6 Conclusions

We derived analytical models for both clustering coefficient and characteristic path length of a 2-dimensional spatial graph with range limited shortcuts that models the topological constraints to which WSNs are subject.

1. **The** model is sufficiently general to accommodate any small world network construction methods in wireless networks previously proposed for WSNs and

is, to the authors' best knowledge, the first analytical model for networks with these constraints.

2. We observed that for sufficiently dense networks characteristic path length can be reduced significantly by replacing a $\mu \approx \mathcal{O}(0.005 - 0.05)$ fraction of the local scale nodes by global scale nodes providing shortcuts in the network. The order of μ , the fraction of nodes that are designated shortcut nodes, is about the same as the value of β , the rewiring probability, in Watts' small world network construction method.
3. Whichever small world network construction method is applied carries with it a cost. The model lends itself for the task of designing WSNs, e.g. determining the number of shortcut nodes required to achieve a certain characteristic path length.

APPENDIX

Lemma 1. *We derive the expression for $d(\Gamma(v))$, the number of links between the set of nodes consisting of v and all its neighbors $\Gamma(v)$. The region covered by $\Gamma(v)$ is approximated by a circular region of radius R centered at v . Consider another node v' at a distance r from node v shown in figure 7.7. Then the number of links from v' to other nodes in $\Gamma(v)$ is $\Delta(r)\rho'$. The narrow ring of width dr is inhabited by $2\pi r\rho'$ other nodes at the same distance r from v . The number of links from all nodes at distance r from v is $2\pi r\rho'\Delta(r)\rho'$. Then, to find $d(\Gamma(v))$, the total number of links between nodes v and $\Gamma(v)$, we integrate $2\pi r\rho'\Delta(r)\rho'$ with respect to r over $[0, R]$, the*

radii of communication of v .

$$\begin{aligned}
d(\Gamma(v)) &= \int_0^R \rho' 2\pi r \Delta(r) \rho' dr \\
&= \int_0^R \rho'^2 2\pi r \left(2R^2 \arccos \frac{r}{2R} - \frac{Rr}{2} \sqrt{1 - \frac{r^2}{4R^2}} \right) dr \\
&= 4\pi R^2 \rho'^2 \int_0^R r \arccos \frac{r}{2R} dr - \frac{\pi}{2} \rho'^2 \int_0^R r^2 \sqrt{4R^2 - r^2} dr \\
&= 4\pi R^2 \rho'^2 \cdot \mathcal{A}(R) - \frac{\pi}{2} \rho'^2 \cdot \mathcal{B}(R)
\end{aligned} \tag{7.A-1}$$

Where $\mathcal{A}(R)$ and $\mathcal{B}(R)$ are evaluated in equations 7.A-2 and 7.A-3.

$$\begin{aligned}
\mathcal{A}(R) &= \left| \arcsin \left(\frac{r}{2R} \right) R^2 - \frac{1}{2} r \sqrt{1 - \frac{r^2}{4R^2}} R \right. \\
&\quad \left. + \frac{1}{2} r^2 \arccos \left(\frac{r}{2R} \right) \right|_0^R \\
&= \arcsin \left(\frac{1}{2} \right) R^2 - \frac{\sqrt{3}}{4} R + \frac{1}{2} R^2 \arccos \left(\frac{1}{2} \right) \\
&= \frac{\pi}{3} R^2 - \frac{\sqrt{3}}{4} R
\end{aligned} \tag{7.A-2}$$

$$\mathcal{B}(R) = \left| \frac{\frac{1}{3} r^3 \sqrt{4R^2 - r^2} {}_2F_1 \left(1.5; -0.5; 2.5; \frac{r^2}{4R^2} \right)}{\sqrt{1 - \frac{r^2}{4R^2}}} \right|_0^R \tag{7.A-3}$$

${}_2F_1(a; b, c, z)$ denotes the hypergeometric function (equation 7.A-4) and is evaluated in equation 7.A-5,

$${}_2F_1(a; b; c; z) = \frac{\Gamma(c)}{\Gamma(b)\Gamma(c-b)} \int_0^1 \frac{t^{b-1}(1-t)^{c-b-1}}{(1-tz)^a} dt \quad (7.A-4)$$

$$\begin{aligned} {}_2F_1\left(1.5; -0.5; 2.5; \frac{r^2}{4R^2}\right) &= \frac{\Gamma(2.5)}{\Gamma(-0.5)\Gamma(3)} \int_0^1 \frac{t^{-1.5}(1-t)^2}{\left(1 - \frac{r^2}{4R^2}t\right)^{1.5}} dt \\ &= -0.1875 \int_0^1 \frac{t^{-1.5}(1-t)^2}{\left(1 - \frac{r^2}{4R^2}t\right)^{1.5}} dt \end{aligned} \quad (7.A-5)$$

Substituting the value of the hypergeometric function from 7.A-5 into equation 7.A-3 produces equation 7.A-6.

$$\mathcal{B}(R) = \frac{2}{3}R^4 \cdot {}_2F_1(1.5; -0.5; 2.5; 0.25) - 0 = 0.6142 \cdot R^4 \quad (7.A-6)$$

Substituting $\mathcal{A}(R)$ and $\mathcal{B}(R)$ from equations 7.A-2 and 7.A-6 back into equation 7.A-1 gives equation 7.A-7.

$$d(\Gamma(v)) = 2.9604\pi\rho'^2 R^4 - \sqrt{3}\pi\rho'^2 R^3 \quad (7.A-7)$$

Lemma 2. We compute the area of region 3 in figure 7.3 for a node x distance from O denoted as $S(x)$, where $x > \sqrt{Ag}$. In order to proceed we compute the area of triangle BOJ denoted by $\Lambda(x)$. For $x > \sqrt{Ag}$, $\Lambda(x)$ can be expressed in previously defined terms as,

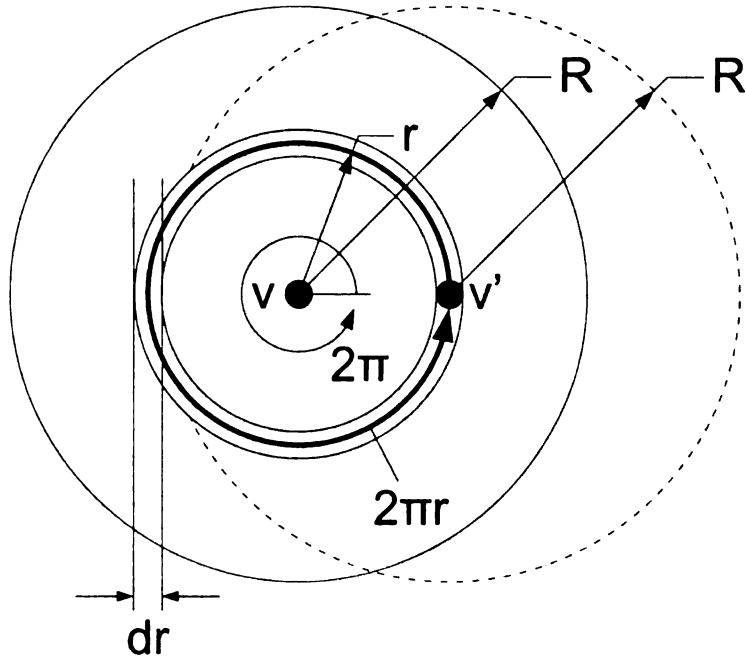


Figure 7.7: Graphical representation of integration term of $d(\Gamma(v))$.

$$\Lambda = \sqrt{A_g} \cdot \sqrt{A} \sin \beta \quad (7.A-8)$$

Similarly, the area of sector JOJ' with central angle γ and radius \sqrt{A} is denoted by $\widehat{\Lambda}(x)$. For $x > \sqrt{A_g}$, $\widehat{\Lambda}(x)$ can be expressed in previously defined terms as,

$$\widehat{\Lambda}(x) = \frac{\gamma(x)}{2\pi} \pi \sqrt{A}^2 = \frac{A}{2} (\pi - \alpha(x) - \beta) \quad (7.A-9)$$

The area of sector BOB' with central angle $2\beta + \gamma$ and radius $\sqrt{A_g}$ is denoted by $\widehat{\widehat{\Lambda}}(x)$. The, for $x > \sqrt{A_g}$, $\widehat{\widehat{\Lambda}}(x)$ can be expressed as,

$$\widehat{\widehat{\Lambda}}(x) = \frac{2\beta + \gamma(x)}{2\pi} \pi \sqrt{A_g}^2 = \frac{2\beta + \gamma(x)}{2} A_g \quad (7.A-10)$$

We now use $\Lambda(x)$, $\widehat{\Lambda}(x)$ and $\widehat{\widehat{\Lambda}}(x)$ to obtain the area of region 3 denoted by $S(x)$.

$$\begin{aligned} S(x) &= 2\Lambda(x) + \widehat{\Lambda}(x) - \widehat{\widehat{\Lambda}}(x) \\ &= 2\sqrt{AA_g} \sin \beta + \frac{A}{2}(\pi - \alpha(x) - \beta) - \frac{2\beta + \gamma(x)}{2} A_g \end{aligned} \quad (7.A-11)$$

Chapter 8

Enabling Cooperative Communication and Diversity Combination in IEEE 802.15.4 Wireless Networks Using Off-the-shelf Sensor Motes

8.1 Introduction

Channel fades and interference effects limit the throughput, useful communication range and (in case of battery powered devices) lifetime of nodes. In this chapter we describe the generalized ‘Poor Man’s SIMO System’ (gPMSS), a readily deployable low-cost, low-power, protocol centric approach that enables cooperative communication in IEEE 802.15.4 [5] wireless sensor networks (WSN). We demonstrate that gPMSS reduces the fraction of packets that are received with bit errors or not received at all by an order of magnitude, thus reducing the number of retransmissions. It makes the use of long range links that are unfeasible due to high packet loss and retransmission rates feasible again. We also show that even in instances where gPMSS is not able to correct all errors from a packet it still succeeds in reducing the number of bit errors. At the receiver side gPMSS uses diversity combining methods adapted from their analog domain counterparts of the same name [19] for digital signals. What makes the application of SIMO diversity combining principles novel from traditional use is that they are applied to the demodulated version of received packets, after Physical layer processing. We demonstrate the efficacy of gPMSS by applying it to bit error traces collected from IEEE 802.15.4 channels that allow detailed analysis and precise reproduction of results. We also demonstrate gPMSS’ effectiveness under real-world conditions by implementation on Crossbow’s Imote2 .NET Micro Framework sensor platform [35].

Enabling the use of long range links (that would otherwise not be used) makes gPMSS a viable protocol due to the benefits and utility of such links by several applications in wireless sensor networks.

Network Lifetime Extension: *Funneling* is the effect of network traffic from multiple sources flowing to a small number of sink nodes [125]. This traffic surge produces congestion in the region around the sink nodes/base station, forcing nodes near sink

nodes to relay more traffic than other nodes and consume power at correspondingly **higher** rates. Since nodes in WSNs have only limited power resources this means that **the sink** node's neighbors will run out of power sooner, leaving the sink node **disconnected** from the rest of the network. Load balancing techniques like [53] attempt to **distribute** the burden of relaying traffic to increase the lifetime of sensor networks. **Employing** gPMSS in such a scenario will grow the set of neighbor nodes of the sink **node** and allow load balancing among more nodes.

Small-world Networks: Several attempts have been made at building small-world **network** [128] topologies in wireless networks to simplify resource discovery and **reducing** average path length to facilitate data dissemination. Proposed architectures **required** hardware modifications such as adding a secondary radio frequency interface ([124],[125]) or building hybrid networks by augmenting wireless networks with wired **shortcuts** ([111],[27]). Since gPMSS is a protocol centric approach it does not require **any** hardware modifications which adds to its appeal as a low-complexity and low-cost **solution**.

Network Connectivity: Long range links can be used to add links between two **components** of a network that are only sparsely connected with one another.

gPMSS adopts well-understood diversity combining methods for analog signals **and** applies them to digital signals (packets). Specifically, gPMSS implements **selection** diversity, equal gain diversity combining and maximal-ratio gain diversity **combining**. The latter relies on a model of the instantaneous bit error rate (BER) driven by channel state information (CSI) [55], i.e. received signal strength indication (RSSI) **and** link quality indication (LQI). We provide proof of concept by applying gPMSS **to** channel traces and demonstrate one order of magnitude reduction in packet losses. **Applying** gPMSS to traces allows more detailed analysis and reproducibility that is **not** possible in a live setup, i.e. the event when receivers are not able to reconstruct **an** error-free version of the transmission. We show that even then we are able to

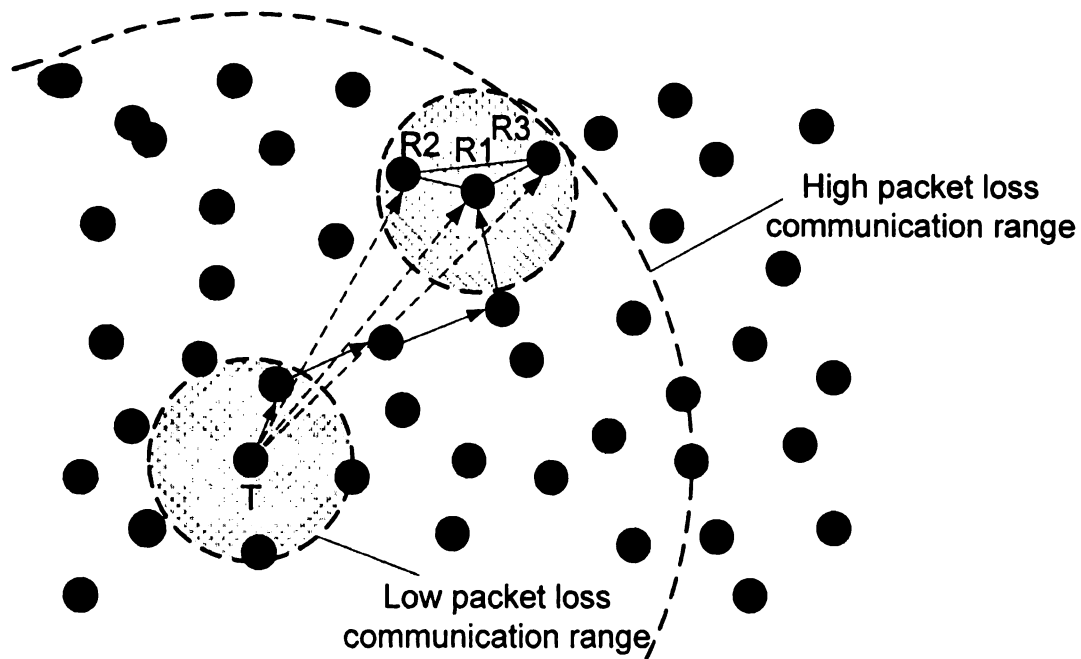


Figure 8.1: Application of generalized gPMSS in a wireless sensor network with mesh topology. Path from transmitter T to receiver $R1$ marks the multihop path that would be taken in a network without gPMSS. Dashed line links between T and receivers $R1$, $R2$ and $R3$ denote the longer range but high loss links that are used under Generalized gPMSS.

significantly reduce the average BER of incorrigible packets. Finally, we implement gPMSS on Imote2 sensor motes [35] using C# and demonstrate a clear reduction in packet losses. Experimental results from IEEE 802.15.4 links indicate that using diversity combining raises packet reception rate (PRR) by up to an additional 130% over those in a single receiver.

Our contributions are threefold;

1. gPMSS is a protocol centric, cross-layer approach which means it can be used in presently deployed wireless sensor networks by making software changes only. It does not require any modifications to hardware but runs on networks of commercial off-the-shelf (COTS) single antenna sensor motes.
2. gPMSS is non-intrusive in the sense that it does not require changes to the pre-existing IEEE 802.15.4 standard.

3. gPMSS is able to reduce power consumed at the transmitter per packet delivered by up to 68%. This represents a significant increase in the lifetime of sensor node.

Figure 8.1 illustrates the difference between routes traversed by a packet sent by transmitter T to a distant node $R1$ when gPMSS is used (dotted arrows represent long range links, solid lines represent links between $R1$, $R2$, $R3$ that form a fully connected graph), and the multi-hop path from node T to $R1$ when it is not used (solid arrows).

The remainder of this chapter is organized as follows. Section 8.2 reviews some related works. Section 8.3 describes the three diversity combining techniques for packet recovery. Section 8.4 describes the gPMSS that enables cooperation between multiple receivers. Section 8.5 describes the trace collection setup and demonstrates a proof of concept of gPMSS in a manner that can be reproduced. Section 8.6 describes the gPMSS implementation on Imote2 and its results. Section 8.7 discusses our results in terms of PRR, retransmission attempts and energy consumption per packet. Section 8.8 concludes this chapter.

8.2 Related Work

The concept of spatial receiver diversity is not new and has been studied extensively in the analog signal domain. Chakraborty et al. proposed the extended automatic repeat-request (ARQ) scheme [24] that recombines spatially diverse versions of a received packet to detect bit errors and an exhaustive search to correct them if their number is less than a threshold value. Extended ARQ has a lot in common with the version of gPMSS that uses equal gain combining and is agnostic of what MAC standard is used, but the results provided in [24] are based on theoretical analysis only. Miu et al. [83] proposed a system that used transmitter diversity to increase

packet reception rate in IEEE 802.11 [51] networks with multiple access points (AP) as senders. The scheme roughly corresponds to gPMSS with selection diversity, without diversity combination for error correction. Miu et al. generalized this approach in [85] for applications beyond streaming video. In [84] Miu extended the idea further to reduce packet losses on the uplink (mobile device to AP). However, this required modifications to IEEE 802.11b AP hardware or deployment of more APs, and uses a dedicated frame combiner connected to all APs through a wired network. It used the equal gain method for detecting bit errors and, like extended ARQ, relied on an exhaustive search of the correct bit values. Cheng and Valenti [26], [123] extended the idea for improving throughput on uplinks in IEEE 802.11a networks by using maximal ratio combining based on CSI measurements. However, like Miu's system it still required a dedicated combiner connected to all APs. In [60] Ji et al. proposed an approach for improving the throughput of downlinks by scheduling transmissions to multiple receivers in IEEE 802.11a/b networks based on explicit feedback from receivers while maintaining fairness. In [7] Bahl made the case for multi-radio transceivers, but as figure 4 in his paper showed, collaboration between network interfaces is possible only when they are all located on the same device. More recently, Woo described SOFT [134] which also exploited receiver diversity for the uplink in IEEE 802.11 networks similar to Miu's in [84], but with diversity combining being performed using maximal ratio combining. Therefore, it too requires a centralized combiner on the wired network that all APs are connected to. To summarize, the gPMSS system presented is distinct from all these prior works on cooperative communication and diversity combining in wireless networks because it is (1) designed for IEEE 802.15.4 networks, (2) is purely implemented in software and COTS notes without modifications to mote hardware, (3) is tested on bit error traces collected from real IEEE 802.15.4 channels, (4) as well as actual implementation on notes.

8.3 SIMO Diversity Combining Techniques

The solution that is described in this section is dubbed the generalized 'Poor-Man's-SIMO-System' because it uses receiver side diversity combining techniques and is built using commercial-off-the-shelf components, without customized or reconfigured hardware. Receiver diversity improves link quality of wireless channels with high losses. This way we reduce losses and retransmissions and increase throughput and channel utilization. This subsection describes linear diversity combining techniques. All these techniques are derivatives of the techniques by the same names presented by Brennan in [19]. Brennan describes scanning diversity, selection diversity, equal gain diversity and maximal-ratio diversity combining. Although the methods described by Brennan were meant for analog signals, we have suitably modified and adapted them for use with demodulated, digital signals. We have included the last three, selection, equal gain and maximal-ratio diversity combining. Readers should know that even when the diversity combining method used is either equal gain or maximal-ratio combining, selection diversity is used whenever at least one receiver possesses an error-free version of a transmissions. Equal gain or maximal ratio combining are only used when none of the gPMSS receivers was able to receive error-free (i.e. the situation described in figure 8.5c). The purposes of diversity combining are twofold.

1. Select an error-free version of a received transmission from among all received versions.
2. If the first goal is not achievable, obtain another version of the transmission, with fewer errors than any of the individual received versions.

8.3.1 Selection Diversity

Selection diversity is the simplest diversity combining technique. Figure 8.2 is an **e**quivalent system diagram of the selection diversity process. The basic idea in is to **s**elect from all received packets the one that is expected to have the fewest errors. **T**his is advantageous when it is used in conjunction with forward error correction (**F**EC) because fewer bit errors are easier to correct than more bit errors. When all **r**eceived versions have errors, the best selection diversity can hope to achieve is pick **t**he version with the fewest bit errors. We define the BER of the n^{th} packet in a **s**equences as,

$$BER = \beta[n] = \frac{\# \text{ of error bits in } n^{th} \text{ recvd pkt}}{\# \text{ of bits in } n^{th} \text{ recvd pkt}}. \quad (8.1)$$

Thus the underlying random process producing the sequence of BER observations $\beta[n]$ is called the BER process and is denoted by B . The term BER is not used in its **s**trict traditional sense where it denotes the long term average probability of bit errors, **s**uch as in a binary symmetric channel (BSC). Instead the BER is computed over each **r**eceived packet. Unfortunately, under ordinary circumstances the BER process is not **d**irectly observable. A packet's failure to pass the cyclic redundancy check (CRC) **t**est only tells us that the number of bits with errors is non-zero ($\beta > 0$), but it **d**oes not give any information about the number of errors. Therefore, we must rely **o**n estimates of the BER. The performance of selection diversity will be determined **b**y the accuracy of the model used to predict the BER of packets that fail the CRC **t**est. We have used Ilyas and Radha's [55] CSI measurement-based model of the BER **p**rocess on IEEE 802.15.4 links. For each received packet the model relies on two CSI **p**arameters, i.e. LQI, and RSSI. Measurement of both RSSI and LQI is mandated by

LC
Pa
Co
W
M

the IEEE 802.15.4 LR-WPAN standard for every received packet. The RSSI random process is denoted by P , and RSSI measured by a receiver R for the n^{th} packet in a sequence is denoted by $\rho^R[n]$.

We used the MICAz [34] to demonstrate proof-of-concept and the Imote2 [35] to demonstrate the functioning gPMSS protocol implementation, both of which use the Chipcon CC2420 radio transceiver [120]. Technically, the CC2420 does not measure the LQI directly. Instead, it measures the correlation C between the first 8 received symbols (of the PHY header) and the corresponding 8 known symbols (preamble). IEEE 802.15.4 uses 16-ary Offset-Quadrature Phase Shift Keying modulation which encodes 4 bits in one symbol. The first 8 symbols, 4 bytes, of the PHY header comprise of the Preamble sequence consisting of 32 binary zeros. The LQI is then defined as,

$$LQI = (C - c_1) \cdot c_2. \quad (8.2)$$

In the Chipcon CC2420 c_1 and c_2 are functions of the packet error rate (PER) measured over an extended period of time and are determined experimentally. c_1 and c_2 scale the 7 bit value of the correlation to the range of an 8 bit number. Since equation 8.2 is merely a shifting and scaling of the measured C we take $c_1 = 0$ and $c_2 = 1$. The LQI random process is denoted by Λ , and LQI measured by a receiver R for the n^{th} packet in a sequence is denoted by $\lambda^R[n]$.

Coming back to our description of the CSI-driven BER model of [55], each pair of LQI and RSSI inputs produces a probability density function (PDF) of the BER of Packets received with those particular CSI measurements. To be useful in the current context, the output of the CSI-driven BER model has to be mapped to a single value. We use $\beta_{X\%}$ to denote the X^{th} percentile of the BER process' PDF ($\beta_{50\%}$ is B 's mean). The instances of the BER model return BER estimates denoted as $\beta^{(R1)}$,

$\beta^{(R2)}$ and $\beta^{(R3)}$. The output selector in figure 8.2 receives as input the estimated BERs $\beta^{(R1)}$, $\beta^{(R2)}$ and $\beta^{(R3)}$. Based on these estimates it selects the receiver with the lowest BER estimate as the least error-prone one and accepts its received copy as the best one and outputs it as $D^{(Sel)}$, i.e.

$$D^{(Sel)} = D^{(Rr)} \quad : r = \arg \min_i \beta^{(Ri)}. \quad (8.3)$$

8.3.2 Equal Gain Diversity

The equal gain diversity combining method described here is depicted by an equivalent system diagram in figure 8.3. Recall that like $D^{(T)}$, the three received copies $\tilde{D}^{(R1)}$, $\tilde{D}^{(R2)}$ and $\tilde{D}^{(R3)}$ are vectors of binary numbers (representing bits) obtained after demodulation of the received carrier signal. Essentially, equal gain diversity combining uses received data $\tilde{D}^{(R1)}$, $\tilde{D}^{(R2)}$ and $\tilde{D}^{(R3)}$ to vote on the value of each output bit. In the example in figure 8.3 performs vector addition of $\tilde{D}^{(R1)}$, $\tilde{D}^{(R2)}$ and $\tilde{D}^{(R3)}$, stores the sums in integers and then adjusts the gain by dividing by the number of receivers N , where $N = 3$ in this example. The result will be an array of rational numbers in the range $[0, 1]$. These numbers are thresholded such that values less than 0.5 are remapped to binary zeros, and values greater than (or equal to) 0.5 to binary ones. The output of the thresholder is $D^{(EG)}$. If $S(\cdot)$ is a function representing the operation of the binary decision thresholder, then for an N -receiver gPMSS cluster the equal gain diversity combining process can be represented as;

$$D^{(EG)} = S \left(\frac{1}{N} \sum_{i=1}^N D^{(Ri)} \right). \quad (8.4)$$

Equal gain diversity combining has two advantages over the preceding **selection diversity** combining.

1. It has lower complexity because it does not rely on a BER model.
2. It offers the possibility to recover from all bit errors introduced by the wireless channel, even when all received versions contain errors.

8.3.3 Maximal Ratio Diversity

The maximal ratio diversity combining method described here is depicted by an **equivalent** system diagram in figure 8.4. It combines elements from **selection** and **equal gain** diversity combining. Maximal ratio combining can be described as **equal gain diversity** but with weighted addition. $\tilde{D}^{(R1)}$, $\tilde{D}^{(R2)}$ and $\tilde{D}^{(R3)}$ are each multiplied by weights w_1 , w_2 and w_3 computed as,

$$w_i = \frac{1}{2} - \beta^{(Ri)} \quad \forall 1 \leq i \leq N \quad (8.5)$$

and added. The sum is then re-normalized by dividing by the number of receivers N (in this case $N = 3$) and thresholded which returns the output $D^{(MR)}$ of the **maximal ratio** combining process;

$$D^{(MR)} = S \left(\frac{1}{N} \sum_{i=1}^N (w_i \cdot D^{(Ri)}) \right). \quad (8.6)$$

In the following subsection we proceed to describe the gPMSS protocol that enables cooperation between receivers.

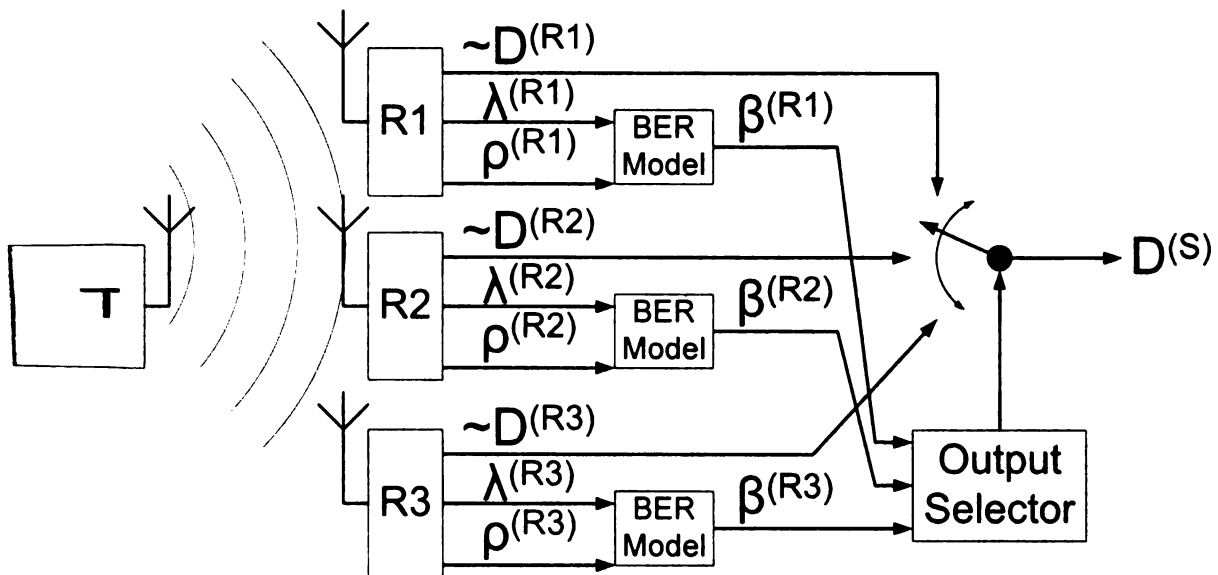


Figure 8.2: Illustration of logical functioning of selection diversity.

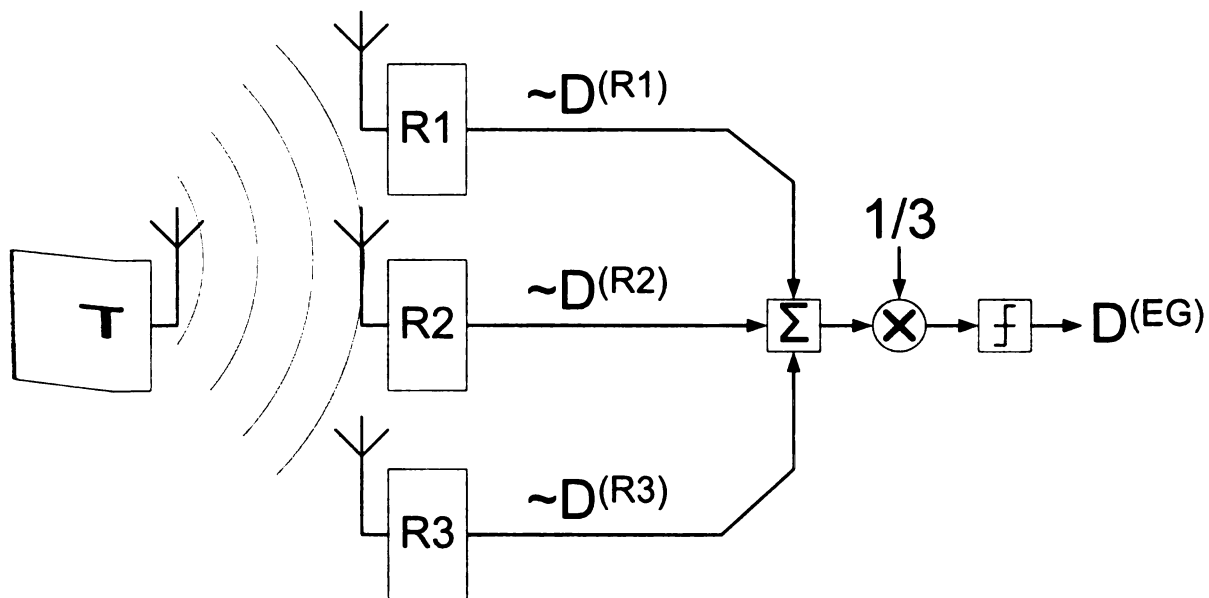


Figure 8.3: Illustration of logical functioning of equal gain diversity.

8.4 gPMSS Protocol

This section describes the operation of the gPMSS protocol. Assume a WSN consisting of a large number of single-antenna COTS receivers communicating over multiple hops with the base station collecting data. According to some topology construction

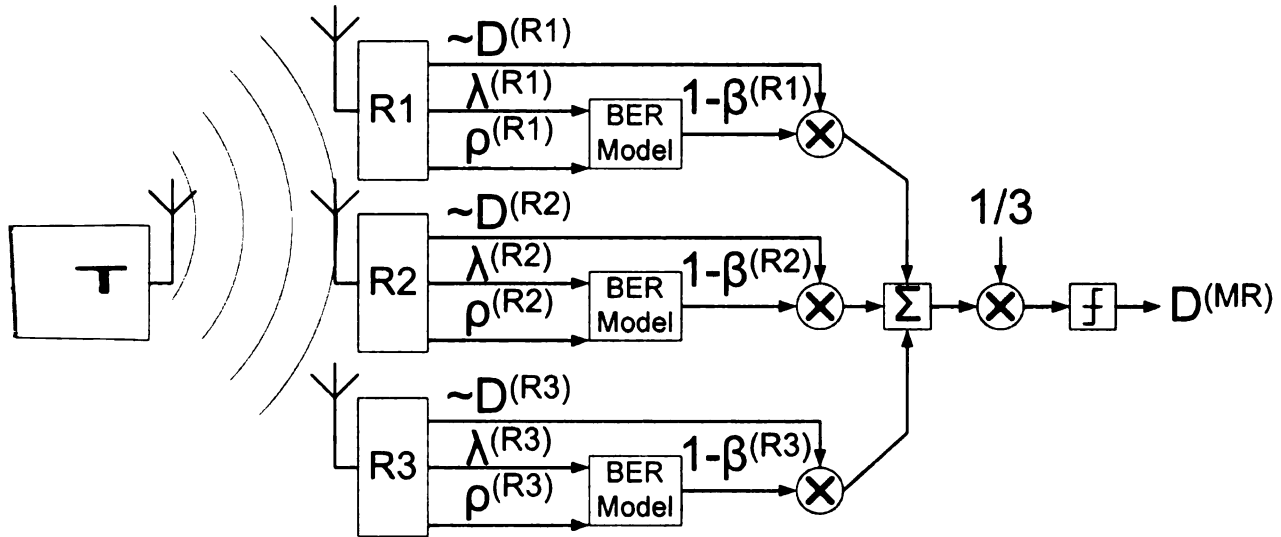
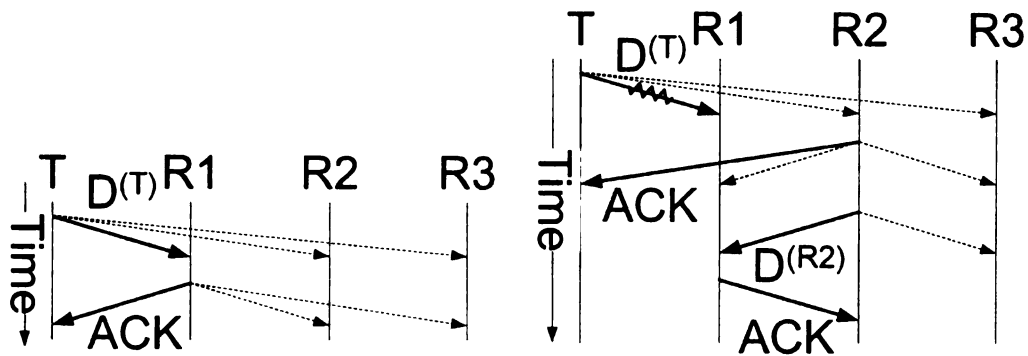


Figure 8.4: Illustration of logical functioning of maximal-ratio gain diversity.

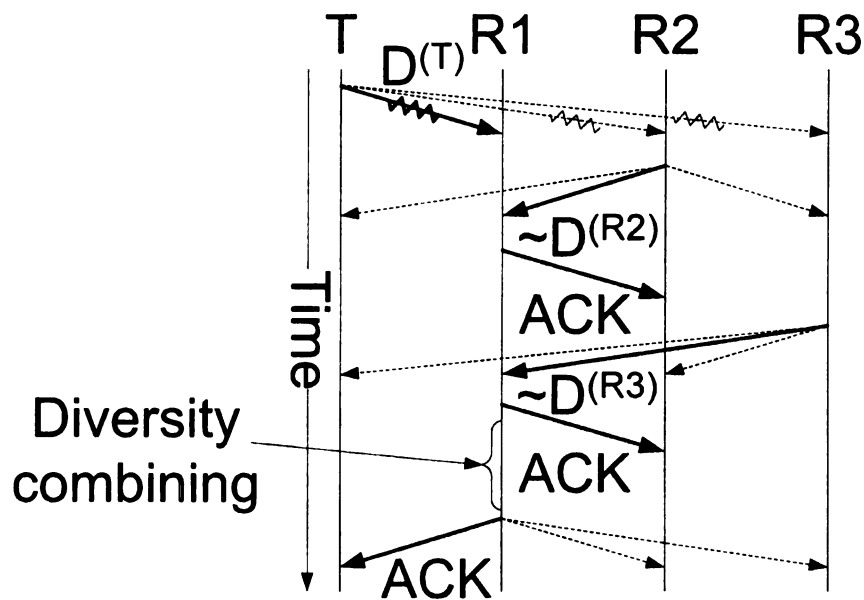
algorithm, a node $R1$ is chosen as an upstream end-point of a link. To use $R1$ as **part** of a set of multiple receivers we propose the gPMSS protocol that defines the **message** exchange between cooperating receiver nodes to handle transmissions that **are** received with errors or not received at all. The following subsection provides a **brief** overview of gPMSS protocol message exchanges for four important operations. **For** illustrative purposes we assume a scenario in which there is a distant transmitter T and a receiver $R1$ with two neighbor nodes $R2$ and $R3$ that are located close enough **to** communicate with $R1$ with few losses.

8.4.1 gPMSS Cluster Creation

The 'Poor Man's SIMO System' described by Ilyas, Kim and Radha in [52] differs **from** gPMSS in the way clusters of receivers are formed. Cluster creation in PMSS was **explicit**, and required the operation by which nodes $R2$ and $R3$ come to be associated **as** cooperating receiver nodes of a node $R1$. In gPMSS nodes take advantage of CSI **of** overheard messages. Figures 8.9 and 8.10 density functions of LQI and RSSI of packets originating from $R1$, $R2$ and $R3$. Nodes in a network with the gPMSS protocol



(a) Reception of an error free packet by a gPMSS cluster. (b) gPMSS message exchanges when parent receiver R1 receives message with errors but child R2 receives error-free.



(c) gPMSS message exchanges for recovery of data when neither parent R1 nor children R2 and R3 receive error-free.

Figure 8.5: gPMSS protocol operations.

will maintain such histograms for all neighbors from which they overhear traffic. A high mean, median or mode of LQI and RSSI density functions is indicative of a link with high PRR. In this way, once a node determines it enjoys good link conditions with a neighbor it will act as a member of that neighbor's cluster of receive nodes. However, unlike in PMSS this association will not be made explicit by an exchange of



messages. Instead, the node will make its contribution by participating in selection diversity and diversity combining functions of the neighbor node. For the following discussion we will assume that this way two nodes $R2$ and $R3$ placed close to $R1$ make the assessment that they enjoy a reliable wireless channel with $R1$ and volunteer to assist it as cooperating receivers.

8.4.2 Error-free Reception by at Least One Recipient

This section describes the exchange of messages under the gPMSS protocol that occurs when at least any one of the receiving nodes receives a transmitted packet without errors. Figure 8.5a depicts the simplest case. The solid lines represent the transmission and reception of a message between source and destination node. The dotted lines represent communication that occurs implicitly as a result of a receiver operating in promiscuous mode, (deliberately) eavesdropping on messages exchanged between other nodes (marked by solid lines). Here T sends a data message $D^{(T)}$ to $R1$ that is overheard by $R2$ and $R3$. $R1$ will promptly responds to T with an acknowledgement (ACK). $R2$ and $R3$ overhear the ACK and recognize that the packet was successfully received by $R1$ and no further action is required.

Figure 8.5b depicts the case where $R1$ is not the final destination. In addition, let us also assume that $R1$ receives the transmission $D^{(T)}$ with errors (marked by a zigzagged arrow), whereas $R2$ and $R3$ receive the same error-free. What happens is that all receivers that receive $D^{(T)}$ error-free choose a random wait-time t_1 from an exponential PDF limited to the range $[0, T_1]$. Let $t_1(R2)$ and $t_1(R3)$ denote $R2$ and $R3$'s random wait-times, respectively. Let $t_1(R2) < t_1(R3)$, then $R2$ will transmit ACK back to T before $R3$. $R3$ will overhear $R2$'s ACK and cancel transmission of its own ACK. At any time, if an ACK packet is lost and not received by T , T will retransmit $D^{(T)}$ (although it may already have been received and ACK'ed). This

way the power consumed in nodes forming the gPMSS cluster to relay packets will be more evenly distributed.

8.4.3 Erroneous Reception by All Recipients

This section describes the exchange of messages under the gPMSS protocol that occurs when all nodes that form a gPMSS cluster receive a transmission with errors. Figure 8.5c depicts this entire transaction. Here T sends a data message $D^{(T)}$ to $R1$ that is overheard by $R2$ and $R3$. Since all receivers $R1$, $R2$ and $R3$ receive with errors none of them is able to respond to T with an ACK within time T_1 . Let $\tilde{D}^{(R1)}$, $\tilde{D}^{(R2)}$ and $\tilde{D}^{(R3)}$ denote the different versions of $D^{(T)}$ as they are received by $R1$, $R2$ and $R3$, respectively. Thus, there is no error-free copy of the transmitted message at any receiver. Nodes $R1$, $R2$ and $R3$ all wait for one another to respond to T with an ACK. When none of the receivers $R1$, $R2$ and $R3$ overhear an ACK going back to T within the T_1 time of receiving, they infer that none of them received $D^{(T)}$ error-free. Instead of requesting a retransmission from T , $R1$ collects the error-prone versions of the D from cooperating receivers, acknowledging each one immediately as it receives them. $R2$ will transmit $\tilde{D}^{(R2)}, \lambda^{(R2)}, \rho^{(R2)}$, which denotes the concatenation of $\tilde{D}^{(R2)}$, the LQI $\lambda^{(R2)}$ and RSSI $\rho^{(R2)}$ with which it was received from T , to $R1$ between $[T_1, T_1 + T_2]$ after it received $\tilde{D}^{(R2)}$. Similarly, $R3$ will transmit $\tilde{D}^{(R3)}, \lambda^{(R3)}, \rho^{(R3)}$ between $[T_1, T_1 + T_2]$ after it received $\tilde{D}^{(R3)}$. Once $R1$ has received $\{\tilde{D}^{(R2)}, \lambda^{(R2)}, \rho^{(R2)}\}$ and $\{\tilde{D}^{(R3)}, \lambda^{(R3)}, \rho^{(R3)}\}$ it executes one of the diversity combining algorithms described in the preceding section in an attempt to recover $D^{(T)}$. If the CRC computed from the recovered packet matches the appended CRC the attempt is successful. On the receiver side T waits for an ACK, any ACK from any of the receivers $R1$, $R2$ or $R3$, for a timeout period of T_T until it attempts retransmission of $D^{(T)}$. Note that $T_T > T_1 + T_2$.

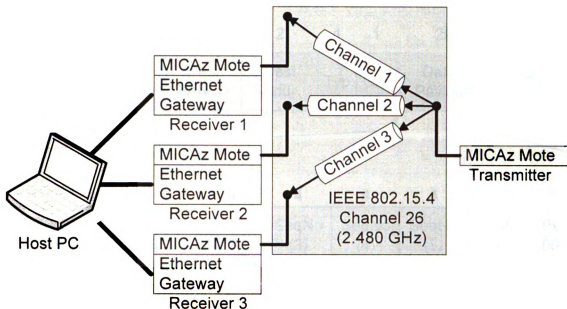


Figure 8.6: Equipment setup for trace collection.

8.5 Trace Based Proof of Concept

In this section we provide proof of concept of gPMSS by testing its performance on bit error traces. We collected error traces of a few million packets in a way that provides, to the authors' best knowledge, the BER a packet is subjected to and the LQI and RSSI with which it is received.

8.5.1 Experimental Setup

The trace-collection setup is depicted in figure 8.6 and consists of a Crossbow MPR2400 MICAz mote [34] transmitter and another three MICAz motes mounted on Crossbow MIB600 Ethernet gateways [33] as receivers. The three receivers R_1 , R_2 and R_3 are connected to a host personal computer (PC) running three instances of Xlisten (a data logging application), one for each receiver. This way a data collection session produces three traces. All traces were collected while operating in channel 26 in the 2.480GHz band. The reason for choosing channel 26 was that it is least prone

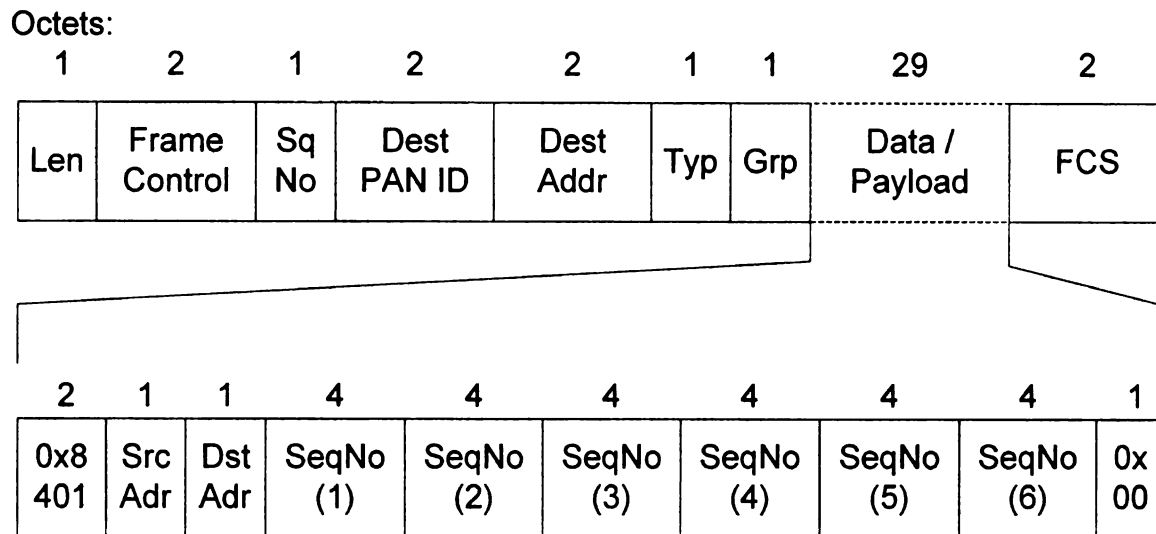


Figure 8.7: CC2420 MAC frame format used for experiments.

to interference from any 802.11b/g frequency channels. Our own experience shows that selecting channel 26 does not completely eliminate interference from co-located 802.11b/g WLANs, but reduces it significantly.

Packet Payload

TinyOS [75] is one of the most widely used open source operating system in WSN devices. TinyOS v1.1 allows various packet formats to be transmitted. We suitably modified code to enable the standard 802.15.4 frame format which TinyOS v1.1 labels CC2420 Frame Format (after the Chipcon CC2420 chipset [120] used in MICAz devices). Strictly speaking, the term packet refers to the protocol data unit (PDU) exchanged between network layers of the transmitter and receiver while the term frame is used for PDU's exchanged between MAC layers. However, since our analysis is restricted to the MAC layer there is little cause for confusion and we use these terms interchangeably to refer to MAC layer PDUs. The exact MAC frame format used is shown in figure 8.7. The size of the frame is 41 bytes and comprises of a 1 byte length field, 2 byte frame control field (FCF), 1 byte sequence number, 2 byte

destination PAN ID, 2 byte destination address, 1 byte type field, 1 byte group field, 29 bytes of data followed by a 2 byte frame check sequence (FCS) containing a CRC. The contents of the payload field are of our own choosing and consist of 3 unused bytes, the source address, the destination address and 6 copies of a 32 bit sequence number. The sequence number in the payload is used to keep track of lost packets. If the sequence number between two consecutively received packets skips one or more numbers that is indicative of a packet loss. The sequence number field alone proves too small for this task in the face of long fades. Note that transmitted packets differ only in the 1 byte sequence number in the header and the six 32 bit sequence numbers in the payload, and the CRC. For a particular trace all remaining bits remain unchanged. However, since the wireless channel will introduce bit errors the copies of the sequence number used to track packet losses in the received packet may differ. For this purpose we use a majority vote of the received sequence numbers to reconstruct the transmitted sequence number and from it the entire packet.

Trace Generation

Bit-level error traces can be generated by comparing a transmitted packet with its received version. A simple bit-wise XOR operation of the transmitted and received packets yields a bit pattern in which a zero ('0') signifies a bit that is received without error while a one ('1') represents an inverted bit. We observe that in some cases the length of the received packet is shorter than the transmitted packets. This constitutes a partial loss and we use the term partially lost packets to refer to such packets. Partially erased packets are logged when bits in the MAC header's length field are inverted and the receiver stops listening to the wireless channel prematurely. It has also been observed that if bits in the length field are inverted in such a way that the length of the incoming packet appears longer than actual the length of the logged packet still equals that of the transmission. Although the length field in the received

packet may falsely indicate a longer packet, the absence of a carrier signal allows the receiver to detect the end of transmission.

8.5.2 Channel State Information

Each received packet's logged entry is accompanied with three pieces of packet level CSI parameters. The first is the FCS status of the packet modeled by random variable Φ with the n^{th} packet's FCS status is represented by $\phi[n]$. Ordinarily receivers only distinguish between two states, i.e. *FCS Pass* (denoted $\phi = 0$) if the CRC value in the FCS field matches the CRC of the received packet, and *FCS Fail* (denoted $\phi \neq 0$) if it does not. Since we have knowledge of packet erasures and size of transmitted packets we extend the definition of FCS status to accommodate the reason for failure. We restrict the definition of *FCS Fail BE* (denoted $\phi = 1$) to mean that the size of a received packet matches the size of the transmitted packet and the CRC failure is due to bit errors (BE). Furthermore we classify a packet as being *FCS Fail PL* (denoted $\phi = 2$) and *FCS Fail CL* (denoted $\phi = 3$), where PL and CL are abbreviations for partial loss and complete loss respectively. Packets that are partially lost cannot pass the CRC test and are marked FCS Fail PL. Packets that are not received at all, i.e. when the decoded sequence number at receiver skips, are marked FCS Fail CL.

Among other CSI there are RSSI and LQI which we described in earlier sections. Completely lost packets, with $\phi = 3$, are assigned $\rho = -128$, $\lambda = 0$, and $\beta = 1$. Thus each received packet is characterized by its FCS Status, LQI, RSSI and BER processes.

8.5.3 Implementation Results

Due to shortage of space and due to the consistent similarities in results, we restrict our discussion to a subset of traces. This particular data set was collected in an

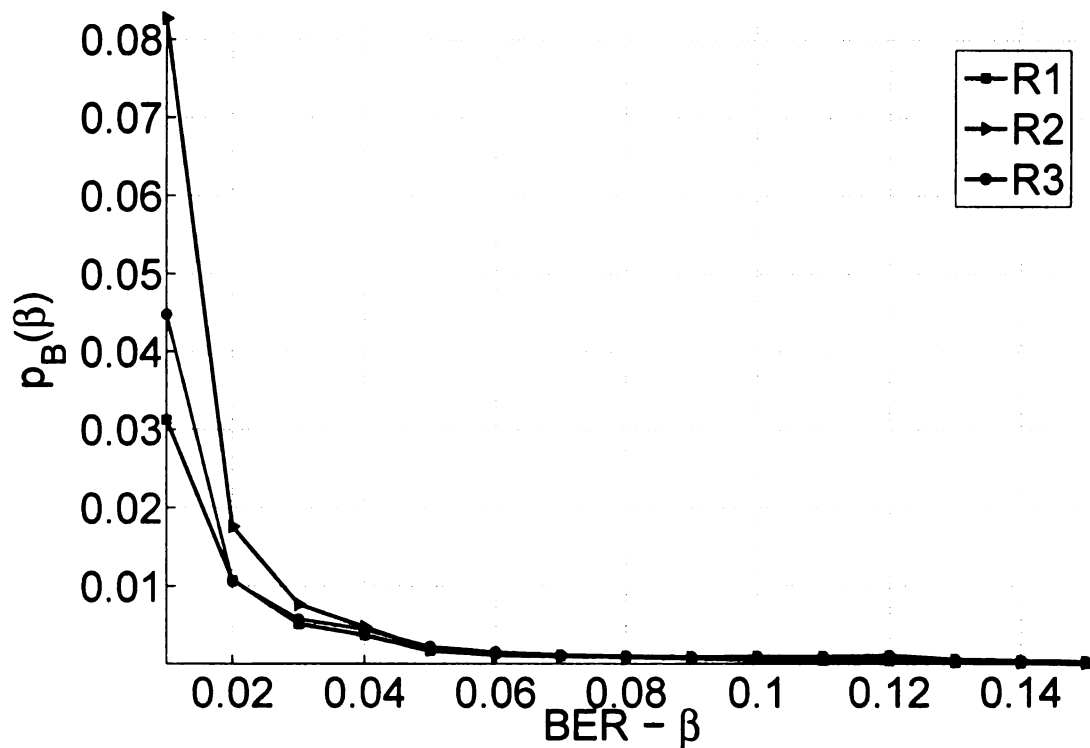


Figure 8.8: PDF of BER experienced by receivers $R1$, $R2$ and $R3$ ($p_B(\beta = 0)$ is cropped out for better view of non-zero range).

office environment over a period of 24 hours consisting of approximately 800,000 transmitted packets. The link between transmitter and receiver was non-line-of-sight, with a wall, a door and several furniture items in the direct line between them. The receivers were separated by a distance of $0.25m$. The gPMSS cluster consisted of three receivers, also Crossbow MICAz motes mounted on MIB600 Ethernet gateways. Figure 8.8 is a cropped portion of the PDF of BERs observed in packets at gPMSS receivers $R1$, $R2$ and $R3$ that excludes $\beta = 0$ for enhanced visibility. Figure 8.9 depicts the PDF of the LQI of all received packets at $R1$, $R2$ and $R3$. Figure 8.10 depicts the PDF of their RSSI. These three figures clearly show that all three receivers experience different channel conditions.

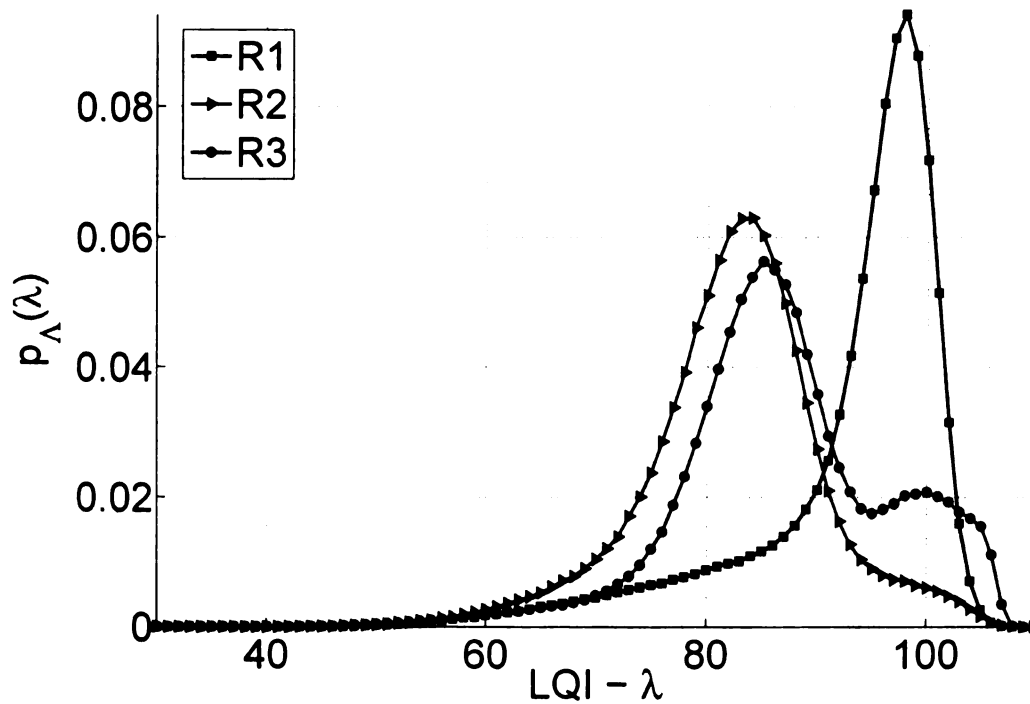


Figure 8.9: PDF of LQI experienced by receivers $R1$, $R2$ and $R3$.

PER and PLR Analysis

We define two quantities based on the FCS status, the packet error rate (PER) and the packet loss rate (PLR);

$$PER = \frac{\# \text{ of rcvd packets with } \phi = 1, 2}{\# \text{ of transmitted packets}}, \quad (8.7)$$

$$PLR = \frac{\# \text{ of rcvd packets with } \phi = 3}{\# \text{ of transmitted packets}}, \quad (8.8)$$

The packet reception rate (PRR) as $PRR = 1 - (PER + PLR)$. In figure 8.11 the first three entries on the horizontal axis plot the PER, PLR and the sum of the two, PER+PLR, for $R1$, $R2$ and $R3$. For individual receivers PER+PLR happens to be

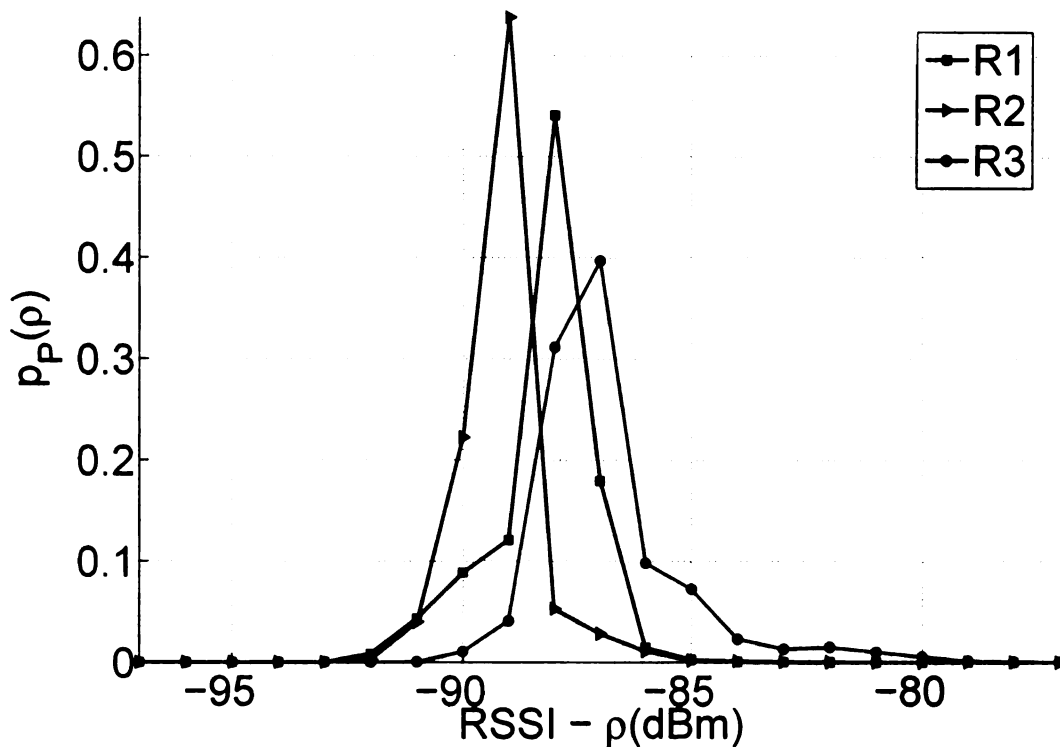


Figure 8.10: PDF of RSSI experienced by receivers $R1$, $R2$ and $R3$.

approximately 7%, 17% and 12%. These figures are followed by plots of these same quantities for the three diversity combining techniques. The simplest technique, selection diversity, appears to track the PER+PLR of the best performing receiver, in this case $R1$. Equal gain and maximal ratio diversity combining both perform better than any individual receiver and selection diversity. This was to be expected. Recall that selection diversity merely tries to pick out the least corrupted version among a set, whereas equal gain and maximal ratio actually attempt to correct errors in received messages by un-weighted and weighted voting, respectively. This is adequately reflected in the plot of PER, PLR and PER+PLRs. Both are able to reduce the PER

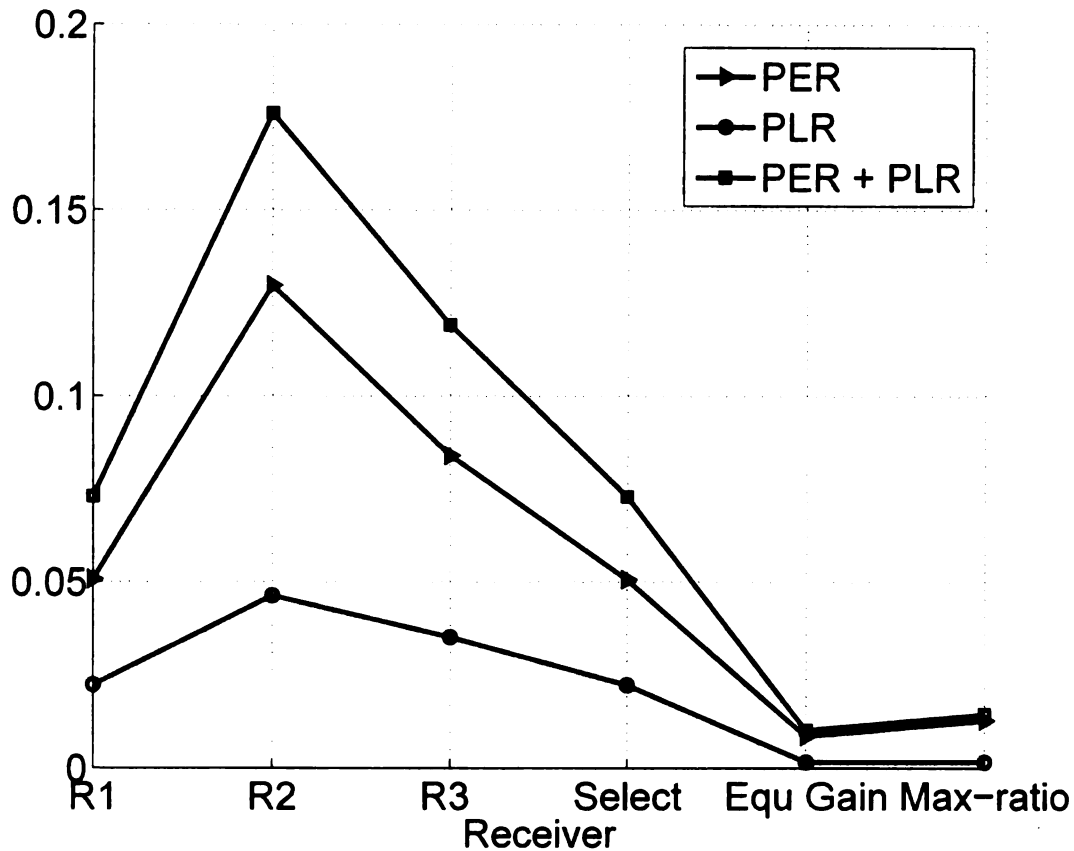


Figure 8.11: PER, PLR and PER+PLR experienced by receivers $R1$, $R2$ and $R3$ without gPMSS diversity combining and with selection, equal gain, and maximal ratio diversity combining.

BER Analysis

In figure 8.12 we plot the histogram (not PDF) of packets with non-zero BER as experienced by individual receivers $R1$, $R2$ and $R3$ without any diversity combining, as well as with different diversity combining methods. Again, the trends exhibited by diversity combining methods are the same across all traces. Figure 8.12 shows that the histogram of the selection diversity combining closely matches that of the best receiving individual receiver, i.e. $R1$. The close match of the histogram of selection diversity with that of $R1$ shows it manages to bring a gPMSS' BER performance up to that of the best receiving node. Thus, the BER model that is at the heart of this diversity combining technique delivers good performance. The result of equal

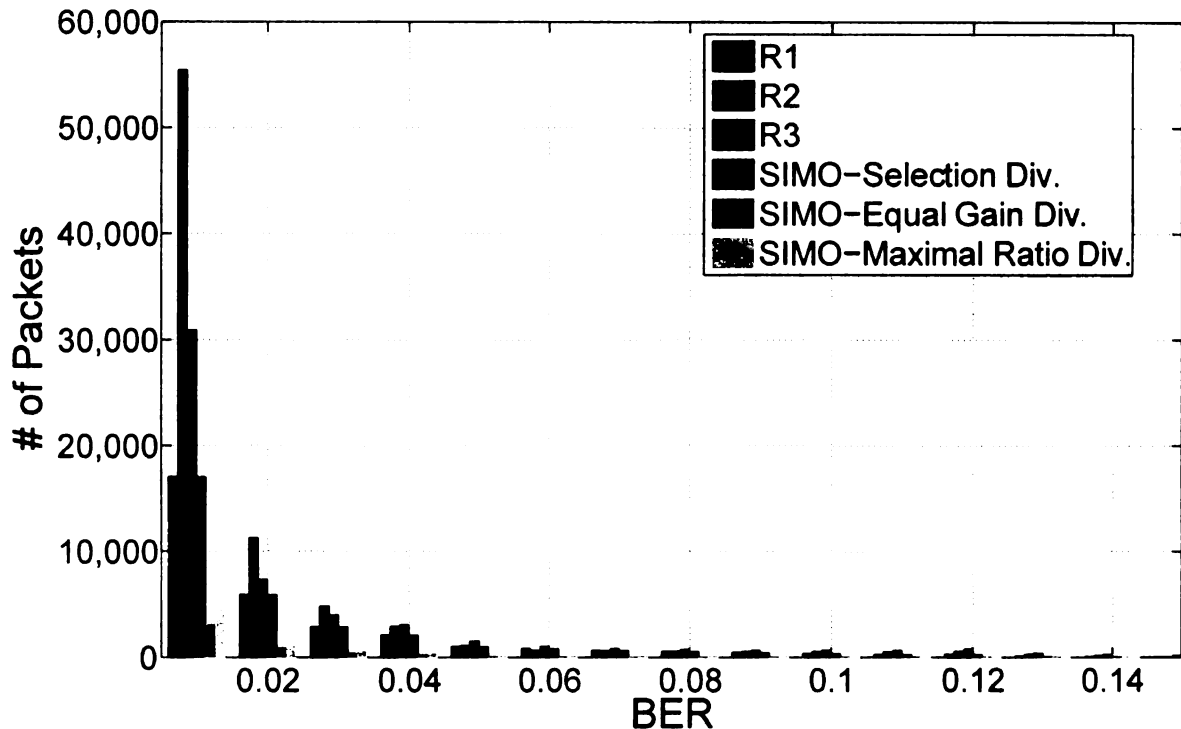


Figure 8.12: Histogram of BERs observed by receivers $R1$, $R2$ and $R3$ without gPMSS diversity combining and with selection, equal gain, and maximal ratio diversity combining.

gain and maximal gain diversity combining are even better. For every BER bin in the histogram, both equal gain and maximal-ratio combining are able to reduce the number of corrupt packets. Both are very close in their performance, but equal gain is consistently beating maximal-ratio combining across all BER bins in figure 8.12, and is also able to maintain this performance across different trace sets.

8.6 gPMSS Protocol Implementation

This section describes our implementation of the gPMSS protocol for motes and analyzes its performance. For the mote platform, we selected the Crossbow's Imote2 with the pre-installed .NET Micro Framework edition [35]. Using this edition of the Imote2 enabled us to implement gPMSS in the C# programming language which simplified

and accelerated development. At this point we would like to clarify that although the Imote2 used for the actual implementation in this section is different from the MICAz we used for trace collection in section 8.5, both use the same Chipcon CC2420 radio transceiver [120] which makes them equivalent for the purpose at hand. As the description of the gPMSS protocol above showed, in a situation when a transmission is received correctly by at least one recipient, gPMSS implements selection diversity described in subsection 8.3.1. But when a transmission is received with errors by all receivers, gPMSS either implements the functionality of an equal gain diversity or maximal ratio diversity combiner. We have implemented both in C# for Imote2.

The maximal ratio diversity combiner depends on the CSI-driven BER model by Ilyas and Radha [55]. Since the BER model takes as input an LQI, RSSI pair λ, ρ we still need to map it to a probability value. In the first instance we find the 90th percentile value of the BER's predicted PDF, i.e. the BER for which the value of the cumulative distribution function (CDF) is 0.9. In the second instance we map PDFs of the BER to their corresponding 50th percentile. We analyze the performance of the gPMSS protocol in a setting with one transmitter and $N = 3$ receivers. The receivers run a complete implementation of the gPMSS protocol described in section 8.4. For the experiment the timeout constants were set to $T_1 = 10 \text{ sec}$, $T_2 = 12 \text{ sec}$ and $T_T = 30 \text{ sec}$. We deliberately chose large values for T_1 , T_2 and T_T to avoid synchronization issues and justify them by the low-rate nature of target applications for IEEE 802.15.4. For the time being we have not attempted to optimize them to maximize throughput while still avoiding synchronization problems. The experiment was conducted at a residence with moderate Wi-Fi network interference.

8.7 Results and Analysis

This section analyzes and compares PRR, energy per packet and effect of retransmission limits on packet delivery rate with and without gPMSS.

8.7.1 Packet Reception Rate

We denote the total number of transmissions made from transmitter T by CT , and the number of retransmissions among them by CR . Similarly, the number of transmitted packets that are received at $R1$, $R2$ and $R3$ without errors are denoted by $C1$, $C2$ and $C3$, respectively. Finally, CS denotes the number of packets for which diversity combining was attempted and succeeded, and CF the number of packets for which it failed. All these values are tabulated in table 8.1. Each row in the table corresponds to a trial experiment using a variant of gPMSS specified in the first column. The results presented here are for three variants, i) Maximal-ratio combining using $\beta_{90\%}$ for the BER point estimate, ii) Maximal-ratio combining using the $\beta_{50\%}$ for the BER point estimate, and iii) equal gain combining. To make sense of the packet counts in table 8.1 and quantitatively assess the benefits of using only selection diversity, and using selection diversity in conjunction with maximal-ratio/ equal gain combining we look at PRRs, denoted by θ . Columns (1), (2) and (3) in table 8.2 contain the PRRs of the baseline configuration in which receivers $R1$, $R2$ and $R3$ do not cooperate. Column (4) contains the PRR when gPMSS is used with the diversity combination method in column (0). Some of the packets received using gPMSS will have been received as a result of selection diversity, while others will have been recovered as a result of diversity combining. The following columns separate the gain in PRR over that in the baseline configuration by providing the additive increase in PRRs of individual receivers. Columns (5), (6) and (7) are additive contributions of selection diversity in θ_{gPMSS} to the PRRs of individual receivers. Thus, $\Delta\theta_{SD,R1}$, $\Delta\theta_{SD,R2}$ and

Table 8.1: Packet counts.

Diversity combining	CT	CR	$C1$	$C2$	$C3$	CS	CF
Max-Ratio $\beta_{90\%}$	3170	855	937	893	957	597	0
Max-Ratio $\beta_{50\%}$	4167	1039	1254	1322	1198	739	0
Equal Gain	3683	879	819	844	825	497	0

$\Delta\theta_{SD,R3}$ are the increments in the PRR with respect to their respective baseline performances θ_{R1} , θ_{R2} and θ_{R3} in non-cooperating mode. Finally, column (8) is the additive contribution of diversity combining $\Delta\theta_{DC}$ to the PRR θ_{gPMSS} of the system with gPMSS. Thus, since the PRR gains in columns (5), (6), (7) and (8) are all additive the relationship between the terms in table 8.2 is,

$$\begin{aligned}
\theta_{gPMSS} &= \theta_{R1} + \Delta\theta_{SD,R1} + \Delta\theta_{DC} \\
&= \theta_{R2} + \Delta\theta_{SD,R2} + \Delta\theta_{DC} \\
&= \theta_{R3} + \Delta\theta_{SD,R3} + \Delta\theta_{DC}.
\end{aligned} \tag{8.9}$$

8.7.2 Energy Per Packet

In this section we compute separately the energy expended by the transmitter T as well as the receiver cluster $R1$, $R2$ and $R3$ per error free packet communicated to any one receiver. Let E_{DAT} denote the energy spent to transmit a data packet, E_{ACK} the energy spent to transmit an ACK packet and $E_{ACK} < E_{DAT}$. Then the energy E_T spent by the transmitter T during the course of an experiment is $C_T \times E_{DAT}$. The energy spent by receivers $R1$, $R2$ and $R3$ in acknowledging these are $E_{R1} = C1 \times E_{ACK}$, $E_{R2} = C2 \times E_{ACK}$ and $E_{R3} = C3 \times E_{ACK}$. Note that although energy is consumed by nodes in tasks other than radio transmissions,

Table 8.2: PRR of individual nodes without gPMSS, PRR with gPMSS protocol, PRR gain for individual receivers $R1$, $R2$ and $R3$ due to selection diversity, and the PRR gain due to diversity combining.

(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	θ_{R1}	θ_{R2}	θ_{R3}	θ_{gPMSS}	$\Delta\theta_{SD,R1}$	$\Delta\theta_{SD,R2}$	$\Delta\theta_{SD,R3}$	$\Delta\theta_{DC}$
Exp1: Max-Ratio $\beta_{90\%}$	0.29	0.28	0.30	0.73	0.25	0.26	0.24	0.19
Exp2: Max-Ratio $\beta_{50\%}$	0.30	0.31	0.29	0.75	0.27	0.26	0.28	0.18
Exp3: Equal Gain	0.22	0.23	0.22	0.76	0.40	0.39	0.40	0.13

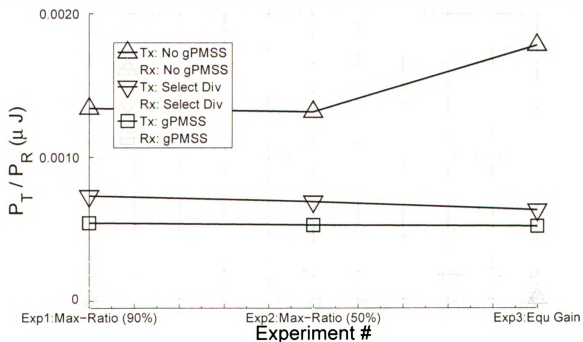


Figure 8.13: The energy in μJ consumed by transmitter and receivers per successfully delivered packet.

Table 8.3: Energy consumed by transmissions at transmitter and receiver side per error-free received packet. Columns (1) and (2) correspond to the baseline scenario using only retransmissions. Columns (3) and (4) correspond to the case when only selection diversity is used by receivers. Columns (5) and (6) corresponds to the case where a full implementation of gPMSS is used that employs diversity combining (equal gain or maximal-ratio) in addition with selection diversity.

(0) Div comb	(1) P_T	(2) P_R	(3) P_T	(4) P_R	(5) P_T	(6) P_R
Exp1: Max- Ratio $\beta_{90\%}$	$3.383E_{DAT}$	E_{ACK}	$1.845E_{DAT}$	E_{ACK}	$1.369E_{DAT}$	$1.516E_{ACK}$ + $0.516E_{DAT}$
Exp2: Max- Ratio $\beta_{50\%}$	$3.323E_{DAT}$	E_{ACK}	$1.744E_{DAT}$	E_{ACK}	$1.332E_{DAT}$	$1.473E_{ACK}$ + $0.473E_{DAT}$
Exp3: Equal Gain	$4.497E_{DAT}$	E_{ACK}	$1.596E_{DAT}$	E_{ACK}	$1.314E_{DAT}$	$1.355E_{ACK}$ + $0.355E_{DAT}$

the power consumed by computations is orders of magnitude less. Since the gPMSS protocol has computational complexity of $O(N)$. We compute the energy per packet consumed at the transmitter P_T and the sum of energy consumed by all receivers together P_R as,

$$P_T = \frac{E_T}{\# \text{ of packets recvd wo errors}} \quad (8.10)$$

$$P_R = \frac{E_R}{\# \text{ of packets recvd wo errors}} \quad (8.11)$$

Thus, P_T and P_R are energy consumption rates of transmitter and receivers obtained by normalizing by number of successfully delivered packets. Table 8.3 lists P_T , the per

decodable packet energy at the transmitter, and P_R , the per decodable packet energy at all receivers combined for all three experiments (listed in column (0)). Columns (1) and (2) in table 8.3 correspond to the baseline case when gPMSS is not used and packets received by $R1$ are retransmitted. Columns (3) and (4) correspond to the case when only selection diversity is used by cooperating receivers. Columns (5) and (6) corresponds to the case where a full implementation of gPMSS is used that employs diversity combination (equal gain or maximal-ratio) in addition to selection diversity. To keep the relationship general the tabulated values are in terms of E_{ACK} and E_{DAT} .

The Intel Imote2 consumes $792nJ/b$ when in transmit/receive mode when it operates at $13MHz$ [35]. Based on this figure E_{DAT} for a 41 byte frame is $260\mu J$ and E_{ACK} for a 5 byte acknowledgement frame is $32\mu J$. Figure 8.13 plots P_T and P_R (in Joules) expended in experiments 1, 2 and 3 when when maximal-ratio combining with $\beta_{90\%}$, maximal-ratio combining with $\beta_{50\%}$ and equal gain combining. As in table 8.3 we also evaluate energy for the cases if no gPMSS and if only selection diversity were used. Black lines correspond to energy consumption of the transmitter, while blue lines correspond to energy consumption of the receivers. Clearly, the transmitter power consumption rate P_T and receiver power consumption rates do not vary significantly across experiments and gPMSS variants. However, there is significant variation in P_T and P_R when gPMSS is not used versus selection diversity versus gPMSS. For all three experiments P_T is highest when gPMSS is not used while the corresponding receiver power consumption rate P_R is lowest. Opting to use selection diversity alone significantly reduces P_T for maximal ratio gain variants (Exp 1 and 2) by about 42% and about 64% for equal gain variant (Exp 3). P_R remains unchanged. Note from the previous section that this is accompanied by a 25% (for Exp 1 and 2) and 40% (for Exp 3) increase in PRR. Thus selection diversity is able to provide significant power savings while increasing PRR at the same time. When gPMSS is

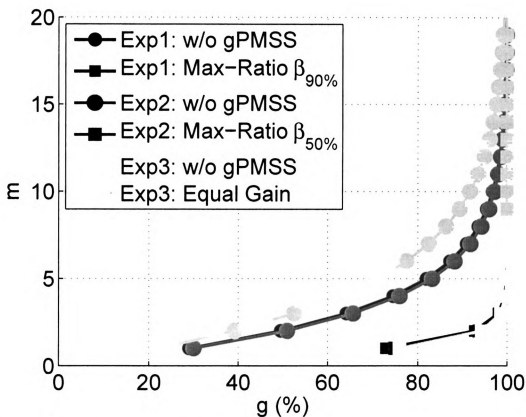


Figure 8.14: Maximum number of transmission attempts m versus delivery guarantee $g(\%)$.

employed P_T is reduced by about 58% (for Exp 1 and 2) and 68% (for Exp 3) over the baseline configuration not using gPMSS. However, this is accompanied by an increase of approximately the same amount of energy on the receiver side. Thus, it appears that gPMSS shifts some of the power consumption from the transmitter side to the receiver side.

8.7.3 Packet Transmission Attempts

The number of times the IEEE 802.15.4 MAC will retry transmitting a packet is controlled by the `maxMaxFrameRetries` attribute whose default value is set to 3 but can be varied from 0 – 7 (refer to IEEE 802.15.4 standard [5]). This limit on the

number of transmission attempts m for a packet limits the maximum PRR that can be guaranteed to g . Conversely, we may ask what is maximum number of transmission attempts m that the MAC must be allowed in order to ensure that at least $g\%$ of packets are received without errors? Figure 8.14 plots m against g for all three experiments. Clearly, to achieve any delivery guarantee $g\%$, fewer transmissions are required with gPMSS, regardless of whether maximal-ratio or equal gain diversity combination is used, compared to the case where gPMSS is not enabled. For example, figure 8.14 shows that to achieve a 95% delivery guarantee we have to allow 9, 9, 13 transmission attempts for the channel conditions observed in experiments 1, 2 and 3. Using gPMSS, however, the maximum number of transmission attempts required to achieve the same delivery guarantee $g = 95\%$ are 3, 3, and 3, respectively. Clearly, the values of m required to achieve $g = 95\%$ without gPMSS exceeds IEEE 802.15.4's capabilities. From the plot in figure 8.14 we see that at IEEE 802.15.4's default value of $m = 4$ the maximum achievable delivery guarantee for the three experiments lies in the range 65 – 75%.

8.8 Conclusions

We presented the gPMSS, a protocol-centric approach to enable receiver cooperation and diversity combining without requiring any changes to mote hardware or the IEEE 802.15.4 LR-WPAN standard. We described three principal mechanisms enabled by gPMSS, namely selection diversity, equal gain and maximal-ratio gain diversity combining. We provide proof-of-concept and demonstrate gPMSS' efficacy by applying these diversity combining techniques on bit error traces collected from a network of IEEE 802.15.4 motes. We demonstrate gPMSS by implementing it on the Intel Imote2 sensor mote running the .NET Micro framework. We analyze the performance of gPMSS in terms of PRR, retransmission attempts and power consumption per de-

livered packet. We saw that gPMSS raises the PRR from 22 – 30% to 73 – 76%, a relative increase of 150 – 245%. Since gPMSS is a protocol-based solution it implies a messaging overhead. We observe that power consumption by the transmitter per correctly delivered packet is reduced up to 68%. We evaluated the effect of retry limit imposed by the IEEE 802.15.4 standard of the on the packet delivery rate that can be achieved. At the default retry limit of 3, ($m = 4$) gPMSS can achieve delivery rates of greater than 99%, against only 65 – 75% when gPMSS is not used. Thus we demonstrate that gPMSS is capable of raising PRR, making use of highly lossy links feasible, thus reducing the number of required retransmission attempts and reducing the energy consumption rate of the transmitter per packet delivered. gPMSS has direct application in the design of small-world topologies in wireless networks to reduce the characteristic path length and diameter of networks which facilitates service discovery and the routing of high priority data in a network. This has the advantage of not needing any additional hardware([124] and [125]), or adding wired connections([27] and [111]). The extension of the effective communication range also has applications in extending the lifetime of nodes surrounding the base station in wireless sensor networks subject to the funneling effect. The larger communication range allows more nodes to communicate with the base station directly and reduces the traffic load from nodes positioned closer to the base station. More generally, gPMSS can be used to connect weakly connected components of a network by adding more links between nodes farther apart.

Chapter 9

Principal Component Centrality as a Measure of Node Centrality in Communication Networks

9.1 Introduction

Centrality [14], [15], [17], [42], [106] is a measure to assess the criticality of a node's position. Node centrality as a measure of a node's importance by virtue of its central location has been in common use by social scientists in the study of social networks for decades. Over the years several different meanings of centrality have emerged. Naturally, the idea of ranking nodes for their ability to spread or detect (positive or negative) influence is of significant interest to social network analysis.

Among many centrality measures, eigenvalue centrality (EVC) is arguably the most successful tool for detecting the most influential node(s) within a social graph. Thus, EVC has been a highly popular centrality measure in the social sciences ([45], [126], [14], [41], [38], [40], [127], [121], [16], [15]) (it is often referred to simply as centrality). As we demonstrate later in this chapter, one key shortcoming of EVC is its focus on (virtually) a single influential set of nodes that tend to cluster within a single neighborhood. In other words, EVC has the tendency of identifying a set of influential nodes that are all within the same region of a graph. This shortcoming may not represent a major issue for many social science problems and Internet applications, such as PageRank, where EVC has been used extensively [72]. Meanwhile, when dealing with massive networks/graphs, it is hardly the case that there is a single neighborhood of influential nodes; rather, there are usually multiple influential neighborhoods most of which are not detected or identified by EVC.

In order to identify influential neighborhoods, there is a need to associate such neighborhoods with some form of an objective measure of centrality that can be evaluated and searched for. To that end, one can think of a centrality plane that is overlaid over the underlying graph under consideration. This centrality plane may contain multiple centrality score maxima, each of which is centered on an influential neighborhood.

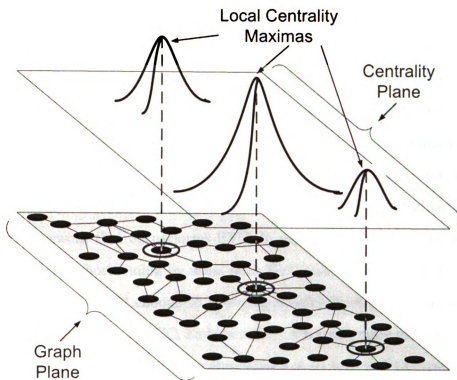


Figure 9.1: This figure shows a graph on the lower plane, overlaid with another plane of the interpolated surface plot of node centrality scores. The centrality planes typically exhibit a number of peaks or local maxima.

Nodes that have centrality score higher than other nodes are located under a centrality peak and are more central than any of their neighbors. We use the term hubs to refer to nodes forming centrality maxima. Figure 9.1 illustrates this concept. Thus, these hubs form the kernel of influential neighborhoods in networks. Hence, our focus in this research is on identifying influential neighborhoods rather than influential nodes. We will show that EVC has a tendency to be too narrowly focused on a dominating neighborhood. To this end, we introduce a new measure of centrality that we call principal component centrality (PCC) that gradually widens the focus of EVC in a controlled manner. More importantly, PCC provides a general framework for transforming graphs into a spectral space analogous to popular signal transforms that operate on random signals.

In this chapter, we give a brief review of common centrality measures accom-

panied by a critique of their application to wireless network topologies. We then introduce PCC, a node centrality measure that is inspired by the Karhunen Loève transform (KLT) and principal component analysis (PCA). In essence, PCC is a general transform of graphs that can provide vital insight into the centrality and related characteristics of such graphs. Similar to the KLT of a signal, the proposed PCC of a graph gives a form of compact representation that identifies *influential nodes* and more importantly *influential neighborhoods*. Hence, PCC provides an elegant *graph transform* framework that outperforms EVC. In particular, early in this chapter, we demonstrate EVCs shortcoming by using both EVC and PCC to compute node centralities in a network small enough to allow meaningful illustration. This is followed by a thorough description of PCC, and its utility in transforming massive real-world networks/graphs. We also develop the equivalence of an inverse PCC transform that attempts to reconstruct a representation of the original graph from its influential neighborhoods.

The rest of this chapter is organized as follows. Section 9.2 gives a background review of existing centrality measures for graphs, highlights problems in EVC and motivates our development of a new node centrality. Section 9.3 defines the PCC measure of centrality. Section 9.4 describes in detail the advantages, mathematical interpretation, visualization and the effect of varying number of features of PCC. Section 9.5 concludes the chapter.

9.2 Background

Let \mathbf{A} denote the adjacency matrix of a graph $G(V, E)$ consisting of the set of nodes $V = \{v_1, v_2, v_3, \dots, v_N\}$ of size N and set of undirected edges E . When a link is present between two nodes v_i and v_j both $A_{i,j}$ and $A_{j,i}$ are set equal to 1 and set to 0 otherwise. Let $\Gamma(v_i)$ denote the neighborhood of v_i , the set of nodes v_i is connected

to directly.

9.2.1 Degree Centrality

The degree centrality of a node in a graph is a measure of the relative importance to the graph's connectivity. The degree centrality of a node is defined as the number of edges incident on it. Nodes with more incident edges have higher degree centrality than nodes with fewer incident edges. If d_i denotes the degree of node v_i then its degree centrality is computed by:

$$C_D(v_i) = \frac{d_i}{N - 1} \quad (9.1)$$

Degree centrality is a measure of a node's rate of dissemination (of an infection) in the immediate short term. It has the advantage that its computation does not require nodes to exchange information. However, it has two significant disadvantages;

1. Without an exchange of centrality information with other nodes, it is not possible to interpret and evaluate an individual node's centrality relative to that of others.
2. Degree centrality does not take into account the centrality of its neighbors.

9.2.2 Closeness Centrality

The closeness centrality of a node is defined as the mean length of geodesic paths to all other nodes. Intuitively, nodes occupying a more central location within the graph are expected to have shorter paths. Closeness centrality is a measure of the rate at which a node can spread an infection to all reachable nodes. Closeness is a suitable measure of centrality when the flow of commodity in the network follows

geodesic paths. Closeness centrality is a good measure of the average detection time in a network with flows of non-replicating commodity following geodesic paths.

9.2.3 Betweenness Centrality

The betweenness centrality of a node is defined as the fraction of geodesic paths (shortest paths) out of all geodesic paths between all pairs of nodes passing through that node. Thus, nodes located on more geodesic paths have a higher betweenness centrality than nodes located on fewer geodesic paths. Intuitively, since the subproblem optimality principle holds for the shortest path problem, a node's location on a geodesic path implies close proximity to all other nodes on that path. A node's betweenness can be interpreted as a measure of disruption caused when the node is removed from the network. Like closeness, betweenness too assumes that the flow of commodity is along geodesics. Betweenness centrality is a good measure of the average probability of detection of flows in a network with non-replicating commodity following geodesic paths.

9.2.4 Eigenvector Centrality

Eigenvector centrality (EVC) is a relative score recursively defined as a function of the number and strength of connections to its neighbors and as well as those neighbors' centralities. Let $x(i)$ be the EVC score of a node v_i . Then,

$$\begin{aligned}
 x(i) &= \frac{1}{\lambda} \sum_{j \in \Gamma(v_i)} x(j) \\
 &= \frac{1}{\lambda} \sum_{j=1}^N A_{i,j} x(j)
 \end{aligned} \tag{9.2}$$

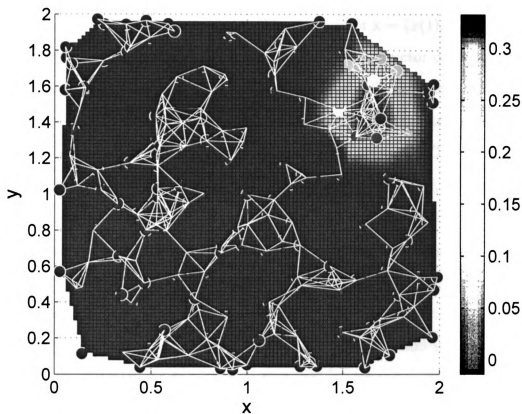


Figure 9.2: A spatial graph of 200 nodes. Node colors are indicative of the range in which their EVC falls.

Here λ is a constant. Equation 9.2 can be rewritten in vector form equation 9.3 where $\mathbf{x} = \{x(1), x(2), x(3), \dots, x(N)\}^T$ is the vector of EVC scores of all nodes.

$$\begin{aligned} \mathbf{x} &= \frac{1}{\lambda} \mathbf{A} \mathbf{x} \\ \lambda \mathbf{x} &= \mathbf{A} \mathbf{x} \end{aligned} \tag{9.3}$$

This is the well known eigenvector equation where this centrality takes its name from. λ is an eigenvalue and \mathbf{x} is the corresponding eigenvector of matrix \mathbf{A} . Obviously several eigenvalue/eigenvector pairs exist for an adjacency matrix \mathbf{A} . The EVC of nodes are defined on the basis of the Perron eigenvalue λ_A (the Perron eigenvalue

is the largest of all eigenvalues of \mathbf{A} and is also called the principal eigenvalue). If λ is any other eigenvalue of \mathbf{A} then $\lambda_A > |\lambda|$. The eigenvector $\mathbf{x} = \{x(1), x(2), \dots, x(N)\}'$ corresponding to the Perron eigenvalue is the Perron eigenvector or principal eigenvector. Thus the EVC of a node v_i is the corresponding element $x(i)$ of the Perron eigenvector \mathbf{x} . Note that when the adjacency matrix \mathbf{A} is symmetric all elements of the principal eigenvector \mathbf{x} are positive. As mentioned above, EVC is widely used in the social sciences ([82], [127], [14], [42], [40], [41], [128], [126], [16], [15]) and is often referred to simply as centrality.

EVC does not suffer from the same problems as degree, closeness and betweenness centralities. In computing a node's EVC it takes into consideration its neighbors's EVC scores. Because of its recursive definition, EVC is suited to measure nodes' power to influence other nodes in the network both directly and indirectly through its neighbors. Connections to neighbors that are in turn well connected themselves are rated higher than connections to neighbors that are weakly connected. Like closeness and betweenness, the EVC of a node provides a network-wide perspective. At the same time it can take advantage of distributed methods of computing eigenvectors/eigenvalues of a matrix but does not have to bear the overhead of excess network traffic. Sankaralingam [108], Kohlschütter [69] and Canright, Engø-Monsen and Jeliasity [20], Bischof [12], Bai [8] and Tisseur [121] proposed some parallel algorithms for computing eigenvectors and eigenvalues of adjacency matrices.

9.2.5 The Need for a New Centrality Measure

In the preceding sections we highlighted some of the key characteristics of the most common measures of centrality. Our discussion left us with only one viable measure of centrality that takes into consideration the centrality scores of a node's neighbors and which provides a network-wide perspective, i.e. EVC. EVC has been used extensively

to great effect in the study and analysis of a wide variety of networks that are shown to exhibit *small-world* and *scale-free* properties. In [21] Canright and Engo-Monsen correlated EVC with the instantaneous rate of spread of contagion on a Gnutella network peer-to-peer graph, a social network of students in Oslo, a collaboration graph of researchers at Telenor R&D and a snapshot of a collaboration graph of the Santa Fe Institute. In [90] Newman analyzed the use of EVC in a lexical network of co-occurring words in Reuters newswire stories. In [22] Carreras et al. used EVC to study the spread of epidemics in mobile networks. They used three sets of traces collected by Intel Cambridge, a trace of the public transportation network from the DieselNet project at the University of Massachusetts at Amherst and mobility and interaction traces from MIT's Reality Mining project.

Now consider the graph in figure 9.2. It consists of 200 nodes and is typical of networks of stationary WSN as well as mobile WSNs such as the cellphone based Nokia SensorPlanet project ([95], [62]). Its nodes are assigned one of six colors from the adjacent color palette. Each of the six colors represents one of six bins of a histogram spanning, in uniform step sizes, the range from the smallest to the largest EVCs. As the legend accompanying figure 9.2 shows, blue represents the lowest EVCs and red the highest. We make the following observations:

1. EVCs are tightly clustered around a very small region with respect to the total size of the network and drops off sharply as one moves away from the node of peak EVC.
2. EVC is unable to provide much centrality information for the vast majority of nodes in the network.
3. The position of the peak EVC node appears somewhat 'arbitrary' because a visual inspection shows that almost equally significant clusters of nodes can be

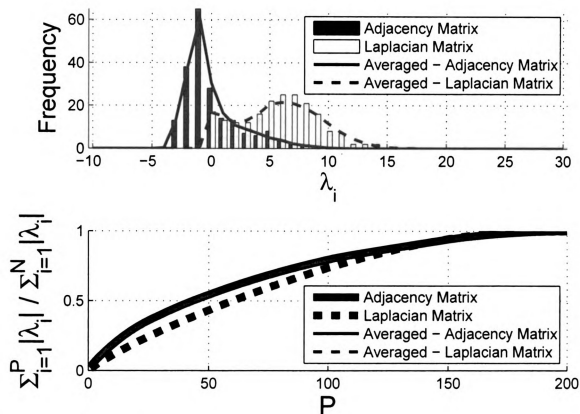


Figure 9.3: [Top] Histogram of eigenvalues of adjacency matrix and Laplacian matrix \mathbf{A} of network in figure 9.2; [Bottom] Cumulative sum of the sequence of eigenvalues of adjacency matrix and Laplacian matrix of network in figure 9.2 when sorted in descending order of magnitudes. In both figures the lines plotted in red color are averages of 50 networks generated randomly with the same parameters.

visually spotted in other locations in the graph. Counter to intuition, the high EVC cluster is connected to the rest of the network by a single link.

9.3 Principal Component Centrality

The EVC of a node is recursively defined as a measure of centrality that is proportional to the number of neighbors of a node and their respective EVCs. As we saw in section 9.2.4, the mathematical expression for the vector of node EVCs is equivalent to the principal eigenvector. Our motivation for PCC as a new measure of node centrality

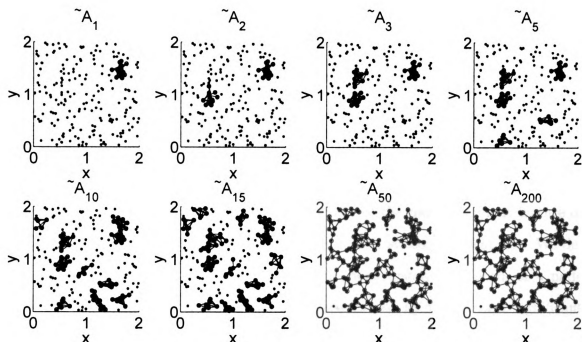


Figure 9.4: Reconstructed topologies of the graph from figure 9.2 using only the first 1, 2, 3, 5, 10, 15, 50 and all 200 eigenvectors.

may be understood by looking at EVC through the lens of the KLT. When the KLT is derived from an $N \times N$ covariance matrix of N random variables, the principal eigenvector is the most dominant feature vector, i.e. the direction in N -dimensional hyperspace along which the spread of data points is maximized. Similarly, the second eigenvector (corresponding to the second largest eigenvalue) is representative of the second most significant feature of the data set. It may also be thought of as the most significant feature after the data points are collapsed along the direction of the principal eigenvector. When the covariance matrix is computed empirically from a set of data points, the eigendecomposition is the well known PCA [38]. Since we are operating on the adjacency matrix derived from graph data we call the node centrality proposed in this research PCC. In a covariance matrix, a non-zero entry with a 'large' magnitude at positions (i, j) and (j, i) is representative of a strong relationship between the i -th and j -th random variables. A non-zero entry in the

adjacency matrix representing a link from one node to another is, in a broad sense, also an indication of a 'relationship' between the two nodes. Based on this understanding we draw an analogy between graph adjacency matrix and covariance matrix.

In the preceding section we described various centrality measures from literature. Among them, EVC is the node centrality most often used in the study of social networks and other networks with small-world properties. While EVC assigns centrality to nodes according to the strength of the most dominant feature of the data set, PCC takes into consideration additional, subsequent features. We define the PCC of a node in a graph as the Euclidean distance/ ℓ^2 norm of a node from the origin in the P -dimensional eigenspace formed by the P most significant eigenvectors. For a graph consisting of a single connected component, the N eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_N| = 0$ correspond to the normalized eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$. The eigenvector/eigenvalue pairs are indexed in order of descending magnitude of eigenvalues. When $P = 1$, PCC equals a scaled version of EVC. Unlike other measures of centrality, the parameter P in PCC can be used as a tuning parameter to adjust the number of eigenvectors included in the PCC. The question of selection of an appropriate value of P will be addressed in subsequent subsection 9.4.4. Let \mathbf{X} denote the $N \times N$ matrix of concatenated eigenvectors $\mathbf{X} = [\mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_N]$ and let $\Lambda = [\lambda_1 \lambda_2 \dots \lambda_N]'$ be the vector of eigenvalues. Furthermore, if $P < N$ and if matrix \mathbf{X} has dimensions $N \times N$, then $\mathbf{X}_{N \times P}$ will denote the submatrix of \mathbf{X} consisting of the first N rows and first P columns. Then PCC can be expressed in matrix form as:

$$\mathbf{C}_P = \sqrt{\left((\mathbf{A}\mathbf{X}_{N \times P}) \odot (\mathbf{A}\mathbf{X}_{N \times P}) \right) \mathbf{1}_{P \times 1}} \quad (9.4)$$

The ' \odot ' operator is the Hadamard (or entrywise product or Schur product) op-

erator. Equation 9.4 can also be written in terms of the eigenvalue and eigenvector matrices Λ and \mathbf{X} , of the adjacency matrix \mathbf{A} :

$$\mathbf{C}_P = \sqrt{(\mathbf{X}_{N \times P} \odot \mathbf{X}_{N \times P}) (\Lambda_{P \times 1} \odot \Lambda_{P \times 1})}. \quad (9.5)$$

It is important to note a major difference between a traditional "signal transform" under KLT as compared with the proposed PCC "graph transform". First, recall that, under KLT, a transform matrix \mathbf{T} is derived from a covariance matrix \mathbf{C} ; and then the eigenvector-based transform \mathbf{T} is applied on any realization of the random signal that has covariance \mathbf{C} . Meanwhile, under the proposed PCC, the adjacency matrix \mathbf{A} plays a dual role: at one hand, it plays the role of the covariance matrix of the KLT; and on the other hand, one can think of \mathbf{A} as being the "signal" that is represented compactly by the PCC vector \mathbf{C}_P . Effectively, the adjacency matrix \mathbf{A} represents the graph (i.e., "signal") that we are interested in analyzing; and at the same time \mathbf{A} is used to derive the eigendecomposition; and hence, we have the dual role for \mathbf{A} . Later, we will develop the equivalence of an inverse PCC, and we will see this dual role of the adjacency matrix \mathbf{A} again.

9.4 Evaluation

9.4.1 Interpretation of Eigenvalues

The definition of PCC is based on the graph adjacency matrix \mathbf{A} . For a matrix \mathbf{A} of size $N \times N$ its eigenvectors \mathbf{x}_i for $1 \leq i \leq N$ are interpreted as N -dimensional features (feature vectors) of the set of N -dimensional data points represented by their covariance (adjacency) matrix \mathbf{A} . The magnitude of an eigenvalue corresponding to an eigenvector provides a measure of the importance and prominence of the feature

represented by it. The eigenvalue λ_i is the power of the corresponding feature \mathbf{x}_i in \mathbf{A} .

An alternative representation of a graph's topology is the graph Laplacian matrix which is frequently used in spectral graph theory [28]. The graph Laplacian can be obtained from the adjacency matrix by setting the diagonal entries of the adjacency matrix to $A_{i,i} = -\sum_{j=1; i \neq j}^N A_{i,j}$, i.e. a diagonal entry in a Laplacian matrix is the negative of the sum of all off-diagonal entries in the same row in the adjacency matrix. This definition applies equally to weighted and unweighted graphs. The graph Laplacian is always positive-semidefinite which means all of its eigenvalues are non-negative with at least one eigenvalue equal to 0. The adjacency matrix, however, does not guarantee positive semidefiniteness and typically has several negative eigenvalues. This is the reason the ordering of features is based on magnitudes of eigenvalues. The bar chart at the top of figure 9.3 plots histograms of eigenvalues for both adjacency and Laplacian matrices of the network in figure 9.2. But why then, did we not use the Laplacian matrix in the first place? The reason is that the eigendecomposition of the adjacency matrix yields greater energy compaction than that of the Laplacian. The middle plot in figure 9.3 shows the normalized, cumulative function of the sorted sequence of eigenvalue powers. The line for the eigenvalue derived from the adjacency matrix rises faster than that of the Laplacian matrix. The adjacency matrix' curve indicates that 25%, 50% and 75% of total power is captured by the first 15 (7.5%), 44 (22%) and 89 (44.5%) features, respectively. In contrast, the Laplacian matrix' eigendecomposition shows that the same power levels are contained in its first 26 (13%), 61 (30.5%) and 103 (51.5%) features, respectively. Thus eigendecomposition of the adjacency matrix of graphs offers more energy compaction, i.e. a set of features of the adjacency matrix captures more energy than the same number of features of the corresponding Laplacian matrix.

C_{15} – 3D Spectral Drawing

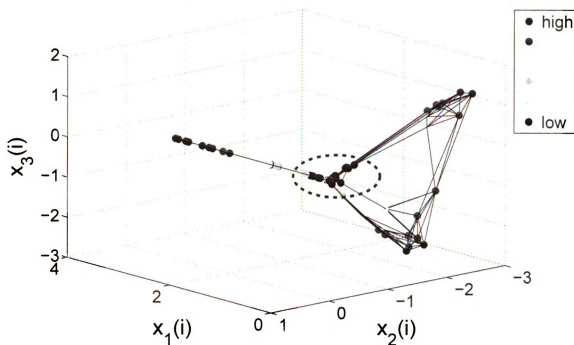


Figure 9.5: Spectral drawing of graph in three dimensions using entries of \mathbf{x}_1 , \mathbf{x}_2 , and \mathbf{x}_3 for the three coordinate axes. Nodes are colored according to their C_{15} PCC.

9.4.2 Interpretation of Eigenvectors

EVC interprets the elements of the Perron-eigenvector \mathbf{x}_1 of adjacency matrix \mathbf{A} as measures of corresponding nodes' centralities in the network topology (see section 9.2.4). Research on scale-free network topologies has demonstrated EVC's usefulness. However, when applied to large spatial graphs of uniformly, randomly deployed nodes such as the one in figure 9.2, EVC fails to assign significant scores to a large fraction of nodes. For a broader understanding that encompasses all eigenvectors we revert to the interpretation of eigenvectors as features. One way of understanding PCC is in terms of PCA [38], where PCC takes part of its name from. PCA finds the eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_N$ and eigenvalues of G 's adjacency matrix \mathbf{A} . Every eigenvector represents a feature of the adjacency matrix. To understand how these feature vectors are to be interpreted in graphical terms, refer to equation 9.6

which uses eigenvectors and eigenvalues to reconstruct an approximation $\tilde{\mathbf{A}}_P$ of the adjacency matrix \mathbf{A} . Reconstruction can be performed to varying degrees of accuracy depending on P , the number of features/ eigenvectors used. If we set $P = N$ in equation 9.6 (all eigenvectors/eigenvalues are used), the adjacency matrix can be reconstructed without losses (see He [45]). Here, $\hat{\Lambda}$ denotes the diagonal matrix of eigenvalues sorted in descending order of magnitude on the diagonal (from upper left corner to lower right corner).

$$\tilde{\mathbf{A}}_P = \mathbf{X}_{N \times P} \hat{\Lambda}_{P \times N} \mathbf{X}_{N \times N}^T \quad (9.6)$$

$$\text{where } \hat{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix}.$$

To illustrate, consider the unweighted, undirected graph $G(V, E)$ shown in figure 9.2 with adjacency matrix \mathbf{A} . \mathbf{A} 's entries are either 0 or 1. However, this is not necessarily true for $\tilde{\mathbf{A}}_P$, the version of the matrix reconstructed using the P most significant eigenvectors. The entries in $\tilde{\mathbf{A}}_P$ will very likely contain a lot of fractions. Therefore, before viewing the recovered topology in the reconstructed adjacency matrix $\tilde{\mathbf{A}}_P$ its entries have to be thresholded. Prior to plotting the topology, we rounded values less than 0.5 down to 0 and round values larger than or equal to 0.5 up to 1. Figure 9.4 plots the adjacency matrix reconstructed from the most significant 1, 2, 3, 5, 10, 15, 50 and all 200 feature vectors. The plot for $\tilde{\mathbf{A}}_1$ shows that the recovered topology information is highly localized to the vicinity of nodes with the highest EVC. The plot using $\tilde{\mathbf{A}}_2$ adds another highly connected but still very localized cluster to the network. Adding more feature vectors extends the set of connected nodes in

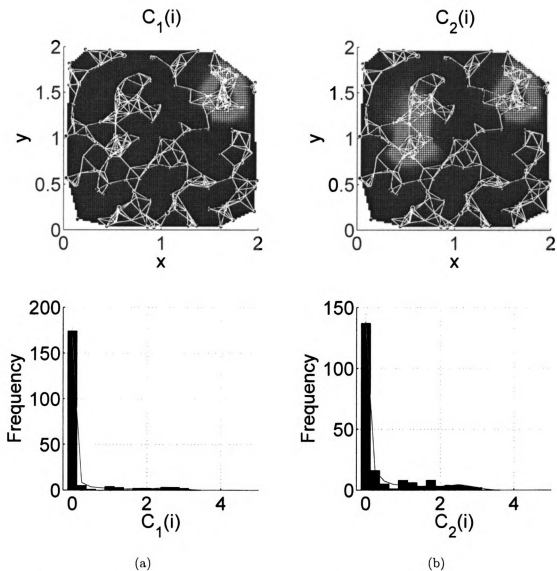


Figure 9.6: PCC of nodes in network of figure 9.2 when computed using first (a) 1 and (b) 2 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.

various parts of the network. As more eigenvectors are added to the computation of PCC it has the effect of increasing the resolution of centrality scores in nodes lying in less well connected regions of the network.

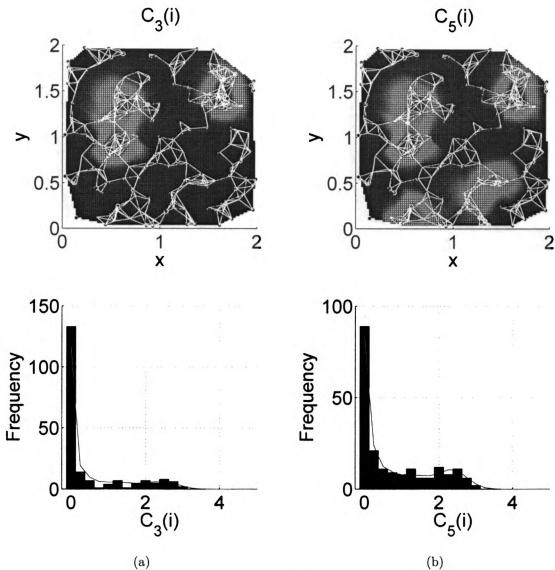


Figure 9.7: PCC of nodes in network of figure 9.2 when computed using first (a) 3 and (b) 5 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.

9.4.3 Graphical Interpretation of PCC

In this section we evaluate the usefulness of the PCC scores assigned to nodes of a network. Recall that a node's PCC is its ℓ^2 norm in P -dimensional eigenspace. Perceptual limitations restrict us from redrawing the graph in any eigenspace with

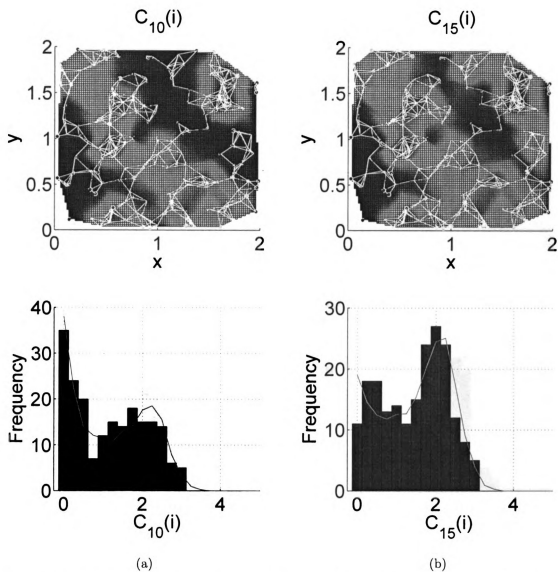


Figure 9.8: PCC of nodes in network of figure 9.2 when computed using first (a) 10 and (b) 15 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes. The lineplot in the histogram represents the average PCC histograms of 50 randomly generated networks with the same parameters as the network in figure 9.2.

more than 3 dimensions. Figure 9.5 is a drawing of the graph in figure 9.2 in the 3-dimensional eigenspace formed by the 3 most significant eigenvectors of the adjacency matrix \mathbf{A} . Nodes are colored according to their C_{15} PCC scores, derived from the 15 most significant eigenvectors, divided into 6 equally sized intervals between the lowest

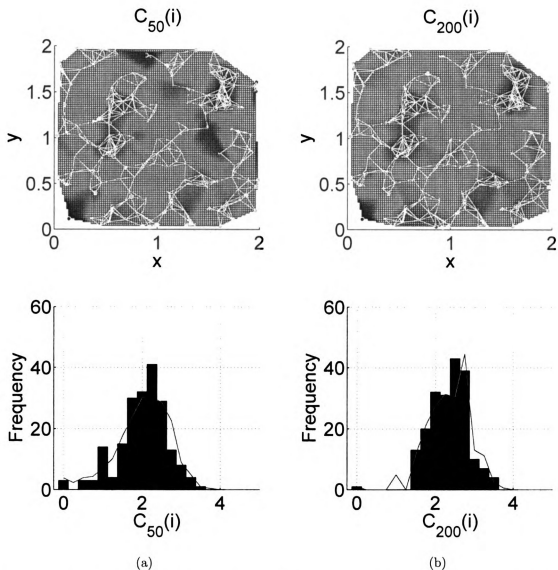


Figure 9.9: PCC of nodes in network of figure 9.2 when computed using first (a) 50 and (b) all 200 eigenvectors. The histograms accompanying each graph plot show the distribution of PCC of their nodes.

and highest PCC score. Based on the interpretation of PCC we expect nodes with higher (red) PCC scores to be located farther away from the origin at $(0, 0, 0)$ than nodes with lower (blue) PCC scores. From figure 9.5 we can see that this is clearly the case. For clarification, the cluster of low-PCC nodes around the origin $(0, 0, 0)$ is marked with a red, dashed oval.

9.4.4 Effect of Number of Features on PCC

In this section we study the effect varying the number of eigenvectors P has on PCC. For an illustrated example we revert to the randomly generated network topology of 200 nodes in figure 9.2. We compute PCC while varying P from 1 through 2, 3, 5, 10, 15, 50 and 200. Figures 9.6a, 9.6b, 9.7a, 9.7b, 9.8a, 9.8b, 9.9a and 9.9b re-plot the network with nodes colored to indicate their PCC scores. The bin size for all histograms is set to 0.25. Recall that since PCC score at $P = 1$ are a scaled versions of EVC, the figure 9.6a represents the baseline case of EVC. In figure 9.6a, EVC identifies a small cluster in the upper right corner as the nodes most central to the network. Note that ironically this cluster is separable from the larger graph by the removal of merely one link! On the other hand, clusters of nodes in the larger, better connected part of the graph are assigned EVC on the low end of the scale. As P is increased from figure 9.6b through 9.9b, more clusters of high PCC nodes pop up. As expected, the accompanying histograms below each graph plot show that this has the effect of increasing the variance of PCC scores.

Adding successively more features/eigenvectors will have the obvious effect of increasing the sum total of node PCC scores, i.e. $\mathbf{1}_{1 \times N} \mathbf{C}_m > \mathbf{1}_{1 \times N} \mathbf{C}_n$ when $m > n$. However, it is unclear how much PCC's scores change as P is varied from 1 through N . In [20] Canright *et al.* use the phase difference between eigenvectors computed in successive iterations as a stopping criteria for their fully distributed method for computing the principal eigenvector. We use the phase angle between PCC vectors and EVC to study the effect of adding more features. We compute the phase angle $\phi(n)$ of a PCC vector using n features with the EVC vector as,

$$\phi(P) = \arccos \left(\frac{\mathbf{C}_P}{|\mathbf{C}_P|} \cdot \frac{\mathbf{C}_E}{|\mathbf{C}_E|} \right). \quad (9.7)$$

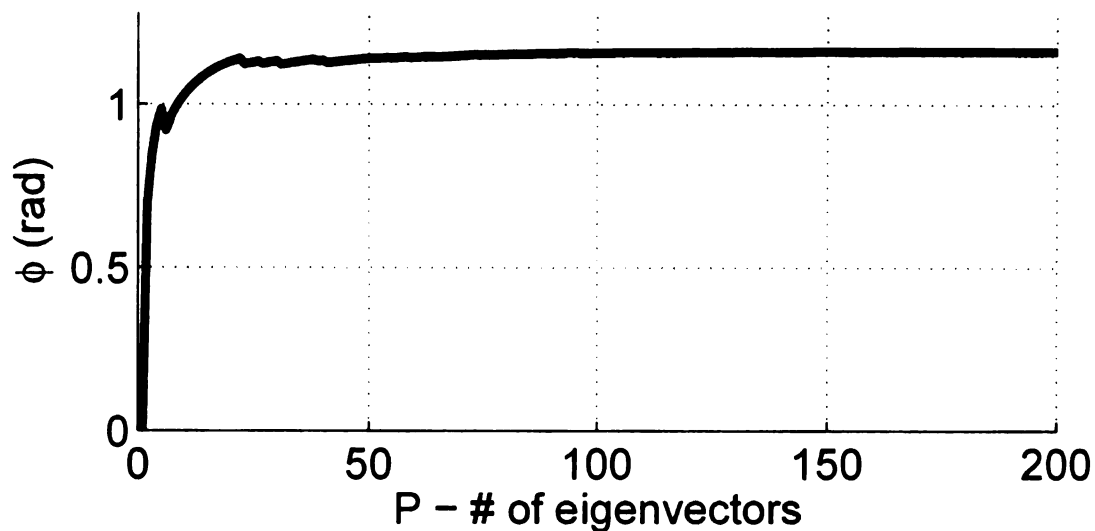


Figure 9.10: Plot of phase angles ϕ (in radians) of PCC vectors with the EVC vector for the graph in figures 9.6, 9.7, 9.8 and 9.9.

Here, ‘ \cdot ’ denotes the inner product operator. The relationship of the phase angle with the number of features used in PCC for the network under consideration is plotted in figure 9.10. Initially, the function of phase angle ϕ rises sharply and then levels off almost completely at 22 features. This means that, in this example, the relative PCCs of nodes cease to change with the addition of more features beyond the first 22 features. The phase angle plot may be used for determining how many features are sufficient for the computation of PCC of a network.

9.5 Conclusions

We reviewed previously defined measures of centrality and pointed out their shortcomings in general and EVC in particular. We introduced PCC, a new measure of node centrality. PCC is based on PCA and the KLT which takes the view of treating a graphs adjacency matrix as a covariance matrix. PCC interprets a node’s centrality as its ℓ^2 norm from the origin in the eigenspace formed by the P most significant feature vectors (eigenvectors) of the adjacency matrix. Unlike EVC, PCC allows the addition

of more features for the computation of node centralities. We explore two criteria for the selection of the number of features to use for PCC; a) The relative contribution of each feature's power (eigenvalue) to the total power of adjacency matrix and b) Incremental changes in the phase angle of the PCC with P features and the EVC as P is increased. We also provide a visual interpretation of significant eigenvectors of an adjacency matrix. The use of the adjacency matrix is compared with that of the Laplacian and it is shown that eigendecomposition of the adjacency matrix yields significantly higher degree of energy compaction than does the Laplacian at the same number of features. We also investigated the effect of adding successive eigenvectors and the information they contain by looking at reconstructions of the original graph's topology using a subset of features.

In the future we intend to extend the definition of PCC so it can be applied to both directed and undirected graphs. Furthermore, we propose to formulate a distributed method for computing PCC along the lines of Canright's method [20] for computing EVC in peer-to-peer systems.

Chapter 10

Conclusions

The following three sections list conclusions we learnt from the results of our research.

10.1 Channel Modeling

Our analysis and modeling of errors observed in IEEE 802.15.4 traces in chapters 3 and 4 leads us to make the following conclusions.

1. LQI and RSSI exhibit moderate negative correlation with the BER process (and strong positive correlation with each other).
2. LQI and RSSI can be used to reduce the variance of BER's estimated PDF of packets failing the CRC.
3. The CSI driven BER model remains valid across a variety of physical environments.
4. All IEEE 802.15.4 channels, regardless of channel selection or physical environment, exhibit a memory length of at most 2 bits and 2 symbols, respectively.
5. Based on the correlation function and analysis for LRD we conclude that various estimates of Hurst parameter may or may not detect packet level memory in 802.15.4 channels. The memory, however, is not due to the channel's inherent properties at those frequencies, but due to interference from IEEE 802.11b/g traffic and beacon frames, i.e. if interference is periodic the channel appears to have memory, if it is not periodic there is no memory.
6. The Abry-Veitch and Whittle estimators' consistent relative insensitivity to changes in average PER, average PLR, average CBER and interference across different traces leads us to conclude that they are better measures of the IEEE 802.15.4 channel's inherent degree of LRD.

7. The aggregate variance, R/S, periodogram, and absolute moment estimators' strong dependence on average PER and average PLR leads us to conclude that these estimators are good detectors of in-band (WLAN) interference.
8. The average CBER to which a packet is subjected by a channel is inversely related to the average PER/average PLR. Thus, it appears that interference produces higher BERs in packets than do channel fades.
9. We introduced RMI as a standardized version of Shannon mutual information and apply it to the BER process captured in bit traces. We observe that interference free IEEE 802.15.4 channels are memoryless, while channels experiencing significant interference from IEEE 802.11b/g networks sharing the $2.4GHz$ ISM band, a common source of interference, have true memory lengths varying in the narrow range of 0 to $2sec$.

10.2 Network Lifetime

Our analysis and modeling of the network lifetime problem in wireless sensor networks (WSN) in chapters 5 and 6 leads us to make the following conclusions.

1. We propose a new definition of network lifetime consisting of the tuple of mean and variance of node power consumption rates in a WSN. This interpretation of network lifetime is more inclusive and considers the power consumption of sensors across the network.
2. We formulated the optimization problem for the new objective function in the form of a budget constrained QP and showed that a solution exists. The solution of the QP, however, has a high complexity.

3. As an alternative to the QP, we developed a greedy dynamic program formulation that chooses routes in a way that optimizes for our objective from sets of paths that would be considered sub-optimal in the shortest path sense. Four variants are developed based on the BED, BND, ED and ND algorithms for the discovery of alternative routes.
4. We also observe that the routes generated by the ND algorithm are very similar to those proposed in previously proposed load balancing techniques such as Baek and de Veciana's in [6]. Under a many-to-one traffic flow the same ND paths, when used in conjunction with DPA) yield the worst performance out of the four route discovery algorithms.
5. A statistical performance comparison of these four route discovery algorithms for networks of 100 nodes and $\lambda = 0$ shows that on average the BND and BED in conjunction with the DPA yield the best performance. BND and BED yield reductions of up to 28% and 36% in variance of power consumption rates at the cost of raising average node power consumption by 15% and 21%, respectively.
6. The computational complexity of variants of the DPA vary from $O(N^3)$ to $O(N^4)$ which is significantly lower than the full search of the solution space which is of complexity $O(N!^N)$. However, for randomly generated networks of 100 nodes we consistently observed that the time to run the DPA on a PC is of the order of a few seconds.
7. Analysis by means of diffusion plots verified that DPA reduced power consumption of sensors that experience highest power consumption under shortest path routing algorithms. Diffusion plots also show that the reduction power consumption is highest under BND, followed closely by BED.
8. The resulting route selection method is one that is suitable for applications

with many-to-one traffic flows. Route discovery algorithms and DPA assume availability of global network topology information which is usually available at the base station. While this makes the DPA a centralized solution we envision it finding applications in critical infrastructure protection/control/monitoring, surveillance, and environmental/agricultural monitoring applications with infrequent topology changes.

10.3 WSN Topology

Our analysis small-world properties of networks with range limited shortcuts, the application of cooperative communication and diversity combining concepts and centrality measures for WSNs in chapters 7, 8 and 9 leads us to make the following conclusions.

1. From the analytical model of characteristic path length and clustering coefficient we observed that for sufficiently dense networks characteristic path length can be reduced significantly by replacing a $\mu \approx O(0.005 - 0.05)$ fraction of the local scale nodes by global scale nodes providing shortcuts in the network.
2. The order of μ , the fraction of nodes that are designated shortcut nodes, is about the same as the value of β , the rewiring probability, in Watts' small-world network construction method.
3. The model lends itself for the task of designing WSNs, e.g. determining the number of shortcut nodes required to achieve a certain characteristic path length.
4. We demonstrate gPMSS by application to IEEE 802.15.4 SIMO channel traces and implementation on the Crossbow Imote2 sensor mote. We analyzed the performance of gPMSS in terms of PRR, retransmission attempts and power

consumption per delivered packet. We saw that for a setup with 3 receivers gPMSS raises the PRR from 22 – 30% to 73 – 76%, a relative increase of 150 – 245%.

5. We observe gPMSS reduces transmission power per correctly delivered packet by up to 68%.
6. We evaluated the effect of retry limit imposed by the IEEE 802.15.4 standard of the on the packet delivery rate that can be achieved. At the default retry limit of 3, ($m = 4$) gPMSS can achieve delivery rates of greater than 99%, against only 65 – 75% when gPMSS is not used.
7. By making use of lossy links feasible, gPMSS can be used as a mechanism for adding shortcut links to enable small-world network topologies in WSNs.
8. Shortcuts implemented by gPMSS will reduce characteristic path length and diameter of networks which facilitates service discovery and the routing of high priority data in a network. It has the advantage of not requiring any hardware modifications ([124] and [125]), or adding wired connections([27] and [111]).
9. gPMSS used at the base station can increase communication range and the number of sensors at 1 hop distance, thereby increasing the number of critical sensors. This extends the lifetime of nodes surrounding the base station in WSNs subject to the funneling effect. The larger communication range allows more nodes to communicate with the base station directly and reduces the traffic load from nodes positioned closer to the base station. More generally, gPMSS can be used to connect weakly connected components of a network by adding more links between nodes farther apart.
10. We conducted a review of pre-existing measures of node centrality and their shortcomings with regard to WSNs. We introduced PCC, a new measure of

node centrality. A node's PCC can be interpreted as its ℓ^2 norm from the origin in the eigenspace formed by the P most significant feature vectors (eigenvectors) of the adjacency matrix.

11. While we see PCC as having wider applications, an immediate application of direct relevance is its use in identifying nodes for the placement of shared network resources, e.g. endpoints of shortcut links created through gPMSS.
12. To select an appropriate number of features for the computation of PCC we explored two methods.
 - The relative contribution of each additional feature's power (eigenvalue) to the total power of adjacency matrix. The use of the adjacency matrix is compared with that of the Laplacian. We concluded that eigendecomposition of the adjacency matrix yields significantly higher degree of energy compaction than the Laplacian at the same number of features. However, the cumulative scree plot of neither matrix yields a clear cutoff region for the selection of number of features P to use for PCC.
 - Incremental changes in the phase angle of C_P (PCC with P features) and C_1 (EVC) as P is increased. We concluded that PCC reaches steady state values with the use of only 5 to 10% most significant features out of all available feature vectors. Selecting number of features P using the phase angle function $\phi(P)$ yields a very clear cutoff region.

BIBLIOGRAPHY

- [1] L. Adamic. The small world web. *Lecture Notes in Computer Science*, pages 443–452, 1999.
- [2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393–422, 2002.
- [3] L.A.N. Amaral, A. Scala, M. Barthelemy, and H.E. Stanley. Classes of small-world networks. *Proceedings of the National Academy of Sciences*, page 200327197, 2000.
- [4] ANSI/IEEE. Ansi/ieee std 802, part 15.1: Wireless personal area network. Technical report, IEEE, 2002.
- [5] ANSI/IEEE. Ansi/ieee std 802, part 15.4: Low rate wireless personal area networks. Technical report, ANSI/IEEE, 2006.
- [6] S.J. Baek and G. de Veciana. Spatial energy balancing in large-scale wireless multihop networks. In *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, 2005.
- [7] P. Bahl, A. Adya, J. Padhye, and A. Walman. Reconsidering wireless systems with multiple radios. *ACM SIGCOMM Computer Communication Review*, 34(5):39–46, 2004.
- [8] Z. Bai, J. Demmel, J. Dongarra, A. Petitet, H. Robinson, and K. Stanley. The spectral decomposition of nonsymmetric matrices on distributed memory parallel computers. *SIAM Journal on Scientific Computing*, 18:1446–1461, 1997.
- [9] J. Barros and S.D. Servetto. On the capacity of the reachback channel in wireless sensor networks. In *IEEE Workshop on Multimedia Signal Processing*, pages 408–411, 2002.
- [10] J. Beran. *Statistics for Long-Memory Processes*. Chapman & Hall/CRC, 1994.

- [11] M. Bhardwaj and A.P. Chandrakasan. Bounding the lifetime of sensor networks via optimal role assignment. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2002.
- [12] C. Bischof and Institute for Defense Analyses. *Parallel Performance of a Symmetric Eigensolver Based on the Invariant Subspace Decomposition Approach*. Supercomputing Research Center: IDA, 1994.
- [13] B. Bollobás. *Random Graphs*. Cambridge University Press, 2001.
- [14] P. Bonacich. Power and centrality: A family of measures. *American Journal of Sociology*, 92:1170–1182, 1987.
- [15] P. Bonacich and P. Lloyd. Eigenvector-like measures of centrality for asymmetric relations. *Social Networks*, 23(3):191–201, 2001.
- [16] S.P. Borgatti. Centrality and network flow. *Social Networks*, 27(1):55–71, 2005.
- [17] S.P. Borgatti and M.G. Everett. A graph-theoretic perspective on centrality. *Social Networks*, 28(4):466–484, 2006.
- [18] S.P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- [19] D.G. Brennan. Linear diversity combining techniques. *Proceedings of the IEEE*, 91(2):331–356, 2003.
- [20] G. Canright, K. Engø-Monsen, and M. Jelasity. Efficient and robust fully distributed power method with an application to link analysis. *Department of Computer Science, University of Bologna, Tech. Rep. UBLCS-2005-17*, pages 2005–17, 2005.
- [21] G.S. Canright and K. Engø-Monsen. Spreading on networks: A topographic view. *ComplexUs*, 3(1-3):131–146, 2006.
- [22] I. Carreras, D. Miorandi, G.S. Canright, and K. Engø-Monsen. Understanding the spread of epidemics in highly partitioned mobile networks. In *Proceedings of the 1st international conference on Bio inspired models of network, information and computing systems*. ACM New York, NY, USA, 2006.
- [23] D. Cavalcanti, D. Agrawal, J. Kelner, and D. Sadok. Exploiting the small-world effect to increase connectivity in wireless ad hoc networks. In *IEEE International Conference on Telecommunications*, 2004.
- [24] S.S. Chakraborty, E. Yli-Juuti, and M. Liinajarja. An arq scheme with packet combining. *IEEE Communications Letters*, 2(7):200–202, 1998.

- [25] J.H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom)*, 2000.
- [26] S. Cheng and M.C. Valenti. Macrodiversity packet combining for the ieee 802.11 a uplink. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'05)*, volume 1, 2005.
- [27] R. Chitradurga and A. Helmy. Analysis of wired short cuts in wireless sensor networks. In *IEEE/ACS International Conference on Pervasive Services (ICPS'04)*, pages 167–177, July 2004.
- [28] Fan R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92) (Cbms Regional Conference Series in Mathematics)*. American Mathematical Society, February 1997.
- [29] R.K. Colwell and D.J. Futuyma. On the measurement of niche breadth and overlap. *Ecology*, 52(4):567–576, 1971.
- [30] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to algorithms 2nd ed.* MIT Press, Cambridge, MA, 2001.
- [31] R.A. Costa and J. Barros. Dual radio networks: Capacity and connectivity. In *Intl Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops (WiOpt'07)*, pages 1–6, 2007.
- [32] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley-Interscience New York, 2006.
- [33] Crossbow. *MIB600CA, Ethernet Interface Board Datasheet*. Crossbow Technology Inc., <http://www.xbow.com/Products/productdetails.aspx?sid=179>, 6020-0055-04 rev a edition, Oct 2006.
- [34] Crossbow. *MPR2400, ZigBee, 2.4 GHz, MICAz Mote Datasheet*. Crossbow Technology Inc., <http://www.xbow.com/Products/productdetails.aspx?sid=164>, 6020-0060-04 rev a edition, Oct 2006.
- [35] Crossbow Technology. *Crossbow Imote2 Datasheet*, 6020-0117-02 rev a edition, April 2007.
- [36] T. Dimitar, F. Sonja, M. Jani, and G. Aksenti. Small world application layer for ad hoc networks. In *Proceedings of Telekomunikacioni Forum Telfor*, 2003.
- [37] S. Dixit, E. Yanmaz, and O.K. Tonguz. On the design of self-organized cellular wireless networks. *IEEE Communications Magazine*, 43(7):86–93, 2005.

- [38] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley New York, 2nd edition, 2001.
- [39] H. Everett. Generalized lagrange multiplier method for solving problems of optimum allocation of resources. *Operations Research*, 11:399–417, 1963.
- [40] M.G. Everett and S.P. Borgatti. *Extending centrality*. Cambridge University Press, 2005.
- [41] L.C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.
- [42] N.E. Friedkin. Theoretical foundations for centrality measures. *American Journal of Sociology*, 96(6):1478, 1991.
- [43] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.
- [44] J.A. Hartigan and M.A. Wong. A k-means clustering algorithm. *JR Stat. Soc. Ser. C-Appl. Stat*, 28:100–108, 1979.
- [45] H. He. Eigenvectors and reconstruction. *the electronic journal of combinatorics*, 14(14):1, 2007.
- [46] A. Helmy. Small large-scale wireless networks: Mobility-assisted resource discovery. *Arxiv preprint cs/0207069*, 2002.
- [47] A. Helmy. Small worlds in wireless networks. *IEEE Communications Letters*, 7(10):490–492, 2003.
- [48] I. Howitt and J.A. Gutierrez. Ieee 802.15. 4 low rate-wireless personal area network coexistence issues. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'03)*, volume 3, 2003.
- [49] J.P. Hubaux, T. Gross, J.Y. Le Boudec, and M. Vetterli. Toward self-organized mobile ad hoc networks: the terminodesproject. *IEEE Communications Magazine*, 39(1):118–124, 2001.
- [50] R.F. i Cancho and R.V. Sole. The small world of human language. *Proceedings of the Royal Society B: Biological Sciences*, 268(1482):2261–2265, 2001.
- [51] IEEE. Ieee 802.11b-1999, part 11: Wireless lan: Higher speed physical layer (phy) extension in the 2.4 ghz band. Technical report, IEEE, 2003.
- [52] M. U. Ilyas, M. Kim, and H. Radha. Reducing packet losses in networks of commodity ieee 802.15.4 sensor motes using cooperative communication and diversity combination. In *Proceedings of the 28th IEEE International Conference on Computer Communications (INFOCOM'09)*, Rio de Janeiro, Brazil, April 2009.

- [53] M. U. Ilyas and H. Radha. Increasing network lifetime of an iee 802.15. 4 wireless sensor network by energy efficient routing. In *Proceedings of IEEE International Conference on Communications*, 2006.
- [54] M. U. Ilyas and H. Radha. Long range dependence of iee 802.15.4 wireless channels. In *Proceedings of the IEEE International Conference on Communications (ICC'08)*, May 2008.
- [55] M. U. Ilyas and H. Radha. Measurement based analysis and modeling of the error process in iee 802.15.4 lr-wpans. In *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM'08)*. IEEE, IEEE, April 2008.
- [56] M. U. Ilyas and H. Radha. Measuring packet-level memory length of 802.15.4 wireless channels with relative mutual information. In *Proceedings of the IEEE International Conference on Communications (ICC'08)*, May 2008.
- [57] M. U. Ilyas and H. Radha. On measuring memory length of the error rate process in wireless channels. In *Proceedings of the 42nd Annual Conf. on Information Sciences and Systems (CISS'08)*, March 2008.
- [58] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *ACM/IEEE International Conference on Mobile Computing and Networking*, pages 56–67. Boston, USA, 2000.
- [59] S.S. Iyengar and R.R. Brooks. *Distributed sensor networks*. CRC Press, 2004.
- [60] Z. Ji, Y. Yang, J. Zhou, M. Takai, and R. Bagrodia. Exploiting medium access diversity in rate adaptive wireless lans. In *Intl conference on Mobile Computing and Networking*, pages 345–359. ACM New York, NY, USA, 2004.
- [61] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 353(153-181):152, 1996.
- [62] A. Kansal, M. Goraczko, and F. Zhao. Building a sensor network of mobile phones. In *Proceedings of the 6th international conference on Information processing in sensor networks*, page 548. ACM, 2007.
- [63] T. Karagiannis, M. Faloutsos, and R. H. Riedi. Long-range dependence: now you see it, now you don't! In *Global Telecommunications Conference (GLOBECOM'02)*, volume 3, 2002.
- [64] S. Karande, S.A. Khayam, Y. Cho, K. Misra, H. Radha, J. Kim, and J.W. Hong. On channel state inference and prediction using observable variables in 802.11 b network. In *Proceedings of the IEEE International Conference on Communications (ICC'07)*, pages 4554–4559, 2007.

- [65] R. Khanna, H. Liu, and H.-H. Chen. Self-organization of sensor networks using genetic algorithms. In *Proceedings of IEEE International Conference on Communications*, 2006.
- [66] S. A. Khayam and H. Radha. Linear-complexity models for wireless mac-to-mac channels. *Wireless Networks*, 11(5):543–555, 2005.
- [67] S. A. Khayam and H. Radha. Constant-complexity models for wireless channels. In *Proceedings of the 25th Annual Joint Conf. on the IEEE Computer and Communications Societies (INFOCOM06)*, 2006.
- [68] S.A. Khayam, S. Karande, H. Radha, and D. Loguinov. Performance analysis and modeling of errors and losses over 802.11 b lans for high-bit-rate real-time multimedia. *Signal Processing: Image Communication*, 18(7):575–595, 2003.
- [69] C. Kohlschütter, P. Chirita, and W. Nejdl. Efficient parallel computation of pagerank. *Lecture Notes in Computer Science*, 3936:241, 2006.
- [70] A. Konrad, B.Y. Zhao, A.D. Joseph, and R. Ludwig. A markov-based channel model algorithm for wireless networks. *Wireless Networks*, 9(3):189–199, 2003.
- [71] L. Krishnamachari, D. Estrin, and S. Wicker. The impact of data aggregation in wireless sensor networks. In *Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops*, pages 575–578, 2002.
- [72] A.N. Langville, C.D. Meyer, and P. FernÁndez. Googles pagerank and beyond: the science of search engine rankings. *The Mathematical Intelligencer*, 30(1):68–69, 2008.
- [73] V. Latora and M. Marchiori. Efficient behavior of small-world networks. *Physical Review Letters*, 87(19):198701–1–198701–4, 2001.
- [74] V. Latora and M. Marchiori. Is the boston subway a small-world network? *Physica A: Statistical Mechanics and its Applications*, 314(1-4):109–113, 2002.
- [75] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, and E. Brewer. *TinyOS: An Operating System for Sensor Networks*, volume -. Springer Verlag, 2005.
- [76] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.
- [77] S. Lin, J. Zhang, G. Zhou, L. Gu, J.A. Stankovic, and T. He. Atpc: adaptive transmission power control for wireless sensor networks. In *Proceedings of the 4th International Conf. on Embedded Networked Sensor Systems*, pages 223–236. ACM Press New York, NY, USA, 2006.

- [78] R. Madan and S. Lall. Distributed algorithms for maximum lifetime routing in wireless sensor networks. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, volume 2, 2004.
- [79] S. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong. Tag: a tiny aggregation service for ad-hoc sensor networks. In *Proceedings of the ACM Symposium on Operating System Design and Implementation (OSDI'02)*, 2002.
- [80] D. Marco, E.J. Duarte-Melo, M. Liu, and D.L. Neuhoff. On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data. In *Proceedings of the 2nd International Workshop Information Processing in Sensor Networks (IPSN'03), Palo Alto, CA, USA, April 22-23, 2003*. Springer, 2003.
- [81] D. Marco and D.L. Neuhoff. Reliability vs. efficiency in distributed source coding for field-gathering sensor networks. In *Proceedings of the 3rd International symposium on Information processing in sensor networks (IPSN'04)*, pages 161–168. ACM Press New York, NY, USA, 2004.
- [82] S. Milgram. The small world problem. *Psychology Today*, 2(1):60–67, 1967.
- [83] A. Miu, J.G. Apostolopoulos, W. Tan, and M. Trott. Low-latency wireless video over 802.11 networks using path diversity. In *Proceedings of the IEEE International Conference on Multimedia & Expo*, volume 3, pages 441–444, July 2003.
- [84] A. Miu, H. Balakrishnan, and C.E. Koksal. Improving loss resilience with multi-radio diversity in wireless networks. In *Proceedings of the The 11th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 16–30. ACM, Aug-Sep 2005.
- [85] A. Miu, G. Tan, H. Balakrishnan, and J. Apostolopoulos. Divert: fine-grained path selection for wireless lans. In *Proceedings of the 2nd International conference on Mobile systems, applications, and services*, pages 203–216. ACM New York, NY, USA, 2004.
- [86] A.F. Molisch, K. Balakrishnan, C.C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, and K. Siwiak. Ieee 802.15. 4a channel model-final report. *IEEE P*, 15:802–1504, 2006.
- [87] J.M. Montoya and R.V. SolÉ. Small world patterns in food webs. *Journal of Theoretical Biology*, 214(3):405–412, 2002.
- [88] C. Moore and M.E.J. Newman. Epidemics and percolation in small-world networks. *Physical Review E*, 61(5):5678–5682, 2000.
- [89] C. Moore and M.E.J. Newman. Exact solution of site and bond percolation on small-world networks. *Physical Review E*, 62(5):7059–7064, 2000.

- [90] M.E.J. Newman. Analysis of weighted networks. *Physical Review E*, 70(5):56131, 2004.
- [91] M.E.J. Newman, C. Moore, and D.J. Watts. Mean-field solution of the small-world network model. *Physical Review Letters*, 84(14):3201–3204, 2000.
- [92] M.E.J. Newman and D.J. Watts. Scaling and percolation in the small-world network model. *Physical Review*, 60(6):7332–7342, 1999.
- [93] G. T. Nguyen, B. Noble, R. H. Katz, and M. Satyanarayanan. A trace-based approach for modeling wireless channel behavior. In *Proceedings of the Winter Simulation Conf. 1996*, pages 597–604, 1996.
- [94] A. Ortega and K. Ramchandran. From rate-distortion theory to commercial image and video compression technology. *IEEE Signal Processing Magazine*, 15(6):20–22, 1998.
- [95] B.P. Pál and P. Scientist. Sensorplanet: An open global research framework for mobile device centric wireless sensor networks. In *MobiSensors: NSF Workshop on Data Management for Mobile Sensor Networks, Pittsburgh, 2007*.
- [96] S. A. Pandit and R. E. Amritkar. Characterization and control of small-world networks. *Physical Review*, 60(2):1119–1122, 1999.
- [97] S. Patten, B. Krishnamachari, and R. Govindan. The impact of spatial correlation on routing with compression in wireless sensor networks. In *Proceedings of the 3rd International symposium on Information processing in sensor networks (IPSN'04)*, pages 28–35. ACM Press New York, NY, USA, 2004.
- [98] M. Penrose. *Random Geometric Graphs*. Oxford University Press, 2003.
- [99] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers. In *Proceedings of the Conf. on Communications architectures, protocols and applications (SIGCOMM'94)*, pages 234–244. ACM Press New York, NY, USA, 1994.
- [100] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, volume 2, pages 90–100. Feb, 1999.
- [101] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN'05)*, pages 364–369, 2005.
- [102] S. Pollin, M. Ergen, M. Timmers, A. Dejonghe, L. Van der Perre, I. Moerman, F. Catthoor, and A. Bahai. Distributed cognitive coexistence of 802.15. 4 with 802.11. In *Proceedings of the 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, pages 1–5, 2006.

- [103] H. Radha, M. Vetterli, and R. Leonardi. Image compression using binary space partitioning trees. *IEEE Transactions on Image Processing*, 5:1610–1624, 1996.
- [104] T. S. Rappaport. *Wireless Communications Principles and Practice*. Prentice Hall PTR Upper Saddle River, NJ, 2002.
- [105] C. Reis, Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Measurement-based models of delivery and interference in static wireless networks. *ACM SIGCOMM Computer Communication Review*, 36(4):51–62, 2006.
- [106] B. Ruhnau. Eigenvector-centrality a node-centrality? *Social Networks*, 22(4):357–365, 2000.
- [107] A. Sankar and Z. Liu. Maximum lifetime routing in wireless ad-hoc networks. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom)*, volume 2, 2004.
- [108] K. Sankaralingam, S. Sethumadhavan, and J.C. Browne. Distributed pagerank for p2p systems. In *High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on*, pages 58–68, 2003.
- [109] G.M. Schuster and A.K. Katsaggelos. *Rate-Distortion Based Video Compression: Optimal Video Frame Compression and Object Boundary Encoding*. Kluwer Academic Publishers, 1997.
- [110] P. Sen, S. Dasgupta, A. Chatterjee, P.A. Sreeram, G. Mukherjee, and S.S. Manna. Small-world properties of the indian railway network. *Physical Review E*, 67(3):36106, 2003.
- [111] G. Sharma and R. Mazumdar. Hybrid sensor networks: a small world. In *ACM Intl Symposium on Mobile Ad Hoc Networking and Computing*, pages 366–377. ACM Press New York, NY, USA, 2005.
- [112] A. Sikora and V.F. Groza. Coexistence of ieee802. 15.4 with other systems in the 2.4 ghz-ism-band. In *Proceedings of IEEE Instrumentation and Measurement Technology Conference (IMTC05)*, volume 3, pages 1786–1791, 2005.
- [113] S. Singh, M. Woo, and C.S. Raghavendra. Power-aware routing in mobile ad hoc networks. In *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pages 181–190, 1998.
- [114] D. Slepian and J. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19(4):471–480, 1973.
- [115] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. Some implications of low power wireless to ip networking. In *Proceedings of the 5th Workshop on Hot Topics in Networks (HotNets-V)*, 2006.

- [116] K. Srinivasan and P. Levis. Rssi is under appreciated. In *Proceedings of the 3rd Workshop on Embedded Networked Sensors (EmNets'06)*, 2006.
- [117] W. Stallings. *Wireless Communications & Networks*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 2004.
- [118] G.J. Sullivan and T. Wiegand. Rate-distortion optimization for video compression. *IEEE Signal Processing Magazine*, 15(6):74–90, 1998.
- [119] A.S. Tanenbaum. *Computer Networks*. Prentice Hall PTR, 2002.
- [120] TI and Chipcon. *Chipcon AS SmartRF CC2420 Preliminary Datasheet (rev 1.2), 2004-06-09*. TI Chipcon, 1.2 edition, June 2004.
- [121] F. Tisseur and J. Dongarra. A parallel divide and conquer algorithm for the symmetric eigenvalue problem on distributed memory architectures. *SIAM Journal on Scientific Computing*, 20:2223, 1999.
- [122] H.L. Van Trees. *Detection, Estimation, and Modulation Theory Part I*. MIT Press, Cambridge, MA, 1968.
- [123] M.C. Valenti. Improving uplink performance by macrodiversity combining packets from adjacent access points. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'03)*, volume 1, 2003.
- [124] C.Y. Wan, A.T. Campbell, and J. Crowcroft. A case for all-wireless, dual-radio virtual sinks. In *Intl Conf on Embedded Networked Sensor Systems*, pages 267–268. ACM Press New York, NY, USA, July 2004.
- [125] C.Y. Wan, S.B. Eisenman, A.T. Campbell, and J. Crowcroft. Siphon: overload traffic management using multi-radio virtual sinks in sensor networks. In *Intl Conf on Embedded Networked Sensor Systems*, pages 116–129. ACM Press New York, NY, USA, 2005.
- [126] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*. Cambridge University Press, 1994.
- [127] D.J. Watts. Networks, dynamics, and the small-world phenomenon. *American Journal of Sociology*, 105(2):493–527, 1999.
- [128] D.J. Watts. *Small Worlds: The Dynamics of Networks between Order and Randomness*. Princeton University Press, 1999.
- [129] D.J. Watts and S.H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.
- [130] P. Whoriskey. Instant-messagers really are about six degrees from kevin bacon. *Washington Post*, page A01, August 2 2008.

- [131] G.P. Williams. *Chaos Theory Tamed*. Joseph Henry Press, 1997.
- [132] C. Won, J.H. Youn, H. Ali, H. Sharif, and J. Deogun. Adaptive radio channel allocation for supporting coexistence of 802.15. 4 and 802.11 b. In *Proceedings of the Vehicular Technology Conference (Fall)*, 2005.
- [133] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the 1st International Conf. on Embedded Networked Sensor Systems*, pages 14–27. ACM Press New York, NY, USA, 2003.
- [134] G.R. Woo, P. Kheradpour, D. Shen, and D. Katabi. Beyond the bits: cooperative packet recovery using physical layer information. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking (MobiCom)*, pages 147–158. ACM Press New York, NY, USA, September 2007.
- [135] L. Xiao, M. Johansson, and S.P. Boyd. Simultaneous routing and resource allocation via dual decomposition. *IEEE Transactions on Communications*, 52(7):1136–1144, 2004.
- [136] D.G. Yoon, S.Y. Shin, W.H. Kwon, and H.S. Park. Packet error rate analysis of ieee 802.11 b under ieee 802.15.4 interference. In *Proceedings of the 63rd IEEE Vehicular Technology Conference (VTC'06-Spring)*, volume 3, 2006.
- [137] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *Proceedings of the 1st Annual IEEE Communications Society Conf. on Sensor and Ad Hoc Communications and Networks (SECON'04)*., volume 2004, pages 517–526, 2004.