Technische Universität Dresden

Analysis of the Cost of Handover in a
Mobile Wireless Sensor Network

Qian Dong

von der
Fakultät Informatik der
Technischen Universität Dresden

zur Erlangung des akademischen Grades eines

Doktoringenieurs
(Dr.-Ing.)

genehmigte Dissertation

Tag der Einreichung: 14.12.2012
Tag der Verteidigung: 26.02.2013

Supervisors

Prof. Alexander Schill
Dr. Waltenegus Dargie
Chair of Computer Networks
Faculty of Computer Science
Technical University of Dresden
Dresden, Germany

Reviewers

Prof. Alexander Schill
Chair of Computer Networks
Faculty of Computer Science
Technical University of Dresden
Dresden, Germany

Prof. Ralf Lehnert
Chair for Telecommunications
Communications Laboratory
Faculty of Electrical and Computer Engineering
Technical University of Dresden
Dresden, Germany

Prof. Christel Baier
Chair of Algebraic and Logic Foundations of Computer Science
Faculty of Computer Science
Technical University of Dresden
Dresden, Germany

Prof. Martin Wollschlaeger
Industrial Communications
Institute of Applied Computer Science
Faculty of Computer Science
Technical University of Dresden
Dresden, Germany

Prof. Michel Misson
LIMOS Laboratory
Networks and Protocols Research Group
Blaise Pascal University
Clermont-Ferrand, France

Declaration

I hereby assure that I created the present work without inadmissible help of third parties and without use of other auxiliary means than those specified; thoughts that are directly or indirectly taken over from other sources are marked as such. I did not make use of the assistance of a commercial dissertation consultant. Third-parties did neither directly nor indirectly receive benefits of pecuniary value for works that are related to the content of the present dissertation. Up to now, neither in Germany nor in any other country the work has been presented in this or a similar form to another examination agency, and it has not yet been published either. I confirm that I acknowledge the applicable doctorate regulations of the Fakultät Informatik of the Technische Universität Dresden.

Dresden, 14.12.2012

Abstract

Handling mobility in wireless sensor networks can pose formidable challenges in protocol design, especially, at the link layer. Since most of the proposed Medium Access Control (MAC) protocols do not accommodate mobility, a node has two options to deal with a deteriorating link: (a) to continue data transmission until the link breaks and then establishes a new link with a new relay node; or (b) to seamlessly transfer the communication to a better link parallel to the data transmission over the existing link. Different from option (a) where a node can only search for a new link after the original link disrupts, option (b) enables a node to perceive the change in the quality of a link in advance. The link quality prediction is implemented by an adaptive handover mechanism. Both approaches will inevitably introduce latency.

This thesis aims to quantify and compare such latency. Specifically, it investigates the latency of packet transmission in a mobile wireless sensor network with and without the support of a handover mechanism. To start with, the thesis elaborates the effect of mobility on the performance of the existing MAC protocols, and the need to maintain an unbroken link during data transmission. To implement the handover, a target MAC protocol is required to be selected first. Since the Receiver-Initiated MAC protocol (RI-MAC) uses only short beacon and data packets during communication that substantially reduce overhearing, collision probability and data recovery cost, it is chosen as the carrier for the latency evaluation.

Even though RI-MAC performs well in many aspects, it has several demerits. To address the monotonous increment in the backoff window size and to reduce the occurrence frequency of the dwell time, a burst data transmission pattern is adopted to optimize RI-MAC. With the optimization, the

protocol reduces the long idle listening time that a node has to wait before data transmission, and thus, works well in a static scenario. However, due to the high probability of link disconnection, the burst data transmission does not perform well in case of mobility.

For the sake of accommodating mobility, an adaptive handover mechanism is developed on top of the optimized RI-MAC. Once a node evaluates that the data packets cannot be completely transmitted before the link terminates, it will search for a new relay node while keeping communicating with the original collaborator. It is implemented by embedding a neighbor discovery request in a data packet that will be transmitted in a broadcast channel. Neighbors of the node will participate in the handover process as long as they are in an active state and their distance to the transmitter does not exceed a pre-defined threshold.

As a proof-of-concept for the handover mechanism, a mathematical model is established. The transmission rate, the moving speed of human beings, the duty cycle and the network density are all taken into consideration. The analytical result shows that the communication latency decreases with an increment in the network density and the duty cycle when the handover mechanism is used, whereas the latency exhibits a reverse trend when the handover mechanism is not used.

To validate the mathematics-based evaluation, the NS2 network simulator is employed. The simulation result is found to perform in accordance with the analytical result. It is asserted that the latency of packet transmission with the handover support is much less than that without the handover support. The communication latency can be saved by at least 0.28s when the handover mechanism is applied. This figure can even grow as the duty cycle and the network density increase. From this perspective, the handover mechanism is verified to improve the latency of packet transmission as far as mobility is concerned.

Acknowledgements

Writing this doctoral thesis, ever since the beginning of my PhD program, has not been an easy process. I would like to firstly express my gratitude to my supervisors, Dr. Waltenegus Dargie, Prof. Alexander Schill, Prof. Ralf Lehnert and Prof. Michel Misson. Dr. Dargie inspired and encouraged me to explore ambitious research topics, and provided me with high-level guidance towards my academic dream. I have been under his supervision ever since my master thesis. I am grateful for all the discussions and cooperation we have had, which inspire my passion to science. Prof. Schill has always been a dedicated and supportive supervisor in my research. I am grateful for this support and encouragement of my sometimes impulsive ideas and plans. Prof. Lehnert constantly encouraged me to tackle challenging issues and posed valuable feedback in my status talk. This trained my in-depth thinking ability and helped me to ponder every detail of my PhD work. In addition, I wish to specially thank Prof. Misson from the Networks and Protocols Research Group in Blaise Pascal University, France. I am grateful for his valuable help through commenting and advising on the topic and structure of my thesis. We have cooperated on research activities for three months, which are appreciated learning experiences in my PhD program.

In addition to my supervisors, I would like to sincerely thank my fellow colleagues and friends both in Germany and France. Without their generous help, it is impossible for me to make progress on the challenging research topic. Foremost, Dr. Alexandre Guitton has been an inspiring guide, co-operative partner and loyal friend. My research progress has always been facilitated by his insights, comments and encouragement. It has been a true pleasure to have him as a friend and colleague. Besides, Dr. Weiran Cai, Jianjun Wen, Jinshu Zhao, Möbius Christoph, Dr. François Delobel and

Dr. abrehet mohammed have all given me highly-appreciated help in my research path and warm concern for my life abroad.

I would like to acknowledge the financial support from the projects of Flex-Cloud (Flexible Service Architectures for Cloud Computing) and Senseive (A Middleware for Wired and Wireless Sensor Networks), which have been the main funding source of my PhD program. In addition, I would like to thank the support from the DAAD (Deutscher Akademischer Austausch Dienst) foundation and the Protocols Research Group in Blaise Pascal University of France.

Last but definitely not least, my most gratitude goes to my family. My mother Peirong and father xiongbao have been supportive and understanding of my career and life choices. Any humble achievement I have made is owing to their nurturing and encouragement.

Contents

List of Publications	ix
List of Figures	xi
List of Tables	xv
List of Abbreviations	xvii
1 Motivation	1
1.1 Applications of Wireless Sensor Networks	1
1.1.1 Static Applications	1
1.1.2 Mobile Applications	1
1.2 Mobility and its Challenges	3
1.3 Objectives of Thesis	5
1.4 Skeleton of Thesis	7
1.5 Organization of Thesis	9
2 Related Work	11
2.1 Wireless Sensor Networks	11
2.1.1 A Wireless Sensor Network	11
2.1.2 A Wireless Sensor Node	11
2.1.3 The Protocol Stack of a Sensor Node	13
2.2 MAC Layer Design	13
2.2.1 Attributes of MAC Protocols	14
2.2.1.1 Energy	14
2.2.1.2 Latency	15
2.2.1.3 Scalability	16

CONTENTS

2.2.1.4	Throughput	16
2.2.1.5	Fairness	16
2.2.2	Non-mobility-aware MAC Protocols	17
2.3	Mobility Patterns and Models	22
2.3.1	Mobility Patterns	22
2.3.1.1	Pedestrian mobility pattern	22
2.3.1.2	Vehicular mobility pattern	22
2.3.1.3	Dynamic medium mobility pattern	22
2.3.2	Mobility Models	23
2.3.2.1	Entity mobility models	23
2.3.2.2	Group mobility models	23
2.4	Sink Mobility and Node Mobility	24
2.4.1	Sink Mobility	24
2.4.1.1	Mobile base station	24
2.4.1.2	Mobile data collector	25
2.4.1.3	Rendezvous-based mobility	26
2.4.2	Node Mobility	27
2.5	Mobility Evaluation	27
2.5.1	Mobility Estimation Technique	27
2.5.1.1	Linear model	27
2.5.1.2	Auto-regression model	28
2.5.1.3	Information theoretic model	28
2.5.1.4	Markov chain based model	28
2.5.1.5	Pattern matching model	28
2.5.2	Data Source for Mobility Estimation	28
2.5.2.1	Global Positioning System (GPS)	29
2.5.2.2	Pedometers	29
2.5.2.3	Robotics	29
2.5.2.4	Radio Frequency Identification (RFID)	29
2.5.2.5	Anchor node	29
2.5.2.6	Time-Of-Arrival (TOA)	30
2.5.2.7	Angle-Of-Arrival (AOA)	30
2.5.2.8	Received Signal Strength Indicator (RSSI)	30

2.5.2.9	Signal-to-Noise Ratio (SNR)	30
2.5.2.10	Ultrasound	31
2.5.2.11	Accelerometers	31
2.5.2.12	Triangulation and trilateration	31
2.5.2.13	Hybrid	31
2.6	Mobility-Aware MAC Protocols	32
2.6.1	MS-MAC	32
2.6.2	M-MAC	35
2.6.3	M-TDMA	37
2.6.4	MobiSense	38
2.6.5	MCMAC	39
2.6.6	Discussion and Comparison	40
2.6.7	Lessons Learned and Open Issues	44
3	Selection of a Hybrid MAC Protocol	47
3.1	Reason for the Selection of RIMAC	47
3.2	Description of RI-MAC	48
3.3	Merit and Demerit of RI-MAC	50
3.4	Optimization of RI-MAC	52
4	Effect of Mobility on Latency	55
4.1	Problems of Burst Transmission due to Mobility	55
4.2	Communication Pattern before Link Breaks	56
4.3	Communication Pattern after Link Breaks	57
5	Handover Mechanism	61
5.1	Working Mechanism	61
5.2	Determination of the Distance Threshold	67
5.3	Handover Trade-off	68
5.4	Merit of the Handover Mechanism	68
6	Latency Analysis with Handover	71
6.1	Handover Latency	71
6.1.1	For Network Density $M = 2$	72
6.1.2	For Network Density $M > 2$	82

CONTENTS

6.2	Latency before Handover	86
6.3	Latency after Handover	88
7	Latency Analysis without Handover	91
7.1	Latency in the Original Link	91
7.1.1	Latency before $T1$	91
7.1.2	Latency between $T1$ and $T3$	91
7.2	Latency for a New Link Establishment	93
8	Analytical Result	99
8.1	Parameter Setup	99
8.2	Analytical Result and Discussion	100
8.2.1	Latency with Handover	100
8.2.1.1	Latency before $T1$	100
8.2.1.2	Latency between $T1$ and $T2$	102
8.2.1.3	Total latency	105
8.2.2	Latency without Handover	106
8.2.2.1	Latency before $T3$	106
8.2.2.2	Latency after $T3$	107
8.2.2.3	Total latency	108
8.2.3	Latency Comparison	109
9	Simulation Result	111
9.1	Parameter Setup	111
9.2	Topology Setting	112
9.3	Simulation Result and Discussion	116
9.3.1	Latency with Handover	116
9.3.1.1	Latency between $T1$ and $T2$	117
9.3.1.2	Latency after $T2$	118
9.3.1.3	Total latency	119
9.3.2	Latency without Handover	121
9.3.2.1	Latency after $T3$	122
9.3.2.2	Total latency	123
9.3.3	Latency Comparison	124

10 Conclusion	127
10.1 Mobility Challenges	127
10.2 Options for Link Deterioration	127
10.3 RI-MAC and its Demerits	128
10.4 Optimization of RI-MAC	128
10.5 Handover Mechanism	129
10.6 Mathematical Model Establishment	131
10.7 Mathematics-based Evaluation	132
10.8 Simulation-based Evaluation	133
10.9 Contributions of Thesis	135
10.10Future Work	136
Bibliography	137
A Simulation Details	147
A.1 Simulation with Handover	147
A.2 Simulation without Handover	150

CONTENTS

List of Publications

The thesis contains the research in the following publications with new unpublished material (in Chapter 9).

Journal Articles

1. Q. Dong and W. Dargie. A Survey on mobility and mobility-aware MAC protocols in wireless sensor networks. *Communications Surveys & Tutorials*, IEEE, PP(99): 1-13, 2012.

Conference Proceedings

1. Qian Dong, W. Dargie. Performance analysis of a handover mechanism for a mobile wireless sensor network. In *Proceedings of the 10th Annual IEEE Consumer Communications & Networking Conference (CCNC'13)*, Las Vegas, Nevada USA, January 11-14, 2013.
2. Qian Dong and Waltenegus Dargie. Analysis of the cost of handover in a mobile wireless sensor network. In *Proceedings of the 10th International Conference on Wired/Wireless Internet Communication (WWIC'12)*, pages 315-322, Island of Santorini, Greece, June 6-8, 2012.
3. Qian Dong, W. Dargie. Evaluation of the reliability of RSSI for indoor localization. In *Proceedings of the third International Conference on Wireless Communications in Unusual and Confined Areas (ICWCUCA'12)*, Clermont-ferrand, France, August 28-30, 2012.

List of Publications

4. Qian Dong and Waltenegus Darige. Analysis of collision probability in unsaturated situation. In Proceedings of the 25th ACM Symposium On Applied Computing, pages 772-777, Sierre, Switzerland. March 22-26, 2010.
5. Qian Dong, W. Dargie, and A. Schill. Effects of sampling rate on collision probability in hybrid MAC protocols in WSN. In Globecom Workshops, pages 213-218, December, 2010.
6. Qian Dong, W. Dargie, and A. Schill. The energy cost of control packets in hybrid MAC protocols. In Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications (HPCC'10), pages 609-615, Melbourne, Australia, September, 2010.
7. Qian Dong, W. Dargie, Effects of mobility on latency in a wireless sensor network that accommodates mobile nodes. In Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), Shanghai, China, 2013, submitted.

List of Figures

1.1	A link may break during data communication due to mobility	6
2.1	The structure of a sensor node	12
2.2	The data communication pattern in a wireless sensor network	12
2.3	TDMA working mechanism	17
2.4	The working mechanism of 802.11 protocol	18
2.5	The working mechanism of SMAC	19
2.6	The working mechanism of B-MAC and X-MAC	20
2.7	Mobile base station vs. mobile data collector vs. rendezvous-based mobility pattern	26
2.8	Synchronization frequency adjustment in MS-MAC	34
2.9	Dynamic frame time in M-MAC	35
2.10	Mobility estimation for the next frame in M-MAC	36
2.11	Working principle of M-TDMA	37
2.12	Working mechanism of MobiSense	39
2.13	Architecture of MCMAC	40
3.1	Working principle of RI-MAC	49
3.2	The format of a RI-MAC beacon for an IEEE 802.15.4 radio	50
3.3	The design of dwell time in RI-MAC	50
3.4	Optimization on top of RI-MAC	52
4.1	The communication pattern before the link breaks	57
4.2	The establishment of a new link	58
5.1	Distance traveled during the transmission of the $(n - w)$ data packets .	62

LIST OF FIGURES

5.2	Working principle of the handover mechanism	63
5.3	The format of a beacon in the optimized RI-MAC for an IEEE 802.15.4 radio	66
6.1	The handover continues until it finally succeeds when $M = 2$	73
6.2	Impossible cases that lead to a handover failure for $j = 2$	76
6.3	Impossible cases that lead to a handover failure for $j = 3$	79
6.4	The handover continues until it finally succeeds when $M > 2$	83
7.1	The establishment of a new link	93
8.1	Latency before $T1$ in the cases with and without the handover support .	101
8.2	Handover latency for $M = 2$	102
8.3	Handover latency for $M \geq 2$	103
8.4	Average handover latency under different network densities	104
8.5	Total latency for transmitting a burst of data packets with the handover support	105
8.6	Latency before the link breaks when the handover mechanism is not applied	106
8.7	Latency required for setting up a new link	107
8.8	Total latency for transmitting a burst of data packets without the han- dover support	108
8.9	Latency comparison between the cases with and without the handover support	109
8.10	The difference between the latency when the handover mechanism is used and when it is not used	110
9.1	Node generation	112
9.2	Node configuration	113
9.3	An UDP agent is attached to a node if it is a transmitter	113
9.4	A sink agent is attached to a node if it is a receiver	114
9.5	Configuration of the seed to enable random simulation results	114
9.6	Topology of a mobile wireless sensor network	115
9.7	Average handover latency under different network densities	117
9.8	Average latency after handover	119

LIST OF FIGURES

9.9	Total latency for transmitting a burst of data packets with the handover support	120
9.10	Latency for the new link establishment	122
9.11	Total latency for transmitting a burst of data packets without the handover support	123
9.12	Total latency comparison between the cases with and without the handover support	124
A.1	The format of a trace file with the handover support	148
A.2	Data sets obtained from a trace file	149
A.3	The format of a trace file without the handover support	151

LIST OF FIGURES

List of Tables

2.1	comparison among the non-mobility-aware MAC protocols	21
2.2	comparison among the data sources for mobility estimation	33
2.3	comparison among the mobility-aware MAC protocols	41
6.1	description of parameters	72
8.1	parameter list	100

LIST OF ABBREVIATIONS

List of Abbreviations

ACK	Acknowledgement
ADCs	Analog-to-Digital Converters
AOA	Angle-Of-Arrival
BEB	Binary Exponential Backoff
BSE	Bovine Spongiform Encephalopathy
BW	backoff window
CCA	Clear Channel Assessment
CRD	Controlled Release Device
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear-To-Send
DCF	Distributed Coordination Function
DST	acknowledgement
FCF	Frame Control Field
FCS	Frame Check Sequence
FMD	Foot and Mouth Disease
GPS	Global Positioning System
GSP	Global Synchronization Period
ID	Global Identification
LLC	Logical Link Control
LOS	Line-Of-Sight
MAC	Medium Access Control
MCS	Mobile Cluster Slots

LIST OF ABBREVIATIONS

OSI	Open System Interconnection
RFID	Radio Frequency Identification
RI-MAC	Receiver-Initiated MAC protocol
RPGM	Reference Point Group Mobility
RSSI	Received Signal Strength Indicator
RTS	Request-To-Send
RWM	Random Waypoint Mobility
SAS	Static Active Slots
SFD	Start of Frame Delimiter
SIFS	Short Interface Space
SNR	Signal-to-Noise Ratio
SRC	source address
TOA	Time-Of-Arrival
WI	the maximum waiting interval

1

Motivation

1.1 Applications of Wireless Sensor Networks

1.1.1 Static Applications

Wireless sensor networks are one of the popular technologies of the future. A large number of applications have been proposed for wireless sensor networks. However, most of them assume a static deployment of sensor nodes or an infrequent change of network topology. For example, Mainwaring et al. (56) deploy wireless sensor networks to monitor the activities of sea birds by gathering data from humidity, temperature, barometric pressure, and light sensors. Kim et al. (47) use wireless sensor networks for structural health monitoring in which the structural integrity of bridges and buildings is inspected using accelerometer sensors. Werner-Allen et al. (95) employ seismic and infrasonic sensors to monitor active volcanoes and report that 230 volcano events are captured within three weeks time. Stoianov et al. (81) use hydraulic, acoustic and vibration sensors to monitor large diameter, bulk-water transmission pipelines. Similar applications include precision agriculture in which temperature, humidity, and pH are extracted from environments using wireless sensor networks to efficiently utilize irrigation, herbicide, pesticide, and fertilizer (12, 13).

1.1.2 Mobile Applications

Sensor nodes can also exhibit mobility characteristic when they continually move due to external forces, or attached on the mobile objects or persons. In some applications, mobility plays a key role. For example, biomedical sensor nodes can be attached to the

1. MOTIVATION

bodies of patients (19) and nurses (18) to monitor their activities; workers in disaster recovery scenes (54) and oil extraction and refinery areas (17) can carry sensing devices to avoid dangerous situations; sensor nodes can also be employed to report or debrief soldiers the events encountered during a mission (49).

Another well known application is to real-time monitor the animal health conditions and behaviors in an outdoor environment (62). The need for individual and herd-wide monitoring of livestock from a physical and physiological perspective arises from the nature of the difficulties involved with managing farms with large grazing areas. There has always been a need for livestock producers to observe their animals as often as possible. Management concerns such as interactions between cows, bulls, and their offspring are of great interest to farmers. The matters including water quality in dams or rivers, the amount, nature and nutritive value of the pasture, the state of fences, as well as potential stock rustling may also influence the health condition of animals in an indirectly way.

Inattention to the health and wellbeing of animals can lead to the reduced productivity, the illness, and even the death of valuable stock. In recent history, there have been a number of major animal disease outbreaks in Europe. The two most significant incidents are the Bovine Spongiform Encephalopathy (BSE) outbreak and Foot and Mouth Disease (FMD) outbreak, which take place in the year of 1986 and 2011, respectively. According to statistics, 4.5 million cows are incinerated after the identification of BSE and 4 million livestock are slaughtered to stop the spread of FMD (48).

Nevertheless, it is often the case that farmers have neither the time nor resources to watch the animals regularly. Even they do, they may not be in a position to identify some of the more deep-rooted problems associated with the metabolic or reproductive state of the individuals. Unlike humans who can describe their situation to a doctor, animals are usually unable to explain symptoms to their observers.

Therefore, it is always desirable to develop a platform technology to observe the health conditions and behaviors of animals without much human intervention. This can be implemented via a mobile wireless sensor network by mounting sensor nodes on or inside the body of livestock. The samples reported by mobile sensors that instantly capture the changes of animals' metabolic, behavioral or welfare status, together with the outputs obtained from fixed sensors (static sensors) that describe food and water

availability, help to reduce the current reliance on manpower and improve the decision-making process.

Specifically, in cattle health monitoring applications, a sensor node can be placed inside a standard plastic barrel of a Controlled Release Device (CRD) to routinely medicate animals. One CRD is inserted into the rumen of each cattle via a cannula for monitoring the cattle temperature which is known as the most important factor indicating the animal health condition (59). Similarly, wireless sensor nodes with acceleration detection functionalities can be bound with the legs of cattle to predict their estrus time. It has been found that cattle are restless during the estrus period, and this can be informed by sensor nodes with a high deviation of acceleration values (91). Successful artificial insemination based on correct estrus detection can result in efficient reproductive cycle in dairy farms (75). A wireless sensor network can also be utilized for predicting the calving-time of animals. A sensor node placed in the vagina of a preparturient cow is usually expelled a couple of hours prior to parturition. The calving-time can also be predicted via the observation of the different fluctuation of the sensor node before and after the membrane rupture of the cow (91).

1.2 Mobility and its Challenges

In wireless sensor networks, nodes can be static or mobile, depending on the application requirements. Nodes may leave a network due to hardware failure or battery depletion. Likewise, new nodes may enter a network either to replace the failed nodes or to provide additional support. Such regular network topology changes are understood as **weak mobility** (4).

Most of the energy-efficient medium access control protocols assume a static deployment or an infrequent topology change. Typically, these protocols define duty cycles for nodes to be able to periodically sleep, and thereby, avoid idle listening and overhearing. For example, in SMAC (98) and TMAC (88), nodes update their knowledge about their neighbors by exchanging synchronization packets. Such an update is carried out only occasionally at the beginning of a frame. Additionally, the four way RTS/CTS/DATA/ACK handshake scheme may be broken during the movement of nodes. Hence, mobility may not be detected timely in these protocols. Likewise, preamble-based protocols such as BMAC (69), XMAC (10) and WiseMAC (28) avoid

1. MOTIVATION

the need for periodic synchronization by enabling a receiver to learn the communication attempt of its neighbors once it receives a preamble. In receiver-initiated MAC protocols such as RI-MAC (84), each time a receiver wakes up from a sleep state, it broadcasts a beacon to all its neighbors to inform that it is ready to receive packet. However, these protocols enable nodes to perceive a change in their surrounding only at the beginning of each active period. Consequently, due to the duty cycle, there is a delay in packet transmission whenever a topology change occurs. This delay can be high in multi-hop networks. However, since weak mobility takes place infrequently, the delay it introduces may be tolerable.

In contrast, **strong mobility** is characterized by concurrent node joins and failures as well as physical movement of nodes. Physical mobility is caused by the deliberate movement of objects or persons on which sensor nodes are attached, or external forces such as wind, water, or air (4).

Dealing with strong mobility can pose some formidable challenges in protocol design, particularly, at the link and network layers. For example, mobility can lead to deterioration in the quality of an established link, and therefore, data transmission is prone to failure. This will in turn increase the rate of packet retransmission and the overall energy consumption. Mobility can also lead to frequent route changes, resulting in a considerable packet delivery delay and an increased design complexity of network layer protocols. In addition, a mobile node cannot immediately begin transmitting data packets upon joining a network. Instead, it has to wait for a certain amount of time before it can be fully integrated. This is because only some rather than all of the time slots are designed to accommodate mobility in schedule-based MAC protocols. Likewise, in contention-based MAC protocols, a mobile node can only enter a network after it receives a synchronization packet from a neighbor node already existing in the network. Since such a packet is only broadcasted periodically, the mobile node has to spend a long time interval on average in idle listening before joining the network. In contention-based MAC protocols, mobility may also increase the packet collision rate while in schedule-based MAC protocols, two-hop neighborhood information may become inconsistent once nodes enter or leave, leading to schedule inconsistencies.

1.3 Objectives of Thesis

All these difficulties caused by node movement require mobility handling techniques to adapt to network dynamics. Since most of the existing or proposed MAC protocols do not accommodate mobility, an established link between two nodes may break during the data communication. A node has two options to deal with a deteriorating link: (a) to continue data transmission until the link breaks and then establishes a new link with a new relay node; or (b) to seamlessly transfer the communication to a better link parallel to the data transmission over the existing link.

For option (a), a node can only search for a new link after the original link disrupts. The new link establishment may call for a long period of time, since the node may not find a new neighbor node to resume its data transmission as soon as the original link terminates. Instead, it may happen that one of the other channel competitors successfully seizes the medium once the link deteriorates. The node, therefore, can only contend for the channel after the winner node first completes its data transmission. It may also happen that the node discovers no new relay node to proceed with its data communication once the original link breaks. This is especially true when nodes apply asynchronous duty cycles in MAC layer protocols. In such a case, the node has to wait until at least one surrounding neighbor node wakes up from a sleep period. The waiting duration for the new link establishment depends on multiple factors, such as the queue size, the packet arrival rate, the network density, as well as the duty cycle.

Different from option (a), option (b) enables a node to perceive the change in the quality of a link in advance. The link quality prediction is implemented by designing a seamless handover mechanism that extends an existing MAC protocol. Once a node evaluates that the data packets cannot be completely transmitted before the link terminates, it will search for a new relay node while keeping communicating with the original collaborator. It is achieved by embedding a neighbor discovery request in a data packet that will be transmitted in a broadcast channel. Neighbors of the node will participate in the handover process as long as they are in an active state and their distance to the transmitter does not exceed a pre-defined threshold.

To exhaustively explain the two options for dealing with a deteriorating link, a specific mobility scenario is given. This is illustrated in Figure 1.1. Here A , B , C and D are the sensor nodes that can either be placed inside or on the body of the livestock

1. MOTIVATION

Figure 1.1: A link may break during data communication due to mobility

depending on the application requirements. Since animals move from time to time in the outdoor environment, these nodes may exhibit mobility characteristic. We assume a node in a wireless sensor network can always find a multi-hop route to forward its source data to the sink. Initially, node A transmits data packets with node B and a communication link is established between them. As node B gradually moves away from node A , node B will eventually get out of the radio transmission range of node A . As a result, the connection between nodes A and B will definitely break at a particular moment. By applying option (a), node A can do nothing but wait until the current link disrupts. Only after receiving no signal from node B , can it look for another neighbor node on the way to the sink, say, node C , to proceed with its remaining data transmission.

An alternative is to make use of an adaptive handover technique developed on top of an existing MAC protocol. The handover mechanism enables a node to transfer the communication from one sensor node to another without breaking the original link. Once detecting the deterioration in the quality of the current link, node A learns that its data packets may not be completely transmitted before the link terminates. Therefore,

it will search for a new relay node while keeping transmitting data packets with the current node. As soon as node C is discovered, node A will switch its communication from node B to C to resume its remaining data transmission.

Obviously, both approaches (a) and (b) will inevitably introduce latency. This thesis aims to quantify and compare such latency. Specifically, the thesis investigates the latency of packet transmission in a mobile wireless sensor network with and without the support of a handover mechanism.

1.4 Skeleton of Thesis

To figure out whether and how much the latency of packet transmission during mobility can be improved if a handover mechanism is utilized, the thesis accomplishes the following tasks step by step.

- Select a hybrid MAC protocol.
- Develop an adaptive handover mechanism.
- Embed the handover mechanism in the selected MAC protocol.
- Set up a mathematical model to quantitatively analyze the packet transmission latency with and without the support of the handover mechanism.
- Implement the NS2 network simulator to simulate the packet transmission latency with and without the support of the handover mechanism.
- Compare the mathematics- and simulation-based evaluation results.

Specifically, the thesis starts with the elaboration of the effect of mobility on the performance of the existing MAC protocols, and the need to maintain an unbroken link during data transmission. To implement the handover, a target MAC protocol is required to be selected first. Since the Receiver-Initiated MAC protocol (RI-MAC) uses only short beacon and data packets during communication that substantially reduce overhearing, collision probability and data recovery cost, it is chosen as the carrier for the latency evaluation.

1. MOTIVATION

Even though RI-MAC performs well in many aspects, it has several demerits. To address the monotonous increment in the backoff window size and to reduce the occurrence frequency of the dwell time, a burst data transmission pattern is adopted to optimize RI-MAC. With the optimization, the protocol reduces the long idle listening time that a node has to wait before data transmission, and thus, works well in a static scenario. However, due to the high probability of link disconnection, the burst data transmission does not perform well in case of mobility.

For the sake of accommodating mobility, an adaptive handover mechanism is developed on top of the optimized RI-MAC. Once a node evaluates that the data packets cannot be completely transmitted before the link terminates, it will search for a new relay node while keeping communicating with the original collaborator. It is implemented by embedding a neighbor discovery request in a data packet that will be transmitted in a broadcast channel. Neighbors of the node will participate in the handover process as long as they are in an active state and their distance to the transmitter does not exceed a pre-defined threshold.

As a proof-of-concept for the handover mechanism, a mathematical model is established. The transmission rate, the moving speed of human beings, the duty cycle and the network density are all taken into consideration. The analytical result shows that the communication latency decreases with an increment in the network density and the duty cycle when the handover mechanism is used, whereas the latency exhibits a reverse trend when the handover mechanism is not used.

To validate the mathematics-based evaluation, the NS2 network simulator is employed. The simulation result is found to perform in accordance with the analytical result. It is asserted that the latency of packet transmission with the handover support is much less than that without the handover support. The communication latency can be saved by at least 0.28s when the handover mechanism is applied. This figure can even grow as the duty cycle and the network density increase. From this perspective, the handover mechanism is verified to improve the latency of packet transmission as far as mobility is concerned.

1.5 Organization of Thesis

The remaining part of this thesis is organized as follows: in Section 2, related work is summarized. In Section 3, a brief introduction to RI-MAC and its optimization is described. In Section 4, the effect of mobility on the performance of the optimized RI-MAC is investigated. In Section 5, the handover mechanism is presented. In Sections 6 and 7, the performance of the optimized RI-MAC with and without the handover support is mathematically evaluated. In Sections 8 and 9, the analytical and simulation results are compared and observations are discussed. Finally, in Section 10, concluding remarks are given.

1. MOTIVATION

2

Related Work

2.1 Wireless Sensor Networks

2.1.1 A Wireless Sensor Network

A wireless sensor network normally consists of a large number of nodes which can sense a variety of physical phenomena, process the raw data locally, and deliver the result over a wireless multi-hop link (22). Sensor nodes are able to self-organize in order to establish and maintain the networks without human intervention. Due to the small size of nodes and the wireless communication, sensor networks exhibit several attractive features. For example, nodes can be easily installed in places which can otherwise be inaccessible or expensive to wired systems. Likewise, deployment and maintenance operations can take place without disrupting the normal operation of the structure or process they monitor. Since the number of sensor nodes in a wireless sensor network is usually large, Global Identification (ID) of nodes may not be used considering the overall overhead.

2.1.2 A Wireless Sensor Node

A sensor node is composed of four basic components: a sensing unit, a processing unit, a transceiver unit, and a power unit. The structure of a sensor node is illustrated in Figure 2.1. A sensing unit is usually made up of two subunits: sensors and Analog-to-Digital Converters (ADCs). The ADC converts the analog signals produced by sensors based on the observed events to digital signals, and the digital signals are transferred to the processing unit. The processing unit along with the storage unit manages the procedures to enable sensor nodes to accomplish the sensing tasks collaboratively. A

2. RELATED WORK

transceiver unit connects a sensor node to a wireless network. The power unit, as the most important component of a sensor node, is supported by power scavenging units such as solar cell.

Figure 2.1: The structure of a sensor node

The main task of a sensor node in a wireless network is to detect events, perform quick data processing, and transmit sampled data. After sensing in environment, a sensor node will transfer the information via a multi-hop link. Instead of directly transmitting the raw data, each sensor node forwards only the required data to its one-hop neighbor after making use of the in-network processing ability to locally carry out simple computations. The data packets are transferred and processed hop by hop until they are finally reached at the sink. The data communication pattern in a wireless sensor network is described in Figure 2.2.

Figure 2.2: The data communication pattern in a wireless sensor network

Since a wireless sensor node is a microelectronic device, it has limited power source. Usually, it is less than 0.5 Ah (3). The embedded radios enable a wireless communication between sensor nodes and the type of the communication is usually broadcast. This, however, introduces several side effects for MAC layer protocols, such as collision, overhearing and hidden terminal problems.¹ Once the radio is turned on, the energy of a sensor node is being consumed. Since the replenishment of power resources may be impossible, the lifetime of a sensor node reveals a strong dependence on the battery lifetime. As a wireless sensor network is constituted of multiple individual sensor nodes, the malfunctioning of a few nodes may cause significant topology changes, rerouting of data packets, and even reorganization of the entire network.

2.1.3 The Protocol Stack of a Sensor Node

The protocol stack is applied to the sink and all the sensor nodes. It combines energy and routing awareness, integrates data with networking protocols, enables efficient communication through the wireless medium, and promotes cooperative efforts of sensor nodes. The protocol stack consists of five layers. The physical layer addresses the need of robust modulation, transmission, and receiving techniques. The data link layer takes charge of the multiplexing of data streams, the data frame detection, the medium access and error control. It ensures reliable point-to-point and point-to-multipoint connections in a wireless sensor network. The network layer is responsible for routing the data supplied by the transport layer. The transport layer, however, takes care of maintaining the data flow according to the application requirement of sensor networks. Depending on the diverse sensing tasks, different types of application software are built and utilized on the application layer (3).

2.2 MAC Layer Design

The Medium Access Control (MAC) layer is a sub-layer of the data link layer specified in the seven-layer Open System Interconnection (OSI) reference model. It acts as an interface between the Logical Link Control (LLC) sub-layer and the physical layer. The

¹These side effects will be explained exhaustively in Section 2.2.

2. RELATED WORK

MAC layer provides addressing and channel access control mechanisms that enable sensor nodes to communication with each other within a multiple access wireless network that incorporates a shared medium.

Protocols designed for the MAC layer must achieve two objectives. One is to establish a communication link for data transfer. As there may be hundreds or thousands of sensor nodes densely scattered in a sensing field, all the single-hop communication links will enable the data transmission in a multi-hop manner. This is necessary for creating a basic network infrastructure. The other objective of the MAC layer protocols is to regulate the access to a shared media so that communication resources can be fairly and efficiently shared between contending nodes.

2.2.1 Attributes of MAC Protocols

For the sake of designing a good MAC protocol, many attributes should be taken into account, such as energy, latency, scalability, throughput, fairness and so forth. Which attributes are more significant than others depend on the design intention, the performance requirement and the application orientation of the MAC protocol.

2.2.1.1 Energy

Since most of the MAC protocols in wireless sensor networks are designed only for static deployment, energy efficiency is considered as the most important factor. In fact, there are four major sources of the energy waste for sensor nodes (92).

The first one is collision. When two or more packets sent from different senders are received by one and the same receiver at the same moment, these packets are corrupted. This happens because of the broadcasting property of the radio transmission. After corruption, the packets need to be retransmitted, which introduces extra energy consumption.

The second source is overhearing. It means a node receives packets which are destined to others. Overhearing consumes energy because receiving needless information is a waste.

The third source is idle listening. This means a node's radio is turned on but it neither transmits nor receives. As idle listening is the major source of the energy waste, multiple contention-based MAC protocols apply duty cycle to keep the sleep phase of a node as long as possible.

Control packet (RTS, CTS and ACK) is another source of the energy waste. It consumes significant amount of energy which however does not directly benefit the real data transmission. Nevertheless, it is a good method to avoid problems such as collision and overhearing, which in turn are another two major sources of the energy waste.

2.2.1.2 Latency

Recent advances in wireless sensor networks have resulted in the capability to remotely sense environment. Since sensor nodes are often deployed in remote or hard-to-reach regions, wireless sensor networks are required to operate unattended for long periods. Therefore, network lifetime extension becomes a key concern for many delay-sensitive applications (such as fire detection or a tsunami alarm). This in turn calls for the latency improvement for the development of MAC layer protocols. The term latency is defined as the delay elapsed between the time at which an incoming event is sensed by a source node and the time at which the sensed information is retrieved by the data collecting sink (25).

In many sensor network applications, nodes are idle for a long time if no sensing event occurs. Given the fact that the data sampling rate is very low, it is unnecessary to keep nodes listening all the time. To prolong the overall network lifetime, plenty of MAC protocols reduce the listen time by letting nodes go to periodic sleep mode. However, latency can only be saved at the cost of the extra energy consumption. This is because to align the sleep-wake schedules of nodes, a synchronization packet has to be broadcasted periodically.

To address this problem, the asynchronous MAC protocols are proposed. In these protocols, nodes wake up independently with each other. Due to the asynchronous schedules, a node may not find an active neighbor once it wakes up. Hence, it cannot transmit data packets until at least one of its neighbors enters a listen period. From this perspective, even though the asynchronous MAC protocols eliminate the need for schedule maintenance and thus save energy, the latency of packet transmission is increased to some extent (44). Even worse, nodes with random position, random radio transmission range and random duty cycle will make the communication latency variable, leading to a more problematic situation (25).

2. RELATED WORK

2.2.1.3 Scalability

The number of sensor nodes deployed in a studying phenomenon may be on the order of hundreds or thousands, or even more. As a result, the network density can range from few sensor nodes to few hundred sensor nodes in a region less than 10 meters in diameter (3). Plenty of sensor nodes poses challenges with respect to the scalability of wireless networks. One of the basic concerns in building large-scale networks is that minor inefficiencies which can be well tolerated in small networks can accumulate and become dominant factors influencing the network performance. To guarantee scalability, it is required that the performance of a sensor network should not be degraded when its size grows whereas the buffer space of each sensor device remains constant throughout (42).

2.2.1.4 Throughput

In a wireless network with n nodes under a random network configuration, each node has a throughput capacity in the order of $\Theta(1/\sqrt{n \log n})$ (36). Even under an arbitrary network architecture in which the location of nodes and the traffic pattern can be optimally controlled, the network could only provide a per-node throughput of $\Theta(1/\sqrt{n})$ (36). The per-node throughput decreases as the number of nodes increases. When a node transmits, its neighbor nodes are prohibited from transmitting. This happens because the wireless medium is shared by all the nodes in the network. Therefore, the network throughput is interference-limited (38).

A small transmission range is necessary to limit the interference and consequently leads to high throughput. However, a smaller transmission range indicates that a packet has to be transmitted through multiple hops, which inevitably leads to higher transmission delay (32). In other words, the high throughput is usually obtained by sacrificing the network latency.

2.2.1.5 Fairness

A sensor node will be congested if it receives more traffic than its maximum forwarding rate. The nature of the sensor deployment leads to unpredictable patterns of the network connectivity and the network density. They can, in turn, cause uneven bandwidth provisioning on the forwarding paths. The data sources are often clustered at sensitive

areas under scrutiny and may take similar paths to the base station (53). Therefore, the nodes that are nearer to the sink have higher probability to get congested than others and thus may run out of the energy first. If the upstream neighbors attempt to send data to a congested sensor node, their efforts are deemed to be wasted, and even worse, counterproductive because they compete for channel access with neighboring sensor nodes. Finally, the data loss due to congestion may jeopardize the mission of the application (68).

Medium access at the sink is inherently unfair when using a contention-based MAC protocol. Upstream flows (from senders to the gateway) tend to occupy the whole media; the downstream flows (from the gateway to receivers) will almost stop transmission when multiple upstream and downstream flows co-exist. Some MAC protocols aim at promoting the per-node fairness (such as 802.11 (8)), while others focus more on the application-level fairness, which can be achieved by requiring all the sensor nodes to coordinate their effects for the same purpose (such as SMAC (92)).

2.2.2 Non-mobility-aware MAC Protocols

Figure 2.3: TDMA working mechanism

The existing or proposed MAC protocols in the literature can be broadly categorized into schedule-based and contention-based protocols. In schedule-based protocols, the channel is pre-allocated to nodes in terms of time, frequency or coding (such as TDMA (34), FDMA (89) and CDMA (41)). For example, in TDMA MAC, time is divided into frames and each frame is further divided into multiple time slots. Each time slot is assigned to exactly one sensor node. As a result, a node is only allowed to transmit data in its unique time slot. In this way, collision between data packets can be completely avoided. However, when new nodes join or old nodes leave a network, it is very difficult to align or reallocate time slots. Therefore, the precise synchronization required by the schedule-based MAC protocols amounts to inefficient channel utilization as well as high

2. RELATED WORK

energy consumption. The working mechanism of TDMA MAC is illustrated in Figure 2.3.

Figure 2.4: The working mechanism of 802.11 protocol

Unlike the schedule-based MAC protocols in which the channel is pre-allocated to nodes, nodes compete for a shared medium to operate communication in the contention-based MAC protocols. A typical example of these approaches is the IEEE 802.11 protocol (8). The fundamental mechanism it applies to access the medium is the Distributed Coordination Function (DCF). This is a random access scheme, based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) technique. Retransmission of collided packets is managed according to a binary exponential backoff rule. DCF describes a four way hand-shaking mechanism for data transmission. Before sending a packet, a node reserves the channel by broadcasting a short Request-To-Send (RTS) frame. The destination node acknowledges the receipt of a RTS frame by sending back a Clear-To-Send (CTS) frame. Only after the transmitter and the receiver occupy the medium, can the normal data packets and the response Acknowledgement (ACK) frames start to be transmitted. By using control packet, collision can be avoided and the retransmission cost can be considerably decreased. The working mechanism of the 802.11 protocol is described in Figure 2.4.

In many wireless sensor network applications, nodes are in an idle state for a long time if no sensing event occurs. Given the fact that the sampling rate is very low, it is not necessary to keep nodes listening all the time. Therefore, in order to reduce the time for idle listening, duty cycle is introduced in the recent contention-based MAC protocols. Instead of listening to the medium all the time, each node periodically cycles between an awake and a sleep states. Standard MAC protocols developed for duty-cycled wireless sensor networks are categorized into synchronous and asynchronous

Figure 2.5: The working mechanism of SMAC

approaches, along with hybrid combinations. Synchronous MAC protocols define a low duty cycle and synchronize the listening and sleeping schedules for all the nodes (such as S-MAC (92) and T-MAC (20)). For example, in SMAC, each listen phase is divided into two parts. The first part is used for schedule synchronization. This is accomplished via a periodically transmitted SYNC packet. An SYNC packet is very short, including the address of the transmitting node and the time for its next sleep. Once a node enters a wireless sensor network, it first listens to the channel for a period of time. If it hears a SYNC packet from one of its neighbors during this time, it will adjust its own schedule according to it and goes to the sleep state as soon as the timer fires. If no SYNC packet is received, the node will randomly and independently choose a schedule and sends its corresponding SYNC packet in a broadcast mode. If the node receives a different schedule after it selects its own one, it will accept both schedules. This makes a node at the border of a network listening for a longer time. The second part is used for data communication by applying the message passing mechanism. Instead of sending a data packet each time, SMAC divides the long message into many small fragments and transmits them in burst. This optimization highly reduces the retransmission cost if only a few bits have been corrupted during the data transmission. In addition, the control packet of RTS and CTS prevents the neighbor overhearing problem. The working mechanism of SMAC is described in Figure 2.5.

Even though the synchronous MAC protocols greatly reduce the time of node idle listening, the required schedule synchronization brings extra communication overhead and increases design complexity. In order to address these issues, the asynchronous MAC protocols are proposed by decoupling time schedules of nodes (such as B-MAC

2. RELATED WORK

Figure 2.6: The working mechanism of B-MAC and X-MAC

(70), WiseMAC (29), X-MAC (11) and RI-MAC (85)). Instead of using the control packet of RTS and CTS, the asynchronous MAC protocols use either preamble or beacon to reserve the medium. For example, B-MAC utilizes an extended preamble to achieve low power communication. If a node wishes to transmit, it precedes the data packet with a preamble that is set longer than a node's sleep period. This guarantees the detection of preamble by a receiver no matter when it wakes up. Since the target address is embedded in the header of each data packet, a node can only figure out whether it is the intended receiver after it overhears the entire preamble. In order to minimize the time for overhearing, X-MAC is developed. Instead of sending a long preamble, a series of short strobed preambles are transmitted, each including the target address. In this way, once a receiver wakes up and detects a short preamble, it will either reply with an early ACK packet if it is the intended receiver or directly go to sleep otherwise. As illustrated in Figure 2.6, the asynchronous approaches save much energy and time.

A comparison summary of the five representative non-mobility-aware MAC protocols is given in Table 2.1.

Table 2.1: comparison among the non-mobility-aware MAC protocols

MAC	Characteristics	Advantages	Disadvantages
TDMA	1. Schedule-based MAC 2. Medium is pre-allocated to nodes	1. It avoids collision 2. It does not introduce control overhead	1. It requires precise time synchronization 2. Reallocating time slots is difficult
802.11	1. Contention-based MAC 2. It applies DCF scheme	1. Collision is reduced by the binary exponential backoff rule 2. Collision does not occur on data	1. It introduces control overhead 2. Cost of overhearing and idle listening is high
SMAC	1. Synchronous contention-based MAC 2. It uses duty cycle 3. It uses message passing mechanism	1. Cost of idle listening and overhearing is highly reduced 2. Retransmission cost is highly reduced	1. It requires coarse-grained time synchronization 2. Nodes may have multiple schedules
BMAC	1. Asynchronous contention-based MAC 2. It applies a long preamble	1. Synchronization cost is reduced 2. Collision is avoided	1. The long preamble is costly 2. Cost of overhearing is high
XMAC	1. Asynchronous contention-based MAC 2. It applies short strobed preambles	1. Synchronization cost is reduced 2. Overhearing is minimized	1. Overhearing still occurs 2. The preamble partition method is essential

2. RELATED WORK

2.3 Mobility Patterns and Models

Understanding the mobility patterns of nodes is essential to design realistic models and resource efficient mobility estimation mechanisms. Based on the expected mobility patterns, protocol design can make plausible assumptions in dealing with communication handover. This section deals with higher-level mobility patterns in wireless sensor networks and the abstract models that can represent them.

2.3.1 Mobility Patterns

A mobility pattern is the movement pattern of real-life objects such as people and vehicles, which can be characterized by properties such as dimension, limitation, group behavior and predictability. There are mainly three mobility patterns relevant to wireless sensor applications.

2.3.1.1 Pedestrian mobility pattern

It describes the motion characteristics of people. Sensor nodes are attached to the body of people when they walk around. This pattern is manifested by its limited speed, obstacle avoidance, and chaotic nature. The movement in this pattern is two-dimensional, and may or may not show a group behavior (74).

2.3.1.2 Vehicular mobility pattern

It describes the movement of vehicles which are equipped with sensor nodes. Vehicles can communicate with each other conveniently by capturing traffic conditions and other information. Vehicle movement is one-dimensional and characterizes a group behavior at a high speed (74).

2.3.1.3 Dynamic medium mobility pattern

It occurs when nodes move through a medium, such as wind, water or other fluids. This mobility can be one-, two- or three-dimensional depending on the type of the medium. The difference between the pedestrian and dynamic medium mobility patterns lies in the nature of the medium. The medium is factitious in the former pattern whereas it is natural in the latter pattern.

2.3.2 Mobility Models

A mobility model is a formal mathematical model generalizing the characteristics of mobility patterns. A mobility model falls into one of the two categories, namely, trace-based and synthetic models (72). In a **trace-based** model, real-life mobility patterns are collected from a large number of participants for a long observation period. However, the real movement trajectory of mobile nodes is difficult to be modeled even when sufficient historical data are obtained and recurrent mobile pattern occurs. **Synthetic** models, on the other hand, attempt to represent the behaviors of real-world mobile objects. However, they cannot produce the precise description of mobile patterns. There are several synthetic mobility models, but here only two of them are considered.

2.3.2.1 Entity mobility models

A typical example is the Random Walk Mobility Model. It expresses the mobility of a node as it travels from its current location to a new location within a pre-defined time period or distance, by randomly choosing a direction and speed. A node changes its direction and speed once the time expires or the maximum permitted distance is reached. It's proven that a random walk on a one- or two-dimensional surface returns to the origin with complete certainty (93). A similar model is the Random Waypoint Mobility Model, which includes pause time between changes in the direction and/or speed. Mobility is greatly affected by the pause time and speed of nodes. For example, a fast movement of nodes and a long pause time results in a more stable network than a slow movement of nodes and a short pause time (14).

2.3.2.2 Group mobility models

In these models the movement of nodes with respect to other nodes is of primary interest (2). For example, the Exponential Correlated Random Mobility Model creates a motion function to predict the new location of mobile nodes in the next time slot so as to mimic an erratic movement. In the Nomadic Community Mobility Model, a set of mobile nodes collectively roam from one place to another according to the location of a reference node. Individuals move randomly within their own spaces following a random entity mobility model. Unlike the Nomadic Community Mobility Model in which the

2. RELATED WORK

mobile nodes share a common reference point, the Column Mobility Model requires a one to one mapping of the anchor and the mobile subjects (14).

2.4 Sink Mobility and Node Mobility

A number of approaches exploit the mobility of nodes for data collection. The focus of these approaches can be mainly classified into two types: sink mobility and node mobility. The former mainly affects network protocol design, since individual nodes should be able to determine the routes to the ultimate destination of the information they transmit. The latter mainly affects MAC protocol design, since individual nodes should be able to dynamically and seamlessly use shared communication media without affecting their neighbors. In sink mobility, the sink moves and routes itself in the network to collect data from static nodes. However, a more complicated and challenging case is node mobility, where sensor nodes actively move from place to place and during their movement they attempt to maintain an end-to-end communication link.

2.4.1 Sink Mobility

Many have argued that the concept of a stationary sink can introduce a number of drawbacks. For example, the nodes placed around a base station can easily become bottlenecks by quickly depleting their energy reserve (6). One way to deal with this problem is to deploy multiple mobile sinks, so that the load can be evenly distributed among nodes and the lifetime of the network can thus be increased (96). Nevertheless, a high network dynamics due to mobile sinks can degrade the performance of the network. It has been shown that the end-to-end packet delivery ratio dropped down to 50% when nodes moved at $0.5m/s$ on average (the maximum speed was $1m/s$); this figure dropped to less than 20% when the maximum speed was greater than $5m/s$ (80).

In the proposed MAC and routing protocols, sink mobility can be categorized into three types, according to the movement pattern of mobile sinks and their manners of data collection.

2.4.1.1 Mobile base station

The position of a mobile base station changes during its operation time. Data generated by sensors are relayed to the mobile base station without long term buffering (27). The

communication load distribution studied in (55) shows that a network's lifetime can be improved even if an optimally placed fixed sink is replaced by a randomly moving mobile base station. The idea is to enable relay nodes to evenly consume energy and, thereby, optimize the overall energy consumption of the network. The study also shows that the optimal movement strategy is to follow the periphery if the deployment area is circular. A similar result is obtained when the sensor nodes are placed in grids (90). The base station relocation method (33) periodically recalculates the ideal position of a mobile base station along the periphery of a sensing field.

2.4.1.2 Mobile data collector

Employing many relay nodes or long-range communication interfaces to maintain connectivity can be very expensive. A potential solution to this problem is to use mobile data collectors (data mules (77)) that gather buffered information from the source nodes over a single-hop communication link (27). Proposed approaches can be classified into three classes, with respect to the pattern of the sink mobility.

Random mobility: The mobile data collectors move randomly and collect the buffered samples opportunistically. The received data from the one-hop sensors are transferred to a wireless access point. Since the trajectory of mobile data collectors is random, the message transmission delay can be high (77).

Predictable mobility: In the predictable mobility pattern, the static nodes are assumed to know the moving route of the mobile data collector, and this information will be used to predict the time that data transfer may take place. Based on this predicted time and location, sensors schedule their sleep and listen periods. In this way, the network can optimize its energy consumption (16).

Controlled mobility: In some circumstances, data may be transmitted at different rates due to the change of events or event occurrence interval. This may lead to the loss of data if a transmitter cannot finish the transmission to the mobile data collector before its buffer overflows. In order to accommodate variable transmission rates, a heuristic solution called Earliest Deadline First (79) is proposed. The aim is to actively control the movement of the mobile data collector in real time. The node to be visited next by the mobile data collector is chosen as the one that has the earliest buffer overflow deadline. However, the approach does not work well if nodes with consecutive deadlines are located far away from each other.

2. RELATED WORK

2.4.1.3 Rendezvous-based mobility

The rendezvous-based approach is a hybrid solution that shares some properties of a mobile base station and a mobile data collector. In this mobility pattern, sensor nodes deliver data to rendezvous points which are close to the path of mobile devices and the sampled data will be buffered at rendezvous points until they are relayed to the mobile sinks (27). For example, the autonomous mobile router-based scheme (45) enables nodes which are out of the communication range of a mobile sink to buffer their data at those which can be visited directly by the mobile sink.

The difference between the three sink mobility patterns is displayed in Figure 2.7. The use of relay nodes in the mobile-base-station pattern implies that the distance between source nodes and the sink can be larger than a single hop. Therefore, the area that the mobile sink moves is limited. On the contrary, there is no extra relay nodes in the mobile-data-collector pattern. This indicates that source nodes need only forward their data to the sink directly (one-hop distance). As a result, the mobile sink has to pass along all the source nodes to collect samples, leading to a long moving range and a long communication cycle. The rendezvous-based mobility pattern, however, is a hybrid solution. It reduces the number of relay nodes to save energy, but at the same time increases energy due to the incremental sink mobility scope and data collection cycle.

Figure 2.7: Mobile base station vs. mobile data collector vs. rendezvous-based mobility pattern

2.4.2 Node Mobility

Instead of the sink, individual nodes can be mobile either continuously or intermittently; and intentionally or accidentally, depending on the nature of the application. Node mobility can have additional merits besides the ones discussed in Section 1, such as improving coverage in situations in which nodes occasionally become disconnected due to initial uneven or random deployments as well as unpredictable failures; or when they run out of battery (103).

If the mobile subjects upon which the nodes are attached are human beings, mobility can be considered as either a macroscopic or microscopic aspect. As a **macroscopic** aspect, it reflects the mobility habit based on everyday activities (e.g., going back and forth from house to office; taking a break; going to a meeting or working with colleagues). As a **microscopic** aspect, it reflects the way humans interact with their surrounding environment (e.g., indoor, outdoor, road, or network) (31). Since only the movements observed at the radio range of wireless interfaces are of interest, they are more influenced by the microscopic mobility.

2.5 Mobility Evaluation

2.5.1 Mobility Estimation Technique

Mobility patterns and models establish the basis for designing self-organizing algorithms and communication protocols that deal with mobile nodes. A mobility estimation technique employs a mobility model to predict a link quality, reserve resources, and facilitate a handover process (94). A large number of estimation approaches utilize one or more of the following models:

2.5.1.1 Linear model

A linear model predicts a node's future state based on its current and past states, and can be static or dynamic. A static model usually forecasts a mobile node's position based on GPS information, by assuming it moves with the same speed and in the same direction (82). By employing Kalman filters and extended Kalman filters (94), it is possible to deal with the dynamic aspects of mobility.

2. RELATED WORK

2.5.1.2 Auto-regression model

An auto-regressive model is proposed to capture the characteristics of realistic movements of nodes. The model applies the Yule-Walker technique (51) to the training data and the Kalman filter to predict the location, velocity and acceleration of a mobile node with a dedicated hardware (101) (102).

2.5.1.3 Information theoretic model

A node in this model maintains the history of base stations or nearby mobile individuals it encounters and applies a compression algorithm (104) to generate a dictionary of the recurrent observed paths (7).

2.5.1.4 Markov chain based model

A calibrated Markov chain is produced with states and active/inactive cycles representing the access points and the behavior of mobile nodes, respectively. State conditions describe obstacles or restrictions in the environment (9).

2.5.1.5 Pattern matching model

A node first searches for patterns similar to the current scenario in its stored history and the one with the highest cross-correlation (21) is selected as a base for the link prediction (31).

Farkas et al. (31) claim that the pattern matching based approaches are more accurate than linear approaches for the following reasons: (i) the patterns of movement are usually clear whatever means of transport is used; (ii) the patterns are reflected in the signal space; (iii) the patterns convey more information such as the effects of the environment and the recurrent displacements; (iv) a priori knowledge of the geographic environment of the network is not needed.

2.5.2 Data Source for Mobility Estimation

The scope and usefulness of these techniques depends on the way the raw data are acquired and processed. In fact, multiple sources can be utilized as the raw data for mobility estimation. Here, only some of these approaches are briefly discussed.

2.5.2.1 Global Positioning System (GPS)

GPS gives the absolute coordinates of a mobile node, but it is expensive and energy consuming (57). It also suffers from frequent satellite disconnections in indoor environments (65).

2.5.2.2 Pedometers

A pedometer is a portable and electronic/electromechanical device that counts each step a person takes by detecting the motion of the person's hips. Algorithms for navigating a mobile node by using the hop-count based metric is simple and scalable (50). This method, however, is highly dependent on the network density and path length, and thus is coarse-grained and error-prone (64).

2.5.2.3 Robotics

A robot can localize itself in both mapped and unmapped terrains by employing the method which represents the posterior distribution of possible locations via a set of weighted samples. New measurements such as observations of new landmarks are incorporated to filter the previous mobility prediction and update the data of location (39). However, such estimation suffers from rotational and translational errors (76), even if a map of the environment and sensory information is given.

2.5.2.4 Radio Frequency Identification (RFID)

RFID is a technology that employs radio frequency signals to exchange data between a reader and an electronic tag attached to an object for the purpose of identification and tracking. RFID readers are located strategically in the field (63). One of its drawbacks is the relative short communication range ($1-2m$) and the inhibition to future extensions.

2.5.2.5 Anchor node

Anchors are a set of static nodes with globally known or unknown positions. In the literature, they are also referred to as reference nodes or seeds (39). Anchor nodes periodically broadcast beacon messages. By receiving beacons from enough sources, mobile nodes can localize themselves. In some cases, robots equipped with GPS are deployed into a wireless sensor network to act as reference nodes, so that sensors can

2. RELATED WORK

localize themselves with the information given by the robots (78). The accuracy of the localization depends on the distance between the mobile and the reference points as well as the number of the anchor nodes (71). If the distance is too long or the anchor nodes are too less, the location estimation errors can be high. Moreover, the loss or malfunctioning of anchor nodes can affect the estimation mechanism (97).

2.5.2.6 Time-Of-Arrival (TOA)

TOA finds the distance between a transmitter and a receiver via a one way propagation time by exploiting the relationship between the light speed and the carrier frequency of a signal (2). However, all the nodes, with no information when messages will come, have to keep awake all the time.

2.5.2.7 Angle-Of-Arrival (AOA)

AOA is usually employed as prior-knowledge for the triangulation localization method (40). The information of the arriving angle can be obtained by using either goniometers, gyroscopes or compass.

2.5.2.8 Received Signal Strength Indicator (RSSI)

RSSI (30) represents the relationship between a transmission and a received powers. It can be employed to compute the distance of separation between a transmitter and a receiver when a good portion of the electromagnetic wave propagates in a Line-Of-Sight (LOS) link. It has been used in a number of mobility-aware MAC protocols. However, the measure of RSSI fluctuates significantly due to deleterious effects of fading and shadowing and thus provides a lower accurate result than GPS.

2.5.2.9 Signal-to-Noise Ratio (SNR)

Deriving connectivity information from position information is not straightforward, since it requires a one-to-one mapping between distance and signal quality. Hence, alternative to RSSI, SNR is utilized as a measure of a node's link state. It is easy to be monitored and does not require any special hardware (31).

2.5.2.10 Ultrasound

A mobile node with an ultrasonic sensor measures the distance to a node by exploiting the ultrasonic signal propagation time. However, the transmission range of an ultrasound signal is small as it cannot propagate further than radio frequency wave (50). It also adds size, cost, and energy supply to each device. Therefore even though ultrasound based localization approach can achieve high accuracy, it is not suitable for wireless sensor networks.

2.5.2.11 Accelerometers

Accelerations are generated due to both translational and rotational movements of an object. An accelerometer-based mechanism is shown to be an accurate, robust and practical method for objectively monitoring the free movement of objects and persons. The mechanism responds to both frequency and intensity of movement (97). However, these devices increase the cost and size of a node and may not always be available or deployable. Moreover, accelerometer readings are sensitive of the node placement (58).

2.5.2.12 Triangulation and trilateration

The localization of mobile nodes can also be accomplished through triangulation in a one-hop neighborhood (15). Once a local estimation is made for each node, a global localization can be established by calculating differences in terms of the distance and direction between each node and a particular central node, or a dense group of nodes (15). However, this mechanism requires the use of isotropic antennas, which is expensive and less practical.

A trilateration requires priori-knowledge of the location of at least three nodes. The distance between nodes can be determined only within a certain degree of certainty (40).

2.5.2.13 Hybrid

Instead of directly using one of the above techniques, one can make use of the combinations of two or more localization measurement schemes. For example, mobile nodes can be aware of the relative distance between anchor nodes and themselves through RSSI values (60), through beacons broadcasted by anchor nodes, or through hop-based

2. RELATED WORK

techniques (64). However, the latter approach does not perform well in non-uniform topology where the position estimation can be coarse. Likewise, by reading records from both an odometer and a compass (76), a node can determine its translation as well as orientation at each particular time.

A comparison of the localization techniques is summarised in Table 2.2.

2.6 Mobility-Aware MAC Protocols

Node mobility causes, among other things, the deterioration of existing communication links. Once such a deterioration in a link quality is detected, one of the following strategies can be adopted to maintain an end-to-end data transmission (87):

- The transmitting node continues with the data transmission if the remaining time is enough to complete the task.
- The transmitting and receiving nodes negotiate for a dynamic rate adaptation, so that data can be transmitted at a higher speed before the link breaks.
- The transmitter initiates a seamless handover without breaking the existing one to transfer transmission to a better link.

These decisions require the link layer to interact with the application, network and physical layers, since these layers provide information pertaining to the data size, the transmission rates supported by the transceivers, and the transmission power range that can be adapted. In the following subsections, a few mobility-aware MAC protocols which support strong node mobility are investigated.

2.6.1 MS-MAC

The mobility-aware MAC protocol for sensor networks (MS-MAC) (67) extends SMAC to support mobility. It introduces coordinated sleep/listen duty cycles and periodically synchronizes the schedule of nodes. The synchronization is done by broadcasting a SYNC packet at the beginning of the listen phase every predefined number of cycles (for example, 10 seconds every 2 minutes). A node first tries to follow the existing schedules by listening for a certain amount of time. If no SYNC packet is received, the node will randomly choose a time to go to sleep and immediately broadcasts this

2.6 Mobility-Aware MAC Protocols

Table 2.2: comparison among the data sources for mobility estimation

Localization Technique	Characteristics	Advantages	Disadvantages
GPS	measures absolute coordinate	precise; simple	high price; unavailable in enclosed space
Pedometers	employ hop-count approaches	simple; scalable	coarse-grained; error-prone
Robotics	employ the Monte Carlo method	localize in mapped and unmapped terrains	prone to rotational and translational errors
RFID	data are exchanged via radio waves	capable of identification and tracking	short communication range; passive; difficult for future extensions
Anchor node	with globally known or unknown positions	nodes can be accurately localized if anchors are enough	cumulative estimation errors; may be unavailable
TOA	employs a propagation time	no additional cost	error prone; energy inefficiency
AOA	employs Goniometers, gyroscopes or compass	prior-knowledge for the triangulation method	inaccurate; cannot be used alone
RSSI	measures relative distance	no additional cost	signal fading, shadowing and variation
SNR	reflects a node's current connectivity	can be monitored by off-the-shelf devices	no direct relationship between SNR and link quality
Ultrasound	employs a propagation time of ultrasonic signal	Accurate	the receivable range is limited; adds size and cost to devices
Accelerometers	responds to frequency and intensity of movement	accurate; robust; practical	adds cost and size; may not be available or deployable
Triangulation & trilateration	calculates global coordinate based on local coordinate	no additional cost	needs recalculation; a few nodes' locations should be prior known

2. RELATED WORK

information. However, if a node receives a different schedule after it selects one, it will adopt both schedules. Therefore, the border nodes in a virtual cluster may have two or more different schedules, where a virtual cluster is formed by nodes with the same schedule.

Even in the situation where mobility is detected inside a virtual cluster, SMAC can perform well, since nodes have no difficulties in communicating with each other by having the same schedules, even though the cluster topology is changed due to this **intra-cluster mobility** (43). However, if mobility is detected across virtual clusters, SMAC is unable to handle it, since before a connection with a new cluster can be set up, a node has to wait for a long time (as much as 2 minutes in our example) to receive the SYNC packet from the new cluster and updates its own schedule accordingly. In the meantime, it cannot communicate with its old neighbors once it moves out of the cluster range. The member and topology changes of a virtual cluster caused by the **inter-cluster mobility** leads to the disconnection of the mobile node from the network (43).

Figure 2.8: Synchronization frequency adjustment in MS-MAC

To expedite the connection set up process, MS-MAC enables each node to discover the presence as well as the level of mobility within its neighborhood, based on the RSSI values obtained from the SYNC messages transmitted by its neighbors. If the RSSI value from one and the same neighbor changes during a time interval, it realizes

that either this neighbor, the node itself, or both of them are moving, since a one-to-one mapping between the distance and the RSSI values is assumed. Depending on the change of the RSSI values, the relative moving speed of the mobile individual can be deduced. Based on this information, the node broadcasts a SYNC message containing its own schedule and additional mobility information (the maximum estimated speed in the neighborhood). Upon receiving this packet, all the neighbors create an active zone by adjusting the synchronization frequency if the node is to move from one virtual cluster to another. The synchronization frequency, however, depends on the maximum speed of the surrounding neighbors. As can be observed in Figure 2.8, the faster the speed is, the higher the synchronization frequency will be. Therefore, by making the mobile node awake as much as possible, the connection with a new virtual cluster will be quickly set up before the node losses all its neighbors.

However, if a node does not detect any change in the coming RSSI values or if the node is not a border node, the mobility information in the SYNC message is set to be empty and thus no active zone will be created.

2.6.2 M-MAC

The mobility-adaptive, collision-free medium access control protocol (M-MAC) (4) is a schedule-based MAC protocol following the design principle of TRAMA (73). Instead of the fixed frame length in TRAMA, M-MAC introduces a flexible frame time that enables the protocol to dynamically adapt to mobility, making it suitable for wireless sensor environments, as Figure 2.9 describes.

Figure 2.9: Dynamic frame time in M-MAC

In M-MAC, time is divided into rounds and each round is composed of k frames (k is an integer larger than 1). At the beginning of each frame, all the nodes in the network predict their mobility states at several different time points of the next frame based on

2. RELATED WORK

the $AR - 1$ mobility estimation model (102). The average of these location estimations is regarded as a node's location prediction for the next frame. This information is transmitted to the node's corresponding cluster head. Since the cluster head never goes into sleep, it is able to collect the values of all its members and broadcasts them in the last slot of a frame. This ensures that all the nodes in the cluster have the best knowledge of the predicted mobility states of its current and potential two-hop neighbors. A node calculates the relative distance between the center node (node B in Figure 2.10) and itself, in order to learn whether it will enter or leave the cluster in the next frame (the cluster is the collection of the 2-hop neighbors of node B). For example, the big circle in Figure 2.10 represents the cluster range. In frame i , nodes $A1$, $A2$ and $A3$ are initially within B 's two-hop neighbor range. However, both their absolute and relative locations change with respect to B 's position in frame $i + 1$, since $A1$ and $A2$ move out of B 's two-hop range. In contrast, $A3$ still lies beside B due to its similar movement speed and direction with B . A new node $A4$ moves into B 's view in the next frame.

Figure 2.10: Mobility estimation for the next frame in M-MAC

According to this comparison, a node independently proposes a new frame duration and transmits it to the cluster head. The head, by averaging the duration estimations from all the members, produces the mean frame size and broadcasts it to all the nodes. If this value is less than the previous one stored at a node, it increases the random access interval and decreases the scheduled access interval while keeping the frame time constant. The frame time, however, can only be changed at the end of a round by employing Global Synchronization Period (GSP). The GSP averages the predicated

frame durations from all the cluster heads and disseminates this mean value in the network as the frame size for the next round.

2.6.3 M-TDMA

The mobility-aware TDMA-based MAC protocol for mobile sensor networks (M-TDMA) (43) has been proposed to extend the TDMA mechanism for adapting to the changes in a network topology. Unlike a pure TDMA, M-TDMA partitions the network into non-overlapping clusters using the FLOC algorithm (23), with each cluster having its own head. Each node within a cluster is assigned a unique slot. To deal with mobility, some of these slots are shared across clusters and some of them are kept free for future allocation. To this end, M-TDMA splits a given round into two parts, namely, the control part and the data part. The control part is used to adapt to mobility, whereas nodes transmit packets in the data part. Some of the slots at the end of the data part are reserved for the future entering nodes as well as the message retransmissions.

As shown in Figure 2.11, the control part is composed of the first three slots. In the first slot, the head broadcasts cluster information, such as the ID, the head status, the cluster schedule and the round number. If a node receives this information, it knows that it is still in the original cluster, and therefore, it only updates its state in the second slot. If the node does not receive any message, it notices that it has left the original cluster but has not entered into any new cluster. In this case, it has to wait until the cluster information is received in a future round. But if the node receives cluster information from a different head, it then learns that it has joined a new cluster and informs its presence in the second time slot.

Figure 2.11: Working principle of M-TDMA

On receiving a new ID, the head checks whether any unassigned slot is left in the data part. If more than one is free, the head directly assigns one to the new node in the

2. RELATED WORK

third slot and updates the cluster schedule. However, if only one slot is free, instead of fully allocating the slot to the new node, the head halves the bandwidth by doubling the period in which the new node transmits, as the other half is kept for further entering nodes. In this case, the head updates the schedule by maintaining a sequence of IDs, with the last element serving as a place holder. If no slot is left, the head checks the place holder and further halves the remaining bandwidth.

The head, however, may not receive any information in the second slot when two or more nodes simultaneously announce their appearance in the second slot or no node joins in the cluster. Similarly, a new node may not receive the slot assignment in the third slot if the cluster head is out of range or its packet had collision with other packets in the second slot. In both cases, the node has to randomly back-off and retransmits its ID in the next round. In the data part, nodes transmit and receive based on a normal TDMA mechanism.

2.6.4 MobiSense

MobiSense (35) is a cross layer architecture that combines the MAC and routing layers to achieve energy efficient communication in a micro-mobility scenario. Here nodes are organized into clusters, in which strategically placed static nodes act as cluster heads and mobile nodes move between them. To increase the network throughput and to simplify the network management, MobiSense adopts multi-channel communications by requiring adjacent cluster-heads to operate in different channels. The aim is to reduce the interference between the clusters and to allow the cluster-heads to dynamically schedule traffic locally.

MobiSense organizes a super-frame into a synchronization slot, downlink and uplink data transmission slots, admission mini-slots, and discovery slots. All the cluster heads send synchronization packets at the beginning of each frame to inform mobile nodes about the changes in their uplink and downlink transmission. Each cluster head sends a probe in the discovery slots on a common channel to advertise the information about the channel it uses for communication, the current cluster size, and the timing of its access window. As a result, by listening to these slots, mobile nodes that want to join the network or handover from one cluster to another can rapidly gather network information and build the prioritized lists of access points. Since the size of the discovery slots is fixed, nodes only listen for a fixed duration. After determining the new cluster, a mobile

node randomly selects an access mini-slot and sends its own join request message. Since the collision probability is low under moderate contention scenarios, MobiSense achieves a low admission delay and a fast network convergence. The downlink and uplink slots are used for two-way data communications. The working mechanism of MobiSense is explained in Figure 2.12.

Figure 2.12: Working mechanism of MobiSense

2.6.5 MCMAC

The mobile cluster MAC (MCMAC) (61) is a schedule-based MAC protocol which extends LMAC (37) and GMAC (66) to support cluster mobility. Unlike most of the proposed mobility-aware MAC protocol, MCMAC is optimized for those nodes which travel in group. This is particularly the case in Body Area Networks, such as in health-care applications, where a number of biomedical sensors are travelling together, being attached to the body of a patient.

MCMAC categorizes the sensor nodes into a static network and a mobile cluster. The protocol defines a Reference Point Group Mobility (RPGM) model and a Random Waypoint Mobility (RWM) model to mimic the movement characteristics of mobile clusters and the individual node movement within a cluster. A frame in MCMAC is divided into an active and a sleep period. Since the slot assignment method is different for static and mobile nodes, the active period is further divided into Static Active Slots (SAS) and Mobile Cluster Slots (MCS). Static nodes communicate with each other in the SAS part by dynamically occupying a unique transmission slot in its two-hop neighborhood. A static node can only transmit data in the specific slot it chooses and receives data in the remaining part of SAS. A guard time is inserted at the start and the end of every transmission slot to compute a node's phase difference with its direct

2. RELATED WORK

Figure 2.13: Architecture of MCMAC

neighbors for synchronization. The working principle of static nodes is described in Figure 2.13 (a).

The MCS part is used for nodes in a mobile cluster to communicate with each other. Since the size of a cluster can be small (a human body) and all enclosed nodes are typically within each other's one- or two-hop neighborhood, each slot in this part is assigned to exactly one node. In case of multiple clusters, a transmission slot is extended to include a CSMA period to avoid collision between two mobile nodes in two different clusters which share the same transmission slot. A mobile node selects a random time in CSMA period to sense the medium before data transmission, as Figure 2.13 (b) illustrates. Static and mobile nodes also listen to the MCS and SAS parts, respectively, to enable a communication between them.

2.6.6 Discussion and Comparison

Here, the merits and demerits of these mobility-aware MAC protocols will be discussed and compared in detail. For MS-MAC protocol, one obvious advantage is that a mobile

2.6 Mobility-Aware MAC Protocols

Table 2.3: comparison among the mobility-aware MAC protocols

MAC	Characteristics	Advantages	Disadvantages
MS-MAC	<ol style="list-style-type: none"> 1. contention-based 2. runs synchronization frequently 	<ol style="list-style-type: none"> 1. communicating while setting up a new link 2. synchronization frequency adapts to node moving speed 	<ol style="list-style-type: none"> 1. trades-off a higher energy cost for a less delay 2. a neighbor of the mobile node consumes high energy
M-MAC	<ol style="list-style-type: none"> 1. schedule-based 2. adjusts frame size and proportion within a frame 	<ol style="list-style-type: none"> 1. time slot is allocated by changing a frame's size 2. frame proportion is changed more frequently than frame size 	<ol style="list-style-type: none"> 1. computational intensive 2. accuracy depends on AR-1 model and historical statistics
M-TDMA	<ol style="list-style-type: none"> 1. schedule-based 2. uses control part to get mobility information and data part to allocate time slots 	<ol style="list-style-type: none"> 1. guarantees collision-freedom 2. does not rely on any localization algorithm 3. mobility is adapted without changing frame size 	<ol style="list-style-type: none"> 1. several assumptions are made 2. disconnection with the network may occur 3. energy and latency are increased
MobiSense	<ol style="list-style-type: none"> 1. schedule-based 2. uses mini-slots, discovery slots and multi-channel communication to handle mobility 	<ol style="list-style-type: none"> 1. rapid network information gathering 2. local scheduling 3. rapid admission and fast network convergence 	<ol style="list-style-type: none"> 1. multi-channel requires media resource management 2. order of discovery and mini slots is critical 3. mobility cannot be handled in time
MCMAC	<ol style="list-style-type: none"> 1. schedule-based 2. different slot assignment scheme is used for static and mobile nodes 	<ol style="list-style-type: none"> 1. decentralized frame synchronization 2. avoids adaptation time once cluster moves 	<ol style="list-style-type: none"> 1. hidden terminal problem 3. single channel limits the bandwidth and makes the throughput unscalable

2. RELATED WORK

node can keep transmitting and receiving data packets with its original neighbors while establishing connection with the nodes in a new virtual cluster. However, frequent synchronization can lead to high energy consumption. And such energy consumption is not limited to the mobile node itself, but also to all the neighbors which are two-hops away from it. Therefore, the neighbors may deplete their energy quickly even if they are stationary. From this point of view, the handover mechanism of MS-MAC is actually implemented by trading-off a higher energy expenditure of the neighbors around a mobile node for a lower latency in setting up a connection with a new virtual cluster.

M-MAC deals with mobility-related delays by adjusting the size of a frame and the proportion of the scheduled access interval to the random access interval within a frame. However, it has several inherent limitations. Being computationally intensive is one of the limitations of M-MAC. The accuracy of the mobility handling mechanism depends on the accuracy of the predicted size of the next frame, which is estimated by all individual mobile nodes in the current round. This estimation, however, is made according to the change of the relative distance between nodes obtained by employing the $AR - 1$ location evaluation model. Furthermore, the adjustment of the size of both the scheduled access interval and the random access interval for the next frame as well as the size of the frame for the next round is only managed according to the historical statistics. Therefore, the node movements, which still occur in the current frame and round, are not taken into consideration after mobility is predicted, leading to latency and inaccuracy of mobility estimation.

Unlike MS-MAC which aims at decreasing the connection time by creating an active zone around a mobile node, M-TDMA targets at the absence of collision while supporting mobility. Unlike M-MAC which introduces the mobility adaptive frame by predicting the future position of all the nodes, M-TDMA does not rely on any localization service and the frame size is never changed. Instead, M-TDMA divides a frame into a control part to learn about mobility and a data part to allocate time slots to the newly entering nodes. However, M-TDMA has its own drawbacks, not least of which is its making several assumptions. To start with, it assumes that any individual node remains within a cluster for at least one round unless it crashes. This, however, depends on the moving direction and speed a mobile node chooses. Secondly, it assumes that a node may not stay for more than one period without hearing from a head. This actually

related with the state of the head and the network topology. Thirdly, it assumes that heads would not collide for more than two consecutive rounds, even though it depends on the network partitioning algorithm and network density. In addition to these assumptions, M-TDMA introduces a big latency, since mobility can only be detected in the second slot and thus a node which enters the cluster after this time has to wait until the next round comes. This implies that during this time, the node is disconnected from the entire network. Furthermore, a new node may not receive the slot allocation in the third slot, because either a collision occurs among the new nodes in the second slot; or a collision occurs among the heads in the third slot. In both cases, the new node has to re-announce its presence in the future round, leading to an increased energy consumption and latency.

MobiSense proposes a multi-channel medium access mechanism to provide reliable data transfer and fast handover as well as dynamic and distributed transmission scheduling in micro-mobility environments. By enabling clusters to operate in different channels, interference between them is reduced and traffic demands can be adapted to the need of individual mobile nodes. In addition, it allows some of the clusters to multiplex data to the sink while others serve their mobile nodes. The use of the discovery slots on a predetermined common channel allows mobile nodes to rapidly gather network information, prevents unsuccessful join attempts, and maintains a constant network discovery delay. Moreover, the well-defined access mini-slots avoid collision between data transmissions and join requests and enables distributed network formation and faster network convergence.

Along with some of its merits, MobiSense has some demerits as well. Firstly, it assumes the presence of some fixed static nodes. Secondly, the use of multi-channel is expensive, since it requires careful management of the channel resource and a more advanced receiver design. Thirdly, the order of the discovery slots and the access mini-slots is critical for handover, since a mobile node cannot send a join request before knowing the cluster information in the discovery slots. Fourthly, mobility cannot be handled instantaneously, since nodes can only handover during the access mini-slots. Even worse, the handover request may not be processed in case of a collision.

MCMAC supports continuous cluster mobility. Instead of using a synchronization slot, it inserts guard times in a node's transmission slot to indicate the phase difference to its direct neighbors. This mechanism ensures decentralized frame synchronization.

2. RELATED WORK

MCMAC also enables a node to dynamically select its transmission slot based on its neighbors' occupancy bit-sets. In addition, the different active slot assignment for static and mobile nodes avoids the need for an adaptation time after a cluster movement. Several static nodes can share one and the same transmission slot if the distance between any two of them is larger than two hops. This improves the channel capacity. The size of the SAS part, therefore, depends on the network density. Unlike the SAS part, each slot in the MCS part is allocated to exactly one mobile node in a cluster to avoid interference between mobile nodes, since a cluster can be small and mobile nodes are usually within each other's one- or two-hop distance. The MCS part, however, can also be shared in case of multiple mobile clusters in the network, by using a hybrid contention-based and schedule-based channel access mechanism. A simple Carrier Sense Multiple Access (CSMA) period is adopted for sensing the channel status before the real data transmission. As a result, the size of the MCS part is equal to the maximum number of nodes in the biggest cluster.

Even though the CSMA mechanism is introduced, collision cannot be completely avoided, since there is a switching delay between the carrier sensing mode and a data transmission. Therefore, it may happen that a node senses a free channel while another competing node is preparing for data transmission. Collision can also happen due to the hidden terminal problem.

A comparison summary of the five mobility-aware MAC protocols is given in Table 2.3.

2.6.7 Lessons Learned and Open Issues

Wireless sensor networks that support mobile nodes have a large number of applications in areas such as healthcare, supply-chain, toxic gas detection in disaster areas, etc. Due to the criticality of the applications, the way mobility should be captured and handled requires a careful planning. Most of the proposed mobility-aware MAC protocols investigated earlier assume a mixed deployment in which both static and mobile nodes are present. In fact, most implicitly assume that the number of static nodes is significantly larger than the number of the mobile nodes. We believe that this is a realistic assumption.

Experiment and simulation results indicate that accurate mobility estimation is essential to avoid unnecessary oscillation in link establishment (or node associations

in cluster-based networks) (4, 87). However, the results also indicate the existence of a strong trade-off between estimation accuracy, estimation time, and signal processing cost (both in terms of energy consumption and computing resources). Therefore, mobility estimation techniques that optimize this trade-off are required. Moreover, estimation techniques that are based only on RSSI values perform poorly, leading to frequent oscillations even when nodes are not mobile. Unfortunately, most of the proposed mobility handling approaches rely on this technique. The limited size of a sensor node and its limited resources put significant constraints on the type of data sources that can be used for mobility estimation.

We believe that the mobility patterns can help to improve estimation accuracy and reduce the cost of mobility estimation. For example, knowledge of the pattern of mobility of the object or person can significantly reduce both false positives and false negatives. We realize that most of the proposed MAC protocols do not fully take advantage of this knowledge. Finally, work still remains to quantify the cost of mobility. Strong mobility requires that neighbor nodes (static nodes) should remain awake when a request to join comes from a mobile node. To maintain an unbroken link, research questions such as (1) how many of these nodes should remain awake? (2) for how long should nodes remain awake? and (3) how different should the schedules of these nodes be? depend on the mobility patterns, the energy reserve of the static nodes, the acceptable end-to-end delay in data communication, the duty cycles and the network density. These issues have to be carefully studied and their significance with respect to network lifetime should be quantified.

2. RELATED WORK

3

Selection of a Hybrid MAC Protocol

3.1 Reason for the Selection of RIMAC

The existing or proposed MAC protocols in the literature can be broadly categorized into schedule- and contention-based protocols. In schedule-based approaches such as TDMA MAC (34), precise synchronization is required to allocate time slots to each node. The difficulty of aligning and reallocating time slots in these protocols amounts to inefficient channel utilization as well as high energy consumption even in static sensor networks, not let alone in mobile scenarios. In addition, mobility cannot be detected and handled in time, since mobility handling mechanism can only be embedded in one or some of the time slots rather than all. Therefore, it has a high probability that a mobile node has to wait for a certain amount of time before it can be fully integrated in a network.

The contention-based MAC protocols can be further divided into two categories: synchronous and asynchronous. In many sensor network applications, nodes are in idle state for long period of time if no sensing event is detected. Since the sampling rate during this time is very low, it is a waste to keep nodes listening all the time. To save energy, synchronous protocols such as DW-MAC (83) and RMAC (26) define a low duty cycle and enable nodes to synchronize sleeping schedules with their neighbors. Radios are turned on and off at the same time for all the nodes and data are only communicated in the listen period. Even though the synchronous MAC protocols greatly

3. SELECTION OF A HYBRID MAC PROTOCOL

reduce the time of node idle listening, the required schedule synchronization leads to extra communication overhead and design complexity. This is because a node should not only broadcast its own synchronization packet periodically, but also align its schedule according to the schedules of its neighbors. As a side effect, a node may wake up multiple times if its neighbors have different schedules.

The asynchronous approaches, on the other hand, are mostly preamble initiated, such as XMAC (10) and BMAC (70). These protocols avoid the need for synchronization by enabling nodes to define their own sleeping schedules while the duty cycles are kept constant. Communication is operated by preceding data transmission with preamble packets. However, if preamble packets are set too long, a receiver has to wait for a long while before it can receive a data packet or send an early ACK frame. This makes energy inefficient at both the receiver and the transmitter. In addition, the long preamble packets can cause the overhearing problem, since the non-target receivers have to also stay awake during the whole preamble time. Even though they finally find that the data packet is not destined for them. This results in a big transmission delay and low channel utilization.

Receiver-initiated MAC protocols are asynchronous MAC protocols, but do not employ preamble packets. For example, in RI-MAC (86), a sender remains active and waits silently until a receiver explicitly signifies the start time for data transmission. The signifier is implemented by broadcasting a short beacon. As only beacon and data packets are transmitted in RI-MAC, the occupancy of medium can be significantly decreased. This makes room for other nodes to exchange data. Due to the receiver-initiated design, RI-MAC substantially reduces overhearing, achieves low collision probability, and minimizes the data recovery cost.

3.2 Description of RI-MAC

Figure 3.1 gives an overview of the operation principle of RI-MAC. Each receiver will broadcast a base beacon as soon as it completes a sleep phase and detects a free channel. The base beacon contains only the source address information (SRC) which announces that the node is ready to receive data packets. All the neighbors of the receiver with pending data will start transmission upon receiving the base beacon. However, prior to the actual data transmission, a sender is required to use Clear Channel Assessment

(CCA) check to make sure that the medium has been idle for a period of time. This time duration is set as a Short Interface Space (SIFS) interval plus the maximum propagation delay. The CCA check prevents a sender from starting data transmission while the intended receiver is generating a feedback message to a data packet just received from another sender.

Figure 3.1: Working principle of RI-MAC

If two or more packets are transmitted simultaneously, collision will occur. The receiver can detect collision by checking the Start of Frame Delimiter (SFD) of an incoming data packet. If no SFD is found in time, but some channel activity is detected by the CCA check, the receiver will learn that a collision has occurred and will generate another beacon. This new beacon is transmitted after a random backoff time in order to avoid possible collisions with beacons sent by other nodes. In addition to the SRC information, a back-off window field (BW) is included in the beacon specifying the backoff window size senders should use when they contend for the channel in the next competition. The node that has chosen the smallest backoff value is the medium winner and can start transmission once the backoff timer fires. By overhearing a data packet, all the other senders will immediately give up their transmission attempt. In this way, the BW field in RI-MAC reduces the collision probability among potential transmitters.

However, if a data packet wins the channel, the receiver will reply to the transmitter with an ACK beacon (DST) serving as the acknowledgement. The transmission of the ACK beacon starts after a SIFS delay, regardless of the medium status. A node that receives a beacon can determine which fields are present in the beacon by looking at the size of it. Nodes other than the sender will ignore the DST information in a beacon and treat it as a request for the initiation of new data transmission. Figure 3.2 illustrates the beacon format in RI-MAC. A beacon should always contain a SRC field regardless

3. SELECTION OF A HYBRID MAC PROTOCOL

of the role it serves. Not alike, BW and DST are optional fields in beacons, described by the dashed rectangles in the figure. The frame length, Frame Control Field (FCF), and Frame Check Sequence (FCS) are the fields directly taken from the IEEE 802.15.4 standard.

Figure 3.2: The format of a RI-MAC beacon for an IEEE 802.15.4 radio

After successfully receiving a data packet, the receiver remains awake for at most a dwell time in order to allow queued packets to be sent to it. The duration of the dwell time is defined as the summation of the BW value in the last beacon, the interval of SIFS, and the maximum propagation delay, as described in Figure 3.3. If no packet is received during this time, the receiver will go to sleep immediately. Since the BW value in a beacon is automatically adjusted according to the collision frequency, so is the dwell time.

Figure 3.3: The design of dwell time in RI-MAC

3.3 Merit and Demerit of RI-MAC

RI-MAC significantly reduces the amount of time a pair of nodes occupy the medium before they reach a rendezvous time for data communication. This short medium oc-

cupation time enables more contending nodes to exchange data packets with their intended receivers, which helps to increase the network capacity, the data throughput and the channel utilization efficiency. More importantly, this increase is adaptive, by letting a beacon serve both as an acknowledgment to the previously received data packet and as an invitation for the new data transmission. The self-adaptation of dwell time also enables RI-MAC to use the shortest waiting time under light channel contention while avoiding collisions under heavy channel contention. Additionally, the well developed collision detection and lost packet recovery scheme allow RI-MAC to achieve high packet delivery ratio under a wide range of traffic loads.

Even though RI-MAC performs well in many aspects, it has some demerits. Firstly, in a round of communication that begins with a base beacon transmission and ends with the expiration of a dwell time, the BW size in beacons never decreases. Instead, it either remains unchanged if no collision occurs or keeps increasing otherwise, as described in Figure 3.1. The increase can be fast in case of small transmission intervals, since the collision probability will quickly grow when more data packets are transmitted simultaneously. As time goes by, the backoff window size will become larger, and thus, a sender has to remain idle for a long period of time before it can transmit a data packet. The BW field in beacons can be rather large even under light traffic load, if the packet transmission rate is high initially but decreases afterwards. Since the waiting time does not adapt to the data arrival rate, RI-MAC does not work efficiently in terms of energy and latency.

Secondly, using only the dwell time for a receiver to judge whether there are nodes preparing to communicate with it is not enough. A receiver receives no data packet during a dwell time cannot necessarily indicate the empty queue of its surrounding neighbors. Instead of an idle channel, the reason that a receiver does not receive any data packet during a dwell time can be because of an unsuccessfully transmitted beacon. Due to the link termination, the packet collision, or the third-party interception, neighbor nodes with pending data may not receive a beacon during a dwell time, even though the receiver has already broadcasted one. Consequently, it may happen that a receiver has to go to sleep after the dwell time expires even though there are transmitters wishing to communicate with it.

3. SELECTION OF A HYBRID MAC PROTOCOL

3.4 Optimization of RI-MAC

In order to address the monotonous increment in the backoff window size and to reduce the occurrence probability of the dwell time, a burst transmission pattern of packets can be adopted. After a node wins the medium, instead of sending a data packet each time, it transmits a pre-defined number of packets in burst. In mobile wireless sensor networks, there are several applications require the transfer of a large quantity of data in a short period of time. This is because applications in mobile scenarios are usually delay sensitive. Once a node observes an increase in its queue backlog, it will perform low-overhead coordination with one of its neighbors and reserve the medium for a period during which data packets are transmitted in burst. Since the burst transmission pattern alleviates the channel contention during data communication, the throughput from the sources to the sink can be boosted and the end-to-end latency can be highly reduced (52).

Figure 3.4: Optimization on top of RI-MAC

The burst transmission will be justified only if a node has sufficient data packets in the queue before it contends for the medium. Under the burst transmission pattern, a node begins the data communication with the broadcast of a base beacon. Since data packets are transmitted in a burst manner, the beacon between every two neighbor packets is unnecessary to include the SRC field. Hence, it is only set as the ACK frame, as shown in Figure 3.4. Every time a data packet is transmitted, the sender waits for an ACK beacon from the receiver. If it fails in receiving, it will immediately perform retransmission by extending the reserved transmission time for one more data packet. Other neighbors of the transmitter, after intercepting the ACK beacons, will realize that the medium is currently occupied, and therefore, will not compete for the channel

in the midst of data communication. In this way, collision can be avoided and the size of the BW in beacons can be considerably reduced. Since both the occurrence probability and the duration of the dwell time are directly proportional to the collision frequency, a slow increment in the BW size of beacons will lead to a short and infrequent dwell time. Even though such improvement is achieved at the cost of unfairness which however is not the prime concern in wireless sensor networks (99).

3. SELECTION OF A HYBRID MAC PROTOCOL

4

Effect of Mobility on Latency

4.1 Problems of Burst Transmission due to Mobility

The burst transmission pattern works well on the optimized RI-MAC in static applications, since it highly reduces the unnecessary long idle listening time of nodes and therefore, achieves low data transmission latency. However, in case of mobility, the burst transmission may not perform well.

Once a transmitter receives a base beacon from a receiver, it will occupy the medium by transmitting an in-queue data packet. If either or both of the nodes are mobile, the link that has already been established between them may break in the middle of the communication. Since a large number of data packets are sent in a burst manner, the duration a node occupies the medium each time becomes longer. This results in a higher probability that a node moves out of the radio transmission range of the other node during their data communication. Consequently, in a mobile scenario, the probability of link disconnection under the burst transmission is larger than that under the situation in which only one single packet is transmitted at a time.

The deterioration of the quality of an established link makes the data transmission prone to failure. This is because the specific packet, either a beacon or a data packet, transmitted at the moment of link disconnection can be lost. Therefore, extra time is needed for the retransmission of the failed packet, leading to an increase in the data retransmission cost. After the link breaks, to resume the remaining data transfer, the transmitter will attempt to establish a new link. However, it may not be able to set up a new connection immediately after the original one disrupts, because it should discover

4. EFFECT OF MOBILITY ON LATENCY

the presence of at least one of its neighbors first. Since the asynchronous duty cycles are applied in RI-MAC, the sleeping schedules of nodes can be different from each other. Usually, a node has to wait for a long period of time to receive a beacon before it can newly contend for the medium. This may cause a considerable packet delivery delay.

4.2 Communication Pattern before Link Breaks

The link termination caused by node mobility under the burst transmission pattern is illustrated in Figure 4.1. Once a receiver, $R1$, wakes up and detects a free medium, it will broadcast a base beacon. To explain the general situation, we assume the network density is larger than one. That is, a node in a wireless sensor network has at least two neighbors within its maximum radio transmission range. The neighbors of the receiver, that have already woken up from a sleep period and have accumulated data packets in the queue, will receive the base beacon and send a corresponding data packet. Since no SRC information is included in the base beacon, collision must occur on the simultaneously transmitted data packets. This leads to a new beacon transmission from $R1$ in which a BW field is additionally embedded.

As soon as the neighbor transmitters of $R1$ receive this beacon, they will attempt to contend for the medium by randomly choosing a backoff value according to the BW information. The one which has selected the smallest value will win the medium and start with its packet transmission in a burst way. Suppose the winner node is mobile and moves in a direction that increases the distance between it and the corresponding receiver. If the mobile transmitter is very close to the maximum radio transmission range of the receiver when it successfully seizes the medium, the established communication link will have a high probability to break. Therefore, the mobile transmitter may not be able to complete transmitting all its data packets that are required to be sent in burst at a time. The communication link will disrupt when it finally moves out of the radio transmission range of the receiver. In Figure 4.1, $T3$ indicates the link termination time.

At the time of the link disconnection, either the transmitter fails in receiving an ACK beacon, or the receiver fails in receiving a data packet. However, the data packet has to be retransmitted in any case.

Figure 4.1: The communication pattern before the link breaks

4.3 Communication Pattern after Link Breaks

As soon as the link breaks at $T3$, the transmitter that has occupied the medium during $t2$, has to look for a new link. The medium winner in the original link, $S1$, can only resume its remaining data transmission until it receives a beacon from one of its surrounding receivers. The link set up time can be very different depending on when the transmitter is able to win the channel. If at least one extra receiver of $S1$ is awake before or at the moment of the link disruption, a base beacon will be broadcasted. Therefore, the transmitter can at once receive the beacon at $T3$. This enables $S1$ to establish a new connection with a minimum delay. However, the time required for setting up a new link can also be quite long, since the transmitter may not immediately receive a beacon from one of its neighbors due to the asynchronous duty cycles that nodes apply in the RI-MAC protocol.

Even though the original transmitter, $S1$, receives a beacon from a new receiver, $R2$, as soon as the link breaks, extra latency can be introduced. This is because multiple neighbors of $R2$ are medium competitors and $S1$ is only one of them. Since at least two nodes are assumed to be actively present around a particular node, a collision must occur on the data packets transmitted by these nodes in the first medium competition. This makes a BW field to be additionally included in the next beacon. According to the backoff value each transmitter selects, there are three possible transmission patterns in the second medium competition. It may happen that:

- (a) the original transmitter wins the channel;

4. EFFECT OF MOBILITY ON LATENCY

(a) S1 wins the medium

(b) One of the other transmitters wins the medium

(c) No transmitter wins the medium due to a collision

Figure 4.2: The establishment of a new link

4.3 Communication Pattern after Link Breaks

- (b) one of the other transmitters wins the channel;
- (c) no one wins the channel due to a collision.

If the original transmitter again wins the channel in the second competition, it will immediately proceed with its remaining data transmission. Therefore, case (a) introduces the minimum duration for the original transmitter to seize the medium. The sign for the accomplishment of a new link establishment is that the last data packet required to be transmitted in burst has been sent. From this perspective, case (a) also provides the minimum time for transmitting all the bursty data packets, and thus, for setting up a new connection.

In contrast, case (b) introduces the maximum time for a new link establishment. This is because instead of the original transmitter, one of the other active transmitters occupies the medium in the second competition. Therefore, $S1$ can do nothing but wait until the winner node first finishes its own data transmission. Afterwards, all the transmitters will contend for the channel again. $S1$ can only start to send data packets if it wins the channel this time. With the increment of the network density, the number of neighbors of a particular node increases. This makes the probability that the original transmitter wins the medium at each competition round smaller, leading to a longer period of time for setting up a new link.

In terms of the latency for establishing a new link, case (c) lies in between cases (a) and (b). Under this communication pattern, no one wins the channel. Instead, a collision occurs on the data packets, since at least two active transmitters select the same backoff value. Consequently, to resume the remaining data transmission, the original transmitter has to attempt to seize the medium in the following channel contention. All the three communication patterns for the new link establishment is shown in Figure 7.1.

4. EFFECT OF MOBILITY ON LATENCY

5

Handover Mechanism

In order to reduce the extra time caused by the link termination and the following new link establishment, a handover mechanism is proposed. Handover is defined as the process of transferring data communication from one sensor node to another without breaking the original link (46). It is necessary to be triggered only when there is a potential for an existing link between two nodes to break in the middle of data exchange. This, in turn, requires that the data communication should be comparatively long and the probability of link disruption should be reasonably high. Therefore, to justify handover, a node, that has woken up from a sleep period and has occupied the medium, should transmit a number of the in-queue data packets in burst. Since RI-MAC in its optimized condition meets both requirements, a handover mechanism can be developed and applied on top of it.

5.1 Working Mechanism

Before applying the handover mechanism to transfer the data communication from one sensor node to another, we need to figure out whether and when the link established by two nodes will break. In our approach, a pre-defined distance threshold is used to begin a handover. A transmitter can determine the distance with respect to its receiver by examining the received ACK beacons. The distance can be evaluated via multiple location estimation techniques (such as RSSI, GPS and accelerometer based approaches) in the outdoor environment.¹

¹This thesis assumes that a relationship between the link quality and the relative distance between two nodes can be established, but it does not examine how this relationship can be accurately

Figure 5.1: Distance traveled during the transmission of the $(n - w)$ data packets

Assume a node transmits n data packets in burst in the optimized RI-MAC once it seizes the medium. If the first w data packets are used for the distance estimation, then $(n - w)$ data packets are left to be transmitted after the distance is evaluated. The value of w can be variable when different types of the distance determination techniques are utilized by the handover mechanism. The more reliable the distance determination technique is, the smaller the value of w will be. If an established link between a transmitter and a receiver does not break during the communication of the $(n - w)$ data packets, the time calls for sending them can be easily quantified. Given a specific movement pattern, the corresponding distance a mobile node travels during this time can be assessed. It is denoted by $d_{(n-w)}$ in Figure 5.1. The distance threshold, d , is defined by excluding $d_{(n-w)}$ from the maximum radio transmission range of a node, R , under the situation that the last data packet is transmitted at the moment of the link disruption. During the communication between two nodes, once the transmitter estimates that the distance with respect to its receiver is larger than d based on the first w data packets, it will learn that it cannot complete transmitting the remaining $(n - w)$ established.

data packets before the link breaks. After learning it, the transmitter will immediately initiate a handover to search for a new relay node but without interrupting the data communication with the current receiver.

Figure 5.2 summarizes a handover process. As soon as a transmitter, $S1$, wins the channel, it will begin to transmit a data packet. After the intended receiver, $R1$, receives it, it will send back an ACK beacon. Based on the first w ACK beacons received from $R1$, $S1$ can deduce the quality of the established link and evaluate the current distance to $R1$. If the distance is estimated to be larger than d , $S1$ will trigger the handover mechanism. This is implemented by letting $S1$ send a data packet in a broadcast mode. This data packet includes a Boolean value indicating it is a handover request.

Figure 5.2: Working principle of the handover mechanism

As soon as $R1$ receives the handover request, it will realize that $S1$ is searching for a new receiver and the link between it and $S1$ is deteriorating. Since all the neighbors of $S1$ can receive the handover request, they can evaluate the relative distance between $S1$ and themselves based on a location determination technique. Being as the original receiver, $R1$ will only reply with a normal ACK beacon regardless of the distance estimated. Asking the original receiver to send an ACK beacon is necessary, since the acknowledgement ensures that the event information included in the data packet will not be lost, even if the handover attempt fails. However, the other neighbors of $S1$ will try to send back a handover reply after they receive the handover request. But not all of them are eligible to participate. Instead, only those active nodes whose distance

5. HANDOVER MECHANISM

with respect to $S1$ is evaluated to not exceed d are qualified. The distance threshold ensures that once the transmitter transfers the communication to a new relay node, its remaining data packets can be completely transmitted before the new link breaks. In other words, it guarantees that the handover will be triggered at most once during a node's data transmission. Since the handover mechanism introduces latency itself, it is preferable to reduce the handover frequency as much as possible.

If the neighbors of $S1$ are active, they will definitely receive the handover request. This is because a node should broadcast a beacon as soon as it wakes up and detects a free medium. But in case of a busy medium, the node has to back-off for a period of time during which it will overhear the handover request. Among the active receivers, if more than one node estimates that its distance with respect to $S1$ is not larger than d , collision may occur. This is because all the qualified neighbors of $S1$ will try to transmit a handover reply beacon once they receive the handover request. Therefore, it may happen that the transmitter receives several handover reply beacons at the same time. To avoid collision, a new field is included in each data packet indicating the maximum waiting interval the original transmitter should wait for a handover reply beacon. We name this field by WI, which is composed of multiple time slots. Unlike the back-off field used in beacons whose value is variable, the size of WI is the same in all the handover requests. A node, that is qualified to participate in the handover process, will randomly choose a back-off value ranging from 0 to WI as soon as it receives a handover request. The one that selects the smallest value will win the medium and will be regarded as the new receiver for $S1$.

However, if there is no active receiver whose distance with respect to $S1$ is smaller than d , or if collision occurs, $S1$ will not receive a valid handover reply beacon before the WI expires. In this case, $S1$ will initiate a new handover attempt by broadcasting another handover request. This continues until it eventually receives a handover reply beacon from one of its neighbors. In order to justify handover, the time for obtaining a handover reply beacon should happen before the existing link breaks. Specifically, the mobile transmitter $S1$ should receive a handover reply beacon prior to moving out of the radio transmission range of $R1$. This enables the original transmitter to transfer its data packets from a weak sensor node to a strong one, and thus, provides an unbroken communication link. As described in Figure 5.2, the parameters $T1$, $T2$ and $T3$ denote the time at which a handover starts, a handover completes, and an established link

terminates, respectively. To make the handover mechanism meaningful, $T2$ should occur in advance of $T3$.

Once the original transmitter successfully receives a handover reply beacon, the handover process accomplishes. From this moment on, $S1$ will resume transmitting its remaining $(n - w)$ data packets with the newly found receiver, $R2$, in a burst manner. The original receiver, $R1$, by snooping on the data packet between $S1$ and $R2$, will realize that it is no longer the intended receiver. Therefore, to save energy on idle listening, it will enter into a sleep state at once. Different from $R1$, as soon as $R2$ receives a data packet that is destined to it, it will be aware that it is selected as a new receiver. As a result, it will prepare an ACK beacon to assist the transmitter in the data communication. After $R2$ receives the last data packet from $S1$, it will broadcast a beacon containing all the DST, BW and SRC fields. This helps to invite the new data transmission from other transmitters.

By taking advantage of a Boolean field in a data packet, the transmission mode can be controlled to either broadcast or unicast. A Boolean field with the value *true* signals that the data packet is a handover request. Once a receiver receives such a data packet, depending on whether its own ID matches the DST value given in the data packet or not, it will give different feedbacks. If the node finds the distance between the transmitter and itself is not larger than d whereas its ID does not match the DST value, it will learn that it is a qualified relay candidate. Therefore, it will attempt to respond with a handover reply after backing-off for a period of time. However, if the node finds the DST value of the data packet is the same with its own ID, it will realize that it is the original receiver. By knowing that the quality of the link is deteriorating and the transmitter is looking for a new relay node, the original receiver will reply with a normal ACK beacon. From this perspective, the Boolean field in handover requests helps the neighbor nodes of the transmitter to be clear of their roles.

Similarly, a Boolean field is also employed in each handover reply beacon. If the original transmitter receives a beacon from a node in which the Boolean field with the value *true* is embedded, it will learn that the handover attempt succeeds, and thus, will immediately transfer its communication from the original receiver to this newly discovered node. However, if the Boolean field of the beacon is *false*, the transmitter will know that it is only an ACK beacon transmitted by the original receiver. Consequently, it will keep waiting until a handover reply beacon is received or the maximum

5. HANDOVER MECHANISM

waiting interval expires. If the original transmitter detects some channel activity via the CCA check but without the beacon Frame Delimiter (SFD), it will learn that a collision has occurred on the handover reply beacons. In this case, it will prepare a new handover request and broadcast it in the next round. However, if the original transmitter receives nothing during the WI interval, it will deduce that no qualified receivers are available at the moment. Therefore, it will continue the handover process until at least one competent receiver appears before the link breaks.

Figure 5.3: The format of a beacon in the optimized RI-MAC for an IEEE 802.15.4 radio

Neighbors of the receiver except the original transmitter will not compete for the channel after they receive a handover reply beacon. Since the Boolean value can give different instructions to nodes serving different roles, collision between data packets can be avoided. This makes an additional BW field unnecessary to be used in handover reply beacons. The original transmitter, therefore, as soon as it receives a handover reply beacon, will immediately proceed with the remaining data transmission without the need for backing-off.

The format of a beacon in the optimized RI-MAC is summarized in Figure 5.3. Dashed rectangles indicate optional fields. Different from the beacon format in RI-MAC in which the SRC field is a must, the SRC field is only involved in beacons at the beginning and the end of the data communication in the optimized RI-MAC. In addition, a new field called *handover* will be included in beacons acting as a handover reply when mobility is detected.

5.2 Determination of the Distance Threshold

As mentioned before, once a transmitter occupies the medium and finds that its distance with respect to the receiver exceeds d , it will begin to broadcast a handover request. Suppose n data packets are transmitted in burst. Among them, the first w data packets are used to obtain the first w ACK beacons which enable the transmitter to estimate the distance. After the distance is evaluated, $(n - w)$ data packets are left for sending. The time required for the transmission of these $(n - w)$ packets is given by:

$$T_{data}^{n-w} = (n - w)(T_{data} + T_b + 2T_{SIFS}) \quad (5.1)$$

Here, the term $(T_{data} + T_b + 2T_{SIFS})$ indicates the time for transmitting a data packet. The parameters T_{data} , T_b and T_{SIFS} denote the time required to send a data packet, to broadcast a beacon, and to switch the radio from transmitting to receiving or receiving to transmitting state, respectively.

The distance a node travels during T_{data}^{n-w} can be variable depending on the different moving styles it may have. However, most of the mobility models which generalize the movement characteristic from real applications verify that a node usually changes its direction and speed once the time expires or the maximum permitted distance is reached (14). Therefore, we assume a node moves at a uniform speed v along the radius of the radio transmission range, R , of its partner. Under this assumption, the smallest distance over which the remaining $(n - w)$ data packets can be transmitted is quantified as:

$$s = (n - w)(T_{data} + T_b + 2T_{SIFS})v \quad (5.2)$$

The distance threshold should be defined as the radio transmission range moving out of which the transmitter cannot finish transmitting its data packets to the receiver before the link breaks. Therefore, d should be set as $(R - S)$, which can be further written as:

$$d = R - (n - w)(T_{data} + T_b + 2T_{SIFS})v \quad (5.3)$$

The distance threshold expressed here actually gives the earliest time to begin the handover mechanism. If the mobile transmitter does not follow an uniform rectilinear

5. HANDOVER MECHANISM

translation, or if its movement trajectory does not across the center of the radio circle of the receiver, even if the distance is estimated to be larger than d , the link will not break during the transmission of the $(n-w)$ data packets. From this perspective, the triggering of the handover mechanism by using this distance threshold may be unnecessary in some situations. Nevertheless, it definitely guarantees an unbroken communication link regardless of the moving pattern of a mobile node. Since a node's movement style is unpredictable, aiming at providing an uninterrupted link, this distance threshold is preferable to be used as a sign for the initialization of the handover mechanism.

5.3 Handover Trade-off

Determination of the distance threshold is a tradeoff. On one hand, the handover mechanism is meaningful to be investigated if the occurrence of the link termination is not rare. This requires a comparatively long communication, and therefore, more data packets should be transmitted in burst. The increased number of data packets results in a smaller value of the distance threshold.

On the other hand, since multiple data packets are transmitted at a time in the optimized RI-MAC, the packet communication duration can be long, leading to an even longer listen period. This, in turn, calls for a large duty cycle. However, for the sake of achieving high performance in terms of energy consumption, latency and throughput, the number of data packets that are transmitted in burst should be small. This requires a greater distance threshold to support handover.

5.4 Merit of the Handover Mechanism

A MAC protocol dealing with mobility is faced with many critical requirements. To begin with, to reduce the packet transmission latency, it should enable a mobile node to establish a link with a static node as quickly as possible. Secondly, it should reduce the cost of frequent disconnections and link establishments due. Thirdly, it should enable the transfer of plenty of packets once a link is established and after it breaks. Fourthly, it should contribute to the optimization of the overall lifetime of the network. None of the existing or proposed MAC protocols fulfill these requirements in their entirety.

With the development of the adaptive handover mechanism, the optimized RI-MAC successfully overcomes these challenges and makes the following contributions:

- The handover mechanism enables to transmit data packets with a minimum delay. Once the distance between a transmitter and a receiver is estimated to be larger than the threshold, a handover request will be embedded in each data packet. This indicates that the event sensed in environment will be broadcasted together with a neighbor discovery request. Since the transmitter needs a period of time for detecting a qualified receiver, the data transmission rate in the case of mobility cannot be as large as that in the static scenario. However, to mostly reduce the duration required for transmitting a data packet, the handover mechanism defines a constant waiting interval which limits the time that the transmitter has to wait for a handover reply beacon. If the transmitter does not receive any response until WI expires, it will broadcast another packet. In this way, the handover mechanism enables a node to transmit data packets at a maximum possible rate without affecting the neighbor discovery.
- The handover mechanism is triggered only when necessary. Hence, it does not influence the performance of a MAC protocol in a static setting. This is realized by the Boolean field included in each data packet. If the transmitter evaluates that the distance with respect to its receiver does not exceed the threshold, it will set the Boolean field to *false*. All the neighbors of the transmitter, after receiving the data packet, will learn that it is not looking for a new receiver. Therefore, only the original receiver will reply with an ACK beacon. However, when the quality of the link deteriorates, the Boolean field will be set to *true*. Based on the Boolean value along with the destination information embedded in the data packet, the neighbors of the transmitter will be clear of their roles and all the qualified receivers can give feedbacks accordingly.
- Data transmission does not significantly interfere with neighbor discovery and vice versa. This is because the nodes that take responsibility for the data reception and for the handover reply are different. When a data packet including a neighbor discovery request is broadcasted, only the original receiver will acknowledge with an ACK beacon. Whether the transmitter receives a handover reply

5. HANDOVER MECHANISM

beacon during the waiting interval, the sensed data will be successfully transferred. Likewise, only the active neighbors whose distance to the transmitter does not exceed the threshold are able to participate in the new link establishment. The original receiver cannot take part in the handover process, even though the transmitter may not receive any handover response.

6

Latency Analysis with Handover

In this chapter, a mathematical model is set up to theoretically analyze the latency required for the transmission of n data packets. During the transmission, a handover mechanism is applied to transfer the communication a better link when the quality of the current link deteriorates. Particularly, the average time the handover mechanism itself introduces will be quantified and the parameters that influence this quantity will be investigated.

6.1 Handover Latency

The handover latency is defined as the interval between $T1$ and $T2$ as shown in Figure 5.2. $T1$ is the time at which the transmitter evaluates that its distance with respect to the receiver exceeds d . $T2$ is the time at which the transmitter discovers a new relay node.

As mentioned earlier, among the neighbors of the transmitter (excepting the original receiver), only those which are already active and whose distance with respect to the transmitter does not exceed d will attempt to send back a handover reply. If the number of these nodes is larger than one, a collision between the handover reply beacons may occur, even though a WI field is provided in each handover request to avoid it. Therefore, the latency that the handover mechanism introduces is closely related to the number of nodes around the transmitter that may participate in the handover process. We represent this number by N . By also including the original receiver, the network density can be expressed. Denoting it by M , N can be formulated as ($N = M - 1$).

6. LATENCY ANALYSIS WITH HANDOVER

Table 6.1: description of parameters

i	The maximum number of handover attempts
p_j^f	The probability that the node is unable to transmit a handover reply in the first j attempts
p_j^s	The probability that the node is able to transmit a handover reply in the j_{th} attempt
t_j^f	The time consumed in the j_{th} handover attempt that fails
t_j^s	The time consumed in the j_{th} handover attempt that succeeds
j	The index of a handover attempt ($j \in (0, i)$)

If there is only one qualified receiver participating in the handover, collision on the handover reply beacons will be absolutely refrained. Otherwise, collision may occur. Therefore, in order to provide an accurate expression, the performance of the handover mechanism should be analyzed differently for different values of M .

6.1.1 For Network Density $M = 2$

To justify handover, at least two nodes should be located within the maximum radio transmission range of the transmitter. This requires that the network density should be at least two. As one of the nodes is the original receiver, only one neighbor of the transmitter will be involved in the handover process when $M = 2$. In this case, collision will not occur on the handover reply beacons. Nevertheless, a handover may still fail because (a) the node does not receive the handover request since it is in a sleep state, or (b) the distance it estimates is larger than d . If the transmitter does not receive any handover reply during the WI duration in the first handover, it will immediately broadcast another request. This attempt continues until a handover reply beacon is eventually received by the transmitter. Figure 6.1 describes a complete handover process. In the first three attempts, the handover fails. But it succeeds in the fourth attempt.

The handover succeeds in the first attempt will introduce the shorted latency for transferring the data communication. However, the handover succeeds in the last j_{th} attempt is the worst case. It indicates that the node is unable to send a handover reply beacon in the first $(j - 1)$ attempts due to a small estimated distance or a sleeping

Figure 6.1: The handover continues until it finally succeeds when $M = 2$

state. This will lead to a maximum latency for maintaining an unbroken link. As the transmitter may receive a reply beacon at any handover attempt, the average duration of the handover process for a single neighbor (except the original receiver), $\overline{t_{M=2}}$, can be expressed as:

$$\overline{t_{M=2}} = \frac{p_1^s t_1^s + p_1^f p_2^s (t_1^f + t_2^s) + p_2^f p_3^s (t_1^f + t_2^f + t_3^s) \cdots + p_{i-1}^f p_i^s (t_1^f + t_2^f + \dots + t_{i-1}^f + t_i^s)}{p_1^s + p_1^f p_2^s + p_2^f p_3^s + \cdots + p_{i-1}^f p_i^s} \quad (6.1)$$

The meaning of all the parameters in Equation 6.1 is explained in Table 6.1. The value of p_j^s is the product of the probability that a node is awake, p_a , and the probability that the estimated distance is not larger than the threshold, p_d . Since p_a is equal to the duty cycle D ($p_a = D$), and p_d depends only on the network density ($p_d = \frac{\pi d^2}{\pi R^2}$), the value of p_j^s is regardless of the attempt at which the handover succeeds. Consequently, it is the same for all j , and thus, p_j^s can be substituted by p^s which can be expressed as:

$$p^s = D \frac{d^2}{R^2} \quad (6.2)$$

6. LATENCY ANALYSIS WITH HANDOVER

Due to the substitution, Equation 6.1 can be simplified to:

$$\overline{t_{M=2}} = \frac{p^s t_1^s + \sum_{j=1}^{i-1} \left(p^s p_j^f ((t_1^f + \dots + t_j^f) + t_{j+1}^s) \right)}{p^s (1 + \sum_{j=1}^{i-1} p_j^f)} \quad (6.3)$$

Since the parameters that determine the duration of a handover (with a success or a failure) have nothing to do with the index of attempts, the value of t_j^f is the same for all j and so does the value of t_j^s . Hence, t_j^f and t_j^s can be replaced by t^f and t^s , respectively. Therefore, Equation 6.1 can be further simplified to:

$$\overline{t_{M=2}} = \frac{p^s t^s + \sum_{j=1}^{i-1} \left(p^s p_j^f (j t^f + t^s) \right)}{p^s (1 + \sum_{j=1}^{i-1} p_j^f)} \quad (6.4)$$

As illustrated in Figure 6.1, the time durations in which a handover attempt fails, t^f , and succeeds, t^s , can be expressed as:

$$t^f = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + WI\sigma \quad (6.5)$$

And

$$t^s = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + t_l^{backoff} \quad (6.6)$$

The parameters $N_{request}$, N_{reply} , σ and $t_l^{backoff}$ denote, in respective order, the size of a handover request, the size of a handover reply, a slot duration, and the average backoff interval determined by l nodes. Here, l signifies the number of active neighbors of a transmitter (without considering the original receiver) whose distance is estimated to not exceed d . Since only one neighbor exists in this situation and the successful handover indicates that the node is awake and its distance with respect to the transmitter is not larger than d , $l = 1$. Therefore, the average backoff duration determined by only one node, $t_{l=1}^{backoff}$, can be expressed as:

$$t_{l=1}^{backoff} = \frac{\sum_{m=1}^{WI} (m\sigma)}{WI} \quad (6.7)$$

To enable the performance evaluation of the handover mechanism, the initial distance between a transmitter and a receiver should be larger than the threshold if the transmitter is moving away from the receiver. Since the transmitter may be initially located at any place between d and R within the radio transmission range of the receiver,

the distance it travels until the link breaks on average when the handover mechanism is not applied can be evaluated as: $s = \frac{R-d}{2}$. Since a node is assumed to move at a uniform speed v along the radius of the radio circle of its communication partner, the time needed to cover this distance can be expressed as:

$$T3 - T1 = \frac{R - d}{2v} \quad (6.8)$$

To maintain an unbroken link, the handover should succeed before the link terminates. This requires an upper bound on the handover duration. Specifically, the interval $(t_1^f + t_2^f + \dots + t_{i-1}^f + t_i^s)$ should be shorter than $(T3 - T1)$. Since $(t_1^f + t_2^f + \dots + t_{i-1}^f + t_i^s)$ is equivalent to $((i-1)t^f + t^s)$, this requirement can be expressed as:

$$(i-1)t^f + t^s \leq T3 - T1 \quad (6.9)$$

By replacing the parameters t^f and t^s with Equations 6.5 and 6.6, the latest handover attempt at which the transmitter should receive a handover reply beacon can be evaluated. Therefore, the index i given in Equation (6.4) is limited to:

$$i \leq \frac{\frac{R-d}{2v} - \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + t_l^{backoff}}{\frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + WI\sigma} + 1 \quad (6.10)$$

In Equation (6.4), the coefficient p_1^f (the weight for calculating the duration of a handover process which fails in the first attempt but succeeds in the second attempt) can be expressed as $(1 - p_a p_d)^1$. Since $(1 - p_a p_d)^1$ is equal to $(p_a(1 - p_d) + (1 - p_a)p_d + (1 - p_a)(1 - p_d))$, the coefficient p_1^f can be unfolded as:

$$p_1^f = p_a(1 - p_d) + (1 - p_a)p_d + (1 - p_a)(1 - p_d) \quad (6.11)$$

Equation (6.11) actually explains the reason that the node is not able to transmit a handover reply beacon in the first attempt. The handover failure comes from three sources:

- (1) The node is awake but it estimates its distance with respect to the transmitter to be larger than d ;
- (2) The node has not entered into a listen period yet, even though the distance between it and the transmitter does not exceed d ;

6. LATENCY ANALYSIS WITH HANDOVER

- (3) The node is still in a sleep period and its distance to the transmitter is larger than d .

Nevertheless, p_j^f cannot be described as $(1 - p_a p_d)^j$ when $j \in (1, i - 1]$. This is because the term $(1 - p_a p_d)^j$ implies that in each of the first j attempts, the failure of the handover is a result of either one or both of the following reasons: (a) the node estimates its distance with respect to the transmitter to be larger than d ; (b) the node has not entered into a listen period yet. Therefore, one of the reasons that can lead to a handover failure is that the node is in a sleep state in the first handover attempt, wakes up in the second attempt, and enters into a sleep state again in the third attempt, etc. This, however, does not make sense, since the time spent on the handover process until it succeeds is much shorter than the listen phase. Due to the introduction of the duty cycle and its small value in most of the cases, this time will be even shorter than the sleep phase. As a result, transition from an active to a sleep state or the other way around can only occur once during a handover operation. Consequently, all the impossible cases that can cause the handover failure in the first j attempts should be excluded.

handover cases	1st fails		2nd fails		3rd succeeds	
	awake	less than d	awake	less than d	awake	less than d
1	F	T	F	T	T	T
2	F	T	T	F	T	T
3	F	T	F	F	T	T
4	T	F	F	T	T	T
5	T	F	T	F	T	T
6	T	F	F	F	T	T
7	F	F	F	T	T	T
8	F	F	T	F	T	T
9	F	F	F	F	T	T

Figure 6.2: Impossible cases that lead to a handover failure for $j = 2$

Figure 6.2 gives an example. It displays the impossible instances when $j = 2$. p_2^f is the probability that the handover fails in the first and second attempts but succeeds in the third attempt. Under each attempt, the handover failure can be attributed to one or both of the above reasons (a) and (b). The Boolean values T and F represent whether these two conditions are satisfied or not. For a handover to be successful at last, the node has to be awake and located within the radio transmission range of the transmitter with the radius d . This is the reason why the value of the node state is T in the last two columns for all the cases. Since the node is unable to send a handover reply beacon in the first two attempts, the values for whether it is being awake and whether its distance is estimated to not larger than the threshold should not be T at the same time. Instead, they should be either T and F , F and T , or F and F in the first and second handover attempts. By taking all the combinations of the Boolean values into account, there are together nine handover cases. However, not all of them are valid for calculating the probability p_2^f . As highlighted in Figure 6.2, the fourth and sixth cases do not make sense, since the radio state of the node in these cases changes more than once. Therefore, these two cases should be discarded for the evaluation of the probability p_2^f .

We use the parameters $A1$ and $A2$ to denote the probability that the handover fails in the first and second attempts due to the reasons given in the fourth and sixth cases. By excluding these two invalid probabilities, the weight for calculating the duration of the handover process which fails in the first two attempts but succeeds in the third attempt, p_2^f , can be expressed as:

$$p_2^f = (1 - p_a p_d)^2 - A1 - A2 \quad (6.12)$$

Since the item $(1 - p_a p_d)$ is equivalent to the polynomial $(p_a(1 - p_d) + (1 - p_a)p_d + (1 - p_a)(1 - p_d))$, the term $(1 - p_a p_d)^2$ given in Equation (6.12) can be quantified as $(p_a(1 - p_d) + (1 - p_a)p_d + (1 - p_a)(1 - p_d))^2$. Therefore, Equation (6.12) can be transferred to:

$$p_2^f = \left(p_a(1 - p_d) + (1 - p_a)p_d + (1 - p_a)(1 - p_d) \right)^2 - A1 - A2 \quad (6.13)$$

6. LATENCY ANALYSIS WITH HANDOVER

After reducing the degree of the polynomial $(p_a(1-p_d)+(1-p_a)p_d+(1-p_a)(1-p_d))^2$ from two to one and after simplification, Equation (6.13) can be expressed as:

$$\begin{aligned}
p_2^f = & (1-p_a)p_d(1-p_a)p_d \\
& + (1-p_a)p_dp_a(1-p_d) \\
& + (1-p_a)p_d(1-p_a)(1-p_d) \\
& + p_a(1-p_d)(1-p_a)p_d \\
& + p_a(1-p_d)p_a(1-p_d) \\
& + p_a(1-p_d)(1-p_a)(1-p_d) \\
& + (1-p_a)(1-p_d)(1-p_a)p_d \\
& + (1-p_a)(1-p_d)p_a(1-p_d) \\
& + (1-p_a)(1-p_d)(1-p_a)(1-p_d) \\
& - A1 - A2
\end{aligned} \tag{6.14}$$

Each probability item in Equation (6.14) refers to each case illustrated in Figure 6.2. Therefore, there are nine items together describing all the possible reasons that can lead to a handover failure in the first two attempts. Among them, two is not realistic. Obviously, these two invalid probabilities of the handover failure are $p_a(1-p_d)(1-p_a)p_d$ and $p_a(1-p_d)(1-p_a)(1-p_d)$. After learning this, the values of $A1$ and $A2$ can be solved, and thus, expressed as:

$$A1 = p_a(1-p_d)(1-p_a)p_d \tag{6.15}$$

And

$$A2 = p_a(1-p_d)(1-p_a)(1-p_d) \tag{6.16}$$

By inserting Equations (6.15) and (6.16) into Equation (6.12), the probability that the handover fails in the first two attempts but succeeds in the third attempt, p_2^f , can be expressed as:

$$p_2^f = (1-p_ap_d)^2 - p_a(1-p_d)(1-p_a)p_d - p_a(1-p_d)(1-p_a)(1-p_d) \tag{6.17}$$

After simplification, Equation (6.17) can be rewritten as:

$$p_2^f = (1 - p_a p_d)^2 - p_a(1 - p_d)(1 - p_a) \quad (6.18)$$

handover cases	1 st fails	2 nd fails	3 rd fails	4 th succeeds
1	T	T	T	T
2	T	T	F	T
3	T	F	T	T
4	T	F	F	T
5	F	T	T	T
6	F	T	F	T
7	F	F	T	T
8	F	F	F	T

Figure 6.3: Impossible cases that lead to a handover failure for $j = 3$

Similarly, several cases can lead to a handover failure in the first three attempts when $j = 3$, but some of them are invalid. Unlike that the radio state of the node can only change once during the handover process, there is no limitation on the frequency that the distance between the node and the transmitter varies. As a result, only the condition that whether the node is awake or not should be taken into consideration for figuring out the meaningless cases. This is why the factor of distance is not presented in Figure 6.3. Since the handover is required to be eventually successful, the node has to be active in the fourth attempt, causing the value of the node state to be T in the last column for all the cases. As emphasized in Figure 6.3, four out of eight cases do not make sense. Given that the node is already active, the reason of a handover failure can only be that its relative distance is larger than d . However, the phenomenon that the node is in a sleep period can sufficiently lead to a handover failure regardless of the estimated distance. Based on this rule and with the exclusion of the four meaningless

6. LATENCY ANALYSIS WITH HANDOVER

cases, the probability that a handover fails in the first three attempts but succeeds in the fourth attempt, p_3^f , can be quantified as:

$$\begin{aligned}
 p_3^f &= (1 - p_a p_d)^3 \\
 &\quad - p_a(1 - p_d)p_a(1 - p_d)(1 - p_a) \\
 &\quad - p_a(1 - p_d)(1 - p_a)p_a(1 - p_d) \\
 &\quad - p_a(1 - p_d)(1 - p_a)(1 - p_a) \\
 &\quad - (1 - p_a)p_a(1 - p_d)(1 - p_a)
 \end{aligned} \tag{6.19}$$

After simplification, Equation (6.19) can be transferred to:

$$p_3^f = (1 - p_a p_d)^3 - \left(2p_a^2(1 - p_d)^2(1 - p_a) + 2p_a(1 - p_d)(1 - p_a)^2 \right) \tag{6.20}$$

In the same way, by removing all the impossible cases, the probability that the handover fails in the first four attempts but succeeds in the fifth attempt, p_4^f , and the probability that the handover fails in the first five attempts but succeeds in the sixth attempt, p_5^f , can be solved. They are quantified as:

$$p_4^f = (1 - p_a p_d)^4 - \left(3p_a^3(1 - p_a)(1 - p_d)^3 + 5p_a^2(1 - p_a)^2(1 - p_d)^2 + 3p_a(1 - p_a)^3(1 - p_d) \right) \tag{6.21}$$

And

$$\begin{aligned}
 p_5^f &= (1 - p_a p_d)^5 \\
 &\quad - \left(4p_a^4(1 - p_d)^4(1 - p_a) \right. \\
 &\quad + 9p_a^3(1 - p_d)^3(1 - p_a)^2 \\
 &\quad + 9p_a^2(1 - p_d)^2(1 - p_a)^3 \\
 &\quad \left. + 4p_a(1 - p_d)(1 - p_a)^4 \right)
 \end{aligned} \tag{6.22}$$

Likewise, the equation expressing the probability of a handover failure in the first j attempts, p_j^f , can be given for all j , where $j \in (1, i)$. In order to find the general term formula of p_j^f , the parameters x and y are used to represent the terms $p_a(1 - p_d)$

and $(1 - p_a)$, respectively. After transformation, the probability that the node is in an active state, p_a , and the probability that its distance with respect to the transmitter is not greater than the threshold, p_d , can be expressed as:

$$p_a = 1 - y \quad (6.23)$$

And

$$p_d = 1 - \frac{x}{1 - y} \quad (6.24)$$

By inserting Equations (6.23) and (6.24) into Equations (6.11), (6.18), (6.20), (6.21) and (6.22), the probability of the handover failure for $j \in [1, 5]$ can be expressed as:

$$p_1^f = x + y \quad (6.25)$$

$$p_2^f = x^2 + xy + y^2 \quad (6.26)$$

$$p_3^f = x^3 + x^2y + xy^2 + y^3 \quad (6.27)$$

$$p_4^f = x^4 + x^3y + x^2y^2 + xy^3 + y^4 \quad (6.28)$$

$$p_5^f = x^5 + x^4y + x^3y^2 + x^2y^3 + xy^4 + y^5 \quad (6.29)$$

All these equations are discovered to meet Pascal's triangle rule (5). Therefore, after transformation by using formulas $(x = p_a(1 - p_d))$ and $(y = 1 - p_a)$, the probability that the handover fails in the first j attempts, p_j^f , can be expressed as:

$$p_j^f = p_a^j(1 - p_d)^j + p_a^{j-1}(1 - p_d)^{j-1}(1 - p_a) + \cdots + p_a(1 - p_d)(1 - p_a)^{j-1} + (1 - p_a)^j \quad (6.30)$$

After generalization, Equation (6.30) can be simplified as:

$$p_j^f = \sum_{m=0}^j p_a^{j-m}(1 - p_d)^{j-m}(1 - p_a)^m \quad (6.31)$$

6. LATENCY ANALYSIS WITH HANDOVER

By combining Equations (6.6) and (6.7), the duration of a successful handover attempt can be estimated. Finally, by inserting Equations (6.2), (6.5), (6.6) and (6.31) into Equation (6.4) and after simplification, the average handover latency for a single neighbor (except the original receiver) can be evaluated.

6.1.2 For Network Density $M > 2$

When the network density is larger than two, the number of neighbors of the transmitter (except the original receiver) is greater than one. Therefore, a handover failure can occur not only due to the fact that no candidate is awake or\and the relative distance is larger than d , but also due to the collision on the handover reply beacons. Since different number of beacons transmitted simultaneously can lead to different back-off durations which will further influence the latency introduced by the handover, the occurrence of collision makes the analysis more complicated. Specifically, the challenge for the evaluation of the handover latency in this situation lies in the correlation among three elements, namely:

- (1) the number of nodes that are active;
- (2) the number of nodes whose relative distance does not exceed the threshold;
- (3) the number of nodes that contribute to a collision.

Of these, the last one is determined by the first two. Figure 6.4 gives an example for $M = 4$ to explain the correlation among the three factors. In this situation, except the original receiver, $R1$, the transmitter, $S1$, has three neighbors, $R2$, $R3$ and $R4$. Once the transmitter finds that the distance between it and $R1$ exceeds the threshold, it will broadcast a handover request. Even though one of the surrounding neighbors, $R2$, has already woken up from a sleep period at the moment, it cannot acknowledge with a handover reply beacon. This is because it estimates its distance with respect to the transmitter to be larger than d . As $S1$ receives nothing in the maximum waiting interval, it initiates another handover. During the new attempt, $R3$ is in an active state, and $S1$ has moved into the radio transmission range of both $R2$ and $R3$ with the radius d . Therefore, both $R2$ and $R3$ are qualified to broadcast a handover reply beacon. However, before transmission, they are required to select a random backoff time according to the WI field provided in the handover request. As both of them

Figure 6.4: The handover continues until it finally succeeds when $M > 2$

choose the same value, a collision occurs, leading to the third handover attempt. Even though, this time, all the relay candidates, $R2$, $R3$ and $R4$, meet the requirement for participating in the handover process, no valid handover reply beacon is received by $S1$. The same backoff value they choose causes another handover failure. However, as no collision happens in the fourth attempt, the transmitter succeeds in the handover, and thus, will transfer its communication to the better link.

For the sake of reducing the evaluation complexity caused by the correlation between the three elements, we have made the following assumptions. Firstly, k out of N neighbors of the transmitter are selected, which are assumed to be in an active state. Then of the k neighbors, l of them are further selected supposing that their distance with respect to the transmitter is less than or equal to the distance threshold. In other words, only k nodes are awake and l of them are enclosed within the radio transmission range of the sender (this range is now equal to d) for every handover attempt. Hence, the situation that k nodes are awake in the first attempt, more than k nodes are awake in the second attempt, and less than k nodes are awake in the third attempt will not

6. LATENCY ANALYSIS WITH HANDOVER

occur. The same argument holds for the l nodes. With these premises, the analysis of the handover latency when the network density is greater than two becomes tractable.

Therefore, the average handover latency in the situation that more than one neighbor (except the original receiver) of the transmitter is present, $\overline{t_{M>2}}$, can be expressed as:

$$\overline{t_{M>2}} = \frac{\sum_{k=1}^{M-1} \sum_{l=1}^k (p(k_awake)p(l \leq d)\overline{t_l})}{\sum_{k=1}^{M-1} \sum_{l=1}^k (p(k_awake)p(l \leq d))} \quad (6.32)$$

The terms $p(k_awake)$ and $p(l \leq d)$ in Equation (6.32) represent the probability that k nodes are awake and l out of k nodes have a relative distance not larger than d . By considering the duty cycle, D , the distance threshold, d , and the radio transmission range of a node, R , the probabilities $p(k_awake)$ and $p(l \leq d)$ can be expressed as:

$$p(k_awake) = C_{M-1}^k D^k (1 - D)^{M-1-k} \quad (6.33)$$

And

$$p(l \leq d) = C_k^l \left(\frac{\pi d^2}{\pi R^2}\right)^l \left(1 - \frac{\pi d^2}{\pi R^2}\right)^{k-l} \quad (6.34)$$

Since the term $\overline{t_l}$ refers to the average handover latency determined by collision on the l nodes, its value is mainly relevant to the number of competitors, l . If such number equals to one, no collision will occur and therefore, the handover will be definitely successful in the first attempt. As a result, $\overline{t_{l=1}}$ shares the same value with t^s which is evaluated in the situation that $M = 2$. After combining Equations (6.6) and (6.7), $\overline{t_{l=1}}$ can be expressed as:

$$\overline{t_{l=1}} = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + \frac{\sum_{m=1}^{WI} (m\sigma)}{WI} \quad (6.35)$$

If l is more than one, a collision will occur. Different values of l will result in different collision probabilities on the handover reply beacons and thus different back-off times. These will in turn affect the average handover latency. Since the handover is assumed

to be successful, in the worst case, at i_{th} attempt, the average handover latency caused only by a collision due to the l nodes ($l \geq 2$) can be expressed as:

$$\overline{t_{l \geq 2}} = \frac{p_l^s \left(t^s + \sum_{j=1}^{i-1} ((1 - p_l^s)^j (j t^f + t^s)) \right)}{p_l^s \left(1 + \sum_{j=1}^{i-1} (1 - p_l^s)^j \right)} \quad (6.36)$$

Since the denominator is a geometric progression, Equation (6.36) can be simplified as:

$$\overline{t_{l \geq 2}} = \frac{p_l^s \left(t^s + \sum_{j=1}^{i-1} ((1 - p_l^s)^j (j t^f + t^s)) \right)}{1 - (1 - p_l^s)^i} \quad (6.37)$$

Here, p_l^s denotes the probability that no collision occurs on the handover reply beacons transmitted by the l nodes. If one of the l nodes transmits a reply beacon in a particular time slot, to make the handover successful, all the remaining $(l - 1)$ nodes have to select a time slot having a larger value. But whether these time slots are the same or not is unimportant, since after sensing the busy medium, the $(l - 1)$ nodes will cancel their beacon transmission and thus collision will not occur in between. Therefore, p_l^s can be expressed as:

$$p_l^s = \frac{C_l^1 \left(\sum_{m=1}^{WI-1} (WI - m)^{l-1} \right)}{WI^l} \quad (6.38)$$

The average back-off time, $t_l^{backoff}$, which determines the value of t^s in Equation (6.6), depends on the time slot that is selected on average by a node for the transmission of the handover reply. Therefore, it should be quantified by considering all the time slots ranging from 0 to $(WI - 1)$, in which no collision occurs among the l nodes. Consequently, $t_l^{backoff}$ can be written as:

$$t_l^{backoff} = \frac{\sum_{m=1}^{WI-1} \left(C_l^1 \frac{(WI-m)^{l-1}}{WI^l} m \sigma \right)}{\sum_{m=1}^{WI-1} \left(C_l^1 \frac{(WI-m)^{l-1}}{WI^l} \right)} \quad (6.39)$$

After simplification, Equation (6.39) can be reduced to:

$$t_l^{backoff} = \frac{\sigma \left(\sum_{m=1}^{WI-1} ((WI - m)^{l-1} m) \right)}{\sum_{m=1}^{WI-1} (WI - j)^{l-1}} \quad (6.40)$$

6. LATENCY ANALYSIS WITH HANDOVER

By inserting Equation (6.40) into Equation (6.6), the value of t^s for $l \geq 2$ can be expressed as:

$$t^s = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + \frac{\sigma \left(\sum_{m=1}^{WI-1} ((WI-m)^{l-1}m) \right)}{\sum_{m=1}^{WI-1} (WI-j)^{l-1}} \quad (6.41)$$

By inserting Equations (6.5), (6.38) and (6.41) into Equation (6.37), the value of $\overline{t_{l \geq 2}}$ can be computed. Finally, by combining Equations (6.33), (6.34), (6.35), (6.37) and (6.32), the average handover latency for the network density larger than two ($M > 2$) can be evaluated.

6.2 Latency before Handover

The communication pattern before $T1$ is mainly used for the link establishment between two nodes and the decision for the initiation of the handover mechanism. Specifically, once a transmitter wins the channel, it will start to transmit its data packets in burst. Each time the communicating receiver receives a data packet, it will reply to the transmitter with an ACK beacon. According to the first w ACK beacons, the transmitter can evaluate the relative distance between it and the receiver with a certain degree of accuracy. This can be accomplished by using one or several location evaluation techniques. If the distance is estimated to be larger than the pre-defined threshold, d , the transmitter will learn that its remaining data packets cannot be completely transmitted before the link breaks. Hence, it will immediately trigger the handover.

The purpose of this thesis is to evaluate the packet transmission latency when the handover mechanism is applied and when it is not applied. In order to make a fair comparison, the communication pattern before $T1$ should be the same for both situations. This requires that both of them should use the same value of w for the distance estimation. In addition, to simplify the quantification of the latency introduced before handover, the transmitter is assumed to win the channel at the second contention regardless of the network density. This holds for both cases. With these premises, the time consumed in the data communication before $T1$, t_1^{with} , can be expressed as:

$$t_1^{with} = T_{CCA} + 2\frac{N_b}{R_t} + T_{SIFS} + \frac{N_{data}}{R_t} + T_r + T_{backoff1} + n_1^{with}T_{unit} \quad (6.42)$$

The above terms T_r and T_{unit} represent the random backoff duration the receiver has to wait after a collision, and the time the transmitter needs to send a data packet. They can be expressed as:

$$T_r = \frac{\sum_{m=1}^{BW} (m\sigma)}{BW} \quad (6.43)$$

And

$$T_{unit} = \frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS} \quad (6.44)$$

In order to justify handover, we assume the transmitter has at least two neighbors. Among them, one is the original receiver communicating with the transmitter before a new link is found. The other is the new receiver discovered during the handover process which helps the transmitter to resume the remaining data transmission. To be in accordance with this requirement, the number of neighbors of the original receiver should be at least two as well. As N denotes the number of transmitters surrounding the receiver (except the original transmitter), and k denotes the number of active nodes among N , k should be equal to or larger than one ($k \geq 1$). Since the original transmitter is definitely awake at the beginning of the communication, it will participate in the medium contention, and thus, should be also taken into account for the evaluation of the backoff duration. Consequently, the collision occurred after the base beacon is broadcasted attributes to the data packets transmitted by all the $(N + 1)$ channel competitors. This makes the total active nodes to be greater than one ($k \geq 2$).

Since k can vary between 2 and $(N + 1)$ and N can be expressed by the network density ($N = M - 1$), the first and the only average backoff interval occurred before handover, $T_{backoff1}$, given in Equation (6.42) can be quantified as:

$$t_{backoff1} = \frac{\sum_{k=2}^M (p(k_awake)\overline{t_k})}{\sum_{k=2}^M p(k_awake)} \quad (6.45)$$

Here, the term $\overline{t_k}$ signifies the backoff interval determined by k nodes out of which one wins the channel in the second contention. Therefore, $\overline{t_k}$ should be evaluated as:

$$\overline{t_k} = \frac{\sum_{u=1}^{BW^{1-1}} \left(C_k^1 (BW - u)^{k-1} u \sigma \right)}{\sum_{u=1}^{BW^{1-1}} \left(C_k^1 (BW - u)^{k-1} \right)} \quad (6.46)$$

6. LATENCY ANALYSIS WITH HANDOVER

After detecting a collision, the receiver will adjust the BW field in the next beacon using the Binary Exponential Backoff (BEB) scheme that takes values of 31, 63, 127 and 255. The variable $BW1$ shown in Equation (6.46) indicates the smallest backoff value 31.

By combining all the Equations (6.42)-(6.46), the average latency of the packet transmission before the initiation of the handover mechanism can be evaluated.

6.3 Latency after Handover

As soon as the transmitter finds a new relay node, it will transfer the communication to it immediately. As w data packets are sent before $T1$ enabling the transmitter to estimate the relative distance between it and the receiver, $n_1^{with} = w$.

However, the average number of data packets transmitted during the handover process, n_2^{with} , will be different for different network densities. Therefore, it should be considered separately according to the values of M , and thus, can be formulated as:

$$n_2^{with} = \frac{\bar{t}_m - t_m^s}{tf} + 1 \quad (6.47)$$

Here, m can be replaced by either $M = 2$ or $M > 2$ depending on the number of neighbors of the transmitter. The time duration of a successful handover process, t_m^s , has the same value for the case $M = 2$ and the case $M > 2$ but $l = 1$. However, when the number of nodes that are active and have a relative distance not exceed the threshold is greater than two ($l \geq 2$), the value of t_m^s varies. Consequently, the parameter t_m^s should be evaluated as a piecewise function, and thus, can be expressed as:

$$t_m^s = \begin{cases} \frac{N_{request} + N_b + N_{reply}}{R_t} + t_{SIFS} + \frac{\sum_{m=1}^{WI} (m\sigma)}{WI} & (M = 2) | (M > 2 \& l = 1) \\ \frac{N_{request} + N_b + N_{reply}}{R_t} + t_{SIFS} + \frac{\sigma \left(\sum_{m=1}^{WI-1} ((WI-m)^{l-1} m) \right)}{\sum_{m=1}^{WI-1} (WI-j)^{l-1}} & (M > 2 \& l \geq 2) \end{cases} \quad (6.48)$$

Since the total number of data packets transmitted in burst is n , the time required for the remaining data transmission over the newly discovered link, t_3^{with} , can be expressed as:

$$t_3^{with} = (n - n_1^{with} - n_2^{with}) \left(\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS} \right) - T_{SIFS} \quad (6.49)$$

6.3 Latency after Handover

Finally, by adding together the latency introduced before handover, t_1^{with} , during handover, $\overline{t_{M=2}}$ or $\overline{t_{M>2}}$, and after handover, t_3^{with} , the time required to transmit the entire n data packets in the optimized RI-MAC in which the handover mechanism is used can be evaluated.

6. LATENCY ANALYSIS WITH HANDOVER

7

Latency Analysis without Handover

7.1 Latency in the Original Link

7.1.1 Latency before $T1$

In order to evaluate the latency required for the transmission of n data packets during which the link breaks, the transmitter that wins the channel at the beginning of the communication should be located out of the radio transmission range of the receiver (the radius of the radio circle is d now), if they move in opposite directions. Since the time duration consumed before $T1$ (as described in Figure 4.1) is the same for both the situations with and without the utilization of the handover mechanism, the time interval introduced before $T1$ when the handover mechanism is not applied, $t_1^{without}$, shares the same value with t_1^{with} . Therefore, $t_1^{without}$ can also be expressed as:

$$t_1^{without} = T_{CCA} + 2\frac{N_b}{R_t} + T_{SIFS} + \frac{N_{data}}{R_t} + T_r + T_{backoff1} + wT_{unit} \quad (7.1)$$

7.1.2 Latency between $T1$ and $T3$

If a transmitter whose distance with respect to the receiver exceeds d wins the channel, its data transmission will be interrupted because the link will break in the middle of the communication when the handover mechanism is not used. Since the transmitter is in an uniform rectilinear motion and may be initially located at any place between d and R within the radio transmission range of the receiver, the distance it travels between

7. LATENCY ANALYSIS WITHOUT HANDOVER

$T1$ and $T3$ (as shown in Figure 4.1) on average can be evaluated as: $s = \frac{R-d}{2}$. As we suppose that the transmitter always moves at an uniform speed v along the radius of the radio transmission range of the receiver, the time needed to cover this distance, $t_2^{without}$, can be written as:

$$t_2^{without} = \frac{R-d}{2v} \quad (7.2)$$

Once the transmitter moves out of the maximum radio transmission range of the receiver, the link established between them will disrupt. Consequently, at the moment of the link termination, the transmitter may not receive an ACK beacon from the receiver even though it has already sent a data packet. Likewise, the receiver may not receive any data packet after it broadcasts a beacon. But the packet has to be retransmitted in any case. Therefore, the number of data packets transmitted during $t_2^{without}$ can be evaluated as:

$$n_2^{without} = \frac{t_2^{without} - \frac{(\frac{N_{data}}{R_t} + t_{SIFS} + \frac{N_b}{2R_t}) + \frac{N_{data}}{2R_t}}{2}}{\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2t_{SIFS}} \quad (7.3)$$

After inserting Equation (7.2) into Equation (7.3) and after simplification, the number of data packets transmitted during $t_2^{without}$ can be expressed as:

$$n_2^{without} = \frac{\frac{R-d}{2v} - \left(\frac{3N_{data}}{4R_t} + \frac{T_{SIFS}}{2} + \frac{N_b}{4R_t} \right)}{\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS}} \quad (7.4)$$

Since the total number of data packets transmitted in burst is composed of the number of data packets sent before $T1$, between $T1$ and $T3$, and after $T3$, the number of data packets left for transmission after the original link breaks, $n_3^{without}$, can be quantified as:

$$n_3^{without} = n - n_1^{without} - n_2^{without} \quad (7.5)$$

Since the number of data packets transmitted during $t_1^{without}$ and $t_2^{without}$ has already been calculated, the number of the remaining data packets that are required to be transmitted over a new link can be completely expressed as:

$$n_3^{without} = n - w - \frac{\frac{R-d}{2v} - \left(\frac{3N_{data}}{4R_t} + \frac{T_{SIFS}}{2} + \frac{N_b}{4R_t} \right)}{\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS}} \quad (7.6)$$

7.2 Latency for a New Link Establishment

Figure 7.1: The establishment of a new link

Once the link breaks, the transmitter will search for a new link. It will keep idle listening until it receives a base beacon transmitted by one of its remaining neighbors. If at least one extra receiver is awake before or at the time of the link disruption, a base beacon will be broadcasted at $T3$. This enables the transmitter to establish a new link with a minimum delay. However, if no other receiver is active before or at the moment of the link disconnection, extra latency for setting up a new link will be introduced. This waiting time can be very long, since nodes apply asynchronous duty cycles in the optimized RI-MAC.

Even if the original transmitter detects a new receiver as soon as the link terminates, extra latency can be introduced. This is because it may not win the medium during the

7. LATENCY ANALYSIS WITHOUT HANDOVER

channel contention. In order to carry out handover, at least two nodes are assumed to exist around a particular node. As a result, after a new receiver broadcasts a beacon, a collision on the data packets must occur in the first medium competition regardless of the network density. This necessitates a BW field to be additionally included in the next beacon. According to the BW information, each node will randomly select a backoff duration ranging from 0 to BW before the data transmission. In terms of the backoff values chosen by all the transmitters, three communication patterns may occur for the new link establishment. These communication patterns are summarized from the view of the original transmitter. As illustrated in Figure 7.1, it may happen that:

- (a) the original transmitter wins the channel;
- (b) one of the other transmitters wins the channel;
- (c) no one wins the channel due to a collision.

If the original transmitter successfully occupies the channel at the second contention, it can immediately proceed with its remaining data transmission. Therefore, case (a) introduces the minimum communication delay after a new receiver is discovered. For case (b), instead of the original transmitter, one of the other transmitters wins the channel. Therefore, the transmitter cannot resume sending its data packets until the nodes that win the channel complete their data transmission first. The occurrence probability of this case increases with the increment in the network density. As a result, it may happen that the original transmitter fails in seizing the medium at multiple competitions before it finally wins. Since this case may result in a large amount of time, it is the main drawback of the optimized RI-MAC in terms of the packet transmission latency when the handover mechanism is not applied. However, no one wins the channel in case (c). Instead, a collision occurs at the second medium contention. Consequently, the original transmitter can only contend for the medium after the collision. The probability that data packets are transmitted simultaneously is affected by the network density, the duty cycle, the node schedule as well as the backoff waiting interval.

In order to simplify the latency evaluation, we assume that the original transmitter will seize the medium at latest at the third attempt. Hence, before it wins the channel, either case (b) or case (c) will occur but they can only occur once. This indicates that the time evaluated for setting up a new link here is much less than the average value.

7.2 Latency for a New Link Establishment

For different network densities, the number of nodes being awake is different. Since all the active nodes will participate in contending for the channel, different number of active nodes will result in different backoff durations. This further influences the setting up time of a new link. Therefore, the latency required for the remaining data transmission after the link disconnection, $t_3^{without}$, should be quantified as:

$$t_3^{without} = \frac{\sum_{k=1}^{M-1} (C_{M-1}^k D^k (1-D)^{M-1-k}) t_3^{without.k}}{\sum_{k=1}^{M-1} C_{M-1}^k D^k (1-D)^{M-1-k}} \quad (7.7)$$

In Equation (7.7), the term $t_3^{without.k}$ represents the average latency for setting up a new link for each k by taking all the three possible transmission patterns into account. From the perspective of the original transmitter, $t_3^{without.k}$ can be expressed as:

$$t_3^{without.k} = \frac{p_1^{col} p_2^{s1} t_a + p_1^{col} p_2^{others} t_b + p_1^{col} p_2^{col} t_c}{p_1^{col} p_2^{s1} + p_1^{col} p_2^{others} + p_1^{col} p_2^{col}} \quad (7.8)$$

Here, the four probabilities, p_1^{col} , p_2^{s1} , p_2^{others} , and p_2^{col} , that determine the values of t_a , t_b and t_c denote, in respective order, the probability that a collision occurs on the k data packets at the first transmission attempt, the probability that the original transmitter wins the medium, the probability that one of the other neighbors wins the medium, and the probability that a collision occurs in the second channel contention.

After the link breaks, the original transmitter will detect a new receiver sooner or later by receiving a base beacon. Since no BW field is embedded in the base beacon, a collision will definitely occur at the first transmission attempt. This makes the probability, p_1^{col} , equal to one.

$$p_1^{col} = 1 \quad (7.9)$$

Once the new receiver detects the collision, it will add a BW field in its next beacon. According to this backoff information, each of the k active nodes will select a waiting value at the second transmission attempt. Depending on the values they select, one of the three communication patterns will occur. Therefore, the probabilities p_2^{s1} , p_2^{others} , and p_2^{col} are related with k and can be quantified, in respective order, as:

$$p_2^{s1} = \frac{\sum_{m=1}^{BW1-1} (BW1-m)^k}{BW1^{k+1}} \quad (7.10)$$

7. LATENCY ANALYSIS WITHOUT HANDOVER

$$p_2^{others} = \frac{\sum_{m=1}^{BW1-1} C_k^1 (BW1 - m)^k}{BW1^{k+1}} \quad (7.11)$$

And

$$p_2^{col} = \frac{\sum_{u=1}^{BW1-1} (\sum_{m=2}^{k+1} C_{k+1}^m (BW1 - u)^{k+1-m}) + 1}{BW1^{k+1}} \quad (7.12)$$

By inserting Equations (7.9)-(7.12) into Equation (7.8), the parameter $t_3^{without}$ can be expressed as:

$$\begin{aligned} t_3^{without.k} &= \frac{\sum_{m=1}^{BW1-1} (BW1 - m)^k}{BW1^{k+1}} t_a + \frac{\sum_{m=1}^{BW1-1} C_k^1 (BW1 - m)^k}{BW1^{k+1}} t_b \\ &+ \frac{\sum_{u=1}^{BW1-1} (\sum_{m=2}^{k+1} C_{k+1}^m (BW1 - u)^{k+1-m}) + 1}{BW1^{k+1}} t_c \end{aligned} \quad (7.13)$$

The time periods t_a , t_b and t_c denote, in respective order, the latency that is required for setting up a new link, during which $S1$ wins the medium, one of the other nodes wins the medium, and no one wins the medium due to a collision in the second transmission attempt. As displayed in Figure 7.1, t_a , t_b and t_c can be expressed as:

$$t_a = T_{CCA} + \frac{2N_b}{R_t} + \frac{N_{data}}{R_t} + T_r + T_{backoff1} + n_3^{without} T_{unit} \quad (7.14)$$

$$t_b = T_{CCA} + \frac{2N_b}{R_t} + \frac{N_{data}}{R_t} + T_r + 2T_{backoff1} + (n + n_3^{without}) T_{unit} - T_{SIFS} \quad (7.15)$$

$$t_c = T_{CCA} + \frac{3N_b}{R_t} + \frac{2N_{data}}{R_t} + 2T_r + T_{backoff1}^{col} + T_{backoff2} + n_3^{without} T_{unit} \quad (7.16)$$

The parameters $T_{backoffm}$ and $T_{backoff1}^{col}$ denote the back-off interval determined by $(k+1)$ nodes out of which one wins the channel in the $(m+1)_{th}$ contention ($m = 1, 2$), and the back-off interval determined by a collision in the second contention, respectively. These two parameters can be further expressed as:

$$T_{backoffm} = \frac{\sum_{u=1}^{BWm-1} (BWm - u)^k u \sigma}{\sum_{u=1}^{BWm-1} (BWm - u)^k} \quad (m = 1, 2) \quad (7.17)$$

7.2 Latency for a New Link Establishment

$$T_{backoff1}^{col} = \frac{\sum_{u=1}^{BW1-1} \left(\left(\sum_{m=2}^{k+1} C_{k+1}^m (BW1 - u)^{k+1-m} \right) u \sigma \right) + BW1\sigma}{\sum_{u=1}^{BW1-1} \left(\sum_{m=2}^{k+1} C_{k+1}^m (BW1 - u)^{k+1-m} \right) + 1} \quad (7.18)$$

By combining Equations (7.13)-(7.18) with Equation (7.7), the time needed to set up a new link after the original link breaks can be obtained. Finally, by combining Equations (7.1), (7.2) and (7.7), the time required for transmitting n data packets without the handover support can be evaluated.

7. LATENCY ANALYSIS WITHOUT HANDOVER

8

Analytical Result

8.1 Parameter Setup

We employ Matlab (version 7.0.1) to visualize the packet transmission latency that is quantified in the mathematical model when the handover mechanism is used and when it is not used in the optimized RI-MAC. Especially, the effect of the duty cycle and the network density on the latency is investigated.

The analytical model is applied to the network with a moderate density, in which nodes have asynchronous duty cycles and are uniformly distributed. We define the number of data packets transmitted in burst to be 140 and the average moving speed of human beings to be $1.5m/s$. Consequently, if a transmitter wins the medium after it moves out of the radio transmission range of its receiver with the radius $24.5m$, the link established between them will break in the midst of their data communication. Therefore, this $24.5m$, which is $0.5m$ away from the maximum radio transmission range of a node, is used as the distance threshold for the initiation of the handover mechanism.

We also assume that the mobile wireless sensor network operates under ideal channel conditions and the data packets are sent at the rate of $250Kbps$. Since the burst transmission pattern is utilized in the optimized RI-MAC once the handover mechanism is triggered, a node should be able to produce the sensed event information at a rate greater than or equal to $250Kbps$. All the other parameters used for the latency evaluation are presented in Table 8.1.

8. ANALYTICAL RESULT

Table 8.1: parameter list

Basic Parameter	Default Value
Beacon	18bytes
Data packet	45bytes
Packets transmitted in burst	140
Transmission rate	250Kbps
Maximum handover attempt	4
Backoff window	31, 63, 127, 255
Maximum waiting interval	31
Distance threshold	24.5m
Nominal transmission range	25m
Average moving speed	1.5m/s
SIFS	192 μ s
Slot time	320 μ s

8.2 Analytical Result and Discussion

8.2.1 Latency with Handover

8.2.1.1 Latency before $T1$

In the mathematical model, the time consumed before $T1$ is the same when the optimized RI-MAC uses the handover mechanism and when it does not. This latency is caused by the initial medium contention and the following communication of the w data packets between a pair of nodes. When the handover mechanism is applied, these w data packets enable the transmitter to obtain the same number of ACK beacons, based on which the transmitter can estimate the distance with respect to its receiver and evaluate the quality of the established link.

Figure 8.1 illustrates the latency introduced before $T1$. With the increment of both the duty cycle and the network density, the latency decreases. This decrease is attributed to the backoff interval a node has to wait before the data communication. For a fixed network density larger than two, the number of neighbors of a particular node that are awake at the same time increases when the duty cycle grows. This enables more neighbors of the receiver to contend for the channel. Since each medium competitor is required to choose a random value between 0 and BW once receiving a beacon, the more

Figure 8.1: Latency before $T1$ in the cases with and without the handover support

the nodes participate in the contention, the smaller the average backoff interval will be. However, when the network density is two, the latency consumed before $T1$ remains unchanged regardless of the duty cycle. This is because the RI-MAC formulates the rule that when a beacon without a BW field is broadcasted, collision will definitely occur if more than one transmitter with pending data are awake. To embody the feature of the base beacon, we assume that at least two nodes take part in the channel contention, even though both of them have a low duty cycle. No matter which one finally wins the medium, the backoff interval remains the same, leading to a latency of $0.0139s$. This is also the reason why the time cost before $T1$ is equal for all the network densities once the duty cycle approaches zero.

Similarly, for a fixed duty cycle, the increment of the network density results in a larger quantity of nodes that are involved in the channel occupation, causing a smaller backoff value. This, in turn, shortens the time that the winner transmitter idle listens before it starts with its own data transmission. Especially, when the duty cycle reaches one, all the neighbors of the receiver are active, and therefore, will all participate in the medium contention. This results in the shortest communication latency which is equal

8. ANALYTICAL RESULT

to $0.0131s$, $0.0126s$ and $0.0123s$ for the network densities $M = 3$, $M = 4$ and $M = 5$, respectively.

8.2.1.2 Latency between $T1$ and $T2$

Figure 8.2: Handover latency for $M = 2$

Once the transmitter detects that the quality of the current link is deteriorating, it will initiate the handover mechanism. How quickly a new link can be discovered depends on three factors. That is whether at least a neighbor node is awake at the moment, whether the node has a distance to the transmitter not exceed d , and whether collision occurs on handover reply beacons if the first two conditions are satisfied.

When the network density is two, except the original receiver, the transmitter only has one neighbor node which may be able to send back a handover reply beacon. Even though collision is completely avoided in this situation, the handover latency can be large due to the first two restrictive conditions. As displayed in Figure 8.2, the handover latency is inversely proportional to the duty cycle and it is bounded within $0.026s$ when the duty cycle is close to 0. This happens because the handover mechanism is assumed to be successful at latest at the i_{th} attempt. Here, i is set to 4. Once the duty cycle

increases to one, the only receiver will keep its radio on all the time. Consequently, the failure of a handover attempt can be only attributed to the large distance between it and the transmitter. On average, the handover latency is 0.008s when the node has a duty cycle of 100%.

Figure 8.3: Handover latency for $M \geq 2$

However, when the network density is larger than two, collision may occur on the handover reply beacons. The correlation among the three factors leads to a more complicated evaluation situation. Figure 8.3 presents the average handover latency under different network densities for $M > 2$. When the network density increases, both the probability that a node whose relative distance is less than d and the probability that a collision occurs among the simultaneously transmitted handover reply beacons increase. But the first (which decreases the handover latency) increases faster than the second (which increases the handover latency). This is because the number of nodes whose relative distance does not exceed d is a necessary condition for the occurrence of a collision between beacons. Since distance is only one of the parameters that determine the collision probability, the handover latency becomes smaller as the network density and the duty cycle increase.

8. ANALYTICAL RESULT

When the duty cycle approaches zero, the handover latency remains unchanged under different network densities for $M > 2$. This is because when the duty cycle is extremely small, the probability that k out of $M - 1$ nodes are awake at the same time is very low. Since the l nodes, which determine the handover duration are selected from the k nodes, l has even a smaller value, and thus, can be regarded identical for all the network densities. Once the duty cycle reaches one, all the neighbors of the transmitter are in an active state. As a result, the failure of a handover attempt can only be caused by the overestimated distance and the collision of beacons. This results in a higher probability that a handover attempt succeeds and a shorter interval for searching for a new relay node. As shown in Figure 8.3, when the duty cycle increases to one, the handover latency decreases to $0.0067s$, $0.0061s$ and $0.0057s$ for the network densities $M = 3$, $M = 4$, and $M = 5$, respectively.

Figure 8.4: Average handover latency under different network densities

In order to give an overall picture of the handover latency under different network densities and duty cycles, we combine the above two figures together. As can be observed in Figure 8.4, there is a big disparity between the results of $M = 2$ and $M > 2$. This happens because different methods are adopted for evaluating the average han-

doer duration for the two cases. For $M = 2$, collision is not taken into the account of the latency evaluation. However, for $M > 2$, a few simplifications make the mathematical expression of the handover latency tractable. Particularly, if a node is awake in the first handover attempt, it is assumed to be also awake during the following handover attempts. Similarly, if a node's relative distance is less than d in the first handover attempt, it is assumed to be also less than d in the following handover attempts. Therefore, multiple cases that can lead to a handover failure are not considered. That is why the probability that a handover succeeds in each attempt highly increases when $M > 2$, leading to a much less handover latency compared to the case when $M = 2$.

8.2.1.3 Total latency

Figure 8.5: Total latency for transmitting a burst of data packets with the handover support

The total latency required for the transmission of n data packets is the summation of the time consumed before T_1 , between T_1 and T_2 , and after T_2 . As illustrated in Figure 8.5, the total communication latency is inversely proportional to both the duty cycle and the network density. When the duty cycle approaches zero, it is close to 0.353s

8. ANALYTICAL RESULT

and $0.367s$ for the network densities $M > 2$ and $M = 2$, respectively. However, when the duty cycle increases to one, the total latency declines to $0.354s$, $0.351s$, $0.35s$ and $0.349s$ for the network densities $M = 2$, $M = 3$, $M = 4$ and $M = 5$, in a respective order.

8.2.2 Latency without Handover

8.2.2.1 Latency before T_3

Figure 8.6: Latency before the link breaks when the handover mechanism is not applied

If a transmitter that is initially located out of the radio transmission range of the receiver (with the radius d) wins the channel, the established link will break during the communication. Without the utilization of the handover mechanism, the transmitter can do nothing but idle listens until it receives a beacon from a new neighbor. Only from then on, can the transmitter attempt to resume transmitting the remaining data. A long delay will be introduced during this new link establishment.

As stated before, the time consumed before T_1 shares the same value regardless of the employment of the handover mechanism. Therefore, the difference in the total latency between the two scenarios (when the handover mechanism is employed and when

it is not) lies in the delay introduced after the first w data packets are transmitted. By also including the time duration for the data transmission between $T1$ and $T3$, the delay introduced before the link termination when the handover mechanism is not used can be evaluated. As displayed in Figure 8.6, the latency curve reflects a similar trend to that given in Figure 8.1. This phenomenon occurs because the transmitter is assumed to always move towards a specific direction at a uniform speed. Therefore, the time traveled during its movement presents a linear characteristic. As can be noticed, for a fixed duty cycle, the transmission latency before $T3$ is increased by $0.16667s$ compared with that before $T1$ for all the network densities.

8.2.2.2 Latency after $T3$

Figure 8.7: Latency required for setting up a new link

Figure 8.7 displays the time that the transmitter needs for its remaining data transmission after $T3$. Once the original link disrupts, the transmitter will wait for the appearance of a new receiver to try to establish a new link with it. With the decrease of the duty cycle and the network density, the number of nodes that are awake at the same time will be minimized. As all the active neighbors of the new receiver are allowed

8. ANALYTICAL RESULT

to participate in the channel contention, the probability that the original transmitter again wins the medium after the link termination becomes higher.

Since the new link establishment will not be completed until the last n th data packet is successfully sent, the communication pattern in which the original transmitter wins the medium at the very start brings the shortest latency for setting up a new link. The duty cycle equal to zero is the extreme case, in which only the original transmitter is awake. This makes the time consumed in the remaining data transmission minimum regardless of the network density ($0.347s$). However, when the duty cycle increases to one, all the neighbors of the new receiver will take part in the channel competition. Since a collision may occur and one of the other transmitters may occupy the medium for several rounds before the original transmitter finally wins, the latency required for the new link establishment will become longer. It is equal to $0.347s$, $0.396s$, $0.419s$ and $0.43s$ for the network densities $M = 2$, $M = 3$, $M = 4$, and $M = 5$, respectively.

8.2.2.3 Total latency

Figure 8.8: Total latency for transmitting a burst of data packets without the handover support

Figure 8.8 provides an overview of the time required for transmitting a burst of data packets when the handover mechanism is not used. The total latency is evaluated by combining the time intervals consumed prior to the link termination and used for the new link establishment. Obviously, the latency introduced before and after $T3$ has different variation tendency under the same duty cycle and network density. It increases in the former case, but decreases in the latter case with the increment in both the parameters. Since the transmitter may not immediately discover a new receiver or win the medium after it discoveries, it needs longer time to proceed with its remaining data transmission after the link breaks compared with the time required for the data transmission before the link breaks. The curve presents the total latency for the communication of n data packets has a similar exhibition with that shown in Figure 8.7. The latency changes from 0.528s to 0.609s when the duty cycle and the network density increase.

8.2.3 Latency Comparison

Figure 8.9: Latency comparison between the cases with and without the handover support

Figure 8.9 makes a comparison of the packet transmission latency in terms of the

8. ANALYTICAL RESULT

employment of the handover mechanism. It shows that the time introduced with the handover support is much less than that without the handover support. Even the longest latency when the handover mechanism is used is much smaller than the shortest latency when the handover mechanism is not used. The latency can be saved by at least 0.16s with the utilization of the handover mechanism.

Figure 8.10: The difference between the latency when the handover mechanism is used and when it is not used

Figure 8.10 displays the difference between the packet transmission latency that is introduced when the handover mechanism is used and when it is not used. The difference becomes minimum (0.16s) when a node in a mobile wireless sensor network has only two neighbors and both of them are equipped with very low duty cycles. The time disparity will further increment as both the duty cycle and the network density increase. For the network density $M = 5$, the latency difference reaches the maximum value (0.26s). This figure can even grow as the network density increases. From this perspective, the handover mechanism improves the performance of the optimized RI-MAC as far as mobility is concerned.

9

Simulation Result

9.1 Parameter Setup

To validate the mathematical result, the NS2 network simulator (version 2.29) is employed. It applies the standard combined free space and the two-ray ground reflection radio propagation model that is commonly used for simulation. Each sensor node is simulated with a single omni-directional antenna. To simplify the evaluation of the packet transmission latency, we do not include routing traffic in the simulation and assume there is a routing protocol providing the shortest path between any two nodes. We also ensure that no network used in our simulation is partitioned.

The key parameters that are used for simulating the radio of sensor nodes are the same as that used in the analytical model. Most of these parameters are taken from the data sheet of CC2420 radio, which is used in popular motes such as MICAz, TelosB and IMote (1). The RSSI sampling delay for CC2420 is $128\mu s$ as reported by Ye et al. (100). This delay is used as the time for a single CCA check in the optimized RI-MAC, i.e., the delay before the actual data transmission starts after a STXONCCA command is strobed (1). The transmission range and the carrier sensing range depend on many factors such as the transmission power, the antenna and the environment. In NS2, these two parameters are modeled after the 914MHz Lucent WaveLAN radio and are respectively set as $250m$ and $550m$, which however is not typical for a sensor node. To reflect the reality, we use $25m$ and $55m$ instead, since they are the values tested during the real experiment (24).

Once a collision is detected, the receiver will adjust the value for the BW field in the

9. SIMULATION RESULT

next beacon by applying a Binary Exponential Backoff (BEB) strategy (8). The initial backoff window and the congestion backoff window are defined as 32 and 8, respectively. They are the default values used in the UPMA package distributed with TinyOS. Since the retransmission count is limited to 5, the BW field in beacons can only take values of 0, 31, 63, 127 and 255 in the evaluation. The backoff duration the receiver has to wait before it broadcasts a new beacon is fixed to 31 time slots. Similarly, the value for the maximum waiting interval that is embedded in each handover request is defined as 32 time slots.

In the simulation, a normal beacon consists of a frame control field, a source address, a BW field and a frame check sequence. It acts as a base beacon if the backoff value defined in the BW field is zero. However, if the value is nonzero, the beacon becomes an ACK frame serving as an acknowledgement. A beacon can also turn into a handover reply by inserting a new Boolean field *handover* with the value *true* in the BW field. In addition, to avoid synchronized beacon transmissions from neighboring nodes, the initial wake-up time of each node is randomized in the optimized RI-MAC.

9.2 Topology Setting

NS (version 2) is an object-oriented, discrete event driven network simulator developed at UC Berkely. It is written in C++ and OTcl. The OTcl script enables to initiate an event scheduler, to set up a network topology and to tell traffic sources when to start and stop transmitting packets. These functions are realized by using the network objects and the plumbing functions in the library. However, the event scheduler and most of the network components are written in C++ and can be available to OTcl through an OTcl linkage that is implemented using tclcl.

```
for {set i 1} {$i <= $opt(nn)} {incr i} {  
    set node_($i) [$ns_ node]  
    set mac_($i) [$node_($i) getMac 0]  
    $god_ new_node $node_($i);  
    $node_($i) random-motion 0;  
}
```

Figure 9.1: Node generation

All the values of the parameters that are used for the latency evaluation in the

optimized RI-MAC are given in an OTcl script. Before building a wireless network topology, network objects such as nodes and links should be created. As Figure 9.1 shows, nn number of nodes are created and each of them is referenced with a MAC ID. To investigate the performance of the handover mechanism, an established link between two nodes is required to break during the data communication when it is not used. Therefore, we disable the random motion function of nodes and control the initial location of the mobile node.

```
$ns_ node-config -adhocRouting AODV \
    -macType "Mac/RI-MAC" \
    -llType LL \
    -ifqType Queue/DropTail/PriQueue \
    -ifqLen 1000 \
    -antType Antenna/OmniAntenna \
    -propType Propagation/TwoRayGround \
    -phyType Phy/WirelessPhy \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channelType Channel/WirelessChannel;
```

Figure 9.2: Node configuration

```
set udp1 [new Agent/UDP]
$udp1 set packetSize_ 12;
$ns_ attach-agent $node_(1) $udp1
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 12;
$cbr1 set random_ 1
$cbr1 set interval_ 0.001
$cbr1 set maxpkts_ 2684354560
$cbr1 attach-agent $udp1
$ns_ at 0.02 "$cbr1 start"
$ns_ at 60 "$cbr1 stop"
```

Figure 9.3: An UDP agent is attached to a node if it is a transmitter

Once a node is created, its properties will be configured. As displayed in Figure 9.2, the channel type, the network interface type, the link layer type, the interface queue type and the antenna and propagation models are all deployed. Different traffic agents

9. SIMULATION RESULT

will be attached to the node when it plays different roles. If it is a transmitter, it will be glued to an UDP agent, as described in Figure 9.3. This agent is linked to a traffic source that generates data packets at a Constant Bit Rate (CBR). To enable the burst transmission, the data arrival interval should be set smaller than the data transmission interval. Aiming at approaching the experimental result, noise is not avoided and it is randomized during the packet communication. However, if the node is a receiver, a sink agent will be generated and attached to it, as shown in Figure 9.4. By configuring the destination address to -1 and the destination port number to 0 , a packet will be transmitted in a broadcast way.

```
set sink1 [new Agent/MyNull]
$ns_ attach-agent $node_(3) $sink1
$ns_ connect $udp1 $sink1
```

Figure 9.4: A sink agent is attached to a node if it is a receiver

As mentioned in Section 8, the radio transmission range and the carrier sensing range are set as $25m$ and $55m$. These two values correspond to the receiving powers $6.35631e - 08w$ and $3.07645e - 07w$, respectively. In addition, the distance threshold $24.5m$ that initiates the handover mechanism can also be viewed from the perspective of the receiving power. It is can be transformed to $3.2033e - 07w$.

```
ns-random $opt(nsseed)
set u [new RNG]
$u seed $opt(nsseed)
set start [$u uniform $opt(start0) $opt(start1)]
```

Figure 9.5: Configuration of the seed to enable random simulation results

The handover mechanism can perform well even if all the nodes in a wireless sensor network are mobile. Nevertheless, only one node is set mobile in the simulation acting as a transmitter. This setting enables a clear observation of the trace file generated from the OTcl script. All the data used as a basis for the latency evaluation are extracted from the trace file. Besides the mobile transmitter, there are some other static nodes locating in the environment acting as transmitters or receivers. To justify handover, a node is required to have at least two neighbors. This indicates that once a beacon is broadcasted, more than one transmitter will contend for the medium. Since all the

nodes run the same program regardless of their states, the mobile transmitter may or may not seize the channel during the competition. This random selection of the winner node is realized by setting the variable *seed* to 0, as presented in Figure 9.5.

Figure 9.6: Topology of a mobile wireless sensor network

After combining all these configurations, a wireless sensor topology will be established. In fact, multiple network topologies can provide a mobile scenario which enables the evaluation of the performance of the handover mechanism. Figure 9.6 described here only gives a simple example. In this topology with the size 80×80 , receiver $R1$ begins the data communication by broadcasting a base beacon. Since all the transmitters are within the radio transmission range of $R1$, all of them are able to contend for the medium. However, to achieve our search purpose, only the situation in which the mobile transmitter wins the channel is considered useful. Besides $R1$, all the receivers are the neighbors of the mobile node. As a result, when the handover mechanism is applied, the mobile transmitter will detect one, some, or even all of them before the link between it and $R1$ terminates. However, when the handover mechanism is not applied, it cannot reach them until the original link deteriorates.

9. SIMULATION RESULT

9.3 Simulation Result and Discussion

9.3.1 Latency with Handover

After the network topology is built and the parameters are set in the OTcl script, the C++ programming language will be used to implement the optimized RI-MAC. For the sake of evaluating the performance of the handover mechanism, a handover reply beacon should be broadcasted before the link between a transmitter and a receiver breaks. Since this handover reply beacon can be transmitted by any of the receivers of the transmitter except the original one, the wake up time of all the receivers should be set earlier than the link termination time. Since nodes are independent of each other, each receiver should be assigned a unique variable representing its own radio-turn-on time. In the simulation in which the handover mechanism is employed, a frame length is divided into 100 partitions, each of which corresponds to a time point. To minimize the number of simulation runs, a node can only wake up at one of these time points. Nevertheless, there can be multiple combinations of the wake up time of the receivers, leading to numerous different simulation results even for one duty cycle.

When the network density is larger than two, the medium competition at the beginning of the commutation among the transmitters of a receiver will occur. Due to the introduction of the random seed, besides the mobile node, all the transmitters are likely to seize the channel. However, only the simulation in which the mobile transmitter wins is considered useful. Therefore, to obtain one useful result, on average M simulations should run for each duty cycle. Here, M denotes the network density, equaling to the total number of the transmitters of the receiver. Aiming at obtaining the latency result that is mostly close to the average value while limiting the number of simulation runs to an acceptable level, the program is executed 20 times under each duty cycle for all the network densities.

Each time after running the program, a trace file recording the events in chronological order will be generated. Based on it, the data sets that we are interested in can be extracted by using the programming language *gawk*. After the final data analysis, the latency of the packet transmission when the handover mechanism is applied can be evaluated (Appendix A.1 shows an example that exhaustively explains the process of the data processing).

9.3.1.1 Latency between $T1$ and $T2$

Figure 9.7: Average handover latency under different network densities

Once the transmitter detects the quality deterioration of the current link, it will trigger the handover mechanism. Only the neighbor nodes that have already entered into an active state and whose distance with respect to the transmitter does not exceed the threshold are allowed to participate in the handover process. However, different from the mathematical model, the factor of distance is not taken into account for the latency evaluation in the simulation. This is because it is rare that a receiver moves backwards and forwards through the distance threshold in reality. In other words, the probability that a receiver whose distance to the transmitter is smaller than d in the first handover attempt, larger than d in the second attempt, and smaller than d again in the third attempt is very low. In addition, since the distance threshold closely approaches the maximum radio transmission range and the network density is larger than one, the transmitter is probable to have at least one receiver (except the original receiver) within its radio circle with the radius d . Therefore, a handover failure cannot be attributed to a large distance in most of the cases. Without the limitation of distance, all the active

9. SIMULATION RESULT

receivers except the original one will attempt to broadcast a handover reply beacon once they receive a handover request.

Unlike the mathematical model in which the method for the latency evaluation is different when $M = 2$ and $M > 2$, the evaluation method is the same for all the network densities in the simulation. Therefore, no big disparity appears in Figure 9.7 when the network density grows from two to three. Furthermore, to mostly approach the experimental result, the limitation that the handover should succeed at lasted at the 4th attempt is relaxed in the simulation, resulting in longer handover latency on average.

As can be observed, the handover latency is inversely proportional to the duty cycle as well as the network density. When the duty cycle is extremely small, the handover latency is equal to 0.085s, 0.059s, 0.044s and 0.036s for the network densities $M = 2$, $M = 3$, $M = 4$, and $M = 5$, respectively. However, when the duty cycle increases to one, all the receivers are active before or at the moment of the transmission of the first handover request. Therefore, the handover process can accomplish at the very beginning. Since the number of medium competitors is much smaller than the number of time slots that constitute the WI field given in a handover request, collision will seldom occur. This is the reason why the handover latency for all the network densities converges at 0.008s when the duty cycle is complete.

9.3.1.2 Latency after T_2

As soon as the transmitter receives a handover reply beacon before the link breaks, the handover process is regarded successful. The transmitter will subsequently transmit its remaining data packets over the newly established link. The time required for the transmission mainly depends on the number of data packets that are left to be sent after the new relay node is found. It is known that w data packets, that are used for the distance evaluation, are sent before the handover mechanism is triggered. Since the total data packets that are transmitted in burst is a constant, the number of the remaining data packets that will be resumed transmitting over the new link is determined by the quantity of the neighbor discovery requests broadcasted during the handover process. This is the reason why the latency after T_2 exhibits a symmetrical characteristic to the handover latency.

Figure 9.8: Average latency after handover

With the increment of the network density, more receivers will take part in the handover process, and thus, the transmitter is able to receive a handover reply beacon at an earlier time. This shortens the handover duration but grows the latency for the remaining data communication. As shown in Figure 9.8, the latency after T_2 increases from $0.261s$ to $0.269s$ as the network density augments from 2 to 5 when the duty cycle is extremely small. However, once the duty cycle reaches one, all the receivers become active and can receive the handover request at the first attempt. Even though a collision on handover reply beacons may occur if the network density is larger than two, the probability of occurrence is low due to the comparatively large WI size and the limited number of medium competitors. Consequently, the latency after handover remains almost the same for all the network densities ($0.274s$).

9.3.1.3 Total latency

The total time consumed in transmitting a burst of data packets with the handover support is illustrated in Figure 9.9. This time is a combination of the duration introduced before handover, during handover, and after handover. The total latency required for

9. SIMULATION RESULT

Figure 9.9: Total latency for transmitting a burst of data packets with the handover support

the transmission of n data packets is inversely proportional to both the duty cycle and the network density.

When the duty cycle approaches zero, it is close to $0.36s$, $0.337s$, $0.324s$ and $0.318s$ for the network densities $M = 2$, $M = 3$, $M = 4$ and $M = 5$, respectively. As can be observed, for the network density $M = 2$, the packet transmission latency in the simulation has almost the same value as that in the analytical model. The difference is merely $0.005s$. However, with the increment in the network density, this difference increases. Especially, when $M = 5$, the total latency is much smaller in the simulation than that in the analytical result, even though the handover latency has a larger value in the simulation than that in the analytical model.

This phenomenon can be attributed to several reasons. Foremost, the handover latency evaluated in the mathematical model is not remarkably accurate. To make the evaluation tractable, several cases that can cause a handover failure are not taken into account, leading to an underestimation of the handover latency in the analytical result. Secondly, rather than occurring only once in the mathematical model, collision

between the handover reply beacons may happen several times in the simulation. The collision frequency actually depends on the number of contending receivers as well as the backoff intervals chosen by them. Thirdly, due to the limited number of simulation runs, the backoff duration before the packet communication can be sharply deviated from the average value calculated in the analytical model. Finally, the number of data packets transmitted during the handover process may be a decimal in the mathematical model, since it is only an average value evaluated from a probabilistic viewpoint. In the contrary, this value obtained from the simulation must be an integer having a definite physical meaning.

When the duty cycle increases to one, the handover is probable to succeed in the first attempt regardless of the number of receivers. This results in almost the identical value of the total packet transmission latency for all the network densities.

9.3.2 Latency without Handover

In order to effectively compare the performance of the optimized RI-MAC in the situations with and without the handover support, the network topology is configured the same for both cases. Similar to that when the handover mechanism is applied, the duty cycle is divided into one hundred partitions for the evaluation of the packet latency when the handover mechanism is not applied. Aiming at justifying handover, the network density is required to be larger than one. To accord with this rule, a receiver should at least have two transmitters. Since the protocol formulates that a collision on the data packets will occur once a base beacon is broadcasted, the two transmitters are further required to be in an active state at the beginning of the communication, even if they are equipped with a low duty cycle.

To focus on our study goal, only the simulations in which the mobile node seizes the medium are regarded valid. However, as the network density increases, the probability that the mobile transmitter wins the channel becomes small. For the sake of evaluating the communication latency based on the same number of the valid samples, the running times of the simulation should be different under different network densities. To this end, we let the program runs $(100 \times M)$ times for the network density equal to M , among which approximately 100 simulation results are useful. The latency for each duty cycle is obtained by averaging the values perceived from all these valid simulation samples.

9. SIMULATION RESULT

Likewise, a trace file will be generated after each simulation when the handover mechanism is not utilized. According to the raw data recorded in each trace file, the useful information that will be used as a basis for the evaluation of the transmission latency can be extracted (Appendix A.2 shows a trace file and its generation in the case without the handover support).

9.3.2.1 Latency after T_3

Figure 9.10: Latency for the new link establishment

The latency for the remaining data transmission after the link disruption shown in the simulation figure exhibits a similar variation trend with that described in the analytical figure, but with more fluctuations. Different from that in the mathematical model, the original transmitter is allowed to win the medium at any competition after the link breaks in the simulation. In other words, the limitations that the original transmitter should win the channel at latest at the third contention and only 140 data packets can be transmitted before it wins are relaxed. Without these restrictions, the latency for setting up a new link will become much larger in the simulation than that in the mathematical model under the same duty cycle and network density. An extreme

case may occur even with a minor probability. In the case, the original transmitter never wins the medium after the link disrupts. Instead, either a collision occurs or one of the other transmitters seizes the channel at each contention.

In fact, the curve fluctuation is a result of the finite simulation runs and the multiple communication patterns that a node can select after the link breaks. The channel competition in which the original transmitter wins and the number of collisions and contention failures occurred before it wins are randomly chosen by the simulation runs. Due to the complete randomness, the latency required for setting up a new link under a small duty cycle can be even greater than that under a big duty cycle, as can be observed in Figure 9.10. Similar phenomenon can appear in terms of the network density. Nevertheless, on the whole, the latency for the new link establishment increases with the increment in both the duty cycle and the network density.

9.3.2.2 Total latency

Figure 9.11: Total latency for transmitting a burst of data packets without the handover support

By including both the time duration spent on the data communication before and

9. SIMULATION RESULT

after the link termination, the total latency consumed in the burst packet transmission can be visualized. As presented in Figure 9.11, the total latency when the handover mechanism is not applied is directly proportional to the duty cycle as well as the network density.

In both Figures 9.10 and 9.11, there is a short straight line appearing at the bottom left corner, showing that the latency stays small and remains the same for all the network densities. This tells that when the duty cycle is less than 5%, all the transmitters expect the original one have already entered into the sleep mode before they can receive a beacon, even though they may have just woken up and obtained a data packet from the upper layer. In the simulation, it is pretty difficult to align the start time of the active period of a node with the time at which the communication begins when the duty cycle is extremely small. As a result, the latency 0.35s is consumed when only the mobile transmitter is active and participates in the data communication.

9.3.3 Latency Comparison

Figure 9.12: Total latency comparison between the cases with and without the handover support

Figure 9.12 gives an overall comparison between the total packet transmission latency introduced when the handover mechanism is used and when it is not used. It shows that the time needed for transmitting a burst of data packets with the handover support is much less than that without the handover support. This happens because the time required for setting up a new link is much longer than the time required for searching for a new receiver during the handover process.

Meanwhile, the latency curves in the situation where the handover mechanism is applied have a smaller fluctuation than that in the situation in which the handover mechanism is not applied. This phenomenon can be attributed to the different mechanisms that the optimized RI-MAC adopts during the handover process and during the new link establishment. In the former case, all the simulation results are almost identical. The only two differences lie in the backoff value that a receiver chooses before the transmission of the handover reply beacon and the attempt at which the handover succeeds. However, in the latter case, the simulation results can be quite different from each other. This is mainly because the three communication patterns after the link termination consume absolutely unequal time duration. If the original transmitter wins the medium at the first contention, it will complete its remaining data transmission with a minimum delay. However, if it wins after multiple channel competitions, a large amount of latency will be introduced. As a result, the range the total latency can vary when the handover mechanism is used is much smaller than that when the handover mechanism is not used.

As can be seen, the difference of the total latency is minimum (0.28s) when a node in a wireless sensor network has only two neighbors and each of them is equipped with a very low duty cycle. It reaches the maximum value (1.35s) when the network density $M = 5$. The difference can further grow with the increment in the duty cycle and the network density. According to the comparison between the packet transmission latency in the cases with and without the handover support given in both the mathematical and simulation results, we can draw the conclusion that the handover mechanism improves the performance of the optimized RI-MAC in a mobile scenario.

9. SIMULATION RESULT

10

Conclusion

10.1 Mobility Challenges

This thesis aims at investigating the latency of packet transmission in a mobile wireless sensor network with and without the support of a handover mechanism. To start with, we elaborate the challenges of mobility on the performance of the existing MAC protocols at link layer; and the need to maintain an unbroken link during data transmission. Mobility can lead to deterioration in the quality of an established link, and therefore, data transmission is prone to failure, which in turn increases the rate of packet retransmission. Mobility can also cause frequent route changes, resulting in a considerable packet delivery delay. In addition, a mobile node cannot immediately begin transmitting data upon joining a network, because it has to wait for a certain amount of time before it can be fully integrated. In contention-based MAC protocols, mobility may increase packet collision while in schedule-based MAC protocols, two-hop neighborhood information may become inconsistent once nodes enter or leave a network, leading to schedule inconsistencies.

10.2 Options for Link Deterioration

Since most of the proposed Medium Access Control (MAC) protocols do not accommodate mobility, a node has two options to deal with a deteriorating link: (a) to continue data transmission until the link breaks and then establishes a new link with a new relay node; or (b) to seamlessly transfer the communication to a better link parallel to the data transmission over the existing link. Different from option (a) where a node can

10. CONCLUSION

only search for a new link after the original link disrupts, option (b) enables a node to perceive the change in the quality of a link in advance. The link quality prediction can be realized by an adaptive handover mechanism. Both approaches will inevitably introduce latency. This thesis aims to quantify and compare such latency.

10.3 RI-MAC and its Demerits

To implement handover, a target MAC protocol is required to be selected first. Since the Receiver-Initiated MAC protocol (RI-MAC) uses only short beacon and data packets during communication that substantially reduces overhearing, collision probability and data recovery cost, it is chosen to as the carrier for the latency evaluation. In RI-MAC, a receiver initiates a communication by broadcasting a beacon once it completes its sleep period. This beacon contains only the source address information (SRC) which announces that the node is ready to receive data packets. If two or more transmitters send packets simultaneously, a collision will occur, leading to the transmission of another beacon which additionally contains a back-off window field (BW). If a transmitter wins the channel, the receiver will reply with an ACK beacon (DST) serving as the acknowledgement. However, if no packet is received within a specified dwell time, the receiver will go to sleep.

Even though RI-MAC performs well in many aspects, it has some demerits. Firstly, in a round of transmission, the BW size in beacons never decreases. Instead, it either remains unchanged or keeps increasing whenever a collision is detected. The increase can be fast in case of small transmission intervals, leading to a large back-off window. Secondly, a receiver is unable to receive any data packet during a dwell time may also because of an unsuccessfully transmitted beacon instead of an idle channel. Consequently, it may happen that a receiver has to go to sleep although there are transmitters wishing to communicate with it.

10.4 Optimization of RI-MAC

To slow down the monotonous increment in the backoff window size and to reduce the occurrence probability of the dwell time, a burst data transmission pattern is adopted to optimize RI-MAC. Instead of contending for a medium to send just a single packet, a

node transmits a pre-defined number of packets in burst. Therefore, the beacon between two data packets is only set as acknowledgement. By intercepting the ACK beacons, competing neighbors will realize that the medium is currently occupied and will refrain from attempting to seize the channel. With this optimization, the protocol reduces the long idle listening time that a node has to wait before data transmission, and thus, works well in a static scenario.

However, in case of mobility, the burst transmission does not perform well. Since a large number of data packets are sent in burst, the duration a node occupies the medium each time becomes longer. This increases the probability of link disconnection and the cost of data retransmission. To resume transmitting the remaining data after the link disrupts, the transmitter will search for a new link. However, it may not immediately establish a new connection due to the utilization of the asynchronous duty cycles in RI-MAC. Instead, the node has to wait for a long period before it can newly contend for the medium, resulting in a considerable packet delivery delay.

10.5 Handover Mechanism

For the sake of accommodating mobility, an adaptive handover mechanism is developed on top of the optimized RI-MAC. We predefine a distance threshold, d , to begin the handover. After the initial back-off and a data transmission, the transmitter will receive an ACK beacon from the intended receiver. Based on a number of ACK beacons, the transmitter can deduce the relative distance between it and the receiver. If the distance is estimated to be larger than d , the transmitter will immediately broadcast a data packet in which a handover request is embedded.

Upon receiving the request, the original receiver will reply with an ACK beacon. However, the remaining neighbors of the transmitter will send back handover reply beacons, but not all of them are eligible to participate. Instead, only those active nodes whose distance with respect to the transmitter does not exceed d are qualified. This ensures that the handover mechanism will be triggered at most once during a node's data transmission.

To avoid collision, a constant waiting interval (WI) field is included in each data packet indicating the maximum time that the transmitter waits for a handover reply. Each qualified receiver will randomly choose a backoff value according to the WI

10. CONCLUSION

field. The one that selects the smallest value will win the channel and become the new receiver. As soon as a new link is established, the transmitter will transfer the communication to the newly found node. After overhearing a data packet between them, the original receiver will enter into a sleep state.

By taking advantage of a Boolean field in a packet, the transmission mode can be controlled to either broadcast or unicast. The Boolean field with the value *true* signals that the data packet is a handover request. The Boolean field is also employed in every handover reply beacon to avoid collision.

With the development of the adaptive handover mechanism, the optimized RI-MAC successfully overcomes the challenges of mobility with the following achievements:

- **The handover mechanism enables to transmit data packets with a minimum delay.** Once the distance between a transmitter and a receiver is estimated to be larger than the threshold, a handover request will be embedded in each data packet. This indicates that a sensed event will be broadcasted together with a neighbor discovery request. Since the transmitter needs a period for detecting a qualified receiver, the data transmission rate in the case of mobility cannot be as large as that in the static scenario. However, to mostly reduce the duration for transmitting a data packet, the handover mechanism defines the WI field to limit the time that the transmitter has to wait for a handover reply beacon. If the transmitter does not receive any response until WI expires, it will broadcast another packet. In this way, the handover mechanism enables a node to transmit data packets at a maximum possible rate without affecting the neighbor discovery.
- **The handover mechanism is triggered only when necessary. Hence, it does not influence the performance of a MAC protocol in a static setting.** This is realized by the Boolean field included in each data packet. When the transmitter evaluates that the distance with respect to its receiver does not exceed the threshold, it will set the Boolean field of a data packet to *false*. All the neighbors after receiving the data packet, will learn that the transmitter is not looking for a new receiver. Therefore, only the original receiver will reply with an ACK beacon. However, when the quality of a link deteriorates, the Boolean field will be set to *true*. Based on the Boolean value along with the destination

information embedded in the data packet, the neighbors of the transmitter will be clear of their roles and all the qualified receivers will give feedbacks accordingly.

- **Data transmission does not significantly interfere with neighbor discovery and vice versa.** This is because the nodes that take responsibility for the data reception and for the handover reply are different. When a data packet including a neighbor discovery request is broadcasted, only the original receiver will acknowledge with an ACK beacon. This ACK beacon is necessary, since it ensures that the packet transmission continues whether the handover attempt is successful or not. Likewise, only the active neighbors whose distance to the transmitter does not exceed the threshold are allowed to participate in the new link establishment. The original receiver cannot take part in the handover process, even though the transmitter may not receive any handover response.

10.6 Mathematical Model Establishment

After qualitatively analyzing the effect of mobility on latency, a mathematical model is established to quantitatively investigate the latency of packet transmission during mobility with and without the support of the handover mechanism. To provide an accurate expression, the latency of the handover mechanism is evaluated differently for different network densities, M . To justify handover, the network density should be at least two. As one of the nodes is the original receiver, only one neighbor of the transmitter will participate in the handover process when $M = 2$. In this case, collision will not occur on handover reply beacons. Nevertheless, a handover may still fail because (a) the node does not receive the handover request since it is in a sleep state, or (b) the distance it estimates is larger than d . However, when $M > 2$, a handover failure can also be caused by (c) the collision on handover reply beacons. For the sake of reducing the evaluation complexity caused by the correlation among the three elements, we assume that k out of $M - 1$ neighbors of the transmitter are awake and l out of k nodes are enclosed within the radio transmission range of the transmitter (this range is now equal to d) for every handover attempt.

However, when the handover mechanism is not used, an established link will break in the midst of communication. Once the link disrupts, the transmitter will search for a new receiver. If at least one extra receiver is awake before or at the moment of the link

10. CONCLUSION

termination, the transmitter will establish a new link with a minimum delay. However, the time for the new link establishment can be very different depending on when the transmitter is able to win the channel. Since at least two nodes are assumed to be present around a particular node in order to carry out handover, a collision on data packets must occur in the first medium contention. This necessitates a BW field to be additionally included in the next beacon. As a result, in the subsequent contentions, it may happen that (a) the original transmitter wins the channel, (b) one of the other transmitters wins the channel, or (c) no one wins the channel due to a collision. For case (a), the transmitter can directly resume its remaining data transmission. For case (b), the transmitter cannot start sending until the node which wins the channel completes its data transmission first. For case (c), the transmitter can only contend for the medium after the collision. To make the latency evaluation tractable, we assume that the original transmitter seizes the medium at latest at the third attempt.

10.7 Mathematics-based Evaluation

The analytical result shows that the communication latency decreases with an increment in the network density and the duty cycle when the handover mechanism is used, whereas the latency exhibits a reverse trend when the handover mechanism is not used.

Particularly, the handover latency is inversely proportional to both the duty cycle and the network density. For $M = 2$, when the duty cycle approaches zero, the handover latency is equal to 0.026s. This is because at least one relay node is assumed to be discovered before the link breaks and the handover is assumed to be successful at latest at the fourth attempt. When the duty cycle increases to one, the handover latency decreases to 0.008s. This 0.008s is entirely attributed to the distance estimated to be larger than the threshold. For $M > 2$, as the network density grows, the number of nodes whose distance to the transmitter is less than d increases. Meanwhile, the probability that a collision occurs among the handover reply beacons increases. But the first (which decreases the handover latency) increases faster than the second (which increases the handover latency). This is because the number of nodes whose relative distance does not exceed d is a necessary condition for the occurrence of collision between beacons. Therefore, the handover latency becomes smaller as the network density increases. When the duty cycle approaches zero, the handover latency remains

unchanged under different network densities. This is because when the duty cycle is extremely small, the probability that k out of $M - 1$ nodes are awake at the same time is very low. Since the l nodes, which determine the handover duration are selected from the k nodes, l has even a smaller value, and thus, can be regarded identical for all the network densities. Once the duty cycle reaches one, a handover failure can be caused by both large distance and collisions. This results in a higher probability of handover success and a shorter handover interval. There is a big disparity between the results of $M = 2$ and $M > 2$. This appears because different methods are adopted to evaluate the handover latency for the two situations. Multiple cases that can lead to a handover failure are not taken into account when $M > 2$, leading to a much less handover latency compared to the situation when $M = 2$.

When the handover mechanism is not used, as the duty cycle and the network density decrease, the number of nodes that are able to contend for the medium becomes small, leading to a shorter time for setting up a new link. The duty cycle equal to zero is the extreme case, in which only the original transmitter is awake. This makes the time consumed in the remaining data transmission minimum regardless of the network density ($0.347s$). However, when the duty cycle increases to one, all the neighbors of the new receiver will take part in the channel competition. Since the original transmitter may fail in seizing the medium in multiple contentions before it finally wins, the latency for the new link establishment can be long.

The comparison shows that the time introduced with the handover support is much less than that without the handover support. The difference is minimum ($0.16s$) when a node has only two neighbors and each of them is equipped with a low duty cycle. The difference can further grow with the increment in the duty cycle and the network density.

10.8 Simulation-based Evaluation

The NS2 network simulator is employed in the end to validate the mathematics-based evaluation. Once the transmitter detects the link deterioration, it will trigger the handover mechanism. However, different from the mathematical model, the factor of distance is not taken into account for the latency evaluation in the simulation. This is because it is rare that a receiver moves backwards and forwards through the distance

10. CONCLUSION

threshold in reality. In addition, since the distance threshold closely approaches the maximum radio transmission range and the network density is larger than one, the transmitter is probable to have at least two receivers within its radio circle with the radius d . Therefore, a handover failure cannot be attributed to a large distance in most of the cases. Moreover, unlike the mathematical model in which the method for the evaluation of the handover latency is different for $M = 2$ and $M > 2$, the evaluation method is the same for all the network densities in the simulation. Therefore, no big disparity appears when the network density grows from two to three. Furthermore, the limitation that the handover should succeed at lasted at the fourth attempt is relaxed, resulting in longer handover latency in the simulation.

The handover latency is inversely proportional to the duty cycle as well as the network density. When the duty cycle increases to one, the handover process can accomplish at the very beginning. Since the number of medium competitors is much less than the number of time slots that constitute the WI field given in a handover request, collision will seldom occur. This is the reason why the handover latency for all the network densities converges at $0.008s$ when the duty cycle is complete.

The total time consumed in transmitting a burst of data packets with the handover support is inversely proportional to both the duty cycle and the network density. When the duty cycle approaches zero, the packet transmission latency in the simulation has almost the same value as that in the analytical model for $M = 2$. The difference is merely $0.005s$. However, with the increment in the network density, the total latency is much smaller in the simulation than that in the analytical result, even though the handover latency has a larger value in the simulation. This phenomenon is caused by several reasons. Foremost, the handover latency evaluated in the mathematical model is not remarkably accurate, since a few cases that can result in a handover failure are not taken into account, leading to an underestimation of the handover latency. Secondly, rather than occurring only once in the mathematical model, collision between handover reply beacons may happen several times in the simulation. Thirdly, due to the limited number of simulation runs, the backoff duration before handover can be sharply deviated from the average value calculated in the mathematical model. Finally, the number of data packets transmitted during the handover process may be a decimal in the mathematical model. However, it must be an integer having a definite physical meaning in the simulation.

When the handover mechanism is not used, different from that in the mathematical model, the original transmitter is allowed to win the medium at any competition after the link breaks in the simulation. In other words, the limitations that the original transmitter can win the channel at latest at the third contention and only 140 data packets can be transmitted before it wins are relaxed. Without these restrictions, the latency for setting up a new link becomes much larger in the simulation than that in the mathematical model. Nevertheless, the latency displayed in the simulation figure exhibits a similar variation trend with that described in the analytical figure, but with more fluctuations. The curve fluctuation is a result of the finite simulation runs and the multiple communication patterns that a node can select after the link breaks. The channel competition in which the original transmitter wins and the number of collisions and contention failures occurred before it wins are randomly chosen by the simulation runs. Due to the complete randomness, the latency required for setting up a new link under a small duty cycle can be even greater than that under a big duty cycle. Similar phenomenon appears for the network density. Nevertheless, on the whole, the latency for the new link establishment increases with the increment in both the duty cycle and the network density.

The simulation result is found to perform in accordance with the analytical result. It is asserted that the latency of packet transmission with the handover support is much less than that without the handover support. The latency can be saved by at least 0.28s when the handover mechanism is applied. This figure can even grow as the duty cycle and the network density increase. The simulation result matches the analytical result and further verifies that the handover mechanism improves the latency of packet transmission in a mobile wireless sensor network.

10.9 Contributions of Thesis

Based on both the analytical and the simulation results, the thesis is verified to successfully achieved the formulated objectives. In summery, it makes the following contributions:

- Elaborating the challenges of mobility on the performance of the existing MAC protocols and the need to maintain an unbroken link during data transmission;

10. CONCLUSION

- Developing an adaptive handover mechanism to reduce the packet transmission latency in a mobile wireless sensor network;
- Establishing a mathematical model to examine the performance of the handover mechanism;
- Utilizing the NS2 network simulator to simulate the performance of the handover mechanism.

10.10 Future Work

In the future, to validate the mathematics- and simulation-based evaluation reported in this thesis, and to explore hardware platform-dependent trends and problems, we plan to investigate the latency of packet transmission in an implementation in TinyOS on IMotes. By doing so, whether and how much the performance of a hybrid MAC protocol can be improved in a mobile wireless sensor network with the handover support can be experimentally evaluated.

Bibliography

- [1] CC2420 datasheet. <http://www.ti.com>. 111
- [2] Hüseyin Akcan, Vassil Kriakov, Hervé Brönnimann, and Alex Delis. GPS-Free node localization in mobile wireless sensor networks. *Proceedings of the 5th ACM international workshop on Data engineering for wireless and mobile access*, pages 35–42, 2006. 23, 30
- [3] I. F. Akyildiz, Su Weilian, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114, 2002. 13, 16
- [4] Muneeb Ali, Tashfeen Suleman, and Zartash Afzal Uzmi. MMAC: a mobility-adaptive, collision-free MAC protocol for wireless sensor networks. *24th IEEE International Performance, Computing, and Communications Conference*, pages 401 – 407, April 2005. 3, 4, 35, 45
- [5] J.-P. Allouche, F. von Haeseler, H.-O. Peitgen, and G. Skordev. Linear cellular automata, finite automata and pascal’s triangle. *Discrete Appl. Math.*, 66:1–22, April 1996. 81
- [6] W. Alsalih, S. Akl, and H. Hassanein. Placement of multiple mobile base stations in wireless sensor networks. *IEEE International Symposium on Signal Processing and Information Technology*, pages 229 – 233, December 2007. 24
- [7] Amiya Bhattacharya and Sajal K. Das. LeZi-update: an information-theoretic framework for personal mobility tracking in PCS networks. *Wireless Networks*, 8:121–135, March 2002. 28
- [8] G. Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *Selected Areas in Communications, IEEE Journal on*, 18(3):535–547, 2000. 17, 18, 112
- [9] Mathias Boc, Anne Fladenmuller, and Marcelo Dias de Amorim. Otiy: Locators tracking nodes. *Proceedings of the 2007 ACM CoNEXT conference*, 14:1–12, 2007. 28
- [10] Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks. *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 307–320, 2006. 3, 48

BIBLIOGRAPHY

- [11] Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks, 2006. 20
- [12] J. Burrell, T. Brooke, and R. Beckwith. Vineyard computing: sensor networks in agricultural production. *IEEE Pervasive Computing*, 3:38 – 45, March 2004. 1
- [13] Alberto Camilli, Carlos E. and Antonio M. Saraiva, Andr R. Hirakawa, and Pedro L.P. Corra. From wireless sensors to field mapping: Anatomy of an application for precision agriculture. *Computers and Electronics in Agriculture*, pages 25–36, August 2007. 1
- [14] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. *Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2:483–502, 2002. 23, 24, 67
- [15] S. Capkun, M. Hamdi, and J. Hubaux. GPS-free positioning in mobile ad hoc networks. *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, 5:157–167, January 2001. 31
- [16] Arnab Chakrabarti, Ashutosh Sabharwal, and Behnaam Aazhang. Using predictable observer mobility for power efficient design of sensor networks. *Proceedings of the 2nd international conference on Information processing in sensor networks*, 2634:129–145, 2003. 25
- [17] Xiaojuan Chao, Waltenegus Dargie, and Guan Lin. Energy model for H2S monitoring wireless sensor network. *Proceedings of the 2008 11th IEEE International Conference on Computational Science and Engineering*, pages 402–409, 2008. 2
- [18] Mingang Cheng, Masako Kanai-Pak, Noriaki Kuwahara, Hiromi Itoh Ozaku, Kiyoshi Kogure, and Jun Ota. Dynamic scheduling based inpatient nursing support: applicability evaluation by laboratory experiments. *Proceedings of the 3rd ACM International Workshop on Context-Awareness for Self-Managing Systems*, pages 48–54, 2009. 2
- [19] S. Dagtas, Y. Natchetoi, and H. Wu. An integrated wireless sensing and mobile processing architecture for assisted living and healthcare applications. *Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments*, pages 70–72, 2007. 2
- [20] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks, 2003. 19
- [21] Waltenegus Dargie and Mieso K. Denko. Analysis of error-agnostic time- and frequency-domain features extracted from measurements of 3-D accelerometer sensors. *IEEE System Journal*, 4:26–33, March 2010. 28

- [22] Waltenegus Dargie and Christian Poellabauer. *Fundamentals of Wireless Sensor Networks: Theory and Practice*. Wiley Publishing, 2010. 11
- [23] Murat Demirbas, Anish Arora, Vineet Mittal, and Vinodkrishnan Kulathumani. A fault-local self-stabilizing clustering service for wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 17:912–922, September 2006. 37
- [24] Qian Dong and Waltenegus Dargie. Evaluation of the reliability of rssi for indoor localization. In *Proceedings of the third International Conference on Wireless Communications in Unusual and Confined Areas (ICWCUCA '12)*, August 28-30, 2012. 111
- [25] Olivier Dousse, Petteri Mannersalo, and Patrick Thiran. Latency of wireless sensor networks with uncoordinated power saving mechanisms, 2004. 15
- [26] Shu Du, A.K. Saha, and D.B. Johnson. Rmac: A routing-enhanced duty-cycle mac protocol for wireless sensor networks. *26th IEEE International Conference on Computer Communications (INFOCOM 2007)*, pages 1478 – 1486, May 2007. 47
- [27] E. Ekici, Yaoyao Gu, and D. Bozdag. Mobility-based communication in wireless sensor networks. *IEEE In Communications Magazine*, 44:56–62, July 2006. 24, 25, 26
- [28] Amre EL-Hoiydi and Jean-Dominique Decotignie. WiseMAC: an ultra low power MAC protocol for multi-hop wireless sensor networks. In *Proceedings of the First International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS 2004)*, pages 18–31, July 2004. 3
- [29] Amre El-Hoiydi and Jean-Dominique Decotignie. Low power downlink mac protocols for infrastructure wireless sensor networks. *Mob. Netw. Appl.*, 10(5):675–690, 2005. 20
- [30] E. Elnahrawy, Xiaoyan Li, and R.P. Martin. The limits of localization using signal strength: a comparative study. *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pages 406 – 414, October 2004. 30
- [31] Károly Farkas, Theus Hossmann, Franck Legendre, Bernhard Plattner, and Sajal K. Das. Link quality prediction in mesh networks. *Computer Communications*, 31:1497–1512, May 2008. 27, 28, 30
- [32] A. El Gamal, J. Mammen, B. Prabhakar, and D. Shah. Throughput-delay trade-off in wireless networks. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, page 475. 16
- [33] S.R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan. Energy efficient schemes for wireless sensor networks with multiple mobile base stations. *IEEE Global Telecommunications Conference*, 1:377–381, December 2003. 25

BIBLIOGRAPHY

- [34] Haigang Gong, Ming Liu, Lingfei Yu, and Xiaomin Wang. An event driven tdma protocol for wireless sensor networks. *WRI International Conference on Communications and Mobile Computing, 2009*, 2:132 – 136, Jan 2009. 17, 47
- [35] A. Gonga, O. Landsiedel, and M. Johansson. MobiSense: Power-efficient micro-mobility in wireless sensor networks. *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems*, June 2011. 38
- [36] P. Gupta and P. R. Kumar. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2):388–404, 2000. 16
- [37] L.van Hoesel and P. Havinga. Collision-free time slot reuse in multi-hop wireless sensor networks. *Proceedings of the 2005 International Conference on Intelligent Sensors, Sensor Networks and Information Processing Conference*, pages 101–107, December 2005. 39
- [38] Dai Hong-Ning. Throughput and delay in wireless sensor networks using directional antennas. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pages 421–426. 16
- [39] Lingxuan Hu and David Evans. Localization for mobile sensor networks. *Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 45–57, 2004. 29
- [40] Xu Huang, M. Barralet, and D. Sharma. Behaviors of antenna polarization for RSSI location identification. *International Conference on Networks Security, Wireless Communications and Trusted Computing*, 1:151 – 154, April 2009. 30, 31
- [41] Tan Hwee-Xian, W. K. G. Seah, and Chan Kwang-Mien. Distributed cdma code assignment for wireless sensor networks. In *Radio and Wireless Symposium, 2006 IEEE*, pages 139–142. 17
- [42] P. R. Jelenkovic, P. Momcilovic, and M. S. Squillante. Scalability of wireless networks. *Networking, IEEE/ACM Transactions on*, 15(2):295–308, 2007. 16
- [43] Arshad Jhumka and Sandeep Kulkarni. On the design of mobility-tolerant TDMA-based media access control (MAC) protocol for mobile sensor networks. *Proceedings of the 4th international conference on Distributed computing and internet technology*, 4882:42–53, 2007. 34, 37
- [44] Kim Joohwan, Lin Xiaojun, N. B. Shroff, and P. Sinha. Minimizing delay and maximizing lifetime for wireless sensor networks with anycast. *Networking, IEEE/ACM Transactions on*, 18(2):515–528, 2010. 15
- [45] Aman Kansal, Arun A. Somasundara, David D. Jea, Mani B. Srivastava, and Deborah Estrin. Intelligent fluid infrastructure for embedded networks. *Proceedings of the 2nd*

- international conference on Mobile systems, applications, and services*, pages 111–124, 2004. 26
- [46] Meriem Kassar, Brigitte Kervella, and Guy Pujolle. An overview of vertical handover decision strategies in heterogeneous wireless networks. *Computer Communications*, pages 2607–2620, 2008. 61
- [47] Sukun Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon. Health monitoring of civil infrastructures using wireless sensor networks. *6th International Symposium on Information Processing in Sensor Networks*, pages 254 – 263, April 2007. 1
- [48] Kae Hsiang Kwong, Tsung Ta Wu, Hock Guan Goh, and Bruce Stephen. Wireless sensor networks in agriculture: Cattle monitoring for farming industries. *Electromagnetics Research Symposium*, 5:31–35, 2009. 2
- [49] Sang Hyuk Lee, Soobin Lee, Heecheol Song, and Hwang Soo Lee. Wireless sensor network design for tactical military applications: remote large-scale environments. *Proceedings of the 28th IEEE conference on Military communications*, pages 911–917, 2009. 2
- [50] Woo-Yong Lee, Kyeong Hur, and Doo-Seop Eom. Navigation of mobile node in wireless sensor networks without localization. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 1 – 7, August 2008. 29, 31
- [51] Jae S. Lim and Alan V. Oppenheim. *Advanced topics in signal processing*. Prentice-Hall, Inc., 1987. 28
- [52] Ren-Shiou Liu, Kai-Wei Fan, and P. Sinha. Clearburst: Burst scheduling for contention-free transmission in sensor networks. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 1899 –1904, 31 2008-april 3 2008. 52
- [53] Yong-Min Liu and Wu-Yi Lu. Improved fairness using dccp in wireless sensor networks. In *Networking and Digital Society, 2009. ICNDS '09. International Conference on*, volume 1, pages 141–144. 17
- [54] Konrad Lorincz, David J. Malan, Thaddeus R. F. Fulford-Jones, Alan Nawoj, Antony Clavel, Victor Shnayder, Geoffrey Mainland, Matt Welsh, and Steve Moulton. Sensor networks for emergency response: challenges and opportunities. *IEEE Pervasive Computing*, 3:16–23, October 2004. 2
- [55] Jun Luo and J.-P. Hubaux. Joint mobility and routing for lifetime elongation in wireless sensor networks. *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, 3:1735 – 1746, March 2005. 25

BIBLIOGRAPHY

- [56] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97. ACM, 2002. 1
- [57] Raluca Marin-Perianu, Mihai Marin-Perianu, Paul Havinga, and Hans Scholten. Movement-based group awareness with wireless sensor networks. *Proceedings of the 5th international conference on Pervasive computing*, pages 298–315, 2007. 29
- [58] Merryn J Mathie, Adelle CF Coster, Nigel H Lovell, and Branko G Celler. Accelerometry: providing an integrated, practical method for long-term, ambulatory monitoring of human movement. *Physiological Measurement*, 25:R1–R20, February 2004. 31
- [59] Kevin Mayer, Keith Ellis, and Ken Taylor. Cattle health monitoring using wireless sensor networks. *Second IASTED International Conference on Communication and Computer Networks*, 2004. 3
- [60] Hirokazu Miura, Kazuhiko Hirano, Noriyuki Matsuda, Hirokazu Taki, Norihiro Abe, and Satoshi Hori. Indoor localization for mobile node based on RSSI. *Proceedings of the 11th international conference, KES 2007 and XVII Italian workshop on neural networks conference on Knowledge-based intelligent information and engineering systems: Part III*, pages 1065–1072, 2007. 31
- [61] M. Nabi, M. Blagojevic, M. Geilen, T. Basten, and T. Hendriks. MCMAC: An optimized medium access control protocol for mobile clusters in wireless sensor networks. *7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, pages 1–9, June 2010. 39
- [62] E. S. Nadimi, H. T. Søgaaard, T. Bak, and F. W. Oudshoorn. Zigbee-based wireless sensor networks for monitoring animal presence and pasture time in a strip of new grass. *Comput. Electron. Agric.*, 61(2):79–87, May 2008. 2
- [63] Mun Leng Ng, Kin Seong Leong, and P.H. Hall, D.M.and Cole. A small passive UHF RFID tag for livestock identification. *IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, 1:67–70, August 2005. 29
- [64] Dragos Niculescu and Badri Nath. DV based positioning in ad hoc networks. *Journal of Telecommunication Systems*, 22:267–280, 2003. 29, 32
- [65] Frank W. Oudshoorn, Troels Kristensen, and Esmaeil Shahrak Nadimi. Dairy cow defecation and urination frequency and spatial distribution related to time limited grazing. *Livestock Science*, 113:62–73, January 2008. 29
- [66] P.A.Anemaet. Distributed G-MAC: A flexible MAC protocol for servicing gossip algorithms. *Master's thesis, Technical University of Delft*, 2008. 39

- [67] H. Pham and S. Jha. An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC). *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pages 558 – 560, October 2004. 32
- [68] S. Pilosof, Ramjee Ramachandran, D. Raz, Y. Shavitt, and Sinha Prasun. Understanding tcp fairness over wireless lan. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 2, pages 863–872 vol.2. 17
- [69] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, 2004. 3
- [70] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks, 2004. 20, 48
- [71] Nissanka B. Priyantha, Hari Balakrishnan, Erik Demaine, and Seth Teller. Anchor-free distributed localization in sensor networks. *Technical Report, MIT Laboratory for Computer Science*, 2003. 30
- [72] Anjali Raja and Xiao Su. Mobility handling in MAC for wireless ad hoc networks. *Wireless Communications and Mobile Computing*, 9:303–311, March 2009. 23
- [73] Venkatesh Rajendran, Katia Obraczka, and J. J. Garcia-Luna-Aceves. Energy-efficient, collision-free medium access control for wireless sensor networks. *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 181–192, 2003. 35
- [74] Christian Schindelhauer. Mobility in wireless networks. *SOFSEM*, pages 100–116, 2006. 22
- [75] P.L. Senger. The estrus detection problem: New concepts, technologies, and possibilities. *Journal of Dairy Science*, 77(9):2745 – 2753, 1994. 3
- [76] C.K. Seovv, W.K.G. Seah, and Z. Liu. Hybrid mobile wireless sensor network cooperative localization. *IEEE 22nd International Symposium on Intelligent Control*, pages 29 – 34, October 2007. 29, 32
- [77] R.C. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: modeling a three-tier architecture for sparse sensor networks. *Proceedings of the First IEEE. 2003 IEEE International Workshop on Sensor Network Protocols and Applications*, pages 30 – 41, May 2003. 25
- [78] M.L. Sichitiu and V. Ramadurai. Localization of wireless sensor networks with a mobile beacon. *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pages 174 – 183, October 2004. 30

BIBLIOGRAPHY

- [79] Arun A. Somasundara, Aditya Ramamoorthy, and Mani B. Srivastava. Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines. *Proceedings of the 25th IEEE International Real-Time Systems Symposium*, pages 296–305, December 2004. 25
- [80] Dongjin Son, Ahmed Helmy, and Bhaskar Krishnamachari. The effect of mobility-induced location errors on geographic routing in mobile ad hoc and sensor networks: analysis and improvement using mobility prediction. *IEEE Transactions on Mobile Computing*, 3:233–245, July 2004. 24
- [81] I. Stoianov, L. Nachman, S. Madden, T. Tokmouline, and M. Csail. PIPENET: A wireless sensor network for pipeline monitoring. *6th International Symposium on Information Processing in Sensor Networks*, pages 25–27, April 2007. 1
- [82] William Su, Sung-Ju Lee, and Mario Gerla. Mobility prediction and routing in ad hoc wireless networks. *International Journal of Network Management*, 11:3–30, January 2001. 27
- [83] Yanjun Sun, Shu Du, Omer Gurewitz, and David B. Johnson. Dw-mac: a low latency, energy efficient demand-wakeup mac protocol for wireless sensor networks. In *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing, MobiHoc '08*, pages 53–62, New York, NY, USA, 2008. ACM. 47
- [84] Yanjun Sun, Omer Gurewitz, and David B. Johnson. RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks. *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 1–14, 2008. 4
- [85] Yanjun Sun, Omer Gurewitz, and David B. Johnson. Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks, 2008. 20
- [86] Yanjun Sun, Omer Gurewitz, and David B. Johnson. Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 1–14, 2008. 48
- [87] Zhiyong Tang and Waltenegus Dargie. A mobility-aware medium access control protocol for wireless sensor networks. *The fifth IEEE international workshop on Heterogeneous, Multi-Hop, Wireless and Mobile Networks (Globecom 2010)*, December 2010. 32, 45
- [88] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 171–180, 2003. 3

- [89] Yadong Wan, Qin Wang, Xiaotong Zhang, Jinwu Xu, Lei Geng, Lei Li, and Ruofei Wang. A hybrid tdm-fdm mac protocol for wireless sensor network using timestamp self-adjusting synchronization mechanism. In *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pages 2705–2709. 17
- [90] Z. Maria Wang, Stefano Basagni, Emanuel Melachrinoudis, and Chiara Petrioli. Exploiting sink mobility for maximizing sensor networks lifetime. *Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 287a – 287a, January 2005. 25
- [91] T. Watanabe, A. Sakurai, and K. Kitazaki. Dairy cattle monitoring using wireless acceleration-sensor networks. In *Sensors, 2008 IEEE*, pages 526 –529, oct. 2008. 3
- [92] Ye Wei, J. Heidemann, and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1567–1576 vol.3. 14, 17, 19
- [93] E. W. Weisstein. The CRC concise encyclopedia of mathematics. *CRC Press*, 1998. 23
- [94] Greg Welch and Gary Bishop. An introduction to the kalman filter. *Technical Report, University of North Carolina at Chapel Hill*, 1995. 27
- [95] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10:18 – 25, March 2006. 1
- [96] Dongliang Xie, Mingxing Chen, Chen Canfeng, and Jian Ma. A patrol grid protocol for mobile wireless sensor network. *Proceedings of the 4th Annual International Conference on Wireless Internet*, pages 1–8, November 2008. 24
- [97] Yurong Xu, Yi Ouyang, Zhengyi Le, James Ford, and Fillia Makedon. Mobile anchor-free localization for wireless sensor networks. *Proceedings of the 3rd IEEE international conference on Distributed computing in sensor systems*, 4549:96–109, 2007. 30, 31
- [98] Wei Ye, J. Heidemann, and D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. *IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, 3:1567 – 1576, 2002. 3
- [99] Wei Ye, J. Heidemann, and D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. *IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, 3:1567 – 1576, 2002. 53
- [100] Wei Ye, Fabio Silva, and John Heidemann. Ultra-low duty cycle mac with scheduled channel polling. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, SenSys '06, pages 321–334, New York, NY, USA, 2006. ACM. 111

BIBLIOGRAPHY

- [101] Zainab R. Zaidi and Brian L. Mark. Real-time mobility tracking algorithms for cellular networks based on kalman filtering. *IEEE Transactions on Mobile Computing*, 4:195–208, March 2005. 28
- [102] Z.R. Zaidi and B.L. Mark. Mobility estimation for wireless networks based on an autoregressive model. *IEEE GLOBECOM 2004*, 6:3405 – 3409, 2004. 28, 36
- [103] S. Zhou, M.-Y. Wu, and W. Shu. Finding optimal placements for mobile sensor: wireless sensor network topology adjustment. *Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication*, 2:529 – 532, 2004. 27
- [104] Jacob Ziv and Abraham Lempel. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, IT-23:337 – 343, May 1977. 28

Appendix A

Simulation Details

A.1 Simulation with Handover

Figure A.1 presents a trace file for the network density $M = 3$ when the handover mechanism is applied. This trace file is generated after running a simulation in which the mobile transmitter wins the medium. Here, nodes 1, 2, 3 and 4, 5, 6 signify the transmitters and the receivers, respectively. Particularly, node 1 is the mobile transmitter. It moves away from its original receiver 4 once it seizes the channel. The letters s , r and D in the first column denote, in respective order, the packet transmission, receipt and discard. The second, third, sixth, seventh and eighth columns represent the action time, the node ID, the packet sequence number, the packet type and the packet size, respectively.

As displayed in Figure A.1a, receiver 4 broadcasts a beacon at $0.229208000s$. Since it is a base beacon, each of the surrounding transmitters 1, 2, 3 will send a data packet without backing-off after receiving it, leading to a collision. This results in a new beacon broadcast from receiver 4 after a random delay. Since transmitter 1 selects the shorted waiting value according to the BW field of the beacon, it wins the medium and transmits its first data packet at $0.233208235s$. To minimize the simulation time, the number of beacons used for the distance evaluation is set as one here ($w = 1$). If the relative distance is estimated to be larger than the threshold, transmitter 1 will immediately start the handover mechanism by sending a data packet at $0.235384400s$ acting as a handover request. Being as the original receiver, node 4 only acknowledges with an ACK beacon. Since at the moment, both nodes 5 and 6 are in the sleep mode,

A. SIMULATION DETAILS

```

s 0.229208000 _4_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.229752076 _3_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.229752076 _2_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.229752083 _1_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
s 0.229944076 _3_ MAC --- 1 cbr 45 [0 ffffffff ffff0003 0] ----- [3:0 -1:0 32 0] [1] 0 0
s 0.229944076 _2_ MAC --- 1 cbr 45 [0 ffffffff ffff0002 0] ----- [2:0 -1:0 32 0] [2] 0 0
s 0.229944083 _1_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 0 0
D 0.229944152 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0003 0] ----- [3:0 -1:0 32 0] [1] 0 0
D 0.229944152 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0002 0] ----- [2:0 -1:0 32 0] [2] 0 0
D 0.229944165 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 0 0
s 0.231704152 _4_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.232248228 _3_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.232248228 _2_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 0.232248235 _1_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
s 0.233208235 _1_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 0 0
D 0.234648244 _2_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 0 0
D 0.234648244 _3_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 0 0
r 0.234648317 _4_ MAC --- 1 cbr 32 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [1] 1 0
s 0.234840317 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 0.235192400 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 0.235384400 _1_ MAC --- 2 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [2] 0 0
D 0.236824409 _2_ MAC --- 2 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [2] 0 0
D 0.236824409 _3_ MAC --- 2 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [2] 0 0
r 0.236824482 _4_ MAC --- 2 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [2] 1 0

```

(a)

```

s 0.237016482 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 0.237368565 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 0.247288565 _1_ MAC --- 3 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [3] 0 0
D 0.248728574 _2_ MAC --- 3 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [3] 0 0
D 0.248728574 _3_ MAC --- 3 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [3] 0 0
r 0.248728647 _4_ MAC --- 3 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [3] 1 0
s 0.248920647 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 0.249272730 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 0.259192730 _1_ MAC --- 4 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [0] 0 0
D 0.260632740 _2_ MAC --- 4 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [0] 0 0
D 0.260632740 _3_ MAC --- 4 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [0] 0 0
r 0.260632813 _4_ MAC --- 4 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [0] 1 0
s 0.260824813 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 0.261176895 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 0.271096895 _1_ MAC --- 5 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 0 0
D 0.272536905 _2_ MAC --- 5 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 0 0
D 0.272536905 _3_ MAC --- 5 cbr 45 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 0 0
r 0.272536962 _5_ MAC --- 5 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 1 0
r 0.272536962 _6_ MAC --- 5 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 1 0
r 0.272536978 _4_ MAC --- 5 cbr 32 [0 ffffffff ffff0001 100] ----- [1:0 -1:0 32 0] [6] 1 0

```

(b)

```

s 0.272728978 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 0.273081061 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 0.279704962 _5_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
r 0.280057028 _1_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
s 0.280249028 _1_ MAC --- 6 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [10] 0 0
D 0.281689038 _2_ MAC --- 6 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [10] 0 0
D 0.281689038 _3_ MAC --- 6 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [10] 0 0
r 0.281689095 _5_ MAC --- 6 cbr 32 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [10] 1 0
s 0.281689095 _5_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
r 0.282041161 _1_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
s 0.282233161 _1_ MAC --- 7 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [4] 0 0
D 0.283673171 _2_ MAC --- 7 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [4] 0 0
D 0.283673171 _3_ MAC --- 7 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [4] 0 0
r 0.283673227 _5_ MAC --- 7 cbr 32 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [4] 1 0
s 0.283673227 _5_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
r 0.284025294 _1_ MAC --- 0 ACK 18 [0 ffff0001 5 100]
s 0.284217294 _1_ MAC --- 8 cbr 45 [0 ffffffff ffff0001 0] ----- [1:0 -1:0 32 0] [8] 0 0

```

(c)

Figure A.1: The format of a trace file with the handover support

they cannot participate in the handover process, and thus, the first handover attempt fails. The failure will not stop until at least one of them wakes up. As displayed in Figure A.1b, before the mobile transmitter broadcasts the fourth handover request at 0.271096895s, both of the receivers become qualified, and thus, will contend for the channel. As shown in Figure A.1c, node 5 eventually wins, since it responses with a handover reply beacon (also denoted by ACK) at an earlier time 0.279704962s. As soon as transmitter 1 receives this feedback, it will transfer its communication to this new receiver. The new link is established at 0.280249028s.

Deserving to be mentioned, even if the mobile transmitter wins the medium in all the simulations under the same duty cycle and network density, the trace files can have different formats. This is because the backoff intervals selected by either the transmitters before the data transmission or the receivers before the handover reply broadcast are rather random. In addition, collision may occur during the normal data communication as well as the handover process.

```

3 28 5 start 0.229208 1_does_not_win handover_time 0.000000 total_time -0.229208
3 28 5 start 0.229208 1_wins handover_start 0.244024 handover_end 0.260409 end 0.532235
3 28 5 start 0.229208 1_wins handover_start 0.236664 handover_end 0.268473 end 0.538315
3 28 5 start 0.229208 1_does_not_win handover_time 0.000000 total_time -0.229208
3 28 6 start 0.229208 1_does_not_win handover_time 0.000000 total_time -0.229208
3 28 6 start 0.229208 1_wins handover_start 0.238264 handover_end 0.273593 end 0.543435
3 28 6 start 0.229208 1_does_not_win handover_time 0.000000 total_time -0.229208
3 28 6 start 0.229208 1_wins handover_start 0.245944 handover_end 0.266809 end 0.538635

```

Figure A.2: Data sets obtained from a trace file

After a trace file is generated, the programming language *gawk* is used to analyze the trace to get the data sets that we are interested in. Figure A.2 shows a small segment of a long *gawk* file. The first column denotes the duty cycle with one hundred multiples; the second and third columns signify the wake up time of the contending receivers. The words *1_wins* and *1_does_not_win* indicate whether the mobile transmitter seizes the medium or not at the beginning of the communication. Only those lines with *1_wins* are valid and will be used for the quantification of the handover latency. The meaning of all the other time stamps is obvious. After picking up and averaging the useful simulation results for each duty cycle, the latency of the packet transmission when the handover mechanism is used can be evaluated.

A. SIMULATION DETAILS

A.2 Simulation without Handover

Figure A.3 displays a trace file for the network density $M = 3$ when the handover mechanism is not applied. This trace file is generated under a large duty cycle. Therefore, before a base beacon is broadcasted at the beginning of the communication, all the three transmitters have already woken up from a sleep period. Since every event is recorded in the trace file, the size of it is rather big. For space saving, the events that denote the discard of packets are not displayed in the figure.

As illustrated in Figure A.3a, receiver 4 at 10.069208000s wakes up and broadcasts a base beacon. Since all the transmitters are active at the moment, they will send a data packet respectively without backing-off, leading to a collision. To avoid future collisions on data packets, receiver 4 at 10.080984165s broadcasts another beacon including an additional BW field. Since the mobile transmitter selects the smallest waiting value, it wins the medium and transmits its first data packet at 10.085048248s. From this moment on, a link between nodes 1 and 4 has been established. However, it will eventually terminate due to the unemployment of the handover mechanism. As described in Figure A.3b, after acknowledging the receipt of the 70th data packet with an ACK beacon, receiver 4 receives no more data packet from node 1. This happens because the mobile transmitter has already moved out of the maximum radio transmission range of it.

To proceed with the data transmission, node 1 will search for a new link. The new link establishment begins with the detection of a new receiver. As can be observed, after receiver 5 wakes up and broadcasts a base beacon at 10.259208000s, all the transmitters will contend for the medium by sending their data packets. Different from nodes 2 and 3 which are going to transmit their first data packet, node 1 plans to retransmit its 70th data packet that is failed to be sent in the original link. Once node 5 detects a collision, it broadcasts a new beacon at 10.264264131s. Rather than the mobile transmitter, node 2 wins the channel this time. It will not stop transmitting its data packets until the last one is successfully sent. This can be noticed in Figure A.3c. Only after receiver 5 receives the 140th data packet from node 2, will it broadcast a new beacon containing all the SRC, BW and DST fields. This beacon acts as an acknowledgement for node 2 (denoted by *ACK*), and meanwhile as a transmission invitation for all the transmitters (denoted

A.2 Simulation without Handover

```
s 10.069208000 _4_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.069752083 _1_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.069752083 _3_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.069752083 _2_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
s 10.069944083 _1_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [2] 0 0
s 10.069944083 _3_ MAC --- 1 cbr 45 [0 ffffffff ffff0003 800] ----- [3:0 -1:0 32 0] [1] 0 0
s 10.069944083 _2_ MAC --- 1 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 0 0
D 10.069944166 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [2] 0 0
D 10.069944166 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0003 800] ----- [3:0 -1:0 32 0] [1] 0 0
D 10.069944166 _4_ MAC COL 1 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 0 0
s 10.080984165 _4_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.081528248 _1_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.081528248 _3_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
r 10.081528248 _2_ MAC --- 0 UNKN 17 [0 ffffffff 4 0]
s 10.085048248 _1_ MAC --- 1 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [2] 0 0
r 10.086488330 _4_ MAC --- 1 cbr 32 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [2] 1 0
s 10.086680330 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 10.087032413 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 10.087224413 _1_ MAC --- 2 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [3] 0 0
r 10.088664495 _4_ MAC --- 2 cbr 32 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [3] 1 0
s 10.088856495 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
r 10.089208578 _1_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 10.089400578 _1_ MAC --- 3 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [4] 0 0
```

(a)

```
s 10.235203695 _1_ MAC --- 70 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 0 0
r 10.236643778 _4_ MAC --- 70 cbr 32 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 1 0
s 10.236835778 _4_ MAC --- 0 ACK 18 [0 ffff0001 4 0]
s 10.259208000 _5_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.259752066 _1_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.259752067 _2_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.259752067 _3_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
s 10.259944066 _1_ MAC --- 70 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 0 0
s 10.259944067 _2_ MAC --- 1 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 0 0
s 10.259944067 _3_ MAC --- 1 cbr 45 [0 ffffffff ffff0003 800] ----- [3:0 -1:0 32 0] [1] 0 0
D 10.259944134 _5_ MAC COL 70 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 0 0
D 10.259944134 _5_ MAC COL 1 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 0 0
D 10.259944134 _5_ MAC COL 1 cbr 45 [0 ffffffff ffff0003 800] ----- [3:0 -1:0 32 0] [1] 0 0
s 10.264264131 _5_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.264808197 _1_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.264808198 _2_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
r 10.264808198 _3_ MAC --- 0 UNKN 17 [0 ffffffff 5 0]
s 10.266088198 _2_ MAC --- 1 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 0 0
r 10.267528265 _5_ MAC --- 1 cbr 32 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [0] 1 0
s 10.267720265 _5_ MAC --- 0 ACK 18 [0 ffff0002 5 0]
r 10.268072332 _2_ MAC --- 0 ACK 18 [0 ffff0002 5 0]
s 10.268264332 _2_ MAC --- 2 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [3] 0 0
r 10.269704399 _5_ MAC --- 2 cbr 32 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [3] 1 0
s 10.269896399 _5_ MAC --- 0 ACK 18 [0 ffff0002 5 0]
```

(b)

```
s 10.568570824 _2_ MAC --- 140 cbr 45 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [138] 0 0
r 10.570010891 _5_ MAC --- 140 cbr 32 [0 ffffffff ffff0002 800] ----- [2:0 -1:0 32 0] [138] 1 0
s 10.570202891 _5_ MAC --- 0 ACK 18 [0 ffff0002 5 0]
r 10.570554955 _1_ MAC --- 0 UNKN 18 [0 ffffffff 5 0]
r 10.570554958 _2_ MAC --- 0 ACK 18 [0 ffff0002 5 0]
r 10.570554958 _3_ MAC --- 0 UNKN 18 [0 ffffffff 5 0]
s 10.573114955 _1_ MAC --- 70 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 0 0
r 10.574555020 _5_ MAC --- 70 cbr 32 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [69] 1 0
s 10.574747020 _5_ MAC --- 0 ACK 18 [0 ffff0001 5 0]
r 10.575099084 _1_ MAC --- 0 ACK 18 [0 ffff0001 5 0]
s 10.575291084 _1_ MAC --- 71 cbr 45 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [70] 0 0
r 10.576731148 _5_ MAC --- 71 cbr 32 [0 ffffffff ffff0001 800] ----- [1:0 -1:0 32 0] [70] 1 0
s 10.576923148 _5_ MAC --- 0 ACK 18 [0 ffff0001 5 0]
r 10.577275212 _1_ MAC --- 0 ACK 18 [0 ffff0001 5 0]
```

(c)

Figure A.3: The format of a trace file without the handover support

A. SIMULATION DETAILS

by *UNKN*). Since the mobile transmitter wins the channel this time at 10.573114955, it can eventually start to resume its remaining data transmission.

According to the trace file, the programming language *gawk* is used to extract the useful information based on which the communication latency can be evaluated. The format of the *gawk* file when the handover mechanism is not used is similar to that when it is used.