Energy-Efficient and Lifetime Aware Routing in WSNs

by

Wilawan Rukpakavong

A Doctoral Thesis

Submitted in partial fulfilment
of the requirements for the award of

Doctor of Philosophy
of
Loughborough University

10th January 2014

Abstract

Network lifetime is an important performance metric in Wireless Sensor Networks (WSNs). Transmission Power Control (TPC) is a well-established method to minimise energy consumption in transmission in order to extend node lifetime and, consequently, lead to solutions that help extend network lifetime. The accurate lifetime estimation of sensor nodes is useful for routing to make more energy-efficient decisions and prolong lifetime. This research proposes an Energy-Efficient TPC (EETPC) mechanism using the measured Received Signal Strength (RSS) to calculate the ideal transmission power. This includes the investigation of the impact factors on RSS, such as distance, height above ground, multipath environment, the capability of node, noise and interference, and temperature. Furthermore, a Dynamic Node Lifetime Estimation (DNLE) technique for WSNs is also presented, including the impact factors on node lifetime, such as battery type, model, brand, self-discharge, discharge rate, age, charge cycles, and temperature. In addition, an Energy-Efficient and Lifetime Aware Routing (EELAR) algorithm is designed and developed for prolonging network lifetime in multihop WSNs. The proposed routing algorithm includes transmission power and lifetime metrics for path selection in addition to the Expected Transmission Count (ETX) metric.

Both simulation and real hardware testbed experiments are used to verify the effectiveness of the proposed schemes. The simulation experiments run on the AVRORA simulator for two hardware platforms: Mica2 and MicaZ. The testbed experiments run on two real hardware platforms: the N740 NanoSensor and Mica2. The corresponding implementations are on two operating systems: Contiki and TinyOS. The proposed TPC mechanism covers those investigated factors and gives an overall performance better than the existing techniques, i.e. it gives lower packet loss and power consumption rates, while delays do not significantly increase. It can be applied for single-hop with multihoming and multihop networks. Using the DNLE technique, node lifetime can be predicted more accurately, which can be applied for both static and dynamic loads. EELAR gives the best performance on packet loss rate, average node lifetime and network lifetime compared to the other algorithms and no significant difference is found between each algorithm with the packet delay.

Acknowledgements

First of all I would like to thank Thammasat University for the scholarship they provided me during four and a half years of my PhD study at Loughborough University.

I would like to thank my supervisors, Dr Iain Phillips and Dr Lin Guan, for their support and valuable suggestions. They provided me all information I needed to focus and achieve my goals.

I would like to give my special thanks to my friends, Dr Peter Bull and Dr Pornrawee Thunnithet, for their extensive comments and encouragement. They have helped improve my writing skills and have also helped me to present my research in a better way. A special word of thanks also goes to Kannikar Subsomboon, for her encouragement and support.

Finally, thanks go to my family for being a great source of motivation for me.

Publications

- W. Rukpakavong and I. Phillips and L. Guan "Neighbour Discovery for Transmit Power Adjustment in IEEE 802.15.4 Using RSSI" *Proceedings of 4th IEEE Conference on New Technologies, Mobility and Security (NTMS) IFIP International*, 7-10 February 2011.

- W. Rukpakavong and I. Phillips and L. Guan and G. Oikonomou "RPL Router Discovery for Supporting Energy-Efficient Transmission in Single-hop 6LoWPAN" *Proceedings of IEEE ICC'12 WS-E2Nets Workshop On Energy Efficiency in Wireless Networks & Wireless Networks for Energy Efficiency*, 10-15 June 2012.

- W. Rukpakavong, I. Phillips and L. Guan "Lifetime Estimation of Sensor Device with AA NiMH Battery" *ICICM 2012 : IACSIT International Conference on Information Communication and Management*, 26-27 October 2012.

- W. Rukpakavong, L. Guan and I. Phillips "Dynamic Node Lifetime Estimation for Wireless Sensor Networks" *IEEE Sensors Journal*, vol.14, no.5, 2014.

Contents

Abstract **3**

Acknowledgements **5**

Publications **7**

1 Introduction **23**
1.1 Motivation . . . 23
1.2 Research Objectives . . . 24
1.3 Original Contributions . . . 25
1.4 Structure of the Thesis . . . 26

2 Background **29**
2.1 Introduction . . . 29
2.2 An Overview of Wireless Sensor Networks (WSNs) . . . 29
2.2.1 Hardware Platforms . . . 31
2.2.2 Operating Systems . . . 34
2.2.3 WSN Simulators . . . 37
2.2.4 Neighbour Discovery (ND) . . . 39
2.2.5 Routing in WSNs . . . 40
2.2.6 Wireless Sensor Batteries . . . 41
2.2.7 Energy Consumption and Energy Aware Techniques . . . 44
2.2.8 Node and Network Lifetime . . . 48
2.3 Integrating IP in WSNs . . . 52
2.3.1 IEEE 802.15.4 . . . 53
2.3.2 IPv6 over Low-power Wireless PAN (6LoWPAN) . . . 56
2.3.3 6LoWPAN Neighbour Discovery Protocol . . . 57
2.3.4 Routing Protocol for Low-Power and Lossy Networks (RPL) 59
2.4 Wireless Transmission . . . 60
2.4.1 Antenna . . . 60
2.4.2 Path Loss (PL), Received and Transmitted Power . . . 61

2.4.3 Received Signal Strength Indicator (RSSI) . . . 61
2.4.4 Link Quality Indicator (LQI) . . . 62
2.5 Summary and Discussion . . . 62

3 Transmission Power Control (TPC) 65
3.1 Introduction . . . 65
3.2 Existing Transmission Power Control Algorithms in MANETs . . . 66
3.2.1 MAC Layer . . . 66
3.2.2 Network Layer . . . 66
3.2.3 Discussion . . . 67
3.3 Existing Transmission Power Control Algorithms in WSNs . . . 68
3.3.1 Finding the Ideal Power . . . 68
3.3.2 Dynamic Transmission Power Adjustment . . . 71
3.4 Summary and Discussion . . . 74

4 Node Lifetime Estimation in WSNs 75
4.1 Introduction . . . 75
4.2 Existing Lifetime Estimation Models . . . 76
4.2.1 Battery Capacity Estimation . . . 76
4.2.2 Current Consumption Estimation . . . 80
4.2.3 Lifetime Estimation . . . 82
4.3 Summary and Discussion . . . 83

5 Energy and Lifetime Aware Routing 85
5.1 Introduction . . . 85
5.2 Existing Energy and Lifetime Aware Routing . . . 85
5.2.1 Resource Aware and Link Quality (RLQ) . . . 86
5.2.2 Collection Tree Protocol (CTP) . . . 87
5.2.3 Energy and Link Quality Based Routing Tree (ELQR) . . . 91
5.2.4 Energy Aware and Link Quality Based Routing (ELR) . . . 92
5.2.5 Routing Protocol for Low-power and lossy network (RPL) . 94
5.2.6 Discussion . . . 96
5.3 Summary and Discussion . . . 98

6 EETPC: Energy-Efficient TPC 99
6.1 Introduction . . . 99
6.2 Experimental Setup . . . 99
6.3 The Impact Factors on RSS . . . 100
6.3.1 Distance . . . 100
6.3.2 Transmission Power . . . 101

6.3.3 Height above ground . . . 102
6.3.4 Multipath . . . 103
6.3.5 Node's Capability . . . 104
6.3.6 Temperature Effect . . . 104
6.3.7 Noise and Interference . . . 106
6.4 EETPC Design and Implementation . . . 109
6.4.1 EETPC Design . . . 109
6.4.2 EETPC Implementation . . . 111
6.5 EETPC Evaluation . . . 113
6.5.1 Finding the ideal Power . . . 113
6.5.2 Dynamic Transmission Power Adjustment . . . 125
6.6 Summary and Discussion . . . 128

7 DNLE: Dynamic Node Lifetime Estimation 131
7.1 Introduction . . . 131
7.2 The Impact Factors on Node Lifetime . . . 131
7.2.1 Battery Types, Brands and Models . . . 131
7.2.2 Self-discharge . . . 133
7.2.3 Discharge Rate . . . 134
7.2.4 Ageing . . . 135
7.2.5 Charge Cycles . . . 135
7.2.6 Temperature . . . 135
7.3 DNLE Design and Implementation . . . 137
7.3.1 DNLE Design . . . 138
7.3.2 DNLE Implementation . . . 140
7.4 DNLE Evaluation . . . 141
7.4.1 Scenario1: Static Load . . . 142
7.4.2 Scenario2: Dynamic Load . . . 144
7.5 Summary and Discussion . . . 146

8 EELAR: Energy-Efficient and Lifetime Aware Routing 149
8.1 Introduction . . . 149
8.2 EELAR Design and Implementation . . . 149
8.2.1 EELAR Design . . . 150
8.2.2 EELAR Implementation . . . 153
8.3 EELAR Evaluation . . . 155
8.3.1 Testbed Experiments . . . 156
8.3.2 Simulation Experiments . . . 160
8.4 Summary and Discussion . . . 164

9 Conclusions and Future Work **165**
9.1 Conclusions 165
9.2 Future Work 166
9.2.1 Energy-Efficient Transmission Power Control (EETCP) . . . 166
9.2.2 Dynamic Node Lifetime Estimation (DNLE) 166
9.2.3 Energy-Efficient and Lifetime Aware Routing (EELAR) . . . 167

References **169**

List of Figures

1.1 Sensor Node Components 23

2.1 Star Topology (Single-hop Network) 30
2.2 Mesh Topology (Multihop Network) 30
2.3 Mica2 Sensor Node 32
2.4 MicaZ Sensor Node[5] 33
2.5 Nanosensor N740 34
2.6 Compilation Process for NS2 and Real Hardware 37
2.7 Compilation Process for TOSSIM/Cooja and Real Hardware 38
2.8 Compilation Process for AVRORA and Real Hardware 39
2.9 Simple ND for Symmetric Link 39
2.10 Simple ND for Asymmetric Link 40
2.11 An Electrochemical of Battery Cell [58] 41
2.12 Self-discharing Process [113] 41
2.13 Temperature and discharge rate effect on alkaline batteries [122] . . 43
2.14 Temperature and Discharge Rate Effect on NiMH Batteries [122] . . 44
2.15 Energy Consumption Model in FSM of CC1000 (a) and CC2420 (b) 45
2.16 Processor States 46
2.17 Other Devices States 46
2.18 Switching Between Sleep and Active Modes 47
2.19 Discovering Between Active Sensors 47
2.20 Basic Transmission to Neighbours 48
2.21 Network Fails When All Nodes Die (a) Before Failure, (b) Network Failure 49
2.22 Network Fails When First Node Dies (a) Before Failure, (b) Network Failure 49
2.23 Topology Change (a) Before Node Failure, (b) After Node N0 Failure 49
2.24 Network Fails with Some Alive Nodes (a) Before Failure, (b) Network Failure 50
2.25 Network Fails When All Nodes in the Coverage Area Dies (a) Before Failure, (b) Network Failure 50

2.26 Network Fails with Some Alive Nodes in the Coverage Area (a) Before Failure, (b) Network Failure . . . 51
2.27 Network Fails When Number of Connected Nodes < 50% (a) Before Failure, (b) Network Failure . . . 51
2.28 Network Fails When Number of Connected Nodes in the Coverage < 50% (a) Before Failure, (b) Network Failure . . . 51
2.29 Protocol Stack of 6LoWPAN . . . 52
2.30 IEEE 802.15.4 PPDU Format [50] . . . 54
2.31 IEEE 802.15.4 MAC Frame Format [50] . . . 55
2.32 Layout of 6LoWPAN Headers . . . 56
2.33 Simple 6LoWPAN . . . 57
2.34 Router Discovery and Node Registration in 6LOWPAN-ND . . . 58
2.35 Multihop Registration in 6LOWPAN-ND . . . 58
2.36 RPL for Router Discovery . . . 59
2.37 Wireless Communication Systems [89] . . . 60

3.1 The Relationship Between Transmission Power and Current Consumption of CC1000 and CC2420 . . . 66
3.2 Best Neighbour in Every Cone . . . 67
3.3 Three Collinear Nodes within Reach of Each Other . . . 69
3.4 Scanning with Different Transmission Power Levels . . . 70
3.5 Maintenance the Number of Neighbours . . . 72

4.1 Battery Capacity Indicator Developed by Heyer[51] . . . 77
4.2 Computerized Battery Analyzer (CBA-III) [87] . . . 79

5.1 Four-Bits Link Estimator [36] . . . 87
5.2 Neighbour Replacement Policy . . . 88
5.3 The CTP Routing Frame [40] . . . 90
5.4 CTP Algorithm . . . 91
5.5 ELQR Algorithm . . . 92
5.6 bestETXRoute Algorithm . . . 93
5.7 bestEnergyRoute Algorithm . . . 93
5.8 ELR Algorithm . . . 94
5.9 The RPL Control Message Option for DAG Metric Container [118] 94
5.10 The RPL Routing Metric/Constraint Object [115] . . . 95
5.11 Example Routing Issue . . . 97

6.1 Experimental Sites (a) an Empty Room (b) a Corridor . . . 100
6.2 RSS and Distance Experiment . . . 101
6.3 Measured RSS, Estimated RSS and Distances . . . 101

6.4 Comparison RSS of Different Transmission Power Levels for (a) Mica2 (b) N740 . . . 102
6.5 Comparison RSS of Different Antenna Heights above ground 103
6.6 Comparison between Measured RSS_{max} and Estimated RSS_{max} for Mica2 and N740 . . . 103
6.7 Signal Reflection . . . 104
6.8 Comparison between RSS of Node1 and Node 2 . . . 104
6.9 Temperature Experiment . . . 105
6.10 Effects of High Temperature on RSS for (a) Mica2 (b) N740 Nanosensor . . . 106
6.11 Effects of Low Temperature on RSS for (a) Mica2 (b) N740 Nanosensor . . . 107
6.12 Effects of Noise and Interference on RSS for (a) Heater (b) Refrigerator . . . 108
6.13 Modified Asymmetric Link ND . . . 111
6.14 Subset of RPL for supporting EETPC (a) upward traffic (b) upward and downward traffics . . . 112
6.15 (a) Hidden Terminal (b) Concurrent Transmission . . . 114
6.16 Energy Consumption with Different Transmission Power Levels . . . 115
6.17 Comparison Testbed Result of Discovery Process at Strong Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption . . . 116
6.18 Testbed Overall Energy Consumption at Strong Signal Position . . . 117
6.19 Comparison Testbed Result of Discovery Process at Weak Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption . . . 118
6.20 Testbed Overall Energy Consumption at Weak Signal Position . . . 118
6.21 Comparison Simulation Result of Discovery Process at Strong Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption . . . 120
6.22 Simulation Overall Energy Consumption at Strong Signal Position . 120
6.23 Comparison Simulation Result of Discovery Process at Weak Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption . . . 121
6.24 Simulation Overall Energy Consumption at Weak Signal Position . 122
6.25 16 Nodes in 8x16m Area . . . 123
6.26 Comparison Topology for Choosing Base Station between (a) First (b) Minimum TX power . . . 123
6.27 Multihoming Network Result . . . 124

6.28 Topology Construction by RPL 124
6.29 Multihop Network Result 125
6.30 Comparison of Decreasing Temperature between (a) Static (b) RSSI Feedback (c) Updated RSS ($AccRSS_{max}$) by Discovery Process (d) Temperature Effect Equation 127
6.31 Comparison of Increasing Temperature between (a) Static (b) RSSI Feedback (c) Updated RSS by Discovery Process (d) Temperature Effect Equation 128

7.1 Dynamic Noad Lifetime Estimation Experiment 142
7.2 The Deviation of Remaining Lifetime Estimation for Static Load Testbed of N740 with A-Alkaline Batteries at (a) 22 °C (b) 10 °C . . 143
7.3 The Deviation of Remaining Lifetime Estimation for Dynamic Load Testbed of N740 with A-Alkaline Batteries at (a) 22 °C (b) 10 °C . 145

8.1 The Architecture of EELAR 150
8.2 Case1:Path Selection Based on Link ETX 151
8.3 Case1:Path Selection Based on Tx Power 151
8.4 Case2:Path Selection Based on Path Lifetime 152
8.5 Case2:Path Selection Based on Link ETX 152
8.6 Case2:Path Selection Based on Tx Power 152
8.7 Case3:Path Selection Based on Path ETX 153
8.8 Modified CTP Routing Frame 154
8.9 Testbed Scenario1 with Three Nodes 157
8.10 Testbed Topology Construction of Three Nodes for (a) RPL/CTP and ELR (b) EELAR 157
8.11 Testbed Scenario2 with Four Nodes 158
8.12 Testbed Topology Construction of Four Nodes for (a) RPL (b) ELR (c) EELAR 159
8.13 Simulation Scenario3 with Thirty-two Nodes 162
8.14 Simulation Topology Construction of Thirty-two Nodes for (a) RPL (b) ELR (c) EELAR 163

List of Tables

2.1 Mica2 Operating Conditions 32
2.2 MicaZ Operating Conditions 33
2.3 CC2431 Operating Conditions 34
2.4 IEEE 802.15.4 Features 53
2.5 The Effective Area and Gain of Antennas 61

4.1 The Relationship Between the Specific Gravity and SoC [18] 76
4.2 The Relationship Between Output Voltage and SoC [11] 77
4.3 The Intervals in Voltage and the Corresponding SoC for Sony US18500G3 Li-ion Battery [18] 78

5.1 Routing Metric/Constraint Type 95
5.2 The Summation of Routing Metrics for Path Selection 97

6.1 Testbed Experimental Result at Strong Signal Position 117
6.2 Testbed Experimental Result at Weak Signal Position 117
6.3 Simulation Experimental Result at Strong Signal Position 119
6.4 Simulation Experimental Result at Weak Signal Position 121
6.5 Results of Decreasing Temperature 126
6.6 Results of Increasing Temperature 127

7.1 Quoted Capacity for 100 mA Discharge to 0.9 V Cut-off at Room Temperature 132
7.2 Measured and Estimated Lifetime of 100 mA Discharge to 2.0 V Cut-off at Room Temperature 132
7.3 Lifetime of 100 mA Load with Two AA NiMH Batteries 133
7.4 Measured and Estimated Lifetime for Different Starting Voltage of B-NiMH2000 Batteries 134
7.5 Measured and Estimated Lifetime for Different Starting Voltage of D-NiMH2500 Batteries 134
7.6 Capacity of Different Temperatures and Constant Value for Different batteries 136

7.7 Current Draw at Different Temperatures and S Value for Mica2 and N740 Motes . . . 136
7.8 Ratio Constant Value of Different Batteries for Mica2 and N740 Motes . . . 137
7.9 Lifetime with Different NiMH Batteries and Temperatures for N740 Motes . . . 137
7.10 Preconfigured Capacities of Batteries . . . 141
7.11 Average Deviation of Static Load Lifetime Estimation Testbed Results for Mica2 and N740 Motes with Different Batteries . . . 144
7.12 Average Deviation of Dynamic Lode Lifetime Estimation Testbed Results for Mica2 and N740 Motes with Different Batteries . . . 146

8.1 Testbed Performance Comparison of Three Path Selection Algorithms for Three Nodes of Mica2 . . . 157
8.2 Testbed Performance Comparison of Three Path Selection Algorithms for Three Nodes of N740 . . . 157
8.3 Testbed Performance Comparison of Three Path Selection Algorithms for Four Nodes of N740 . . . 159
8.4 Simulation Performance Comparison of Three Path Selection Algorithms for Three Nodes of Mica2 . . . 161
8.5 Simulation Performance Comparison of Three Path Selection Algorithms for Three Nodes of MicaZ . . . 161
8.6 Simulation Performance Comparison of Three Path Selection Algorithms for Four Nodes of MicaZ . . . 162
8.7 Simulation Performance Comparison of Three Path Selection Algorithms for Thirty-two Nodes of MicaZ . . . 163

List of Abbreviations

6LoWPAN	IPv6 over Low-power Wireless PAN
ADC	Analog-to-Digital Converter
ATPC	Adaptive Transmission Power Control
CTP	Collection Tree Protocol
DAG	Directed Acyclic Graph
DAO	DODAG Destination Advertisement Object
DIO	DODAG Information Object
DNLE	Dynamic Node Lifetime Estimation
DODAG	Destination Oriented Directed Acyclic Graph
EELAR	Energy-Efficient and Lifetime Aware Routing
EETPC	Energy-Efficient Transmission Power Control
EMF	ElectroMotive Force
ETX	Expected Transmission Count
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LOS	Line-Of-Sight
LNSM	Log-Normal Shadow Model
LQI	Link Quality Indicator
mAh	milliAmpere-hour
mAm	milliAmpere-minute
MANET	Mobile Ad-hoc NETwork

ND	Neighbour Discovery
NiMH	Nikel-Metal-Hybride
PRR	Packet Reception Rate
RPL	Routing Protocol for Low-power and lossy network
RSS	Receive Signal Strength
RSSI	Receive Signal Strength Indicator
SoC	State of Charge
SINR	Signal to Interference plus Noise Ratio
TPC	Transmission Power Control
Tx	Transmission
WSN	Wireless Sensor Network

List of Symbols

A	The Effective Area
B	The Battery Constant Value
C	Battery Capacity
C_{rem}	The Remaining Battery Capacity
E	Energy Consumption
G	The Antenna Gain
I	Current Draw
k	Peukert Constant
K	The Adjustment Factor for LOS
Lt	Lifetime
Lt_{rem}	The Remaining Lifetime
PL	Path Loss
S	The Constant of Sensor Circuit
T	The Temperature
T_i	The Start Running Temperature
T_t	The Load Testing Temperature
V_{dd}	Voltage Drain Drain
Vt	Terminal Voltage
Vt_{24h}	The 24h Self-Discharge Terminal Voltage
Vt_f	The Fully Charged Terminal Voltage

Chapter 1

Introduction

In this chapter, the motivation of this research is introduced. Later, it explains the objectives of this research including all contributions.

1.1 Motivation

A Wireless Sensor Network (WSN) consists of a large number of sensor nodes. Each node is a tiny computing device which is made up of four basic components: a sensing unit, processing unit, transceiver unit and power unit as shown in Figure 1.1 [14, 34]. The sensing unit is composed of sensors and Analogue-to-Digital Converters (ADCs), which are used to collect and disseminate environment data. The processing unit comprises of an 8/16 bit microprocessor and small memory storage used to store temporary data during processing. Commercially, each node typically has between 2 and 512 kilobytes of RAM [31, 57]. The transceiver unit is used for sending and receiving data through a wireless channel. All these units are powered by batteries within the power unit.

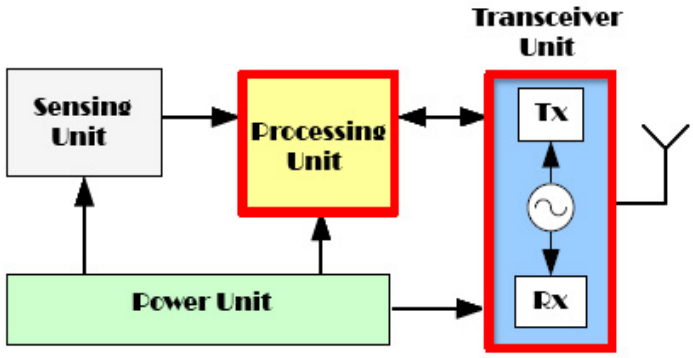


Figure 1.1: Sensor Node Components

For potential large scale networks, sensor nodes are required to be small, low cost and ultimately expendable. Furthermore, due to power limitations, it is necessary for each node to consume as little power as possible in order to extend the lifetime of an individual sensor node and also that of the network. Node lifetime is the time period for which a node can receive, transmit or forward data to others. Therefore, lifetime and energy consumption are important issues in WSNs. With an accurate lifetime estimation, routing algorithms are able to make intelligent decisions that can help conserve energy and prolong the node lifetime.

The energy consumption of the communication unit is much more than that of other units [14, 34, 57]. As a result, trying to reduce the communication energy consumption is a challenging topic in WSNs for prolonging the node lifetime. One of the methods is to minimise the power needed for transmission by using the Transmission Power Control (TPC) technique. The TPC module is normally employed in the MAC/PHY layer; however, it may need support from other processes for effectiveness. For example, the discovery process may provide useful information, such as neighbour nodes with some power saving parameters.

In multihop wireless sensor routing protocols, the most commonly used metric for path selection is the Expected Transmission Count (ETX). The path with the lowest ETX is the best path, which will be selected for forwarding data. However, the transmission (Tx) power and lifetime should be taken as metrics for path selection. In order to balance the network lifetime, a node should forward data to the highest lifetime path. Moreover, a node with the lowest lifetime should select a good quality link between itself and the next hop node in order to reduce energy wastage due to packet loss and avoid retransmissions. Furthermore, a node should select a next hop node that it can send data to with the minimum Tx power in order to prolong its lifetime.

For this reason, it is challenging to have an efficient TPC algorithm that has support from the discovery process, the accurate node lifetime estimation, and the energy-efficient and lifetime aware routing algorithm for lifetime increasing and balancing among the nodes in sensor networks. This motivates this work to study these subjects and generate original contributions in these areas.

1.2 Research Objectives

The main aim of this thesis is to improve the routing algorithm for WSNs in terms of energy efficiency and node lifetime balancing in the network. The specific research objectives associated with achieving the research aim are as follows:

- Research the existing TPC techniques, lifetime estimation models, and en-

ergy or lifetime aware routing algorithms to obtain a better understanding of these topics.

- Propose and design a new TPC mechanism based on RSS in order to reduce the power required for transmission, including the investigation of the impact factors on RSS.
- Propose and design a new lifetime estimation method to improve the accuracy of the lifetime estimation, including the investigation of the impact factors on node lifetime.
- Propose and design a new routing algorithm to increase both node and network lifetime.
- Implement the proposed methods on real sensor platforms to ensure that they are implementable in real WSNs.
- Use both simulator and real hardware motes to evaluate the performance of the proposed techniques and compare with the other approaches.

1.3 Original Contributions

The contributions of this research are listed as follows:

- Investigation of the impact factors on Receive Signal Strength (RSS)

This research investigates the impact factors on RSS, such as distance, height above ground, multipath environment, the capability of the node, temperature and interference noise. RSS estimation for any temperatures is proposed to explain the effect of this impact factor.

- Energy-Efficient Transmission Power Control (EETPC) mechanism

This research proposes a new mechanism for finding the ideal transmission power in order to reduce the energy consumption and extend node lifetime. This method can be applied for single-hop, multihoming and multihop WSNs. The proposed method includes the implementation on real hardware and software platforms in WSNs. Moreover, the discovery algorithms for supporting the transmission power control technique are presented. These algorithms are a simple Neighbour Discovery (ND) for asymmetric links and a subset of RPL (Routing Protocol for Low-power and lossy networks).

- Investigation of the impact factors on node lifetime

This research investigates the impact factors on node lifetime, such as battery type, brand and model, self-discharge, charging rate, age, charging cycles and temperature. Lifetime equations for any starting voltage, ageing, charge cycles and temperatures are proposed to explain the effect of the impact factors.

- Dynamic Node Lifetime Estimation (DNLE) mechanism

This research proposes a dynamic node lifetime estimation mechanism for estimating the running time of sensor nodes which covers the investigated factors. It includes the implementation technique on real hardware and software platforms with different commercial batteries.

- Energy-Efficient and Lifetime Aware Routing (EELAR) algorithm

This research proposes a new energy-efficient and lifetime aware algorithm for path selection in order to maximise node and network lifetime in WSNs. This algorithm can be embedded into the existing WSN routing protocols: CTP (Collection Tree Protocol) and RPL (Routing Protocol for Low-power and lossy networks).

1.4 Structure of the Thesis

The rest of the thesis is organised as follows:

Chapter 2 gives the overview of WSNs and their topologies including their differences from other wireless paradigms, sensor node platform and operating systems, the basic concept of neighbour discovery and routing protocol, WSN simulators, wireless sensor batteries, and node and network lifetime. Then, integrating IP in WSNs is described as well as energy consumption and energy aware techniques. Finally, the concept of wireless communication and some terms are described.

Chapter 3 introduces the basic concept of transmission power control technique and the current transmission power control algorithms are explored.

Chapter 4 explains the basic concept of node lifetime and the current lifetime estimation techniques are presented.

In Chapter 5, the concept of network lifetime is discussed and lifetime aware routing techniques are explored.

Chapter 6 presents the proposed equation and mechanism for finding the ideal transmission power. The feasibility of the proposed scheme is validated by both real hardware testbed and simulation experiments using performance metrics, such as delay, energy consumption and packet loss rate with different scenarios.

In Chapter 7, the proposed node lifetime estimation technique is presented. Testbed experiments are designed and implemented to verify the proposed technique.

For Chapter 8, the new energy-efficient and lifetime aware algorithm is proposed for path selection in routing layer. The path selection algorithm is designed and implemented on CTP and RPL. The feasibility of the proposed scheme is validated by both real hardware testbed and simulation experiments using performance metrics, such as average node lifetime, network lifetime, packet loss and packet delay with different scenarios.

Finally, Chapter 9 summarises the whole thesis, and proposes some areas for further study.

Chapter 2

Background

2.1 Introduction

An overview of WSNs is explained in this chapter. This includes WSN topologies and the differences between WSNs and Mobile Ad-hoc NETworks (MANETs). Then, popular sensor node platforms, operating systems, and WSN simulators are discussed. Later, the basic concepts of neighbour discovery and routing protocols are explained. Wireless sensor batteries as well as node and network lifetime concepts are described. Then, integrating IP in WSNs is also described. Finally, the concept of energy consumption, energy aware techniques, and wireless communication and some terms are explained.

2.2 An Overview of Wireless Sensor Networks (WSNs)

A WSN typically consists of two device types [124]. The first type is sensor nodes, also known as motes. The second is the base station, or gateway, or sink, which collects all data from the sensor nodes and stores it for later use. Each sensor node performs the main tasks, such as event detection, local processing and reporting to the base station. Some sensor nodes may fail due to a lack of power, or have physical damage or environmental interference. The failure of sensor nodes should not affect the overall task of the sensor network. Therefore, it is necessary to use a lot of nodes to obtain a reliable system. Two common topologies in WSNs are star and mesh. For a star topology or single-hop WSN, all sensor nodes can communicate with the base station directly as shown in Figure 2.1. This topology is simple and does not require a routing protocol or extra overhead in the messages. However, this network has a limited coverage area which restricts applications to follow the range of radio communication.

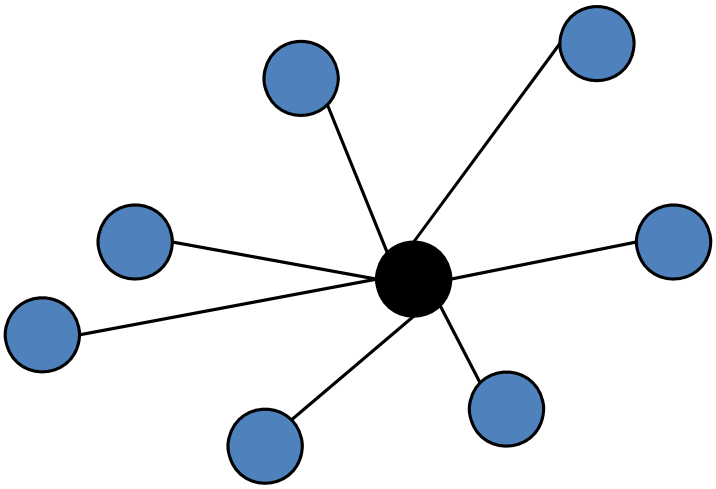


Figure 2.1: Star Topology (Single-hop Network)

The mesh topology covers a wide area by forming a multihop network. If the nodes are not within the transmission range of the base station, their messages have to be forwarded by other nodes as in Figure 2.2. The mesh topology requires an efficient and light ad hoc routing protocol. The role of relay nodes is very important, particularly the neighbours of the base stations. These nodes will consume more energy than others because they have to transmit their own data as well as forwarding data from others. In some cases, many sensor nodes may be unable to communicate with the base stations owing to the low energy of the relay nodes.

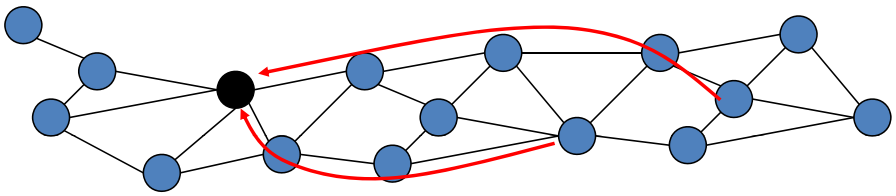


Figure 2.2: Mesh Topology (Multihop Network)

In order to design WSNs to provide some services that are needed, it is necessary to study the difference between the WSNs and other wireless networks. A WSN is similar to a Mobile wireless Ad-hoc NETwork (MANET) [26]. Both are composed of wireless devices that can dynamically self-organise to form a network without necessarily using any pre-existing infrastructure. However, there are some differences between WSN and MANET. First, a MANET is usually a distributed network, while a WSN is a centralised system. A MANET device will normally open a communication channel with other devices in the network as part of its normal functionality. On the other hand, normal traffic in a WSN is sent from

a sensor node to the base station. Second, many sensors may generate the same data within the phenomenon concerned. Such redundancy needs to be exploited to improve energy and bandwidth utilisation. Third, unlike a node in a MANET, a sensor node usually has limited battery power, computation and memory capacity, which requires careful resource management. Finally, a MANET topology may change rapidly and unpredictably because it is a dynamic network with devices continuously entering and leaving the group. But in the case of WSN, all nodes usually stay inside the network.

2.2.1 Hardware Platforms

Hardware solutions are normally based on popular microprocessor groups, such as the MSP430 family from Texas Instruments, 8051 MCU from Intel, and the ATmega from Atmel semiconductor. Chipcon CC2420, the standard IEEE 802.15.4 compliant radio chips, is one of the most commonly used for the RF transceiver [92]. The current majority of research in sensor networks is based on a series of sensor nodes called MICA motes, developed by UC Berkeley [52]. One of the popular motes is the Mica2 featuring a CC1000 radio and the ATMega128L. The latest line of motes is Telos, the low power wireless sensor using the MSP430 microcontroller, Chipcon CC2420 IEEE 802.15.4 radio and a USB port for easy programming [108]. This thesis focuses on 3 hardware platforms: Mica2, MicaZ and N740 NanoSensor. These platforms cover 2 microprocessor groups: 8051 MCU and ATmega, and 2 radio transceivers which are the CC1000 and CC2420.

Mica2

Mica2 [4] is the sensor product developed by Crossbow Technology Inc. Mica2 is based on the Atmel AVR ATmega128L [12] and Chipcon CC1000 [1] radio interface. The ATmega128L is a low-power 8MHz microcontroller with 128 kB flash memory, 4 kB RAM [4]. The ChipCon CC1000 has 3 radio states: SLEEP, TX and RX. It allows for sending at a frequency of 900 MHz with up to 38.4 kbps data rates and programming the transmission power from -20 to 10 dBm (this research focuses on only 4 power levels: -15, -10, -5 and 0 dBm). Moreover, it provides a Received Signal Strength Indicator (RSSI) value of a received packet by measuring the input power. The RSSI is very useful for a great deal of research including this study. Mica2 image and the operating conditions are depicted in Figure 2.3 and Table 2.1 [1, 4, 12].

Figure 2.3: Mica2 Sensor Node

Table 2.1: Mica2 Operating Conditions

Supply voltage	2.7 to 3.3 V
RF frequency range	869-910 MHz
Transmit bit rate	38.4 Kbps
Nominal output power in TX mode	5 dBm
Receiver Sensitivity	-104 dBm
Programmable output power	-20 to 10 dBm
Current consumption: Radio SLEEP	1 μA
Current consumption: Radio RX	9.6 mA
Current consumption: Radio TX,0/-5/-10/-15 dBm	16.8/13.8/10.1/9.3 mA
Current consumption: MCU active	8 mA
Current consumption: Power down	< 15 μA

MicaZ

MicaZ [5], also developed by Crossbow Technology Inc., is based on the Atmel AVR ATmega128L [12] and Chipcon CC2420 [7] radio interface. The ChipCon CC2420 allows for sending at a frequency of 2.4 GHz with up to 250 kbps data rates and programming the output power from -25 to 0 dBm. The RSSI value of a received packet is also provided by using the average of 8 symbol periods (128 μs). Furthermore, a new state, called IDLE, is introduced for radio transceiver, which is different from traditional radio like CC1000. Many MAC protocols use this state for energy saving instead of using SLEEP state since it is faster to transit from IDLE to TX or RX. An image of the MicaZ and the operating conditions are shown in Figure 2.4 and Table 2.2 [5, 7, 12].

Figure 2.4: MicaZ Sensor Node[5]

Table 2.2: MicaZ Operating Conditions

Supply voltage	2.7 to 3.3 V
RF frequency range	2.400 to 2483.5 MHz
Transmit bit rate	250 Kbps
Nominal output power in TX mode	0 dBm
Receiver Sensitivity	-94 dBm
Programmable output power	-25 to 0 dBm
Current consumption: Radio SLEEP	1 μA
Current consumption: Radio IDLE	20 μA
Current consumption: Radio RX	19.7 mA
Current consumption: Radio TX, 0/-5/-10/-15 dBm	17.4/14/11/9.9 mA
Current consumption: MCU active	8 mA
Current consumption: Power down	< 15 μA

N740 NanoSensor

The N740 NanoSensor [69] is a sensor product from Sensinode, the pioneer in IP-based wireless sensor networking technology. It is based on CC2430/2431 [8, 9]. The CC2430/2431 technology combines the CC2420 RF transceiver with an enhanced 8051 microcontroller. The Intel 8051 MCU architecture was designed with up to 128 kB flash memory and 8 kB of RAM. The energy consumption of this processor is based on the processor activity, which can be categorised into 3 levels: high, medium and low activities. Moreover, it provides 3 power down modes to be chosen. Furthermore, this platform includes a location engine used to estimate the position of nodes in a network. An image of the N740 NanoSensor and the operating conditions are depicted in Figure 2.5 and Table 2.3 [8, 9].

Figure 2.5: Nanosensor N740

Table 2.3: CC2431 Operating Conditions

Supply voltage	2 to 3.6 V
RF frequency range	2400 to 2483.5 MHz
Transmit bit rate	250 Kbps
Nominal output power in TX mode	0 dBm
Receiver Sensitivity	-92 dBm
Programmable output power	-25 to 0 dBm
Current consumption: Radio SLEEP	N/A
Current consumption: Radio IDLE	N/A
Current consumption: Radio RX	17.2 mA
Current consumption: Radio TX, -0.4/-5.7/-10.5/-15.4 dBm	17.4/12.4/10.6/9.7 mA
Current consumption: MCU low/medium/high activity	9.5/10.5/12.3 mA
Current consumption: Power down Mode 1/2/3	190/0.5/0.3 μA
Time to estimate node locator	< 40 μs
Location range	64 x 64 m
Reference node location resolution	0.5 m

2.2.2 Operating Systems

Traditionally, sensor nodes are usually programmed using an operating system. Several operating systems were developed for WSNs which vary from traditional operating systems in terms of goals and techniques. The concept of event driven operating systems has been implemented owing to the event driven nature of sensor network applications. Two poplular event-driven operating systems, TinyOS and Contiki, are examied and used for the experiments. These operating systems support IPv6 network, called 6LoWPAN (IPv6 over Low power Wireless Personal Area Network). The details of 6LoWPAN will be described in section 2.3.2.

TinyOS

TinyOS [19, 35, 38] is an open source operating system originated at UC Berkeley. It is an event-driven OS with the core requiring only 400 bytes of code and data memory. Its architecture is component-based, which can support concurrent programmes with very low memory requirements.

- Architecture

  TinyOS can be classified as a monolithic architecture class. It uses the component model. All components are written in the C-extension, NesC. Each component is an independent computational entity. Components have 3 computational abstractions: commands, events, and tasks. Commands and events are used for inter-component communication, while tasks are used to express intra-component concurrency. A command is a request to run a service, while the event signals the completion of that service. TinyOS has a single shared stack of both kernel space and user space.

- Programming Model

  Earlier versions of TinyOS did not provide any multi-threading support. This version imposed atomicity by disabling the interrupts, but it does not allow interrupts to be disabled at the user level threads due to system performance and usability issues. The latest version, TinyOS version 2.1, supports multiple threads, called TOS Threads. TOS threads use a cooperative threading method. Application level threads can preempt only other application level threads. The TinyOS scheduler is a high priority thread. The new version also provides synchronisation to support atomic operations in user level threads. The message passing technique is used for communication between the application threads and the kernel.

- Communication protocol support

  TinyOS provides an implementation of several MAC protocols: a single hop TDMA protocol, a TDMA/CSMA hybrid protocol which implements Z-MACs slot stealing optimisation, B-MAC, and an optional implementation of an IEEE 802.15.4 compliant MAC. There are two implementations of 6LoWPAN [47]: 6lowpancli and blip. TinyOS provides many multihop routing protocols including the Collection Tree Protocol (CTP) and IPv6 Routing Protocol for Low power and Lossy Networks (RPL).

Contiki

Contiki [31, 32, 35], implemented in the C language, is a well-known operating system designed for constrained devices. It is an open source lightweight multi-tasking operating system providing dynamic loading and unloading of individual programmes. It uses a protothreads programming technique which combines the benefits of event-driven and multi-threaded programming. Its kernel is event-driven, while the system support is a preemptive multi-threading library.

- Architecture

  The Contiki OS follows the modular architecture. At the kernel level it follows the event driven model, but it provides optional threading facilities to individual processes. The Contiki kernel comprises of a lightweight event scheduler that dispatches events to running processes. Process execution is triggered by events dispatched by the kernel to the processes or by a polling mechanism. The polling mechanism is used to avoid race conditions. Any scheduled event will run to completion, however, event handlers can use internal mechanisms for preemption. All OS facilities, e.g. sensor data handling, communication, and device drivers are provided in the form of services. Each service has its interface and implementation. Applications using a particular service need to know the service interface.

- Programming Model

  Contiki supports preemptive multi-threading. Multi-threading is implemented as a library on top of the event-driven kernel. The library can be linked with applications that require multi-threading. The Contiki multi-threading library is divided into two parts: a platform independent part and a platform specific part. The platform independent part interfaces to the event kernel and the platform specific part of the library implements stack switching and preemption primitives. Preemption is implemented using the timer interrupt and the thread state is stored on a stack. For multi-threading, Contiki uses protothreads [31, 32]. The main features of protothreads are very small memory overhead and no extra stack for a thread. No process synchronisation is provided in Contiki because events have to run until completion and Contiki does not allow interrupt handlers to post new events,

- Communication protocol support

  Contiki implements three communication stacks, Rime, μIP and μIPv6 respectively. Rime is a lightweight non-IP stack at the MAC layer. Rime supports many MAC layer protocols, such as Null-MAC, SMAC, X-MAC, CX-MAC, ContikiMAC and IEEE 802.15.4. The μIP, called micro IP, is IPv4

stack that enables internet connectivity. The implementation of 6LoWPAN in Contiki is called sicslowPAN [102] which is the μIPv6 stack based on RFC4944. The μIPv6 supports Neighbour Discovery, ICMPv6, UDP and TCP. RPL implementation is also available in Contiki.

2.2.3 WSN Simulators

Simulation is a feasible approach to the quantitative analysis of sensor networks. To accurately evaluate the performance of proposed techniques, a small-scale testbed based on the Mica2 and N740 NanoSensor nodes is deployed on both single-hop and multihop scenarios. However, simulation analysis complements the experimental evaluation in large-scale static WSN scenarios or different hardware platforms (i.e. MicaZ), where a testbed implementation becomes unfeasible.

NS-2

Many studies use the NS-2 [55] simulator which can support a considerable range of protocols in all layers. However, NS-2 is designed as a general network simulator. Therefore, it does not consider some unique characteristics of WSN, such as problems of the bandwidth, power consumption or energy saving in WSN. Moreover, C++/Otcl language is used for NS-2, while another language is used in the implementation for real hardware. This means that there are two source codes, one for the simulation and another for real hardware as shown in Figure 2.6.

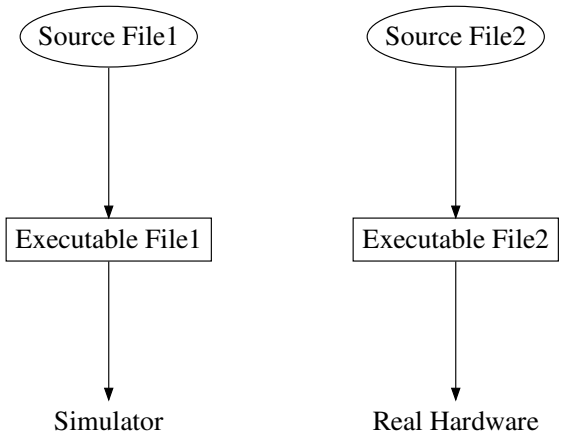


Figure 2.6: Compilation Process for NS2 and Real Hardware

TOSSIM and COOJA

A number of simulators have been developed for understanding the behaviour of WSNs. TOSSIM [65] is an event driven application-level simulator which can be used for TinyOS-based WSNs, while Cooja [80] allows for simulating the hardware details of the Contiki-based sensor nodes. The same source code for the real hardware can be used for both TOSSIM and Cooja. Nevertheless, these two simulators are operating system dependent, which means an application has to be compiled on the TinyOS environment for TOSSIM and Contiki environment for Cooja. The compilation process is illustrated in Figure 2.7.

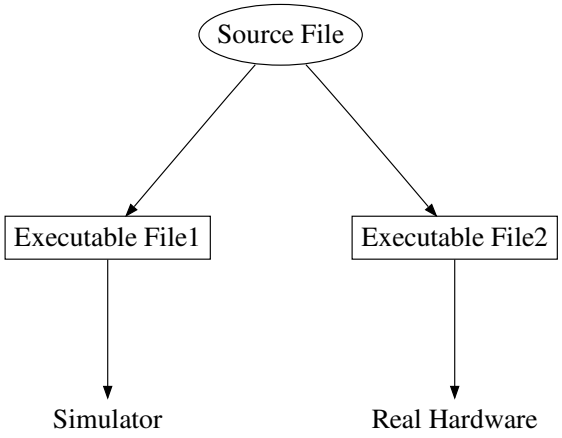


Figure 2.7: Compilation Process for TOSSIM/Cooja and Real Hardware

AVRORA

AVORA [112], a language and operating system independent simulator, is developed by the UCLA as an open-source simulator for embedded sensing programmes. It can run actual AVR microcontroller programmes, and accurately simulate the devices and the radio communication. This simulator has an energy consumption analysis tool (AEON) [78] to simulate the energy consumption of each component on the Mica2 and MicaZ mote, including the radio and the CPU. The same executable file running on the real hardware can be run on this simulator, as shown in Figure 2.8.

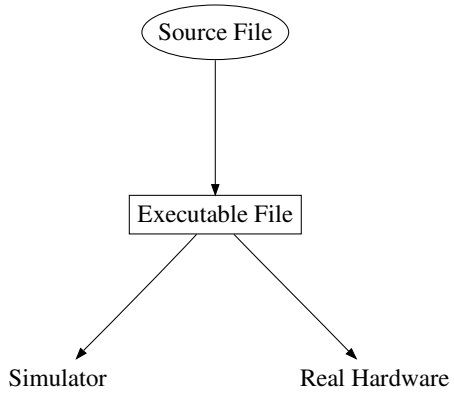


Figure 2.8: Compilation Process for AVRORA and Real Hardware

2.2.4 Neighbour Discovery (ND)

With a large number of sensors, nodes are randomly placed at distributed locations over the area of interest. Normally, all nodes are not previously configured and the network structure cannot be pre-engineered. A large number of nodes should organise themselves to efficiently perform the tasks required by the application after they have been deployed. When a node is powered on, its neighbour table is empty, which means it does not have any knowledge about other installed nodes. Accordingly, each node must send wireless queries to find neighbouring nodes and establish a network topology. Thus, the first step, called neighbour discovery, is to detect the one-hop neighbours with which it can communicate directly. The ND process may be included in other protocols, such as medium-access control protocols and routing protocols and/or may provide some useful information for them. Ideally, the process of neighbour discovery should finish as soon as possible as this will usually indicate a reduced energy requirement. Moreover, it allows other processes to start more quickly. For symmetric links, a node starts sending a "Hello" packet by broadcasting. All nodes that receive that "Hello" message then add that sender node to their neighbour list. The basic ND for symmetric links is shown in Figure 2.9.

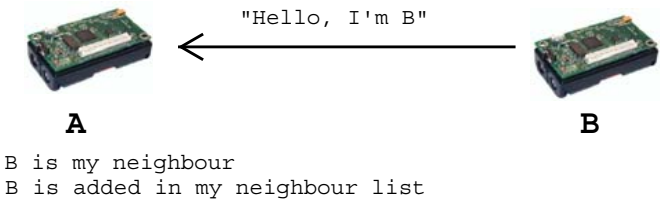


Figure 2.9: Simple ND for Symmetric Link

However, in real WSN, asymmetric links are very common. For example, Node A can receive messages from B but it may not be able to send any messages back to B. The asymmetric links are caused by many factors, such as transmission medium and the variations of node transmission power, especially on a network of heterogeneous sensors. Therefore, the basic discovery process for asymmetric link needs an acknowledgement of the "Hello" packet, as shown in Figure 2.10.

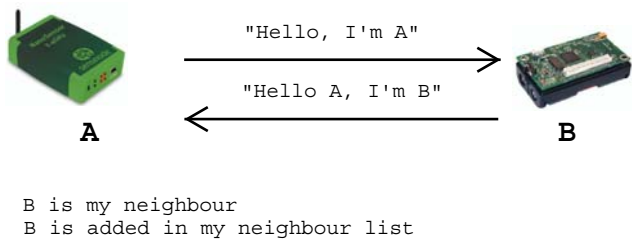


Figure 2.10: Simple ND for Asymmetric Link

2.2.5 Routing in WSNs

To provide a mesh topology in WSN, an ad hoc routing protocol is needed. Since the nature of WSNs is similar to MANET, several interesting routing protocols in MANET are applied in WSN with the light version. Those protocols are based on table-driven, on-demand and hybrid protocols. The table driven protocols usually maintain the routing table of the whole network, which requires huge resources and causes a long delay in updating the internal table. On-demand protocols only establish routes in an on demand basis, which normally presents a high rate of global flooding. A large overhead can easily overwhelm network resources. Hybrid protocols combine both table driven and on-demand protocols, which means an efficient method is required for adapting to operate in WSN. Due to several characteristics and limited resources, routing in sensor networks is very challenging. For example, many applications of sensor networks require the flow from multiple sources to a particular node, the base station. Moreover, data traffic has significant redundancy since multiple sensors in the same regions may generate the same data. Therefore, this redundancy needs to be handled by the routing protocols for data reduction, which leads to energy saving and lower bandwidth utilisation. The WSN routing protocols have been designed to provide low latency, reliable and fault tolerant communication, quick reconfiguration and minimum energy consumption. It is necessary to balance the need to accommodate the limited processing and communication capabilities of the sensor nodes against the overhead required.

2.2.6 Wireless Sensor Batteries

A battery is a power source that converts chemical energy into electrical energy. A battery cell consists of an anode, a cathode and the electrolyte, which separates the two electrodes. An oxidation reaction at the anode takes place during the discharge [58]. An electrochemical of a battery cell is shown in Figure 2.11.

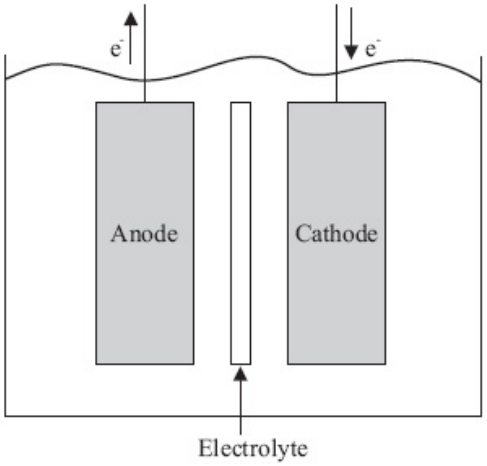


Figure 2.11: An Electrochemical of Battery Cell [58]

Overall battery capacity is measured in milli-Amp-hours (mAh) or Amp-hours (Ah). Sometimes there is insufficient potential energy in the battery to get the remaining charge out if its voltage drops below a certain level. For example, an AA-size of alkaline/Nikel-Metal-Hydride (NiMH) cell is considered as an empty battery if its voltage drops below 1.0 V [3, 10, 11, 13]. Even when not being used, the battery capacity is reduced by the self-discharge process, as shown in Figure 2.12. The amount of electrical self-discharge varies with battery type and chemistry.

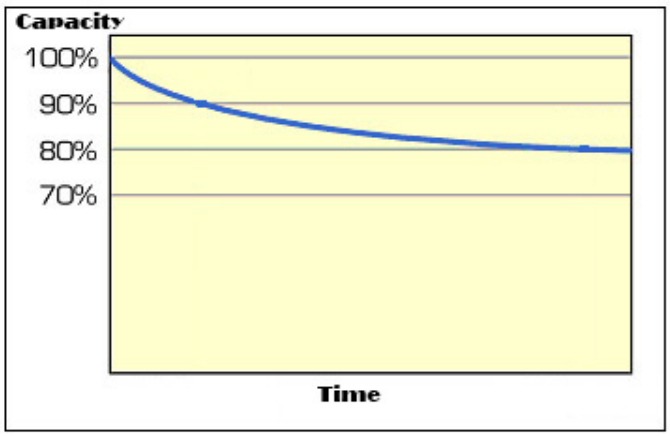


Figure 2.12: Self-discharing Process [113]

There are fundamental battery concepts [58, 122]. The first is the temperature effect. Cool temperatures can slow down the self-discharge process, while the self-discharge rate increases at higher temperature. In contrast, batteries cannot supply full capacity in cool temperatures as they do at the higher temperature. The second fundamental concept is the discharge rate effect. The capacity of a battery is related to the current draw of the device. High current draw reduces battery capacity, e.g. a battery gives a capacity of 1000 mAh for 5 mA current draw, while it gives a current draw of 200 mA with a capacity of 500 mAh. The current draw is often expressed as a C-rate, a measure of the rate in 1 hour which is relative to the battery's maximum capacity. For example, the 1C discharge current of 100 Ah battery capacity is equal to a discharge current of 100 A, while 5C and $\frac{C}{2}$ rate for this battery would be 500 A and 50 A. Some of the variables used to describe a battery are explained as follows [109]:

- State of Charge (SoC)

  SoC is an expression of the current battery capacity as a percentage of maximum capacity. To determine the change in battery capacity over time, SoC is generally calculated using current integration. The case of 100% SoC implies a full battery capacity, while 0% SoC implies the empty capacity.

- Terminal Voltage (Vt)

  Terminal voltage is the voltage between the battery terminals with load applied. It varies with SoC and discharge/charge current.

- Voltage Drain Drain (V_{dd})

  Voltage Drain Drain is the positive supply voltage of a field effect semiconductor device. If the current flows between the battery terminals, it can be assumed that V_{dd} is equal to Vt.

- Nominal Voltage

  Nominal Voltage is the reported or reference voltage of the battery, also sometimes thought of as the normal voltage of the battery.

- Cut-off Voltage

  Cut-off Voltage is the minimum allowable voltage. It is this voltage that generally defines the empty state of the battery.

Since a sensor node is typically powered by AA-size batteries, this research studies two common types of AA-size batteries: Alkaline and Nikel-Metal-Hydride (NiMH). A pair of AA cells is used for many sensor devices, including Mica2, MicaZ and N740 NanoSensor.

Alkaline Batteries

Alkaline is the most common type of AA-size battery. Normally, an alkaline battery produces 1.5 volts. It is low cost, widely available, and suitable for low current draw devices at room temperature. Alkaline cells retain the stored energy best and in storage for 7-10 years and they have the self-discharge rates of around 2-3% per year [113]. However, this battery type is not suitable for using in the cold and under high current draws [111, 122]. Figure 2.13 shows the temperature and discharge rate effect on alkaline batteries.

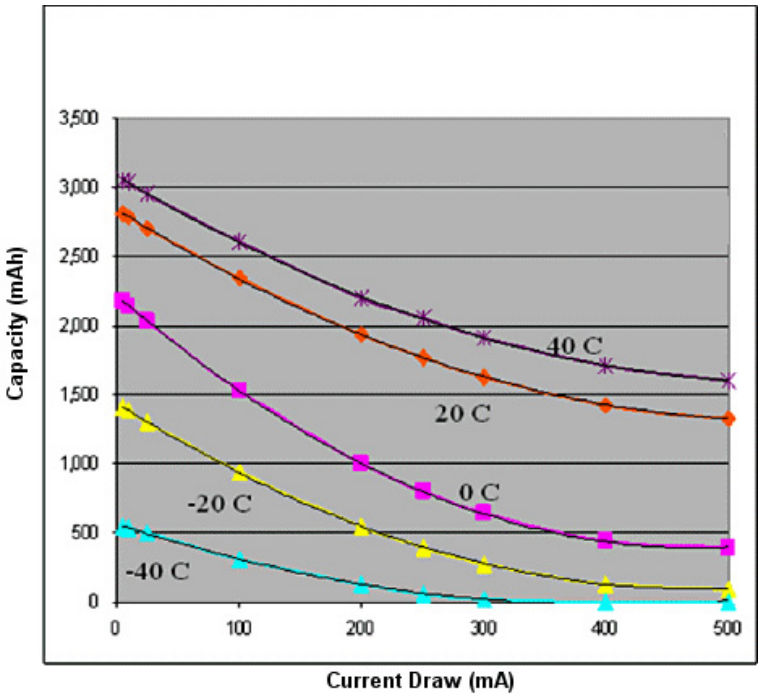


Figure 2.13: Temperature and discharge rate effect on alkaline batteries [122]

Nikel-Metal-Hydride (NiMH) Batteries

NiMH cell is a common AA rechargeable battery at 1.2 volts which has capacity ranging from 1100 mAh (milliampere-hour) to 3100 mAh. The starting voltage of a fully charged cell in good condition is about 1.4-1.45 volts with the nominal voltage 1.2V. The self discharge rate of NiMH batteries is higher than the rate of alkaline batteries. NiMH cells have the self-discharge rates of 5-10% or more in the first 24 hours and then 0.5-1% per day at room temperature [60, 113]. However, this battery type performs better at low temperatures and high current draw. Figure 2.14 shows temperature and discharge rate effect on NiMH batteries.

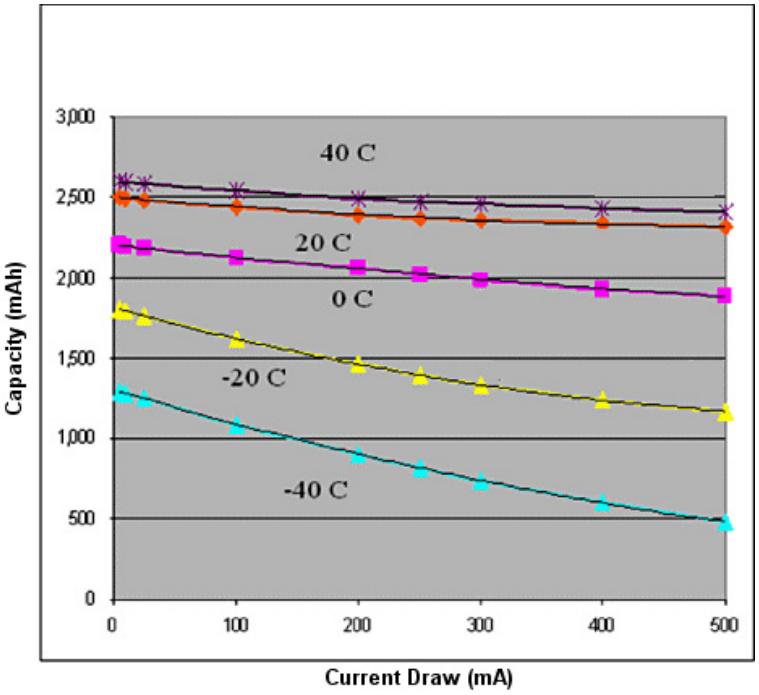


Figure 2.14: Temperature and Discharge Rate Effect on NiMH Batteries [122]

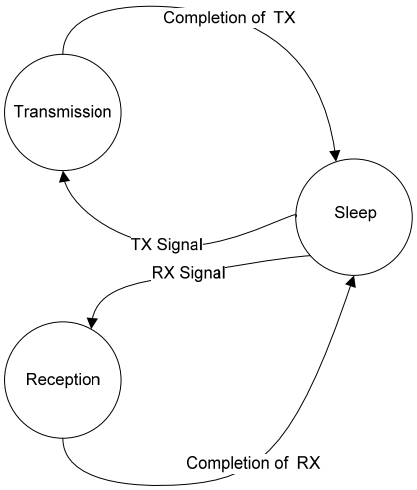
2.2.7 Energy Consumption and Energy Aware Techniques

Energy Consumption

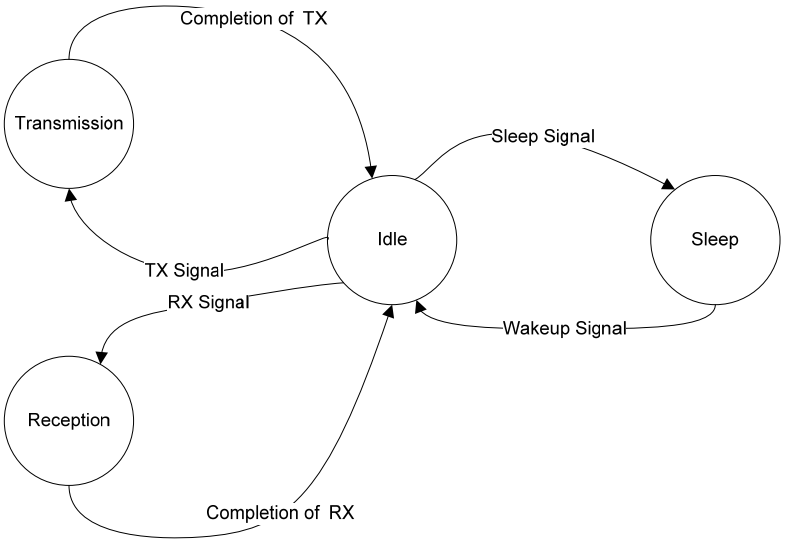
To find the energy consumption in joule, the formula is:

$$E = P * t \tag{2.1}$$

where t is the time spent for running and P is the power which can be calculated by $V * I$. V is the voltage in the node and I is the current consumption. For real measurement, the current consumption (I) and voltage (V) are measured by using the ampmeter and voltmeter (or multimeter). Another technique for energy consumption calculation is summarising the energy consumption of all components. For modern radio transmitters, the energy consumption can be classified into four consumption states: transmission, reception, idle and sleep. The first two states are when a node is transmitting and receiving data packets. In idle mode, a node is waiting for any data transfers and it can transmit and receive. The lowest power consumption state is the sleep state where a node can neither transmit nor receive until it is woken up. Palit et al. [81] proposed the energy consumption model in FSM (Finite State Machine) for wireless communication. CC1000 has only three states and four state transitions, while CC2420 has four states and six state transitions, as shown in Figure 2.15.



(a)



(b)

Figure 2.15: Energy Consumption Model in FSM of CC1000 (a) and CC2420 (b)

$$E_{comm} = \sum_{j=1}^{k} P_j t_j + \sum_{i,j=1, i \neq j}^{k} E_{ij} st_{ij} \tag{2.2}$$

The energy consumption for all states and transitions varies according to the specification of the transmitter model. The mathematical energy consumption of a node is given by formula 2.2 [34]. This formula defines a set of possible states s_1,,s_k. The energy consumption is given by the sum of the consumptions within these states plus the sum of the energy needed for switching between the states. The energy consumption within a state s_j is calculated by using the needed power in that state P_j and time running in that state t_j. The energy needed for switching

from state s_i to s_j is denoted as E_{ij}, while st_{ij} denotes the number of switching time.

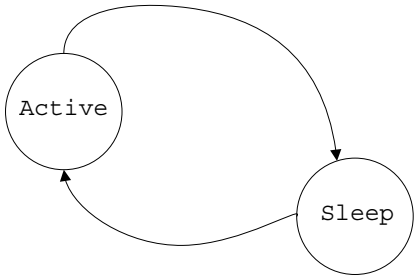


Figure 2.16: Processor States

Normally, a processor has two main states: active and sleep (or power down mode) as shown in Figure 2.16. Some processors may offer more than one power down mode for supporting the energy saving techniques, e.g. Intel 8051 MCU allows three power down modes. The energy consumption for CPU states can be defined as:

$$E_{CPU} = P_a t_a + P_s t_s + E_{as} st_{as} + E_{sa} st_{sa} \tag{2.3}$$

where P_a and P_s are the needed power in active and sleep states, t_a and t_s are time running in those states, E_{as} and st_{as} are the energy needed and the number of switching time from active to sleep states, E_{sa} and st_{sa} are the energy needed and the number of switching time from sleep to active states.

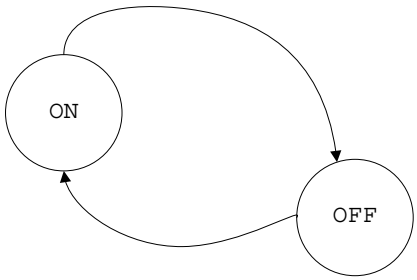


Figure 2.17: Other Devices States

For other devices, there are only two states: on and off as shown in Figure 2.16. The energy consumption can be defined as:

$$E_{oth} = P_{on} t_{on} + P_{off} t_{off} + E_{of} st_{of} + E_{fo} st_{fo} \tag{2.4}$$

where P_{on} and P_{off} are the needed power in on and off states, t_{on} and t_{off} are time running in those states, E_{of} and st_{of} are the energy needed and the number of switching time from on to off states, E_{fo} and st_{fo} are the energy needed

and the number of switching time from off to on states. Then, the total energy consumption can be calculated as:

$$E_{tot} = E_{comm} + E_{CPU} + E_{oth} \tag{2.5}$$

Energy Aware Techniques

Since the sensor network should operate unattended for a long time, its most precious resource is energy. There are two categories of energy aware techniques: Power Management and Power Control [124].

Power Management Energy is consumed when the radio and processor are active. When a sensor does not need to do any activities, its radio transmitter and processor should be in sleep state in order to minimise energy consumption. Therefore, each sensor will switch between sleep mode and "active mode" as shown in Figure 2.18. However, in order to communicate, two sensors should be in "active mode' simultaneously as shown in Figure 2.19. Clock synchronisation is needed for sleep/wake-up schedules.

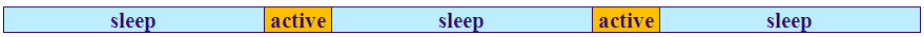


Figure 2.18: Switching Between Sleep and Active Modes

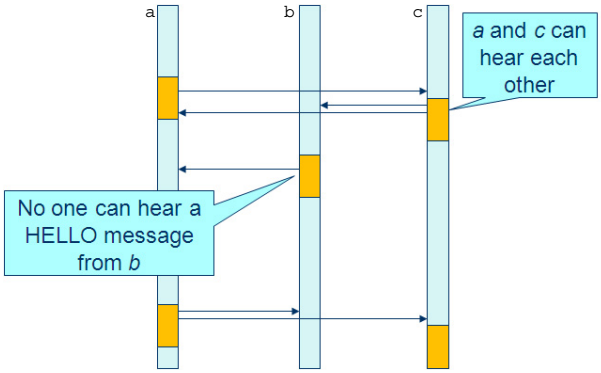


Figure 2.19: Discovering Between Active Sensors

Power Control For the power control techniques, a node should transmit each packet with the minimum power required for successful transmission and reception to each neighbour. From Figure 2.20, if $P(x, y)$ is the minimum power for transmission of a packet from x to y, it is possible that $P(A, B) \neq P(A, C) \neq P(A, D)$.

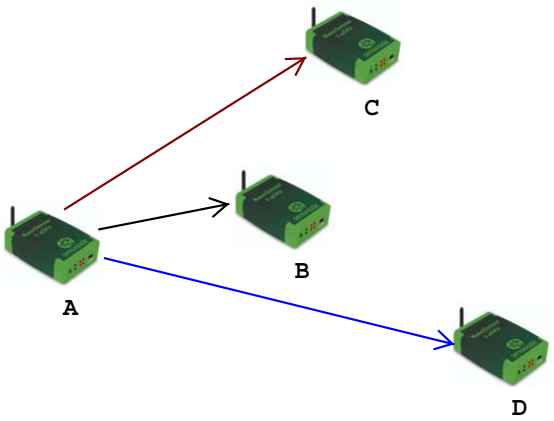


Figure 2.20: Basic Transmission to Neighbours

2.2.8 Node and Network Lifetime

Node Lifetime

The lifetime of the network basically depends on the lifetime of the single nodes that constitute the network. Therefore, the lifetime of the individual nodes should be predicted accurately and the result of the node lifetime estimation model is further used to derive the network lifetime metric. The lifetime (Lt) of a sensor node basically depends on two factors, the capacity of the battery (C) and current consumption needed by that node (I), which is expressed as [18]:

$$Lt = \frac{C}{I^k} \tag{2.6}$$

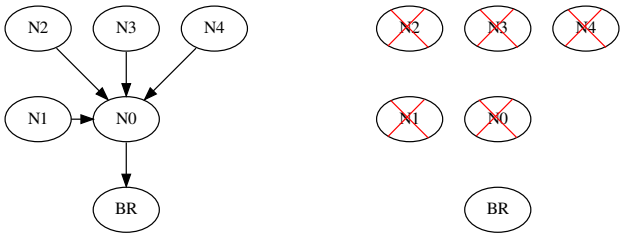
where k is the peukert constant which depends on battery type [2, 3, 11, 13, 18].

Network Lifetime

Network lifetime depends on the lifetime of all nodes in the network. Many network lifetime definitions are discussed as follows.

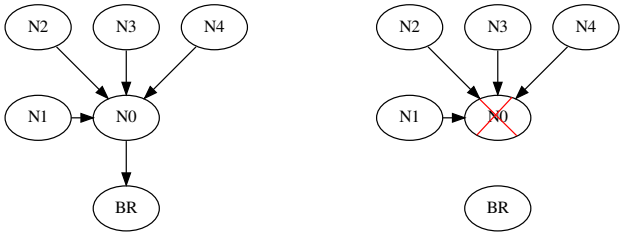
Based on Number of Alive Nodes In the first definition, Tian and Georganas [110] define the network lifetime as the time until all nodes have been drained of their energy, as shown in Figure 2.21. Later, Madan et al. [71] define that the network lifetime ends as soon as the first node fails, as shown in Figure 2.22. Xue and GanzChen [121] define the network lifetime as the time when the amount of dead nodes reaches a specified percentage (k %). When $k = 100$, it is the same as all nodes die.

For the definition of network lifetime as the time until the first node is drained of its energy, it is assumed that the network topology does not change at all.



(a) (b)

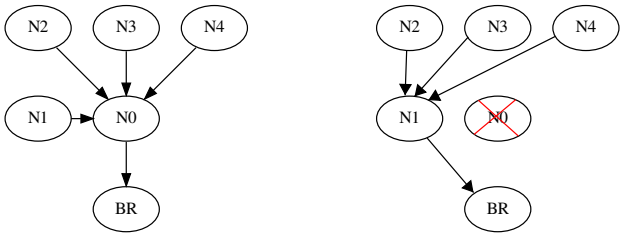
Figure 2.21: Network Fails When All Nodes Die (a) Before Failure, (b) Network Failure



(a) (b)

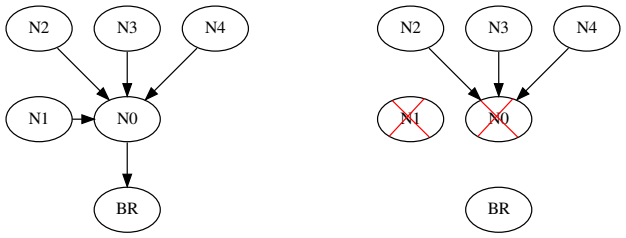
Figure 2.22: Network Fails When First Node Dies (a) Before Failure, (b) Network Failure

However, many routing protocols are able to cope with the failure of one node and all remaining nodes can continue to operate as Figure 2.23. For the definition as the time until all nodes die, it is not optimistic since a sensor network may stop services a long time before the last node finally fails, as shown in Figure 2.24. Therefore, defining network lifetime solely based on the number of alive nodes is insufficient.



(a) (b)

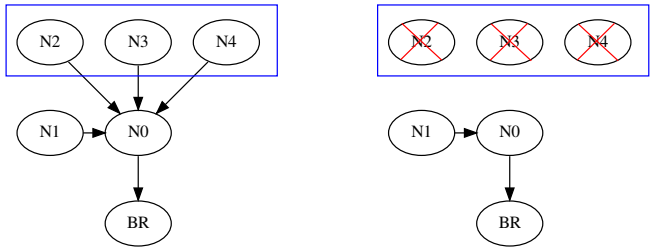
Figure 2.23: Topology Change (a) Before Node Failure, (b) After Node N0 Failure



(a) (b)

Figure 2.24: Network Fails with Some Alive Nodes (a) Before Failure, (b) Network Failure

Based on Sensor Coverage Bhardwaj et al. [21] define the network lifetime as the time that the region of interest is covered by at least one node. For example, the region of interest is covered by N2, N3 and N4. Therefore, the network lifetime ends if these three nodes die as shown in Figure 2.25.

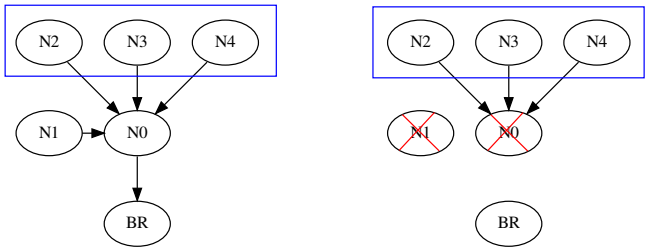


(a) (b)

Figure 2.25: Network Fails When All Nodes in the Coverage Area Dies (a) Before Failure, (b) Network Failure

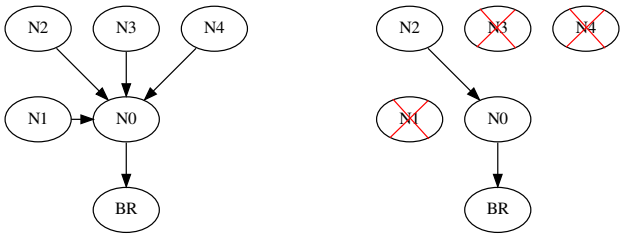
The definition of network lifetime based on coverage is also not sufficient since it is not guaranteed that the measured data can be transmitted to the base station, as shown in Figure 2.26.

Based on Connectivity This definition takes the connectivity of the network into account. A connected node means that the node has the ability to transmit data to the base station. Carbunar et al. [27] define the network lifetime based on the percentage of nodes that have a path to the base station. For example, the network lifetime ends when the percentage of connected nodes is less than 50% as shown in Figure 2.27. Baydere et al. [17] define the network lifetime in terms of the number of packets that could be transmitted to the base station. This number is an indicator of connected nodes.



(a) (b)

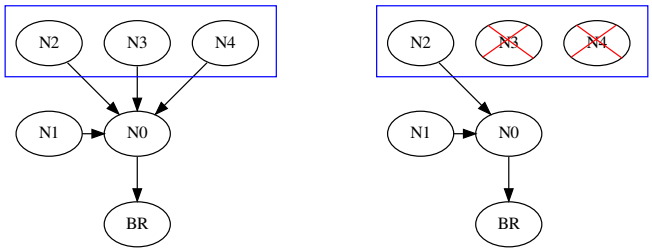
Figure 2.26: Network Fails with Some Alive Nodes in the Coverage Area (a) Before Failure, (b) Network Failure



(a) (b)

Figure 2.27: Network Fails When Number of Connected Nodes $< 50\%$ (a) Before Failure, (b) Network Failure

Based on Sensor Coverage and Connectivity The network lifetime is defined by Cardei and Wu [28] as the time until the number of connectivity nodes in the coverage area drops below a predefined threshold. For example, the network lifetime ends when the percentage of connected nodes in the coverage area is less than 50% as shown in Figure 2.28.



(a) (b)

Figure 2.28: Network Fails When Number of Connected Nodes in the Coverage $<$ 50% (a) Before Failure, (b) Network Failure

The definition of network lifetime based on connectivity is certainly a good idea. However, the network may be considered failed if there are not enough connected nodes in the target region. Therefore, it is necessary to include both the coverage and the connectivity. The connectivity threshold may depend on the application requirements.

2.3 Integrating IP in WSNs

For traditional WSN protocols, it is necessary to provide specific and complex gateways to allow the communication between conventional networks and WSNs. Therefore, one of several challenges and problems in WSNs is integration with the Internet. The integration of IP directly in sensor nodes offers several advantages. For example, it allows any devices connected to the Internet to communicate directly, via a network-layer protocol, with a specific sensor. Moreover, it also offers a transparent and easy approach for developing new generation of applications. However, to integrate IP in WSNs, there are several significant things to be considered. Since a WSN consists of a large number of sensor nodes, IPv6 is required for providing enough addresses to integrate IP networks and WSNs. Moreover, it is necessary to reduce the complexity and the header length of IP because sensor nodes have low speed, low memory, limited processing and small frame size at the data link layer, i.e. the maximum transfer unit (MTU) is only 127 bytes. As a result, IETF proposed a standard, called 6LoWPAN (IPv6 over Low power Wireless Personal Area Network), which includes mechanisms to effectively compress IPv6 addresses over IEEE 802.15.4 [53, 75]. Figure 2.29 shows the IPv6 stack with 6LoWPAN.

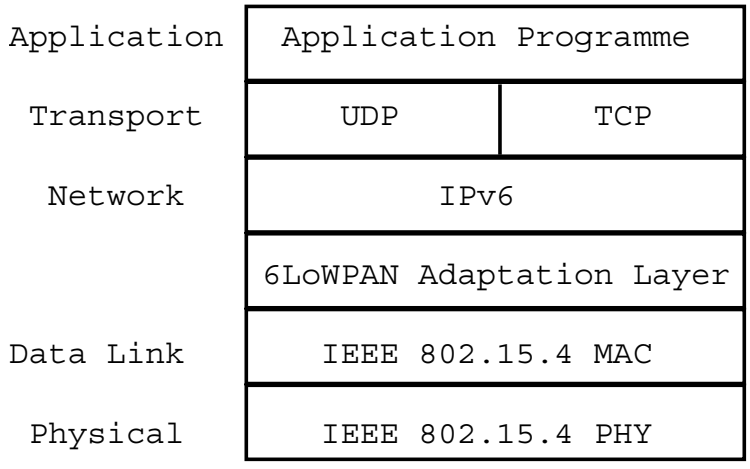


Figure 2.29: Protocol Stack of 6LoWPAN

2.3.1 IEEE 802.15.4

The Wireless Personal Area Network (WPAN) is a short-range network for interconnecting wireless devices centred on a limited personal operating space. The IEEE 802.15.4 standard, developed by the 802.15.4 Task Group, was designed for Low-Rate WPANs [6, 20, 50]. It focuses on low complexity, low power consumption and low data rate wireless communication between low-cost devices which is suitable for WSNs. The data rate of Low-Rate WPAN (LoWPAN), between 20 kbps to 250 kbps, depends on the transmitting frequency as shown in Table 2.4. This standard defines both the physical layer (PHY) and medium access control (MAC) layer specifications.

Table 2.4: IEEE 802.15.4 Features

Frequency and Data rates	868-868.8 MHz and 20Kbps 902-928 MHz and 40 Kbps 2400-2483.5 MHz and 250 Kbps
Range	10-20 m
Addressing	64-bit extended addresses 16-bit short addresses
Network nodes	Up to 2^{64} devices
Security	128 AES
Channel access	CSMA-CA

Physical Layer

The basic tasks of the physical layer (PHY) are summarised as follows [6, 59]:

- Data transmission and reception
- Channel frequency selection
- Activate and deactivate the radio transceiver
- Calculate Link Quality Indication (LQI) for received packets
- Energy Detection (ED) within the current channel
- Clear Channel Assessment (CCA) for Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA)

Spreading and modulation techniques are used in data transmission and reception. Under 802.15.4, nodes can operate in 27 different channels. Thus, all nodes should be able to tune their transceivers into a certain channel. To save energy for some periods, a node should be able to turn its radio off (sleep). Link quality

indication can be calculated by using receiver energy detection (ED), a Signal to Interference plus Noise Ratio (SINR) while receiving a packet. The ED is an estimate of the received signal power within the bandwidth of a current channel. CCA is used for energy detection or carrier sense.

The Physical Protocol Data Unit (PPDU) is shown in Figure 2.30. It consists of three components: SHR (Synchronisation Header), PHR (PHY Header) and PHY Payload. A SHR is composed of a 4 byte preamble of binary zeros and 1 byte for the start-of-packet delimiter (SFD) that notifies the end of the preamble. The PHR contains 7 bits for the frame length and a reserved bit for future use. The PHY payload, or PHY Service Data Unit (PSDU), is for carrying variable length data which must not be more than 127 bytes.

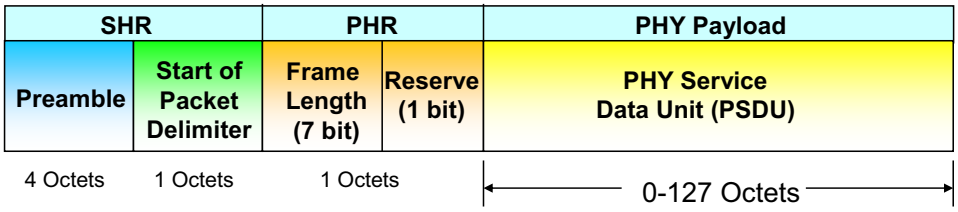


Figure 2.30: IEEE 802.15.4 PPDU Format [50]

MAC Layer

IEEE 802.15.4 defines three types of operating mode: PAN coordinator, coordinator and end-device. It also defines two types of devices: FFDs (Full-function devices) and RFD (Reduced-function devices). An RFD can only operate as an end-device, while a FFD can be a PAN coordinator, a coordinator, or an end-device because it has more powerful resources than a RFD. The MAC layer is responsible for the following tasks [63].

- Generating network beacons if the device is a coordinator
- Synchronising to the beacons
- Supporting PAN association and disassociation
- Supporting device security
- Employing the CSMA-CA mechanism for channel access
- Handling and maintaining the guaranteed time slot (GTS) mechanism
- Providing a reliable link between two peer MAC entities

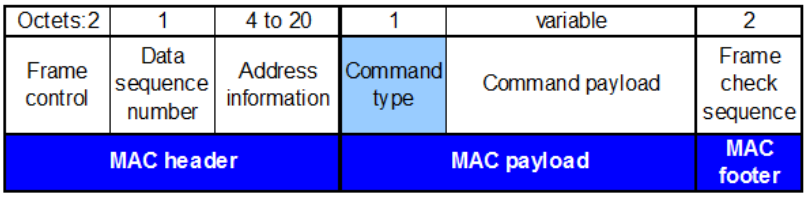


Figure 2.31: IEEE 802.15.4 MAC Frame Format [50]

The MAC frame format is depicted as Figure 2.31. It can be divided into three parts: MAC header, MAC payload and MAC footer. The MAC header is composed of frame control, sequence number and address information like destination PAN identifier, destination address, source PAN identifier and source address. IEEE 802.15.4 supports both short (16 bits) and extended (64 bits) addressing. The MAC payload is for carrying variable length information specific to frame types. There are four frame types: beacon, data, acknowledge and MAC command type. The MAC footer contains two bytes frame check sequence (FSC). The FSC is computed by using the standard generator polynomial of degree 16: $x^{16} + x^{12} + x^5 + 1$

As compared to 802.11, RTS (request-to-send) and CTS (clear-to-send) are not employed in 802.15.4 CSMA-CA mechanism. Moreover, the 802.15.4 MAC layer defines two modes of operation: beacon mode and non-beacon mode. Beacon mode is based on a superframe structure. It is divided into 16 equally sized slots for assigning to each node. The coordinator sends a beacon to synchronise devices, identify the PAN and describe the superframe structure. This beacon is broadcast in the first slot of each superframe. There may be two parts in a superframe. The first one is an active part, called the contention access period (CAP). This part is occupied by using slotted the CSMA-CA method. Another part, called the contention-free period (CFP), is dedicated as guaranteed time slots (GTSs) for high priority applications.

In non-beacon mode, there are no pre-allocated slots. When a node wants to transmit a frame, it has to wait for a random back-off period which is calculated by:

$$CSMA_{Waiting} = N * BP \tag{2.7}$$

where N is random between 0 and $2^{BE} - 1$, BP is the time for transmitting 20 symbols, BE can have a value between the minimum BE (macMinBE, the default value is 3) and the maximum BE (macMaxBE with the default of 5), and a symbol equals 4 bit requiring 16μs of transmission time on 250Kbps channel.

After waiting, it performs CCA to check the status of the channel. Data will be transmitted, if the channel is free. In case the channel is busy, the node increments BE value (not more than macMaxBE) for waiting before retrying. However, the transmission is considered to fail if the number of retries is over the maximum threshold.

2.3.2 IPv6 over Low-power Wireless PAN (6LoWPAN)

The main objective of 6lowPAN, proposed by the IETF, is to integrate IPv6 in LoWPANs supported by IEEE 802.15.4.[98, 103]. Three RFCs, RFC 4919, RFC 4944 and RFC 6282 [53, 61, 75], defined a LoWPAN adaptation layer and frame format, a fragmentation and reassembly mechanism and an addressing scheme, where nodes can generate the IPv6 link local address from the EUI64bits or 16bits MAC address. The Maximum Transmission Unit (MTU) of an IPv6 packet is 1280 bytes, where 40 bytes belong to the packet header. This header would be an enormous overhead for transmitting over IEEE 802.15.4. As a result, header compression is defined as one of the key elements in 6LoWPAN. This feature allows the protocol to compress the 40 bytes of standard IPV6 to just 2 bytes. In the case that the packet size overcomes the IEEE MAC payload, it is necessary to fragment it into several packets and require the reassembly of all packet parts at the receiver side. They also define the support for mesh networks with layer-two forwarding. Figure 2.32 shows these key elements in 6LoWPAN headers.

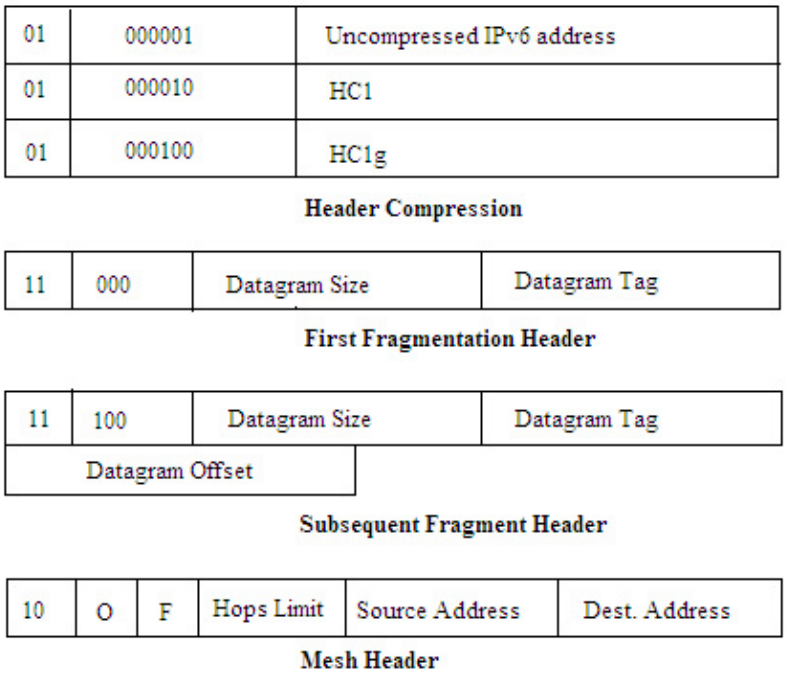


Figure 2.32: Layout of 6LoWPAN Headers

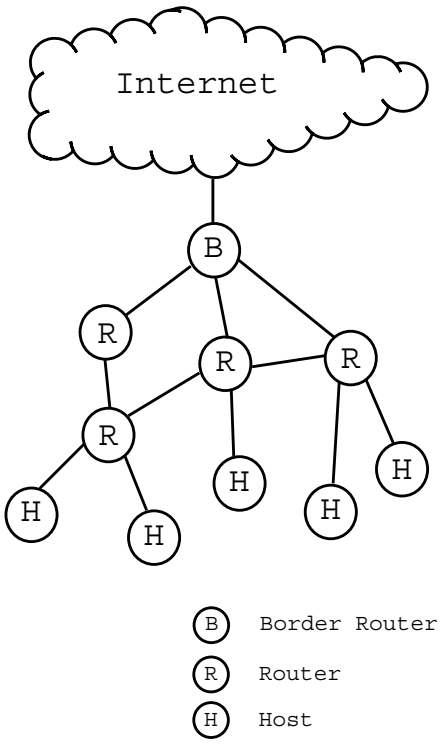


Figure 2.33: Simple 6LoWPAN

A simple 6LoWPAN consists of a border router, routers and hosts. A 6LoWPAN is connected to the Internet through the border router. Other nodes may play the role of router or host (or leaf node). If a node is not adjacent to the border router, one or more routers will relay messages between the border router and that node.

2.3.3 6LoWPAN Neighbour Discovery Protocol

In a large number of sensors, nodes are randomly placed at distributed locations over the area of interest. Self-configuration is very important as it reduces the cost of installation for building large-scale systems. All nodes should organise themselves to efficiently perform the tasks required by the application after they have been deployed. When a node is powered on, it does not have any knowledge about other installed nodes. Thus, in the first step, called Neighbour Discovery (ND), the node has to detect the one-hop neighbours. In 6LoWPAN network, hosts and routers must find the neighbouring routers. Since IEEE 802.15.4 does not support multicast capabilities, the IPv6 ND protocol is not appropriate for WSNs because a packet has to be sent in a broadcast way, which is fundamentally expensive. As a result, 6LoWPAN-ND [99] was introduced with an additional type of node, called edge router or border router. A border router will perform some complex functions in order to reduce the task complexity of hosts and routers. The new 6LoWPAN-ND concept is a whiteboard which can be used for Duplicate

Address Detection. Each node starts sending a Router Solicitation (RS) message and waiting for a Router Announcement (RA) message from the border router. Then, it has to register to the border router by sending a Node Registration (NR) message and the border router responds with a Node Confirmation (NC) message. Figure 2.34 shows basic router discovery and node registration.

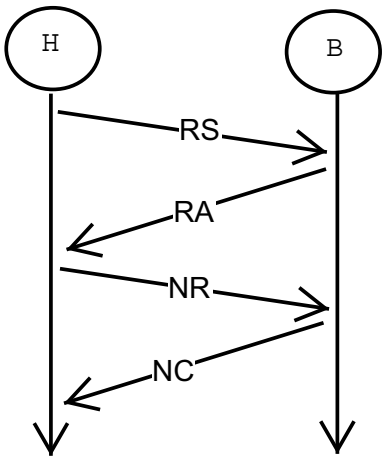


Figure 2.34: Router Discovery and Node Registration in 6LOWPAN-ND

It will be more complicated if the node is not adjacent to a border router. The registration process can be done by multihop registration. The router will relay NR and NC messages between the border router and that node as depicted in Figure 2.35.

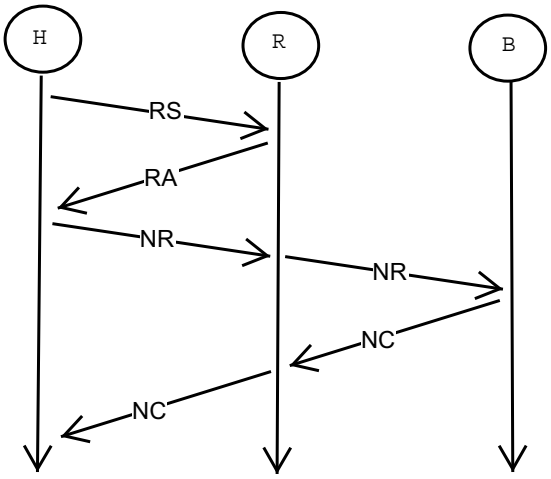


Figure 2.35: Multihop Registration in 6LOWPAN-ND

2.3.4 Routing Protocol for Low-Power and Lossy Networks (RPL)

The IPv6 Routing Protocol for Low-power and lossy networks (RPL) [119] was developed to provide efficient routing paths. The RPL protocol specifies a set of new ICMPv6 (Internet Control Message Protocol version 6) messages to exchange between nodes. Using RPL, a node has to join a Destination Oriented Directed Acyclic Graph (DODAG). In the initial step, the RPL root (or border router) advertises a DODAG Information Object (DIO) message which includes the graph information for selecting DODAG parents. After receiving the DIO message, a node makes a decision to join the graph or not. After joining a graph, the root becomes the parent of the node. If the node is configured as a router, it then advertises the DIO message to its neighbours to form its sub-DODAG. In contrast, it does not send the DIO message, if the node is a host. There are some optional steps after choosing the parent node. For supporting both upward (sensors-to-border router) and downward (border router-to-sensors) routing, a node sends a DAO (DODAG Destination Advertisement Object) message to its parent to inform of its presence and reachability. Then the parent may send a DAO acknowledgement back to that node if an acknowledge bit is set in the DAO message. After that, that node can send and receive messages to and from the border router via its parent. DIO and DAO message exchanges are subsequently used for route maintenance. These RPL steps for the router discovery in a 6LoWPAN network are illustrated in Figure 2.36.

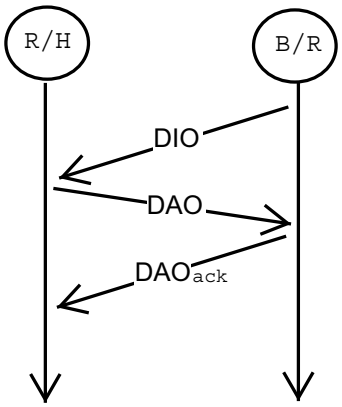


Figure 2.36: RPL for Router Discovery

2.4 Wireless Transmission

In wireless transmission, a sender transmitter sends the electrical signals and then an antenna will transform them into the electromagnetic signals for transferring these signals through space. At the distance of d from the transmitter, an antenna of the receiver detects the electromagnetic signals and transforms them back to the electric signals. The energy loss between the sender transmitter and receiver is the ratio of the transmitted power to the received power. It normally includes all possible elements, such as Line-Of-Sight (LOS) path loss, antenna gain and feeder loss of transmitting and receiving antennas as shown in Figure 2.37.

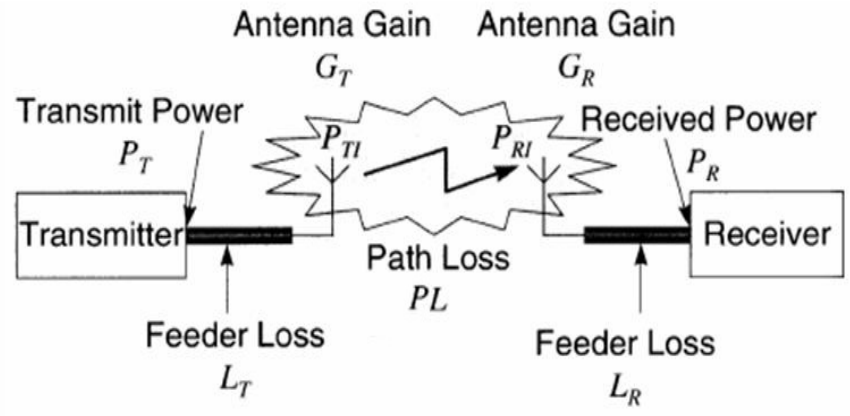


Figure 2.37: Wireless Communication Systems [89]

2.4.1 Antenna

An antenna converts between radio-frequency electrical energy and electromagnetic energy to/from a radio transmitter. It will radiate power in all directions with a common way of the radiation pattern. The simplest pattern is associated with an ideal isotropic antenna which is a point in space that radiates power in all directions equally. The effective area of an antenna is related to the physical size of the antenna and shape and an antenna gain is a measure of the directionality of an antenna. The relationship between effective area (A) and antenna gain (G) is [79, 96, 105, 106]:

$$A = \frac{G\lambda^2}{4\pi} \tag{2.8}$$

There are two classes of antennas: fixed-area and fixed-gain antennas. For fixed-area, the effective area is independent of frequency, while the gain increases quadratically with frequency for fixed-gain antennas, the common type of antenna in WSN. Table 2.5 shows the effective area (A) and antenna gain (G) of some antenna types [79, 96, 105].

Table 2.5: The Effective Area and Gain of Antennas

Type of Antenna	A (m^2)	G
Isotropic	$\frac{\lambda^2}{4\pi}$	1.00
Infinitesimal or Hertzian dipole	$\frac{1.5\lambda^2}{4\pi}$	1.50
Half-wave ($\lambda/2$) dipole	$\frac{1.64\lambda^2}{4\pi}$	1.64
Hertzian monopole ($\lambda/4$) monopole	$\frac{3\lambda^2}{4\pi}$	3.00
Quarter-wave ($\lambda/4$) monopole	$\frac{3.28\lambda^2}{4\pi}$	3.28

2.4.2 Path Loss (PL), Received and Transmitted Power

Path Loss (PL) represents signal level attenuation affected by free-space loss, refraction, diffraction, reflection, aperture-medium coupling loss, and absorption. The received power (P_R) is the difference between the transmitted power (P_T) and PL. The relationship can be defined as:

$$P_R = P_T - PL \tag{2.9}$$

Converting Equation 2.9 to dBm, P_R can be represented as Received Signal Strength (RSS) [106]:

$$RSS = P_T(dBm) - PL(dBm) \tag{2.10}$$

Log-Normal Shadow Model (LNSM)

Log-Normal Shadow Model (LNSM) is a general model which provides many parameters to be configured for both indoor and outdoor environments. The calculation formula is as follows:

$$PL(dBm) = \overline{PL}(d_0) + 10\ \eta\ log(\frac{d}{d_0}) + X_\sigma \tag{2.11}$$

where d_0 and $\overline{PL}(d_0)$ are the reference distance and the RSS of that reference distance measured by the experiment , η is a path loss index depending on propagation environment, X_σ is zero-mean Gaussian random variable.

2.4.3 Received Signal Strength Indicator (RSSI)

The Received Signal Strength Indicator (RSSI) is a measured and estimated value for the RSS provided by modern wireless radio transceivers. For example, for the

Texas Instruments CC2420 RF hardware [7], the RSSI value is averaged over 8 symbol periods (128 μs) and RSSI accuracy is specified as $\pm$ 6 dB. The RSS-RSSI mapping is shown in 2.12.

$$RSS = RSSI + NF \tag{2.12}$$

where NF is the noise floor and usually constant, for example, CC2420 reports as -45 dBm. RSSI is a simple indication of how strong the signal is at the receiver.

2.4.4 Link Quality Indicator (LQI)

Normally, if the received signal strength is very strong, the link is considered to be very good. However, sometimes RSSI might include noise and interference. Therefore, the Link Quality Indicator (LQI) should be considered with RSSI. Generally, RSSI is a measurement of the signal power of an incoming packet, while LQI is a measurement of the quality of a received packet and it is more closely connected to the Signal-to-Noise Ratio (SNR). The SNR is a ratio of received signal power to background noise level. LQI is limited to the range 0 through 255 [6]. The formula of LQI for CC2420 is:

$$LQI = (CORR - a) \cdot b \tag{2.13}$$

where $CORR$ is an average correlation value, a and b are found empirically based on Packet Error Rate (PER) measurements as a function of the correlation value. $CORR$ is the signal-to-noise ratio estimation, for example, CC2420 reports $CORR$ of each incoming packet, based on the first eight symbols following the start frame delimiter (SFD).

2.5 Summary and Discussion

Two common WSN topologies: star (single-hop) and mesh (multihop), have been described. A star topology is simple, but it limits applications to a single hop communication range. For a large coverage area, a mesh topology with multihop routing is needed. Topology is dynamically self-organising for both WSN and MANET. However, there are some differences between these two network types. WSN is a centralised structure, while MANET is a distributed system. Moreover, traffic of WSNs normally targets to base station with data redundancy. In addition, a sensor node has limited resources, such as low memory capacity, low processor capability and low battery power. Therefore, a WSN requires careful resource management.

Three hardware platforms used for simulation and real testbed experiments, Mica2, MicaZ and N740 NanoSersor, have been discussed in this chapter. These platforms are based on the popular microprocessor groups, such as Intel 8051 MCU and Atmel ATmega, with the two most commonly used radio controllers: CC1000 and CC2420. ATmega128L has a single power down state, while 8051 MCU has three power down states. This is an important factor for the duty cycle methods in MAC protocols to save node energy. CC1100 has three radio states: SLEEP, TX and RX, while CC2420 adds a new state, IDLE. This will be useful for radio management. Two operating systems used to develop simulation and real testbed experiments are also described. TinyOS is a component-based operating system written in NesC. It supports many MAC protocols, such as TDMA, TDMA/CSMA, Z-MAC, B-MAC and IEEE 802.15.4. It also supports 6LoWPAN with UDP. CTP and RPL routing protocols are also available in TinyOS. Contiki is a protothreads based operating system using both event-driven and multi-threading techniques. It is implemented in C language. MAC layer protocols supported in Contiki are SMAC, X-MAC, CX-MAC, ContikiMAC, and IEEE 802.15.4. 6LOWPAN implementation is also available in this OS including ICMPv6, UDP, TCP and RPL. This chapter also provides information about WSN simulators.

Neighbour discovery is important for self-organised networks. The ND process in WSNs should be aware of asymmetric links which are very common in WSN. The ND process may be integrated in the routing algorithm, e.g. it is a part of routing protocols to find the best path.

Alkaline and NiMH batteries are the selected battery types in this study. Alkaline has slow self-discharge, but it is not appropriate for using in low temperatures and high current draws, while NiMH has fast self-discharging, particularly on the first day, but it can operate under cool temperature and has a high consumption rate.

This chapter also provides the basic concepts of energy consumption, focusing on communication and processor energy consumption, which require energy and power values and time spent in each state. Energy aware techniques: power management and power control, are described. Power management focuses on switching between sleep and active states, while power control focuses on reducing transmission power. Node lifetime and several network lifetime definitions have been discussed. For the definition of network lifetime, the number of alive nodes is insufficient since network topology may be changed due to routing algorithms. Moreover, it is possible that the network lifetime ends even though there are still some alive nodes. Similarly, the definition of network lifetime based on coverage is also not sufficient since all alive nodes may be unable to send the measured

data to the base station. Although the definition of network lifetime based on connectivity is optimistic, the network can fail if the connected nodes are not in the target region. Therefore, both the coverage and the connectivity should be included in the network lifetime detinition. 6LOWPAN, the IPv6 in WSN, is explained including 6LOWPAN-ND and RPL routing protocol. Finally, the basic concepts and terms of wireless communication are explained. This information will relate to path loss calculation. The equations for RSSI and LQI calculation are also presented.

Chapter 3

Transmission Power Control (TPC)

3.1 Introduction

Transmission Power Control (TPC) is a technique provided by radio transceivers to enable nodes to dynamically control transmission power which results in multiple coverage ranges and multiple transmission energy consumptions. For example, CC1000 [1] and CC2420 [7] allow dynamic change of transmission power during runtime. The relationship between transmission power and current consumption based on data sheet are depicted in Figure 3.1. The current consumption of the largest transmission power levels, which are 5 dBm for CC1000 and 0 dBm for CC2420, will cost 25.4 mA and 17.4 mA, while the smallest transmission power levels (-20 dBm and -25 dBm) will cost 8.6 mA and 8.5 mA. Comparing to the maximum transmission power, the energy consumption of the minimum transmission power levels will be reduced around 66 % and 51 % for CC1000 and CC2420, respectively.

TPC has several advantages [41, 74, 107]. First, it can reduce energy consumption if all nodes send all packets with the minimum power needed for successful transmission. Second, it can reduce the amount of collisions in the network if all nodes transmit packets with the optimal transmission power, because reducing transmission power decreases the coverage range. Last, a reduction of coverage area can lead to the overhearing energy consumption reduction of neighbour nodes. However, unreliable asymmetric links may occur due to reducing transmission power. Therefore, it is challenging to find the ideal transmission power that achieves the communication reliability requirement. The existing TPC algorithms are presented including the discussion sections. The proposed TPC mechanism will be presented in Chapter 6.

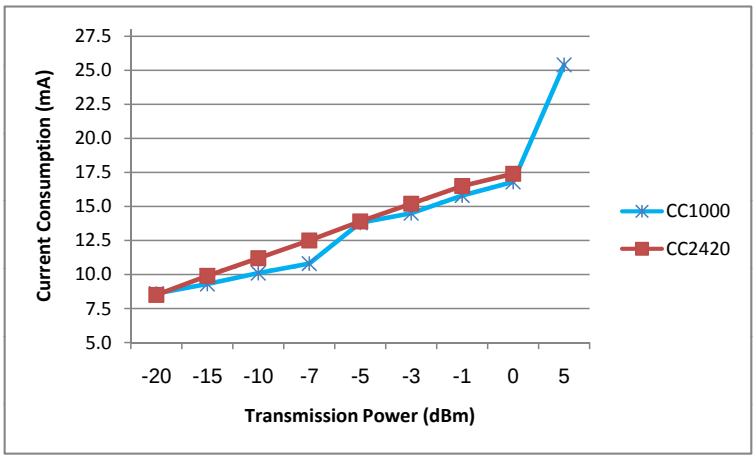


Figure 3.1: The Relationship Between Transmission Power and Current Consumption of CC1000 and CC2420

3.2 Existing Transmission Power Control Algorithms in MANETs

TPC in MANETs has been studied in two different layers: MAC and network. For MAC layer, the measurement scope is usually limited to a single hop. On a network layer, the routes must be composed of energy-efficient links. The adjustment of the transmission power can significantly improve the capacity of the network [42]

3.2.1 MAC Layer

Jung and Vaidya [107] proposed a power control MAC protocol that allowed nodes to identify the ideal transmission power at each individual packet. A power control mechanism was incorporated into the IEEE 802.11 RTS-CTS handshake. This scheme allowed each node to increase or decrease its power level dynamically by maintaining a table for the minimum transmit power necessary to communicate with neighbour nodes. The Power Controlled Multiple Access (PCMA) protocol [74] used two channels; one channel is for busy tones, while the other for all other packets. Instead of RTS-CTS, the busy tones was used to overcome the hidden terminal problem. The signal strength of busy tones received by a node was utilised to determine the highest power level at which this node may transmit without interfering with other on-going transmissions.

3.2.2 Network Layer

Cone-based topology [116] was introduced by Wattenhofer et al. for determining the minimal power consumption in a multihop wireless ad hoc network. Each node performs neighbour discovery by broadcasting a message with minimum power.

It continues the process by increasing its transmission radius until reaching the maximum transmission power. All receiving nodes acknowledge these broadcast messages. Their algorithm assumes that the node can determine the direction of the sender when receiving a message. The objective is to find the best neighbour in every cone of a degree where $\alpha = 2\ \pi\ /3$ as illustrated in Figure 3.2. The best neighbour means the lowest power needed for transmission to that neighbour.

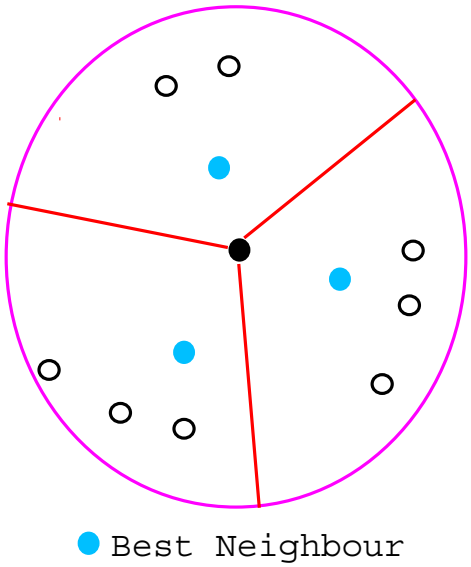


Figure 3.2: Best Neighbour in Every Cone

Since WiFi and WiMax devices adjust the data rate of a link according to the distance between receiver and transmitter or environment noise and interference, Macedo et. al. [70] proposed the modification of routing protocols to cope with TPC and Rate Adaptation (RA). The modifications could be applied to either proactive or reactive protocols that employ cost functions. This method is based on:

$$P_{Tmin} = (N + I) \cdot SINR_m \cdot PL \tag{3.1}$$

where $N + I$ is the signal and interference at the receiver, $SINR_m$ is the $SINR$ for the rate m and PL is the path loss.

3.2.3 Discussion

Existing MAC-level methods for the calculation of the transmission power and modulation are applied to one link at a time which is not suffient for multihop network. TPC-aware routing protocols to reduce the energy consumption of MANETs have to execute several instances of a routing algorithm for each available transmission power. The route chosen to forward data is the one which employs the smallest transmission power level. This strategy demands a high amount of energy

due to the execution of several instances of the routing algorithm. Further, the integration of RA and TPC is not neccessary in WNSs since the fixed data-rate on individual links is normally assumed.

3.3 Existing Transmission Power Control Algorithms in WSNs

In WSNs, the main objectives of transmission power control (TPC) can be divided into two categories. The first is to find the ideal transmission power for transmission energy consumption reduction. The second is to dynamically adjust transmission power to cope with focused factors, such as noise and signal fluctuation, number of neighbours and packet reception ratio.

3.3.1 Finding the Ideal Power

In the following, the term "ideal" is defined as the power level which is between the minimum and maximum allowable transmission powers for successfully transmitting messages from a node to another. In this study, the ideal transmission power focuses on the energy-efficient way which is the effort made to reduce the transmission power and still achieve the communication reliability requirement. Two methods have been carried out to find the ideal transmission power before starting data packet transmission. These two existing approaches are distance-based and scanning-based.

Distance-based

The distance-based technique assumes that distances between the senders and receivers or node positions are provided (either by the special device, GPS, or the neighbour discovery process). The research works focused on the path loss model which is a function of distance. Rodoplu and Meng [91] pointed out a simple radio propagation model for transmitting power roll-off as given by the path loss model. If the received power falls as $\frac{1}{d^n}$ then relaying information between nodes may be more energy efficient than transmission directly over long distances, where d is the distance and distance factor $n \geq 2$. As an illustration in Figure 3.3, it is assumed that there are three collinear nodes A, B, and C using the same radio transmitter, and all nodes can communicate with each other. If node A wants to send a message to C, A has two options for sending: transmitting the message directly to C, or relaying through B. The energy consumption for the first option is td_{AC}^n, where t denotes the pre-detection threshold (in mW) at each receiver. For

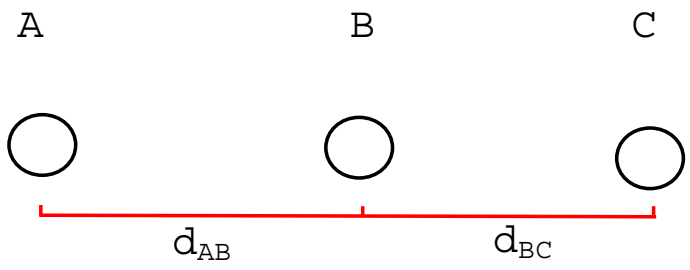


Figure 3.3: Three Collinear Nodes within Reach of Each Other

the other option, it requires the power $td^n_{AB} + td^n_{BC} + Rx_B + Pr_B$ where Rx_B and Pr_B are receiving and processing power at the relay node B, respectively. It is possible that sending the message from A through the relay node B may result in lower total power consumption than transmitting directly to C. Their research also proposed a position-based algorithm to set up a minimum energy network when the three nodes lie within a two dimensional plane, which has motivated other studies to explore transmission power control based on distances.

Ramanathan and Rosales-Hain [88] proposed an equation for finding the new transmit power p_d within the desired distance as:

$$p_d = p_c - 5 \cdot \varepsilon \cdot log\left(\frac{d_d}{d_c}\right) \tag{3.2}$$

where p_c and d_c denote the current transmit power of a node and the current distance in a network of density, while ε and d_d are the environment condition and the desired distance.

Rappaport [90] revised the equation for the optimal power p_i required by node i to transmit data to node j as:

$$\frac{p_i}{\delta^{\alpha}_{i,j}} \geq \beta \tag{3.3}$$

where α is the distance-power gradient, β is the transmission quality parameter, and $\delta_{i,j}$ is the distance between these two nodes. Normally, the value of β is set to 1 and the value of α depends on environmental conditions (≥ 2). Blumenthal et al. [23] proposed a new method to calculate the distance between a sender and a receiver by using the minimal transmission power based on Equation 3.3. Xu [120] proposed another technique to calculate the distance based on the Log-Normal Shadowing Model (LNSM). Receive Signal Strength (RSS) can be calculated based on LNSM as:

$$RSS(dBm) = RSS(d_0) + 10\ \eta\ log(\frac{d_0}{d}) + X_\sigma \tag{3.4}$$

where d_0 and $RSS(d_0)$ are the reference distance and the RSS of that reference distance measuring by the experiment, η is a path loss index depending on propaga-

tion environment and X_σ is zero-mean Gaussian random variable. For LOS of indoor environment, η is around 1.6 to 1.8 and X_σ is often used as 10.5 dBm [100, 37].

Scanning-based

Scanning-based approach of ND process for finding the optimal transmission power. Each node performs neighbour discovery by broadcasting a message with minimum power. It continues the process by increasing its transmission radius until reaching the maximum transmission power. All receiving nodes acknowledge these broadcast messages. Then, a node can record the minimal transmission power for transmission to each neighbour as illustrated in Figure 3.4.

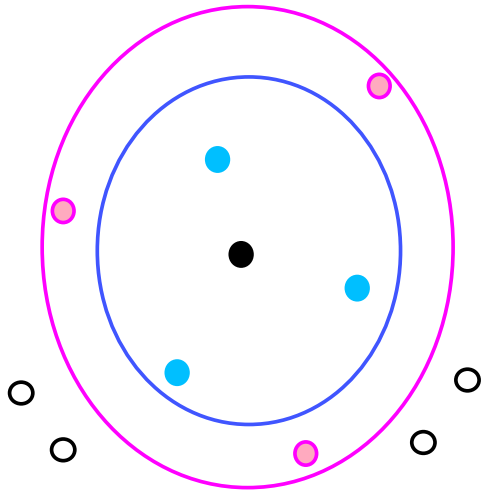


Figure 3.4: Scanning with Different Transmission Power Levels

Lin et al. [68] proposed a technique called Adaptive Transmission Power Control (ATPC). In this technique, every node has to broadcast messages with different transmission power levels in the initialisation phase until the ideal transmission power is found for transmission to the target node. This means that it might send only one broadcast message for the best case, or have to send l broadcast messages if there are l transmission power levels for the worst case.

Discussion

For the distance-based approach, Received Signal Strength (RSS) can be predicted, which leads to optimal power prediction. However, the distances between nodes are difficult to determine in real deployment. For example, each node may need a preconfigured position or an attached positioning device like GPS (Global Positioning System). Moreover, in reality it is not always true that the path loss

increases if distance increases for some cases, such as in multipath or obstruction environment [68, 93, 95]. Therefore, distance might be not appropriate for estimating the transmission power.

For a scanning-based approach, the ideal power can be obtained without RSS estimation. This ideal power can cover many effects including multipath and obstruction. However, using this technique may prolong the discovery time since a node may have to broadcast with all transmission power levels during the discovery phase. This leads to an increase of the energy consumption for the discovery process. Furthermore, if the process of neighbour discovery finishes late, it will cause a delayed start for other processes.

3.3.2 Dynamic Transmission Power Adjustment

For dynamic adjustment, it is assumed that the optimal transmission power is already known (e.g. by using pre-defined transmission power, distance or scanning based techniques). RSSI, LQI, Packet Reception Rate, number of neighbours and temperature are the factors focused on by the existing approaches.

RSSI Feedback-based

All nodes start sending with the current optimal power and they adjust the radio transmission power depending on RSSI values and two threshold values: upper and lower thresholds. If the RSSI values are higher than the upper threshold, the transmission power is decreased. In contrast, the transmission power is increased if the RSSI values are lower than the lower threshold. The receiver has to monitor the RSSI and send it to the sender. An example of this approach is topology control of multihop wireless networks using transmit power adjustment [88].

LQI Feedback-based

Similar to RSSI feedback-based, it is assumed that the optimal transmission power is already known. Instead of using RSSI, this technique uses LQI for transmission power adjustment. If LQI values are higher than the upper threshold, the transmission power is decreased, while it is increased if LQI values are lower than the lower threshold. An example of this approach is the second phase of ATPC [68]. In ATPC, the LQI is monitored by the receiver. When the link quality falls below the desired level, a notification packet is sent from the receiver to the sender for increasing the transmission power.

Packet Reception Rate (PRR)

Son et al. [104] used the Packet Reception Rate (PRR) for transmission power adjustment. The links are divided into two types: unreliable and good links. The communication link between two nodes is a good link if the PRR in both directions is higher than the PRR threshold value. Otherwise, it is an unreliable link. The purpose of this technique is to convert unreliable links into good links by increasing the transmission power. The transmission power for each link can be reduced to the lowest transmission power if the link is still good. The quality of links is determined by the receivers and then sent back to the senders.

Number of Nieghbours

Blough et al. [22] proposed the algorithm based on the principle of maintaining the number of neighbours. This algorithm adjusted the transmission power for each node to connect a specific number of neighbouring nodes. This specific number is a threshold value which is used for determining the target transmission power for sensors. If the number of neighbours is below the threshold, the node will increase the transmission range until it detects the proper number of neighbours. Figure 3.5 describes the step of this algorithm. Gerharz et al. [39] included the second phase of operations by setting the critical value. The transmission range will be decreased if the number of neighbours is higher than the critical value. This is to maintain the minimum number of neighbour nodes.

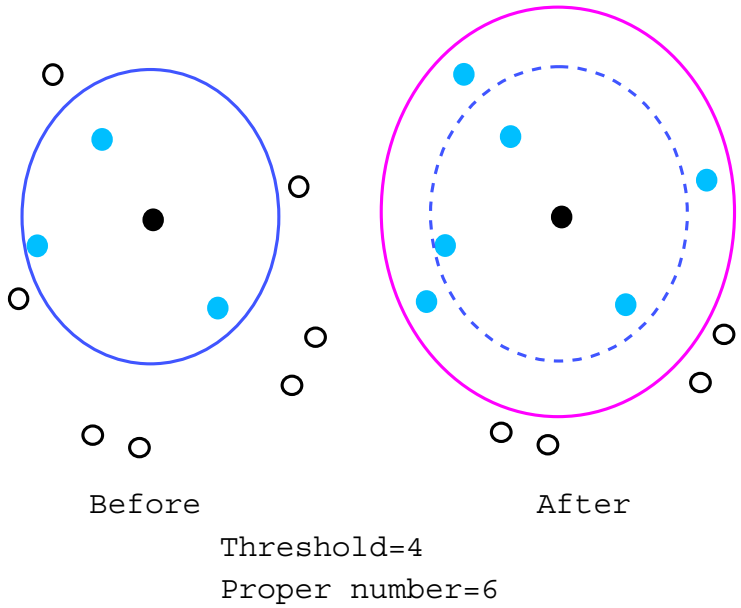


Figure 3.5: Maintenance the Number of Neighbours

Temperature

Lin et al. [68] observed that the daily variation of RSS is around 6 dB, which may be caused by temperature, humidity, and other factors. Bannister et al. [16] studied the relationship between RSS and temperature in the Sonoran Desert of the Southwestern United States, where daily summer time temperatures may vary from 25 to 65°C. They found that RSS values tend to decrease when the temperature increases, which results in link quality and connectivity reduction between nodes. To compensate for the signal strength dropping due to the temperature rising, they proposed this equation:

$$P_r - T_L(T) \geq P_s \tag{3.5}$$

where P_r is the received power measured at the receiver, P_s is the radio sensitivity, and $T_L(T)$ is the temperature effect at T°C which is calculated as:

$$T_L(T) = 0.1996 \cdot (T - 25) \tag{3.6}$$

where T is the temperature between 25 and 65°C. Lee and Chung [64] extended the equation by defining $T_L(T)$ as the $RSSI_{loss}$. Their study focused only CC2420 radio transceiver and proposed an equation for transmission power adjustment as:

$$P_{level} = \left(\frac{P_{out} + 40}{12}\right)^{2.91} \tag{3.7}$$

where P_{leve} and P_{out} are the corresponding power level and the output power (dBm) of CC2420.

Discussion

The CC1000 radio unit provides RSSI, while the latest model, such as CC2420, provides the additional metric, LQI. Therefore, several approaches used RSSI and LQI to determine the link quality and adjust the transmission power. Moreover, PRR may be used for indicating link quality and transmission power adjustment. In these existing approaches, link monitoring has to be conducted as the interesting factors may change over time. A feedback or acknowledgement system is used after the receivers have successfully received the messages. As a result, this requires more power for computation and additional data delivery for an issue of link monitoring. A sensor is likely to face power depletion in the case of more frequent monitoring. Some works used the number of neighbours as a factor to adjust the transmission power to keep the number of neighbours within a desired range. This scheme focused on multihop communication as several intermediate nodes

are required for relaying or forwarding a packet to the base station. It might not be suitable for several requirements of some WSNs applications which need only a single-hop environment. Using temperature for transmission power adjustment is very interesting. However, existing works proposed formulas for only specific radio transceivers, such as CC2420. Extending to design for a generic radio transceiver will be useful.

3.4 Summary and Discussion

TPC is a technique for transmitting data efficiently over wireless channels with the minimum transmission power for maintaining reliability. This can reduce the interference among the nodes or the overhearing energy consumption. In MANETs, TPC-aware algorithms have been proposed in both MAC and network layers. However, they focused on IEEE 802.11. For WSNs, the objective of existing schemes can be categorised into two groups: finding the ideal power and dynamic transmission power adjustment. To find the ideal power, distance-based and scanning-based schemes were used by several approaches. For the distance-based approach, both RSS and optimal power can be predicted, but it is difficult and may not be suitable for implementation in real deployment. For the scanning-based approach, the optimal power can be obtained but it will prolong the discovery time and lead to an increase in the energy consumption. RSSI, LQI and PRR were used to provide transmission power adjustment mechanisms based on feedback from the receiver. However, this requires more power for computation and additional data delivery. Keeping the number of neighbours is one of the techniques for transmission power adjustment, but it is not appropriate for single-hop networks. Some works used temperature to adjust the transmission power, but only a specific radio transceiver is considered. Therefore, proposing a new generic technique for finding the optimal transmission power including transmission power adjustment, which supports different requirements for both single-hop and multihop communications, is a research opportunity for WSNs.

Chapter 4

Node Lifetime Estimation in WSNs

4.1 Introduction

WSNs have two critical constraints: the first is that sensor nodes are often battery powered and thus have limited energy budgets; the second is that sensor nodes have a large number of nodes and are usually deployed unattended, causing difficulty when replacing recharging batteries across the entire network. Therefore, network lifetime (i.e. time when the network is usefully working) is an important issue. With accurate lifetime estimation of the sensor nodes, application designers can prevent service interruptions for critical applications. Moreover, many protocol layers, such as the MAC and routing layers, are able to make intelligent decisions that can help conserve energy and prolong lifetime. Two important factors, battery capacity and current consumption, are used for node lifetime estimation. In many studies [33, 77, 93, 95], quoted capacity and calculations based on data sheets are usually used for battery capacity and measurements made for current consumption estimation. Furthermore, a fixed temperature is normally assumed (e.g. 25 °C). This means that running a static load programme multiple times always consumes the same energy. However, in reality programmes will have different energy requirements resulting in different lifetimes [94]. In addition, the non-linear behaviour of the batteries needs to be taken into account. The existing battery capacity and current consumption estimation models are presented including the discussion sections. The proposed node lifetime estimation technique will be presented in Chapter 7.

4.2 Existing Lifetime Estimation Models

Based on Equation 2.6, it is therefore necessary to estimate both battery capacity and current consumption as well as have a knowledge of the battery type to determine an estimate of node lifetime. Existing capacity, current consumption and lifetime estimation methods are discussed.

4.2.1 Battery Capacity Estimation

There are several methods for determining the State of Charge (SoC) of the battery. This research focuses the methods of electrochemical, voltage measurement, load testing and the electromotive force, which can be applied for alkaline and NiMH battery capacity estimation.

Electrochemical method

Since chemical energy in battery cells is converted into electrical energy through an electrochemical reaction, many studies [123, 76] propose electrochemical models based on the chemical processes that take place in the battery. These models describe the battery processes in great detail. However, they are very complex and require highly detailed knowledge of the electrochemical process, which makes them difficult to configure and deploy. An easy and accurate way is provided by measuring the specific gravity of the electrolyte in the battery using a hydrometer [18]. The specific gravity varies according to SoC level. When the SoC level decreases, the density of the electrolyte becomes lighter and the specific gravity becomes lower. An example of the relationship between the specific gravity and SoC is shown in Table 4.1 [18].

Table 4.1: The Relationship Between the Specific Gravity and SoC [18]

Specific Gravity	SoC (%)
1.265	100
1.239	75
1.200	50
1.170	25
<1.110	0

Voltage Measurement

Voltage measurement is a popular method for estimating current capacity, especially for mobile phone applications. For example, Heyer [49, 51] introduced a

single-meter device for indicating the battery capacity on the basis of the measured battery voltage as shown in Figure 4.1. This technique requires a look-up table in which fixed voltage values are stored and used in order to indicate SoC. For example, table 4.2 provides the relationship between voltage and SoC for Energizer NiMH battery [11].

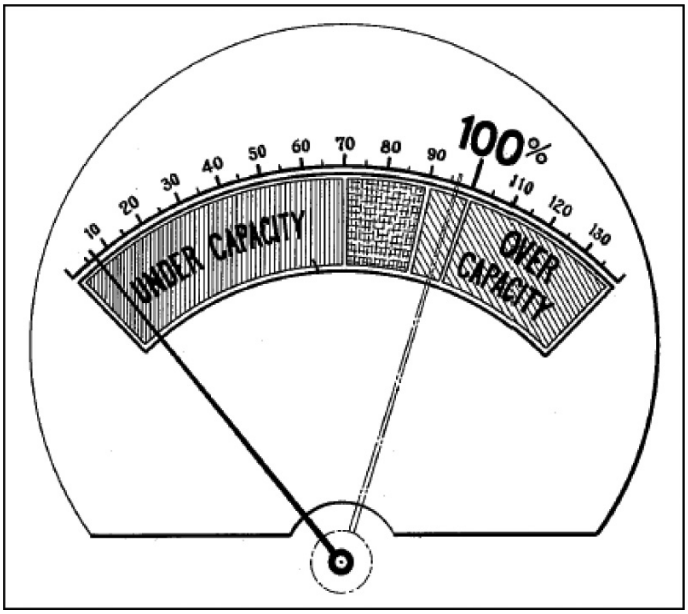


Figure 4.1: Battery Capacity Indicator Developed by Heyer[51]

Table 4.2: The Relationship Between Output Voltage and SoC [11]

Voltage (V)	SoC (%)
1.45	100
1.34	90
1.30	80
1.28	50-70
1.27	40
1.25	30
1.22	20
1.14	10
<1	0

The ElectroMotive Force (EMF)

The EMF is the internal driving force of a battery, providing energy to a load. Many studies [18, 45] found that there is a good linear relationship between the EMF and the SoC and this relationship does not change during cycling of the battery. To estimate SoC based on the EMF, a piecewise linear function is required. The intervals in voltage and the corresponding SoC are presented in Table 4.3 [18]:

$$SoC = SoC_{low} + \frac{V_m - V_{low}}{V_{high} - V_{low}} (SoC_{high} - SoC_{low}) \quad (4.1)$$

where V_m is the measured battery voltage value, V_{low} and V_{high} are the specific values from the EMF curve for the voltages corresponding to the SoC_{low} and SoC_{high}, e.g. in Table 4.3, $V_{low} = 4.08$ and $V_{high} = 4.24$ corresponding to $SoC_{low} = 85$ and $SoC_{high} = 100$, respectively. Therefore, the capacity of battery for 4.10 volts is $\approx 87\,\%$ of the maximum capacity.

Table 4.3: The Intervals in Voltage and the Corresponding SoC for Sony US18500G3 Li-ion Battery [18]

Interval number	Interval Voltage(V)	SoC(%)
1	4.08-4.24	85.0-100
2	4.06-4.08	81.7-85.0
3	4.02-4.06	76.7-81.7
4	3.98-4.02	73.4-76.7
5	3.88-3.98	58.4-73.4
6	3.80-3.88	22.0-58.4
7	3.68-3.80	8.7-22.0
8	3.54-3.68	5.4-8.7
9	3.32-3.54	2.1-5.4
10	3.00-3.32	0.5-2.1
11	2.50-3.00	0.0-0.5

Load Testing

Battery capacity drops due to many factors, such as ageing and life cycle. Battery capacity testing by a load tester serves to determine the actual capacity of the battery. The load is usually designed to represent the expected conditions in which the battery may be used. With a battery load tester, a specific discharge current is applied to the battery while measuring the voltage drop. This is the most accurate and reliable battery testing technique [49]. Many battery load tester products support both alkaline and NiMH batteries, such as the ZTS MBT-MIL Multi-Battery Tester [54] and Ansmann Energy Check LCD Battery Tester [15]. However, these products automatically initiate a timed pulse load test on the battery upon detection in a terminal. This load test cannot be modified. In addition, these testers do not provide information related to temperature. The Computerised Battery Analyser-III (CBA-III) [87], a product from the West Mountain Radio, is a computer calibrated for high accuracy which uses an on-board microcontroller. A pulse width modulation system is used for controlling a pair of power MOS FET transistors using both electronic and software current regulation. The

CBA-III allows for defining the load test from 0.1 A to 40 A. Moreover, it provides information about the total amount of energy stored in a battery (capacity in amp-hours) and graphically displays and charts voltage versus amp-hours. Furthermore, a CBA-III supports the temperature measurement of a battery under test using the external temperature probe. In addition, a computer can connect to CBA-III via the USB interface on-board microcontroller for collecting the data. Figure 4.2 shows CBA-III device and accessories.

Figure 4.2: Computerized Battery Analyzer (CBA-III) [87]

Temperature Effect

Based on the famous Arrhenius equation, Sheridan et al. [101] proposed the ratio of capacity at two different temperatures in Kelvin (T_1, T_2) as:

$$\frac{C(T_2)}{C(T_1)} = \exp\left(B \cdot \frac{(T_2 - T_1)}{T_2 \cdot T_1}\right) \tag{4.2}$$

where B is a constant value obtained by dividing the activation energy (in $\mathrm{J\,mol^{-1}}$) with the gas constant (in $\mathrm{J\,mol^{-1}\,K^{-1}}$) of the battery. The relationship between battery capacity and lifetime was studied experimentally by Nguyen et al. [77] for different alkaline battery brands. The effect of temperature on NiMH battery capacity with different commercial AA-size NiMH batteries (Duracell, Energizer and GP) was explored by Pierozynski [85]. His work reported that the battery capacity is almost equal 96 % at room temperature and it is dropped to 74 % at $-20\,^{\circ}\mathrm{C}$ and 58 % at $-30\,^{\circ}\mathrm{C}$. The model proposed by Park et al. [25] covered the remaining capacity and the effect of temperature on battery capacity. From their

experiment with rising temperature of 20 °C to 60 °C, it was observed that the battery capacity increases around 0.5 % of SoC when temperature increases by 1 °C. With decreasing temperature from 20 °C to −10 °C, the battery capacity decreases 1.7 % of SoC when temperature decreases 1 °C.

Discussion

Many SoC estimation techniques have been explored. Electrochemical methods require highly specific chemical knowledge which makes it difficult for real hardware implementation. Measuring the specific gravity is the uncomplicated way, but it is difficult to apply to sealed batteries [114, 49]. Voltage measurement is an easy and popular method. However, this measurement technique is not always an accurate indicator [114, 49]. Moreover, it is impossible to take into account every point of voltage in order to provide an accurate SoC indication system. The EMF method is more accurately implementable for many battery types. However, other factors, such as temperature and ageing, must be considered. Load testing is an accurate and reliable battery capacity estimation technique but a load test equipment is required. The temperature effect proposed by Sheridan et al. [101] is a generic model which can be applied for many battery types but it is difficult to find detailed information from a vendor's specifications, such as the activation energy and gas of the battery. The model proposed by Park et al. [25] covered the remaining capacity and the effect of temperature on battery capacity. However, they assumed the battery began with full capacity voltage, which might not apply for NiMH batteries because this battery type has a relatively high self-discharge rate on the first day after full charging. As a result, extending to cover temperature, ageing, self-discharge and other impact factors will be useful.

4.2.2 Current Consumption Estimation

There are several methods for determining the current consumption of sensor nodes. The existing mechanisms are hardware-based and software-based, including temperature effect and supply voltage changes.

Hardware-based Mechanism

To estimate node current consumption, many current consumption models [66, 117] tried to design the special circuits for finding the average leakage current consumption; however, detailed knowledge of electrical circuits is required. Oscilloscopes or ammeters are commercially available for measuring the current draw from the circuit of sensor nodes with accurate results. Since the unique characteristics of sensor network applications make it difficult to measure the power and

current consumption of each sensor node, Jiang et al. [56] developed hardware-based integrated circuits attached to sensor node boards for measuring the current consumption. This mechanism can capture phenomena such as per-node fluctuations.

Software-based Mechanism

Dunkels et al. [33] proposed a formula to calculate the energy consumption (E).

$$\frac{E}{V} = I_a t_a + I_l t_l + I_t t_t + I_r t_r + \sum I_c t_c \tag{4.3}$$

where V is the supply voltage, I_a and t_a are the current draw of the MCU (Microprocessor Control Unit) and time when the MCU has been running in active mode; I_l and t_l are the current draw and time of the MCU in low power or sleep mode; I_t and t_t are the current draw and the time of the communication device in transmit mode; I_r and t_r are the current draw and time of the communication device in receive mode; I_c and t_c are the current draw and time of other components such as sensors and LEDs. Then, the node current consumption (I) rate is equal to $\frac{E}{V \cdot t}$ where t is the time period. In this model, all current draw, supply voltage and temperature are assumed as fixed values (e.g. 3 V and 25 °C as defined in the data sheet).

Temperature and Supply Voltage Effect

The current consumption of electronic circuit is affected by both temperature and supply voltage [77]. As temperature increases, current draw is increased. Liao et al. [67] proposed a leakage current model with temperature as:

$$I(T) = I_s \cdot \exp\left(\frac{-\alpha}{T - \beta}\right) \tag{4.4}$$

where T is the temperature in Kelvin, I_s is a constant current value, α and β are the empirical constants which are decided by the circuit designs. Later, supply voltage is considered for leakage calculation as [66]:

$$I(T, V_{dd}) = I_s(T_0, V_0) \cdot T^2 \cdot \exp\left(\frac{\alpha \cdot V_{dd} + \beta}{T}\right) \tag{4.5}$$

where T is the temperature in Kelvin, V_{dd} is the supply voltage, I_s is a constant current at the reference temperature T_0 and supply voltage V_0, α and β are the empirical constants which are decided by the circuit designs. It is assumed that the constant current (I_s) is already known.

Discussion

Several mechanisms are used for finding the current draw of sensor nodes. Although a hardware-based mechanism can provide accurate results, it is of significantly high cost and complexity for large scale usage. Moreover, it is difficult to add to the existing hardware. In contrast, a software-based mechanism is easy to add to an existing system without additional per-unit cost. However, fixed voltage and current draw values are assumed for current consumption calculation using a software-based technique, which may not be accurate in real-world deployments as the current draw is usually dynamic, based on temperature and supply voltage. Temperature and supply-voltage-effect formulas focus only on the effects of the leakage current draw. They do not provide methods for finding the constant current draw. Therefore, it is interesting to add the temperature effect to the software-based technique.

4.2.3 Lifetime Estimation

The review of some lifetime estimation techniques for real world WSNs is presented as follows:

Based on Voltage Drop Rate

Hao et al. [46] proposed a technique for lifetime estimation as:

$$Lt = \frac{V_{init} - V_{cut}}{V_{rate}} \tag{4.6}$$

where V_{init} is the initial voltage of battery, V_{cut} is the cut-off voltage and V_{rate} is the voltage drop rate. In their experiment, it is observed that the average voltage drop rates are 16.5mV/day for Mica2 and 21.5mV/day for MicaZ. With an initial voltage level of 3.2V and the cut-off voltage of 2.2V, the estimated lifetimes are 60.6 d for Mica2 and 46.5 d for MicaZ.

Based on Quoted Capacity and Current Consumption

Selvig [97] presented a method to estimate the lifetime for sensor nodes based on CC2430 as:

$$Lt = \frac{C}{\overline{I}} \tag{4.7}$$

where C is the battery capacity in mA h and $\overline{I}$ is the average current consumption. In this experiment, the battery capacity is assumed to be the capacity quoted by the vendor and the average current consumption is measured by an oscilloscope.

Since I=$\frac{E}{V}$, the software-based technique can be applied for finding the average current consumption by using Equation 4.3.

Based on Quoted Capacity and Energy Consumption

Landsiedel and Wehrle [62], proposed an energy monitoring model, called AEON, for a sensor node as:

$$E_{rem} = E_{current} - E \tag{4.8}$$

where $E_{current}$ is the current battery capacity in J, E_{rem} is the remaining battery capacity in J and E is the energy consumption which can be obtained by using Equation 4.3. The initial value of $E_{current}$ is the initial capacity of battery in J. The battery capacity is also assumed to be the capacity quoted by the vendor. For example, the initial capacity of alkaline battery with 2500 mA h is 27 000 J. The lifetime of a node ends if E_{rem} of that node is empty. This model is implemented on top of AVRORA [112], a highly scalable sensor node simulator.

Discussion

The lifetime estimation based on voltage drop rate is suitable for a static load application. However, it is difficult to be applied in the real deployment since a load of each sensor is normally dynamic, which leads to dynamic voltage drop. Moreover, the voltage drop technique is not always an accurate indicator [114, 49]. The other two methods are widely used by many studies. However, they are based on quoted capacity and do not take account of other impact factors on lifetime, such as temperature and consumption rate.

4.3 Summary and Discussion

Node lifetime is an important metric in WSNs. It depends on two important factors, which are battery capacity and current consumption of the device. Several existing works for finding the battery capacity and the current draw are explored.

Many electrochemical techniques can describe the battery processes and capacities in good detail but they require highly detailed knowledge of electrochemicals. The gravity measurement is an uncomplicated technique. However, it can only be applied to unsealed batteries. Voltage measurement is a commonly used technique for battery capacity estimation but it cannot provide an accurate capacity indicator. More accuracy is provided by the EMF method which is able to be implemented for many battery types. However, it does not include other impact factors on battery capacity, such as temperature and ageing. Load testing covers temperature and ageing effects and provides accurate and reliable battery capacity

estimation. However, it requires a special device, called a load tester. A generic model for the temperature effect is proposed by Sheridan et al. [101], which can be applied to many battery types. However, information details of the battery are required. Park et al. [25] proposed a model covering the remaining capacity and the effect of temperature on battery capacity. However, the full capacity voltage is assumed which might not be applied for NiMH battery type since there is a high self-discharge rate on the first day after full charging for this battery type.

To find the current draw of sensor nodes, the hardware-based method can provide accurate results. However, it is significantly high cost and difficult to add to the existing hardware. A software-based mechanism is easy to include into the existing system without additional per-unit cost, but fixed voltage and current draw values are normally assumed for current consumption calculation. The temperature and supply voltage effect formulas focused only the effect value on the leakage current draw but the method for finding the constant current draw was not provided.

Some lifetime estimation methods are presented. Monitoring the voltage drop is not appropriate in real world deployment, while the other methods do not consider some impact factors on lifetime, such as temperature and consumption rate.

As a result of accurate node lifetime estimation, the challenge is to propose a new generic technique for finding the battery capacity which covers temperature, ageing and self-discharging as well as other impact factors. Moreover, the software-based technique for finding the current consumption should be extended to cover the effect of temperature.

Chapter 5

Energy and Lifetime Aware Routing

5.1 Introduction

A sensor network is considered alive if it consists of a number of nodes which are able to transmit the collected sensor data to the base station [30]. Many approaches aim to balance the consumption of energy in wireless networks that maximises the number of alive nodes and can lead to an increase of network lifetime. The existing works in energy and lifetime aware routing algorithms are presented, including the discussion sections.

5.2 Existing Energy and Lifetime Aware Routing

Owing to energy constrained networks, energy consumption is an important metric in WSNs. Moreover, packet retransmissions should be minimised by sending packets through a good quality link in order to enhance the lifetime of the network. Therefore, link quality is normally included in the routing algorithm to select a good path. Furthermore, routing algorithms for WSNs have to ensure reliable multihop communication under the conditions where the sensor nodes have a limited transmission range, processing power and storage capability. To meet those conditions, many energy and lifetime aware routing protocols have been proposed. The following subsections present a review of some energy and lifetime aware routing protocols for real world WSNs.

5.2.1 Resource Aware and Link Quality (RLQ)

The Expected Transmission Count (ETX) [29] is the expected number of transmissions and retransmissions required to deliver a packet from the sender to the destination. It is a common link quality metric used for many routing algorithms in wireless networks. The lowest ETX means the least number of packet transmissions per packet delivery to the next hop or base station. The quality of a link based on ETX is calculated as:

$$ETX = \frac{1}{d_f \cdot d_r} \tag{5.1}$$

where d_f is the probability that a data packet successfully arrives at the receiver and d_r is the probability that the ACK packet is successfully received at the sender. The ETX calculation is modified by Gungor et al. to:

$$ETX = \sum_{i=0}^{K} i \cdot (1 - PRR)^i \cdot PRR \tag{5.2}$$

where PRR and K are the Packet Reception Rate and the maximum number of retransmissions before the packet is ignored. These values are reported by the MAC/PHY layer.

In RLQ [44], the routing decision is based on the link cost which depends on the energy consumption for transmission and reception, residual energy and the link quality. The link cost is defined as:

$$Link_{cost} = \eta_{tx} + \eta_{rx} \tag{5.3}$$

where η_{tx} and η_{rx} are the normalised energy cost for the sender and the receiver, which are calculated as:

$$\eta_{tx} = [(C_{tx-data} + C_{rx-ack}) \cdot ETX]^x \cdot \left[1 + \left(1 - \frac{E_{tx-res}}{E_{tx-init}}\right)\right]^y \tag{5.4}$$

$$\eta_{rx} = [(C_{rx-data} + C_{tx-ack}) \cdot ETX]^x \cdot \left[1 + \left(1 - \frac{E_{rx-res}}{E_{rx-init}}\right)\right]^y \tag{5.5}$$

where x and y are the weighting factors; $C_{tx-data}$ and C_{rx-ack} are the energy consumption during transmitting data and receiving acknowledgement for the sender; $E_{tx-init}$ and E_{tx-res} are the initial and remaining energy of the sender; $C_{rx-data}$ and C_{tx-ack} are the energy consumption during receiving data and transmitting acknowledgement for the receiver; $E_{rx-init}$ and E_{rx-res} are the initial and remaining energy of the receiver.

To change the routing decision, the weighting factors are adjusted. The routing

decision is based on the minimum hop count if both x and y are zero. If x=1 and y=0, the routing decision is based on link quality (the total energy consumption). If x=0 and y=1, the routing decision is based on the residual energy of the node. If both are 1, the routing decision is based on both link quality and the residual energy of the node.

5.2.2 Collection Tree Protocol (CTP)

CTP [40] is a tree-based distance vector routing protocol designed for sensor networks. It uses three mechanisms to overcome the challenges faced by distance vector routing protocols in WSN. These mechanisms are link quality estimation, datapath validation and adaptive beaconing.

- Link Quality Estimation

  In CTP, the ETX is calculated at link layer. The popular technique is a four-bit link estimator [36]. Four bits are used for accurate link estimation from three layers as shown in Figure 5.1. The first 2-bits: COMPARE and PIN, are taken from the network layer to identify the importance of that link. The next bit, ACK, is from the link layer which relates to packet acknowledgement information. The last bit, WHITE, is from the physical layer to identify the quality of signal, LQI, RSSI and SNR.

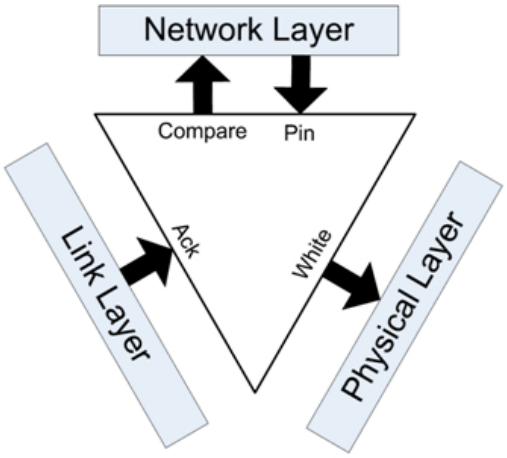


Figure 5.1: Four-Bits Link Estimator [36]

When a node receives a packet from a neighbour which does not exist in the neighbour table, this neighbour will be added in the table if there are free entries available. Otherwise, the neighbour is evaluated for replacing an entry in the table which is unpinned (PIN bit is not set) and has the worst link quality (ETX value > ETX threshold). If no such entry can be

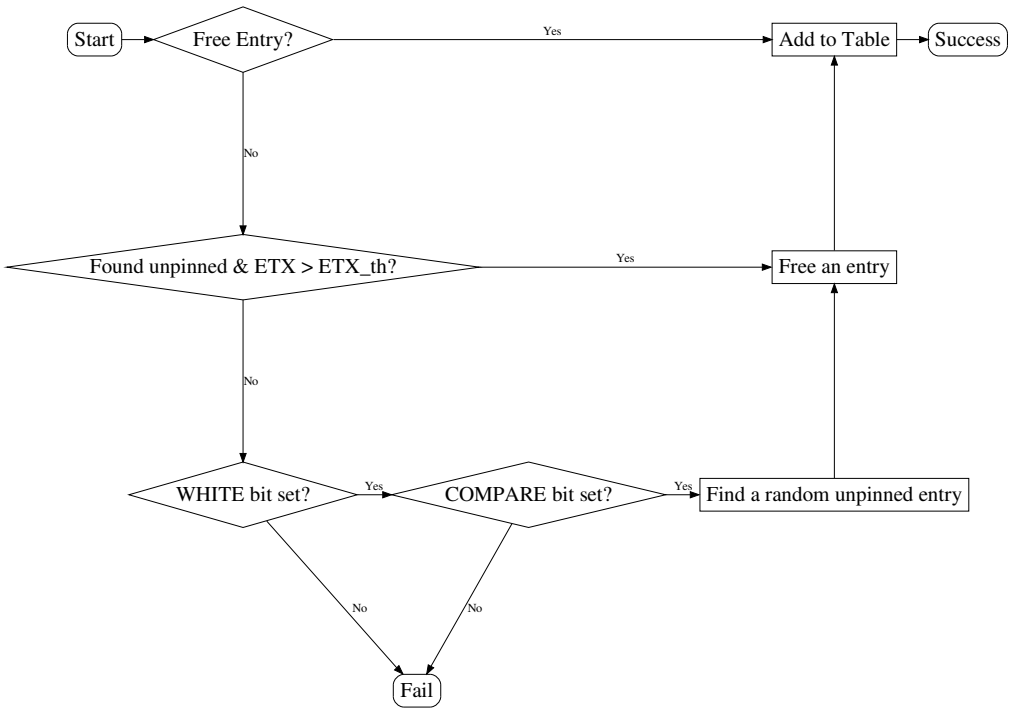


Figure 5.2: Neighbour Replacement Policy

found, the WHITE and COMPARE bits are used for further decision. If both WHITE and COMPARE bits are set for that neighbour, it will be considered for insertion by replacing one of the unpinned entries in the table (random selection). In contrast, if its WHITE or COMPARE bits are not set, it will not be considered for insertion. The ACK bit is used for ETX calculation on the sender side.

The ETX estimation on the sender side is [48, 43]:

$$ETX_s = \frac{T_p}{A_p} \tag{5.6}$$

where T_p is the total number of sent packets and A_p is the number of acknowledged packets. On the receiver side, the Packet Reception Ratio (PRR) is calculated based on the number of received packets (R_p) and total sent packets (T_p) [48, 43]. The receiver uses packet sequences to determine the number of total sent packets.

$$PRR = \frac{R_p}{T_p} \tag{5.7}$$

The resulting PRR value is then inversed to turn it into an ETX value as:

$$ETX_r = \frac{1}{PRR} \tag{5.8}$$

The window mean with Exponentially Weighted Moving Average (EWMA) is applied for every n sent or received packets in order to increase stability against instant fluctuations as [43]:

$$ETX_s(n) = \alpha \cdot ETX_s(n-1) + (1-\alpha) \cdot ETX_s \tag{5.9}$$

$$ETX_r(n) = \alpha \cdot ETX_r(n-1) + (1-\alpha) \cdot ETX_r \tag{5.10}$$

where $ETX_s(n-1)$ and $ETX_r(n-1)$ are the ETX values of the previous packet and α is a weighting factor between 0 and 1. Based on Equation 5.1, these two of ETX values are combined by applying d_f with $\frac{1}{ETX_s(n)}$ and d_r with $\frac{1}{ETX_r(n)}$. After that, ETX of four-bit estimator is calculated by using a second EWMA as:

$$ETX_{4B}(n) = \alpha \cdot ETX_{4B}(n-1) + (1-\alpha) \cdot ETX \tag{5.11}$$

- Datapath Validation

  Looping may occur due to changing link qualities, which causes network congestion and energy drain because of looping packets. As a result, these problems must be detected as quickly as possible. To detect these problems, datapath validation is used at the time of data packet transmission. Data packet transmissions and receptions are used for probing this problem. CTP can detect looping if the packets do not make progress towards the destination in the routing metric space.

- Adaptive Beaconing

  Routing protocols normally send broadcast packets at a fixed interval. With a small interval the protocol is more responsive to the changes in the network. However, it uses more bandwidth and energy. For a large interval, it uses less bandwidth and energy but it gives slow response to the change of network status. CTP introduced adaptive beaconing to break this tradeoff. Nodes send beacons faster if the topology is inconsistent and has problems. Otherwise, they reduce the beaconing rate exponentially. Therefore, CTP can quickly respond to adverse wireless dynamics while incurring low control overhead in the long term. The interval timer is reset to a small value when

one or more of the following conditions are met: (1) the routing table is empty; (2) The routing ETX of node increases by ≥ 1 transmission; (3) the node hears a packet with the "Pull" bit set (which means the other node requests the routing information).

The CTP routing frame is shown in Figure 5.3.

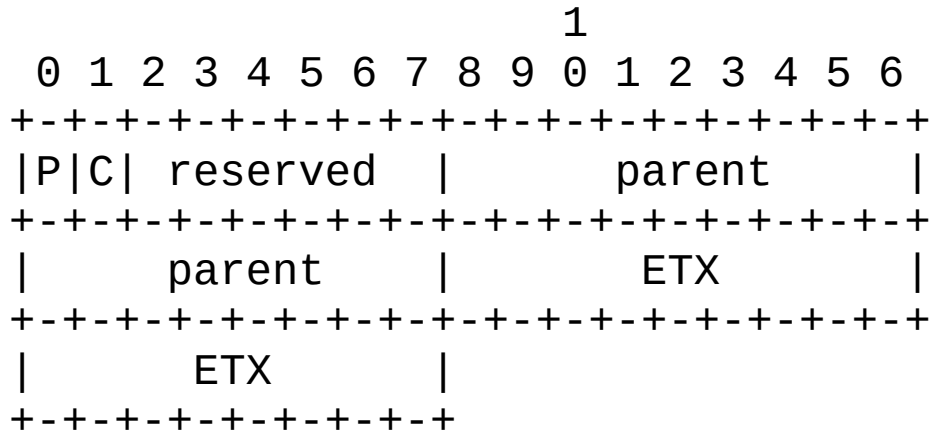


Figure 5.3: The CTP Routing Frame [40]

Field definitions are as follows:

- P: Routing pull

  The P bit allows nodes to request routing information from other nodes. If a node with a valid route hears a packet with the P bit set, it SHOULD transmit a routing frame in the near future.

- C: Congestion notification

  If a node drops a CTP data frame, it MUST set the C field on the next routing frame it transmits.

- Parent

  The node's current parent.

- ETX

  The node's current ETX routing metric value.

During the start topology formation, one of all nodes advertises itself as a root node (with ETX=0) by broadcasting a beacon frame to other nodes. The rest of the nodes join the routing tree by sending the request for routing information until they receive replied frames which contain node id and an ETX metric. The path with lowest cost is selected for forwarding data. The path cost is a cumulative of ETX from the node to the root. A node will forward packets to its parent which is the next hop neighbour of the selected path. After finishing parent selection, that node starts broadcasting beacon frames which are similar to the root node to establish connections with new adjacent nodes. The path selection algorithm of CTP is shown in Figure 5.4.

```
1. Initialize()
minETX ← 0xFFFF;
2. ParentSelection()
for RoutingTable[i]
  If(minETX > RoutingTable[i].ETX) & (RoutingTable[i].valid))
    minETX ← RoutingTable[i].ETX; bestETXRoute ← RoutingTable[i].nodeid;
  endif
endfor
Return bestETXRoute.nodeid;
```

Figure 5.4: CTP Algorithm

5.2.3 Energy and Link Quality Based Routing Tree (ELQR)

ELQR [84] is extended from CTP. It takes the routing decision based on ETX and the residual energy of the nodes. The routing table has an additional entry for the neighbour node's residual energy. From the routing table, the algorithm searches for the node with the highest residual energy and the node with minimum ETX. The node will be selected as the parent node if it has high energy and minimum ETX. If no one is satisfied to be a parent node, then the node with maximum energy with ETX value below threshold β will be considered as the parent or the minimum ETX with energy above threshold α will be considered as a candidate. The next optimum energy/ETX nodes are considered by searching the table again if none of these conditions are met. The nodes with low energy are removed from the routing table (marked as invalid) to avoid a hole in the network, which leads to network partitioning. Similarly, the nodes with bad wireless link quality are also removed from the routing table to avoid a huge number of retransmissions,

which leads to unnecessary energy waste. The path selection algorithm is shown in Figure 5.5.

```
1. Initialize()
maxEnergy ← 0; minETX ← 0xFFFF; β ← 50;
2. RouteSearch()
for RoutingTable[i]
  If((maxEnergy < RoutingTable[i].energy) & (RoutingTable[i].valid))
    maxEnergy ← RoutingTable[i].energy;
    bestEnergyRoute ← RoutingTable[i].nodeid;
  endif
  If(minETX > RoutingTable[i].ETX) & (RoutingTable[i].valid))
    minETX ← RoutingTable[i].ETX; bestETXRoute ← RoutingTable[i].nodeid;
  endif
endfor
3. ParentSelection()
//Choose the link with high energy and good quality link
If (bestETXRoute == bestEnergyRoute)
  Parent ← bestEnergyRoute.nodeid;
//Choose the best ETX path
elseif (bestETXRoute.Energy > α)
  Parent ← bestETXRoute.nodeid;
//Choose the best Energy path
elseif (bestEnergyRoute.ETX < (minETX + β))
  Parent ← bestEnergyRoute.nodeid; β ← β + β * Round / 100
// search for the next alternative parent
elseif
  bestEnergyRoute.valid ← 0; bestETXRoute.valid ← 0; Repeat step 1;
endif
Return Parent
```

Figure 5.5: ELQR Algorithm

5.2.4 Energy Aware and Link Quality Based Routing (ELR)

ELR [83], also extended from CTP, is based on energy awareness and link quality. In this algorithm, the routing table has an additional entry of energy which indicates the lowest energy of nodes in the route. There are two threshold parameters: an energy threshold (E_{th}) and ETX difference threshold ($ETX_{diff-th}$). After finding the best ETX route and the best energy route, the difference ETX (ETX_{diff}) is calculated. The node will be selected as parent node if it has both

the highest route energy and lowest ETX. In case the condition is not met, the node with the best energy route will be selected if its ETX_{diff} is less than or equal to $ETX_{diff-th}$. In the case of no one being chosen, the node will be selected if it has the best ETX route and energy greater than the energy threshhold. Otherwise, the node with the best ETX route is removed from the routing table (set to invalid) and the cycle is repeated until a best route has been selected. The algorithms for finding the best ETX route, the best energy route and path selection are shown in Figure 5.6, 5.7 and 5.8.

```
bestETXRoute()
minETX ← 0xFFFF;
for RoutingTable[i]
  If(minETX > RoutingTable[i].ETX) & (RoutingTable[i].valid))
    minETX ← RoutingTable[i].ETX;
   R_a ← RoutingTable[i].nodeid;
  endif
endfor
Return R_a;
```

Figure 5.6: bestETXRoute Algorithm

```
bestEnergyRoute()
maxEnery ← 0;
for RoutingTable[i]
  If((maxEnergy < RoutingTable[i].energy) & (RoutingTable[i].valid))
    maxEnergy ← RoutingTable[i].energy;
    R_b ← RoutingTable[i].nodeid;
  endif
endfor
Return R_b;
```

Figure 5.7: bestEnergyRoute Algorithm

```
1. Initialize()
R_a ← bestETXRoute(); R_b ← bestEnergyRoute();
ETX_diff ← R_b.ETX - R_a.ETX;
2. ParentSelection()
If (R_a == R_b)
  Return R_a;
elseif (ETX_diff ≤ ETX_diff−th)
  Return R_b;
elseif (R_a > E_th)
   Return R_a;
else
  R_a.valid ← 0;
  Repeat step 1;
endif
```

Figure 5.8: ELR Algorithm

5.2.5 Routing Protocol for Low-power and lossy network (RPL)

The RPL is an IPv6 Routing Protocol for WSNs. The design of RPL is also based on CTP mechanisms [119]. RPL has control message options with a generic format that can be used with different routing metrics and constraints. The control message format of the Directed Acyclic Graph (DAG) metric container is illustrated in Figure 5.9.

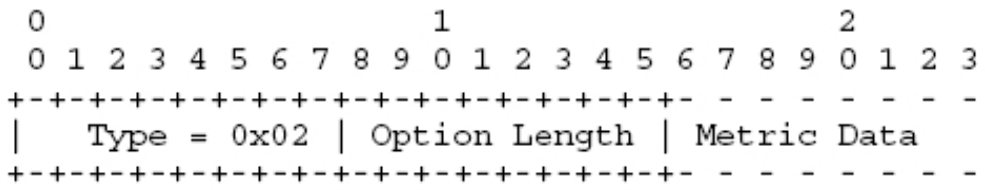


Figure 5.9: The RPL Control Message Option for DAG Metric Container [118]

In the Metric Data field, the metrics and constraints will be carried in the DAG metric container format as defined in Figure 5.10.
Field definitions are as follows:

- Routing-MC-Type (Routing Metric/Constraint Type - 8 bits)

  This field uniquely identifies each Routing Metric/Constraint object and is managed by IANA as shown in Table 5.1 [115].

- Res Flags field (6 bits)

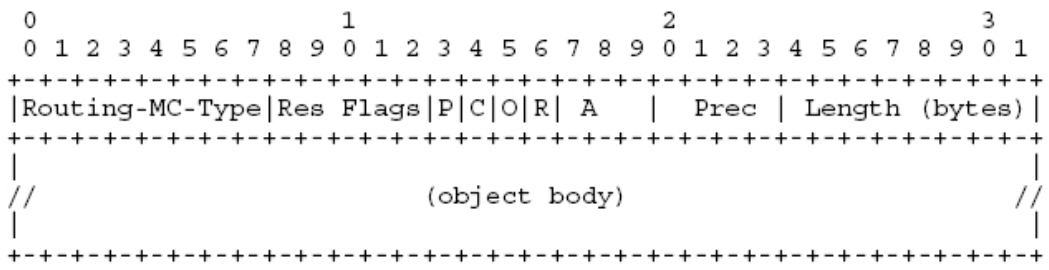


Figure 5.10: The RPL Routing Metric/Constraint Object [115]

Table 5.1: Routing Metric/Constraint Type

Value	Description
0	Unassigned
1	Node State and Attribute
2	Node Energy
3	Hop Count
4	Link Throughput
5	Link Latency
6	Link Quality Level
7	Link ETX
8	Link Color
9-255	Unassigned

This is an unassigned-bits field and considered as reserved.

- P flag

  This field is only used for recorded metrics. When it is cleared, all nodes along the path successfully record the corresponding metric. When it is set, this indicates that one or several nodes along the path could not record the metric of interest (either because of lack of knowledge or because this was prevented by policy).

- C flag

  When the C flag is set, this means that the Routing Metric/Constraint object refers to a routing constraint. When cleared, the routing object refers to a routing metric.

- O flag

  This flag is used exclusively for routing constraints (C flag is set). When set, this indicates that the constraint specified in the body of the object is optional. When cleared, the constraint is mandatory. If the C flag is zero, the O flag must be set to zero on transmission and ignored on reception.

- R flag

  This flag is only relevant for a routing metric (C=0) and MUST be cleared for C=1. When it is set, this means the routing metric is recorded along the path. In contrast, if it is cleared, the routing metric is aggregated.

- A Field (3 bits)

  This field is only relevant for metrics and is used for indicating: if A=0, the routing metric is additive; if A=1, the routing metric reports a maximum; if A=2, the routing metric reports a minimum; if A=3, the routing metric is multiplicative. This field has no meaning when the C flag is set (i.e. when the Routing Metric/Constraint object refers to a routing constraint) and is only valid when the 'R' bit is cleared. Otherwise, the A field has to be set to 0 and must be ignored on receipt.

- Prec field (4 bits)

  The Prec field indicates the precedence of this Routing Metric/Constraint object relative to other objects in the container. This is useful when a DAG Metric Container contains several Routing Metric objects. Its value ranges from 0 to 15. The value 0 means the highest precedence.

- Length (8 bits)

  this field defines the length of the object body, expressed in bytes. It ranges from 0 to 255.

Although RPL supports several metrics, ETX is the most widely used and well-acknowledged in RPL implementations on both TinyOS (TinyRPL) and Contiki (ContikiRPL). Therefore, the path selection algorithm is similar to CTP. With 2 bytes length of ETX metric, the length field in the Routing Metric/Constraint Object has a value of 2 and the option length field in the RPL Control Message Option has a value of 6. The routing beacon of RPL is called the DODAG Information Object (DIO) message. The beacon timer is called the trickle timer.

5.2.6 Discussion

Three metrics: energy consumption, residual energy and ETX, have been taken into account for path selection of the surveyed algorithms. Table 5.2 summarises the metrics used for each algorithm.

In RLQ, all three metrics are used for path selection. However, the sender node only considers the metrics of its link to its neighbours. This can lead to the

Table 5.2: The Summation of Routing Metrics for Path Selection

Routing Algorithm	Metrics		
	Energy Consumption	Residual Energy	ETX
RLQ	Next hop	Next hop	Next hop
CTP	-	-	Path
RPL	-	-	Path
ELQR	-	Next hop	Path
ELR	-	Path	Path

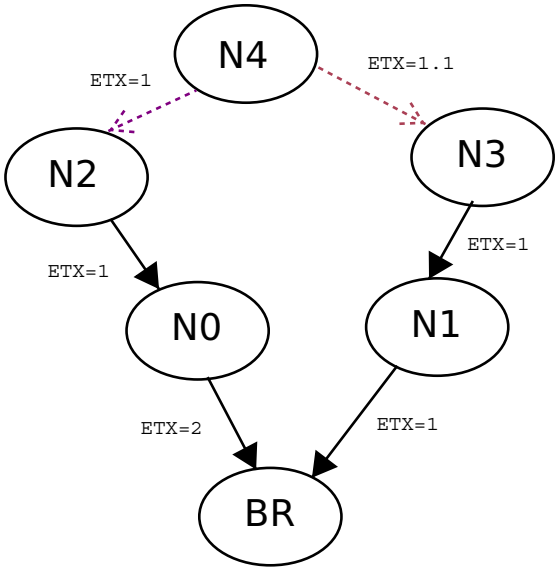


Figure 5.11: Example Routing Issue

reduction of node lifetime and, consequently, the network lifetime. Figure 5.11 illustrates the routing issue. Node $N4$ has two possible paths to the base station, via $N3$ or $N2$. Since the ETX of link $N4 \rightarrow N2$ is better than $N4 \rightarrow N3$, $N2$ will be selected even though it is not a good path (high packet retransmissions at $N0 \rightarrow BR$). CTP and RPL overcome this problem by using path ETX instead of next hop ETX. The sum of ETX from $N4 \rightarrow BR$ via $N3$ (path ETX = 3.1) is lower than via $N2$ (path ETX = 4). However with energy constrained, overloading can occur at $N2$ and $N0$. Consequently, they will be drained of energy faster. This problem is solved in ELQR by using the residual energy metric. Nevertheless, ELQR considers only the residual energy of the neighbour nodes, which is not sufficient information for lifetime balancing among forwarding nodes. ELR extends to using path residual energy instead of the residual energy of the next hop. Therefore, it has a higher number of alive nodes than the other three algorithms.

Normally, node lifetime is short with low residual energy. However, it is possible that a node with high residual energy runs out of energy before another with lower residual energy because it also has a high energy consumption rate. Since node lifetime depends on both residual energy and the energy consumption rate, a routing algorithm should include both of them in the routing metrics by changing from path residual energy to path lifetime for an increase in network lifetime.

5.3 Summary and Discussion

Five routing algorithms have been discussed which are RLQ, CTP, RPL, ELQR and ELR. Although RLQ takes energy consumption, residual energy and ETX metrics into account for path selection, it only considers the metrics for next hop view. In CTP and RPL, it considers the metric in the entire routing path but only path ETX is taken into account. The residual energy of next hop is added in ELQR but it is not sufficient information for lifetime balancing among forwarding nodes. ELR extends to the use of path residual energy. However, it is still not enough information for lifetime balancing since node lifetime depends on both residual energy and energy consumption rate. Therefore, it is interesting to replace path residual energy with path lifetime in the routing metrics for an increase of network lifetime.

Chapter 6

EETPC: Energy-Efficient TPC

6.1 Introduction

RSS can be affected by several factors, such as distance, transmission power, height above ground, multipath reflection environment, node's capabilities, temperature, and noise and interference. Therefore, these factors are investigated. The new TPC mechanism, called EETPC (Energy-Efficient Transmission Power Control) is proposed, which includes new equations and ND technique for supporting EE TPC. In this chapter, the impact factors on RSS are presented. Then, it describes the proposed energy-efficient transmission mechanism including an implementation on real hardware and software platforms in WSNs. Finally, the feasibility of the proposed schemes is validated by both real hardware testbed and simulation experiments using performance metrics such as delay, energy consumption and packet loss rate with different scenarios.

6.2 Experimental Setup

Mica2 and N740 NanoSensor platforms are used in these experiments. Mica2 nodes are equipped with Chipcon CC1000 radio and a standard $\lambda/4$-monopole antenna. For N740 NanoSensor platform, a node is equipped with a CC2431 System-on-Chip, an IEEE 802.15.4 compliant RF transceiver and $\lambda/2$-dipole antenna. Experimental programmes are developed in TinyOS for Mica2 and Contiki for N740 Nanosensor. One node is connected to a laptop (via serial for Mica2 and USB for N740) for collecting statistics. The experiments have been conducted in two indoor environments: an empty room and a corridor as shown in Figure 6.1.

(a)

(b)

Figure 6.1: Experimental Sites (a) an Empty Room (b) a Corridor

6.3 The Impact Factors on RSS

6.3.1 Distance

To investigate the impact of distance on RSS, an experiment was set up where a sender sent 500 packets with 0 dBm transmission power to the receiver every 3 seconds at several distances as shown in Figure 6.2. This experiment was set up in an empty indoor room with little or no significant multipath. The result of measured RSS and estimated RSS based on Equation 3.4 with η =1.6, $X_\sigma = 0$ dBm and $d_0 = 0.5$m is in Figure 6.3. It shows that RSS decreases as the distance increases. Estimated RSS values are close to (± 4) the measured RSS values.

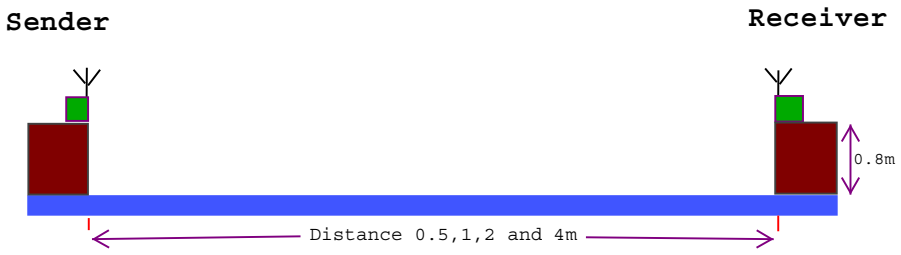


Figure 6.2: RSS and Distance Experiment

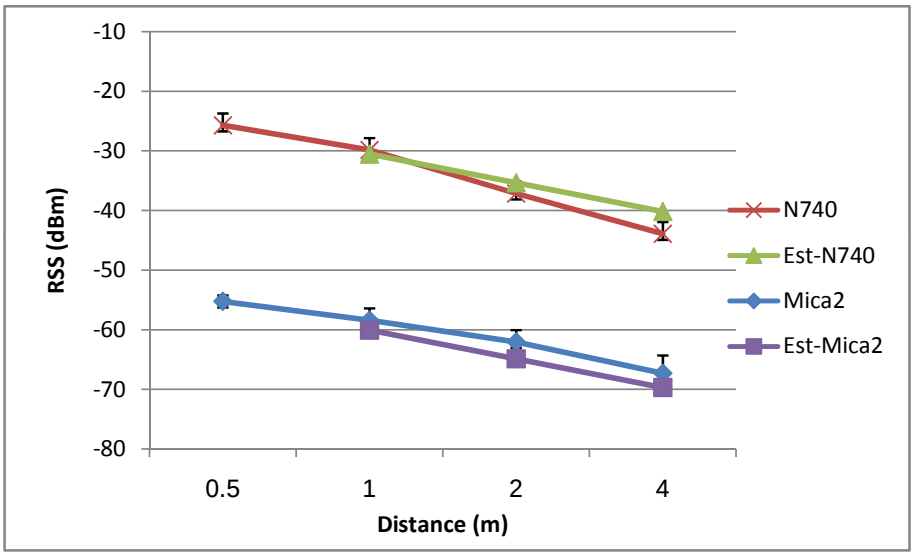
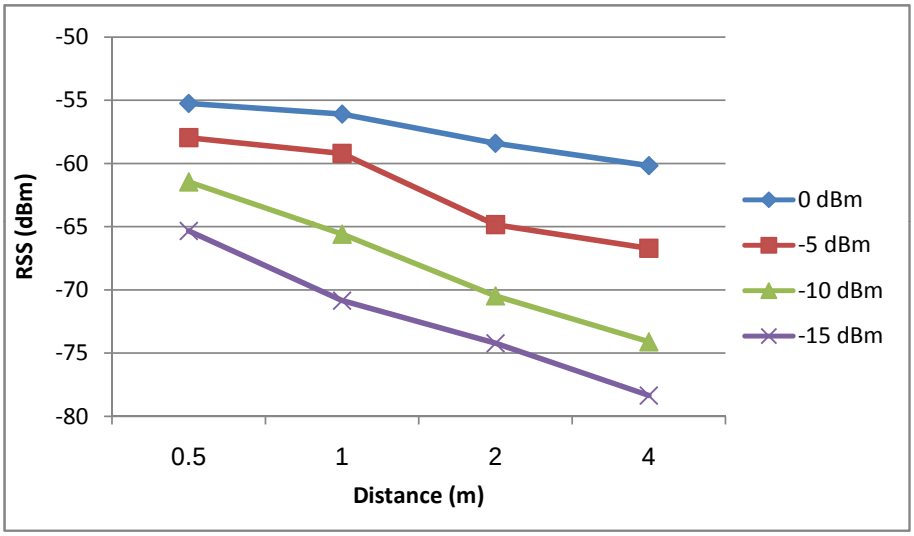


Figure 6.3: Measured RSS, Estimated RSS and Distances

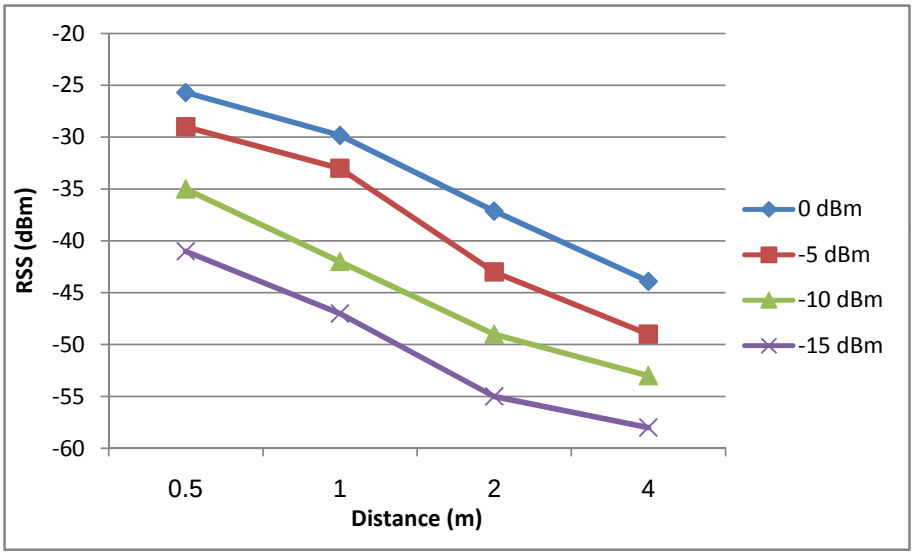
6.3.2 Transmission Power

To investigate the relationship between RSS and transmission power, the sender sent out 500 packets with different transmission power levels. Fig 6.4 illustrates the comparison between average measured RSS with different transmission power levels Mica2 and N740.

Regarding the results in Fig 6.4, it can be concluded that the relationship between RSS and transmission power level is a linear correlation. These results exhibit similar behaviour to other studies [68, 73] which investigated in various environments and over a long period of time. Moreover, the linear traces for measured RSS at different transmission power levels are approximately parallel to each other and follow almost the same difference in dBm as it is between the power levels, for example, RSS of -5 dBm transmission power is less than RSS of 0 dBm transmission power around 5 dBm. Therefore, it can be assumed that there is the same path loss for all transmission power levels at the same distance and environment.



(a)



(b)

Figure 6.4: Comparison RSS of Different Transmission Power Levels for (a) Mica2 (b) N740

6.3.3 Height above ground

To investigate the impact of antenna height above ground on RSS, an experiment was set up where a sender sent 500 packets with maximum transmission power with 0.8 m above ground and at the ground level. Both transmitting and receiving nodes are put on the same height of ground.

For ground level, the heights of antenna (H_1) are 0.08 m and 0.1 m for Mica2 and N740 (measuring from physical height of antenna). For 0.8 m above ground, the heights of antenna (H_1) are 0.88m and 0.9m for Mica2 and N740. The different signal strength between 0.8 m height and ground level for Mica2 and N740 are around 10 and 9 dBm, respectively, which is shown in Figure 6.5.

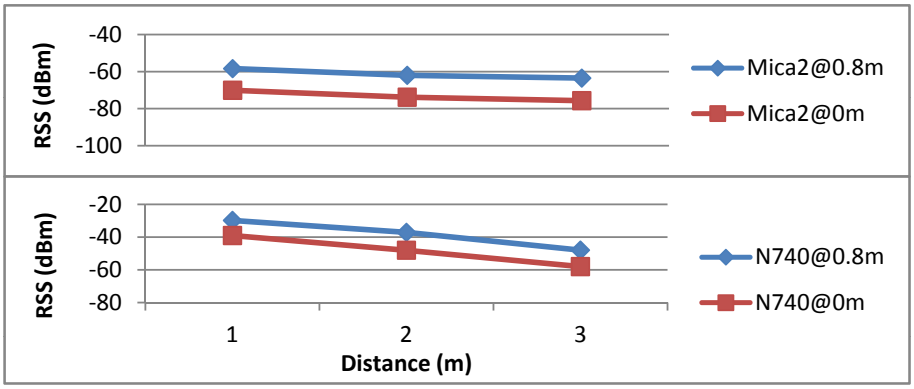


Figure 6.5: Comparison RSS of Different Antenna Heights above ground

6.3.4 Multipath

To investigate the impact of multipath on RSS, an experiment is set up where a sender sends 500 packets with maximum transmission power. This experiment is set up in the corridor environment. Figure 6.6 shows the comparison between estimated values based on LNSM with η =1.6, $X_\sigma = 0$ dBm and $d_0 = 1$m and the average measured values.

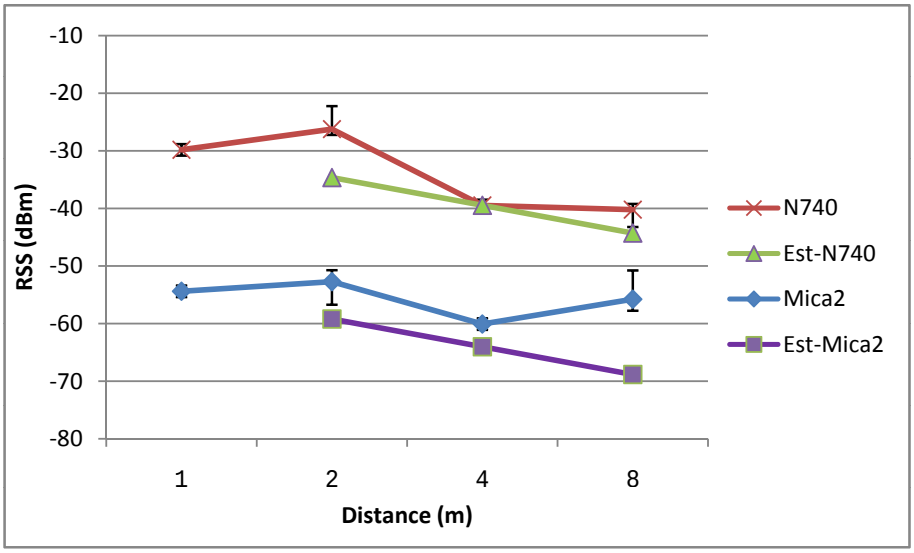


Figure 6.6: Comparison between Measured RSS_{max} and Estimated RSS_{max} for Mica2 and N740

Generally, RSS is inversely proportional to the square of the distance. However, the average measured values of RSS vary due to a problem of wave reflection from the corridors wall as shown in Figure 6.7. Depending on phase, indirect path signals can amplify the direct signal as well as weaken it. If the indirect signals arrive in phase with the direct one, it increases the amplitude and making overall signal appear stronger on the receiving end as in the 2 and 8 metre distances. Therefore, X_σ has to be adjusted at each distance for RSS estimation. Moreover,

even though signal strength is high, it exhibits low quality and high fluctuations.

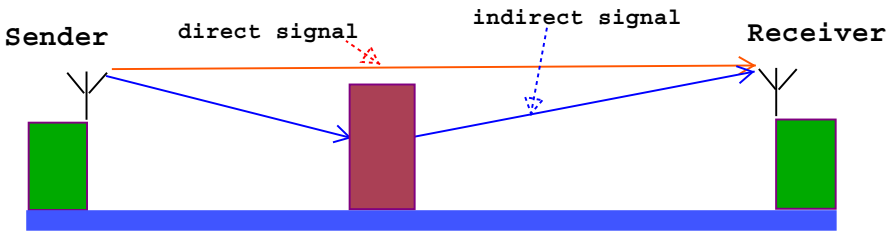


Figure 6.7: Signal Reflection

6.3.5 Node's Capability

To investigate the effect of node's transmission capability, two senders (Node1 and Node2) send 500 packets in the same environment with several distances. As shown in Figure 6.8, all the RSS values of Node1 are a bit higher than the values of Node2. Therefore, it can be concluded that the same path loss model cannot be used for all sensors, even though they have the same hardware. It needs to be adjusted depending on the capability of each node.

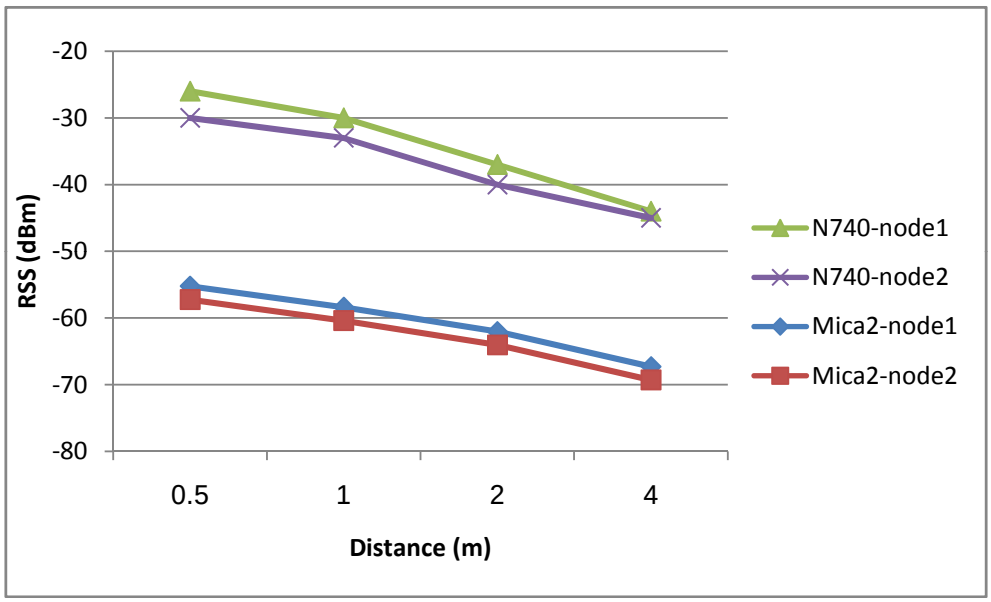


Figure 6.8: Comparison between RSS of Node1 and Node 2

6.3.6 Temperature Effect

This experiment is carried out indoors. Two nodes, called the sender and forwarder, are placed in the control environment, while another node, called the

receiver, is placed in the normal room conditions as shown in Figure 6.9. Only the sender node is not equipped with antenna. The effect of temperature is investigated in the two different settings. In the first test, the controlled environment is exposed to a temperature variation from 22 °C to 28 °C caused by a heater. For the second test, the sender and forwarder are wrapped in plastic bags and put in the refrigerator with the temperature variation from 25 °C to 5 °C. Every 5 minutes, the sender reads 10 temperature values and sends 10 packets to the forwarder. Then, these packets are forwarded to the receiver attached with the measured RSS of received packets from the sender. After that, the receiver forwards those messages attached with the measured RSS of received packets from the forwarder for collecting statistics. These 10 packets are used for averaging the temperature and RSS.

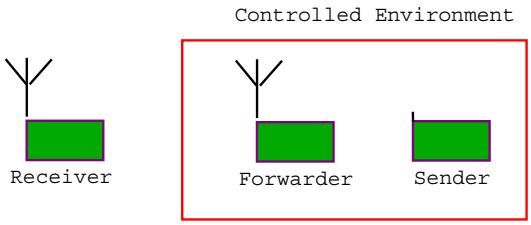
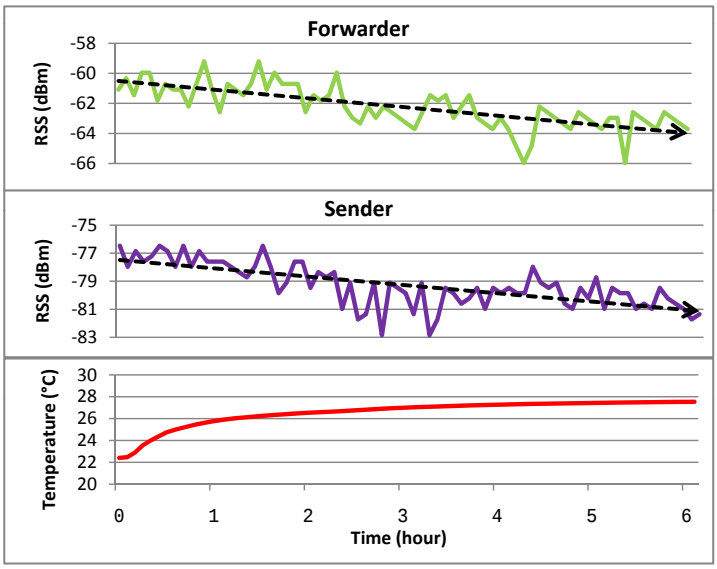


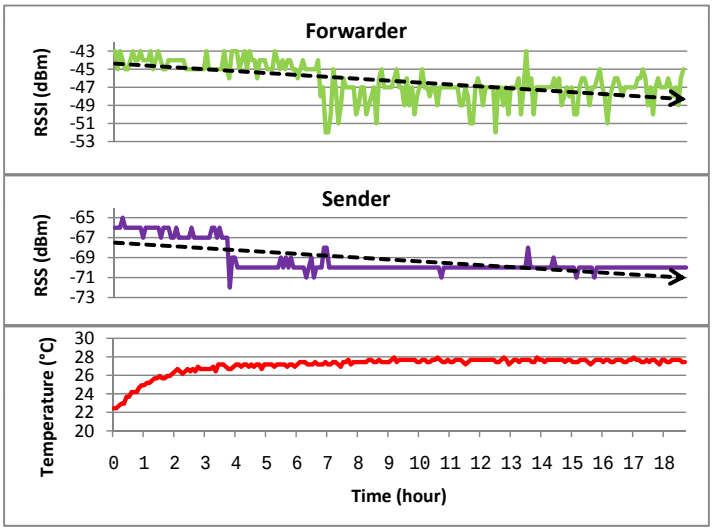
Figure 6.9: Temperature Experiment

The RSS collected with an increase of temperatures for the sender and forwarder are shown in Figure 6.10. The high temperature affects the signal strength of the forwarder. When the temperature rises from 22 °C to 28 °C, the RSS of the forwarder decreases approximately 2 dBm, which means the RSS decreases around 0.33 dBm when temperature increases 1 °C). This shows the effect of temperature on the transmitter while the temperature of the receiver is fixed at room temperature (around 22 °C). Moreover, it is found that when both the transmitting and receiving nodes are in the rising temperature, the effects are summed as 4-5 dBm decrease in RSS. Both Mica2 and N740 exhibit similar trends as well as the results from the temperature experimented in Bannister et al. [16].

From the second test, an increase of RSS for Mica2 is 3 dBm when the temperature of the transmitting node is cooled from 25 °C to 10 °C, while RSS increases 4 dBm for N740 from 25 °C to 5 °C. This means that the RSS increases around 0.2 dBm when temperature decreases 1 °C for both Mica2 and N740. When both transmitting and receiving nodes are cooled, RSS increases around 5 dBm for Mica2 and 7 dBm for N740, which are close to double in the effects on only the transmitting node. These results also exhibit the same trends as the temperature experiments of Boano et al. [24]. A slight difference may be caused by humidity or other factors.



(a)

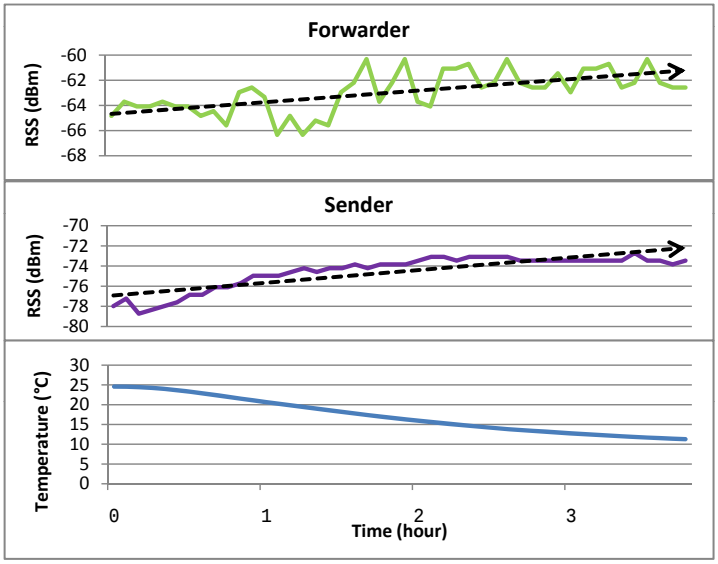


(b)

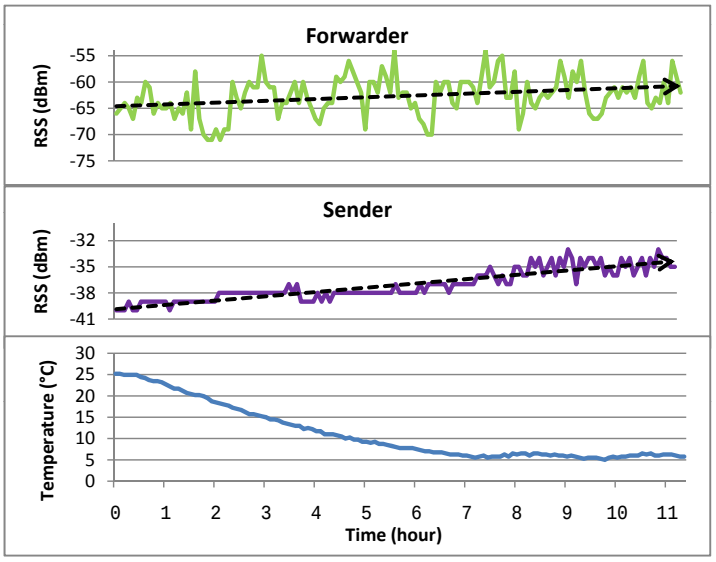
Figure 6.10: Effects of High Temperature on RSS for (a) Mica2 (b) N740 Nanosensor

6.3.7 Noise and Interference

Noise is the unwanted signal which may affect RSS. Normally, the receiver can receive packets if the Signal-to-Noise Ratio (SNR) is greater than the SNR threshold. However, even if it can receive packets, noise can interfere with its normal operation, resulting in degradation of the RSS. To study the effect of noise and interference on RSS, a node, called a sender, is placed in the controlled environment, while another node, called a receiver, is placed in the normal room conditions. Since magnetic interference from electrical devices can interfere the wireless signal, the effect of noise and interference is investigated in two different settings. In the first test, the controlled environment is near a heater. For the second test, the



(a)

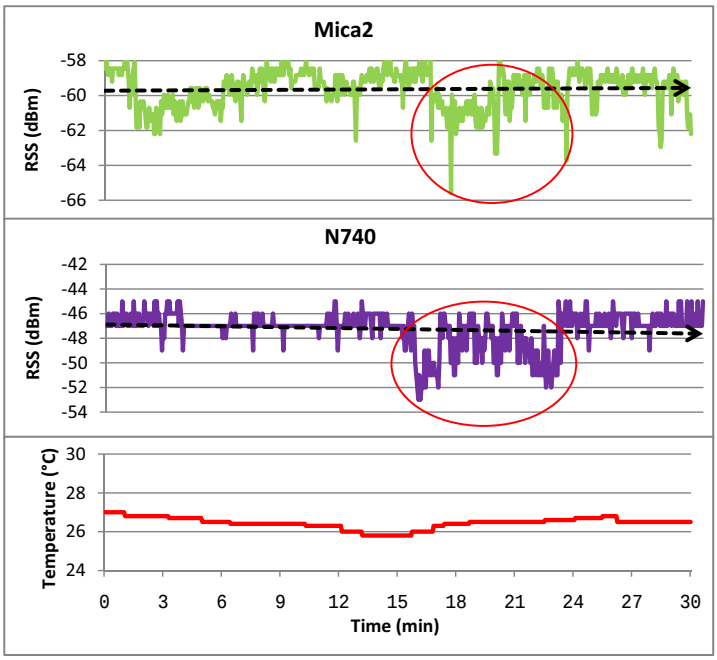


(b)

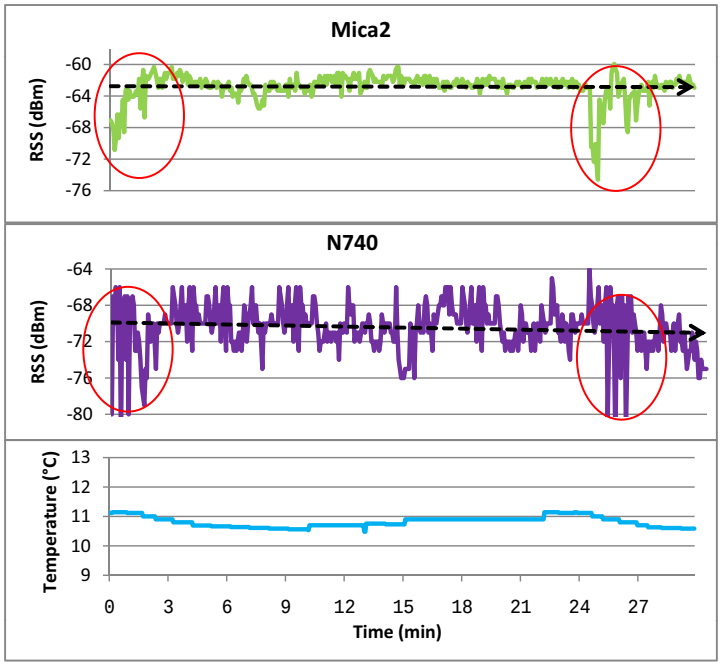
Figure 6.11: Effects of Low Temperature on RSS for (a) Mica2 (b) N740 Nanosensor

controlled environment is in the refrigerator. Every 3 seconds, the sender node reads the temperature value and sends a packet to the receiver node. The receiver measures the RSS of every received packet. Two platforms: Mica2 and N740, are investigated concurrently in these two tests.

For the near heater test, the average RSS of Mica2 and N740 are -60 dBm and -47 dBm, while the average RSS of Mica2 and N740 are -64 dBm and -70 dBm for the refrigerator test. Received signal fluctuations by normal noise of Mica2 and N740 are around $\pm$ 2 and $\pm$ 3 for the near heater test, while they are $\pm$ 2 and $\pm$ 5 for the refrigerator test. Heater and refrigerator have an automatic on/cut-off



(a)



(b)

Figure 6.12: Effects of Noise and Interference on RSS for (a) Heater (b) Refrigerator

depended on temperature. When the temperature drops lower than a specified value (e.g. 26 °C at 15 mins of running time), the heater switches to run and generates the interference signals. In a similar way, the refrigerator switches to run and generates the interference signals when the temperature rises higher than a specified value (e.g. 11 °C at 0 and 24 mins of running time). The degradation of

signal strength and more fluctuation occur during the period of interference. For both Mica2 and N740, the level of degradation by the heater inteference and the refrigerator inteference are around 2 dBm and 5 dBm, while the signal fluctuations increase to ± 6 and ± 10.

6.4 EETPC Design and Implementation

Transmission Power Control (TPC) is normally implemented in MAC/PHY layer. However, if the routing layer is not taken into account, TPC might be useless. For example, the routing layer always chooses an energy-inefficient path through which a node has to send packets with the maximum transmission power. In this dissertation, the proposed routing algorithm in Chapter 8 has already included transmission power as one of the routing metrics for path selection. The main objective of TPC in this section is energy consumption reduction for next hop transmission in WSNs.

6.4.1 EETPC Design

If P_{Tmax} denotes the maximum transmission power, then 2.10 is transformed to 6.1 by replacing P_T with P_{Tmax}. Thus, RSS_{max} is the received power at the destination for transmission with P_{Tmax} which can be calculated as:

$$RSS_{max} = P_{Tmax} - PL \tag{6.1}$$

RSS Estimation

Based on the experiment in section 6.3.2, PL is the same for all transmission power levels. Therefore, the new RSS estimation for any transmission power is written by combining 6.1 and 2.10 as:

$$RSS(P_T) = P_T + RSS_{max} - P_{Tmax} \tag{6.2}$$

Finding the Ideal Transmission Power

From Equation 6.2, if RSS is replaced by the receiver sensitivity (RSS_{min}), the ideal transmission power (P_{Tideal}) will be:

$$P_{Tideal} = RSS_{min} - RSS_{max} + P_{Tmax} + X \tag{6.3}$$

The accuracy of the equation can be improved by adding X to cope with signal fluctuation and accuracy of RSSI measurement by the device. Values for both

receiver sensitivity (RSS_{min}) and maximum transmission power (P_{Tmax}) can be obtained from the device data sheet, e.g. for CC2431 these values are -92 dBm and 0 dBm, respectively [8]. The measured RSS_{max} can be obtained during the discovery process. The value of X can be obtained by a survey measurement made over a wide range of locations. The simple calculation for X is:

$$X = RSS_h - RSS_l + RSSI_{accuracy} \tag{6.4}$$

where RSS_h and RSS_l are the highest and lowest RSS values obtained in the survey measurement, and $RSSI_{accuracy}$ is the accuracy of RSSI measurement by the device which can be obtained from the device data sheet (e.g. 6 dBm for CC2420). The transmission power will be set to the closest power level provided by the radio device which is greater than or equal to the calculated one. In the worst case, the transmission power will be set to the maximum power if the calculated value is greater than the maximum power level provided by the radio device.

Dynamic Transmission Power Adjustment

Two methods are proposed for dynamic transmission power adjustment. The first is the formation of updated and accurate RSS_{max}. The second is the power compensation equation due to temperature change.

- The Accurate RSS_{max}

Based on the proposed equation, the accurate RSS_{max} value is essential. For dynamic power adjustment, the sender node has to keep updated on its current RSS_{max} and finding the accurate RSS_{max} ($AccRSS_{max}$) by using the Weighted Moving Average (WMA) as in Equation 6.5.

$$AccRSS_{max} = \alpha \cdot PrevRSS_{max} + (1 - \alpha) \cdot CurrRSS_{max} \tag{6.5}$$

where α is the weighting factor, $PrevRSS_{max}$ is the previous obtained RSS_{max}, and $CurrRSS_{max}$ is the current RSS_{max}. In this research, the current data is assigned a higher weight than the previous one, i.e. a double weight as $\alpha = 0.33$. Then, the new appropriate transmission power level can be calculated by using Equation 6.3 with RSS_{max} as $AccRSS_{max}$.

- Temperature Effect Equation

Temperature effect can be included for transmission power adjustment as:

$$P_{Tideal}(T) = PrevP_{Tideal} + \tau \cdot diffT \tag{6.6}$$

where $PrevP_{Tideal}$ is the previous ideal transmission power calculated as P_{Tideal} using Equation 6.3, $diffT$ is the temperature changes (± Kelvin) after the previous ideal transmission power calculation and τ is temperature effect value, e.g. around 0.3 dBm obtained from temperature effect experiments in subsection 6.3.6.

6.4.2 EETPC Implementation

The proposed equation for finding the ideal power needs RSS_{max} which can be provided by the ND process. Two existing ND algorithms are focused and modified. These algorithms are the asymmetric link ND and subset of Routing Protocol for Low-power and lossy network (RPL). The modified asymmetric link ND is proposed for single-hop and multihoming networks, while the modified subset of RPL is proposed for the multihop network.

Modified Asymmetric Link ND

For asymmetric link ND, after a node has been switched on, it starts broadcasting a hello packet with maximum transmission power to the base station (BS). Then, the base station replies with a unicast message with $RSS_{max}(node, BS)$ to that node as shown in Figure. 6.13. Then, that node can calculate the ideal power based on the proposed Equation 6.3. This process should run periodically to keep RSS_{max} updated. For a multihoming network with multiple base stations, a node chooses the best base station to which it can transmit data packets with the lowest transmission power.

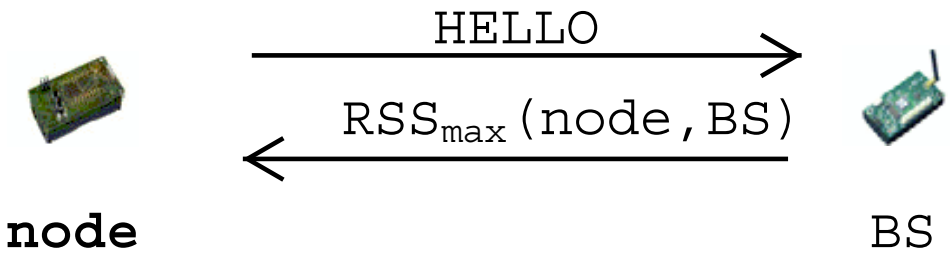
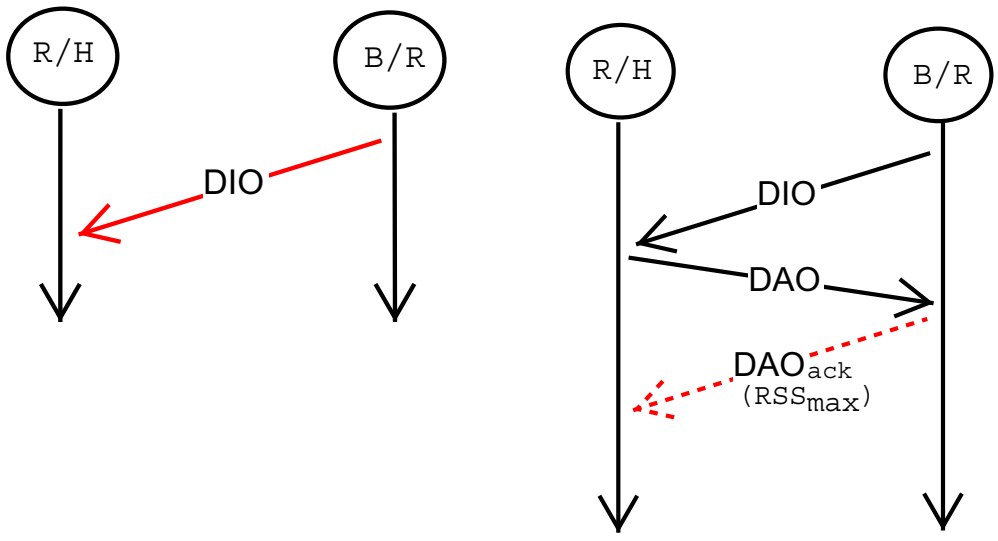


Figure 6.13: Modified Asymmetric Link ND

Modified subset of RPL

RPL router discovery is a subset of RPL protocol in the step of DIO and DAO message exchanges. Instead of using a static time period, RPL uses an adaptive timer mechanism, called the trickle timer. This mechanism controls the sending rate of DIO and DAO messages. An increase/decrease of the trickle timer depends on the stability of the network. If the consistency network is detected, the trickle

timer will be raised, which results in fewer DIO and DAO messages being sent in the network, while it is reduced where the inconsistency is detected. This consistency also reflects a strong signal of the receiving packet. To embed the proposed idea into the RPL router discovery process, some steps are modified. If RPL is configured only for traffic in the upward direction, RSS_{max} can be obtained by measuring the RSS value of the DIO message sending with the maximum transmission power by a broader router (B) or a router (R). It is assumed that the link between two nodes is symmetrical in this case. If RPL is configured to support both upward and downward traffics and DAO acknowledgement is always requested, RSS_{max} can be obtained from the DAO acknowledgement. While a host (H) or a router sends a DAO message, it uses the maximum transmission power. After receiving a DAO message, the receiving node (B or R) replies back a DAO_{ack} message which includes the measured RSS_{max} of that received DAO message. This RSS_{max} value is placed in the reserved byte of DAO_{ack} message. Therefore, no extra byte is required for the acknowledgement message. Then, that node can calculate the ideal power for transmission of data packets by using the proposed Equation 6.3. These steps are presented in Figure 6.14.



(a) (b)

Figure 6.14: Subset of RPL for supporting EETPC (a) upward traffic (b) upward and downward traffics

6.5 EETPC Evaluation

This section evaluates the EETPC equations and technique on three hardware platforms: Mica2, N740 and MicaZ. In this study, N740 sensors are run on real testbed experiments, while MicaZ nodes are run on the simulator. Mica2 are conducted on both a real testbed and simulator. The simulator used in this study is the AVRORA [112], a simulator based on Atmel AVR processor. It supports energy consumption calculation based on data sheet and a free-space path loss propagation model for RSS estimation. Moreover, noise and interference can be defined for each running time period. This allows for different environments even at the same position, which results in the different RSS values. In addition, if the RSS is lower than the noise value at that time, packet loss will occur. Four transmission power levels are used for this study, which are 0 dBm, -5 dBm, -10 dBm and -15 dBm. N740 platform is implemented on Contiki OS, while Mica2 platform is implemented on the TinyOS environment. MicaZ platform can be implemented on both Contiki and TinyOS environments.

6.5.1 Finding the ideal Power

Performance Metrics

Changing transmission power may affect three performance metrics: delay, packet loss rate and energy consumption. These metrics are evaluated through testbed and simulation experiments.

- Delay

There are two parts for delay which are ND delay and packet transmission delay. The ND delay is measured as the time from starting the discovery process by the sender node until receiving some information from the base station for finding the ideal transmission power. The packet transmission delay is measured as the time from data packet sending from the sender node until receiving that packet at the base station. Two delay factors, transmit time and radio propagation time, are important in the data packet transfer between two nodes. Transmit time is the time for a radio device to transmit a packet over a radio link. This also includes the power switching operation before starting sending the packet frame. Radio propagation time is the time for a signal to propagate over the air to reach the receiver. The packet transmission delay is obtained by using the Round Trip Time (RTT) of different transmission power levels testing.
Since the radio coverage of a WSN device is short and usually less than 100 metres, the propagation delay is negligible for different transmission power levels due to

radio propagation speed being 300 metres per microsecond [86]. The power switching operation takes very little time (in nanoseconds) [1, 7]. Therefore, it can be concluded that changing transmission power has no effect on packet transmission delay.

- Packet Loss Rate

There are pros and cons of changing transmission power for packet loss rate. Reducing transmission power causes a weak signal and may lead to an increase of packet loss. Moreover, lower transmission can exacerbate the classic hidden terminal problem and result in an increase of packet loss. However, the hidden terminal can also occur with the maximum power transmission and it has been coped with in the collision avoidance mechanism employed at the MAC layer. In addition, decreasing transmission power can reduce packet loss by reducing the collision domain and then allows concurrent transmission of nodes on the same channel as shown in Figure 6.15.

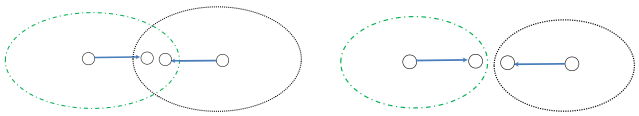


Figure 6.15: (a) Hidden Terminal (b) Concurrent Transmission

- Energy Consumption

Switching transmission power may increase the energy consumed by some components, e.g. CPU and radio transmitter need to run the power switching operations. However, this increase of energy consumption is very little compared to the transmission energy consumption, which is the dominant part in the total energy consumption. Reducing transmission power decreases the energy required for transmission. Moreover, it also reduces the receiving packets which are not addressed to neighbour nodes, which results in the overhearing energy consumption reduction of neighbour nodes. Based on the data sheet, it is assumed that the transitions between states for all devices are zero for calculation as in Equation 2.5. The AVRORA simulator also uses this technique to estimate the energy consumption of components and devices, such as CPU, LEDs, radio transceiver, sensor board and flash memory. Energy consumption for a node of three sensor platforms when it transmits 100 packets with different transmission power levels is shown in Figure 6.16.

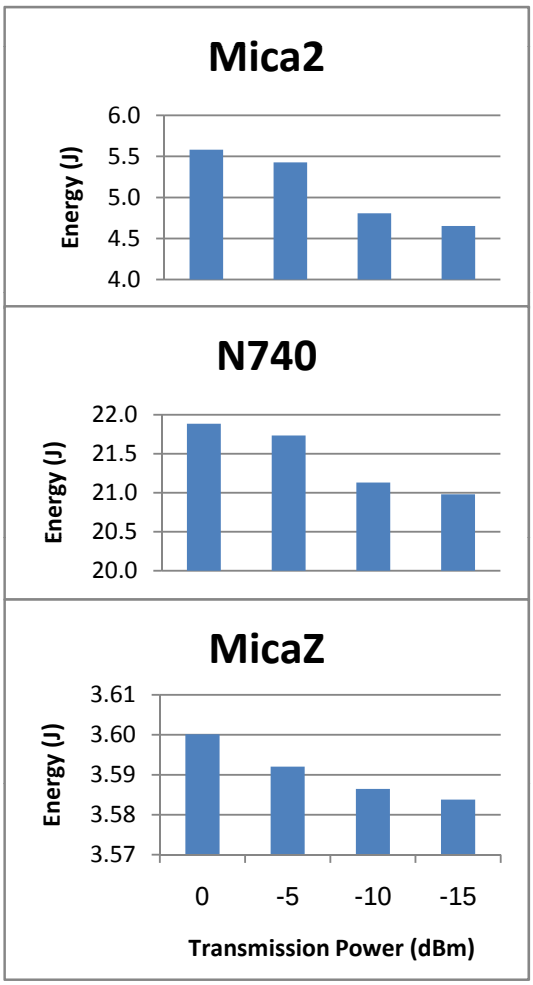


Figure 6.16: Energy Consumption with Different Transmission Power Levels

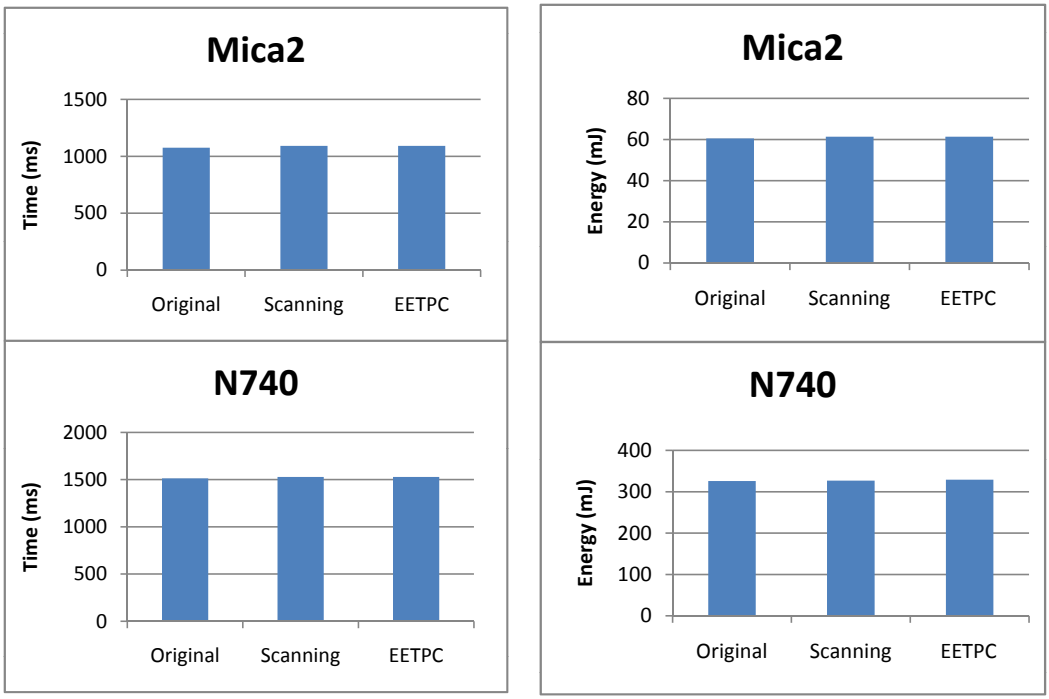
Testbed Experiments

Three techniques: the original with maximum power, scanning-based and EETPC, are investigated for finding the ideal transmission power. They are implemented inside the standard MAC protocol provided in TinyOS for Mica2 and on Rime stack with Null-MAC in Contiki OS for N740. The packet retransmission is disabled. Based on EETPC Equation 6.3, 10 dBm is used for X. The values of -85 dBm and -95 dBm are used as RSS_{min} for N740 and Mica2 in order to ensure that no transmission power reduction has been carried out for low quality signals. These experiments consist of two nodes, the sender and the base station. For finding the base station, the sender node starts the asymmetric ND process as shown in Figure 6.13 by sending a HELLO message with the maximum transmission power for the original one and EETPC, while it sends with the minimum power for the scanning based technique. If it does not receive an acknowledgement of that HELLO message in 1 second, it retransmits that packet with the maximum power for original and EETPC, while it retransmits with the next transmission

power level for the scanning based technique. After completing the ND process, the sender starts sending 100 data packets with sequence numbers for measuring packet loss rate. The interval of sending each packet is 3 seconds. Two scenarios are applied in the experiments.

- Scenario1: Strong Signal Position

For this scenario, both the sender and the base station are placed at 0.8 m above ground level at 1 metre distance for obtaining strong signal from the sender to the receiver. All tests were repeated 3 times to find the average performance results.



(a) (b)

Figure 6.17: Comparison Testbed Result of Discovery Process at Strong Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption

The experimental results show a similar behaviour for both sensor platforms. The average discovery period and discovery energy consumption at strong signal position are shown in Figure 6.17. All techniques consume the same amount of energy since they have almost the same discovery period. However as in Table 6.1, scanning and EETPC techniques can provide the minimal transmission power (-15 dBm), while the original technique always uses the maximum power level (0 dBm). Therefore, the overall energy consumption can be reduced around 3 - 20% for scanning and EETPC techniques compared to the original one as shown in Figure 6.18.

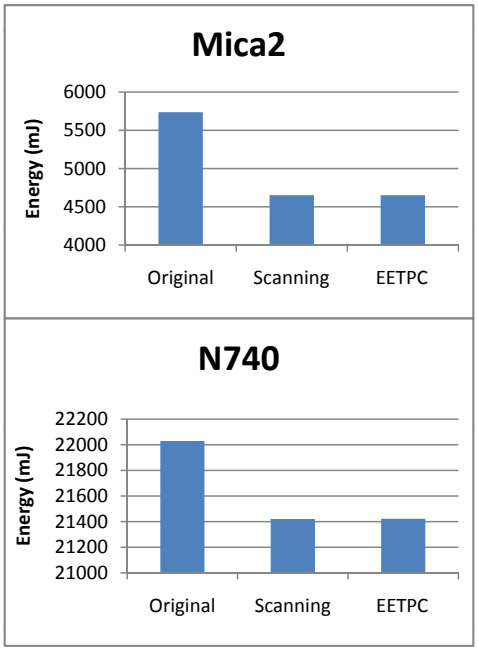


Figure 6.18: Testbed Overall Energy Consumption at Strong Signal Position

Table 6.1: Testbed Experimental Result at Strong Signal Position

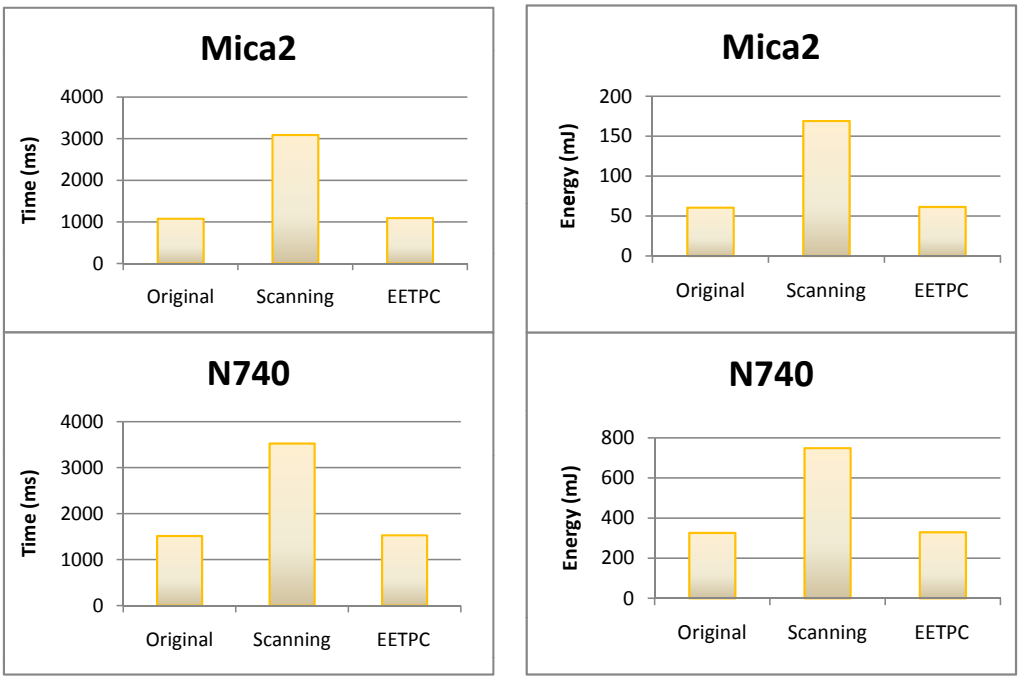
	Mica2	N740
$\overline{RSS_{max}}$ (dBm)	-60	-32
Original TX Power (dBm)	0	0
Scanning TX Power (dBm)	-15	-15
EETPC TX Power (dBm)	-15	-15
Original Packet Loss (%)	0	0
Scanning Packet Loss (%)	0	0
EETPC Packet Loss (%)	0	0

- Scenario2: Weak Signal Position

For obtaining weak signal, the sender is not equipped with antenna and both the sender and the base station are placed at the ground level. All tests were repeated 3 times to find the average performance results.

Table 6.2: Testbed Experimental Result at Weak Signal Position

	Mica2	N740
$\overline{RSS_{max}}$ (dBm)	-100	-80
Original TX Power (dBm)	0	0
Scanning TX Power (dBm)	-10	-10
EETPC TX Power (dBm)	0	0
Original Packet Loss (%)	0	0
Scanning Packet Loss (%)	20	5
EETPC Packet Loss (%)	0	0



(a) (b)

Figure 6.19: Comparison Testbed Result of Discovery Process at Weak Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption

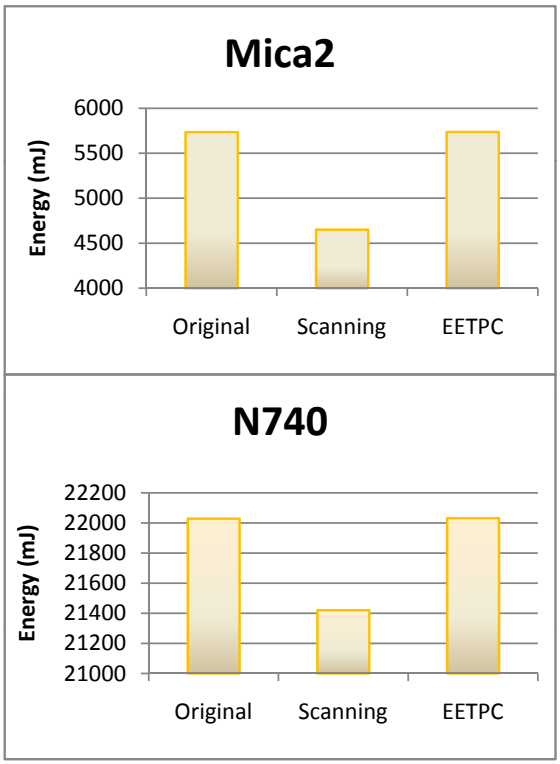


Figure 6.20: Testbed Overall Energy Consumption at Weak Signal Position

At the weak signal position in Figure 6.19, the scanning-based approach has more delay in the discovery process since the sender has to send many broadcast

messages until the target power level has been found, while EETPC sends only one message to get the target one. This long delay leads to an increase in the energy consumption for the discovery process. Furthermore, when the process of neighbour discovery finishes late, it causes a delay in the start of data packet sending. The scanning-based technique can obtain the lowest transmission power level and its overall energy consumption is the lowest one in the sequence packet delivery without retransmission, as in Figure 6.20. However, packet loss may occur when the signal strength drops near the receiver sensitivity. That is why the packet loss occurs 5-20% for the scanning-based approach, as in Table 6.2. This will lead to higher overall energy consumption on a system with enabled retransmission.

Simulation Experiments

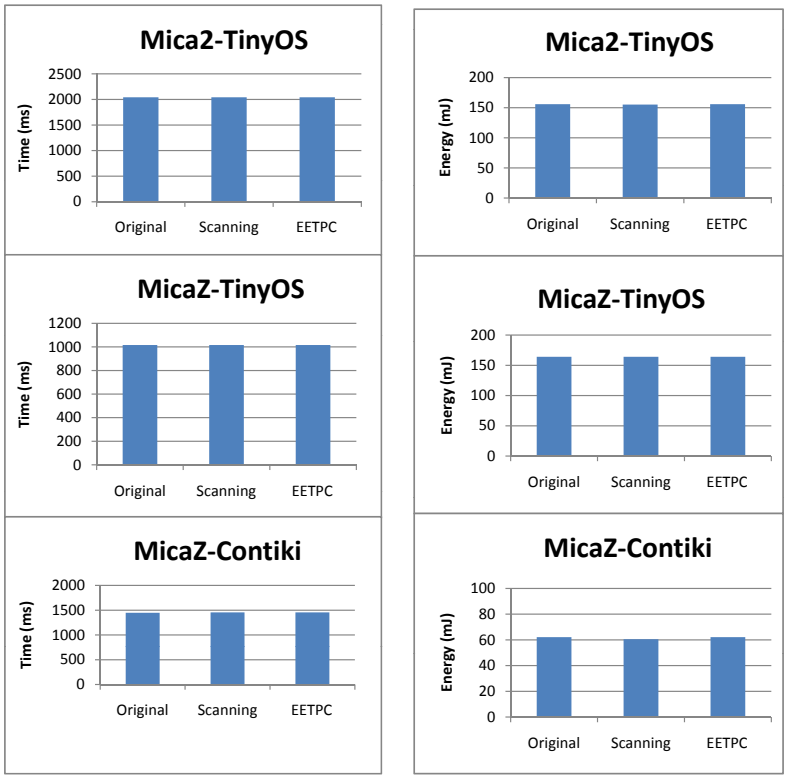
With similar scenarios to the testbed experiments, three techniques for finding the ideal transmission power are implemented in both TinyOS and Contiki with two hardware platforms: Mica2 and MicaZ. Moreover, two more scenarios are added for testing with multihoming and multihop networks. MicaZ platform is implemented in both TinyOS and Contiki environments, while only Mica2 platform is implemented in TinyOS. The packet retransmission is disabled. These simulation experiments run on the AVRORA simulator. The value of -95 dBm is used as RSS_{min} for Mica2 and -90 dBm for MicaZ.

- Scenario1: Strong Signal Position

To obtain a strong signal, the distance between the sender and the base station is 1 metre. All tests were repeated 3 times to find the average performance results. The experimental results exhibit similar behaviours to the testbed experimental results. In Figure 6.21 and Table 6.3, discovery times for all techniques are almost the same for strong signal position. Both scanning and EETPC techniques can provide the minimal transmission power (-15 dBm) and they give lower energy consumption than the original technique.

Table 6.3: Simulation Experimental Result at Strong Signal Position

	Mica2-TinyOS	MicaZ-TinyOS	MicaZ-Contiki
$\overline{RSS_{max}}$ (dBm)	-56	-57	-57
Original TX Power (dBm)	0	0	0
Scanning TX Power (dBm)	-15	-15	-15
EETPC TX Power (dBm)	-15	-15	-15
Original Packet Loss (%)	0	0	0
Scanning Packet Loss (%)	0	0	0
EETPC Packet Loss (%)	0	0	0



(a) (b)

Figure 6.21: Comparison Simulation Result of Discovery Process at Strong Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption

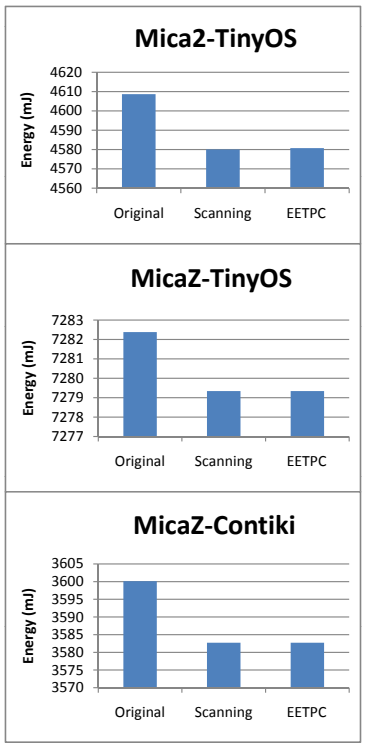


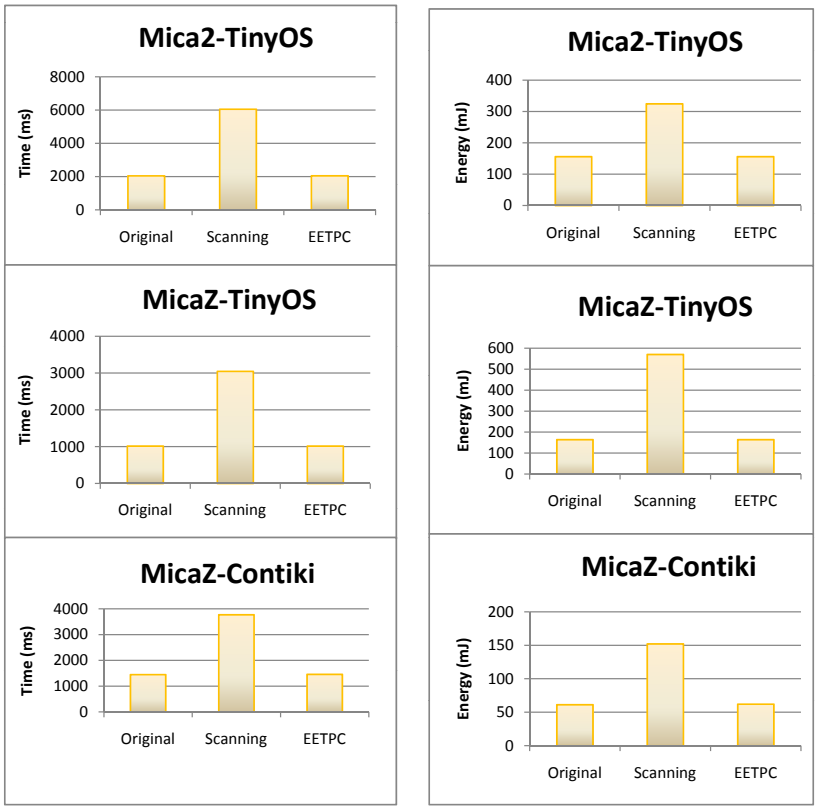
Figure 6.22: Simulation Overall Energy Consumption at Strong Signal Position

- Scenario2: Weak Signal Position

To obtain a weak signal, the sender and the base station are placed at 10 metres distance. The scanning-based approach provides the lowest transmission power level and the lowest overall energy consumption. However, it has more delay and energy consumption in the discovery process. Moreover, packet losses of 4-7% occur for the scanning-based approach, as in Table 6.4.

Table 6.4: Simulation Experimental Result at Weak Signal Position

	Mica2-TinyOS	MicaZ-TinyOS	MicaZ-Contiki
$\overline{RSS_{max}}$ (dBm)	-97	-83	-84
Original TX Power (dBm)	0	0	0
Scanning TX Power (dBm)	-10	-10	-10
EETPC TX Power (dBm)	0	0	0
Original Packet Loss (%)	0	0	0
Scanning Packet Loss (%)	6	4	7
EETPC Packet Loss (%)	0	0	0



(a) (b)

Figure 6.23: Comparison Simulation Result of Discovery Process at Weak Signal Position between Original, Scanning and EETPC (a) Delay (b) Energy Consumption

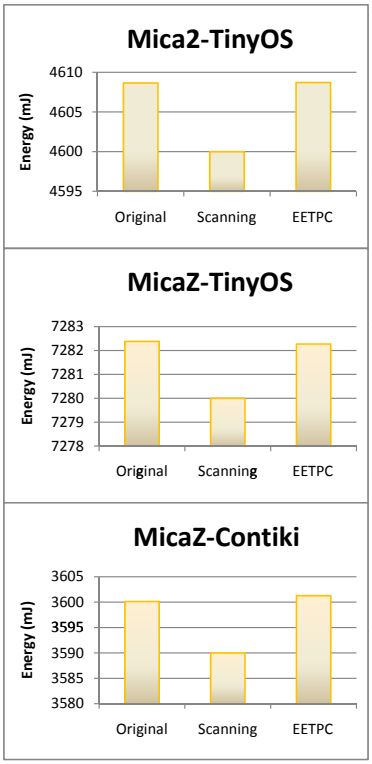


Figure 6.24: Simulation Overall Energy Consumption at Weak Signal Position

- Scenario3: Multihoming Environment

This experiment is conducted on the AVRORA simulator with 16 nodes located randomly in the 8x16m area as shown in Figure 6.25. The experiment is implemented on Rime stack in Contiki OS for MicaZ. All nodes transmit packets at the same channel. To study collision in a controlled manner, carrier sense, random back-off and packet retransmission are disabled to allow free transmission of concurrent packets. Node0 and Node1 are the base stations. All other nodes need to start the ND process to find the base stations. Two techniques are investigated. For the first one, a node chooses the first base station which sends the replied message and then that node sends data packets with the maximum transmission power. For the second one, as the proposed technique for multihoming a node chooses the best base station to which it can send data packets with the lowest transmission power. Topology constructions for these two techniques are shown in Figure 6.26.

As shown in Figure 6.27, the overall packet loss and power consumption rates of EETPC technique are lower than transmission with the maximum powers. Reducing transmission power to the ideal one can decrease the energy consumption of transmission. Moreover, it reduces the collision domain and then allows concurrent transmission of nodes to different base stations, which results in packet loss reduction.

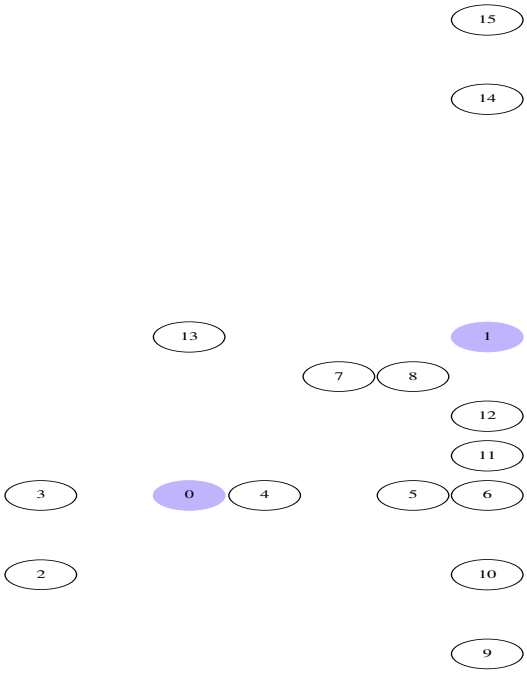
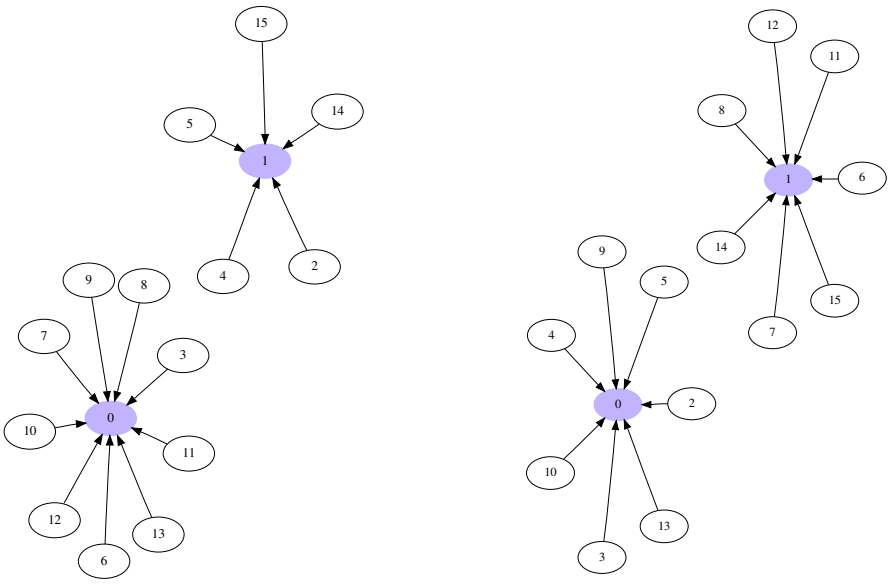


Figure 6.25: 16 Nodes in 8x16m Area



(a) (b)

Figure 6.26: Comparison Topology for Choosing Base Station between (a) First (b) Minimum TX power

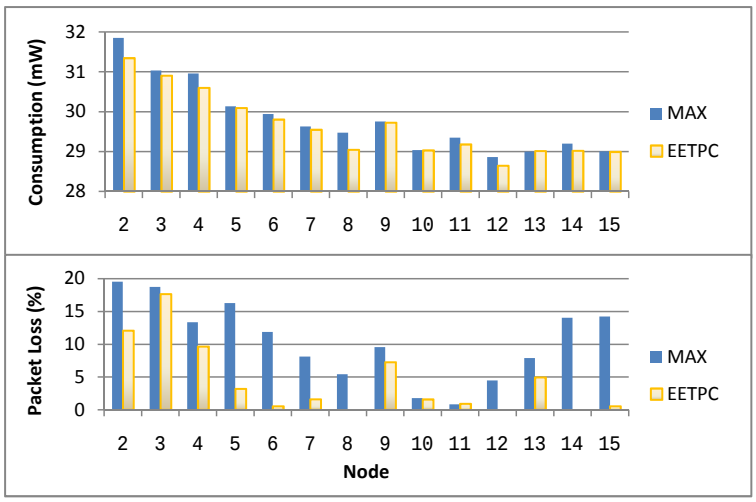


Figure 6.27: Multihoming Network Result

- Scenario4: Multihop Environment

In this experiment, the EETPC technique is implemented for MicaZ platform using UDP protocol and RPL on μIPv6 in Contiki OS. MAC protocol is Null-MAC, which is disabled the carrier sense, random back-off and retransmit functions. All 16 nodes are placed randomly in the 8x16m area at the same positions as in the multihoming experiment. Node0 is configured as root node (also called the base station) in RPL.

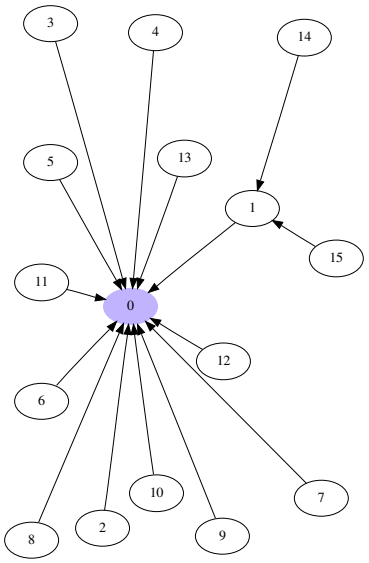


Figure 6.28: Topology Construction by RPL

Two techniques are investigated. For the first one with the original RPL, a node sends packets with the maximum transmission power. For the second one with modified router discovery of RPL, a node sends packets with the ideal transmission power. Since EETPC does not change RPL path selection, topology construction is the same for both original RPL and modified RPL, as shown in Figure 6.28. Overall, the EETPC technique consumes energy power equal to or lower than using the maximum transmission power. Concurrent transmission of nodes occurs between multihop nodes (Node 14 and Node 15) and other single-hop nodes. This is why packet loss reduction occurs significantly at Node14 and Node15.

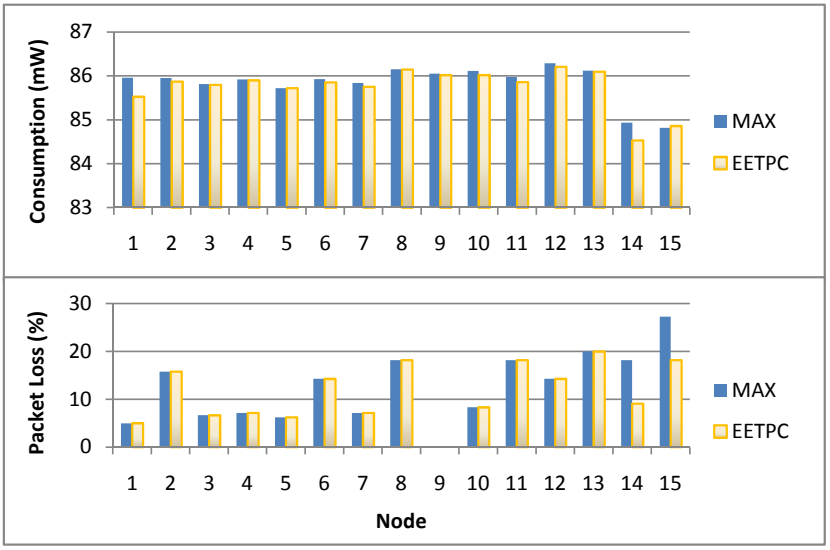


Figure 6.29: Multihop Network Result

6.5.2 Dynamic Transmission Power Adjustment

In these experiments, four techniques are investigated which are static transmission power, RSSI Feedback-based, updated RSS_{max} ($AccRSS_{max}$ as Equation 6.5 with $\alpha = 0.9$) by ND process and temperature effect based on Equation 6.6. The last two techniques are proposed in the EETPC design section. The experiment is implemented on Rime stack in Contiki OS for MicaZ and runs on the AVRORA simulator. Since the AVRORA simulator does not support temperature reading and the temperature effect on RSS, it is modified to include a temperature sensor reading and the temperature effect on RSS. It is assumed that the normal temperature is 25 °C. An equation for calculating new RSS with temperature effect is RSS_{new} = RSS - τ (T-25), where τ is a temperature effect value (e.g. 0.3) and T is the current temperature value. The value of 0.25 is chosen for increasing or decreasing temperature every 5 minutes. The value of noise and interference is

also changed every 5 minutes. These experiments consist of two nodes, called the sender and the base station. The sender starts finding the base station by sending a HELLO message with the maximum transmission power and uses the EETPC equation 6.3 for the ideal transmission power calculation. After completion of the ND process, the sender starts sending a data packets every 3 seconds. Only with the RSSI feedback-based technique, the base station replies back an acknowledgement of each data packet attached with current RSS value for transmission power adjustment. The transmission power will be increased if the current RSS $<$ -90 (assuming -90 is the lower threshold value), while it will be decreased if the current RSS $>$ -75 (assuming -75 is the upper threshold value). For static transmission power, the transmission power is not changed during the experiment period. For updated RSS_{max} by the discovery process, the ND process is repeated every 5 minutes and the new RSS_{max} is obtained to recalculate the ideal power. For temperature effect, the sender reads the temperature value during the discovery process and repeats the temperature reading every 5 minutes for adjusting transmission power based on equation 6.6. Running time is 4 hours for both decreasing and increasing temperature tests.

Scenario1: Decreasing Temperature

RSS increases when temperature decreases. In Figure 6.30 and Table 6.5, static technique uses the static transmission power, therefore it give the worst transmission power level. For the temperature effect technique, it can provide the best transmission power level and lowest energy consumption. However, packet loss occurs since it does not cover the noise and interference changing. The RSSI feedback technique can cover this problem as well as RSS changed due to temperature, but it consumes high energy owing to requiring an acknowledge packet for each sending one. For updated RSS by the discover process, energy consumption is lower than RSSI feedback because it updates the current RSS every 5 minutes.

Table 6.5: Results of Decreasing Temperature

Techniques	Packet Loss (%)	Energy (J)
Static	0	444.51
RSSI Feedback	0	446.50
Updated RSS by Discovery Process	0	444.16
Temperature Effect Equation	1.43	444.00

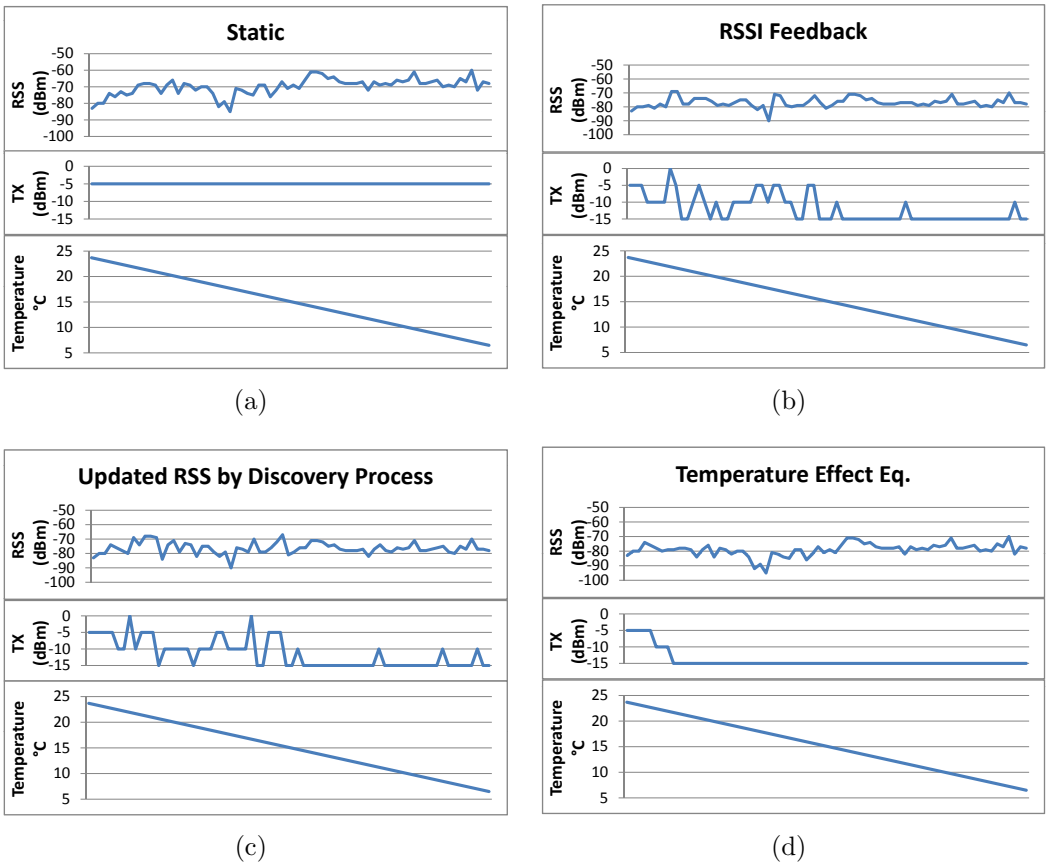


Figure 6.30: Comparison of Decreasing Temperature between (a) Static (b) RSSI Feedback (c) Updated RSS ($AccRSS_{max}$) by Discovery Process (d) Temperature Effect Equation

Scenario2: Increasing Temperature

With increasing temperature in Figure 6.31 and Table 6.6, the static technique does not change the transmission power. Therefore, the highest packet loss occurs. Similar to the reducing temperature test, the RSSI feedback technique can cover RSS changing due to the temperature and interference effects, but it consumes the highest energy. The temperature effect technique provides lower energy consumption but gives the higher packet loss. Updated RSS by the discovery process technique gives a compromise between energy consumption and packet loss.

Table 6.6: Results of Increasing Temperature

Techniques	Packet Loss (%)	Energy (J)
Static	6.06	444.00
RSSI Feedback	0	446.41
Updated RSS by Discovery Process	1.52	444.08
Temperature Effect Equation	3.03	444.00

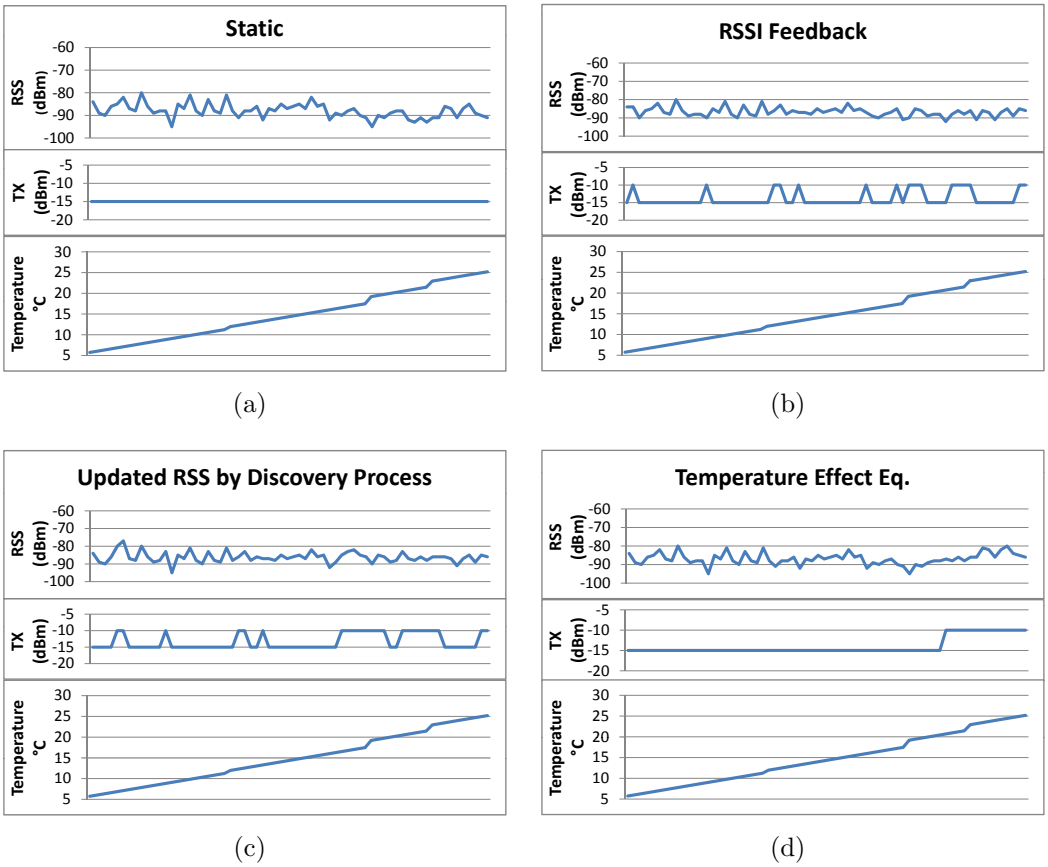


Figure 6.31: Comparison of Increasing Temperature between (a) Static (b) RSSI Feedback (c) Updated RSS by Discovery Process (d) Temperature Effect Equation

6.6 Summary and Discussion

Several factors, such as distance, transmission power, height above ground, multipath reflection environment, nodes capabilities and temperature, are investigated and analysed. RSS estimation equation for any temperatures is proposed to explain the effect of this impact factor. Normally, RSS decreases as the distance increases, but multipath refection can amplify or weaken it. Moreover, RSS can be affected by an antenna which depends on its effective length or effective height. Reducing the effective length of the antenna decreases the signal strength. Placing a node high above the ground may help to get the best reception since it increases the effective height of an antenna. Due to the capability of the node, the same path loss model cannot be used for all sensors, even if they have the same hardware. A high temperature decreases the signal strength of both the sender and receiver, while RSS increases when temperature decreases. Therefore, compensation may be required for the signal strength dropping or increasing due to temperature change.

The RSS value for each transmission power level can be estimated by using

either the distance-based path loss model or the RSS estimation proposed in the EETPC design. The distance-based approach does not cover multipath, while the EETPC estimation technique does. Experimental results show that RSS estimation by using the distance-based path loss may need to be adjusted for each node's capabilities and multipath environments which are normally found in real world situations. Moreover, for cover with other factors, it requires many complex factor values to be computed, such as the distance between sender and receiver, antenna gain, radio frequency and so on. In contrast, the EETPC equation, as proposed in this work, uses only the measured RSS_{max} value for calculation. This measured RSS_{max} can be automatically obtained during the neighbour discovery process.

For finding the ideal transmission power, the discovery process has been modified for providing RSS_{max} value without adding new steps or messages. This RSS value is then used for finding the transmission power based on EETPC for the ideal transmission power equation. The equation adds a value X to cope with signal fluctuation and the accuracy of RSSI measurement by the device. Three performance metrics: delay, energy consumption and the packet loss rate, are investigated. Both scanning-based and EETPC techniques can provide the ideal transmission power and lowest energy consumption. However, at the weak signal point, scanning-based approach has a long delay in the discovery process for finding the ideal power. This long delay leads to an increase of the energy consumption for the discovery process and causes a delay in the start of data packet sending. Moreover, the scanning-based may obtain too low transmission power levels which can cause a greater packet loss rate. When applying the EETPC technique for multihoming and multihop networks, it gives lower packet loss and power consumption rates compared to the original technique sending with maximum transmission power. The EETPC technique provides an ideal transmission power which leads to a decrease in the transmission energy consumption. Moreover, it reduces the collision domain and then allows concurrent transmission of nodes in the network, which results in packet loss reduction.

For dynamic power adjustment, two techniques are proposed: the temperature effect equation and updated RSS by the repeating discovery process. These two techniques are compared with the RSSI feedback and static transmission power techniques. Since the static technique uses the static transmission power, it gives the worst transmission power level. As for the temperature effect technique, it can provide better transmission power level and lower energy consumption. However, packet loss may occur since it does not cover the environment changing, such as noise and interference. The RSSI feedback technique can cover environment changing, but it consumes high energy owing to requiring an acknowledge packet for each sending one. Updated RSS by the discovery technique gives a compromise

between energy consumption and packet loss and seems to be the best solution. However, tradeoff between energy consumption and packet loss is needed. More frequent updates lead to the consumption of more energy, like the RSSI feedback technique, while less frequent updates lead to packet loss, as with the static transmission power technique. For less frequent updates, combining between the updated RSS by the discovery process and the temperature effect techniques may give a better performance.

Chapter 7

DNLE: Dynamic Node Lifetime Estimation

7.1 Introduction

This research proposes formulas and processes for lifetime estimation which covers many factors causing the different lifetime periods, such as battery type, model, brand, self-discharge, discharge rate, age and temperature. These formulas do not require detailed electrical and electrochemical knowledge. Moreover, some limitations, such as memory and arithmetic capabilities, have to be taken into account for real implementation on sensor nodes. In this chapter, the impact factors on node lifetime are presented. Then, it describes the proposed dynamic lifetime estimation including an implementation on real hardware and software platforms in WSNs. Finally, the accuracy of the proposed schemes is validated by a real hardware testbed with different scenarios.

7.2 The Impact Factors on Node Lifetime

Many factors causing the different lifetime periods, such as battery types, models, brands, self-discharge, discharge rate, ageing, charge cycles and temperature, are investigated and analysed in this section.

7.2.1 Battery Types, Brands and Models

The quoted capacity of two AA-size battery types: alkaline and NiMH, from several brands and models is shown in Table 7.1 [2, 13, 11, 3].

Based on the vendor data sheet, an AA-LR6 alkaline battery capacity for discharging at 100 mA to 0.9 V cut-off is around 2750 mA h [2, 13], while NiMH battery capacity is around 110% of the battery models, e.g. D-NiMH2100 model

Table 7.1: Quoted Capacity for 100 mA Discharge to 0.9 V Cut-off at Room Temperature

Brands	Types	Models	Quoted Capacity (mA h)
A	Alkaline(AA-LR6)	A-Alkaline	2750
B	Alkaline(AA-LR6)	B-Alkaline	2750
C	Alkaline(AA-LR6)	C-Alkaline	2750
B	NiMH(AA-HR6)	B-NiMH1300	1430
B	NiMH(AA-HR6)	B-NiMH2000	2200
D	NiMH(AA-HR6)	D-NiMH2100	2310
D	NiMH(AA-HR6)	D-NiMH2500	2750

has 2310 mA h of quoted capacity. However, only 90% from quoted capacity can be used for discharging to a cut-off voltage of 1 V for alkaline batteries, while 93% can be used for NiMH batteries [2, 13, 11, 3]. From 2.6, lifetime can be easily estimated by dividing that battery capacity with the current load and assuming $k = 1$. A constant 100 mA load generated by the CBA-III load tester is used for testing a pair of AA batteries at room temperature (22 °C). For NiMH batteries, they are charged by 500 mA charger and they are tested after being fully charged. The tested batteries are discharged until their terminal voltages reach the cut-off voltage at 2.0 V (of 2 cells). Table 7.2 shows the real measurement and estimation of lifetime. It is obvious that the battery model with high capacity has longer lifetime than the model with low capacity. With the same model of alkaline batteries, lifetime values of different brands have a small difference (±16 min). The differences between estimated lifetime based on vendor data sheet and real measurement lifetime are around 7 to 15%. These differences may be caused by other factors, such as self-discharging, ageing and charge cycles which will be described in the following subsection.

Table 7.2: Measured and Estimated Lifetime of 100 mA Discharge to 2.0 V Cut-off at Room Temperature

Batteries	Lt (min)		Deviation
	Measured	Estimated	(%)
A-Alkaline	1384	1485	7.3
B-Alkaline	1370	1485	8.4
C-Alkaline	1368	1485	8.6
B-NiMH1300	694	796	14.8
B-NiMH2000	1144	1224	7.3
D-NiMH2100	1115	1285	11.5
D-NiMH2500	1410	1530	8.8

7.2.2 Self-discharge

Since alkaline batteries will lose approximately 2% of their capacity per year when stored at 20 °C due to the self-discharging process [2, 13], there is no significant effect on capacity for several months for this type of battery. This means that the starting voltage of alkaline batteries can be assumed to be 1.5V for each cell. In contrast, NiMH batteries have significantly high self-discharge of 20% on the first day and 1-4% per subsequent day. With self-discharging, they lose capacity and their voltages also drop. To study the relationship between starting terminal voltage (Vt) and lifetime, a CBA-III load tester with static 100 mA load testing runs with a pair of AA NiMH batteries from two brands, B-NiMH2000 and D-NiMH2500, on two different starting Vt at room temperature (22 °C). Lifetime periods from the starting Vt to the minimum Vt (cut-off at 2.0 V) are shown in Table 7.3.

Table 7.3: Lifetime of 100 mA Load with Two AA NiMH Batteries

Battery	Starting Vt (V)	Measured Lt (min)
B-NiMH2000	2.81	1111
	2.88	1144
D-NiMH2500	2.71	1197
	2.85	1264

From this experiment, it is obvious that high starting Vt gives more lifetime period than low starting Vt. Higher starting Vt means the higher battery energy results in longer lifetime. The proposed lifetime equation for any starting Vt is:

$$Lt(Vt) = \left(-\tau \cdot \ln\left(\frac{Vt_r}{Vt}\right)\right) + Lt(Vt_r) \tag{7.1}$$

where τ is the time constant representing capacity affected by self-discharging and $Lt(Vt_r)$ is the lifetime of the reference voltage (Vt_r). The time constant τ can be obtained by the following equation [18].

$$\tau = \frac{t}{\ln\left(\frac{Vt_{r1}}{Vt_{r2}}\right)} \tag{7.2}$$

where t is the different time period between starting at Vt_{r1} and Vt_{r2} which is $Lt(Vt_{r1}) - Lt(Vt_{r2})$. From Table 7.3, t of B-NiMH2000 is -33 min, while t of D-NiMH2500 is -67 min. Vt_{r1} and Vt_{r2} of B-NiMH2000 are 2.81 V and 2.88 V, while they are 2.71 and 2.85 V for D-NiMH2500. Therefore, τ, Vt_r and $Lt(Vt_r)$ for B-NiMH2000 are 1341, 2.81, and 1111, while they are 1330, 2.71 and 1197 for D-NiMH2500. These values are then used for calculating lifetime (in minutes) with

any Vt between Vt_{r1} and Vt_{r2} based on 7.1. Measured and estimated lifetime of B-NiMH2000 and D-NiMH2500 batteries with different starting voltages are shown in Table 7.4 and 7.5. The different starting voltages are caused by the self-discharging periods after full charging (1 h, 2 h, 3 h, 4 h and 5 h).

Table 7.4: Measured and Estimated Lifetime for Different Starting Voltage of B-NiMH2000 Batteries

Voltage	Self-discharge	Lt (min)		Deviation
(V)	time (h)	Measured	Estimated	(%)
2.86	1	1138	1134	-0.3
2.85	2	1133	1130	-0.3
2.84	3	1127	1125	-0.2
2.84	4	1121	1125	0.4
2.83	5	1117	1121	0.3

Table 7.5: Measured and Estimated Lifetime for Different Starting Voltage of D-NiMH2500 Batteries

Voltage	Self-discharge	Lt (min)		Deviation
(V)	time (h)	Measured	Estimated	(%)
2.79	1	1250	1236	-1.1
2.78	2	1242	1231	-0.9
2.77	3	1235	1226	-0.7
2.76	4	1225	1221	-0.3
2.76	5	1223	1221	-0.1

7.2.3 Discharge Rate

The capacity of the battery varies with the rate of discharge. From the vendor data sheet with the discharge rate of 25 mA to 100 mA [2, 13, 11], the capacity of the battery increases when the discharge rate decreases for alkaline batteries, while it decreases when the discharge rate decreases for NiMH batteries. If C in 2.6 is the capacity of batteries when they are discharged at 100 mA, peukert constant (k) values are 0.96 and 1.004 for alkaline AA-LR6 and NiMH batteries, respectively [2, 13, 11]. These k constants are for a discharge rate of less than 100 mA. For example, if lifetime for a 100 mA discharge rate for alkaline batteries is 25 h (2500 mA h capacity), the lifetime for a 25 mA discharge rate will be 114 h (calculated from $\frac{2500}{25^{0.96}}$); and if a lifetime for a 100 mA discharge rate for NiMH batteries is 27.5 h (2750 mA h capacity), lifetime for 25 mA of discharge rate will be 108.5 h (calculated from $\frac{2750}{25^{1.004}}$).

7.2.4 Ageing

The capacity of batteries will vary depending on their ageing. Normally, battery packages have a date code on them indicating the use by date that the batteries can be used with good capacity (70-80% of quoted capacity) [11, 13, 18]. Therefore, the ageing effect on lifetime is presented as:

$$Lt(a) = (1 - \rho \cdot a) \cdot Lt(new) \tag{7.3}$$

where a is the age of the battery in years, ρ is the ageing factor and $Lt(new)$ is the lifetime of the new battery. For examples, the capacity of alkaline batteries will reduce 2% per year [2, 13] when stored at room temperature (20 °C). This means that the lifetime of the device will be reduced 2% ($\rho = 0.02$) each year due to the age of the batteries. Therefore, the new AA-LR6 batteries will give 1350 min of lifetime for a device with 100 mA load, while they will give 1323 min of lifetime when storing these batteries for one year before using. For NiMH batteries, the capacity will drop 6% ($\rho = 0.06$) per year due to their ageing when stored at room temperature [3, 11, 13]. Therefore, the new D-NiMH2500 battery will give 1410 min after being fully charged, while they will give 1325 min after being fully charged if they have one year of age.

7.2.5 Charge Cycles

The charge cycles factor has an effect on rechargeable batteries by measuring how many times the batteries have been recharged. NiMH batteries can deliver 100% capacity for up to 300 charge cycles, while their capacity will reduce 0.1% for each charge cycle for more than 300 cycles [11, 3]. As a result, the decrease of lifetime can be described as:

$$Lt(c) = \begin{cases} Lt(new) & \text{if } c \leq 300 \\ (1.3 - 0.001c) \cdot Lt(new) & \text{otherwise} \end{cases} \tag{7.4}$$

where c is the number of charge cycles and $Lt(new)$ is the lifetime of the new battery. For example, the new D-NiMH2500 battery will give 1410 min, while they will give 1269 min at 400 charge cycles ($c = 400$).

7.2.6 Temperature

Both battery capacity and current consumption are affected by temperature. As temperature increases, the current draw is increased and, consequently, shortens the lifetime of the device. However, for alkaline and NiMH cells, high temperature

can provide increased capacity of battery over the operating temperature range 0 to 40 °C. Therefore, it depends on the ratio between increased battery capacity and current draw, which may result in increased or decreased lifetime. Based on 4.2 and information contained in vendor data sheets [101, 2, 13, 11, 3], the constant B which depends on the activation energy and gas of the battery can be calculated as Table 7.6.

Table 7.6: Capacity of Different Temperatures and Constant Value for Different batteries

Batteries	Capacity (mA h@0 °C)	Capacity (mA h@25 °C)	B
A-Alkaline	900	1950	2518.73
B-Alkaline	857	1942	2665.53
B-NiMH2000	1717	2113	677.19
D-NiMH2500	2127	2667	737.11

Based on Equation 4.5, the leakage current draw at two different average temperatures in Kelvin (T_1, T_2) can be defined as:

$$\frac{I(T_2)}{I(T_1)} = \frac{T_2^2}{T_1^2} \cdot \exp\left(\frac{S \cdot (T_1 - T_2)}{T_1 \cdot T_2}\right) \tag{7.5}$$

where S is a constant depending on the circuit and V_{dd}. It is assumed that V_{dd} is the same for these two different temperatures. The real experiments have been conducted for Mica2 and N740 Nanosensor with a simple static load programme at two different temperatures. The V_{dd} is static at 3.0 V. Table 7.7 shows the measured current draw by an ammeter and calculated S constant results.

Table 7.7: Current Draw at Different Temperatures and S Value for Mica2 and N740 Motes

Sensor	Current (mA@5 °C)	Current (mA@25 °C)	S
Mica2	12	18	-1105.43
N740	47	66	-831.94

From Equation 2.6, 4.2 and 7.5, node lifetime with full capacity at two different temperatures is derived as:

$$\frac{Lt(T_2)}{Lt(T_1)} = \frac{T_1^2}{T_2^2} \cdot \exp\left(\sigma \cdot \left(\frac{T_2 - T_1}{T_2 \cdot T_1}\right)\right) \tag{7.6}$$

where σ is the $B + S$ values which can be obtained from Table 7.6 and 7.7 and are presented in Table 7.8.

Table 7.8: Ratio Constant Value of Different Batteries for Mica2 and N740 Motes

Sensor	Batteries	σ
Mica2	A-Alkaline	1413.30
Mica2	B-Alkaline	1560.10
Mica2	B-NiMH2000	-428.24
Mica2	D-NiMH2500	-368.33
N740	A-Alkaline	1686.80
N740	B-Alkaline	1833.59
N740	B-NiMH2000	-154.74
N740	D-NiMH2500	-94.83

The experiments are conducted to test the temperature effect on the lifetime of N740 mote with two NiMH batteries: B-NiMH2000 and D-NiMH2500. A simple programme runs on different temperatures. This programme load as measured by an ammeter at 25 °C is 67 mA. From Table 7.6, capacity of B-NiMH2000 and D-NiMH2500 batteries at 25 °C are 2113 and 2667 mA h. Therefore, node lifetime at 25 °C will be 31.01 h ($\frac{2113}{67^{1.004}}$) and 39.14 h ($\frac{2667}{67^{1.004}}$) for B-NiMH2000 and D-NiMH2500, respectively. However, the experiments run at temperatures between 19 to 24 °C. The results of real lifetime and estimated lifetime based on 7.6 including deviation are presented in Table 7.9.

Table 7.9: Lifetime with Different NiMH Batteries and Temperatures for N740 Motes

Batteries	Temperature	*Lt* (min)		Deviation
	(°C)	Measured	Estimated	(%)
B-NiMH2000	20.0	1953	1941	-0.61
B-NiMH2000	21.0	1920	1924	0.21
B-NiMH2000	24.0	1885	1875	-0.53
D-NiMH2500	19.5	2462	2451	-0.45
D-NiMH2500	20.0	2445	2441	-0.16
D-NiMH2500	22.0	2417	2403	-0.58

7.3 DNLE Design and Implementation

DNLE is based on the following assumptions for both alkaline and NiMH batteries. First, batteries from the same pack have the same performance and behaviour. Second, storing temperature does not affect the capacity change since the batteries are always stored at room temperature. Third, for NiMH batteries, charging method does not affect the capacity change since the same charger is used in these experiments. Fourth, temperature during charging does not affect the capacity

change since the batteries are always charged at room temperature. Last, batteries have suffered fewer than 300 charging cycles.

7.3.1 DNLE Design

The following steps are proposed for finding the lifetime of each node:

Step 1: Finding the real battery capacities

- Alkaline Batteries

It is sometimes difficult to determine the battery capacity reduction due to ageing since the manufacture date is not available. Moreover, some batteries may have lower capacity than quoted. It can be assumed that batteries sampled from the same manufactured batch can be representative of that batch. Each sample pair is tested by the load testing with static 100 mA load in order to find the real battery capacity with cut-off voltage equal to the minimum voltage of sensor motes. In these experiments, the cut-off voltage is 2.2 V. After that, the real battery capacity (in mA h) can be calculated by:

$$C = Lt \cdot 100 \tag{7.7}$$

where Lt is the lifetime of the batteries in hours. For example, if a pair of alkaline batteries gives 22.5 h of lifetime, the capacity value is 2250 mA h. Instead of using quoted capacity, a sample average will be assigned as the capacity for batteries from this batch. The temperature (T_t) when running load testing is recorded. If T_i is the temperature when a node starts running, the initial capacity with temperature effect can be calculated by using 4.2 with T_2 as T_i and T_1 as T_t. We refer to this value as $C(T_i)$.

- NiMH Batteries

For NiMH batteries, each sample pair is tested after full charging in a similar way to the alkaline type. However, since NiMH batteries have significantly high self-discharge on the first day, each sample pair is retested again after letting it 24 h of self-discharge. With self-discharging, the starting voltage of batteries is different from being fully charged. The starting voltage of fully charged batteries is referred to as Vt_f and the starting voltage of batteries with 24 h of self-discharge is referred to as Vt_{24h}. The capacities of two different starting voltages are referred to as C_f for the capacity of fully charged batteries and C_{24h} for the capacity of

24 h of self-discharge batteries. Then, the battery capacity of any starting voltage (Vt) between Vt_{24h} and Vt_f can be calculated as:

$$C(Vt) = \left(-\tau \cdot \ln\left(\frac{Vt_{24h}}{Vt}\right)\right) + C(Vt_{24h}) \tag{7.8}$$

where τ is the constant representing capacity affected by self-discharging which can be obtained by:

$$\tau = \frac{c}{\ln\left(\frac{Vt_{24h}}{Vt_f}\right)} \tag{7.9}$$

where c is $C_{24h} - C_f$. It is assumed that the temperature (T_t) is the same when running load testing with fully charged and 24 h of self-discharge batteries. If T_i is the temperature when a node starts running, the initial capacity with temperature effect can be calculated by using 4.2 with T_2 as T_i and T_1 as T_t. We refer to this value as $C(Vt, T_i)$.

Step2: Finding the current consumption every period

Adapted from Equation 4.3 proposed by Dunkels et al., the current consumption at a period can be calculated as:

$$I = I_a t_a + I_l t_l + I_t t_t + I_r t_r + \sum I_c t_c \tag{7.10}$$

where I_a and t_a are the current consumption of the MCU (Microprocessor Control Unit) and the time when the MCU has been running in active mode during a period; I_l and t_l are the current consumption and time of the MCU in low power or sleep mode during a period; I_t and t_t are the current consumption and the time of the radio transceiver in transmit mode during a period; I_r and t_r are the current consumption and time of the radio transceiver in receive mode during a period; I_c and t_c are the current consumption and time of other components such as sensors and LEDs during a period. All current consumption values in this equation is based on the device data sheet with temperature at 25 °C. The current consumption has to be calculated every period. Thus, the average current consumption ($\overline{I}$) for s periods can be computed by using the Simple Moving Average as:

$$\overline{I}(s) = \gamma \cdot \overline{I}(s-1) + (1-\gamma) \cdot I(s) \tag{7.11}$$

where γ is the weighting value $= \frac{s-1}{s}$, $\overline{I}(s-1)$ is average current consumption of the previous period, and $I(s)$ is the current consumption at the period s. If T is the temperature of the current period, the current consumption $I(s, T)$ and

average current consumption $\overline{I}(s,T)$ can be calculated by using 7.5 with T_2 as T and T_1 as 298.15 (25 °C is 298.15 K).

Remaining lifetime estimation

The remaining capacity (C_{rem}) at period s can be obtained by:

$$C_{rem}(s) = C_{rem}(s-1) - I(s,T)^k \tag{7.12}$$

where k is the peukert constant, $C_{rem}(s-1)$ is the remaining capacity before period s and it is equal to the initial capacity for the first period. This means that if $s = 1$, $C_{rem}(s-1) = C(T_i)$ for alkaline batteries and $C_{rem}(s-1) = C(Vt, T_i)$ for NiMH batteries. If T is the temperature of the current period, the temperature effect on the capacity is calculated by using 4.2, with T_2 as T and T_1 as the temperature of the previous period. We refer to this value as $C_{rem}(s,T)$. Finally, the remaining lifetime at the current period and temperature Lt_{rem}(s,T) can be estimated as:

$$LTrem(s,T) = \frac{C_{rem}(s,T)}{\overline{I}(s,T)^k} \tag{7.13}$$

7.3.2 DNLE Implementation

The proposed steps and equations for DNLE have been implemented on real hardware platforms: Mica2 and N740 NanoSensor. Two existing operating systems: TinyOS and Contiki have been modified. It was assumed that sensor motes support battery voltage and temperature reading. The process is as follows:

Process1: Preconfigured and Starting process

Two AA-LR6 alkaline models (A-Alkaline and B-Alkaline) and two NiMH models (B-NiMH2000 and D-NiMH2500) are the batteries used in the experiments. The capacities of these battery packs are evaluated by using the CBA-III load tester with static 100 mA current at 25 °C. Three pairs of batteries are tested for finding the average values of each battery model. These average capacities are then used as the preconfigured values for sensor nodes. All preconfigured values are given in Table 7.6, 7.7 and 7.10. All nodes may automatically obtain these preconfigured values during the discovery process, but it is beyond the scope of this research. When a node starts, it reads the current temperature and battery starting voltage in order to calculate the initial battery capacity as described in Step1 of the DNLE design section.

Table 7.10: Preconfigured Capacities of Batteries

Battery	Capacity at 25 °C (mA h)
A-Alkaline	1850
B-Alkaline	1800
B-NiMH2000	1585
D-NiMH2500	1814

Process2: Looping process

For finding the current consumption, three components: MCU, radio transceiver and sensing device, are focused on at runtime. The current draw of these components is configured based on the device data sheet. The device driver of these components is modified: time stamps are recorded when the components are turned on, and time differences are computed when the components are turned off. These time differences (with the same unit of the period) are used for the current consumption calculation based on Equation 7.10. One minute is used as the time period (Step2 of the DNLE design section). The battery capacity unit needs to be converted to the same unit of the period (e.g. for a period of 1 minute, 2000 mA h is converted to 120 000 mA min). Temperature readings are repeated every minute for updating the current temperature in order to calculate the effect on the current consumption, average current consumption, remaining capacity and remaining lifetime as described in Step2 and Step3 of the DNLE design section.

7.4 DNLE Evaluation

This section evaluates the proposed technique on real testbed experiments with two different hardware platforms (Mica2 and N740) and four battery models: A-Alkaline, B-Alkaline, B-NiMH2000 and D-NiMH2500. The N740 platform is implemented using Contiki OS, while Mica2 platform is implemented using TinyOS. Two techniques, Selvig's and the DNLE, are investigated. Lifetime estimation is made every minute. For Selvig's method, battery capacity is obtained by using quoted capacity as in Table 7.1, while the current consumption is calculated using Equation 7.10. In these experiments 80 % of quoted capacity is used as a cut-off voltage of 2.2 V for alkaline batteries and 91 % for NiMH batteries [2, 13, 11, 3]. Therefore, usable capacities are 2200 mA h for A-Alkaline, 2200 mA h for B-Alkaline, 2102 mA h for B-NiMH2000 and 2502 mA h for D-NiMH2500. For DNLE, the processes are as described in the DNLE design and implementation section. In general, the current consumption of the sensor node is less than 100 mA. Therefore, the peukert constant values are 0.96 and 1.004 for the alkaline and NiMH

batteries, respectively.

These experiments consist of two nodes, the sender and the base station. The base station is placed in the normal room conditions, while the sender is placed in a controlled environment in order to operate under a static temperature (10 °C and 22 °C) as shown in Figure. 7.1. Two scenarios, static and dynamic loads, are applied in the experiments.

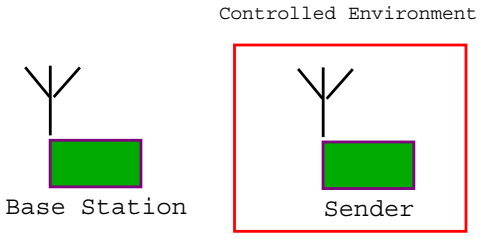


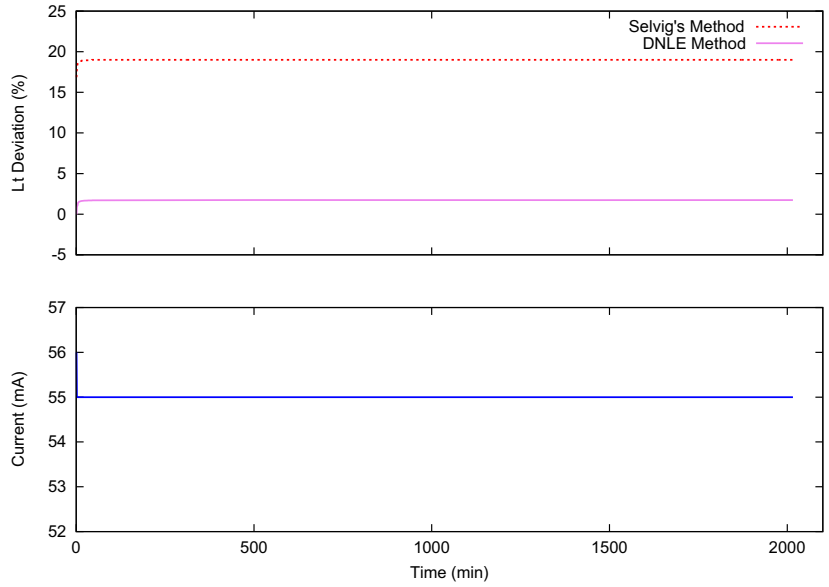
Figure 7.1: Dynamic Noad Lifetime Estimation Experiment

7.4.1 Scenario1: Static Load

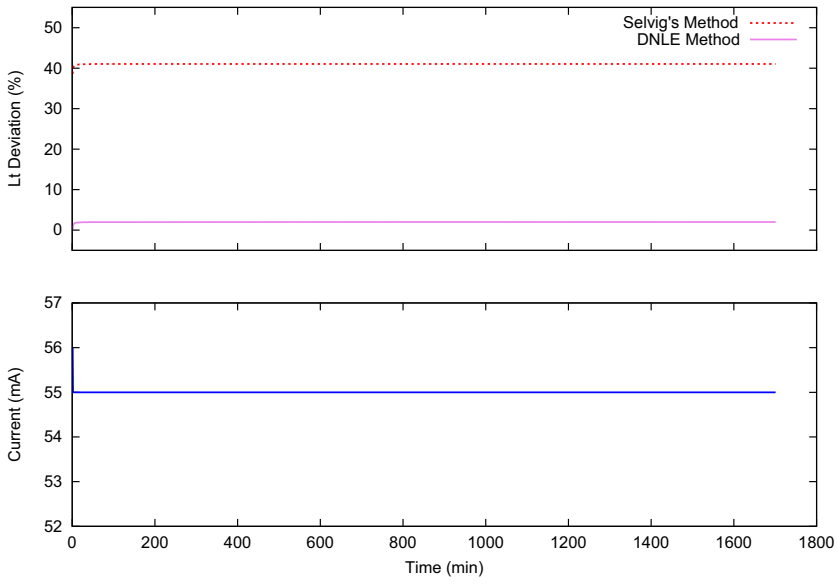
For static load, the sender sends a fixed sized packet of 50 B to the base station every 3 s. Two sensor motes (Node1 and Node2) are used. Node1 runs as the sender for the first 3 times, while Node2 is the sender for the other 3 times. Lifetime estimation is calculated only at the sender node. NiMH batteries are given random starting voltages by a random self-discharging period of 5 min to 600 min. An example deviation of lifetime result details for N740 with A-Alkaline batteries in every minute at two temperatures is illustrated in Figure 7.2.

When the sender node starts, it consumes a slightly higher current for starting up all the components. After that, the current consumption reduces and becomes static. Based on data sheet and 7.10 for three focused components, the current consumption for N740 is 56 mA for the first minute, while it is a static of 55 mA for every minute of the rest. After applying the effect of temperature, the current consumption at the first minute is 53.34 mA for temperature at 22 °C and it is 43.56 mA43.56 at 10 °C. Comparing to the real measurement lifetime at 22 °C, the deviation of remaining lifetime estimation in every minute for Selvig's method is 16.9 % to 19.0 %, while it is −0.1 % to 1.8 % for the DNLE technique. The average deviation at 22 °C of N740 with A-Alkaline batteries is 18.9% for Selvig's method, while it is 1.7% for DNLE. At 10 °C, the deviation of remaining lifetime estimation for Selvig's method is 38.5 % to 41.1 %, while it is 0.3 % to 2.1 % for the DNLE technique. The average deviation at 10 °C of N740 at with A-Alkaline batteries is 41.0% for Selvig's method, while it is 2.0% for DNLE. The average

deviation results for two mote platforms with different battery models are shown in Table 7.11. Since Selvig's method does not take some impact factors (such as charging rate, self-discharging and temperature) into account, their estimated values are very different from the measured ones around 1.9 % to 60.3 %, while they are in the range of −3.5 % to 2.4 % for DNLE.



(a)



(b)

Figure 7.2: The Deviation of Remaining Lifetime Estimation for Static Load Testbed of N740 with A-Alkaline Batteries at (a) 22 °C (b) 10 °C

Table 7.11: Average Deviation of Static Load Lifetime Estimation Testbed Results for Mica2 and N740 Motes with Different Batteries

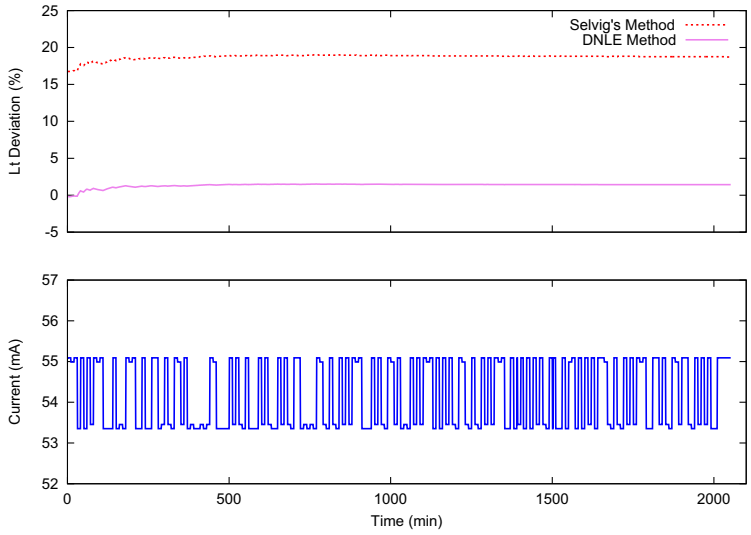
Sensor	Battery	Temp.	Estimated Lt Deviation(%)	
		(°C)	Selvig's Method	DNLE
Mica2	A-Alkaline	22	22.4	-1.0
Mica2	A-Alkaline	10	38.9	-1.4
Mica2	B-Alkaline	22	30.4	-2.5
Mica2	B-Alkaline	10	50.0	-3.5
Mica2	B-NiMH2000	22	60.3	1.5
Mica2	B-NiMH2000	10	33.3	2.4
Mica2	D-NiMH2500	22	53.4	-1.4
Mica2	D-NiMH2500	10	36.6	0.8
N740	A-Alkaline	22	19.1	1.7
N740	A-Alkaline	10	41.1	1.9
N740	B-Alkaline	22	26.3	-0.2
N740	B-Alkaline	10	50.9	-1.3
N740	B-NiMH2000	22	23.5	1.0
N740	B-NiMH2000	10	21.7	2.0
N740	D-NiMH2500	22	41.8	2.1
N740	D-NiMH2500	10	39.5	2.0

7.4.2 Scenario2: Dynamic Load

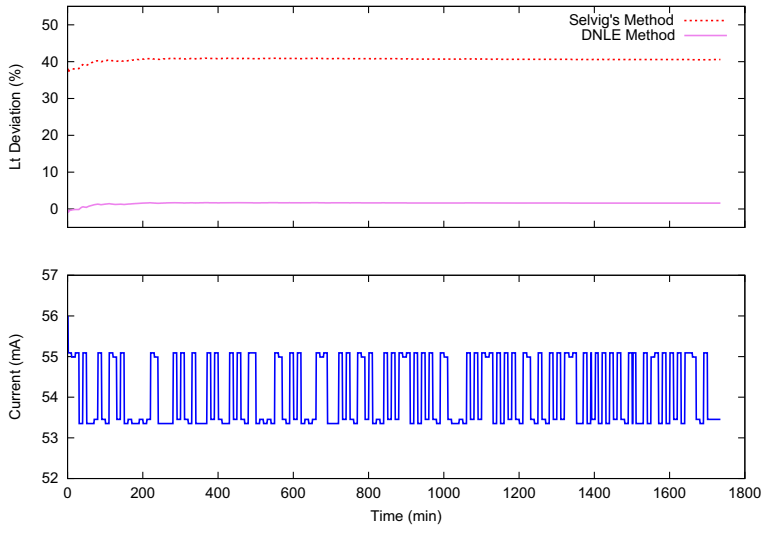
In the real deployment, the load of each sensor is normally dynamic, which may be caused by monitoring events or the number of neighbour nodes. For dynamic load in this experiment, the sender sends a packet of a random size of 20 B to 50 B to the base station every x seconds. The interval time value (x) was changed randomly every 10 min to between 3 s and 6 s. Two sensor motes (Node1 and Node2) are used. Node1 runs as the sender for the first 3 times, while Node2 is the sender for the other 3 times. Lifetime estimation is calculated only at the sender node. For NiMH batteries, they have random starting voltages by a random self-discharging period between 5 min to 600 min. An example deviation of lifetime result details for N740 with A-Alkaline batteries in every minute at two temperatures is illustrated in Figure 7.3.

With this scenario, the current consumption is dynamic. Based on data sheets and Equation 7.10 for three focused components, the current consumption of the N740 is 56 mA for the first minute and 53 mA to 55 mA for the subsequent minutes. After applying the effect of temperature, the current consumption at the first minute is 53.34 mA at 22 °C and it is 43.56 mA at 10 °C. Comparing to the real measurement lifetime at 22 °C, the deviation of remaining lifetime estimation in every minute for Selvig's method is 16.8 % to 20.5 %, while it is −0.2 % to 2.8 % for the DNLE technique. The average deviation at 22 °C of N740 with A-Alkaline

batteries is 18.8% for Selvig's method, while it is 1.5% for DNLE. At 10 °C, the deviation of remaining lifetime estimation for Selvig's method is 36.9 % to 40.9 %, while it is −0.9 % to 1.7 % for DNLE. The average deviation at 10 °C of N740 with A-Alkaline batteries is 40.6 % for Selvig's method, while it is 1.6% for DNLE. The average deviation results for two mote platforms with different battery models are shown in Table 7.12. The deviation Selvig's method is 18.7 % to 57.5 %, while it is −2.1 % to 2.5 % for DNLE.



(a)



(b)

Figure 7.3: The Deviation of Remaining Lifetime Estimation for Dynamic Load Testbed of N740 with A-Alkaline Batteries at (a) 22 °C (b) 10 °C

Table 7.12: Average Deviation of Dynamic Lode Lifetime Estimation Testbed Results for Mica2 and N740 Motes with Different Batteries

Sensor	Battery	Temp.	Estimated Lt Deviation(%)	
		(°C)	Selvig's Method	DNLE
Mica2	A-Alkaline	22	21.0	-1.4
Mica2	A-Alkaline	10	37.8	-1.4
Mica2	B-Alkaline	22	30.6	-1.6
Mica2	B-Alkaline	10	57.5	2.1
Mica2	B-NiMH2000	22	33.9	2.5
Mica2	B-NiMH2000	10	22.4	-1.8
Mica2	D-NiMH2500	22	54.7	-0.6
Mica2	D-NiMH2500	10	32.8	-2.1
N740	A-Alkaline	22	18.7	1.4
N740	A-Alkaline	10	40.5	1.4
N740	B-Alkaline	22	26.0	-0.5
N740	B-Alkaline	10	50.2	-1.7
N740	B-NiMH2000	22	33.6	0.7
N740	B-NiMH2000	10	21.2	1.6
N740	D-NiMH2500	22	41.4	1.8
N740	D-NiMH2500	10	35.9	1.5

7.5 Summary and Discussion

The impact factors on node lifetime, such as battery brand, type, model, self-discharge, charging rate, battery ageing, charge cycles, and temperature, are investigated and analysed. Lifetime equations for any starting voltage, ageing, charge cycles and temperatures are proposed to explain the effect of the impact factors. Batteries with a high capacity model give longer lifetime than the ones with lower capacity. Even if batteries have the same types and models, they may give different lifetime values for different brands. Alkaline batteries have very low self-discharge, while NiMH batteries have significantly high self-discharge. This self-discharge causes the capacity loss and voltage reduction. Therefore, batteries with high starting give a longer lifetime period than low starting voltage. The capacity of batteries varies depending on their ageing and charging cycles (for rechargeable batteries) which results in different lifetime periods. Temperature can shorten or prolong lifetime depending on the effect on battery capacity and current draw. Therefore, finding the effect on both capacity and node current consumption is required for lifetime estimation due to temperature change.

For Dynamic Node Lifetime Estimation (DNLE), several steps and processes are proposed. The load testing method is used to find the real battery capacity which can cover many impact factors, such as battery type, brand, models and ageing. The equation for the battery capacity change due to self-discharge is also

presented for NiMH batteries. Then, the current consumption and the average current consumption have to be calculated based on a data sheet for every period. The temperature effect equations on both battery capacity and current consumption are applied. After that, the remaining capacity and the remaining lifetime equations are proposed.

The experiments have been conducted on real hardware and software platforms in WSNs with different battery models. Two scenarios, dynamic and static loads, are implemented to evaluate the deviation of DNLE compared to Selvig's method. Since Selvig's method does not include some impact factors, their estimated values are very different from the measured ones (around 18 % to 60 %), while they are close to the measured ones for DNLE (−3.5 % to 2.5 %). It can be concluded that node lifetime can be predicted more accurately using DNLE, which can be applied for both off-line and on-line lifetime estimation.

Chapter 8

EELAR: Energy-Efficient and Lifetime Aware Routing

8.1 Introduction

In a multihop wireless sensor routing protocol, such as CTP and RPL, the most commonly used metric for path selection is ETX. The path with lowest ETX is the best path which will be selected for forwarding data. However, the transmission (Tx) power and lifetime are also important key factors and hence have to be taken as metrics in path selection. In order to balance the network lifetime, a node should forward data to the highest lifetime path. Moreover, the node with the lowest lifetime should select a good quality link between the node and the parent in order to reduce energy wastage due to packet loss and avoid retransmission if it is enabled. Furthermore, a node should select a parent that it can send data to with the minimum Tx power in order to prolong its lifetime. This chapter describes the proposed energy-efficient and lifetime aware routing mechanism, which includes both Tx power and lifetime metrics for parent selection in addition to the ETX metric. The feasibility of the proposed scheme is validated by both real testbed and simulation experiments using performance metrics such as the average node lifetime, network lifetime, delay and packet loss rate with different scenarios.

8.2 EELAR Design and Implementation

In the proposed protocol, Energy-Efficient and Lifetime Aware Routing (EELAR), the data-gathering tree is constructed based on three metrics: ETX, lifetime and Tx power. This protocol is based on CTP and RPL. The energy efficiency of EELAR refers to efforts made to reduce the energy consumption of the radio component by sending data with the ideal power, reducing energy wastage due

to packet loss and avoiding retransmissions, while lifetime aware means a way to maximise the network lifetime by minimising the variance of the remaining lifetimes of the nodes in the network.

8.2.1 EELAR Design

The architecture of the proposed algorithm is shown in Figure 8.1. The link ETX value is the expected number of transmissions, including the retransmissions required to deliver a packet to its next hop. The path ETX value for the root is zero and that of other nodes is the sum of the link ETX values from that node to the root. The remaining lifetime estimation is calculated as steps described in Chapter 7 and Tx power is computed as Equation 6.3 in Chapter 5.

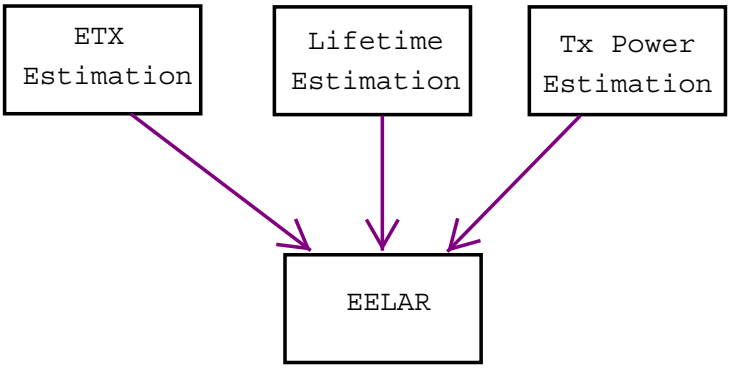


Figure 8.1: The Architecture of EELAR

In this algorithm, there are two threshold parameters: the ETX difference threshold ETX_{th} and the lifetime threshold Lt_{th}. The path lifetime value is the minimum lifetime value of the nodes along that path. ETX_{min} is the smallest path ETX value of all alternative paths. P is a set of candidate parents where all members in P have path ETX $\leq ETX_{min} + ETX_{th}$. $\#P$ is the number of elements of set P. Lt_{min} is the smallest path lifetime value of all members in P. The process of parent selection is:

- Case1: Based on link ETX and Tx power metrics

If ($\#P > 1$ and $P_i \in P$ and the node lifetime $< Lt_{min}$), the node will select P_i which has the minimum link ETX value as a parent. If there are multiple candidate parents with the same lowest link ETX, it will choose one of those parents that it can send data to with the minimum power. For example, with $ETX_{th} = 0.5$, in Figure 8.2, if $N2$ has the lowest lifetime value and two alternative paths: via $N0$ (path ETX=2.2, link ETX =1.2) and $N1$ (path ETX=2.3, link ETX=1), $N2$ will choose N1 as a parent, since link ETX of $N1$ is lower than link ETX of $N0$.

This is to avoid retransmission at the lowest lifetime node. In Figure 8.3, $N2$ will choose $N0$ as a parent since the link ETX of both candidates are the same and then the lower Tx power is considered.

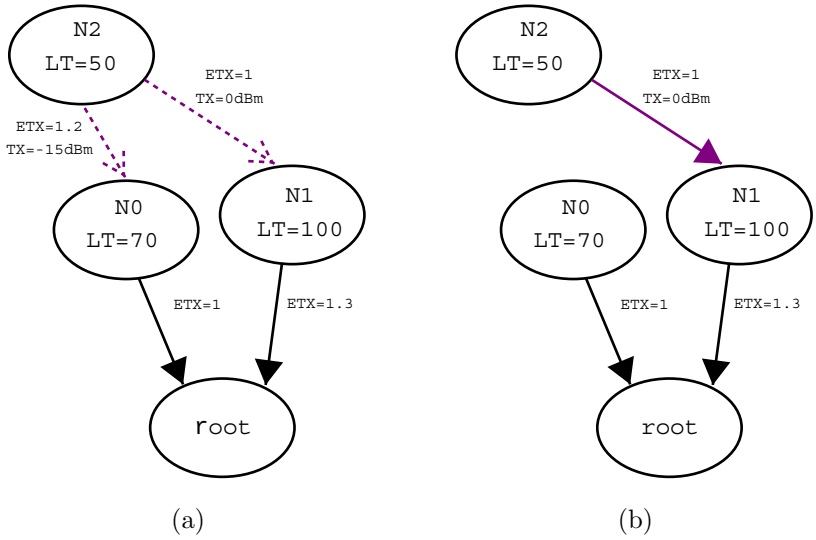


Figure 8.2: Case1:Path Selection Based on Link ETX

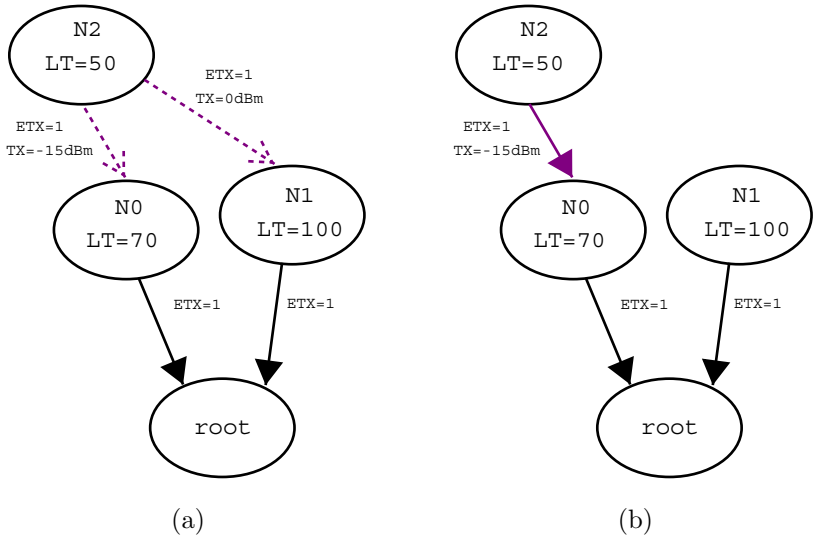


Figure 8.3: Case1:Path Selection Based on Tx Power

- Case2: Based on path lifetime, link ETX and Tx power metrics

If ($\#P > 1$ and $P_i \in P$ and the node lifetime $\geq Lt_{min}$), the node will select P_i which has the highest path lifetime value as a parent as shown in Figure 8.4. If there are multiple candidate parents with the same highest path lifetime, lower link ETX and Tx power are considered, respectively as shown in Figure 8.5 and Figure 8.6.

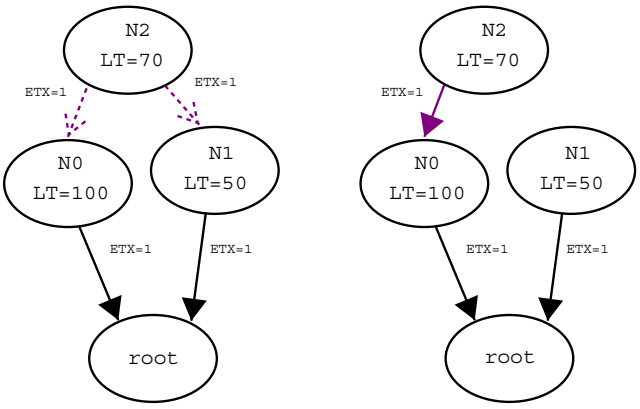


Figure 8.4: Case2:Path Selection Based on Path Lifetime

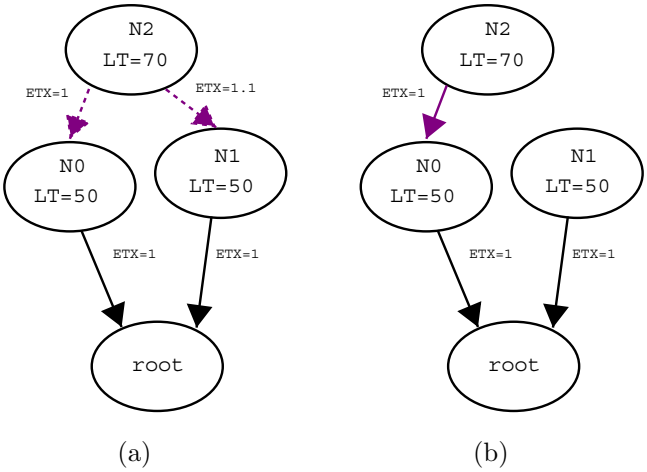


Figure 8.5: Case2:Path Selection Based on Link ETX

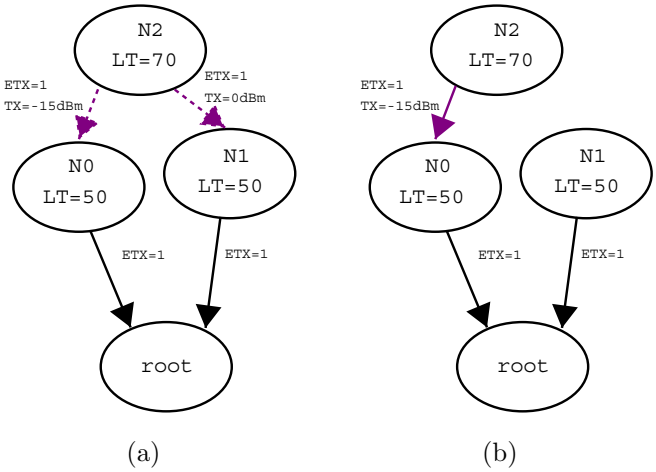


Figure 8.6: Case2:Path Selection Based on Tx Power

- Case3: Based on path ETX metric

If there is only one candidate parent in set P ($\#P = 1$), that candidate parent will be selected. This means a node chooses a parent which has the lowest path ETX. For example as shown in Figure 8.7, $N2$ will choose $N1$ as a parent since this path has the lower path ETX value and the path ETX of another alternative path $> ETX_{min} + ETX_{th}$.

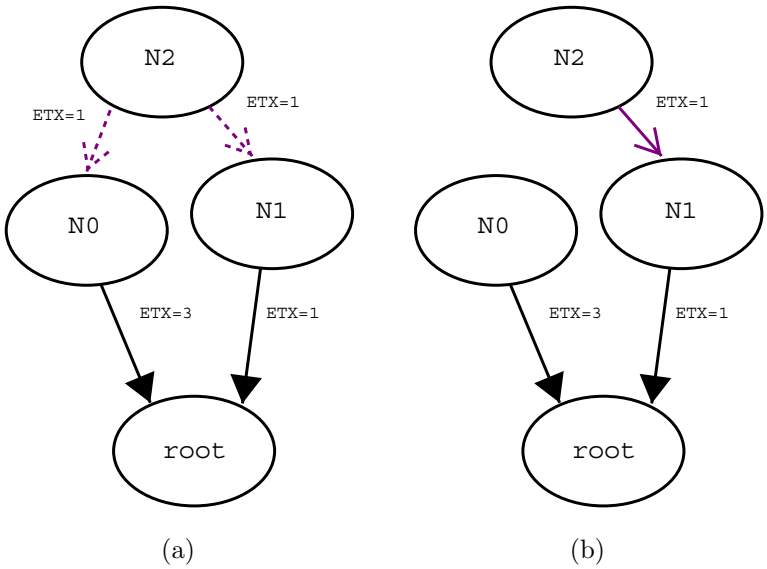


Figure 8.7: Case3:Path Selection Based on Path ETX

After completion of the path selection process, the node has to calculate the ideal power for sending packets to the selected parent, which can decrease the energy consumption of transmission and prolong the lifetime of that node.

8.2.2 EELAR Implementation

Two existing routing algorithms, CTP and RPL, are focused and modified. The modified CTP is implemented on TinyOS, while the modified RPL is implemented on Contiki. The low value of Lt_{th} can lead to an increase of the energy consumption because the routing control messages will be exchanged frequently. For the ETX threshold (ETX_{th}), if $ETX_{th} > 1$, it can lead to an increase of the packet delay due to the increase of the hop count. In this study, ETX_{th} is set to 0.5 and Lt_{th} is set to 30 min.

Modified CTP Path Selection

A new parameter entry called lifetime (Lt) is added in the routing frame, as shown in Figure 8.8. Two entries: lifetime and Tx power, are added in the routing table.

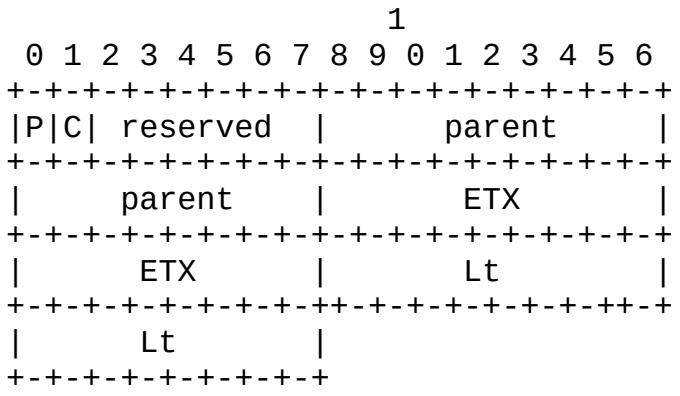


Figure 8.8: Modified CTP Routing Frame

The routing engine CtpRoutingEngine.nc is modified to implement the proposed EELAR algorithm. The lifetime value is sent with the beacon message. After receiving that beacon, the Tx power is calculated and updated by using the RSS of that beacon packet. Then, the nodes will update their neighbour table with the updated lifetime value. A new condition is added for resetting the beacon timer. This condition is when the lifetime of the node is changed more than the lifetime threshold (Lt_{th}= 30min). Therefore, the neighbours will have the most updated lifetime value for recalculating path lifetime. CTP represents ETX values as 16-bit fixed-point real numbers with a precision of hundredths, i.e. an ETX value of 50 represents an ETX of 0.5 in TinyOS 2.1.

Modified RPL Path Selection

The original RPL supports the ETX metric with 2 bytes length of object body. Adding a lifetime entry (2 bytes) can be easily done in the routing frame by changing the option length in the control message option (from 6 to 8 bytes) and length field in the routing metric/constraint object (from 2 to 4 bytes). Then, a lifetime entry is inserted in the object body following the ETX field. Two entries, lifetime and Tx power, are added in the parent structure. The path selection algorithm is modified (best_parent function) to implement the proposed EELAR algorithm. The lifetime value is sent with DIO message. After receiving DIO, the nodes compute the Tx power using the RSS of that DIO message. Then, the nodes will update their neighbour table with the updated lifetime value. When the lifetime of the node is changed more than the lifetime threshold (Lt_{th}), the DIO trickle timer is reset for updating new lifetime values to neighbours. RPL represents ETX values as 16-bit intergers by multiplying with 128, i.e. an ETX value of 64 represents an ETX of 0.5 in Contiki 2.6.

8.3 EELAR Evaluation

This section evaluates the EELAR algorithm on three hardware platforms: Mica2, N740 Nanosensor and MicaZ. In this study, N740 sensors are run on real testbed experiments, while MicaZ nodes are run on the simulator. Mica2 are conducted on both real testbed and simulator. The simulator used in this study is the AVRORA [112], a simulator based on Atmel AVR processor. Four transmission power levels: 0 dBm, -5 dBm, -10 dBm and -15 dBm, are used for this study. Current consumption for each transmission powers are 16.8, 13.8, 10.1 and 9.3 mA for Mica2, they are 17.4, 14, 11 and 9.9 mA for MicaZ, and they are 17.4, 12.4, 10.6 and 9.7 mA for N740. MicaZ and N740 platforms are implemented on Contiki OS with ContiliRPL on top of NullMAC, while Mica2 platform is implemented on TinyOS with CTP on top of B-MAC. Physical hardware/MAC auto acknowledgement is enabled for loss detection of next-hop transmission in order to calculate ETX at the MAC layer.

Performance Metrics

This study focuses on the major traffic pattern in WSN, where all sensor nodes send data messages to the base station. In order to save energy, these messages do not require an acknowledgement from the base station (or root node). Four performance metrics: the average node lifetime, network lifetime, packet delay and packet loss rate are evaluated through testbed and simulation experiments. The lifetime of a node ends when one of the following conditions are met: (1) its terminal voltages reaches the cut-off voltage at 2.2 V; (2) it does not has the ability to transmit data to the root (e.g. cannot find any relay nodes). Both the average lifetime of all nodes and the network lifetime are presented. The network lifetime is defined as it ends when the percentage of connected nodes is less than or equal to $n\%$.

Packet delay (P_{delay}) is measured as the time from a data packet being sent from the sender node until receiving that packet at the root node. An average delay of all messages (from all nodes) are used for the delay performance metric. For testbed experiments, they have been conducted on a few nodes. The maximum hop count is 2 and it can be assumed that there is no queueing delay. Therefore, P_{delay} for each node can be obtained by the pre-experiment testing as:

$$P_{delay} = \frac{t_{ack} - t_s}{2} \tag{8.1}$$

where t_s is time stamp of the packet leaving the sender and t_{ack} is time stamp of the acknowledgement arriving at the sender. From the pre-experiment testing, it

is observed that distance between two single-hop nodes and changing transmission power have no effect on packet transmission delay. For one-hop to the destination, the average P_{delay} values are 21 ms for Mica2 and 20 ms for N740. For a two-hop path, the average P_{delay} values are 47 ms for Mica2 and 43 ms for N740.

For simulation experiments, the simulator supports packet monitoring option. Therefore, the packet delay for each node can be calculated by:

$$P_{delay} = t_r - t_s \tag{8.2}$$

where t_s is time stamp of the packet leaving the sender and t_r is time stamp of the packet arriving at the root.

Sequence number attached with the data packet is used for measuring packet loss. Therefore, packet retransmission in the MAC layer is disabled in order to determine the actual packet loss rate. Moreover, carrier sense and random backoff operations in the MAC layer are also disabled to allow freely transmit concurrent packets and collisions

8.3.1 Testbed Experiments

The original CTP and RPL are modified for ELR and EELAR algorithms. ELR and EELAR are implemented inside the routing engine of CTP in TinyOS for Mica2 and best parent selection routine of RPL in Contiki OS for N740. Based on Equation 6.3, 10 dBm is used for X. The values of -85 dBm and -95 dBm are used as RSS_{min} for N740 and Mica2 in order to ensure that no transmission power reduction has been carried out for low quality signals. Three path selection algorithms: original with minimum path ETX, ELR and EELAR are investigated. Two scenarios are applied in the experiments.

Scenario1: Three Nodes

This scenario consists of three nodes, called the root node, N1 and N2 as shown in Figure 8.9. The root has an unlimited energy source (plug-in power), while the others are powered by fully charged NiMH 1300 mA h batteries and they are in the coverage range of the root. Two network lifetimes are focused on. First, it ends when the first monitoring node fails ($n = 50\%$). Second, it ends when the last node fails ($n = 0\%$). After joining the routing tree, $N1$ and $N2$ start sending data packets to the root. The interval of sending each packet is 3 s. This experiment repeats three times to find the average result of the performance metrics. Topology constructions for three algorithms are shown in Figure 8.10 and the performance comparisons of three algorithms are shown in Table 8.1 and 8.2.

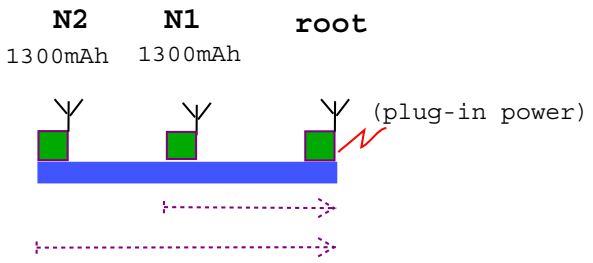
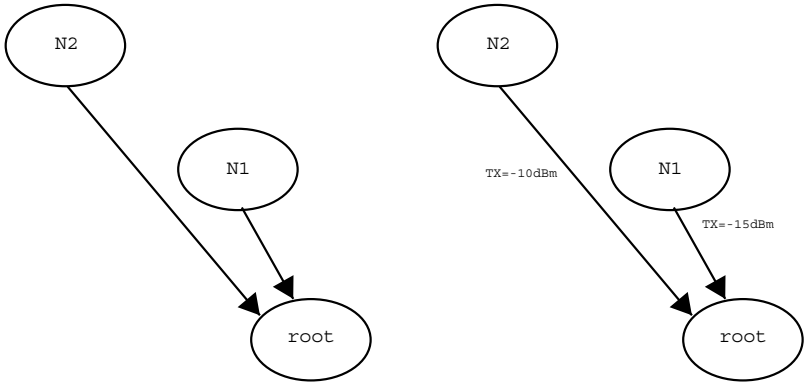


Figure 8.9: Testbed Scenario1 with Three Nodes



(a) (b)

Figure 8.10: Testbed Topology Construction of Three Nodes for (a) RPL/CTP and ELR (b) EELAR

Table 8.1: Testbed Performance Comparison of Three Path Selection Algorithms for Three Nodes of Mica2

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n = 50\%$ (min)	$n = 0\%$ (min)	(ms)	(%)
CTP	4685.0	4681	4689	21.0	2.47
ELR	4677.5	4675	4680	21.0	2.44
EELAR	4787.0	4771	4803	21.0	2.45

Table 8.2: Testbed Performance Comparison of Three Path Selection Algorithms for Three Nodes of N740

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n = 50\,\%$ (min)	$n = 0\,\%$ (min)	(ms)	(%)
RPL	963.5	960	967	20.0	0.25
ELR	958.0	956	960	20.0	0.23
EELAR	1001.5	996	1007	20.0	0.20

All algorithms build the same topology as both $N1$ and $N2$ sending data packets directly to the root. However, for CTP, RPL and ELR, both two nodes send data packets with the maximum transmission power (0 dBm), while they send packets with the ideal power for EELAR (-15 dBm for $N1$ and -10 dBm for $N2$). From the results, the average node lifetime and network lifetime of ELR is a little lower than CTP/RPL because ELR has an energy monitoring process which runs periodically. Moreover, the routing control message size of ELR is increased for carrying the path energy information. The change node energy in the path will trigger the routing message exchanges a little more than CTP/RPL. Therefore, ELR seems to consume energy more than CTP/RPL. EELAR gives the longer average node lifetime and the network lifetimes for both $n = 50\,\%$ and $n = 0\,\%$ even though it also has a monitoring process, an increased of the control message (2 bytes bigger than RPL) and a small increase in the number of control message ($\approx$ 25 messages/node for $Lt_{th} = 30$). This means that EELAR consumes the overall energy less than CTP/RPL, which is contributed to by reducing the transmission power. An increase of the lifetime ($\approx$ 102 min for Mica2 and 40 min for N740 in this case) will lead to a rise in the number of overall successfully received messages during the network lifetime. Delay and packet loss metrics have no significant difference for all algorithms.

Scenario2: Four Nodes

This scenario consists of four N740 nodes, called the root node, $N1$, $N2$ and $N3$ as shown in Figure 8.11. The root has an unlimited energy source (plug-in power), while $N1$, $N2$ and $N3$ are powered by fully charged NiMH 2500 mA h, 2000 mA h and 1300 mA h batteries, respectively. Only $N2$ is equipped with an antenna, while others are not. $N1$ and $N2$ are in the coverage range of the root but $N3$ is not. Therefore, $N3$ has to choose either $N1$ or $N2$ as a parent for forwarding data. After joining the routing tree, $N1$, $N2$ and $N3$ start sending UDP data packets to the root. The interval of sending each packet is 3 s. This experiment repeats three times to find the average result of performance metrics.

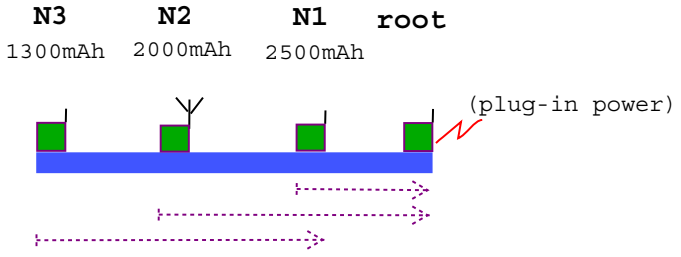


Figure 8.11: Testbed Scenario2 with Four Nodes

Topology constructions for three algorithms are shown in Figure 8.12 and the performance comparison of three algorithms is shown in Table 8.3.

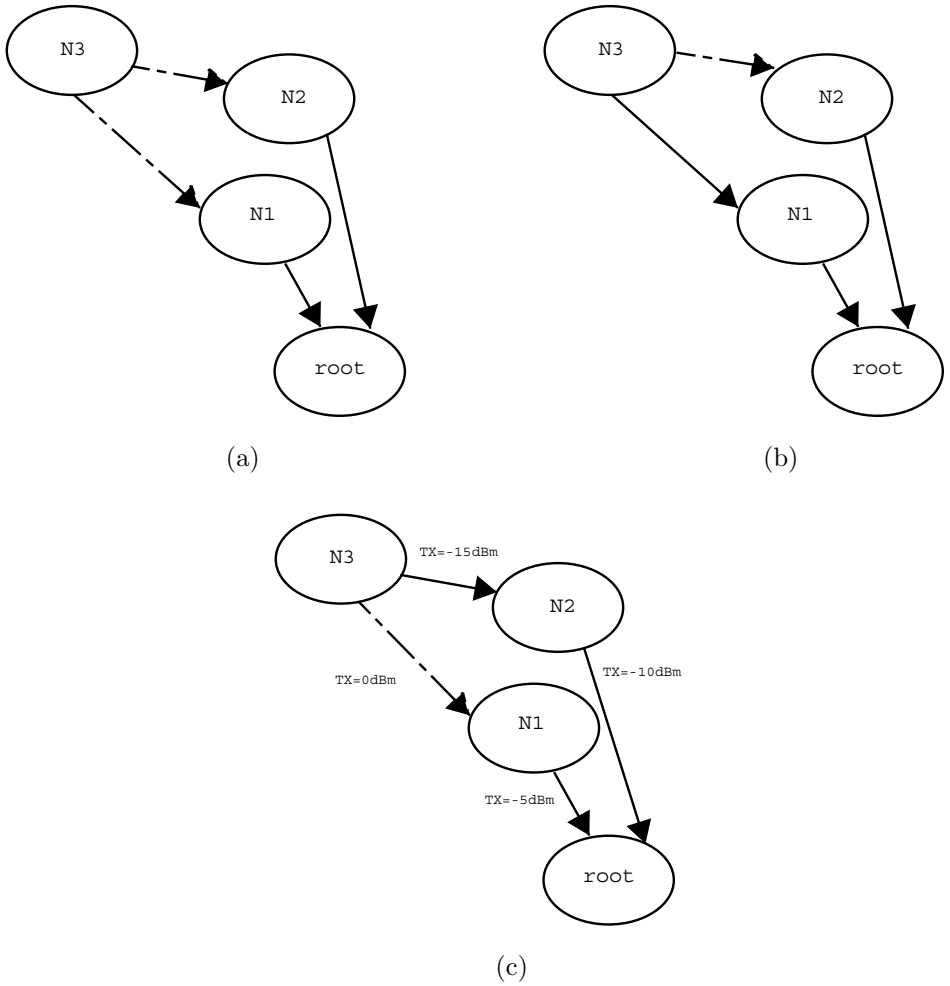


Figure 8.12: Testbed Topology Construction of Four Nodes for (a) RPL (b) ELR (c) EELAR

Table 8.3: Testbed Performance Comparison of Three Path Selection Algorithms for Four Nodes of N740

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n < 70\,\%$ (min)	$n < 35\,\%$ (min)	(ms)	(%)
RPL	1605.0	954	1716	27.7	1.23
ELR	1622.9	950	1739	27.7	1.22
EELAR	1669.3	987	1787	27.7	1.25

Similarly to Scenario1, both $N1$ and $N2$ behave the same by sending data packets directly to the root and they send data with maximum transmission power (0 dBm) for RPL and ELR, while they send data with the ideal power (-10 dBm

and -15 dBm) for EELAR. Parent selection of N3 is different for three algorithms. For RPL, $N3$ chooses N1 as a parent first. However, when packet loss occurs (either between $N3$ →N1 or $N1$ →root) and (path ETX of $N1$ - path ETX of $N2$) $>ETX_{th}$, the parent of $N3$ is changed to $N2$. The parent of N3 will switch back again if (path ETX of $N2$ - path ETX of $N1$) $>ETX_{th}$. For ELR, $N3$ chooses $N1$ as a parent since $N1$ has a higher path energy than $N2$. The parent of $N3$ will be $N2$ if (path ETX of $N1$ - path ETX of $N2$) $>ETX_{th}$, otherwise, the parent is still $N1$. For EELAR, $N3$ will choose $N2$ as a parent since $N3$ has the lowest lifetime and it can send data to $N2$ with the minimum transmission power. The parent of $N3$ will be $N1$ if (path ETX of $N2$ - path ETX of $N1$) $>ETX_{th}$ or link ETX of $N2$ > link ETX of $N1$, otherwise, the parent is still $N2$. The dominant parent of $N3$ is $N1$ for ELR, $N2$ for EELAR, while it is almost equally between $N1$ and $N2$ for RPL. $N3$ is the first node that is empty of energy, while $N2$ is the second. In this scenario, ELR performs a little better than RPL for the average node lifetime and the average node lifetime and network lifetime $n < 35\,\%$, which is contributed to by the selecting high path energy technique. EELAR gives the longest of average node lifetime and network lifetime for both $n < 70\,\%$ (the first node fails) and $n < 35\,\%$ (the second node fails). The lifetime increases ≈ 65 min for N740 in this case, which is contributed to by both reducing transmission power and choosing the longest lifetime path.

8.3.2 Simulation Experiments

The AVRORA simulator supports a free-space path loss propagation model for RSS estimation. The radio coverage is set to the maximum of 10 m distance. This simulator also supports energy and packet monitoring. The simulator calculates the energy consumption of five components which are CPU, LEDs, radio transceiver, sensor board and flash memory. It is modified to support setting the different levels of initial energy for each node. With similar scenarios to the testbed experiments, two algorithms: ELR and EELAR, are implemented in both TinyOS and Contiki with two hardware platforms: Mica2 and MicaZ. Moreover, another senario is added for the large-scale testing with many nodes. The MicaZ platform is implemented in the Contiki environment, while the Mica2 platform is only implemented in TinyOS.

Scenario1: Three Nodes

This scenario consists of three nodes, called the root node, $N1$ and $N2$. $N1$ and $N2$ are placed at a distance of 2 m and 5 m from the root. The root has an unlimited energy source (setting with very high value), while $N1$ and $N2$ are set with similar

energy to 1300 mA h (14040 J). Both $N1$ and $N2$ are in the coverage range of the root. The topology constructions for the three algorithms are the same as the testbed experiment. The performance comparisons of the three algorithms are shown in Table 8.4 and 8.5.

Table 8.4: Simulation Performance Comparison of Three Path Selection Algorithms for Three Nodes of Mica2

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n = 50\%$ (min)	$n = 0\%$ (min)	(ms)	(%)
CTP	5186.0	5186	5186	23.0	0.15
ELR	5183.0	5183	5183	23.0	0.15
EELAR	5307.5	5264	5351	23.0	0.15

Table 8.5: Simulation Performance Comparison of Three Path Selection Algorithms for Three Nodes of MicaZ

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n = 50\,\%$ (min)	$n = 0\,\%$ (min)	(ms)	(%)
RPL	2882.0	2882	2882	16.0	0.21
ELR	2880.0	2880	2880	16.0	0.21
EELAR	2968.0	2961	2975	16.0	0.21

With the simulation, it is assumed that the usable capacities are 100% of the battery capacities. Therefore, the lifetimes of all the algorithms are longer than the results of the testbed experiment. However, these results also exhibit the same trends as the testbed experiment. EELAR performs better on average node lifetime and network lifetime performance both $n = 50\%$ and $n = 0\%$ compared to the other two algorithms.

Scenario2: Four Nodes

This scenario consists of four MicaZ nodes, called the root node, $N1$, $N2$ and $N3$. $N1$, $N2$ and $N3$ are placed at the distance of 3 m, 6 m and 11 m from the root. The root has an unlimited energy source (set at a very high value), while $N1$, $N2$ and $N3$ are set with similar energy as 2500 mA h (27000 J), 2000 mA h (21600 J) and 1300 mA h (14040 J). Both $N1$ and $N2$ are in the coverage range of the root, while $N3$ is not. The topology constructions for the three algorithms are the same

as the testbed experiment. The performance comparisons of the three algorithms are shown in Table 8.6.

Table 8.6: Simulation Performance Comparison of Three Path Selection Algorithms for Four Nodes of MicaZ

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n < 70\,\%$ (min)	$n < 35\,\%$ (min)	(ms)	(%)
RPL	4287.8	2880	4437	22.3	0.93
ELR	4290.3	2877	4440	22.3	0.93
EELAR	4337.6	2917	4487	22.3	0.93

The experimental results show the same trend as the testbed experiment. EELAR gives the best performance on average network and network lifetime both $n<$ 70% and $n<$ 35%.

Scenario3: Thirty-two Nodes

This experiment is conducted with 32 nodes of MicaZ located randomly in a 40x40m area. One of these nodes is the root node with unlimited energy, 10 nodes are set with the energy of 1300 mA h (14040 J), 10 nodes are set with the energy of 2000 mA h (21600 J), and the others are set with the similar energy of 2500 mA h (27000 J) as shown in Figure. 8.13.

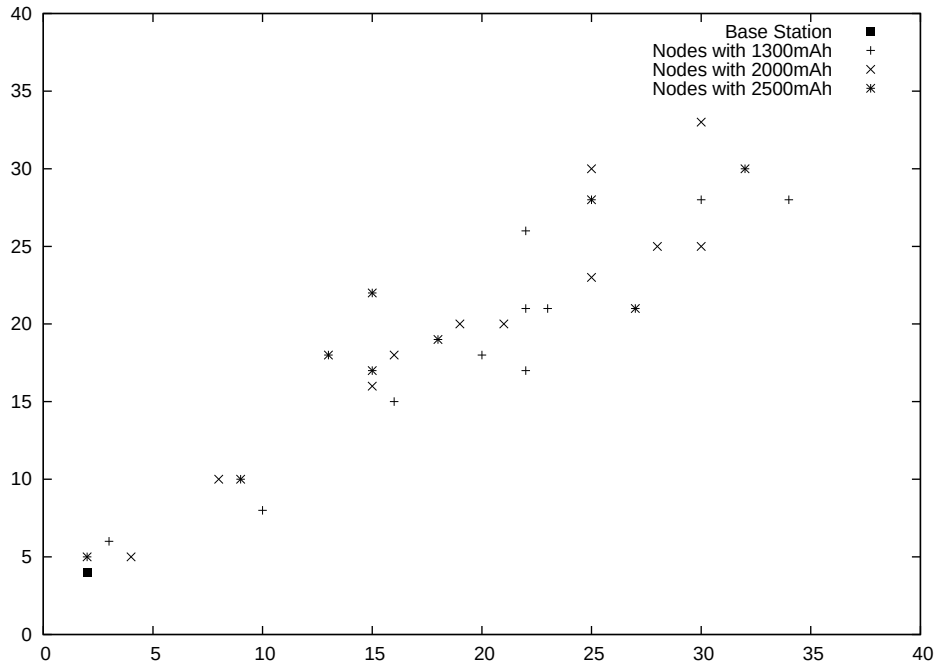
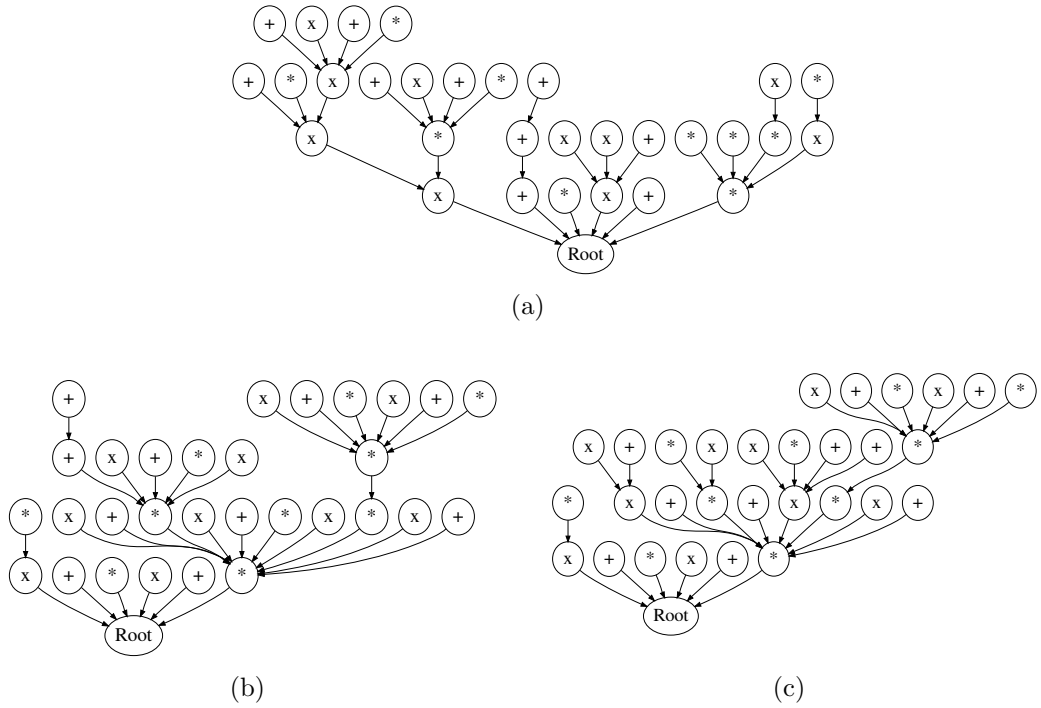


Figure 8.13: Simulation Scenario3 with Thirty-two Nodes

The topology constructions (with dominant parent of each node) for the three algorithms are shown in Figure 8.14 and the performance comparisons of the three algorithms are shown in Table 8.7.



(a) (b) (c)

Figure 8.14: Simulation Topology Construction of Thirty-two Nodes for (a) RPL (b) ELR (c) EELAR

Table 8.7: Simulation Performance Comparison of Three Path Selection Algorithms for Thirty-two Nodes of MicaZ

Algorithm	Performance Metrics				
	Lifetime			Delay	Loss
	average (min)	$n < 70\,\%$ (min)	$n < 36\,\%$ (min)	(ms)	(%)
RPL	4286.2	2882	4434	39.8	1.53
ELR	4291.7	2885	4440	40.9	1.52
EELAR	4312.8	2910	4457	41.5	1.20

EELAR gives the best performance on packet-loss rate, average node lifetime and network lifetime both $n < 70\,\%$ (10 nodes fail) and $n < 36\,\%$ (20 nodes fail), which is contributed to by both reducing transmission power and choosing the longest lifetime path. Reducing transmission power can prolong the lifetime of a node. Moreover, it can reduce the collision domain which will result in a decrease in the probability of packet loss. ELR performs better than RPL on average node lifetime and network lifetime, which is contributed to by selecting a high path energy technique. The time delay of messages may be a slight difference (a little

longer/shorter) for these three algorithms depending on the selected path of each node.

8.4 Summary and Discussion

EELAR, Energy-Efficient and Lifetime Aware Routing, is proposed for path selection based on three metrics: ETX, lifetime and Tx power. This algorithm reduces energy consumption by sending data with the ideal power, reducing packet loss and avoiding retransmissions. It also maximises the network lifetime by minimising the variance of the remaining lifetimes of the nodes in the network.

Both testbed and simulation experiments have been conducted on real hardware and software platforms in WSNs with different battery capacities. ELR gives a better network lifetime than the original CTP/RPL, which is caused by choosing a high path energy method. By decreasing transmission power, EELAR gives the best performance on packet loss rate and network lifetime, when compared to ELR and the original CTP/RPL. It is observed that there is no significant difference in the packet delay with each of these algorithms.

Chapter 9

Conclusions and Future Work

9.1 Conclusions

In multihop wireless sensor routing protocols, the most commonly used metric for path selection is the Expected Transmission Count (ETX) value. The path with the lowest ETX is the best path, which will be selected for forwarding data. However, other metrics, such as transmission (Tx) power and node lifetime, should be taken into account as metrics in path selection. For example, a node with the lowest lifetime should try different ways to reduce energy consumption, such as reducing transmission power by choosing the path that can send data with the lowest Tx power. Moreover, it should select a good quality link between the node and the parent in order to reduce energy wastage due to packet loss and retransmission. Furthermore, a node should forward data to the path that has the longest lifetime in order to balance the lifetime of the network.

This dissertation has three main contributions. Firstly, an Energy-Efficient Transmission Power Control (EETPC) mechanism is proposed that uses measured Received Signal Strength (RSS) to calculate ideal transmission power. This includes the investigation of the factors impacting on RSS, such as distance between nodes, height above ground, multipath environment, the capability of node, noise and interference and temperatures. The proposed TPC mechanism considers these factors and can be applied to single-hop with multihoming and multihop networks. The experiments have been conducted on real hardware and software platforms in WSNs. The EETPC technique provides an ideal transmission power which leads to the decrease of transmission energy consumption. Moreover, it reduces the size of collision domain and allows more simultaneous transmission of nodes in the network which results in minimising the chance of collision. Dynamic power adjustment methods are also suggested for improving the performance of packet loss and energy consumption for transmission.

Secondly, a Dynamic Node Lifetime Estimation (DNLE) technique for WSNs is proposed that considers many impact factors on node lifetime, such as battery type, model, brand, self-discharge, discharge rate, age, charge cycles and temperature. Experiments were conducted on real hardware and software platforms in WSNs with different battery models. Node lifetime can be predicted more accurately using DNLE, which can be applied for both dynamic and static load applications.

Thirdly, an Energy-Efficient and Lifetime Aware Routing (EELAR) algorithm is proposed, which is designed and developed for prolonging network lifetime in multihop WSNs. The proposed routing algorithm considers both transmission power and node lifetime, in addition to the ETX metric. EELAR gives the best performance on packet loss rate, average node lifetime and network lifetime compared to the other algorithms, such as CTP, RPL and ELR.

9.2 Future Work

The proposed methods have some current limitations and possibilities for future extensions as follows:

9.2.1 Energy-Efficient Transmission Power Control (EETCP)

In EETPC, it is assumed that all sensor nodes support an acceptable accuracy of RSSI value (e.g. ± 5 dBm), since the proposed method relies on RSSI to find the RSS for ideal transmission power calculation. Moreover, it is designed with the assumption that all nodes are stationary and their radio transceivers are always on. Therefore, an important development and improvement that could be considered in future research is to cope with the low radio accuracy of the RSSI. In addition, the proposed method could be extended to cover different environments, such as mobility and low duty cycle applications.

9.2.2 Dynamic Node Lifetime Estimation (DNLE)

In DNLE, some impact factors on node lifetime were not taken into account, such as battery recovery effects and humidity. For intermittent discharges, the battery can recover during idle periods, which leads to an extension of the battery lifetime. Moreover, the battery will fill up the internal space allotted for cell discharge if the humidity is high. This results in a short service lifetime. Therefore, future

work could include the battery recovery effects when nodes employ dynamic duty cycling. Furthermore, the impact of humidity could be investigated.

9.2.3 Energy-Efficient and Lifetime Aware Routing (EELAR)

In EELAR, the experiments have been tested with 100% duty cycle (no sleep and the radio is always on). Since changing the duty cycle may have some impact on the routing algorithm, future work could include the effects when nodes employ dynamic duty cycling [72, 82].

References

[1] CC1000 Single Chip Very Low Power RF Transceiver. CC1000 Data Sheet, Available at http://www.ti.com/lit/ds/symlink/cc1000.pdf.

[2] Duracell Alkaline-Manganese Dioxide. Available at http://media.ww2.duracell.com/media/en-US/pdf/gtcl/Technical_Bulletins/Alkaline%20Technical%20Bulletin.pdf.

[3] GP Batteries Nickel Metal Hydride Technical Handbook. Available at http://www.gpina.com/pdf/NiMH_technical.pdf.

[4] Mica2 Wireless Measurement System. Available at http://bullseye.xbow.com:81/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf.

[5] MicaZ Wireless Measurement System. Available at http://bullseye.xbow.com:81/Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf.

[6] IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). Technical report, 2006.

[7] 2.4 GHz IEEE 802.15.4 / ZigBee-ready rf tranceiver. CC2420 Data Sheet (Rev. B), March 2007.

[8] System on Chip for 2.4 GHz ZigBee® / IEEE 802.15.4 with Location Engine. CC2431 Data Sheet (Rev. 2.0.1), May 2007.

[9] A True System on Chip solution for 2.4 GHz IEEE 802.15.4 / ZigBee®. CC2430 Data Sheet (Rev. 2.1), May 2007.

[10] Ansmann specifications. Available at http://docs-europe.electrocomponents.com/webdocs/0d49/0900766b80d49194.pdf, 2009.

[11] Nickel Metal Hydride (NiMH)Handbook and Application Manual, Energizer Nickel Metal Hydride Handbook Version: NiMH01.11. Available at http://data.energizer.com/PDFs/nickelmetalhydride_appman.pdf, 2010.

[12] ATmega128/ATmega128L: 8-bit Atmel microcontroller with 128K-Bytes in-system programmable flash. Available at http://www.atmel.com/Images/2467S.pdf, 2011.

[13] Alkaline Manganese Dioxide Handbook and Application Manual, Energizer Alkaline Handbook Version: Alk2.0. Available at http://data.energizer.com/PDFs/alkaline_appman.pdf, 2012.

[14] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102 – 114, aug 2002.

[15] Ansmann. Ansmann energy check LCD battery tester. Available at http://www.farnell.com/datasheets/39714.pdf.

[16] Kenneth Bannister, Gianni Giorgetti, and Eep K. S. Gupta. Wireless sensor networking for hot applications: Effects of temperature on signal strength, data collection and localization. In *Proceedings of the ACM 15th Workshop on Embedded Networked Sensors*, USA, June 2008.

[17] S. Baydere, Y. Safkan, and O. Durmaz Incel. Lifetime analysis of reliable wireless sensor networks. *IEICE transactions on communications*, E88-B:2465–2472, 2005. Bay05 Imported from DIES.

[18] H.J. Bergveld, W.S. Kruijt, and P.H.L Notten. *Battery Management Systems: Design by Modelling (Philips Research Book Series)*. Springer, 2002.

[19] UC Berkeley. What is tinyos? Available at http://www.tinyos.net, 2004.

[20] M. Bertocco, G. Gamba, A.Sona, and S. Vitturi. Performance measurements of csma/ca-based wireless sensor networks for industrial applications. In *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE*, pages 1 –6, may 2007.

[21] M. Bhardwaj, T. Garnett, and A.P. Chandrakasan. Upper bounds on the lifetime of sensor networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 3, pages 785–790, 2001.

[22] D.M. Blough, M. Leoncini, G. Resta, and P. Santi. The k-neighbors approach to interference bounded and symmetric topology control in ad hoc networks. *Mobile Computing, IEEE Transactions on*, 5(9):1267 –1282, sept. 2006.

[23] Jan Blumenthal and Dirk Timmermann. Minimal transmission power as distance estimation for precise localization in sensor networks. In *Proc. Proc. IWCMC 06*, pages 1331–1336, Vancouver, British Columbia, Canada, July 2006.

[24] Carlo Alberto Boano, James Brown, Zhitao He, Utz Roedig, and Thiemo Voigt. *Low-Power Radio Communication in Industrial Outdoor Deployments: The Impact of Weather Conditions and ATEX-compliance*, pages 159–176. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer, 2009.

[25] K. Lahiri C. Park and A. Ranghunathan. Battery discharge characteristics of wireless sensor nodes: An experimental analysis. In *The 2nd Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON05)*, CA, USA, 2005.

[26] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for Ad Hoc network research. *Wireless Communications & Mobil Computing (WCMC): Special Issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2:483–502, 2002.

[27] Bogdan Carbunar, Ananth Grama, and Jan Vitek. Redundancy and coverage detection in sensor networks. *ACM Transactions on Sensor network (ToSN*, 2:94–128, 2006.

[28] M. Cardei and J. Wu. *Coverage in Wireless Sensor Networks*. CRC Press, 2004.

[29] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *Ninth Annual International Conference on Mobile Computing and Networking*, pages 114–146, 2003.

[30] Isabel Dietrich and Falko Dressler. On the lifetime of wireless sensor networks. *ACM Trans. Sen. Netw.*, 5(1):5:1–5:39, February 2009.

[31] Adam Dunkels. Programming memory-constrained networked embedded systems. Available at http://www.sics.se/~adam/dunkels07programming.pdf, 2007.

[32] Adam Dunkels, Bjorn Gronvall, and Thiemo Voigt. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*, LCN '04, pages 455–462, Washington, DC, USA, 2004. IEEE Computer Society.

[33] Adam Dunkels, Fredrik Osterlind, Nicolas Tsiftes, and Zhitao He. Software-based on-line energy estimation for sensor nodes. In *Proceedings of the 4th Workshop on Embedded Networked Sensors*, pages 28–32, 2007.

[34] Richard Zurawski (ed.). *Embedded Systems Handbook*. CRC Press, London, 2005.

[35] Muhammad Omer Farooq and Thomas Kunz. Operating systems for wireless sensor networks: A survey. *Sensors*, 11(6):5900 – 5930, 2011.

[36] Rodrigo Fonseca, Omprakash Gnawali, Kyle Jamieson, and Philip Levis. Four bit wireless link estimation. In *Proceedings of the HotNets*, 2007.

[37] Vijay K. Garg. *Wireless communications and networking*. Morgan Kaufmann publications, CA,USA, 2007.

[38] David Gay, Philip Levis, Robert von Behren, Matt Welsh, Eric Brewer, and David Culler. The nesc language: A holistic approach to networked embedded systems. *SIGPLAN Not.*, 38(5):1–11, May 2003.

[39] Michael Gerharz, Christian De Waal, and Peter Martini. A cooperative nearest neighbours topology control algorithm for ad hoc networks. In *in Proc. 12th International Conference on Computer Communications and Networks (ICCCN03*, pages 412–417, 2003.

[40] Omprakash Gnawali, Rodrigo Fonseca, Kyle Jamieson, David Moss, and Philip Levis. Collection Tree Protocol. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys'09)*, November 2009.

[41] Javier Gomez. A Case for Variable-Range Transmission Power Control in Wireless Multihop Networks. In *in INFOCOM 04*, pages 1425–1436, 2004.

[42] Javier Gomez and Andrew T. Campbell. Variable-Range Transmission Power Control in Wireless Ad Hoc Networks. *IEEE Transactions on Mobile Computing*, 6(1):87–99, 2007.

[43] V. C. Gungor and M. K. Korkmaz. Wireless link-quality estimation in smart grid environments. *International Journal of Distributed Sensor Networks*, 2012.

[44] V. Cagri Gungor, Chellury Sastry, Zhen Song, and Ryan Integlia. Resource-aware and link quality based routing metric for wireless sensor and actor networks. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 3364–3369, 2007.

[45] Wu Guoliang, Lu Rengui, Zhu Chunbo, and C.C. Chan. State of charge estimation for NiMH battery based on electromotive force method. In *Vehicle Power and Propulsion Conference, VPPC '08*, pages 1 –5, sept. 2008.

[46] Shuai Hao, Mun Choon Chan, Bhojan Anand, and A. L. Ananda. Structure health monitoring at MRT construction sites using wireless sensor networks link measurement at Pasir Panjang station.

[47] M. Harvan and J. Schnwlder. A 6lowpan implementation for tinyos. 2.0. In *6th GI/ITG KuVS Fachgespräch Wireless Sensor Networks*, Aachen, 2007.

[48] Ward Van Heddeghem. Cross-layer link estimation for contiki-based wireless sensor networks. Master Thesis, Vrije Univesiteit Brussel, May 2009.

[49] V. P. Henk, H. J. Bergveld, D. Danilov, P. P.L. Regtien, and P. H.L. Notten. *Battery Management Systems: Accurate State-of-Charge Indication for Battery-Powered Applications*. Springer, 2008.

[50] L. Hester and K. Cornett. *The IEEE 802.15.4 (low rate wireless personal area network) standard*. Motorola Labs, 2002.

[51] B.F.W.H. Heyer. One meter battery tester. US Patent 2,225,051, May 1938.

[52] J. L. Hill and D. E. Culler. Mica: a wireless platform for deeply embedded networks. 22(6):12–24, 2002.

[53] Jonathan Hui (editor) and Pascal Thubert. Compression format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC 6282, September 2011.

[54] ZTS Inc. The ZTS MBT-MIL multi-battery tester (MBT-MIL), 2008.

[55] ISI. The network simulator - ns-2. Available at http://www.isi.edu/nsnam/ns.

[56] X. Jiang, P. Dutta, D. Culler, , and I. Stoica. Micro power meter for energy monitoring of wireless sensor networks at scale. In *Proceedings of the 6th international conference on Information processing in sensor networks, IPSN 07*, page 186195, Massachusetts, USA, 2007.

[57] M. Johnson, M. Healy, P. Van de Ven, M.J. Hayes, J. Nelson, T. Newe, and E. Lewis. A comparative review of wireless sensor network mote technologies. In *Sensors, 2009 IEEE*, pages 1439 –1442, oct. 2009.

[58] M. R. Jongerden and B. R. Haverkort. Battery modeling. In *Technical Report TR-CTIT-08-01*, Centre for Telematics and Information Technology University of Twente, 2008.

[59] S. Kabir, R. Khan, S.H. Hossen, and A. Azfar. Interconnection between 802.15.4 devices and IPv6: Implications and existing approaches. *International Journal of Computer Science Issues*, 7(1):19–31, 2010.

[60] J. J.C. Kopera. Considerations for the utilization of NiMH battery technology in stationary applications. In *BATTCON 2005*, Florida, 2005.

[61] N. Kushalnagar, G. Montenegro, and C. Schumacher. IPv6 over low-power wireless personal area networks (6LoWPANs): Overview, assumptions, problem statement, and goals. RFC 4919, August 2007.

[62] Olaf Landsiedel and Klaus Wehrle. Aeon: Accurate prediction of power consumption in sensor networks. In *Proceedings of The Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*, 2004.

[63] J. Lee. Performance evaluation of IEEE 802.15.4 for low-rate wireless personal area networks. *IEEE Transactions on Consumer Electronics*, 52(3):742–749, 2006.

[64] Jungwook Lee and Kwangsue Chung. An efficient transmission power control scheme for temperature variation in wireless sensor networks. *Sensors*, (3):3078–3093, 2011.

[65] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim: Accurate and scalable simulation of entire tinyos applications, 2003.

[66] W. Liao, L. He, and K. M. Lepak. Temperature and supply voltage aware performance and power modeling at microarchitecture level. *IEEE transaction on computer-aided design of integrated circuits and systems*, 24(7):1042–1053, jul. 2005.

[67] Weiping Liao, Fei Li, and Lei He. Microarchitecture level power and thermal simulation considering temperature dependent leakage model. In *Proceedings of the 2003 international symposium on Low power electronics and design*, ISLPED '03, pages 211–216, New York, NY, USA, 2003. ACM.

[68] S. Lin, J. Zhang, G. Zhou, L. Gu, T. He, and John A. Stankovic. Atpc: Adaptive transmission power control for wireless sensor networks. In *In Proceedings of the Fourth International Conference on Embedded Networked Sensor Systems (SenSys)*, 2006.

[69] Sensinode Ltd. Sensinode Ltd. joins Texas Instruments low-power RF developer network, offers IPv6 wireless network solutions based on 6lowpan

standard technology. Available at http://embedded-computing.com/sensinode-based-6lowpan-standard-technology, 2008.

[70] Daniel F. Macedo, Aldri L. dos Santos, Luiz H. A. Correia, A José Marcos S. Nogueira, and Guy Pujolle. Energy, Transmission Power and Data Rate Aware Routing on Mobile Ad Hoc Networks. Technical report, 2009.

[71] Ritesh Madan, Shuguang Cui, Sanjay Lall, and Andrea Goldsmith. Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. In *In Proc. IEEE INFOCOM*, pages 1964–1975, 2005.

[72] Vasilis Michopoulos, Lin Guan, George C. Oikonomou, and Iain Phillips. DCCC6: Duty Cycle-aware congestion control for 6LoWPAN networks. In *PerCom Workshops*, pages 278–283, 2012.

[73] M.-P. Mihai, N. Adrian, S. Iuliu, M. Gheorghe, M. Mihai, and K. Csaba-Zoltan. Energy-efficient tracking for wireless sensor networks. In *Robotic and Sensors Environments, 2009. ROSE 2009. IEEE International Workshop on*, pages 163 –168, nov. 2009.

[74] Jeffrey P. Monks, Vaduvur Bharghavan, and Wen mei W. Hwu. A Power Controlled Multiple Access Protocol for Wireless Packet Networks, year=2001, month=, volume=1, pages=219-228,. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*.

[75] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler. Transmission of IPv6 packets over IEEE 802.15.4 networks. RFC 4944, September 2007.

[76] Gowri S. Nagarajan and J.W. Van Zee. Characterization of the performance of commercial NiMH batteries. *Journal of Power Sources*, 70(2):173 – 180, 1998.

[77] H. A. Nguyen, A. Forster, D. Puccinelli, and S. Giordano. Sensor node lifetime: An experimental study. In *IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops*, pages 202 – 207, 2011.

[78] B.L. Titzer O. Landsiedel, K. Wehrle and J. Palsberg. Enabling detailed modeling and analysis of sensor networks. *Praxis der Informationsverarbeitung und Kommunikation*, 28(2):101–106, 2005.

[79] Sophocles J. Orfanidis. *Electromagnetic Waves and Antennas*. Rutgers University, Brett Road Piscataway, NJ, USA, 2008.

[80] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. Cross-level sensor network simulation with cooja. In *Local Computer Networks, Proceedings 2006 31st IEEE Conference on*, pages 641 –648, nov. 2006.

[81] Rajesh Palit, Ajit Singh, and Kshirasagar Naik. Modeling the energy cost of applications on portable wireless devices. In *Proceedings of the 11th international symposium on Modeling, analysis and simulation of wireless and mobile systems*, MSWiM '08, pages 346–353, New York, NY, USA, 2008. ACM.

[82] Pangun Park, Sinem Coleri Ergen, Carlo Fischione, and Alberto Sangiovanni-Vincentelli. Duty-cycle Optimization for IEEE 802.15.4 Wireless Sensor Networks. *ACM Trans. Sen. Netw.*, 10(1):12:1–12:32, December 2013.

[83] Dhaval Patel, Bijal Chawla, and Chandresh Parekh. Energy aware and link quality based routing in wireless sensor networks under tinyos-2.x. *International Journal of Modern Engineering Research (IJMER)*, 3(3):1357–1365, may-june 2013.

[84] K. Pavai, D. Sridharan, S.A.V. Satya Murty, and A. Sivagami. Energy and link quality based routing for data gathering tree in wireless sensor networks under tinyos -2.x. *International Journal of Wireless and Mobile Networks*, 2(2):47–60, 2010.

[85] Boguslaw Pierozynski. On the Low Temperature Performance of Nickel-Metal Hydride (NiMH) Batteries. *International Journal of Electrochemical Science*, 6:860–866, 2011.

[86] Su Ping. Delay measurement time synchronization for wireless sensor networks. Intel Research Berkeley Lab, June 2003.

[87] West Mountain Radio. Computerized battery analyzer (CBA).

[88] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 404–413, 2000.

[89] S. Ranvier. Path loss models. Available at http://www.comlab.hut.fi/opetus/333/2004_2005_slides/Path_loss_models.pdf, 2004.

[90] T. Rappaport. *Wireless Communications: Principles and Practice, 2nd Ed.* Prentice Hall, Upper Saddle River, NJ, USA, 2002.

[91] Volkan Rodoplu, Student Member, and Teresa H. Meng. Minimum energy mobile wireless networks. *IEEE Journal on Selected Areas in Communications*, 17:1333–1344, 1999.

[92] Joel J. P. C. Rodrigues and Paulo A. C. S. Neves. A survey on IP-based wireless sensor network solutions. *Int. J. Commun. Syst.*, 23(8):963–981, August 2010.

[93] W. Rukpakavong, I. Phillips, and L. Guan. Neighbour discovery for transmit power adjustment in IEEE 802.15.4 using RSSI. In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, pages 1–4, feb. 2011.

[94] W. Rukpakavong, I. Phillips, and L. Guan. Lifetime estimation of sensor device with AA NiMH batteries. In *IPCSIT vol. 55, International Conference on Information Communication and Management (ICICM)*, pages 98–102, oct. 2012.

[95] W. Rukpakavong, I. Phillips, L. Guan, and G. Oikonomou. RPL router discovery for supporting energy-efficient transmission in single-hop 6LoWPAN. In *Communications (ICC), 2012 IEEE International Conference on*, pages 5721 –5725, jun. 2012.

[96] Matthew N.O. Sadiku. *Elements of Electromagnetics(5th Edition)*. Oxford University Press, 2009.

[97] B. Selvig. Measuring power consumption with CC2430 & Z-Stack. Application Note AN053(Rev. 1.0).

[98] Z. Shelby and C. Bormann. *6LoWPAN: The wireless embedded internet*. J. Wiley, Chichester, U.K, 2009.

[99] Zach Shelby (editor), Samita Chakrabarti, and Erik Nordmark. Neighbor discovery optimization for low-power and lossy networks. Internet Draft, December 2010.

[100] Xingfa Shen, Zhi Wang, Peng Jiang, Ruizhong Lin, and Youxian Sun. Connectivity and rssi based localization scheme for wireless sensor networks. In *Proceedings of the 2005 international conference on Advances in Intelligent Computing - Volume Part II*, ICIC'05, pages 578–587, Berlin, Heidelberg, 2005. Springer-Verlag.

[101] C. Sheridan, J. Petersen, and J. Rohwer. On modifying the Arrhenius equation to compensate for temperature changes for reactions within biological systems. In *Water SA*, volume 38, jan. 2012.

[102] SICS. Sicslowpan - internet for low-power, low-cost wireless. Available at https://www.sics.se/projects/sicslowpan, 2008.

[103] R.M. Silva, J. Silva, and F. Boavida. Evaluating 6lowPAN implementations in WSNs. In *Proc. 9 Conferancia sobre Redes de Computadores*, Portugal, September 2009.

[104] Dongjin Son, Bhaskar Krishnamachari, and John Heidemann. Experimental study of the effects of transmission power control and blacklisting in wireless sensor networks. In *IEEE SECON*, pages 289–298, 2004.

[105] William Stallings. *Wireless Communications & Networks (2nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2004.

[106] William Stallings. *Data and Computer Communications:Eight Edition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2007.

[107] Eun sun Jung and Nitin H. Vaidya. A Power Control MAC Protocol for Ad Hoc Networks. *ACM/Kluwer Journal on Wireless Networks*, 11(1-2):55–66, 2005.

[108] R. Szewczyk, J. Polastre, and D. Culler. Telos: Enabling ultra-low power wireless research. Available at http://www.sics.se/~luca/datacom3/polastre05telos.pdf, 2005.

[109] MIT Electric Vehicle Team. A guide to understanding battery specifications. Available at http://mit.edu/evt/summary_battery_specifications.pdf, 2008.

[110] Di Tian and Nicolas D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 32–41. ACM Press, 2002.

[111] Doru E. Tiliute. Battery management in wireless sensor networks. *Electronics and Electrical Engineering*, 4(76):5900 – 5930, 2007.

[112] Ben L. Titzer. AVRORA: Scalable sensor network simulation with precise timing. In *Proceeding of the 4TH International Conference on information processing in Sensor Networks (IPSN)*, pages 477–482, 2005.

[113] Battery University. Elevating self-discharge. Available at http://batteryuniversity.com/learn/article/elevating_self_discharge.

[114] Battery University. How to measure State-of-Charge. Available at http://batteryuniversity.com/learn/article/how_to_measure_state_of_charge.

[115] JP. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel. Routing metrics used for path calculation in low-power and lossy networks. RFC 6551, March 2012.

[116] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi min Wang. Distributed topology control for power efficient operation in multihop wireless Ad Hoc networks. In *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1388–1397, 2001.

[117] N. H. E. Weste and K. Eshraghian. *The Principles of CMOS VLSI Design*. MA:Addison-Wesley, Reading, U.K, 1993.

[118] T. Winter, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur, and R. Alexander. RPL: IPv6 routing protocol for low-power and lossy networks. RFC 6550, March 2012.

[119] T. Winter (editor), P. Thubert (editor), A. Brandt, T. Clausen, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, and J. P. Vasseur. RPL: IPv6 Routing Protocol for Low power and Lossy Networks. IETF Internet Draft, October 2010.

[120] Jiuqiang Xu. Distance measurement model based on rssi in wsn. *Wireless Sensor Network*, 02(08):606–611, 2010.

[121] Q. Xue and A. Ganz. On the lifetime of large scale sensor networks. *Computer communications*, 29(4):502–510, 2006.

[122] Joel K. Young. A practical guide to battery technologies for wireless sensor networking. Available at http://www.sensorsmag.com/networking-communications/batteries/a-practical-guide-battery-technologies-wireless-sensor-netwo-1499, 2008.

[123] Qi Zhang, Qingzhi Guo, Shengyi Liu, Roger A. Dougal, and Ralph. E. White. Resistive companion modeling of batteries in a virtual test bed. *Journal of Power Sources*, 141(2):359 – 368, 2005.

[124] Jun Zheng and Abbas Jamalipour. *Wireless Sensor Networks: A Networking Perspective*. Wiley-IEEE Press, 2009.