ISBN 978-82-32-2342-6 (printed version)
ISBN 978-82-326-2343-3 (electronic version)
ISSN 1503-8181

NTNU
Norwegian University of
Science and Technology

Doctoral Thesis

Anne-Lena Kampen

Doctoral theses at NTNU, 2017:135

NTNU

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology and
Electrical Engineering
Department of Information Security
and Communication Technology

Doctoral theses at NTNU, 2017:135

Anne-Lena Kampen

Wireless Sensor Networks, energy efficiency and path recovery

NTNU
Norwegian University of
Science and Technology

Anne-Lena Kampen

Wireless Sensor Networks, energy efficiency and path recovery

Thesis for the degree of Philosophiae Doctor

Trondheim, May 2017

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Information Security and Communication Technology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology
and Electrical Engineering
Department of Information Security
and Communication Technology

ISBN 978-82-32-2342-6 (printed version)
ISBN 978-82-326-2343-3 (electronic version)
ISSN 1503-8181

Doctoral theses at NTNU, 2017:135

Printed by Skipnes Kommunikasjon as

Abstract

Wireless Sensor Networks (WSN) consists of sensor nodes equipped with radios for wireless communication. The overall goal of the sensor networks is to gather data. The terminal point for the data is usually a specific node, called the sink. Nodes collaborate to relay data when direct communication between source and sink is impossible.

One of the main issues in WSN is energy consumption. Depleted nodes cannot collect data and the value of the accumulated data at the sink is therefore degraded. Since nodes forward traffic towards the sink, a depleted node may lead to network partitioning, thereby causing part of the network to be unavailable for the sink.

This study investigates the energy consumption related to the radio, and suggests algorithms to reduce the consumption. During operation, the radios switch between different states such as receiving, transmitting, idle and sleep. The amount of energy consumed varies with the radio state. The investigation reveals that the energy consumed to receive packets can have a substantial impact on the total consumption. In order to reduce the energy consumed in receiving, a simple algorithm has been developed that can function as an add-on to common communication protocols. The algorithm enables nodes to enter the sleep state rather than receive traffic that is not addressed to them.

The second topic addressed is the balance of energy consumption among the nodes. Balancing the energy consumption is a means to achieve an even residual-energy level among the nodes. The goal is to avoid early depletion of nodes, thereby preserving network availability. A broad range of different balancing algorithms has been presented in the literature and these have been classified, analyzed and compared. In addition, new balancing algorithms have been suggested. The routing protocol for Low Power and Lossy network (RPL) was used as the basis for assessment and improvements have been suggested. The findings are that, by introducing a minor change in the RPL algorithm, a significant balancing effect can be achieved. However, the best balancing effect is achieved if nodes always transmit data toward the next-hop node with the highest residual-energy level.

The third topic addressed is path recovery algorithms. Radio links may break or a node may die, either due to faults or to the already mentioned energy depletion. Link and node errors may lead to network partitioning. Such errors should be corrected in order to restore network connectivity. Various path-recovery algorithms have been proposed in this respect. This study analyses some suggested algorithms and suggests a few new recovery algorithms. The path recovery algorithms are categorized as either global or local. In global path recovery, the paths are generally recovered during periodic global network updates. Thus, high path-restoration delay may result in networks where the global updates are infrequently run. The local recovery algorithms, on the other hand, are triggered by path breaks and have a local scope. Thus, the local algorithms result in low delay and affect a limited number of nodes. By minimizing the number of affected

nodes, the energy consumed during the process is reduced. However, the study findings show that the percentage of disconnected paths that are recovered is lower when using energy-efficient local recovery algorithms than when using global recovery algorithms. The possible trade-offs between local and global recovery are discussed.

Preface

This dissertation is submitted in partial fulfillment of the requirements for the degree Philosophiae Doctor (PhD) at the Department of Telematics, Norwegian University of Science and Technology (NTNU)

The work was supervised by Professor Øivind Kure at the Norwegian University of Science and Technology (NTNU) and Professor Knut Øvsthus at the Western Norway University of Applied Sciences (HVL).

Acknowledgements

I would like to thank my supervisors Professor Øivind Kure and Professor Knut Øvsthus for their advice and encouragement during these years. You have given me valuable feedback through discussions and guidance, feedback that has been paramount in order to realize this thesis.

I would like to thank my employer, Western Norway University of Applied Sciences, for financing my work.

My gratitude also goes to all my good friends and family, who have been listening to my frustrating jabber when things have seemed impossible. Our conversations have always been very encouraging.

Most importantly though, I am thankful that my two precious boys, Sindre and Håvard, have endured living with me even though I have been such a stressed and absent-minded 'mamma'. In addition, I am thankful for the support and encouragement my husband, Tom Inge Nesheim, has given me during these years.

List of Publications

The author of this thesis has the primary authorship of paper A-D. For paper E, the author of this thesis has contributed as a discussion partner and provided some basic ideas that initiated the study presented in the paper. All papers were published through peer-review conferences and workshops.

PAPER A: Anne-Lena Kampen, Knut Øvsthus and Øivind Kure," **Energy Reduction in Wireless Sensor Networks by Switching Nodes** to Sleep During Packet Forwarding", Proceedings of the 6th International Conference on Sensor Technologies and Applications (SENSORCOMM' 2012), pp. 189-195, 2012, ISBN: 978-1-61208-207-3

PAPER B: Anne-Lena Kampen, Knut Øvsthus and Øivind Kure," **Reconnection strategies in WSN running RPL**", 39th Annual IEEE Conference on Local Computer Networks Workshops (LCN2014), pp. 602-609, 2014, Electronic ISBN: 978-1-4799-3784-4, DOI: 10.1109/LCNW.2014.6927709

PAPER C: Anne-Lena Kampen, Knut Øvsthus and Øivind Kure," **An Analysis of the Need for Dedicated Recovery Methods and Their Applicability in Wireless Sensor Networks Running the Routing Protocol for Low-Power and Lossy Networks**", Proceedings of the 8th International Conference on Sensor Technologies and Applications (SENSORCOMM' 2014), pp. 121-129, 2014, ISBN: 978-1-61208-374-2

PAPER D: Anne-Lena Kampen, Knut Øvsthus and Øivind Kure," **Energy balancing algorithms in Wireless Sensor Networks**", Proceedings of the 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), Volume 5 , pp. 1223-1231, 2015, Electronic ISBN: 978-8-3608-1065-1, DOI: 10.15439/2015F67

PAPER E: Knut Øvsthus, Espen Nilsen, Anne-Lena Kampen and Øivind Kure," **Modelling the Optimal Link Length in Wireless Sensor Networks for Two Different Media Access Protocols**", Sensors & Transducers Volume 185. Issue 2, pp. 21-28, 2015, ISSN: 2306-8515

Contents

List of Figures

List of Terms and Acronyms

ACK	ACKnowledge
AODV	Ad-Hoc On-Demand Distance Vector Routing
CAP	Contention Access Period
CFP	Contention Free Period
CPU	Central-Processor Unit
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTP	Collection Tree Protocol
CTS	Clear To Send
DAG	Directed Acyclic Graph
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DODAG	Destination Oriented Directed Acyclic Graph
DSR	Dynamic Source Routing protocol
ELT	Expected LifeTime
ETX	Expected number of Transmissions
GTS	Guaranteed TimeSlots
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
IP	Internet Protocol
LEACH	Low-Energy Adaptive Clustering Hierarchy
LPL	Low Power Listening
LQI	Link Quality Indication
MAC	Medium Access Control
M2M	Machine-to-Machine

OF	Objective Function
OSI	Open System Interconnection
PAN	Personal Area Network coordinators
PDR	Packet Delivery Ratio
QoS	Quality of Service
RFC	Request for Comments
ROLL	Routing over Low-power and Lossy Networks
RPL	IPv6 Routing Protocol for Low power and Lossy Networks
RREP	Route Reply
RREQ	Route Request
RTS	Request To Send
Rx	Receiving
S-MAC	Sensor-MAC
SMACS	Self-Organizing Medium Access Control for Sensor Networks
SPOF	Single Point Of Failure
TDMA	Time Division Multiple Access
TORA	Temporally Ordered Routing Algorithm
Tx	Transmission
TTL	Time To Live
WSN	Wireless Sensor Networks

PART I

Introduction

Chapter 1 - Introduction

1.1 Background

Internet of Things (IoT) is expected to be one of the major new sources of traffic on the Internet. According to Cisco's predictions, 50 billion devices will be connected to the Internet in 2020 [1]. Of the IPv6 capable devices, machine-to-machine (M2M) is a key segment, expected to reach 1.5 billion by 2020 which is an 11-fold increase from 2015 [2]. As an M2M solution, IoT will offer seamless communication of billions of objects for control and monitoring of sensors. The sensors will range from expensive sensors in a fixed infrastructure, to simple battery operated sensors interconnected by short-range wireless technologies. The latter, called Wireless Sensor Networks (WSN), is the topic of the thesis.

WSN is an interconnected collection of small electronic-sensor devices. The sensor data is forwarded from the sensors to one or more collection nodes, called sinks. In most cases, the sinks and sensors are not in direct contact, and the sensors must forward data on behalf of each other. In this way, sensed data can be transmitted across the network. The sinks can be part of a fixed infrastructure and will typically contain more functionality, processor capacity, and storage resources than the sensor nodes. The rest of the nodes in the WSN, the sensors, have reduced functionality. They are normally low cost devices with limited memory, battery, and microcontroller capacity.

Although the battery capacity of the nodes is limited, the WSNs may operate for months or even years. The reason is that the energy consumed for processing is low. WSN supports large network deployments at limited management cost. These advantageous factors, in addition to the small physical size of the nodes, make WSN suitable for a broad range of applications. The application areas range from industrial processes [3], geriatric care [4], environmental surveillance [5], to military solutions [6].

1.2 Motivation and Challenges

The lifetime of the nodes is individual, since it depends on the detailed operations of the node, the amount of data forwarded, and the amount of processing performed. The depletion of the energy in an individual node affects the whole of the WSN; it reduces

the sensor data collected, since the depleted sensor no longer generates data. In addition, it affects the rest of the nodes, since a depleted node cannot forward data from other nodes. The result may be a higher forwarding load on other nodes affecting their lifetime, and at worst, a partitioning of the network.

Minimization and balancing of energy consumption is important in order to lengthen the lifetime of the individual nodes and the WSN itself. However, due to both the cost and energy optimization, nodes typically have low-power microcontrollers with limited processing power and limited memory. This restricts the complexity and functionality of possible control algorithms.

This thesis addresses various aspects of energy consumption in WSN. This is a critical topic [7], since all operations in the WSN affect the energy usage. Communication is typically one of the major energy consumers. As part of the communication, the radio switches between different states such as receiving, transmitting and listening/idle, each with a different level of energy consumption. In addition, to save energy, the radio may enter the sleep state whenever possible. Processing performed by the microcontroller consumes energy from around one to a few nJ per instruction [8]. A comparison, [9] of the energy consumed by the MSP430 microcontroller and the CC2420 radio shows that in the active state the radio consumes ten times more power than the microcontroller. The graph presented in [10] shows that a thousand compute cycles by the MSP430 microcontroller consume the same amount of energy as transmitting one byte over the CC2420 radio. Hence, the microcontroller´s energy consumption can be ignored if it is assumed that the radio and microcontroller are active at the same time. The sensing operation performed by the sensor consumed energy. However, the amount is a function of the application and is therefore out of the scope of this thesis where the focus is on common aspects.

This thesis focuses on the radio since it is the main energy consumer [11]. The minimization of the energy usage of the radio in WSN is a broad field with many different approaches. Energy usage is a function of a variety of factors: the time a nodes stay in the different states, the transmission power, the number of packets overheard, and the amount of traffic routed through it. Some of these factors are dependent. For example, increasing the transmission power reduces the number of hops to the sink and therefore the forwarding traffic. However, it also increases the number of nodes that overhear a particular packet. Different approaches to energy efficiency target one or more of these issues. The various approaches have been classified into three categories according to the main focus: optimization of the state (sleep protocols) [12], approaches aimed at optimizing the number of neighbors (topology control) [13] and balancing of the traffic over the various nodes [14]. These specific categories were chosen since they reflect the factors that have a large impact on energy consumption. The sleep state is the most energy-efficient state a node may enter; hence, sleep protocols have a very high impact on energy consumption. Receiving and transmission consume energy of the same order of magnitude and are the states where the nodes consume the highest amount of energy. Hence, topology

control efficiently optimizes the energy consumption. Energy balancing is advantageous since it prevents early depletion of nodes.

Sleep protocols save energy by forcing the node to enter the sleep state as often and for as long as possible under different constraints; for example meeting offered load, minimizing delay and so on [12]. The sleep protocols can be divided into two main categories. In the first category, the nodes individually decide their own sleep/active schedule, so a node with data to transmit does not know when the receiver is awake and ready to receive. The transmitting node, therefore, transmits a signaling message ahead of the data packet to inform the receiver of the upcoming data transmission. The signaling packet must extend over a longer period than the longest allowed sleep period. The strength of this protocol is that it reduces idle energy consumption; the weakness is that the energy consumed for transmission is increased.

In the second category of sleep protocol, the nodes agree on a common sleep/active schedule. Generally, the schedule is locally controlled. Nodes choose to follow the first schedule information received from a neighboring node. Nodes with data to transmit start the transmission exactly when the receiver wakes up. This significantly reduces the transmission energy consumption compared to the protocols of the first category. However, periodically generated management packets must be exchanged between neighboring nodes in order keep the sleep/active periods synchronized, and periodic packet exchange consumes energy.

Topology control protocols adjust the transmission range in order to reduce the energy consumption needed to transmit a packet end to end. Multihop WSNs are assumed, hence forwarding drains radio energy for each intermediate hop along the path from source to sink. The optimal number of hops is a trade-off between transmission range and number of overhearing nodes on one hand [15], and number of hops on the other hand [16]. In addition, the Packet Delivery Ratio (PDR) influences the optimal number of hops in a network prone to packet loss [17].

As well as minimizing the total energy consumed for transmission, it is important to balance the energy consumption in the network to avoid early depletion of nodes. Energy balancing algorithms focus on how sensor data is forwarded over the topology. The various protocols apply different metrics that aim to create an even energy balance between the nodes under different constraints such as robustness against failures [14, 18] and delay [19].

Given the broad scope of possible themes, a thesis must focus on a limited subset. The starting point is therefore a study of energy usage in WSN. The articles can be subdivided into investigations of 1) energy usage in WSN, 2) energy consumption balance (which can be considered a subtopic of energy usage), and 3) effectiveness of various path recovery mechanisms. The latter of these is motivated by the fact that network connectivity is a critical aspect of WSN that may be destroyed due to depleting nodes. Path recovery mechanisms aim to reconstruct broken paths.

The first issue is to identify the energy usage of the various operations in the radio. The distribution of energy consumption for network functions was analyzed for a typical chipset used in different scenarios. The result was used to suggest an algorithm that reduces the amount of energy consumed for transmission from source to sink.

The second issue is the energy consumption balance in WSN. The goal of energy balancing is to smooth out lifetime variations between the nodes, in order to avoid early depletion of nodes. This study´s contribution to this field is a broad investigation of previously proposed algorithms to balance the energy consumption. Based on the results, new algorithms have been formulated and evaluated.

The third issue is network connectivity. This is important in order to ensure that the sink receives all sensor data. Depleted nodes can create path breaks if they are to forward traffic for other nodes. Hence, data paths over broken links must be rerouted. This thesis contributes by investigating the impact of depleted nodes on the connectivity of the routing paths. Moreover, new algorithms are suggested and some of the proposed recovery algorithms are evaluated. Investigations on this particular aspect were inspired by solutions for ad-hoc networks. WSN and ad-hoc networks have common features, since they have no fixed infrastructure and the nodes are required to forward data on behalf of other nodes [20]. However, there are major differences in the traffic patterns and the emphasis on energy efficiency. The traffic in WSN is generally directed toward the sink, while peer-to-peer communication is common in ad-hoc networks. In addition, the nodes in WSN have less energy, memory and central-processor unit (CPU) capacity than nodes in ad-hoc networks. Hence, solutions used in ad-hoc networks must be adapted to suit the characteristics of WSN.

1.3 Overview of the work

This section presents the papers included in the thesis. Before a detailed description, a short summary of the relation between the articles and a timeline of the development is presented.

Initially, the aim was to study energy consumption in WSN, and this is therefore the topic of paper A. It was considered that overhearing was responsible for an important part of the nodes' energy consumption. Overhearing is the process in which nodes receive and read data that is not addressed to them. The results suggest that depletion of nodes is unpredictable due to overheard traffic. In order to reduce the unpredictability of energy consumption due to overhearing, the number of neighbors should be reduced. However, a reduction in the number of neighbors means that the number of backup nodes able to mend path breaks is reduced. Thus, path breaks are likely to occur in WSN and path recovery is an important issue. These observations motivated the work of paper B, in which recovery algorithms are investigated. The main result of the investigation is that high path-recovery probability requires high message-exchange overhead. Paper C further elaborates on the issues raised in paper B.

Both papers B and C focus on the connectivity between the nodes and the sink. The results of paper C show that, whatever the node density, some nodes are likely to select next-hop nodes (parent nodes) in such a way that they are left with no backup paths. Thus, the nodes selected as parent are very important for network connectivity, and the early depletion of such nodes should be prevented. However, since parent selection is a distributed process any node may become a parent. Thus, early depletion of any node should be prevented. These findings motivated the investigation into the issue of energy balancing between nodes, which is the topic of paper D. Balancing prevents early depletion of parent as well as other nodes, thereby preventing early path breaks. The findings reveal that the greatest balancing effect is achieved when nodes consistently transmit data through the parent node with the highest amount of residual-energy. However, a significant balancing effect is also achieved with only a small change in the next-hop-node selection algorithm.

In paper E, we collaborated with a research group to revisit the topic of paper A. Paper E is an addition to the thesis since the author participated, but was not the principal contributor. Fig. 1 shows a graphic overview of the issues addressed and publication years of the papers.

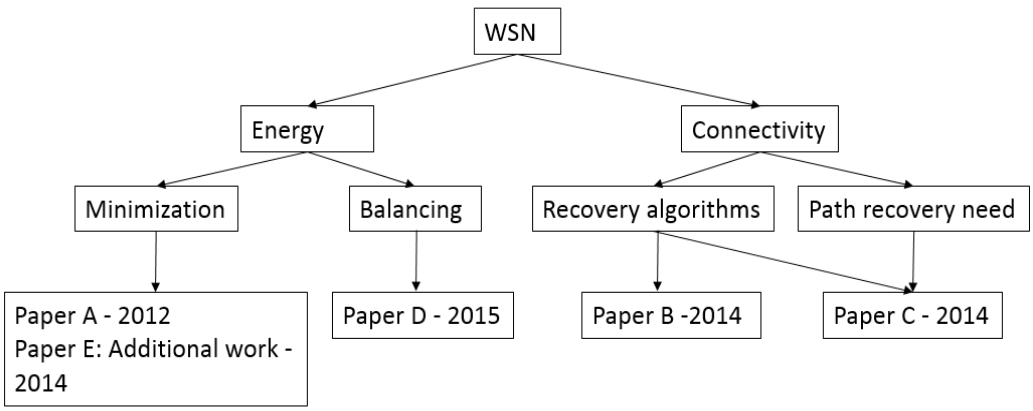

Figure 1 Overview of papers and related issues

In paper A, the energy consumption was investigated with regard to the various states of the radio (transmission, receiving, idle, and sleep) when packets traverse several hops. The idle energy consumption varies according to the Medium Access Control (MAC) layer algorithm. However, it is assumed that the sensors in the WSN are homogeneous, and the same MAC layer protocol is used so that the idle power consumption does not affect the relative power among the nodes. The lowest energy state is, as expected, the sleep state. The highest energy usage states are the receiving and transmission states. Both have energy consumption in the same order of magnitude. Since this equality has not been explored thoroughly in the literature, it was decided to investigate this issue further. In particular, the energy-optimal transmission range applicable for forwarding from source to sink was studied.

The equality between the energy consumed during receiving and transmission influences the energy-optimal transmission range. An increase in the transmission range reduces the number of forwarding's needed to reach the destination. However, the energy consumption for receiving increases with transmission range since the number of overhearing nodes increases.

A model is suggested for calculating the energy-optimal transmission range when transmitting from source to sink. The calculated optimal range is too short to keep the network connected. The optimal transmission range is, therefore, the shortest range needed to avoid network partition. Based on the equality of energy consumption between transmission and receiving, a refinement of the sleep protocol is presented in the paper.

Based on the findings in paper A, it is clear that nodes will deplete their energy at different epochs. The energy consumption of the nodes is unpredictable, since it depends on several unknown factors such as the amount of data generated, the amount of forwarded data and the overheard traffic. In addition, there are also predictable factors; for example, nodes closer to the sink will typically deplete faster than nodes at the edge. Hence, to maintain connected networks, algorithms that can facilitate path recovery following breaks due to depleted nodes are required. In addition, paths may fail due to changes in the radio environment or errors in software or hardware in the nodes. Path recovery algorithms are event-driven since they are triggered by path failures. To some extent, they are dependent on the routing scheme used to create the paths. Our investigation is based on the routing protocol proposed by the Internet Engineering Task Force (IETF): IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [21]. In paper B simulations are used to evaluate such algorithms, and to suggest and evaluate a possible improvement. The findings are that the local recovery algorithms do not always find a new path. In order to guarantee that disconnected paths are recovered, a potentially high number of nodes must participate in the recovery process.

As observed in paper B, the reduction of the recovery area reduces the success rate of the recovery algorithms. In addition, the introduction of recovery algorithms consumes memory space and adds to the computational overhead. Motivated by these observations, the probability for the need for recovery algorithms was calculated, and an on-demand total-network recovery algorithm is suggested in paper C. Naively, one might expect the path breaks to be rare events. After all, sensor networks are supposed to be high-density networks with many alternative paths. However, the calculations and simulations carried out reveal that path breaks are likely to occur somewhere in the network for all investigated node densities.

By balancing the energy consumption between the nodes, the number of times recovery algorithms must be activated is reduced. The result is a more stable network and energy, memory space and processor capacity are saved. In paper D algorithms that aims to balance the energy consumption between the nodes are investigated. Based

on experience acquired working on paper A, the focus is on balancing the energy consumption related to the transmission and receiving of packets. The contribution of paper D is to review a broad range of the balancing algorithms found in literature in a common context. The algorithms have been evaluated previously, but not in the same scenario and network environment. Used in the same scenario, their relative strength and weakness can better be evaluated. In addition, three new balancing algorithms are suggested. The result of the simulations shows that forwarding through the next hop node with the highest residual-energy gives the best balance.

The result of paper A indicates that the transmission range should be short to minimize energy consumption. However, short transmission range means that the path consists of high number of links. Each link added increases the expected number of forwardings required. Hence, forwarding can limit the benefits gained from the reduced number of overhearing nodes. The number of retransmissions increases when the receiving node approached the border area of the sender's transmission range. To evaluate the impact of such retransmission, a function is suggested that relates the PDR and the distance between communicating in paper E. The function may be used when estimating the energy-optimal transmission range.

1.4 Methods

Analytical methods, simulations and testbed experiments can be used to investigate WSN. Analytical methods can generate fundamental knowledge of system behavior, demonstrate general principles that apply, and validate the impact of defined algorithms. Such knowledge lays the foundation for further research and development. The challenge with analytical methods is to crate abstractions that truthfully represent the system under investigation. WSN represents a complex system of nodes interacting with each other and the environment; in addition, the nodes are running several interacting protocols. In order to perform an analytical analyze on such complex systems, there is a trade-off between creating complex abstractions that are not tractable or tractable abstractions that are too simple. On the one hand, realistic abstraction for WSN requires the modeling of a wide range of mechanisms, phenomena and parameter values. It is challenging to identify the right abstraction since many of them are dependent on both local and environmental conditions. Moreover, several parameters are interdependent. On the other hand, abstractions that are too simplified do not represent the characteristics of WSN well enough to provide valuable analytical results.

In paper A an analytical model is presented that calculates the energy consumption for transmission from a source node to the sink node. As the issue under investigation was energy, the focus was on the main energy consuming part of the node, which is the radio. The energy consumed for receiving and transmission was used, therefore, to model a simplified abstraction of the nodes. The consumption for transmission from source to sink was calculated based on transmission range, number of transmissions

and number of nodes receiving each transmission. Analytical analysis was used in order to present a general evaluation that was not connected to any specific protocol. Similar energy calculations were included in the additional work in paper E. However, paper E includes a model that relates the packet loss probability and node distance.

In paper C analytical calculations are used to evaluate the probability that a local recovery algorithm is needed to mend path disconnections. The nodes were assumed to be uniformly distributed inside the network area and Poisson was used to investigate whether a recovery node exists if a former next-hop node is depleted. The analytical calculations were validated against simulations in paper C.

Simulations facilitate the investigation of numerous and varied scenarios at a very low cost. Hence, simulation is a powerful tool to validate the scaling ability of an algorithm. In addition, simulation makes it possible to investigate distributed interactions throughout the network. Moreover, simulated results are easy to reproduce, enabling verification of findings. However, the models used in simulation are abstractions of real-world equipment and ambience. Thus, all of the characteristics of real-world interrelationships and phenomena are not replicated. In addition, pitfall such as unrealistic scenarios and improper parameter values can compromise the credibility of the simulations [22].

Simulations were used in the studies reported in papers A, B and D to test, verify and compare the characteristics of various algorithms. All simulations were generated using Omnet++[23] with MiXiM added for wireless communication. The focus of the simulations was the MAC and the Network layer. To validate the simulations, scenarios were employed in which nodes were deployed in familiar structures and the simulation results were easy to predict. For instance, the basic characteristics of the RPL were implemented and the implementation verified using nodes deployed in a grid structure, in which the forwarding structure is well known. Subsequently, RPL was simulated over larger and larger networks of randomly deployed nodes to ensure that the graph develops as expected.

Multipoint-to-point traffic was assumed since this is the main traffic pattern in industrial and urban low-power networks, according to [24] and [25]. Given the focus on traffic directed toward the sink, only this part of the RPL algorithm was implemented in Omnet++.

Standard statistical methods were employed in order to make the simulation results statistical valid. Every simulation point presented in the graphs is the average value based on 30 to 60 simulation runs with different seeds. The 95 % confidence interval is shown in the figures or it is made explicit in the text.

At the physical layer standard Omnet++ was implemented with the standard Boolean disk model; this is clearly a simple abstraction of real-world communication. Thus, the simulation results do not represent true behavior but they should indicate a valid trend.

In addition, in paper E a Fermi-Dirac function is provided to express the relation between PDR and the distance between the communicating nodes.

The algorithms investigated in papers A and D were supposed to be general, fitting any network shapes. The simulation scenario therefore consists of randomly deployed nodes. The energy consumption of the nodes was calculated based on the number of transmitting and receiving packets, as well as the size of the packets. In order to avoid edge effect in the simulation of paper A, the source node and the sink were located at a distance from the edge of the simulation area that was longer than the longest transmission range. To validate the results of paper A, the results were compared with published and analytic results. The simulation results of paper D were validated against published results.

Path breaks were investigated in paper B, with a node being deliberately deactivated to create the path break. However, a deactivated node in a randomly deployed network may not create a path break. Thus, to ensure that all simulation runs contribute data related to the issue under investigation, randomly deployed nodes were prohibited in the area surrounding the depleted node. The prohibition area ensures that nodes further out in the network do not have a backup path to cover for the depleted node. The simulations in paper B were validated against calculations.

Testbed experiments are extremely useful in providing a true demonstration of the outcome of a suggested algorithm, especially if tested in an environment authentic to the environment in which it will be used. This means that environmental characteristics such as topography, humidity and temperature are authentic. However, such testing is very expensive in both man-hours and money, particularly if multiple scenarios and environments are to be investigated. The latter is essential, since there is no single typical environment for WSN and its various applications. In addition, it is more challenging to identify the elements that affect the test results. In the real world, it is not always easy to isolate the various environmental sources of impacts.

It would have been beneficial to perform testbed implementations to confirm the simulated results. However, the study results are in several areas, and the resources necessary to implement these in a testbed are not available.

1.5 Thesis outline

This thesis has four chapters. Chapter 1 is the introduction, including background, motivation and an overview of the research. The second chapter provides an overview of the WSN and research relevant to the work. The first part presents an overview of the RPL routing protocol[21] ; RPL is an IETF standard open routing protocol and therefore lays the foundation for the work. The rest of chapter 2 describes the challenges addressed in this thesis. Energy consumption is described first since it is the basis for the research. Then follows a discussion of algorithms that are intended to reduce and balance energy consumption, and of algorithms that assess network

connectivity, as well as path-recovery algorithms. Thereafter, Chapter 3 presents the contribution this study makes to the research field, and Chapter 4 contains the conclusion.

Chapter 2 - Related Work

This section summarizes the state of the research relevant for the three topics investigated. There are separate sub-sections discussing the various energy-related issues: energy consumption models, energy conservation strategies, and energy balancing in WSN. In the last few sections, relevant findings related to network connectivity and path recovery for WSNs are summarized.

Given the assumption of multihop WSNs, a routing protocol is needed in order to create paths for data transmission. In [26], the IETF Routing Over Low power and Lossy networks (ROLL) research group discusses the usability of various known routing protocols for low power and lossy networks. The research group concluded that a new protocol specification is needed. Thus, in August 2009, IETF ROLL presented the first draft of the Routing Protocol for Low Power and Lossy networks (RPL). This draft evolved into the RFC 6550 in 2012 [21]. The RPL routing protocol is applicable to a wide range of wireless low power networks[27]; it is frequently cited in the literature and has been tested in both simulations [28] and testbed [29] situations. Based on these facts, RPL [21] was selected as the basic routing protocol for the research. The first sub-section below, therefore, outlines RPL.

2.1 Presentation of RPL routing protocol

RPL [21] is the routing protocol recommended by IETF to provide multipoint-to-point traffic and point-to-multipoint traffic over constrained nodes and interconnections. Point-to-point traffic is also supported. It is a proactive hierarchal soft-state routing protocol. As pointed out in 1.4 , the research focus is on traffic directed toward the sink. The following presentation of RPL is not comprehensive, therefore, but rather a description of how RPL supports multipoint-to-point traffic.

RPL generates an overall Destination Oriented, Directed Acyclic Graph (DODAG), that is directed from the nodes toward the root (sink). The overall DODAG is defined by the routing entries cached in each node constituting the network. The sink manages a DODAGVersionNumber that defines the DODAG version. The DODAGVersionNumber is used to renew the DODAG, and all nodes that are members of a DODAG share the same DODAGVersionNumber.

The rank is a parameter that is used to define a nodes distance from the sink, so that the rank increases as the distance along the shortest path toward the sink increases. The sink's rank is zero.

The sink initiates the formation of the DODAG by emitting DODAG Information Object (DIO) messages. The DODAGVersionNumber and the rank of the sender is part of the information carried in the DIO. A node selects its successor node (parent) based on information gathered through the DIOs received. The parent node is the neighbor that displays the lowest rank and the most recent DODAGVersionNumber. Thus, the nodes receiving the DIO transmitted by the sink select the sink as parent. The nodes calculate their own rank based on the rank of their parent, and the metric-based cost-of-path between itself and its parent. Subsequently, a node re-emits the DIO received from its parent. However, note that the rank information carried in the re-emitted DIO is the rank of the sender. The process is repeated throughout the whole network creating the network-wide DODAG that is rooted at the sink.

The nodes may cache a parent-list containing neighbors that display a rank equal to the lowest rank heard. When parent-lists are used, a preferred parent is selected from among the members of the parent-list. The preferred parent is the current successor node on the path toward the sink. The other members of the parent-list become back-up parents in case the current preferred parent is depleted or otherwise becomes unavailable.

Network faults may change the network topology and create path breaks that RPL is intended to mend. However, to prevent routing loops, the nodes are not allowed to move outward in the DODAG. In other words, a node that is a member of a specific version of a DODAG cannot increase its rank as long as its DODAGVersionNumber is unchanged. If such rank increase were allowed, a node could choose a former child node as a new parent and create a forwarding loop. However, a node may increase its rank if it receives a DIO from a node that is part of a more recent DODAG. A more recent DODAG is indicated by an updated DODAGVersionNumber, and shows that the receiver is not part of the sender's path toward the sink. The reason is that the version number set by the sink is propagated unchanged throughout the DODAG.

RPL uses trickle timer [30] to facilitate the rapid exchange of new information in unstable networks, while keeping the information exchange to a minimum in stable, converged networks. Using trickle timer, the nodes transmit DIO messages at exponentially increasing intervals. A node may suppress transmission of the DIO if it receives already transmitted information.

The simulations presented in [28] show the expected saw-tooth pattern that the management-packet overhead (DIO overhead) creates over time when trickle timer is used. The number of management packets exchanged increases quickly when new a DODAGVersionNumber is published to update the DODAG, because the trickle timers are reset to the lowest value. The number of management packets drops quickly when the network stabilizes. The trickle timer reduces the overhead cost during stable

periods. In [31], the overhead is measured for a testbed of 50 transmitting nodes running TinyRPL. TinyRPL is the implementation of RPL on TinyOS2.0. The minimum trickle interval is 0.128s and the maximum interval is 512s. In the first setup, each source transmitted a data packet with 5s intervals; and in the second setup, the sources transmit a data packet with 10s intervals. In the first setup, the packet reception ratio was 99.88% and each node generated on average 8.96 management packets (DIO) per hour. In the second setup, the figures were 99.96% and 9.01 respectively. Measurements running the Collection Tree Protocol (CTP), which is the most common TinyOS protocol, were conducted in order to compare RPL and CTP. The measurements show that CTP has slightly better values, 99.04% and 8.29, and 99.99% and 08.02 respectively. It is assumed that the difference is the result of the difference in link quality estimation techniques that meant that the TinyRPL experienced a slightly higher parent exchange frequency than the CTP. Parent exchange resets the trickle timer if a nodes rank is changed. However, the advantage of RPL over CTP is that it can support various types of traffic pattern, as well as IPv6.

RFC 6550 [21] describes the basic mode of RPL operation. Optimizations are defined by other documents. For instance, the Objective Function (OF) [32], in conjunction with an applied metric [33], can be used to calculate the rank of the nodes. OF also decides how parents are selected. For instance, the nodes in networks that run the RPL objective function zero (OF0) [32] select a feasible successor in addition to their preferred parent. The feasible successor is a neighbor that can be used as parent according to the loop avoidance rules in the network. It acts as a backup route if the preferred parent is unavailable. Thus, the nodes caching feasible successors have multiple available paths toward the sink. However, it is not compulsory for the nodes to have feasible successors. For instance, nodes lying on a line along an isolated branch of the routing graph will not have any feasible successor nodes. In addition, the optimal DODAG shape may deteriorate since nodes may need to increase their rank in order to enable neighboring nodes to become feasible successors. Such rank increase may cause instabilities and routing loops since nodes move to use each other as parents or feasible successors.

Throughout the simulations and calculations in this study, the hop-count metric was used. The reason for this choice is that this is a well-known and frequently used metric that creates a quite stable graph. In addition, it is easy to check that the routing protocol is functioning without flaws. An alternative measure, the Expected Number of Transmissions (ETX), was considered but was rejected in favor of the more stable hop-count metric. ETX is the expected number of transmission needed to successfully deliver a packet; it varies, therefore, when the link quality changes. Simulation results for both ETX and RPL are presented in [28]. The simulations reveal that hop count occasionally chooses a path that is not the path with the smallest ETX. Nevertheless, the more stable hop-count metric was preferred for the present study.

One of the simulation results presented in [28] shows that the use of RPL with hop-count metric is very valuable, because RPL selects the path that is almost equal to the

ideal shortest path. The result is based both on point-to-point and multipoint-to-point traffic. The network consists of 45 nodes and the longest distance between a node and the sink is four hops. To support point-to-point routing, RPL allows child nodes to provide information about known prefixes; i.e. destinations. Point-to-point packets are forwarded toward the sink before being transmitted outward to the destination. The simulation shows that 90% of the RPL paths are equal or less than 5 hops, while 90% of the ideal shortest paths are equal or less than 4. The good performance of RPL is to some extent due to the location of the sink in the middle of the simulation area.

The RPL convergence process is investigated in [34]. In particular, the focus is on the impact of the redundancy constant, which is used to suppress DIO transmission. The redundancy constant is an integer that has an associate integer counter value. When a node receives DIOs with information that is consistent with its own information, it will increment the counter value. If the counter value is equal to or higher than the redundancy constant at the time the node is about to transmit a DIO, the DIO transmission is suppressed. Otherwise, the information is transmitted. Simulation shows that an increase in the redundancy constant reduces the convergence time [34]. This is because the probability of DIO transmission increases when the redundancy constant increases, which in turn increases the probability for neighboring nodes receiving DIOs early. Simulations were performed using IEEE802.15.4 beaconless Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mode; the number of nodes was 34-99, the number of neighbors was 5-15 and minimum trickle interval was 8ms. The results show that, if the redundancy constant increases from 1 to 2, the convergence time is reduced from 120s to 18s for 80% of the simulation scenarios. However, increasing the redundancy constant beyond the value for the number of neighbors has negligible effect. The minimum trickle interval, that is the minimum time before a node transmits a DIO after receiving a DIO with new information, also affects the convergence time [34]. Thus, a reduction to the interval reduces the convergence time. However, the increased number of collisions limits the positive impact of reduced minimum interval.

The terminology used in this thesis is based on that used in RFC6550. For instance, successor nodes are called parent nodes, and when several successors exist, these are called members of the parent-list.

2.2 Energy consumption in WSN

One of the main factors limiting the lifetime of WSN is the energy consumption pace of the nodes. As nodes are depleted, the data gathered at the sink becomes incomplete, and parts of the network may become unavailable due to network partitioning. Thus, it is important to limit the energy consumption in order to prolong the nodes lifetime.

The nodes contain different energy consuming components such as the microcontroller, the sensor unit and the radio. The energy consumption of the sensor units is not included in the assessments in this study since it is independent of network functionality and varies with sensor type. The microcontroller and the radio are the

components that participate directly in traffic forwarding. Since the radio is the main energy consumer [11], that is the focus of this investigation.

During operation, the radio switches between different states such as Transmission (Tx), Receiving (Rx), idle and sleep. Each radio state consumes a different amount of energy. However, due to the generally short range used in WSN, the power consumed in the Tx state is the same order of magnitude as that consumed in the Rx state [35]. The sleep state consumes the least power as most of the transmitter and receiver circuitry is switched off. Values for the power consumption in the different states for AT86RF230 [36], CC1000 [37] and CC2520 [38] are shown in Table 1. Current consumption is proportional to power consumption since the voltage supply is the same for the different states. Observe that the current consumption for receiving is higher than the current consumed for transmission at the lowest available output power.

Table 1 Energy consumption for different radio states [36-38]

	AT86RF230	CC1000	CC2520
Rx	15.5 mA	9.6 mA	18.5 mA
Tx at max transmit power	16.5 mA	25.4 mA	33.6 mA
Tx at min transmit power	9.5 mA	8.6 mA	16.2 mA
Sleep	20nA	<105 µA	<1µA

2.2.1 Energy consumption models for WSN

Models describing the energy consumption in WSN can provide valuable insight into how the different radio states affect the energy consumed in the network. All significant energy consuming parts of the network should be included to create accurate models. The energy consumption models presented in this section are related the subject investigated in paper A.

The energy consumed in communication in a WSN is generally divided between receiving and transmitting data packets. The energy consumed in transmission can be divided in two categories [11, 39]. The first category depends on the architecture of the transmitter and can be modeled as a fixed value. The second category increases with increasing transmission power; i.e. increases with increasing transmission range. The energy consumed for receiving is modeled as a fixed value.

The power-amplifier drain efficiency is included to model power consumption in [16]. The drain efficiency is defined as the RF output power divided by the DC power consumed by the power amplifier. It is a factor that generally increases with output power, as demonstrated in [40], where it is measured for class A/B amplifiers suitable

for WSNs. The model presented in [16] is used to find an expression for the maximum power that a power-amplifier can consume in a single hop transmission before it is more efficient to split the transmission into several hops. In addition, the model is used to find the energy-optimal number of hops when transmitting over a given distance. Since the drain efficiency increases with the output power, the most energy-efficient transmission found in [16] is use of maximum transmission power; i.e. maximum transmission range and minimum number of hops. The optimal transmission is initially calculated for the simple communication of data packets. The overhead for applied MAC protocol is subsequently included, increasing the energy-optimal transmission range. The MAC protocol included is a sleep algorithm in which the communicating nodes use Request To Send (RTS), Clear To Send (CTS) and Acknowledge (ACK), while nodes that are not part of the communication only listen for RTS. However, packet retransmission is not taken into account. In addition, although there is a heavy focus on the power amplifier characteristic, there is no discussion of the reduced linearity in received signal that would occur when the output power of the radio approaches the maximum ratings. Reduced linearity means that the modulation technique must be simpler; i.e. the data rate must be lower in order to avoid an increase in bit error rate.

The model presented in [41] calculates the energy consumed to correctly receive one data bit. Specifically, the model presents the energy required to transmit one bit a distance of one meter. The reliability of the links, based on the modulation scheme and the fading state is taken into account. The model is used to identify the optimal transmission distance for successful one hop transmission. In addition, the optimal number of hops for multi-hop networks is identified. The conclusion drawn is that a transmission range that is too short or too long leads to poor energy efficiency. The reason is that the circuitry is the main energy consumer for short distances, and the forwarding nodes' consumption is therefore constant. However, at long distances the energy consumption that increases with transmission power starts to dominate, reducing the energy-optimal transmission range. A weakness of the models presented in [16, 41] is that they are based on a 1D node arrangement, where nodes are deployed in a line.

The energy-optimal transmission range is calculated for both 1D and 2D arrangement of nodes in [42]. For 1D calculations, the nodes are arranged in a line. Two different forwarding patterns are investigated. The first is when intermediate nodes simply relay data transmitted by a source node. The second pattern is when the intermediate nodes add their own information. The distance variable energy consumption of the transmitter, the fixed energy consumed by the transmitter, as well as the energy consumed for receiving, is taken into account. However, the energy consumed for receiving is counted only once for each transmitting node, so the energy consumed for overhearing is not taken into account. The results reveal an energy-optimal number of hops, which is clearly lower when the intermediate nodes add their own information since the amount of traffic transmitted in the network is increased. The energy

calculation is expanded to 2D scenarios, which results in a recurrence formula. For the 2D calculations the, nodes are deployed so that they forms circles around the sink. The nodes are uniformly distributed along the circumference of the circles and there is a one-hop distance between each circle. Given a number of source nodes (N) uniformly deployed at a given distance (R) from the sink, the recurrence formula can be used to find the energy-optimal number of intermediate circles; i.e. the number of hops, the optimal transmission range for each circle, and the optimal number of nodes in each circle. However, the energy consumed for overhearing is not included in the calculations. In addition, such a specific node deployment may not be feasible for real WSNs.

The energy consumption model presented in [43] considers receiving as well as transmitting, idle, and sleep energy consumption. The model calculates the energy consumed during the time that data is being transferred. It is assumed that the nodes that participate in forwarding enter the idle state when not transmitting or receiving. Nodes that are not participating in forwarding remain sleeping. Energy consumption using maximum range is compared to relaying through a second node allowing a reduced transmission range. In the calculations, the maximum-range transmission is assumed to consume about three times the energy of the reduced-range transmission. The calculations are concentrated on these two transmission ranges; i.e. transmission energy consumption is not changing continuously with transmission range, but is either maximum or reduced. The results indicate that it is more energy-efficient to use maximum transmission range under low traffic density because the energy consumption is dominated by the idle energy consumption, and nodes that are not participating in forwarding enter the sleep state. Thus, it is energy-efficient to increase the transmission range in order to reduce the number of nodes participating in forwarding. However, it is more energy-efficient to use reduced transmission range when the traffic density is high because the transmission energy consumption dominates the total energy consumption. The weakness of this model is that nodes are presumed only to consume energy when they are part of an active routing path; i.e. the energy that overhearing nodes consume is not taken into account. In addition, the management cost added to make nodes enter the sleep state while not part of the transmission is not taken into account.

A model that calculates the energy-optimal transmission range for broadcast traffic is presented in [44]. The network area is divided into hexagons with side length r. A node with transmission range r is placed in each vertex. The fixed part of the transmission energy consumption as well as the part that increases with increasing transmission range, are taken into account. Given a network area, the model calculates the energy-optimal r and the associated optimal number of nodes needed for flooding. The energy-optimal transmission range is calculated both with and without taking into account the energy consumed in receiving. The energy-optimal transmission range is the same in both cases. The reason is that the hexagons are adjusted according to the nodes' transmission range, so that the number of receivers is unchanged for changing

transmission ranges. Thus, a weakness of the model is that a node receives a fixed number of transmissions, regardless of the transmission range applied. This means that the number of receivers is constant for changing transmission ranges, which is not the case if the node positions are stationary while the transmission range changes.

It was noted that the energy consumed for receiving does not change with changing transmission range and node density in the models discussed above. Although the 2D models presented in [42-44] include the energy consumed in receiving, they assume that the consumption is invariant for changing transmission range. In [42] the energy consumed for receiving is counted only once for each transmitting node. Nodes that are not participating in transmission remain sleeping in [43]. In [44] each node receives a fixed number of transmissions, regardless of the transmission range applied. However, in networks with stationary nodes, the number of overhearing nodes increases with increasing transmission range and node density, thereby increasing the total energy consumed for receiving. This observation motivated the 2D energy consumption model that is present in paper A, where the total energy consumed for receiving changes with transmission range and node density.

2.2.2 Energy conservation strategies

Numerous approaches have been suggested to conserve energy in WSN. The various approaches can be categorized based on where they belong in the Open System Interconnection model (OSI model). Gathering the application, presentation and session layers into a common application layer, there are four categories: physical, MAC, network and application layer.

In the physical layer, the electric circuit setting decides the transmission range and therefor the network topology. In addition, the physical layer approaches focus on data rate, carrier frequency, modulation and so on. Higher order modulation can reduce the energy consumption since higher data rate reduces the transmission time. However, there is a trade-off since the synchronization cost increases for high order modulation and may dominate the energy cost [45]. The carrier frequency affects the energy consumption since the path loss increases with frequency. Moreover, wake-up radios can be used as a physical layer solution. The low-powered wake-up receiver monitors the communication channel and wakes up the main receiver when a predetermined wake-up signal is received [46]. Thus, the main receiver remains in the sleep state most of the time.

At the other end of the OSI model, the application layer approaches focus on approaches such as data compression and data aggregation; these reduce the network traffic, thereby reducing the energy consumed during transmission [10] [47]. Physical and application layer approaches are out of the scope of this thesis and therefore not discussed any further.

The MAC protocol determines how long a node persist in the various radio states. Therefore, the MAC protocols have been the subject of substantial research related to node energy efficiency.

However, efficient link operation does not preclude optimization at the path level; i.e. network layer approaches. The paths should both minimize the energy consumed and avoid low energy nodes. The latter approach reduces node depletion and possible network partitioning. In addition to the layered protocols, cross-layer design combines the functions and parameters of different layers in order to achieve energy optimization.

MAC and network layer approaches are discussed in this section. MAC layer approaches are discussed in 2.2.2.1 since they are important for energy conservation, and the algorithm presented in paper A falls into this category. Next, in 2.2.2.2, network layer approaches are discussed and metrics used to conserve energy are explored. The algorithms assessed and suggested in paper D fall into this category, although the algorithms in paper D mainly focus on balancing energy consumption between the nodes rather than reducing the total energy consumption. A brief presentation of topology control is then presented. The effect of varying transmission range is investigated in papers A and E. The section concludes with a short presentation of cross-layer approaches suitable for RPL running networks. Despite the potential benefits of cross-layer approaches, this study focuses, for several reasons, on energy conservation approaches that do not combine functions and parameters between layers. It was considered easier to exploit the potential of layered approach compared to cross-layer approaches because the exchange of components is more difficult when the layered approach is not complied. In addition, energy analysis is generally concentrated on mechanisms related to a single layer, although communication between layers adds computational and energy costs. However, a presentation of cross layer approaches is included in this section as it provides insight into the energy reducing potential of RPL. In addition, the contribution of the extended work, paper E, is associated with cross-layer approaches as it assesses the impact that MAC layer characteristics has on energy-efficient transmission range and hop count.

2.2.2.1 Mac layer approaches

The MAC layer protocols manage the internode communication since the MAC layer manages the state of the nodes. The communication should be efficient with low traffic delay and high throughput. In addition, the nodes' energy consumption should be low. The latter is the present focus.

A broad range of energy-efficient MAC layer protocols are suggested in the literature [48-50]. These protocols focus on idle listening, collisions, overhearing and overhead due to control packets. Collisions are a waste of energy that necessitates retransmissions. Overhearing wastes energy as nodes receive packets just to discard them. Transmitting and receiving control packets consumes energy. The idle state

energy consumption is often in the same order of magnitude as for the receiving state, and nodes in low data-rate networks can spend a lot of time in the idle state.

Sensor-MAC (S-MAC) [51, 52] and Low Power Listening (LPL) [53] (or B-MAC) are two key MAC-layer protocols. In these protocols, the nodes periodically enter the sleep state in order to reduce energy consumption. Hence, nodes alternate between active and sleep states.

Nodes running B-MAC periodically wake up to listen for activity. If activity is detected, the node stays awake for the time required to receive the transmitted packet. Nodes that have data to transmit start by transmitting a preamble, the length of which matches the longest known sleep interval among its neighbors. Thus, the energy expenditure for transmission is high due to the extra preamble sent. On one hand, the periodic energy consumed for receiving is small as nodes only wake up for a very short time to check for activity. On the other hand, nodes consume energy for listening through the rest of the preamble when activity is detected.

In order to reduce the energy consumed in nodes receiving preamble, it is suggested that the continuous preamble be replaced by a series of small frames that called micro-frames in [54]. The micro-frame contains the address of the destined node of the upcoming transmission, as well as the time schedule for the transmission. Hence, instead of receiving the whole preamble, the nodes enter the sleep state and only the intended receiver wakes up when the actual data packet is sent. Thus, the energy consumed by the nodes receiving preamble is reduced. However, a weakness is that the sender consumes energy for transmitting during the whole preamble period.

Another approach to reduce the preamble transmission time is presented in [55], where the receiver's wake-up time fixes the time for transmitting preamble. The nodes learn about the neighbors schedule through received ACK packets and use this information to transmit the preamble just as the receiver enters the active state. Thus, the energy consumed for both receiving and transmission of preamble is reduced. However, the weakness of this approach is the trade-off between the shorter preamble time and the stringent synchronization requirement. Stringent synchronization adds energy consuming overhead since periodic synchronization information must be exchanged. In addition, ACK is unicasted to the associated sender, thus the preamble time is only reduced for continuous transmissions toward the same next-hop node.

In [56], a time delay is introduced between short preamble packets. The preamble packets contain the identity of the receiver node. The intended receiver uses the time delay to interrupt the preamble transmission. The interruption is triggered when the receiver transmits an ACK back to the sender as soon as the receiver wakes up and determines that it is the intended receiver. Receiving the ACK, the sender immediately stops sending the preamble and start sending the data packet. This algorithm addresses the overhearing problem, reduces transmission energy consumption and reduces the per-hop latency. In addition, to improve the energy optimization, an algorithm is suggested that dynamically adjusts the receivers' duty cycle according to traffic load.

The duty cycle of a node is defied as its active time duration divided by its sleep time duration. Moreover, a way of reducing the need for preamble when the receiver is already awake is suggested: If a transmitter hears an ACK from the node it is about to send to, it will start the transmission just after the ACK, without using preamble. However, the energy consumption is increased compared to B-MAC during periods of no activity; this is due to the gap between the short preamble packets, which makes the nodes stay awake for a longer period than is otherwise needed in order to check the channel for activity.

To reduce energy consumption, the S-MAC [51, 52] protocol synchronizes the schedule for sleep and activity among neighboring nodes. During startup, a node enter the active state and listens for synchronization information from neighbors. However, the node sets its own schedule if no information is received after listening for a fixed time duration. The nodes periodically generate SYNC packets containing their schedule information, and neighboring nodes contend to broadcast their packets. The broadcasted information allows the neighboring nodes to adapt to the schedule of the sender. Neighboring nodes that adapt to the same schedule form a cluster. However, nodes may receive different schedule information from different neighbors, because the individual neighbors may have set their own schedules or have adapted to schedules set by nodes that are not within reach of each other. Nodes lying on the border between two clusters with different schedules (border nodes) must adapt to both schedules to avoid network partitioning. Thus, nodes are free to talk to each other although they belong to different clusters with different sleep/ active schedules. The clusters are therefore called virtual clusters. All nodes periodically listen for a whole synchronization period to detect whether they are border nodes.

The complete cycle of active and sleep states in S-MAC is called a frame [51]. The active interval of the frame is divided into two parts. The first part is for SYNC exchange and the second for data exchange. A node that wants to send a SYNC packet performs a carrier sense check during a random number of time slots. If no transmission is detected at the end of the last time slot in which it performs a carrier sense check, it starts transmitting the SYNC packet. After receiving or transmitting a SYNC packet, unicast transmission follows RTS/CTS/DATA/ACK between the sender and receiver. To save energy, overhearing nodes enter the sleep state just after receiving RTS/CTS. Broadcast data is sent without RTS/CTS.

Testbed experimental results to evaluate S-MAC are presented in [51]. A periodic active and sleep S-MAC with a 50% duty cycle is compared with 802.11. The basic operation of 802.11 is that nodes contend for channel access before transmitting packets. Received packets are acknowledged and, in the version used for comparison, nodes continuously listen for activity at the medium. The scenario consisted of four nodes located in the corners of a square with a relaying node in the middle of the square. Traffic was transmitted from one corner through the middle node towards a node in a second corner. The energy consumed by the three involved nodes was investigated. Compared to the network running 802.11, the sender and receiver running

S-MAC reduced their energy consumption both under light and heavy traffic load. Under light load, a message was generated every 10 second; and under heavy load, a packet was generated every second. The energy consumption difference was highest under light load where the 802.11 used more than three times the energy of the S-MAC. Under high traffic load, the 802.11 used about 1.5 more energy than S-MAC. S-MAC saves energy since idle listening dominates the consumption, and periodic sleep is therefore a crucial energy-saving factor. The relative duration of idle listening is lower in networks with high traffic load, so the difference between S-MAC and 802.11 is lowest under high traffic load.

The sleep time of the periodic active and sleep S-MAC can be increased to reduce energy consumption [51]. However, increased sleep time increases the delay. The latency occurs when a packet for transmission is generated by the higher layers during the sleep periods. The latency is further increased when queues build up in the nodes, because only one transmitter-receiver pair communicates during an active period. Hence, the frame size (complete cycle of active and sleep) must be balanced between delay and energy consumption. In order to reduce the latency, adaptive listening can be used in multihop networks [51]. Using adaptive listening, the nodes that overhear neighbor's RTS or CTS transmissions wake up at the end of the transmission. Thus, the data can be immediately relayed if the overhearing node is the next-hop for the exchanged data. Measurements of 11 nodes placed in a row, in which the first one is the source and the last one is the sink, show the benefit of using adaptive listening [51]. Without adaptive listening, the latency increases from about 1s for one hop to about 11s for 10 hops. When adaptive listening is used, the delay increases from under 1s to about 2.5s. The conclusion is that adaptive listening is very close to MAC without any sleep, where the delay increases from under 0.5s to about 1s. Hence, adaptive sleep reduces delay. However, it increases the energy consumption, as nodes must wake up more frequently to check whether they are the intended destination for the following transmission.

Several other solutions to the trade-off between delay and energy consumption are suggested in the literature. In [57], the traffic condition is used to dynamically change the sleep-active cycle. Receivers calculate the average one-hop latency values, and signals an increase of the cycle if the latency becomes intolerable. The duty cycle is increased, by introducing several active periods to split up the sleep period. Thus, nodes that increase their duty cycle increase the number of times they can receive, thereby reducing the delay. In [58], it is suggested that the nodes queue length can be used to introduce multiple adaptable active periods per SYNC- frame, while in [59] it is suggested that the duration of the active periods be varied based on the traffic load.

S-MAC wastes energy during the periodic active periods. An approach to reduce the energy consumption during the active periods is suggested in [60]. The energy consumed is reduced when a partition is established between the contention period and the listening period. Only nodes with packets to transmit are active during the contention period. The winner of the contention, the node that has chosen the shortest

random back-off time, transmits a signaling packet to inform other contenders that it gained access to the channel. The packet is transmitted in the following listening period. Nodes with nothing to transmit save energy by sleeping through the contention period. A weakness with this algorithm is that nodes that are sleeping through the contention periods must wake up in the listening period to check whether they are the intended receivers. However, overheard RTS and CTS exchanged in the contention period are means that can serve as signal to uninvolved nodes to enter the sleep state; that is, nodes that are not transmitters or receivers of the upcoming data transmission can enter the sleep state throughout the complete listening period. [61] focuses on minimizing the number of different schedules in the network to reduce the energy consumption of border nodes. These nodes consume more energy because they have to follow several schedules. In order to reduce the energy consumed by the border nodes, an algorithm has been suggested that makes all nodes follow the same schedule. Schedule-age is a parameter that has been introduced to indicate how long an associated schedule has existed, and nodes continuously adopt the oldest received schedule. Eventually, the whole network is applying the same oldest schedule. A weakness of the solution is that nodes must listen during an extended period in order to detect whether an older schedule exists in the network. In addition, the schedules of different parts of the network are likely to change over time since the schedule is managed by various random nodes in each neighborhood, and these random nodes have their own individual clock drift. Moreover, any use of dynamic change of the sleep-active period to trade between delay and energy consumption is unfeasible.

Time Division Multiple Access (TDMA) can be used to save energy. TDMA protocols share the medium by dividing the communication into consecutive timeslots [50, 62]. Each sender-receiver pair communicates in their own predetermined timeslot, which is repeated periodically. Each period is often referred to as a TDMA frame. Each sender-receiver pair is assigned one or more individual timeslots in which they can transmit and receive data. Thus, no collisions or overhearing occurs. Guard intervals between the slots are used to avoid synchronization errors. Clustering can be used to distribute TDMA management in WSN. A cluster head is selected in each cluster, and nodes can only talk to members of their own cluster. The cluster head manages the sleep/active schedule of each member of its cluster in order to minimize the energy used to transmit and receive. However, clustering may increase path lengths for transmission between nodes that are members of different clusters, because all paths go through the cluster head. Increase in path length means increased energy consumption.

The IEEE802.15.4 [63] protocol can optionally run Carrier Sense Multiple Access (CSMA) mixed with an operation similar to TDMA. In order to achieve this, the network must operate in beacon-enabled mode. In beacon-enabled mode, some nodes act as Personal Area Network coordinators (PAN) that periodically emit beacons. The time between two beacons forms a superframe; more precisely, the beacon is transmitted in the first slot of each superframe. The beacons synchronize neighboring devices and describe the superframe structure. The time between two beacons is called

the Contention Access Period (CAP). Using slotted CSMA-CA, nodes can communicate with each other during the CAP. To run in a similar way to TDMA, the PAN must introduce a Contention Free Period (CFP) in the superframe; the CAP is divided into a CAP period and CFP period. The CFP operates similarly to TDMA since it consists of Guaranteed Timeslots (GTS) managed by a fixed PAN. The PAN shares the GTS among the nodes. The GTS is allocated to a specific node and is primarily used for communication between the dedicated node and the PAN. The CFP is used by nodes with delay sensitive data or data requiring a specific bandwidth. The superframe can optionally have an inactive period in which the coordinators can enter the sleep state.

Other protocols that operate in a similar manner to TDMA are Low-Energy Adaptive Clustering Hierarchy (LEACH) [64] and Self-Organizing Medium Access Control for Sensor Networks (SMACS) [65]. LEACH divides the network into clusters controlled by a selected cluserhead node. The clusterhead maintains the TDMA schedule in its cluster. The member nodes can remain in the sleep state except during their dedicated time slot. SMACS is a distributed protocol in which nodes discover each other and agree on timeslots and frequency for transmission and reception. The nodes wake up and listen in the receiving time slot. The timeslots are part of a superframe that is fixed for all nodes. A challenge with SMACS is, therefore, the choice of superframe length; the length must be long enough to accommodate the highest number of neighbors, which may be unknown for a random deployed network.

Generally, TDMA has a great advantage in that it creates a collision free medium. However, it is energy consuming to maintain firm synchronization and to manage the slot activity. The nodes must periodically communicate to stay synchronized and low cost-per-node means that the circuitry providing the clock is generally prone to clock drift. Slot management and assignment are particularly challenging when the network topology changes, such as when nodes are added or removed.

Based on studying the MAC layer protocols discussed in this section, the overall observation is that nodes are switched to the sleep state to save energy in both LPL and S-MAC algorithms. In LPL, nodes manage their own individual sleep schedules, while in S-MAC nodes cooperate in order to agree upon a common schedule. Compared to the classic LPL algorithms, the improved LPL versions reduce the energy consumed by overhearing, while increasing the synchronization requirement or increasing the periodic energy consumed to receive. S-MAC employs duty cycling and RTS/CTS to reduce the energy consumed by overhearing. Delay is traded against energy consumption in the improved versions of S-MAC. However, it is considered that the related research lacks an algorithm that both reduces the overhearing energy consumption and, in addition, is simple enough to function as an add-on for most MAC layer protocols. A simple algorithm that may rectify this is suggested in paper A. The algorithm operates in a totally distributed pattern and reduces the overhearing energy consumption during transmission.

2.2.2.2 Network layer approaches

Energy-efficient network layer solutions seek to find the most energy-efficient routing graph. Specifically, for WSNs with multipoint-to-point traffic, the goal is to find the energy-optimal path between each node and the sink. As discussed in [66-68], various metrics have been proposed in order to create such optimal paths. These are metrics such as the energy consumption per packet, the residual-energy of the nodes and the path characteristics. The metrics typically decide how to calculate the cost of each individual hop or each individual node, and the cost of a path is calculated based on the cost of each individual hop or node, along the given path. The path cost is used to select routing paths.

When energy consumption per packet is used as the metric, the cost of a specific hop is calculated as the sum of the energy consumed to transmit and receive a packet over that specific hop. The cost of a path is calculated as the sum of the cost for each hop from the source to the destination. The lowest cost paths are selected to create the overall routing graph. The advantage of this metric is that the total energy consumption in the network is minimized. However, the nodes that are member of low-cost paths are preferred, so they have a high depletion rate due to heavy traffic load.

The routing protocol presented in [69] uses a version of the energy consumption per packet metric. The sum of the transmission power of all hops along each potential path is calculated in order to select the path with the lowest total transmission power. The potential paths are the paths in which all nodes have energy levels higher than a predefined threshold.

The metric which considers the residual-energy of the nodes aim to shift the selected paths away from nodes with low residual-energy levels in order to avoid early depletion of nodes. A version of the metric is proposed in [70]. The energy-cost between a node i and a node j is equal to the required energy for transmission from node i to node j, divided by the available energy of the transmitting node. The total path cost from a source node to the sink is the sum of energy-costs along the path, and the lowest cost path is selected. Because of the cost added, it will favor shorter routes. Thus, it can create imbalanced energy consumption in the network. For instance, two paths connect two areas, A and B, of a network as shown in Fig. 2. The shortest path consists of two nodes a and b. The longest path consists of five nodes, c, d, e, f and g. For simplicity, assume that the energy of node a is equal to the energy of node b, and that nodes c, d, e, f and g have equal energy. The energy consumed for transmitting a packet is equal for all nodes. Then, if the residual-energy of all nodes a – g is equal, the cost of the longest path is double the cost of the shortest path because it is double the length of the shortest path. Hence, the shortest path is selected, and will continue to be selected as long as the residual-energy of nodes a and b is higher than half of the residual-energy of node c, d, e, f and g. This favoring of the shorter path creates imbalanced energy consumption in the network. This imbalance is demonstrated in [18], where an improved cost calculation function is suggested; this function causes the

cost assigned to a node to increase rapidly as the residual-energy decreases. Hence, the cost function is improved by including the energy consumption rate. The aim of the improvement is to protect host spot nodes, nodes that have a high-energy consumption rate. However, the energy consumption rate of the nodes closest to the sink is always highest, and these nodes will therefore decide the path cost due to the high traffic load they experience. Thus, nodes further out in the network will not have a high enough impact to alter the choice of path, although they may have a high consumption rate compared to other nodes at an equal distance from the sink.

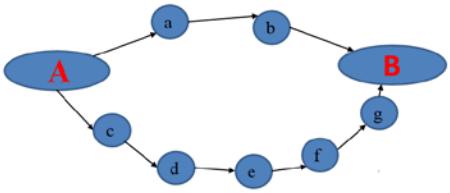

Figure 2 Shorter routes are favored when an additive metric is used

The path characteristics metric considers each path as a unit, and an important part of the path selection process is to compare the different paths' weakest points, which are often the lowest energy nodes. For instance, the highest cost path would be selected when the path cost is set equal to the residual-energy level of the lowest energy node along the path. Such a selection process may balance the energy level among the nodes. However, in contrast to the residual-energy of the node metric that may favor shorter routes, the path characteristic metric can result in poor performance since it may give preference to very long paths. Hence, the path characteristic metric should be combined with a distance related metric. [71] suggests an algorithm that combines distance and energy. First, nodes compare available paths, select a path and advertise the highest energy value. The energy value advertised is equal to the energy level of the node along the path with the lowest amount of energy. Second, a node's rank is calculated by adding three different values: the parent's rank, a fixed value representing the minimum rank increase, and the value calculated by subtracting the value of a node´s own energy level from a fixed reference energy level. Since the rank of low-energy nodes increases more quickly, they are less likely to be selected as parents. However, the paper does not provide a thorough discussion of the impact that rank has on path selection. A well-defined energy-aware metric is used in E-TORA [72], which is an improvement of the Temporally Ordered Routing Algorithm (TORA) [73] protocol. TORA creates a routing Directed Acyclic Graph (DAG) in which the nodes can cache several back-up paths toward the sink. E-TORA goes beyond this, taking the hop count and the energy level of the nodes into consideration when selecting path. Specifically, the energy level used in the cost calculations of E-TORA is the energy level of the lowest energy node along a path. However, in WSN, the lowest energy nodes are those in the vicinity of the sink, because it is these nodes that must forward data generated by all other nodes in addition to transmitting their own generated data. The process of selecting a path is therefore unaffected by energy

imbalance between nodes located further from the sink. Thus, E-TORA is not very useful for WSN. Other approaches used to reduce energy consumption at the network layer involve combining local information and energy [74, 75], and generally minimizing network overhead.

The residual-energy of the nodes and the path characteristics metrics focus on balancing the energy consumption between nodes rather than reducing the total energy consumption. The focus of the present study is on RPL run networks in which the importance of balancing is demonstrated by the real-world experiments presented in [29]. The experiment shows that some nodes become hot-spot nodes since they are used in a disproportionate number of paths. Thus, in paper D, the focus is on various metrics that may be used to balance the energy consumption between nodes, in order to prevent the generation of hot-spot nodes. Balancing energy consumption is further discussed in 2.2.3.

2.2.2.3 Topology control

Topology control is an important energy conserving solution in WSN [76-78]. Topology control protocols balance two a features: transmission range and network connectivity. The transmission range is tuned by setting the output power. High-output power means high energy consumption and a large interference range. The latter increases the collision probability. Collisions cause retransmission, which is a waste of energy. However, low transmission range weakens the network connectivity and increases the path lengths due to the increased number of hops.

Topology control can be achieved adjusting the transmission range, in order to provide satisfactory network connectivity. This approach is used in COMPOW [79], where the power control is optimized using routing protocol entries information. The idea of the algorithm is that the optimum transmission power level is the lowest possible level that maintains the routing table entries associated with maximum transmission power. That is, all the routing table entries that exist when maximum output power is used, must still exist when the optimum transmission power is used. The power level used by all nodes is merged to the smallest common level. In [80], the COMPOW approach is adjusted to suit networks in which node dispersal is non-homogeneous. The individual power level needed to reach each individual destination is cached, and this is the power used when communicating with that particular destination. Node degree, the number of one-hop neighbors, is used for transmission power adjustment in [81]. A slightly different approach is presented in [82, 83] where the transmission power is adapted to connect a neighbor in every direction. Another approach is to base the topology control more directly on graph models as done in [84]. An initial graph is constructed applying maximum output power, which is then improved by wisely reducing output power to prune unneeded edges. The weakness with these topology control algorithms is that the management cost due to message exchanged and computation is high. The management is high both during initialization, to decide the different output power levels needed; and during operation, to tune the radio to the correct level. In addition,

the solutions increase the amount of state information that needs to be cached. Moreover, the dynamic nature of the wireless media can lead to unstable connectivity when the transmission range is fine-tuned to reduce energy consumption.

As an important energy conservation algorithm, topology control is part of the investigation in papers A and E. However, specific topology control protocols are not suggested.

2.2.2.4 Cross layer approaches

OSI [85] is a layered model that is often used to describe data communication. The layered model has several advantages. The most important of these is that it allows technologies at each layer to evolve independently from those at the other layers. Furthermore, it fosters competition since products from different vendors can employ the different layers in the communication stack. However, merging the functionalities of multiple layers into a common functional module may improve the performance of WSNs [86-88]. Two of the important functions that can benefit if the layers are jointly optimized are energy preservation [89] and Quality of Service (QoS)[90].

The MAC layer characteristics are taken into account to suggest a new routing metrics for RPL in [91]. The unslotted version of CSMA/CA of 802.15.4 is used at the MAC layer. The metric suggested is a combination of two sub-metrics. The goal of the first sub-metric is to extend the end-to-end reliability. It does so by considering the probability that the packet will be discarded due to channel access failure; i.e. by considering whether the packet will fail to obtain a clear channel within the maximum number of backoffs. In addition, the first sub-metric considers the probability that packets will be discarded due to retransmission limit. Thus, the reliability of a link is defined as the probability that packets will not be discarded due to retransmission limit or channel access error. The end-to-end reliability is calculated by multiplying the reliability values along a path. The aim of the second sub-metric is to distribute the forwarded traffic in order to provide load balancing in the network. This sub-metric calculates the cost for each individual parent as the sum of the energy the individual parent uses to receive and transmit traffic. During forwarding, these metrics are combined so that the parent with the lowest cost among the parents, and with good enough end-to-end reliability, is used as the next-hop node. A weakness of the approach is that it adds overhead in terms of monitoring of channel and retransmissions, and monitoring of the parents´ traffic load.

Limiting the number of retransmissions reduces the energy consumption, and a cross layer approach which has this as its aim is suggested in [92]. It proposes to enhance the performance of networks running RPL by including the sender´s rank instead of the receiving node’s address in data packets. Receiving nodes with a rank lower than the transmitter compete to become the relaying node. The lowest ranking node wins and relays the data packet. The idea is to reduce the number of retransmissions since there are several potential relay-nodes; i.e the probability increases that the packet will be

correctly received by a neighbor as the number of receivers increases. Hence, the probability that the sender has to retransmit the packet will be reduced. This approach may reduce the number of retransmission, but there is always the chance that there will be duplicate packet transmission during such probabilistic forwarding. In addition, the nodes waste energy since all overheard packets must be received and prepared for transmission

The approach suggested in [93] aims to support the transmission of time-critical data while minimizing energy consumption. In order to do so, the cluster-tree topology of beacon mode IEEE802.15.4 cooperates with an extended version of RPL. RPL is extended to enable opportunistic forwarding over several cached parent nodes. Instead of specifying a specific parent node, the next-hop-node is chosen on a per-packet basis. The IEEE802.15.4 structure is adapted to allow nodes to associate with several parents using appropriately scheduled non-overlapping superframes. Thus, nodes can access the channel during different superframes; for instance, to minimize a one-hop time delay, the parent with the earliest upcoming superframe would be chosen. However, in order to both deliver the packet in time and simultaneously save energy, the nodes would choose the parent with the lowest cumulative ETX among the nodes offering a satisfactory deadline. The cumulative ETX is obtained by summing up the ETX along a path, so the cumulative ETX reflects the energy expenditure of a path. However, a weakness of this approach is that the energy consumption of the nodes increases when they must keep track of the various non-overlapping superframes. In addition, since ETX must be learned, communication between the nodes is needed in order to calculate and update the value of ETX. Moreover, the wireless environment is rarely static, so the link characteristic is prone to temporal changes and the ETX value needs to track these changes. Communication between nodes must be frequent enough to keep the value of ETX updated; management traffic must be exchanged in order to keep track of link dynamics between less frequently used potential members of the parent-list. Otherwise, there is no comparison between the ETX values of the different parents, or the comparison may be erroneous.

Although the approaches discussed above all concentrate on cross-layer communication between MAC and network layers, cooperation between any layers can be performed. The energy consumption model presented in paper E includes both MAC layer characteristics and hop count, where the latter is a characteristic of the network layer. Paper E focuses on transmission range versus energy consumption due to ETX.

2.2.3 Energy balancing in WSN

The aim of balancing energy consumption in WSNs is to avoid early depletion of nodes, which would interfere with network connectivity. The traffic load in multihop WSN is imbalanced since the nodes close to the sink relay traffic on behalf of nodes further out in the network. Hence, the average energy consumption inevitably increases as the distance to the sink decreases. The focus of the balancing algorithms is therefore

on balancing the consumption among the nodes located equidistance from the sink. The traffic load decides the energy consumption of the individual nodes.

The overarching goal of energy balancing is to lengthen the network lifetime. The network lifetime is the time during which the network is operational and capable of fulfilling its tasks. However, various definitions are used to define whether the network is fulfilling its tasks [94].

A conservative definition of network lifetime is the time before the first node depletes; i.e. runs out of energy or otherwise fails to continue operation. This may not be a very constructive definition in networks with a large number of nodes or high node density. In order to focus on the interdependency between the nodes, the time before the network partitions may be a better definition. Network partitioning definitely prevents data from a specific part of the network from reaching the sink.

So far, the definition has a node focus. However, from an application perspective, multiple sensors may cover the same area, and sensor data will be generated even though some of the nodes are cut off from the network or depleted. This may be the case if the coverage range is larger than the transmission range or if the number of nodes is higher than required.

Another definition of network lifetime is the time before an event fails to be reported. This definition is an improvement of the loss-of-coverage definition because it prevents the assumption that the network is dead just because some inactive part of the network is partitioned from the main network.

Independent of definition, the problem is that some nodes drain faster than other nodes and equalizing the drain between the nodes increases the network lifetime. The aim of the current study is therefore to balance the energy consumption between the nodes. To do so, the energy-balancing algorithms were divided into two groups: single-path algorithms and multipath algorithms. In single-path algorithms, the nodes use a fixed parent node in the interval during which no energy information is received from the nodes in the parent-list. In multipath approaches, the nodes continuously spread the traffic between parent nodes according to some defined requirement. Both single-path and multipath algorithms are investigated in paper D, and are therefore discussed in 2.2.3.1 and 2.2.3.2 respectively.

2.2.3.1 Single path algorithms

Several single path algorithms that aim to balance the energy between the nodes are suggested in the literature. The majority of these algorithms require that nodes exchange information about their residual-energy, which is then used to select the forwarding path. Paths are generally selected so that the traffic load on the most depleted nodes is reduced, and this, in turn, reduces the energy consumption and increases the nodes' lifetime. As a network layer approach, single-path energy balancing is also discussed in 2.2.2.2, where the topic is energy conservation.

Single-path algorithms that include energy information in routing protocol update messages are presented in [95-98]. The authors of [95] propose an energy aware objective function for RPL in which the node with the highest residual-energy among the nodes with the lowest ETX is chosen as the preferred parent. In [98], the factors included in the selection of a preferred parent among neighbors are distance to the sink, ETX and residual-energy. Distance to the sink is the primary parent selection criterion: it prevents the formation of routing loops. Second, ETX and residual node energy are used to select stable links and simultaneously balance energy consumption. A dedicated parameter value is used to weight the significance of ETX and residual-energy. However, ETX has a number of weaknesses, as discussed in 2.2.2.4. In [96] path selection criteria are modified by the definition of a new metric so that the cost of a node is calculated by dividing a node's distance to the sink by the node's residual-energy. Thus, low energy nodes are avoided since they are assigned a high cost. However, long paths with high residual-energy nodes are chosen over shorter paths, thereby increasing the energy consumed per packet.

A metric that estimates the Expected Lifetime (ELT) of the nodes is defined in [97]. ELT is the residual-energy of the node divided by the energy it consumes to transmit both its own generated data and the data it relays. Moreover, the expected number of retransmissions is included. The aim of the suggested metric is to minimize the energy consumption of the most constrained node, so path weight is set as the minimum ELT along the path. The path with the largest ELT is selected. Hence, paths that pass through low-energy nodes are not likely to be selected. However, management of ELT gives rise to high computational overhead and ELT must dynamically follow link characteristic changes and traffic changes.

Certain general weaknesses are common for some of the solutions discussed above. The whole path requires signaling, and therefore the overhead increases for algorithms that consider entire paths. This is because each change of preferred parent has to be advertised to all downstream nodes, and may cause all nodes to change preferred path. In addition, limiting the exchange of energy information to routing protocol update messages may prevent the detection of significant changes in parents' energy levels; the frequency for such updates messages exchange is decided by network stability, and not the depletion rate of the nodes. The papers cited in this sub-section report energy balance improvements. However, the comparison is not always complete. For example, many publications do not take into account the influence that the DIO emission frequency has on energy balancing capacity.

In the algorithms discussed above, the nodes use the routing update packets to inform others about their residual-energy level. In contrast, data packets, ACK packets and routing protocol packets all carry energy information in the algorithm suggested in [99]. Energy information is used as a tiebreaker for parent selection, while hop count is the basic metric used to build the parent-lists. Hence, the parent-lists consist of nodes advertising a hop count equal to the shortest hop count received from any neighbor. The preferred parent is the node with the highest amount of residual-energy among the

nodes in the parent-list and data packets are transmitted to the preferred parent. However, if the parent-list is empty, the packet is transmitted to the sibling node with the highest amount of energy; this makes the forwarding more resilient. However, it may create forwarding loops; for instance, sibling nodes may choose to forward through each other. Nevertheless, forwarding through the highest energy node is advantageous if the aim is to balance energy consumption. However, the residual-energy of the parent may not reflect the residual-energy of the nodes further down the path, and the energy balance for nodes further down the path may suffer.

Only ACK packets exchange energy information in the solution suggested in [19], in which the characteristic of the data transmitted decides how to choose among the nodes in the parent-list. Essentially, time-critical data are sent to the parent that offers a packet transmission rate that is high enough for the data to reach the destination in time. Data that is not time critical are sent to the parent with the highest amount of residual-energy, in order to improve the energy balance in the network. A dynamic weighting coefficient is used to weight rate and energy; this means that the number of potential parents increases when the rate requirement is relaxed. The rate is calculated as the number of hops divided by the delay. To measure a one-hop time delay, the waiting time in the queue at a node is added to the transmission delay, which is calculated, in turn, using the transmission time of a packet and the arrival time of the associated ACK. The nodes inform others about their own transmission delay through the ACK packets they transmit. The suggested algorithm was evaluated through simulations, which show that the number of packets transmitted before the first node dies increases by a factor of 3 when the weighting parameter changes from only considering delay to only considering energy. The factor decreases when the delay requirement for the transmitted traffic is stricter. The simulations results prove that, as expected, the balancing efficiency increases with the number of potential parents. A weakness of this algorithm is that energy information gained through ACK is only reactively updated, when a packet is first transmitted. In addition, ACK is sent in reply to a received packet, so no updates are received from the parents that are not used. Moreover, estimation of path delays add to management overhead.

2.2.3.2 Multipath algorithms

In this section, some of the relevant multipath approaches found in the literature are presented. The published algorithms often weight the traffic of the various paths. Weighting based on residual-energy in the path ensures that the traffic load is reduced when the nodes' residual-energy is reduced. Hence, the depletion rate of the lowest residual-energy node is reduced and its lifetime is thereby increased. However, the relationship between the individual nodes' residual-energy remains unchanged.

An alternative to weighting is to divide the transmitted data equally between all potential parent nodes. The advantage of this approach is that the forwarding load may be reduced for the parents of child nodes that transmit a large amount of data, and for those that are parents for several child nodes. The load is reduced if the child nodes

have several alternative parent nodes. Another advantage is that there is no need for state information along the path. The weakness is that all nodes forward equal amounts of data to all alternative parent nodes, although the parent nodes may have different amounts of residual-energy. In addition, there is still a forwarding load imbalance between nodes with unequal numbers of child nodes.

Moghadam et.al in [14] suggest an approach in which the nodes calculate their own energy consumption rate based on their traffic rate and associated receiving and transmission energy cost. Neighbors exchange information regarding their energy consumption rate through hello or ACK packets. Nodes balance the forwarded traffic load between all members of their parent-list in order to smooth the energy consumption between nodes. Furthermore, the transmission range is adjusted to achieve optimal network connectivity. This solution was simulated with different traffic loads and different network connectivity values. A k-connected graph was defined as a graph in which the nodes had at least k parents. Simulations were performed varying the k from one to four. It was found that the 2-connected network, in combination with the suggested energy balancing algorithm, gives the longest network lifetime. While increasing the connectivity above the 2-connected network increases the number of available parents to share the traffic, the increased transmission range means that the power per transmission is increased and there is more interference. Thus, the lifetime and throughput is reduced. The network lifetime is here defined as the time until the first node depletes. The weakness of the algorithm is that it requires a vast amount of state information and a vast amount of calculation. In addition, the transmission range adaptation may considerably increase the overhead.

Another weighting-based protocol is presented in [100], in which the data are spread over several energy-sufficient paths. To be considered energy sufficient, the path must consist of nodes with an energy level higher than a given threshold. The primary metric used to collect potential parent nodes is hop count. The secondary metric used is energy level and node-to-parent signal strength. Hence, the parent-lists consist of nodes advertising a hop-count value equal to the minimum value heard, and with an energy level and a node-to-parent signal strength higher than the specified threshold. The amount of traffic to be transmitted through the various parents, i.e. the traffic rate of the different paths, is assigned by the sink. The condition of the paths is continuously monitored by the sink, which allows for the redistribution of data rates when required. In addition, the sink re-initiates the path search if the number of working paths between the source and the sink becomes less than two. A similar algorithm is suggested in [101], but it is required that the multiple forwarding paths are disjointed. While the algorithms suggested in [100] and [101] improve the energy balance in the network, they also add a centralized management, which makes the network less scalable. For instance, in the suggested algorithms, the traffic overhead is high and increases with the number of nodes since the sink needs to gather the required information in order to set the data rates for the different paths. In addition, the sink transmits information to inform the sources about the data rates of the different paths. Another disadvantage is

delay; management of the paths is delayed since the sink must detect the changing network conditions and inform the source nodes, before corrective action can be initiated.

Another approach, slightly different from the two above, is to divide the traffic between different routing trees, as presented in [102]. The basic metric is hop count and the sink initiates several sessions to create several trees. In each session, the nodes select a new parent that is the node that has been used as a parent the least number of times. Specifically, a node counts the number of trees in which each parent-list member has acted as a preferred parent, and the parent with the lowest count is the next to be selected. If several parents have counts equal to the lowest, Link Quality Indication (LQI) is used as a tiebreaker. LQI is a measure of the strength or quality of a received packet [63], based on the received signal power and the signal-to-noise ratio. The trees are differentiated as the sink assigns each one a unique identification. During data transmission, a node sends data that originates from itself through its preferred tree, while forwarded data is sent through the originating node's preferred tree. Thus, load balance is created as originating and forwarded data can be sent through different trees. A weakness of this approach is that energy consumption and computational overhead increases in order to create and manage several trees.

In paper D solutions are investigated and suggested that relate to balancing the energy consumption in networks running RPL. Both single-path and multipath energy balancing approaches are investigated. Without compromising the RPL standard, nodes running RPL can cache several nodes in the parent-list. Specifically, an associated Objective Function (OF) decides the exact parent selection process, and not the RPL standard. In addition, various metrics can be applied, and the OF defines how to perform calculations related to the selected metric. Thus, several of the approaches discussed in this section can be used in networks running RPL.

2.3 Network connectivity

This section presents a short literature review of network connectivity. The network connectivity in WSNs is a trade-off between node density versus node transmission range, on one hand, and interference on the other hand. Both increased transmission range and increased node density increase the probability that nodes are within the transmission range of each other, which means that the probability of network connectivity increases. On the other hand, interference increases with node density and transmission range, thereby reducing the probability that the transmitted data will be received correctly. Without correct reception, the network appears disconnected since the nodes are prevented from communicating. However, the paper´s focus is on WSN with a low traffic rate, so interference is unlikely. Given this assumption, increased node density or transmission range improves the network connectivity.

The routing protocol imposes a structure among the nodes, and this interrelationship depends on the specific routing protocol. Figure 3 illustrates such an interrelationship among WSN nodes that use hop-count as the routing metric. The network to the left shows the initial node arrangement. The node named S is the sink. The black arrows point to the parent nodes; i.e. the arrows define the paths in the network. The blue dotted line shows that nodes d and g are within reach of node e. The black numbers beside the nodes represent each nodes' rank. In this scenario, node b depletes, leaving node e disconnected. If node e is free to reconnect thorough a new path to reach the sink S, it will reconnect thorough node d or g, as shown in the network on the far right. However, in network running a routing protocol such as RPL, the nodes are not allowed to move outward in the routing graph; moving outward is defined as reconnecting to a node further from the sink. Hence, the disconnected nodes running RPL cannot reconnect through a node further out in the routing graph when a disconnection occurs. In particular, node e cannot reconnect through either node d or node g. This is because a node that moves outward creates a routing loop if it is unintentionally reconnected to its own subDAG; i.e. the moving node is connect to a node whose path toward the sink includes the node making the movement. For instance, if node g was initially using node e as parent in Figure 3, then a loop would be created if the disconnected node e reconnected to node g or one of node g's child nodes.

In order to avoid routing loops during path recovery, RPL requires that a node with an equal or shorter path toward the root replace the depleted node. Such a requirement prevents routing loops but reduces the potential set of possible alternative paths. The end-result is a lower probability that broken paths can be recovered without additional mechanisms. Paper C evaluates how often additional mechanisms, i.e. path recovery algorithms, are needed to regain connectivity.

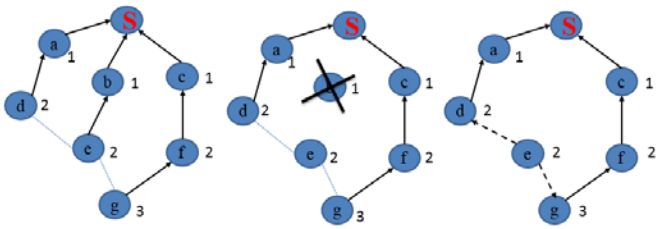

Figure 3 The connectivity decided by a routing graph is more vulnerable than if nodes can be connected through random paths.

In their seminal paper [103], Kleinrock and Silvester discuss network connectivity. Based on cited works [104, 105], they found the average degree, i.e. the number of nodes inside a transmission radius, necessary for the network to be connected with a probability of 0.95. The cited works present two types of graphs: large random graphs, which are graphs that are not defined by geometrical relationships; and Euclidean graphs, in which the existence of edges is not an independent process. The node degree

required to achieve the 0.95 probability of network connectivity increases with the total number of nodes in the network, and is higher for the Euclidean graphs than for the random graphs. Based on the cited works, Kleinrock and Silvester conclude that the degree must be at least four to have a connected network.

An expression that calculates the node density and the transmission range needed to achieve a fully connected network is presented in [106]. The node distribution is assumed to be random and uniform, and the expected number of nodes per unit area is ρ. Since the distribution is uniform, the number of nodes in an area obeys the Poisson distribution and Poisson can calculate the probability that a given number of nodes are located inside an area. Thus, Poisson is used to find the expression for the probability that each node n in the network has at least k neighbors. Referring to the findings presented by Penrose [107], it is assumed that, if the number of nodes is high enough, then it is highly probable that a graph becomes k-connected at the moment each node achieves a minimum node degree k. Thus, [107] is used to argue that the graph is k-connected when the each node has at least k neighbors, and Poisson is used to calculate the probability that all nodes has at least k neighbors. The result found in [106] show that the probability of the nodes being k-connected increases rapidly from 0 to 1 as the transmission range increases, and a node degree of 10.8 is needed to achieve a network that is almost certain connected. In the calculations, the nodes have equal transmission range, r_0, and the links are bidirectional. In [108], corresponding results are found: the connectivity rate increases rapidly with the number of nodes. The model presented in [108] calculates the lower boundary of the network connectivity probability when the network is randomly deployed. As in [106], the calculations performed in [108] are based on the correspondence between node degree and connectivity, presented by Penrose in [107]. The node degree is calculated by considering a square-shaped network, and calculating the probability that the nodes' transmissions cover more than k neighbors. A weakness of these calculations is that there may be nodes that are k-connected, but disconnected from the main part of the network.

In [109], the network is divided into grids and connection probability for the entire network is calculated by progressively combining adjacent grids based on geometrical probability. It was found that the connectivity rate increases rapidly with the number of nodes, which corresponds to the findings of [106] and [108]. A weakness of the method used in [109] is that nodes located in the proximity of the network border have limited coverage area and the probability for isolation is higher. The negative effect of the border is reduced, but not removed, when the network is divided into smaller grids.

Connectivity results depend on node distribution. In [110], simulations of WSNs are presented in which the nodes are deployed following a two-dimensional Gaussian distribution around the sink, instead of the most commonly used random uniform node deployment. During simulation, connection rate was used as a parameter to define the proportion of nodes connected to the sink. A simulation scenario consisting of 500 nodes with a transmission range of 50m was used to investigate network connection. It was observed that the sensor connection rate increases quickly as the transmission

range increases. In addition, a connection rate of 100% is unlikely for random WSN applications when the nodes are distributed according to a two-dimensional Gaussian, with a mean equal to the location of the sink and a standard node deployment deviation of 20m. When the deviation is less than 20m, there is a high probability that the network will be connected, because most of the sensors are deployed close to the sink. However, a reduced connection rate of 90% is likely for deployment deviation as high as 140m. A weakness is that the node arrangement that gives the best connectivity may not fulfil the sensing requirement of the WSN, and this is not discussed in the paper.

Paper C in this study investigates network connectivity. As in the connectivity evaluations discussed above [103, 106, 108, 110], the focus is on randomly deployed networks. However, in contrast to these studies, the specific focus is on investigating connectivity related to routing paths created by the RPL, particularly the impact of depleted nodes. A depleted node may leave a child node in the poisoned state, which means that the former child has no other neighbor at the same or lower rank than the depleted parent node. In other words, the parent-list of the child node is empty. A node in the poisoned state is illustrated in Figure 3. When node b depletes, node e cannot reconnect since none of its neighbors is at an equal or lower rank than node b. Node e is therefore in the poisoned state since it cannot reach nodes a or c. Nodes in this state need a recovery algorithms to reconnect them to the routing graph. Thus, Paper C investigated the probability that disconnected nodes will enter the poisoned state. The findings confirmed those in papers [108, 109] since the depleted nodes impact on the routing paths diminish rapidly with increased node density: The probability that a disconnected node will be able to find a new node at equal or better rank than the former preferred parent increases with the number of neighboring nodes.

2.4 Path recovery

The overall objective of WSNs is to transmit data from sources to the sink. The applied routing protocol decides the path used for transmission. If paths are broken, the WSN fails to achieve its objectives and the data gathered at the sink is incomplete.

This study focused on the RPL routing protocol, which is a soft state protocol. Soft state means that, in order to keep the network operational, the states must be refreshed before they are timed out. The specific routing graph is organized by states cached in each individual node. Path break recovery occurs when these states are refreshed, which happen when the sink initiates a network-wide broadcast of DIO messages containing an updated DODAGVersionNumber. When choosing the time intervals between such update messages, there is a trade-off between the need to keep states updated and energy consumption. Increasing the interval saves energy since the frequency of network-wide broadcasts is reduced. However, the states in the nodes may become obsolete, and damaging conditions such as long-lasting path breaks may occur. Short intervals prevent long-lasting damaging conditions, but increases the management energy consumption due to frequent network-wide broadcasts. However,

local path recovery algorithms remedy path breaks whenever they occur and can therefore both prevent long-lasting damaging conditions, and reduce the energy consumption due to frequent network-wide broadcasts.

The algorithms used for path recovery in RPL running networks can be divided into two different groups: global and local recovery. In global recovery, disconnected paths are recovered when the routing graph of the total network is updated. In local recovery, the node that is entering the poisoned state initiates a recovery process. During this process, the poisoned node exchanges information with nodes in its vicinity. The information exchanged carries the elements essential to identify alternative loop-free paths. The aim of the recovery process should be to discover all available paths, in order to guarantee that the path is recovered whenever possible.

According to W. Xie in [111], the use of loop avoiding techniques in DAG-based routing protocols are not recommended. They claim that the impact of increased overhead introduced by running the loop avoiding technique is more severe than the overhead from the actual routing loops. The overhead is related to the large number of DIO messages that are generated because the loop avoidance operation forces a large number of nodes to change their rank and parent set. In addition, the stabilization time is not always reduced when loop avoidance is used. However, the main weakness of this approach is that the loop avoidance algorithm is not compared with global recovery to ensure loop-free operation. Instead, it is compared with an algorithm in which a node in the poisoned state increases its rank if needed in order to reattach to the DAG. The only requirement in the algorithm used for comparison is that nodes that increase their rank transmit DIO carrying information about their new rank. This information may force subDAG nodes to perform the same operation, since the subDAG of a node n includes all nodes that use node n on their path toward the destination. The weakness of the algorithm used for comparison is that transient loops may occur as nodes use each other as parents before the trickle timer expires. Moreover, persistent loops occur if the DIO with poison information is lost. Thus, the recommendation made by Xie might have been different if the loop avoidance algorithm had been compared with algorithms with guaranteed loop-free path recovery.

In papers B and C various recovery algorithms suitable for RPL networks are suggested and evaluated. The next sections discuss recovery algorithms. 2.4.1 considers global approaches and 2.4.2 considers local approaches.

2.4.1 Global recovery

Global recovery algorithms rely on global graph updates to reinstate disconnected paths. As explained in 2.4 , the sink initiates global routing graph updates by updating the DODAGVersionNumber in RPL networks. Nodes must attach themselves to the most recent DODAG and must always belong to the same DODAG version as their parents. Thus, the entire routing graph is updated as DIOs with updated DODAGVersionNumbers spread throughout the whole network like a wave emanating

from the sink. The algorithm generally discovers all available paths, but DIO messages may be lost. The algorithm guarantees that there will be no routing loops.

In general, proactive routing protocols perform a global recovery periodically. However, to reduce restoration delay, proactive protocols may include recovery algorithms that are triggered by path breaks.

In paper C a solution is suggested for the trade-off between energy and delay that occurs when the frequency of routing graph updates is varied. The solution is to initiate the global recovery procedure on-demand.

2.4.2 Local recovery

The main advantages of the local recovery approaches are the limited energy expenditure and low convergence time. To limit the energy expenditure, the number of nodes participating in the recovery is as small as possible. The disadvantage of this approach is that the recovered path may not be the optimal. In addition, a more serious problem is that alternative paths passing thorough former subDAG nodes may not be discovered.

Various methods can be used to perform local recovery of routing paths in WSN. Among these are sequence numbering, source routing, movable nodes and caching of multiple paths. Caching of multiple paths can function as a proactive approach, rather than a recovery approach to reduce the probability of path disconnections. For RPL, multiple paths involve the caching of several parents during routing graph updates. The weakness of this approach is that optimal paths may be sacrificed in order to gather enough back-up paths. Movable nodes can be used as proactive approach or as a local recovery approach. Used proactively, a node is moved such that it is located where the next disconnection is most likely to occur [112]. Used as a local recovery method, movable nodes can be used to restore connectivity [113]. In addition, movable data collectors and base stations can be employed to facilitate network connectivity [114]. When using the latter, mobile collectors visit the various part of the network to gather data. Subsequently, the mobile collector delivers the data to the sink node. However, moving nodes increase management overhead and node cost. Alternatively, man-hours are required to move the nodes.

Sequence numbering and source routing are frequently used local recovery methods. Generally, the disconnected node emits a request messages in search of alternative paths. Nodes able to offer an alternative path reply to such requests. In sequence-numbering methods, the number ensures the freshness of the path offered in the reply. That is, the sequence number ensures that the broken path is not part of the new path offered. When source routing is applied, the reply packets include the address of every node it has traversed. To prevent routing loops, nodes discard all replies in which they find their own address in the address list. The following discussion considers the sequence number local recovery algorithms used in Ad-hoc On-Demand distance Vector (AODV) [115, 116], and the source routing algorithm used in the Dynamic

Source Routing protocol (DSR) [117]. AODV and DSR were originally suggested for ad-hoc networks, but the ideas these protocols use are well suited to WSN networks.

One of the local recovery algorithms suggested in paper B was inspired by AODV [115], which is a protocol in which destination sequence numbers are used to avoid routing loops and ensure route freshness. Each node caches and manages its own unique sequence number. The sequence number that belonging to a destination is cached in routing table entries that point toward that destination. In other words, nodes cache the sequence number of each known destination. Source nodes emit a Route Request message (RREQ) to discover possible routes to destinations. If the destination is previously known, the RREQ includes the sequence number cached for the destination. Expanding ring searches can be used to avoid network-wide flooding of RREQ. The destination node, or a node that has an active route to the destination with a more recent sequence number, responds to the request. Route Replies, RREP, are unicasted toward the source node through the path created by the associated RREQ. Forward paths are set up when replies travel along the path toward the source node. A great advantage of this algorithm is that, as long as there is a path toward the destination, it is likely to be found. Weakness to this approach are that the energy expenditure for broadcasting of RREQ is high, and its reactive nature introduces delay. Nevertheless, the AODV is interesting, and one of the local recovery algorithms presented in paper B was inspired by AODV.

In [118], the AODV process has been improved by assigning higher sequence number values closer to the destination. The nodes assign themselves a sequence number within a window in which their predecessors´ sequence number define the lower limit, and their successors´ sequence numbers define the upper limit. Since the sequence number values increase toward the destination, nodes caching higher sequence numbers for the destination than the requesting node are able to respond to requests. A weakness of this approach is that information regarding successors´ or predecessors' sequence number can be lost due to packet loss. Such loss can lead to routing loops.

Source routing can be used as a local recovery approach in networks running DSR [117]. Path disconnections, which should be reported in route error messages, cause the source nodes to initiate route request broadcasts. A route request is re-broadcasted until it reaches the intended receiver. Each node appends its own address to the route request packet before it is re-broadcasted. Thus, the reply from the receiver contains the accumulated route from the route request packet. This avoids looping since duplicated addresses are not allowed in the accumulated information received. The accumulated path is included in the data packet sent to define its path toward the destination. An important weakness of this approach is the high overhead created due to the route information included in the packets.

A local recovery algorithm that repairs path breaks discovered during data transmission is suggested in [119]. The suggested algorithms were designed for WSN, so the routing graph is rooted at the sink. Each node forwarding a data packet must get a confirmation

that the successor node receives the packet. A node that detects a path break assumes that the successor is dead and initiates path recovery by emitting a path request. The request is further unicasted by nodes that have alternative paths toward the destination. A response is sent if the request reaches a node with a rank that is better or equal to that of the dead node. If no alternative path is found, the data packet is marked and sent one hop back toward its source. This process is repeated until the data packet reaches its destination or is back at its source. The packet is discarded it if it is returned to it source and no alternative path is found. This algorithm seems interesting, but the weakness is that paths are only recovered if nodes no further away than one-hop from the initial path can provide alternative paths, either directly or through their successors. Specifically, a path going through a node located two hops outward from the initial path will not be identified.

Sibling nodes are used to create redundant routing paths for data packets in [120]. To avoid looping of packets, the sibling is used only once per rank-level in the DAG. Hence, the next-hop node after the sibling must be a node with a lower rank. This requirement is met by means of a flag in the packet header. The suggested solution increases the packet throughput since some of the packets that would otherwise be discarded is able to reach the destination when transmitted through the sibling. However, not all path breaks can be recovered using siblings. For instance, the sibling selected may connect to a common depleted node. In addition, only one sibling per rank-level is allowed, while an alternative path may exist at two-hop distance.

The local recovery algorithm described in RPL, poisoning of subDAG, is no guarantee against routing loops. Poisoning means that nodes entering the poisoned state emit DIO messages advertising a rank of infinity. The poisoned state is described in 2.3 . Receivers of the infinity-rank messages, i.e. nodes that are part of the poisoned node's subDAG, remove the poisoned node from the parent-list. Subsequently, the poisoned node may reattach to the DODAG as it is no longer part of the path toward the sink for its former subDAG nodes. Thus, routing loops seems to be avoided. However, the infinite rank messages may be lost causing the poisoned node to reattach to its own subDAG and create routing loops.

In [121], an algorithm that aims to prevent the routing loops that are created during poisoning is suggested. The suggested algorithm uses DODAG Information Solicitation (DIS) messages to improve the poisoning algorithm. After advertising a rank of infinity, the node in the poisoned state emits a DIS message. The DIS message triggers receiving nodes to transmit DIO messages. It is assumed that the nodes that respond to the DIS are aware of the sender's poisoned state, so that the poisoned node can safely select a new parent among the responding nodes. However, nodes further down the subDAG may cause loops if they happen to receive the DIS message. In addition, recovered paths that pass through a former child node are not discovered. This is because a child node, which itself entered the poisoned state due to the poisoned parent, will not reply to the received DIS. However, these child nodes may

reconnect through an alternative path after completing their own recovery procedure, and this recovered path could have reconnected the initially poisoned node.

Gou [122] has suggested an interesting local recovery approach for networks running RPL. The main idea is to define a rank calculation method that enables the poisoned node to reattach to the DODAG through a longer path without increasing its rank. In order to do so, the rank of the nodes on the recovery path is adjusted. The adjustment is performed without manipulating the logical distance between the nodes lying on the recovered path and the sink. This recovered path is guaranteed loop-free. Since the solution suggested by Guo is discussed both in papers B and C, a thorough description of the solution is provided below.

In order to achieve the required flexibility of the rank, it is represented as a fraction of integers: rank = m / n. The use of a fraction means that it is always possible to place a new rank between any two former consecutive ranks. Therefore, nodes can always decrease their rank while still maintaining a rank higher than their parents' rank. This is used to create eligible restoration paths for the disconnected nodes. The disconnected node, which enters the poisoned state, broadcasts a request message carrying its rank. Receivers with higher or equal rank relay the request to their preferred parents. If the requesting node is the preferred parent of a receiver, the request is dropped. Nodes with lower rank than the requesting node respond the request. The reply is relayed along the same path as an associated request, but in the opposite direction. The nodes on the reply path change their rank to make the path available to the requesting node.

During initiation of the DODAG, the nodes set their rank as (m + p) / (n + q), where (m/n) is the rank of their preferred parent and (p/q) = (1/1). The sink has the rank of (0 / 1). Hence, the rank of the nodes is (0/1) – (1/2) – (2/3) and so forth outward from the sink. During path recovery, the same formula is used for calculation. However, for nodes on the recovered path, (p/q) is set equal to the rank of the requesting node while (m / n) still equals the rank of the preferred parent. If (m/n) < (p/q) then (m/n) < (m + p) / (n + q) < (p/q), so the recalculated ranks of the nodes on the recovered path are between the preferred parent and the requesting node. Thus, the requesting node can reattach through the recovered path without changing its rank.

The approach suggested by Guo has some weaknesses. The rank of the reconnected node is unchanged, however its distance to the sink may increase during the reconnection process. Thus, since rank is used to compare path lengths, the path length comparison performed by neighboring nodes is compromised.

In addition, a deadlock may occur if neighbors enter the poisoned state simultaneously. This is because nodes with pending requests are not forwarding received requests. For instance, two sibling nodes enter the poisoned state simultaneously. Both broadcast a request and, subsequently, discard the request they receive from their sibling since they have their own pending request. The siblings, therefore, cannot find alternative paths going thorough each other. Thus, if one of the poisoned nodes is dependent on the path through its sibling to reconnect, it will remain unconnected.

Moreover, nodes discard requests originating from their parents. Hence, the poisoned node is not able to find the alternative path if the path goes through its child node. This observation motivated this study on recovery algorithms since such detour paths may be able to facilitate reconnection.

Thus, in paper B two different local recovery approaches have been investigated and suggested. One approach is based on sequence numbering, an approach inspired by AODV, while the other improves the poisoning algorithm described in the RFC of RPL and discussed above. The aim of the suggested approaches is to guarantee loop-free recovery, as well as to discover all alternative paths. The Guo approach is also included in the discussion in paper B and C since it identifies alternative paths that are guaranteed loop-free.

Chapter 3 - Contribution

This section summarizes and discusses the contributions. These can be grouped into three areas and short overview of each is presented, followed by a more detailed presentation of each paper.

The first area is energy usage in WSN. Existing energy consumption models underestimated the impact of overhearing on resource usage, therefore our research began with the development of an energy consumption model for WSN. The model is presented in paper A and calculates the energy consumed during the forwarding of packets from source to sink. The model includes the energy consumption for different radio states. The model shows that the energy-optimal transmission range is strongly affected by the overhearing nodes. Previous work has not taken into account the relatively substantial energy usage in overhearing transmissions. Based on the results, a simple approach where overhearing nodes are switched to sleep mode is suggested to reduce the consumption. Paper E elaborates on the energy-optimal transmission range using a more realistic calculation of the relationship between transmission range and PDR.

The second area is the balancing of energy consumption. Related work lacked a thorough comparison of the various balancing approaches since these are not compared using a common network scenario. Thus, a variety of energy balancing algorithms are analyzed in paper D using a common network scenario. Six of the analyzed algorithms are similar to approaches found in the literature, while three of the algorithms are new algorithms that are suggested. One of the suggested algorithms has a significant balancing effect although it requires only a minor change in the parent selection algorithm. However, transmitting through the highest residual-energy parent node gives the best balancing effect.

The third area is network connectivity, in particular path recovery. Recovery solutions are required to lengthen the lifetime of the networks. The energy consumption of the nodes is never perfectly balanced; thus nodes deplete and paths get disconnected at different point in time.

The solution for path recovery presented by Guo represents the state of the art for local recovery in networks running RPL. However, the solution cannot guarantee that existing paths are found. Hence, the focus of the presented solutions is on discovering all existing paths. Thus, two algorithms are suggested in paper B that can be used to

repair broken paths. One of the algorithm is an expansion of the poisoning algorithm suggested in RPL. The other is inspired by the sequence number algorithm used in AODV. The ability of these algorithms to reconnect disconnected nodes is assessed. The algorithm suggested by Guo [122] (which will be referred to as the Guo or Guo algorithm) and discussed in 2.4.2, is included in the assessment.

The findings of paper B show that reliable local recovery creates high message overhead. Local recovery should therefore be avoided if not needed. Hence, the investigation of whether local recovery processes are necessary in networks running RPL began. Existing works do not investigate the connectivity related to recovery rules that exist for routing paths in RPL. Thus, in paper C, an analytical estimate is made of the expected need for local recovery in randomly deployed networks running RPL. In addition, an on-demand global recovery algorithm is suggested that establishes a reliable recovery process.

3.1 Paper A

It is important to understand how the various operational states use energy, and thereby determine the lifetime of the nodes and the network. The decision was made to focus on the radio since it is the main energy consuming part of the nodes [11]. The transmission energy consumption of a node has a fixed part and a part proportional to the transmission range, while the energy consumed to receive is constant as presented in 2.2.1. Hence, the energy consumed increases with the number of nodes covered by a transmission.

Overhearing nodes consume energy to receive a packet, which is then discarded. A literature review showed that the analytical model did not sufficiently accounted for the energy consumed for overhearing. As a starting point, the datasheet for popular chipset was used to calculate the energy consumption for various operations. The nodes consume energy in the same order of magnitude to receive/overhear and to transmit a packet. Motivated by these two observations, an analytical model was developed to facilitate the investigation of how overhearing affects the energy usage during forwarding from source to sink.

The contributions of paper A are threefold. First, a new energy consumption model is presented. Second, this model is used to calculate the energy-optimal transmission range. Third, a simple approach is suggested that will reduce energy consumption for transmission from source to sink.

The suggested model defines the total energy consumption for forwarding from source to destination as the sum of the transmission and receiving at each hop along the forwarding path. See equation (3) paper A.

In order to find a realistic energy-optimal transmission range, parameter values were extracted from datasheets for the energy consumption model. Calculations show that the energy-optimal transmission range is very short, see Figure 2 in paper A. The conclusion is that the optimal range is so short that the network is likely to be

disconnected if it is used. Therefore, it is better to use a transmission range just long enough to avoid network partitioning. In other words, the transmission range is set at a distance that will keep the network connected. Simulation results reported in [15] support the calculated results. The simulation scenario used in [15] consists of nodes with varying transmission ranges in a network where nodes are deployed in a fixed mask pattern. However, if no overhearing occurs, it is optimal to set the longest transmission range possible.

Findings for the energy-optimal transmission range contradict those of [16], where it was found to be most energy-efficient to use the maximum transmission range during forwarding. [16] focuses on the drain efficiency of power amplifiers, which is highest for the highest output power. The main reason for the different results is that the energy consumed by all overhearing nodes is included in paper A. [16] and other related studies [17] on the energy-optimal transmission range do not include the energy consumed for overhearing.

One weakness of the model presented in paper A is that the overhearing nodes are assumed to receive the whole data packet transmitted. Hence, it is not applicable for networks protocols in which the overhearing receivers enter the sleep state after deciding that the transmission is addressed to another node. However, the proposed model can easily be adjusted to suit such protocols by making the energy consumption of the overhearing node a fraction of the addressed node's energy consumption. In addition, the model can be adapted for TDMA protocols by reducing the number of receivers to one.

The results show that overhearing consumes a substantial amount of energy. A simple solution suggested is therefore to signal ahead of transmission the node that should be awake to receive the packet. Each data packet transmission begins with a signaling packet ordering overhearing nodes to enter the sleep state. Using simulation scenarios consisting of randomly deployed nodes shows that, compared to classic CSMA, the suggested approach reduces the energy consumption. The energy difference increases with node density and data packet size. This simple solution can be used in combination with the MAC protocols presented in 2.2.2.1.

Clearly, the suggested solution to reduce energy consumption has limitations. The signaling packet that precedes data transmission introduces delay. In addition, transmission from an exposed node is delayed, as illustrated in Figure 4: Node B transmits a packet to node A. Meanwhile, node C has data to transmit toward node D. However, node C enters the sleep state due to node B's transmission, although C's transmission would not have prevented node A from correctly receiving data from node B. Hence, node C is an exposed node whose data transmission is delayed. However, exposed nodes are a common problem for MAC protocols and delay is often the price to pay for reduced energy consumption. The application decides whether increased delay can be tolerated in order to extend the network lifetime.

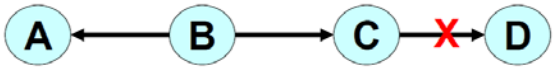

Figure 4 Exposed node

3.2 Paper B

As already discussed, the amount of generated, forwarded and overheard data affects the energy consumption, and thus the lifetime of the individual nodes. Since there are variations in network topology, size of packet generated and frequency for transmission, the lifetime of the nodes and thereby the paths will vary. Hence, algorithms to restore broken paths are needed to maintain network connectivity. As discussed in 2.4.2, the local recovery approaches suggested for RPL has certain weaknesses. They cannot guarantee loop-free paths and cannot guarantee finding a path when a path exists. This motivated the analysis in paper B of a broader class of recovery algorithms that would find a loop-free recovery path if a path existed.

The contribution of paper B is twofold. First, two local recovery algorithms are suggested that are likely to restore a disconnected path whenever an alternative path exists. The algorithms are designed for the RPL routing protocol. Second, an analysis of these algorithms and the Guo approach is undertaken.

The first suggested algorithm is inspired by the AODV [115] algorithm discussed in 2.4.2. A local sequence number is introduced, in addition to the global sequence number. The global sequence number is the DODAGVersionNumber used in classic RPL. The local sequence number is reset to zero each time the DODAG is refreshed, i.e. when the global sequence number is increased. A node that enters the poisoned state increases its own local sequence number before broadcasting a path-recovery request. Nodes with a higher local or higher global sequence number respond to the request. The replying node's own sequence number is included in the reply. Nodes along the reply path update their local sequence number according to the number in the reply. Only the sink is able to increase the local sequence number received in a request in order to generate a reply. Receiving a reply, the disconnected node can reconnect to the DAG through a path that is guaranteed loop-free. The algorithm is called SeqNum.

The second algorithm suggested is called the ACK algorithm. The aim of this algorithm is to reduce the energy consumed during path recovery, while still discovering all potential paths. In addition, the algorithm was our attempt to make a guaranteed loop-free version of the path repair approach suggested in the RFC of RPL, see 2.4.2. A node entering the poisoned state emits a poisoned message, which is acknowledged by all child nodes of the poisoned node. However, child nodes that enter the poisoned state because of the depleted parent node, postpone transmission of ACK until their own initiated poisoning process is finished. Hence, all subDAG nodes are informed before ACK is received, and the poisoned node can therefore safely

reconnect after receiving ACK from all its child nodes. No loop is created during recovery since all former descendants of the poisoned node are informed before ACK is received.

Simulation was used to evaluate the suggested recovery algorithms, and compare them to the Guo algorithm [122] . The main goal was to investigate the recovery algorithms´ ability to reconnect poisoned nodes. The simulation scenario consisted of fixed and randomly deployed nodes, so the network arrangement was not fully random, and this may cause a bias in the simulation results. However, since the aim was to investigate identified path recovery problems, the scenarios were limited to the cases in which nodes enter the poisoned state. Another weakness of the simulation scenario is that the distance to the sink is limited. However, the simulation results can be scaled according to changing network sizes.

The simulation shows that the energy consumed to guarantee the reconnection of broken paths is substantial, see Figure 3 in paper B. The Guo and the ACK algorithms have the lowest message overhead of the three algorithms. However, Guo and ACK cannot guarantee path recovery. The reason that Guo cannot always reconnect is that a deadlock may occur when siblings enter the poisoned state simultaneously. The deadlock occurs if sibling nodes receive requests while they have their own pending request, see 2.4.2. In addition, nodes discard requests originating from their parents. The ACK algorithm cannot always reconnect because ACK messages are sometimes lost.

The SeqNum algorithm adds substantially message overhead, but it is more likely to reconnect path breaks than the Guo and ACK algorithms. Since nodes are likely to be reconnected, periodic global updates are superfluous and can be omitted, which reduces management message overhead and saves energy. In particular, networks in which path breaks rarely occur may profit from replacing periodic updates with the SeqNum approach.

Compared to the simulation results in [122], this study´s results show a substantially poorer path recovery probability for the Guo algorithm. The main difference between the two studies is that a critical node arrangement is analyzed in the current study. In the simulations, the poisoned state is the result of a depleted parent node. Hence, a deadlock may occur if two siblings enter the poisoned state simultaneously. This differs from the approach used in [122], where fluctuations of the channel characteristics are used to create random disconnections. Channel fluctuations cover a limited area, and are therefore less likely to affect sibling nodes simultaneously. Hence, the deadlock explained in 2.4.2 is less likely to occur.

The main weakness of the ACK and SeqNum algorithms is the large computational and message overhead. The ACK algorithm requires that nodes gather and cache child information, and the SeqNum algorithm introduces an extra sequence number that needs to be managed by the nodes. In addition, management packets are exchanged

during path recovery. However, added overhead is required in order to discover all alternative paths and to ensure that the recovered paths are loop-free.

The ACK algorithm fails if packets are lost during path recovery and is therefore not suitable for networks with high packet-loss probability. One of the main disadvantage of the SeqNum algorithm is that the broadcast area can cover the whole network for some scenarios. This is the case if the sink is in one corner of the network and the first path break occurs at the opposite corner of the network. However, since SeqNum is likely to discover all alternative paths, it may make periodic updates superfluous. Thus, periodic routing graph updates can be removed, reducing the energy consumption.

The constructed topology does not provide steady state or stationary results. However, for rare events it can provide acceptable simulation time and better insight. Since steady state or stationary distribution is not our target, the use of special construction topologies is deemed acceptable.

3.3 Paper C

In paper B, it is demonstrated that path recovery mechanisms with a high success rate have a high message overhead, and involve a substantial number of nodes in the path recovery process. A crucial question is therefore whether local recovery algorithms are needed or whether they can be replaced by global recovery. This motivated the analysis contained in paper C. Related work does not evaluate path recovery conforming to the reconnection rules of RPL. Connected RPL paths were therefore analytically modeled to estimate the probability that failed nodes along the path would create path breaks. The connectivity investigation was similar to the related research presented in 2.3 as it considers randomly deployed networks. However, the focus was on connectivity related to paths created by RPL.

The contribution of paper C is twofold. An analytical evaluation was undertaken into the probability that depleted nodes would create path breaks in networks running RPL. Simulations were performed to support the analytical results. In addition, an on-demand global path recovery algorithm to reconnect breaks is suggested. The on-demand algorithm is compared to the ACK algorithm described in paper B and the Guo algorithm.

Analytical calculations were performed to assess the probability of path breaks. The equations used in the assessment were generated by considering nodes located at extreme points in the WSN routing graph. As described in 2.3 , a path recovery algorithm is required when all neighbors of a disconnected node are located further from the sink than the former parent node. In other words, recovery is needed if there is no node left inside the area covered by both the transmission range of the disconnected node and the transmission range of the disconnected node’s grandparent node. In this way, an expression is found for the probability that more than one node is located in the overlapping area between the transmission range of a node and that of its

grandparent. The analytical results were validated by simulations performed in Java [123].

The calculations and simulations show that, in randomly deployed networks running RPL, depleted nodes are likely to create path breaks for all the simulated node densities. The node density is defined as the number of nodes inside the circular area defined by a node's transmission range. When the node density is changed from 8 to 20, the calculations and the simulations reveal that path breaks are likely to occur in 25% to 60% of the cases when a node is depleted. Clearly, the probability for path breaks declines rapidly with increased node density. This is comparable to the findings in papers [108, 109], in which the connectivity is shown to increase rapidly with the number of nodes.

The parent selection procedure is the reason for the counter-intuitive fact that path breaks are also likely to occur in high-density networks. The nodes select parents that display the shortest hop count. Hence, the preferred parents are likely to be located at the border of a node's transmission ranges. This is illustrated in Figure 5, where the dots represent the nodes, and the circles represent the associated transmission range for the nodes with the same color. The farthest left red node selects the blue middle node as a parent in order to reach the black node, which represents the sink. In addition, the preferred parent may also have its parent lying at the border, as illustrated in the figure where the blue middle node selects the sink as its parent. Consecutive nodes along a path find their parent, therefore, near the border. Thus, if one of the first parents in this chain of nodes depletes, a node lying equally close to the border must replace it. For instance, if the blue node in the figure depletes, an alternative backup node must be found in the area where the red circle and the black circle overlap. The closer to the border the original parent node was located, the smaller is the area in which an alternative parent node can be found. Even though the node density is high, the area in which an alternative parent node must be located is so small that there is not likely to be more than one (the original parent node) in it. In order to mend such a path break, a recovery algorithm that enables detours should be included. Clearly, the detours must be loop-free.

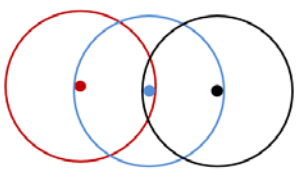

Figure 5 Nodes select parents near the border of their transmission range.

Based on the findings in paper B, 3.2 , that local recovery algorithms increase message and computational overhead, an event-driven recovery approach has been suggested that does not include any local computational overhead. This approach also makes periodic routing graph updates superfluous. Hence, energy can be saved in the long term. The approach is called the on-demand algorithm. The periodic updates are

replaced with global routing graph updates that can be initiated when needed. That is, nodes that enter the poisoned state initiate the global routing graph update by broadcasting an increase-sequence-number message. Broadcasts traverse all possible paths. Hence, the broadcasted message reaches the sink that initiates the global routing graph update as described in 2.4.1. To improve the reliability of the recovery process, the disconnected node may repeatedly transmit the increase-sequence-number. However, the pause between the transmissions must be at least as long as the time required for the associated global recovery process to finish. The message overhead associated with broadcasting the increase-sequence-number can be reduced if nodes located further than a specific number of hops from the poisoned node can change the transmission from broadcast to unicast. However, this reduction in the broadcast area makes it less likely that alternative paths that require long detours will be identified.

The suggested on-demand algorithm clearly has some weaknesses. It may not be an efficient solution in networks with numerous nodes that fail frequently. For instance, networks with unstable links need a mechanism to avoid frequent initiation of global updates. In addition, restoration delay is not taken into account when discussing the event-driven algorithm. Networks that transmit delay sensitive data must clearly take delay into account.

The calculations and the simulations have weaknesses. The analytic model is based on the assumption that the curvature of the border between nodes at different hop distances from the sink approaches a straight line when the distance between sink and nodes is large. However, the border line is, in reality, curved. Both in the model and in the simulations, it is assumed that the packet delivery rate changes from one to zero at the border of the nodes transmission range. In real networks, the delivery rate fluctuates when approaching the border of a node's transmission range. In addition, it is assumed that parent and grandparent nodes create a straight line toward the sink, while it is likely that some nodes would select parent and grandparents that deviate from the straight line between itself and the sink. Moreover, the calculations presented are limited to networks running routing protocols that create a directed acyclic graph. Networks relying on paths based on other principles may give different results.

The graphs in Figure 12 of paper C present the message overheads for the different algorithms. However, the overhead presented depends on parameter settings, and the results for other settings would be different. Thus, the graphs cannot be used for exact prediction, although they show some valid trends.

3.4 Paper D

Paper D considers ways to balance the energy consumption between the nodes in order to reduce the number of times recovery algorithms must be activated. This will, in turn, stabilize the operation of RPL running networks. A survey of proposed balancing methods indicated that several likely candidates had not been sufficiently investigated

and compared. This motivated the analysis presented in paper D of a range of balancing algorithms.

The contribution of paper D is to provide an evaluation of a range of energy balancing algorithms. As discussed in 2.2.3, several algorithms have been proposed for this purpose. However, no prior study has compared the functioning of these algorithms in a common context. In addition to the performance evaluation, three new approaches are suggested that complement the range of algorithms.

As shown in paper A, transmission and receiving are the main energy consuming operations. To balance the consumption, therefore, the forwarding load is shifted from the high-load nodes toward the low-load nodes.

The first and simplest balancing algorithm aims to improve the energy balance in the networks without adding any communication or computational overheads. The algorithm is just a minor change of RPL's preferred parent selection process in which the preferred parent is randomly chosen from among the nodes in the parent-list.

The aim of the second algorithm is to shift data traffic away from nodes that act as a Single Point Of Failure (SPOF). A node acts as a SPOF parent when it is the only member in the parent-list of one or several of its child nodes. Since a depleted SPOF parent partitions the network, it is important to reduce the load on the SPOF parent nodes. Running the SPOF algorithm, nodes weight the traffic between the parents based on the individual parents' residual-energy. Neighboring nodes inform each other about their residual-energy level in emitted DIO messages. However, a node that has a SPOF parent announces its own residual-energy level or that of its SPOF parent, whichever is lower. Thus, the nodes two hops further out shift the selected path away from the SPOF node.

The aim of the third suggested algorithm is to eavesdrop on traffic to continuously monitor neighbors' energy levels. Sender and receiver information included in the eavesdropped traffic is used to adjust the energy information for the associated members of the parent-list. The nodes use the energy information to weight the traffic share among all members of their parent-lists.

The other algorithms assessed have been proposed in the literature. The first one is a simple round-robin algorithm in which the traffic is shared equally between all nodes in the parent-lists. The second and third algorithms weight the traffic share based on the residual-energy of the nodes on their parent-list. In the second, all nodes in the parent-list are used, while in the third, the parent with the lowest residual-energy is avoided. The forth algorithm expands the energy information announced through DIOs by predicting the energy consumption pattern in between the DIO exchanges. The fifth algorithm improves the freshness of the information exchanged through DIO by also including energy information in the ACK packets transmitted. Lastly, in the sixth algorithm the traffic is always sent to the node with the highest level of residual-energy.

In the simulations, the focus was on the residual-energy of the most depleted node, and this was compared to the average residual-energy of the nodes. The smaller the difference, the better the energy balance.

It was found that the optimal energy balancing algorithm is the one that sends the traffic to the highest residual-energy parent node. This result supports the findings presented in [95], where hop count is replaced by ETX. Using the highest energy parent increases the energy level of the most depleted node by 25%. Of the weighting algorithms, the suggested SPOF is the most efficient. This is because, when running the SPOF algorithm, nodes two hops further out than the SPOF node reduce the traffic directed toward the path that includes the low-energy SPOF nodes. Thus, compared to the other weighting algorithms, a lower number of nodes forward traffic toward the specific low-energy nodes. The SPOF algorithm increases the energy level of the most depleted node by 20% compared to the classic RPL.

The use the highest energy nodes among the members of the parent-list, or of the SPOF algorithm, requires the exchange of energy information between nodes. Information exchange causes an insignificant increase in communication expenditure if DIO is used as the information carrier. However, use of DIO means that this information exchange is infrequent in stable networks since DIO exchange frequency is reduced according to the trickle timer [30]. In contrast, our random-parent selection algorithm does not need any exchange of energy information, nor does it add any computational overhead. Nevertheless, it increases the energy level of the most depleted node by 10% compared to classic RPL.

A weakness of the analyzed algorithms is that forwarding decisions are generally based on the parents' energy level, and the potential negative impact on the energy level of nodes further down the path (closer to the sink) is not taken into account. The traffic load on nodes close to the sink may be more balanced if their energy levels are taken into account during path selection. The only algorithm that considers the energy level of nodes more than one hop away is the SPOF algorithm that takes the energy level of the nodes two hop closer to the sink into account. However, taking the energy level of remote nodes into consideration increases the management message overhead because the whole path must be informed about remote path shifts. In addition, the information must be signaled without any significant delay. Delayed path shifts may negatively impact on energy balance, and may even increase the imbalance since the shifts may be made based on obsolete information. In addition, the inclusion of remote nodes may result in an unstable network because one node's path change may cause several higher ranking nodes to perform associated path changes.

As the research in paper A revealed, overhearing is one of the primary energy consuming operations in networks where all nodes overhear the complete transmissions of neighboring nodes. The energy consumed for overhearing has not been included as a factor in the energy balancing algorithms. These algorithms are associated with the network layer, while the energy consumed for overhearing depends

on the MAC layer protocol. In addition, several MAC layer approaches aim to reduce overhearing, and there are protocols, such as TDMA, in which no overhearing occurs.

3.5 Additional study – paper E

The aim of paper E is to reevaluate the energy-optimal transmission range found in paper A, when a more realistic PDR is used. In paper A, the PDR was assumed to change from zero to one at the border of a nodes' transmission range. This is a common assumption, used in many publications. However, paper E presents an expression that models the PDR variation more realistically when the receiving node is located in this position.

The contribution made by paper E in this field is the introduction of a Fermi-Dirac function that suggests a realistic relationship between PDR and transmission distance. In addition, this function is used to provide an expression that facilitates the calculation of the expected number of transmissions along a path in WSN. The number of transmissions reflects the energy consumption. Thus, the expression is used to find the energy-optimal transmission range determined for both TDMA and sparsely deployed CSMA networks.

The average number of transmissions required in order to transmit between source and destination was investigated through simulations of 2D WSNs. The conclusion drawn from the simulations is that it is energy-efficient to use long transmission ranges in networks running TDMA. This is because long transmission ranges reduce the number of hops along the path. This result confirm the conclusion of paper A which states that it is energy-efficient to use long transmission range when there is no overhearing. However, in order to avoid too many retransmissions due to poor PDR, the receiving node should not lie close to the border of the transmitting node's transmission range.

It was also found that, in sparsely deployed networks running CSMA, the path length become ineffectively long if the distance between communicating nodes is too short. In particular, when the transmission range is short, the path between source and sink deviates from the ideal straight line. This is because the probability that there is a node located precisely along this straight line between source and sink declines as the area covered by a transmission is reduced. The result contradicts the findings of paper A, which states that it is energy-efficient to use short transmission range. The reason for the different findings is that paper E focuses on sparsely populated WSN where the number of overhearing nodes are so low that they do not significantly influence on the energy-optimal transmission range. In addition, the energy consumption of the transmitting nodes does not increase with transmission range.

One of the weaknesses of this paper is related to the calculation of energy consumption. The energy consumption for transmitting a packet is assumed to be equal to the number of times the packet is transmitted and received during forwarding from source to sink. However, the energy consumed for transmission also varies with the

transmission range, and this is not taken into account. The reason for this simplification was the focus on the impact of parent node choice at the border of the transmission range, where the PDR starts to change significantly. Clearly, choosing border nodes as parents increases the number of retransmissions due to poor PDR, and the increased number of retransmissions increases the energy consumption.

A best paper award honored papers A and E (additional study). Papers A, C and D were also honored by requests that they be expand for publication in journals.

Chapter 4 - Conclusion

This research has contributed in three different areas of WSN knowledge: energy usage, energy balancing and network connectivity. The overall goal has been to increase the WSN lifetime and improved network availability.

The energy-optimal transmission range in WSN was found to be short. The reason is that reception/overhearing and transmission consume the same order of energy and limiting overhearing with a short range balances the energy needed for additional number of hops. In particular, in networks where no overhearing-reducing mechanism is used, the energy-optimal transmission range is shorter than the range needed to keep the network connected. Although the literature displays a common understanding of the equality between energy consumption during receiving and transmission, its impact on the energy-optimal transmission range is not commonly acknowledged.

Energy balancing algorithms were investigated and compared with respect to total network lifetime in WSN using RPL. It was found that modifying the parent selection procedure, such that a parent is randomly selected among the nodes in the parent-list rather than using a predetermined selection criteria, has a significant balancing effect: The energy of the most depleted node is increased by 10% compared to classic RPL. However, the most efficient balancing effect is achieved when the parent with the highest amount of residual-energy among the nodes in the parent-list is selected. Using the latter selection method increases the energy level of the most depleted node by 25% compared to classic RPL. The weakness is that energy information must be exchanged between the nodes frequently enough to detect changes in energy level between the nodes in the parent-list.

Reducing and balancing energy consumption cannot eliminate the problem that some nodes deplete at a much earlier stage than others. This may destroy the network connectivity. We found that depleted nodes are likely to create path breaks even in dense networks. In particular, for networks running RPL, the path breaks are likely to occur in 25-60% of the cases when a node depletes. These findings are based on analytical calculations and simulations using hop count as the metric.

Repair of disconnected paths is important when it is required that all nodes are capable of reporting their data to the sink. It was found that, in order to make local path

recovery reliable and likely to succeed in finding an alternative path, the number of nodes participating in path recovery must be high. An alternative is to rely on periodic global path updates, with a higher total energy consumption. However, replacing periodic global path update with a method where the disconnected node initiates the global path, makes the recovery process reliable while limiting the energy consumption.

A PhD project has limited resources and as discussed in the methods section, this is reflected in the approach. The results are based on simulation models and theoretical models. Such models are abstractions and the accuracy of the results depends on the validity of the abstraction. Hence, the models represent the real phenomena only to a limited degree. The results should therefore be viewed as indicators of fruitful approaches to explore in actual deployments.

A test bed could to some extent, improve the validity of the results. However, a WSN serves a purpose and collects data from a given set of sensors for different control and management application. This implies that the topology and protocol parameters reflect the usage. A laboratory test bed will not correctly represent such a usage. To test the validity of the results, a wider testing in actual deployment cases is needed. An open issue pertaining to the results is the value of local recovery versus more frequent global repair. We have explored some of the technical details of each of these alternatives. However, the analysis of recovery strategies needs to reflect the actual topology and usage pattern in order to determine the best trade-off. A much needed next step is to use measurement and deployment in copies of industrial WSN representative environments.

Chapter 5 - Bibliography

[1] D. Evans, "The Internet of Things How the Next Evolution of the Internet Is Changing Everything," Cisco, Cisco IBSG2011, Available: http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf, Accessed on: Dec 2016.

[2] (Feb 2016, Dec 2016). *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper*. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[3] M. R. Akhondi, A. Talevski, S. Carlsen, and S. Petersen, "Applications of Wireless Sensor Networks in the Oil, Gas and Resources Industries," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, 2010, pp. 941-948.

[4] P. Pierleoni, A. Belli, L. Palma, M. Palmieri, and L. Pernini, "A WSN Integrated Solution System for Technological Support to the Self-Sufficient Elderly," in *Ambient Assisted Living*: Springer, 2014, pp. 281-290.

[5] K. K. Khedo, R. Perseedoss, and A. Mungur, "A wireless sensor network air pollution monitoring system," *arXiv preprint arXiv:1005.1737,* 2010.

[6] M. Winkler, K.-D. Tuchs, K. Hughes, and G. Barclay, "Theoretical and practical aspects of military wireless sensor networks," *Journal of Telecommunications and Information Technology,* pp. 37-45, 2008.

[7] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad hoc networks,* vol. 7, no. 3, pp. 537-568, 2009.

[8] V. Potdar, A. Sharif, and E. Chang, "Wireless sensor networks: A survey," in *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on*, 2009, pp. 636-641: IEEE.

[9] O. Khader and A. Willig, "An energy consumption analysis of the Wireless HART TDMA protocol," *Computer Communications,* vol. 36, no. 7, pp. 804-816, 2013.

[10] C. M. Sadler and M. Martonosi, "Data compression algorithms for energy-constrained devices in delay tolerant networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, 2006, pp. 265-278: ACM.

[11] V. Raghunathan, C. Schurgers, P. Sung, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *Signal Processing Magazine, IEEE,* vol. 19, no. 2, pp. 40-50, 2002.

[12] L. Wang and Y. Xiao, "A survey of energy-efficient scheduling mechanisms in sensor networks," *Mobile Networks and Applications,* vol. 11, no. 5, pp. 723-740, 2006.

[13] I. Khemapech, A. Miller, and I. Duncan, "A survey of transmission power control in wireless sensor networks," in *Proceedings of the 8th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, 2007, pp. 15-20: Citeseer.

[14] M. N. Moghadam, H. Taheri, and M. Karrari, "Minimum cost load balanced multipath routing protocol for low power and lossy networks," *Wireless Networks,* vol. 20, no. 8, pp. 2469-2479, 2014.

[15] D. P. Dallas and L. W. Hanlen, "Optimal transmission range and node degree for multi-hop routing in wireless sensor networks," in *Proceedings of the 4th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, 2009, pp. 167-174: ACM.

[16] W. Qin, M. Hempstead, and Y. Woodward, "A Realistic Power Consumption Model for Wireless Sensor Network Devices," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on*, 2006, vol. 1, pp. 286-295.

[17] M. Haenggi, "Twelve reasons not to route over many short hops," in *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, 2004, vol. 5, pp. 3130-3134: IEEE.

[18] A. Liu, J. Ren, X. Li, Z. Chen, and X. S. Shen, "Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks," *Computer Networks,* vol. 56, no. 7, pp. 1951-1967, 2012.

[19] P. T. A. Quang and D.-S. Kim, "Enhancing real-time delivery of gradient routing for industrial wireless sensor networks," *Industrial Informatics, IEEE Transactions on,* vol. 8, no. 1, pp. 61-68, 2012.

[20] R. Ramanathan and J. Redi, "A brief overview of ad hoc networks: challenges and directions," *IEEE communications Magazine,* vol. 40, no. 5, pp. 20-22, 2002.

[21] *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, 2012.

[22] S. Kurkowski, T. Camp, and M. Colagrosso, "MANET simulation studies: the incredibles," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 9, no. 4, pp. 50-61, 2005.

[23] (Dec 2016). *OMNET++ Discrete Event Simulator*. Available: https://omnetpp.org

[24] *Industrial Routing Requirements in Low-Power and Lossy Networks*, RFC 5673, 2009.

[25] *Routing Requirements for Urban Low-Power and Lossy Networks*, RFC 5548, 2009.

[26] *Overview of Existing Routing Protocols for Low Power and Lossy Networks*, draft-ietf-roll-protocols-survey-07, 2009.

[27] O. G. P. Levis, "Recommendations for Efficient Implementation of RPL," no. draft-gnawali-roll-rpl-recommendations-05, 2013.

[28] J. T. J. d. O. J. Vasseur, "Performance Evaluation of the Routing Protocol for Low-Power and Lossy Networks (RPL)" *RFC 6687,* 2012.

[29] K. Heurtefeux, H. Menouar, and N. AbuAli, "Experimental evaluation of a routing protocol for wsns: Rpl robustness under study," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on*, 2013, pp. 491-498: IEEE.

[30] *The Trickle Algorithm*, RFC 6206, 2011.

[31] J. Ko, S. Dawson-Haggerty, O. Gnawali, D. Culler, and A. Terzis, "Evaluating the Performance of RPL and 6LoWPAN in TinyOS," in *Workshop on Extending the Internet to Low Power and Lossy Networks (IP+ SN)*, 2011.

[32] *Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)*, RFC 6552, 2012.

[33] *Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks*, RFC 6551, 2012.

[34] H. Kermajani and C. Gomez, "On the network convergence process in RPL over IEEE 802.15. 4 multihop networks: improvement and trade-offs," *Sensors,* vol. 14, no. 7, pp. 11993-12022, 2014.

[35] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM,* vol. 43, no. 5, pp. 51-58, 2000.

[36] (2009, Dec 2016). *AT86RF230*. Available: http://www.atmel.com/Images/doc5131.pdf

[37] (2007, Dec 2016). *CC1000*. Available: http://www.ti.com/lit/ds/symlink/cc1000.pdf

[38] (2007, Dec 2015). *CC2520*. Available: http://www.ti.com/lit/ds/symlink/cc2520.pdf

[39] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, 2000, p. 10 pp. vol.2.

[40] Y. H. Chee, J. Rabaey, and A. M. Niknejad, "A class A/B low power amplifier for wireless sensor networks," in *Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on*, 2004, vol. 4, pp. IV-409-12 Vol.4.

[41] R. Zhang and J.-M. Gorce, "Optimal transmission range for minimum energy consumption in wireless sensor networks," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, 2008, pp. 757-762: IEEE.

[42] A. Gogu, D. Nace, and Y. Challal, "A note on joint optimal transmission range assignment and sensor deployment for wireless sensor networks," in *Telecommunications Network Strategy and Planning Symposium (NETWORKS), 2010 14th International*, 2010, pp. 1-6: IEEE.

[43] G. Xing, C. Lu, Y. Zhang, Q. Huang, and R. Pless, "Minimum power configuration for wireless communication in sensor networks," *ACM Transactions on Sensor Networks (TOSN),* vol. 3, no. 2, p. 11, 2007.

[44] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic, "Optimal transmission radius for energy efficient broadcasting protocols in ad hoc and sensor networks," *Parallel and Distributed Systems, IEEE Transactions on,* vol. 17, no. 6, pp. 536-547, 2006.

[45] J. Ammer and J. Rabacy, "The energy-per-useful-bit metric for evaluating and optimizing sensor network physical layers," in *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, 2006, vol. 2, pp. 695-700: IEEE.

[46] V. Jelicic, M. Magno, D. Brunelli, V. Bilas, and L. Benini, "Analytic comparison of wake-up receivers for WSNs and benefits over the wake-on radio scheme," in *Proceedings of the 7th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, 2012, pp. 99-106: ACM.

[47] N. Kimura and S. Latifi, "A survey on data compression in wireless sensor networks," in *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, 2005, vol. 2, pp. 8-13 Vol. 2.

[48] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung, "MAC Essentials for Wireless Sensor Networks," *Communications Surveys & Tutorials, IEEE,* vol. 12, no. 2, pp. 222-248, 2010.

[49] G. Shi, G. Nan, J. Kou, and R. Rong, "Comprehensive Review of Sleep/Wake Scheduling in Wireless Sensor Networks," in *High Performance Networking, Computing, and Communication Systems*: Springer, 2011, pp. 492-499.

[50] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: a survey," *Communications Magazine, IEEE,* vol. 44, no. 4, pp. 115-121, 2006.

[51] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *Networking, IEEE/ACM Transactions on,* vol. 12, no. 3, pp. 493-506, 2004.

[52] Y. 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*, 2002, vol. 3, pp. 1567-1576 vol.3.

[53] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 95-107: ACM.

[54] A. Bachir, D. Barthel, M. Heusse, and A. Duda, "Micro-frame preamble mac for multihop wireless sensor networks," in *Communications, 2006. ICC'06. IEEE International Conference on*, 2006, vol. 7, pp. 3365-3370: IEEE.

[55] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, 2004, vol. 1, pp. 244-251: IEEE.

[56] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, 2006, pp. 307-320: ACM.

[57] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, 2004, vol. 3, pp. 1534-1539: IEEE.

[58] J. Ai, J. Kong, and D. Turgut, "An adaptive coordinated medium access control for wireless sensor networks," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, 2004, vol. 1, pp. 214-219: IEEE.

[59] T. Van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003, pp. 171-180: ACM.

[60] P. Sthapit and J.-Y. Pyun, "Medium reservation based sensor MAC protocol for low latency and high energy efficiency," *Telecommunication Systems,* vol. 52, no. 4, pp. 2387-2395, 2013.

[61] Y. Li, W. Ye, and J. Heidemann, "Energy and latency control in low duty cycle MAC protocols," in *Wireless Communications and Networking Conference, 2005 IEEE*, 2005, vol. 2, pp. 676-682: IEEE.

[62] C. S. R. M. B. S. Manoy, "Ad Hoc Wireless Networks Architectures and Protocols," ed: Prentice Hall, 2004, p. 857.

[63] *Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Std 802.15.4™-2011, 2011.

[64] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *Wireless Communications, IEEE Transactions on,* vol. 1, no. 4, pp. 660-670, 2002.

[65] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, "Protocols for self-organization of a wireless sensor network," *IEEE personal communications,* vol. 7, no. 5, pp. 16-27, 2000.

[66] N. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *Communications Surveys & Tutorials, IEEE,* vol. 15, no. 2, pp. 551-591, 2013.

[67] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications magazine, IEEE,* vol. 40, no. 8, pp. 102-114, 2002.

[68] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998, pp. 181-190: ACM.

[69] C. K. Toh, H. Cobb, and D. A. Scott, "Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks," in *Communications, 2001. ICC 2001. IEEE International Conference on*, 2001, vol. 9, pp. 2824-2829 vol.9.

[70] C.-S. Ok, S. Lee, P. Mitra, and S. Kumara, "Distributed energy balanced routing for wireless sensor networks," *Computers & Industrial Engineering,* vol. 57, no. 1, pp. 125-135, 2009.

[71] P. O. Kamgueu, E. Nataf, T. D. Ndié, and O. Festor, "Energy-based routing metric for RPL," 2013.

[72] F. Yu, Y. Li, F. Fang, and Q. Chen, "A new TORA-based energy aware routing protocol in mobile ad hoc networks," in *Internet, 2007. ICI 2007. 3rd IEEE/IFIP International Conference in Central Asia on*, 2007, pp. 1-4: IEEE.

[73] V. D. Park, J. P. Macker, and M. S. Corson, "Applicability of the temporally-ordered routing algorithm for use in mobile tactical networks," in *Military Communications Conference, 1998. MILCOM 98. Proceedings., IEEE*, 1998, vol. 2, pp. 426-430 vol.2.

[74] Y. Yu, R. Govindan, and D. Estrin, "Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks," UCLA Computer Science Department, Technical report ucla/csd-tr-01-0023, 2001.

[75] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, 2001, pp. 70-84: ACM.

[76] Z. Gengzhong and L. Qiumei, "A survey on topology control in wireless sensor networks," in *Future Networks, 2010. ICFN'10. Second International Conference on*, 2010, pp. 376-380: IEEE.

[77] S. Jardosh and P. Ranjan, "A Survey: Topology Control For Wireless Sensor Networks," in *Signal Processing, Communications and Networking, 2008. ICSCN '08. International Conference on*, 2008, pp. 422-427.

[78] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM computing surveys (CSUR),* vol. 37, no. 2, pp. 164-194, 2005.

[79] S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. Kumar, "Power control in ad-hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol," in *European wireless conference*, 2002, vol. 2002, pp. 156-162: Florence, Italy.

[80] V. Kawadia and P. R. Kumar, "Power control and clustering in ad hoc networks," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, 2003, vol. 1, pp. 459-469 vol.1.

[81] R. Ramanathan and R. Rosales-Hain, "Topology control of multihop wireless networks using transmit power adjustment," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2000, vol. 2, pp. 404-413 vol.2.

[82] R. Wattenhofer, L. Li, P. Bahl, and W. Yi-Min, "Distributed topology control for power efficient operation in multihop wireless ad hoc networks," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2001, vol. 3, pp. 1388-1397 vol.3.

[83] L. Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks," in *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing*, 2001, pp. 264-273: ACM.

[84] T. M. Chiwewe and G. P. Hancke, "A Distributed Topology Control Technique for Low Interference and Energy Efficiency in Wireless Sensor Networks," *Industrial Informatics, IEEE Transactions on,* vol. 8, no. 1, pp. 11-19, 2012.

[85] J. D. Day and H. Zimmermann, "The OSI reference model," *Proceedings of the IEEE,* vol. 71, no. 12, pp. 1334-1340, 1983.

[86] T. Melodia, M. C. Vuran, and D. Pompili, "The state of the art in cross-layer design for wireless sensor networks," in *Wireless systems and network architectures in next generation internet*: Springer, 2006, pp. 78-92.

[87] L. D. Mendes and J. J. Rodrigues, "A survey on cross-layer solutions for wireless sensor networks," *Journal of Network and Computer Applications,* vol. 34, no. 2, pp. 523-534, 2011.

[88] V. Srivastava and M. Motani, "Cross-layer design: a survey and the road ahead," *Communications Magazine, IEEE,* vol. 43, no. 12, pp. 112-119, 2005.

[89] C. Suh, Y.-B. Ko, and D.-M. Son, "An energy efficient cross-layer MAC protocol for wireless sensor networks," in *Advanced web and network technologies, and applications*: Springer, 2006, pp. 410-419.

[90] Z. Qian and Z. Ya-Qin, "Cross-Layer Design for QoS Support in Multihop Wireless Networks," *Proceedings of the IEEE,* vol. 96, no. 1, pp. 64-76, 2008.

[91] P. Di Marco, G. Athanasiou, P.-V. Mekikis, and C. Fischione, "MAC-aware Routing Metrics for the Internet of Things," *arXiv preprint arXiv:1310.4632,* 2013.

[92] M. R. Akhavan and T. Watteyne, "Enhancing the performance of RPL using a receiver-based MAC protocol in lossy WSNs," in *Telecommunications (ICT), 2011 18th International Conference on*, 2011, pp. 191-194: IEEE.

[93] B. Pavković, F. Theoleyre, and A. Duda, "Multipath opportunistic RPL routing over IEEE 802.15. 4," in *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, 2011, pp. 179-186: ACM.

[94] H. K. A. Willig and H. Karl, "Protocols and architectures for wireless sensor networks," *England: John Wiley & Sons,* 2005.

[95] C. Abreu, M. Ricardo, and P. Mendes, "Energy-aware routing for biomedical wireless sensor networks," *Journal of Network and Computer Applications,* vol. 40, pp. 270-278, 2014.

[96] K. S. Shivaprakasha and M. Kulkarni, "Energy Efficient Shortest Path Routing Protocol for Wireless Sensor Networks," in *Computational Intelligence and Communication Networks (CICN), 2011 International Conference on*, 2011, pp. 333-337.

[97] O. Iova, F. Theoleyre, and T. Noel, "Improving the network lifetime with energy-balancing routing: Application to RPL," in *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP*, 2014, pp. 1-8.

[98] C. Lin-Huang, L. Tsung-Han, C. Shu-Jan, and L. Cheng-Yen, "Energy-Efficient Oriented Routing Algorithm in Wireless Sensor Networks," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, 2013, pp. 3813-3818.

[99] S.-S. Chiang, C.-H. Huang, and K.-C. Chang, "A minimum hop routing protocol for home security systems using wireless sensor networks," *Consumer Electronics, IEEE Transactions on,* vol. 53, no. 4, pp. 1483-1489, 2007.

[100] R. Vidhyapriya and P. Vanathi, "Energy efficient adaptive multipath routing for wireless sensor networks," *IAENG International Journal of Computer Science,* vol. 34, no. 1, pp. 56-64, 2007.

[101] Y. Ming Lu and V. WS Wong, "An energy-efficient multipath routing protocol for wireless sensor networks," *International Journal of Communication Systems,* vol. 20, no. 7, pp. 747-766, 2007.

[102] H. Fariborzi and M. Moghavvemi, "EAMTR: energy aware multi-tree routing for wireless sensor networks," *IET communications,* vol. 3, no. 5, pp. 733-739, 2009.

[103] L. Kleinrock and J. Silvester, "Optimum transmission radii for packet radio networks or why six is a magic number," in *Proceedings of the IEEE National Telecommunications Conference*, 1978, vol. 4, pp. 1-4.

[104] P. Erdös and A. Rényi, "On random graphs, I," *Publicationes Mathematicae (Debrecen),* vol. 6, pp. 290-297, 1959.

[105] H. K. DeWitt, "The theory of random graphs with applications to the probabilistic analysis of optimization algorithms," 1977.

[106] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, 2002, pp. 80-91: ACM.

[107] M. D. Penrose, "On k-connectivity for a geometric random graph," *Random Structures & Algorithms,* vol. 15, no. 2, pp. 145-164, 1999.

[108] X. Xing, G. Wang, J. Wu, and J. Li, "Square region-based coverage and connectivity probability model in wireless sensor networks," in *Collaborative Computing: Networking, Applications and Worksharing, 2009. CollaborateCom 2009. 5th International Conference on*, 2009, pp. 1-8: IEEE.

[109] W. Jia, Y. Fu, and J. Wang, "Analysis of connectivity for sensor networks using geometrical probability," in *Embedded and Ubiquitous Computing*: Springer, 2004, pp. 601-611.

[110] Y. Wang, B. M. Kelly, and X. Li, "On the network connectivity of wireless sensor networks following a random and non-uniform distribution," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on*, 2013, pp. 69-74: IEEE.

[111] W. Xie *et al.*, "Routing loops in DAG-based low power and lossy networks," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, 2010, pp. 888-895: IEEE.

[112] K. Vaidya and M. Younis, "Efficient failure recovery in Wireless Sensor Networks through active, spare designation," in *Distributed Computing in Sensor Systems Workshops (DCOSSW), 2010 6th IEEE International Conference on*, 2010, pp. 1-6.

[113] M. F. Younis, S. Lee, and A. A. Abbasi, "A localized algorithm for restoring internode connectivity in networks of moveable sensors," *Computers, IEEE Transactions on,* vol. 59, no. 12, pp. 1669-1682, 2010.

[114] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, and F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: A survey," *Computer Networks,* vol. 58, pp. 254-283, 2014.

[115] *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, 2003.

[116] *Dynamic MANET On-demand (AODVv2) Routig*, draft-ietf-manet-dymo-23, 2012.

[117] *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, RFC 4728, 2007.

[118] J. Garcia-Luna-Aceves and H. Rangarajan, "A new framework for loop-free on-demand routing using destination sequence numbers," in *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, 2004, pp. 426-435: IEEE.

[119] D. Tian and N. D. Georganas, "Energy efficient routing with guaranteed delivery in wireless sensor networks," in *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, 2003, vol. 3, pp. 1923-1929: IEEE.

[120] Q. Lampin, D. Barthel, and F. Valois, "Efficient Route Redundancy in DAG-based wireless sensor networks," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, 2010, pp. 1-6: IEEE.

[121] J. Tripathi, J. de Oliveira, and J. Vasseur, "Applicability study of RPL with local repair in smart grid substation networks," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, 2010, pp. 262-267: IEEE.

[122] J. Guo, C. Han, P. Orlik, J. Zhang, and K. Ishibashi, "Loop-Free Routing in Low-Power and Lossy Networks," in *SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications*, 2012, pp. 59-66.

[123] K. Arnold, J. Gosling, D. Holmes, and D. Holmes, *The Java programming language*. Addison-wesley Reading, 2000.

Chapter 6 - Appendix 1 Errata

Paper B errata:

Table 1 and the associated text is incorrect. The correct text and table should say:

The first row in Table 1 represents the state of the network after the node dies and before any recovery algorithm is run. Hence, a dying node always disconnects one or several nodes unless a recovery algorithm is activated. The three last rows of Table 1 show the share of simulation runs in which one or several nodes remain unconnected after the recovery algorithm is finished. Applying any of the recovery algorithms reconnects several of the disconnected nodes. Applying the J. Guo algorithm, all nodes gets reconnected in 40% of the simulations. Running ACK method, nodes are reconnected in 99% of the simulations, while applying the SeqNum approach, nodes are reconnected in all the simulations.

TABLE 1

	Share of simulations resulting in unconnected nodes
Standard RPL, without any recovery	100 %
J.Guo	59 %
SeqNum	0 %
ACK	0,5 %

PART II

Research Papers

Paper A:

Energy Reduction in Wireless Sensor Networks by Switching Nodes to Sleep During Packet Forwarding

Anne-Lena Kampen, Knut Øvsthus and Øivind Kure.

Proceedings of the 6th International Conference on Sensor Technologies and Applications (SENSORCOMM' 2012), pp. 189-195, 2012, ISBN: 978-1-61208-207-3

Energy Reduction in Wireless Sensor Networks by Switching Nodes to Sleep During Packet Forwarding

Anne-Lena Kampen[1,2], Knut Øvsthus[1]
[1]Bergen University College, Bergen
[2]ITEM NTNU, Trondheim
Norway
{Anne-Lena.Kampen | knut.ovsthus}@hib.no

Lars Landmark, Øivind Kure
Centre for Quantifiable Quality of Service in Communication Systems* (Q2S)
NTNU, Trondheim
Norway
{ larsla | okure }@q2s.ntnu.no

***Abstract*—Energy consumption determines the lifetime of Wireless Sensor Networks, WSN. In current radio chip sets the energy consumption for receiving a packet is of the same order as transmitting a packet. In such a setting, the transmission range and sleep strategies should be reevaluated. We present a simple extension to the MAC protocol that reduce the waste of energy for processing packets not addressed to a node by letting them sleep during transmission. The nodes enter sleep mode by means of a Transmission Announcement packet, TAN, sent by the transmitter. The performance is evaluated through simulation. Based on a simplified model, we show that the optimal transmission range in such a setting is given by the minimum needed to avoid partitioning. We use data sheet values from three different WSN Transceiver modules to derive parameter values to be used in the model. The model and related analysis concentrates on the energy consumption in transmitting and receiving, since the radio is the main contributor to energy consumption in WSN. We show that it is the energy consumption in receiving that is the main contributor to total energy consumption in WSN.**

Keywords-WSN; Energy Consumption; Sleep control; Optimal transmission range

I. INTRODUCTION

One of the most active research areas in Wireless Sensor Networks (WSN) concerns reduction of the energy consumption of the nodes to increase the lifetime of the WSN. A WSN node consists of several units such as the microcontroller, the memory and the radio, which consumes most energy [1]. Various energy efficient protocols have been proposed to reduce the radio energy consumption. These may be categorized as topology control protocols and sleep management protocols. Topology control protocols use hierarchies and transmission control to limit the number of neighbors (covered nodes) of a node to only those required to avoid network partitioning [2]. This is achieved by reducing the transmission power, and thus shortening the transmission range. But reducing transmission range may degrade the capacity of the network. In their seminal paper, Takagi and Kleinrock determined that the optimal transmission range is when the expected number of neighbors is 8 [3]. However, their work focused on the capacity, and they did not consider the energy consumed listening to packets. Hence, the optimal number of neighbors in order to maximize the lifetime is not evident. Reasons to avoid routing over many short hops are discussed in [4]. Among the listed reasons are interference, energy consumption, path efficiency and end-to-end reliability.

Sleep management protocols schedule redundant nodes to enter sleep mode in order to reduce energy consumption [5]. However, there exist no sharp distinctions between the two mentioned categories, as they may utilize each other qualities to get a more energy efficient network.

Information collected from datasheets for three different WSN Transceiver modules [6][7] shows that the receiving energy consumption is of the same order as transmission energy consumption. In addition, the average number of nodes in a randomly deployed WSN increases quadratic with transmission range, leading to a step increase of energy consumption as transmission range increases. Energy optimization in such a setting requires short transmission range or switching redundant receivers to sleep mode.

The contributions of this article are threefold. First, we present a simple model for calculating the total energy consumption in WSN, taking all the receiving nodes into account. Using the model, we analyze the energy optimal transmission range based on parameters from datasheets for three different WSN Transceiver modules. Last we present a simple energy efficient forwarding approach based on the findings in the analysis. The forwarding approach put redundant nodes to sleep as packets are forwarded.

The rest of the article is organized as follows: related work is introduced in Section 2; the energy consumption model are presented in Section 3; parameter estimation and analysis are done in Section 4; energy preserving forwarding is

*Centre for Quantifiable Quality of Service in Communication Systems, Centre of Excellence, appointed by The Research Council of Norway, and funded by the Research Council, NTNU and Uninett (http://www.q2s.ntnu.no)"

described in Section 5, and related simulations are presented in Section 6; Section 7 presents the conclusion.

II. BACKGROUND AND RELATED WORK

Two main classes of energy optimization solution are described herein. Sleep management and topology control.

Redundant nodes in a densely deployed network may switch to sleep mode without negatively affecting the communication. Sleep protocols may be divided in two groups: local-area-based approaches and backbone-based approaches. In the local-area-based approaches, a node's mode is determined by the mode of the neighboring nodes, and redundant nodes enter sleep mode if it does not negatively affect the connectivity of the network. Examples of protocols in this group are the ones presented in [8][9] [10][11] and [12]. The backbone-based approach selects some nodes to stay active to constitute the backbone of the network. These nodes are responsible for relaying data and scheduling the other nodes to operate in low duty cycles. Clustering is one way of creating a backbone network, in which the clusterhead constitutes the backbone. In LEACH [13], the network is partitioned into clusters and a clusterhead is responsible of organizing the communication in the cluster. Manjeshwar et al. [14] presented an enhanced clustering by letting redundant nodes alternate in handling quires from the clusterhead to avoid unnecessary communication cost. In [15], gridding and clustering are combined in a grid-based clustering technique and the energy-optimizing grid size is evaluated. An overview of sleep management protocols is provided in [5]. Generally, sleep management protocols add synchronization overhead, and are prone to added delay. Our energy optimized forwarding, called Transmitting Announcement (TAN), differentiates from traditional sleep modes. TAN does not require synchronization, and is totally decentralized by simply switching nodes to sleep mode as data packets progress to sink. A detailed description of the approach is given in Section 5.

Topology control approaches adjust the nodes output power to limit the energy consumption of the network. WSN are generally densely populated networks, hence the nodes' output power may be reduced without negatively affecting the connectivity of the network. ATPC [16] proposes a feedback scheme whereby the nodes find the optimal transmission power level for each individual neighbor dynamically. The smallest common transmission power that results in a connected network is found in COMPOW [17], and this power is used by all nodes. CLUSTERPOW [18] integrates routing table information and transmission range to optimize topology control. Dynamic adjustment of transmission range based on node degree is investigated in LINT and LILT [19]. Another example is the one used in CBTC [20], where transmission power is adjusted to reach one neighbor in every sector of a specific degree around the node. A third example is to use graph models, such as used in GG and RNG [21]. They minimize energy consumption by using relay node if this reduces transmission range. An overview of topology control issues and approaches is presented in [2]. Analysis of the energy optimal transmission range is given in Section 4.

There exist several energy consumption models for WSN [22][23][24][25]. However, few of the proposed models consider the receiving energy consumed by the nodes not forwarding packets. These nodes only receive packets to discard them, thus waist energy. The model presented in [26] includes all receivers as data are forwarded from source to destination. However, the distance between the nodes changes as the transmission range change. Hence, the number nodes within the range of a transmitter are constant. The model computes energy consumption for broadcasting. Opposed to the model in [26], the distance between nodes in our model is constant. Hence, the number nodes within the range of a transmitter changes with transmission range. Further, we consider unicast transmission. The energy consumption model is presented in Section 3.

III. ENERGY CONSUMPTION MODEL

Our goal is to investigate the relationship between the nodes transmission range, and total network energy consumption. The aim is to determine the energy optimal transmission range for a given node density. We focus only on the energy used for packet transmission and packet reception. Our scenario is a WSN where the nodes are randomly distributed.

The analysis of energy consumption assesses a source node that is located at a distance D from the sink, without accounting for the network edges. The energy consumption for transmitting data packets depends on the amplifier architecture. A common model for energy consumption per bit has a constant level, k_1, that is independent of the radiated power, plus an offset, k_2, proportional to the radiated power [27]. All nodes have the same transmission range d. Hence, the minimum number of times the packet has to be relayed to reach the sink is D/d. The expression for the energy required to transmit b bit of data is [27]

$$\mathbf{E_{TX}} = (\mathbf{k_1} + \mathbf{k_2} * \mathbf{d^2}) * \frac{\mathbf{D}}{\mathbf{d}} * \mathbf{b}. \qquad (1)$$

In addition, we assume that the energy a node uses for receiving data is constant equal to k_3, energy consumed per received bit [27]. The total number of active nodes receiving data is proportional to the density of active nodes, λ, times the area covered by the emission. The consumed energy per bit for one transmission accounting for the number of receivers is thus, $\pi d^2 * \lambda * k_3$. As stated above, the data must be relayed to reach the destination. Hence, the total consumed energy has to be multiplied by the number of times the data is relayed, D/d. The total energy consumed by nodes that receive b bits becomes

$$\boldsymbol{E_{RX}} = (\boldsymbol{k_3} * \boldsymbol{\pi d^2} * \boldsymbol{\lambda}) * \frac{\boldsymbol{D}}{\boldsymbol{d}} * \boldsymbol{b}. \qquad (2)$$

The total energy consumed in relaying the data from the source node to the sink is calculated by adding (1) and (2).

$$\mathbf{E_{TOT}} = \mathbf{b} * \frac{\mathbf{D}}{\mathbf{d}}((\mathbf{k_1} + \mathbf{k_2} * \mathbf{d^2}) + (\mathbf{k_3} * \boldsymbol{\pi}\mathbf{d^2} * \boldsymbol{\lambda})) \quad (3)$$

Our analysis is with respect to optimal transmission range. Constants that have no influence on the result are omitted for simplicity. The expression is normalized with respect to the constant level of the transmission energy.

$$\mathbf{E_{TOT,NORM}} = \frac{\mathbf{1}}{\mathbf{d}} + \left(\frac{\mathbf{k_2}}{\mathbf{k_1}} + \frac{\mathbf{k_3}}{\mathbf{k_1}} * \boldsymbol{\pi} * \boldsymbol{\lambda}\right) * \mathbf{d} \quad (4)$$

By differentiating (4), the energy optimum transmission range is.

$$\mathbf{d_{opt}} = \sqrt{\frac{\mathbf{1}}{\frac{\mathbf{k_2}}{\mathbf{k_1}} + \frac{\mathbf{k_3}}{\mathbf{k_1}} * \boldsymbol{\pi} * \boldsymbol{\lambda}}} \quad (5)$$

IV. DATASHEET-BASED ESTIMATIONS

In this section, the parameter values for k_1, k_2 and k_3 are estimated based on values extracted from datasheets, and the optimal transmission range are calculated using these parameter values. Three Transceiver modules are investigated: AT86RF230 [6], CC2420 and CC1000 (_868 and _433) [7]. The datasheet [6][7] provides data for transmission with different output powers, and power measurements for receiving, idle and sleep modes. Power measurements are converted to energy by multiplying with the bit-time calculated from the bit-rate of the Transceiver modules, which is 250 kbit/s for AT86RF230 [6] and CC2420 [7], and the highest bitrate for CC1000 [7] is 76.8 kbit/s.

The parameters k_1 and k_2 are estimated based on the relationship between transmission range and output power, which may be expressed by rearranging Friis [28] equation:

$$\mathbf{P_r} = \frac{\mathbf{P_t} * \mathbf{G_t} * \mathbf{G_r} * \left(\frac{\mathbf{c}}{\mathbf{f}}\right)^2}{\mathbf{16} * \boldsymbol{\pi}^2 * \mathbf{d^n}} \quad (6)$$

Rearranging (6) gives:

$$\mathbf{d} = \left(\frac{\mathbf{P_t} * \mathbf{G_t} * \mathbf{G_r} * \left(\frac{\mathbf{c}}{\mathbf{f}}\right)^2}{\mathbf{16} * \boldsymbol{\pi}^2 * \mathbf{P_r}}\right)^{\mathbf{1/n}} \quad (7)$$

The parameters used in these equations are as follows. P_r is the power received by an antenna through free space, P_t is the transmitted power, G_t is the transmitting antenna gain, Gr is the receiving antenna gain, c is the speed of light and d is the distance between the antennas. The red curves in Fig. 1 are plotted using (7) with datasheet values for P_r and P_t, using antenna gain of 1.64, which is the gain of a half wave dipole antenna, and choosing path loss exponent n=3 [29][30]. The curves show output power versus transmission range. To find k_1 and k_2 we need to define the red curves by their corresponding second order equations as k_1 and k_2 represent the parameter values in these second order equations (multiplied by bit-time to convert form power to energy). We use curve fitting to find the equations.

The middlemost of the blue dotted curves in Fig. 1 presents calculated curve fitted lines. The equations for these curves are presented in the respective display. Multiplying the parameter values in these equations with bit-time gives k_1 and k_2. The other two dotted curves show the fitted curve with a +/- 10% change of parameter values, indicating that the real values for k_1 and k_2 are within +/-10%.

The receiving power consumption is illustrated by the straight green line. k_3 is derived by multiplying receiving power consumption and bit-time.

Based on the equations for the curve fitted line for CC1000_868, the values for k_1, k_2 and k_3 are 36.1µJ/bit, 0.06pJ/bit/m2 and 37.5 µJ/bit respectively. Choosing λ=0.1 active nodes/m^2 give an optimal d=1.75m using (5). The average number of covered nodes is then 0.96. Performing the same calculations for CC2420, AT86RF230 and CC1000 433 gives optimal distances of 1.2, 1.4 and 1.7, and average number of covered nodes of 0.45, 0.62 and 0.96, respectively. The required number of neighbors to avoid partitioning is 4 according to the discussion presented in [31] that is based on results from [32][33]. As the calculated number of neighbors is lower than 1, the network is partitioned. Hence, using the energy optimal transmission distance, d, would probably lead to network partitioning.

A. Critical parameters regarding energy consumption

In order to present a clear understanding of the critical parameters determining the energy efficient transmission range, the derivative of the total energy consumption (4) with respect to range is rearranged as:

$$\left(\frac{\mathbf{k_2}}{\mathbf{k_1}}\right)\mathbf{d^2} + \left(\frac{\mathbf{k_3}}{\mathbf{k_1}}\right) * \boldsymbol{\pi} * \boldsymbol{\lambda} * \mathbf{d^2} = \mathbf{1} \quad (8)$$

The term, $\pi d^2 * \lambda$, is equal to the number of active nodes receiving data. Clearly, there must be at least one active receiver in order to make any progress in forwarding, this implies that $\pi d^2 * \lambda$ must be larger than 1. In (8), this means that there is no real value for d that gives a minimum point if k_3 approaches k_1. Estimations of the parameters based on datasheet [6] and [7] indicate that $k_2 << k_1$, and that $k_1 \approx k_3$,see above. Thus, the receiving energy consumption, k_3, is the main contributor to the short transmission length. The reason is that a linear increase of transmission range, d, causes an increase proportional to d^2 in the number of receiving nodes. This result is consistent with the result of the simulations in [34]. Hence, given that $k_3 \approx k_1$, these findings imply that topology control protocols should aim to reduce the transmission range as much as possible.

Fig. 2 shows how the node density impacts the energy optimal transmission range. Increased node density increases the number of receivers, thus, reducing the optimal transmission range. The values used for the parameters k_1, k_2 and k_3 reflects the relationship between the values as found above.

Keeping the number of receiving nodes constant would reduce the impact of k_3 on the optimal transmission range, and thus the total energy consumption.

V. TRANSMISSION ANNOUNCEMENT ,TAN, USED FOR ENERGY REDUCTION

The analysis in Section 4 shows that the receivers are the main contributor to the total network energy consumption. In WSN, generally all nodes within the transmitter vicinity receive the transmitted packet. However, according to the routing protocol, only one, or a subset, of the receivers are assign to forward the packet. The remaining nodes waste energy as they receive the packet just to discard it.

Our proposal is to reduce energy consumption by preventing nodes form receiving packets not intended to them. This is done by the transmitting node. It prevents nodes from receiving ordinary data packets by sending a short signaling packet prior to the data packet.

The proposed data forwarding approach is as follows. Nodes within the range of the transmitter radio, except for the next-hop node, are switched to sleep mode using a signaling packet called TAN. The packet carries the transmission time for the following data packet, and is addressed to the next-hop node determined by the routing table. All nodes receiving the TAN packet not destined to them change to sleep mode during the corresponding data packet transmission. Radios in sleep mode do not amplify receiving data, which prevents the MAC layer form receiving data. The length of the sleeping period is: (2*SIFS) + (ACK length) + (Data packet length). SIFS is the waiting time between transmitting TAN and the data package, in addition to the waiting time between receiving a data package and transmitting ACK. TAN is only used for unicast transmission, since broadcast and multicast are intended for more than one receiver.

The conditions for TAN to be advantageous compared to plain Carrier Sense Multiple Access (CSMA) depend on: the ratio between data and TAN packet size, node density, and the distance between transmitter and receiver. The requirements on the data packet size are found by estimating the breakeven point when energy consumption using TAN equals the energy consumption using CSMA.

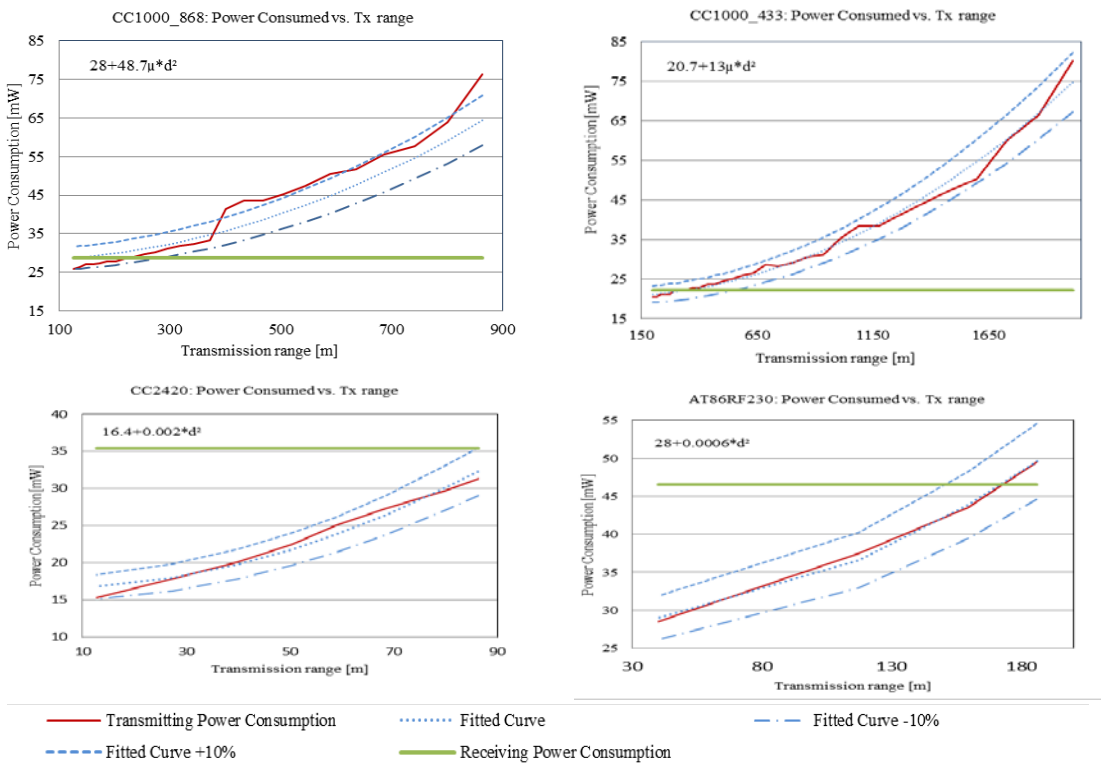

Figure 1. Red curve: power consumption vs. transmission range based on datasheet values. Blue curves: the curve fitted power consumption with +/- 10% change of parameter values. Green curve: receiver power consumption.

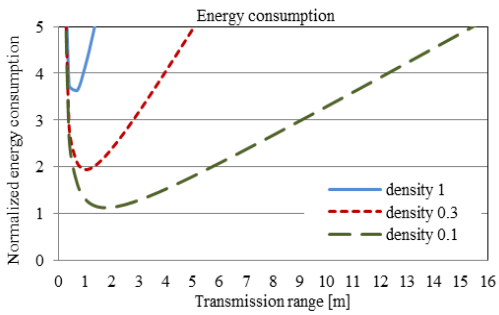

Figure 2. Total normalized energy consumption for sending from a source to the sink. k_1=1, k_2=0.005 and k_3=1.

The breakeven point depends on the localization of the receiver inside the sender's transmission range, and two extreme cases are calculated: (1) when the transmitter and receiver share all neighboring nodes (co-located sender and receiver) and (2) when the receiver is localized on the circumference for the sender's transmission range.

In the first case, the TAN energy consumption for a one hop communication is: $k*(N+1)*b_{TAN} + k*2*b_{Data} + k*2*b_{ACK}$, where the average number of neighbors is N, $b_{reference}$ is the number of data-bits in the referenced packet-type, and the receiving and transmitting energy consumption per bit is assumed to be equal (k). In the second case, the number of nodes receiving ACK increases, and is exactly those nodes that are inside the area of the receiver's transmission range but outside the sender's transmission range. This crescent shaped area may be calculated based on the formulas described in [31], and the number of nodes in the area is found by multiplying by the node density. Thus, the TAN energy consumption for the second cases is: $k*(N+1)*b_{TAN} + k*2*b_{Data} + k*(1 + N - 2\lambda d^2\left(\frac{\pi}{3} - \frac{\sqrt{3}}{4}\right))*b_{ACK}$. The energy consumption using plain CSMA is for both cases: $k*(N+1)*b_{Data} + k*(N+1)*b_{ACK}$.

Based on the equations in the paragraph above and the assumption that ACK and TAN packets size are equal ($b_{ACK}=b_{TAN}$), the breakeven point for case one is:

$$b_{Data} > \frac{2}{N-1} * b_{TAN} \tag{9}$$

Equation (9) shows for N larger than 3, TAN is advantageous even for data packet smaller than the TAN packet. Note that this occurs for co-located source and destination nodes, which is probably rarely the case as it would result in no progress of the forwarded packet.

By using the fact that $N=\lambda\pi d^2$, the equation for the breakeven point for case two is:

$$b_{Data} > \frac{\frac{N}{3}+1+\frac{\sqrt{3}N}{2\pi}}{N-1} * b_{TAN} \tag{10}$$

TAN preserves energy, according to (10), if the data packet is smaller than the TAN packet when the number of neighbors is larger than ~ 5.2. On the average, the number of neighbors needed to make TAN energy efficient for data packet size no bigger than TAN packet sizes, lies between these two extreme values, 3 and 5.2. Clearly, the breakeven data packet size is reduced with an increased number of neighbors.

VI. SIMULATIONS

We evaluate our forwarding scheme in an extension of the OMNET ++ simulator [35] with the MiXiM module for wireless communication. The simulator is extended to separate the receiving and idle energy consumption, and to implement TAN. Our simulations are validated against analytic results.

The simulations compare the energy consumption for relaying unicasting traffic, using a plain CSMA MAC layer protocol and our TAN. The comparison is made by measuring the energy consumption when transmitting 1000 data packets from source to the sink. Edge effects are avoided by placing both the source and the sink at a distance from the edge of the network that is longer than the maximum transmission range. Data is transmitted using the maximum 802.15.4 data packet size, 127 bytes at PHY layer [36]. The size of the TAN packet used in the simulations is 30 bytes. Three scenarios with different number of nodes are simulated. The nodes are placed in a random pattern inside an area of 570 x570 m. The distance between source and the sink is 382 m. The presented simulated results are averaged over 30 simulation runs with different seeds for random deployment of nodes. RPL [37] is used for routing, and the routing tables in the nodes are completed before any data is being forwarded.

Simulations performed to compare the total average energy consumption for varying output power levels are shown in Fig. 3. The output power values are chosen based on datasheet values for CC2420. The simulated scenarios consist of 400 nodes. The related 95% confidence intervals are shown in the figure.

Figure 3 shows that the total network energy consumption is lower in TAN than in plain CSMA. In plain CSMA, the number of redundant receivers increases with increased output power. The energy consumption for next highest output power level is higher than for the highest output power level. This counter intuitive result is traced back to a higher hop count that outweighs the increase in the number of covered nodes. The added number of hops resulted in more transmissions draining more energy.

TAN has only one receiver for each transmission. However, there is a tiny increase of energy consumption as output power increases. It is caused by a higher number of receivers receiving the ACK packet sent from the receiving nodes. Similar to the CSMA, TAN experiences an increase in energy for the next highest power. The added energy consumption is caused by the increase in number of hops, and the corresponding number of ACKs. Note that, there is no

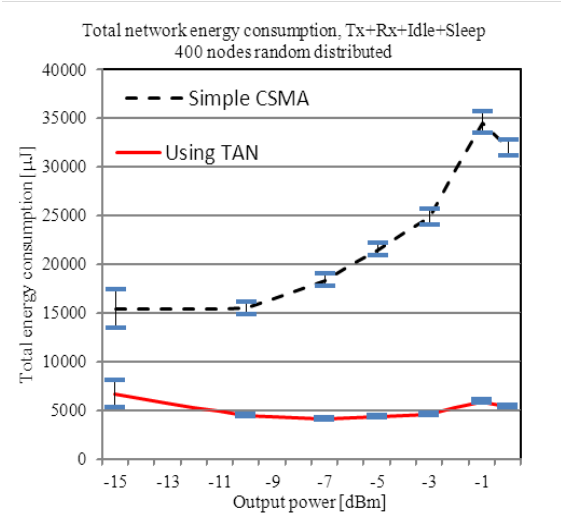

Figure 3. Energy consumption for transmitting 1000 packets in a network consisting of 400 nodes.

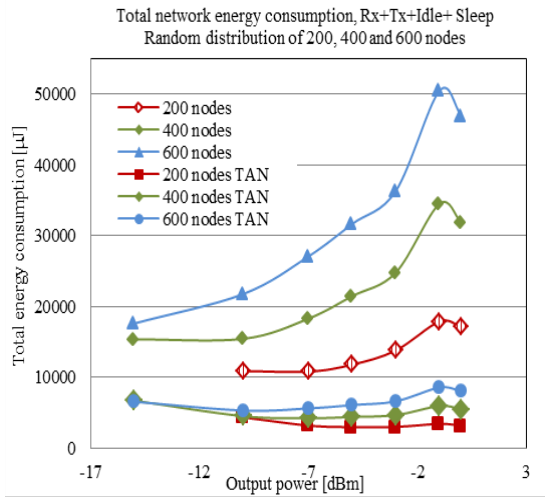

Figure 4. Energy consumption for transmitting with different node densities

difference in packet forwarding as routing is equal for both ordinary CSMA and TAN.

The broader 95% confidence interval at the output power level of -15dBm is caused by the larger deviation in path length. In addition, due to the low node density, some of the simulations at -15dBm do not have a connection from the sensor to the sink. These simulations are omitted as no results with respect to energy consumption due to data transfer are produced. The 95% confidence interval narrows as the output power increases.

Forwarding energy for different node densities versus transmission range is shown in Fig. 4. As expected, the difference between the plain CSMA and TAN increase with increased node density. This means that the advantage of using the TAN increases as the node density increases. Change of data packet size would give similar results. An increase in packet size would lead to higher difference between the TAN and the plain CSMA.

Simulations for -15 dBm output power are omitted in for the 200 nodes scenario in Fig. 4. The reason is that the network is partitioned for these low output powers.

If the number of neighbors is low, no energy is preserved when using TAN as no redundant nodes receives the transmitted data packets. Hence, if the simulation in Fig. 3 were extended with result for lower output powers, the graphs would eventually merge as the number of neighbors approaches one. Likewise, the graphs in Fig. 4 would merge for very low node densities.

Loss of data packet occurs if the intended receiver is in sleep mode caused by TAN packet received from another node. However, these packets would otherwise be destroyed by collision from the ongoing transmission. Thus, the number of lost packets is the same as with CSMA. The solution to avoid losing these packets is to combine TAN with RTS/CTS

VII. Conclusion

Datasheets for WSN Transceiver modules shows that the receiving and transmitting energy consumption are of the same order of magnitude. Furthermore, the average number of receivers increases quadratic with transmission range in a randomly distributed network. Thus, the energy optimal transmission range is short. We calculate the range using parameter values estimated based on datasheet information. The calculation is performed using an energy consumption model that we present. The range is shorter than the minimum needed to avoid network partitioning. The required number of neighbors to keep a network connected is 4 according to [31] and its references, but the calculated optimal range covers less than one neighboring node. Thus, in order to energy optimize a WSN network the transmission range must be kept just large enough to ensure a connected network.

However, if the number of receivers is fixed, the receiving energy consumption is also fixed. Hence, we propose a solution that reduces the number of receivers to consist of only the next hop node towards the sink. The solution is a simple sleep management approach that makes redundant nodes switch to sleep mode during transmission of data packets. A small signaling packet sent prior to the unicast data packets announces the transmission. Simulations compare the proposed approach against simple CSMA using the maximum 802.15.4 packet size. The simulations show that there is a great reduction in total energy consumption when using the proposed approach. The energy savings depends on data packet size and node density.

REFERENCES

[1] Enz, C.C., Scolari, N., and Yodprasit, U.: Ultra Low-Power Radio Design for Wireless Sensor Networks. IEEE Proceedings. International Workshop on Radio-Frequency Integration Technology, pp. 1-17. (2005)

[2] SANTI, P.: Topology Control in Wireless Ad Hoc and Sensor Networks. ACM Journal, Computing Surveys, Vol. 37, No. 2, pp. 164-194. (2005)

[3] Takagi, H. and Kleinrock, L.: Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals. IEEE Transactions on communications, Vol. 32, No. 3, pp. 246-257. (1984)

[4] Haenggi, M.: Twelve Reasons not to Route over Many Short Hops. IEEE Vehicular Technology Conference, Vol. 5, pp. 3130-3134. (2004)

[5] Wang, L. and Xiao, Y.: A Survey of Energy-Efficient Scheduling Mechanismsin Sensor Networks. ACM Journal, Mobile Networks and Applications, Vol. 11, No. 5, pp. 723-740. (2006)

[6] Atmel http://www.atmel.com/Images/doc5131.pdf. (April 24, 2012)

[7] Chipcon Products from Texas Instrument:
- CC2420: http://www.ti.com/lit/ds/symlink/cc2420.pdf. (April 24, 2012)
- CC1000: http://focus.ti.com/lit/ds/symlink/cc1000.pdf. (April 24, 2012)

[8] Xu, Y., Heidemann, J., and Estrin, D.: Geography Informed Energy Conservation for Ad Hoc Routing. ACM Proceedings of the 7th annual International Conference on Mobile computing and networking, pp.70-84. (2001)

[9] Ye, F., Zhong, G., Cheng, J., Lu, S., and Zhang, L.: PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks. IEEE In Proceedings of 23rd International Conference on Distributed Computing Systems, pp.1-10. (2003)

[10] Bulut, E. and Korpeoglu, I.: Sleep scheduling with expected common coverage in wireless sensor networks. ACM Journal Wireless Networks, Vol. 17, No. 1, pp. 19-40. (2011)

[11] Yardibi, T. and Karasan, E.: A distributed activity scheduling algorithm for wireless sensor networks with partial coverage. ACM Journal, Wireless Networks, Vol. 16, No. 1, pp. 213-225. (2010)

[12] Kumar, S., Lai, T.H., and Balogh, J.: On k-Coverage in a Mostly Sleeping Sensor Network. ACM Proceedings of the 10th annual international conference on Mobile computing and networking, pp. 144-158. (2004)

[13] Heinzelman, W.B., Chandrakasan, A.P., and Balakrishnan, H.: An Application-Specific Protocol Architecture for Wireless Microsensor Networks. IEEE Transactions on Wireless Communications, Vol. 1, No. 4, pp.660-670. (2002)

[14] Manjeshwar, A., Zeng, Q.A., and Agrawal, D.P.: An Analytical Model for Information Retrieval in Wireless Sensor Networks Using Enhanced APTEEN Protocol. IEEE Transactions on Parallel and Distributed Systems, Vol. 13, No. 12, pp. 1290-1302. (2002)

[15] Zhuang, Y., Pan, J., and Wu, G.: Energy-Optimal Grid-Based Clustering in Wireless Microsensor Networks. 29th IEEE International Conference on Distributed Computing Systems Workshops, pp. 96-102. IEEE. (2009)

[16] Lin, S., Zhang, J., Zhou, G., Gu, L., He, T., and Stankovic, J.A.: Atpc: Adaptive Transmission Power Control for Wireless Sensor Networks. ACM, In Proceedings of the 4th international conference on Embedded networked sensor systems,pp.223-236. (2006)

[17] Narayanaswamy, S., Kawadia, V., Sreenivas, R.S., and Kumar, P.R.: Power Control in Ad Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW protocol. In Proceeedings of European Wireless Next Generation Wireless Networks: Technologies, Protocols, Services and Applications, pp. 156-162. (2002)

[18] Kawadia, V. and Kumar, P.R.: Power Control and Clustering in Ad Hoc Networks. IEEE, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, Vol. 1, pp. 459-469. (2003)

[19] Ramanathan, R. and Rosales-Hain, R.: Topology Control of Multihop Wireless Networks using Transmit Power Adjustment. IEEE, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 2, pp. 404-413. (2000)

[20] Li, L., Halpern, J.Y., Bahl, P., Wang, Y.M., and Wattenhofer, R.: A Cone-Based Distributed Topology-Control Algorithm for Wireless Multi-Hop Networks. IEEE/ACM, Transactions on Networking, Vol. 13, No. 1, pp.147-159. (2005)

[21] Jaromczyk, J.W. and Toussaint, G.T.: Relative neighborhood graphs and their relatives. IEEE Proceedings of the IEEE, Vol. 80, No. 9, pp. 1502-1517. (1992)

[22] Wang, Q., Hempstead, M., and Yang, W.: A Realistic Power Consumption Model for Wireless Sensor Network Devices. IEEE, Sensor and Ad Hoc Communications and Networks, pp. 286-295. (2006)

[23] Gogu, A., Nace, D., and Challal, Y.: A note on joint optimal transmission range assignment and sensor deployment for Wireless Sensor Networks. IEEE, Telecommunications Network Strategy and Planning Symposium, pp. 1-6. (2010)

[24] Xing, G., Lu, C., Zhang, Y., Huang, Q., and Pless, R.: Minimum Power Configuration for Wireless Communication in Sensor Networks. ACM, Transactions on Sensor Networks, Vol. 3, No. 2, Article 11, pp. 1-33. (2007)

[25] Zhang, R. and Gorce, J.M.: Optimal Transmission Range for Minimum Energy Consumption in Wireless Sensor Networks. IEEE, Wireless Communications and Networking Conference, pp. 757-762. (2008)

[26] Ingelrest, F, Simplot-Ryl, D, and Stojmenovi, I.: Optimal Transmission Radius for Energy Efficient Broadcasting Protocols in Ad Hoc and Sensor Networks. IEEE,

Transactions on Parallel and Distributed Systems, Vol. 17, No. 6, pp. 536-547. (2006)

[27] Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. IEEE, Proceedings of the Hawaii International Conference on System Sciences, pp.1-10. (2000)

[28] Friis, H.T.: A note on a simple transmission formula. In Proceedings of I.R.E., pp. 254-256. (1946)

[29] Andersen, J.B., Rappaport, T.S., and Yoshida, A.: Propagation Measurements and Models for Wireless Communications Channels. IEEE, Communications Magazine, Vol.33, No. 1, pp.42-49. (1995)

[30] Seidel, S.Y. and Rappaport, T.S.: 914 MHz Path Loss Prediction Models for Indoor Wireless Communications in Multifloored Buildings. IEEE, Transactions on antennas and propagation, Vol.40, No.2. pp.207-217. (1992)

[31] Kleinrock, L. and Silvester, J.: Optimum transmission radii for packet radio networks or why six is a magic number. Proceedings of the IEEE National Telecommunications Conference, pp.431-435, (1978)

[32] Erdos, P. and Renyi, A.: On Random Graphs I. Publications Mathematicae, Debrecen 6, pp. 290-297. (1959)

[33] Dewitt, H.: The Theory of Random Graphs with Applications to the Probabilistic Analysis of Optimization Algorithms. Ph.D. Dissertation, Department of Computer Science, U.C.L.A., Los Angeles. (1977)

[34] Dallas, D. P. and Hanlen, L. W.: Optimal Transmission Range and Node Degree for Multi-hop Routing in Wireless Sensor Networks, ACM Proceedings, Workshop on Perfomance Moitoring and Measuremet of Heterogenous Wireless and Wired Networks, pp. 167-174. (2009)

[35] OMNET++ : http://www.omnetpp.org/. (April 24, 2012)

[36] IEEE Std 802.15.4™-2003. Chapter 6, Digital Object Identifier : 10.1109/IEEESTD.2003.94389. (2003)

[37] ROLL IETF Internet-Draft : RPL: IPv6 Routing Protocol for Low power and Lossy Networks. draft-ietf-roll-rpl-19 [work in progress], 13 Mar (2011)

Paper B:
Reconnection strategies in WSN running RPL

Anne-Lena Kampen, Knut Øvsthus and Øivind Kure

39th Annual IEEE Conference on Local Computer Networks Workshops (LCN2014), pp. 602-609, 2014, Electronic ISBN: 978-1-4799-3784-4, DOI: 10.1109/LCNW.2014.6927709

Reconnection Strategies in WSN Running RPL

Anne-Lena Kampen[1,2], Knut Ovsthus[1]
[1]Bergen University College, Bergen
[2]ITEM NTNU, Trondheim
Norway
[1]{Anne-Lena.Kampen | knut.ovsthus}@hib.no
[2]anneleka@stud.ntnu.no

Øivind Kure
NTNU, Trondheim
Norway
okure@item.ntnu.no

***Abstract*— The contribution of this article is two local recovery methods that may reduce overhead and make a more immediate reconnection of disconnected nodes compared to the soft state global recovery method applied in WSNs running RPL [1]. In addition, we perform an analysis of the implication to better judge the context where local recovery methods should be used.**

One of our suggested methods is based on reliable local repair and produces a complete connected network in 99.5% of the simulations. The overhead is in the same order of magnitude as applying the local recovery method suggested by J.Guo in [2]. However, the simulation shows that the J.Guo method has a worse reconnection percentage. Our second proposal is based on local-sequence number and creates high recovery overhead. However, it guarantees recovered network connectivity. Thus it makes periodic network recovery superfluous.

Index Terms—*WSN, RPL, recovery, reconnection, energy, looping.*

I. INTRODUCTION

Wireless sensor networks, WSN, has a wide range of applications, ranging from geriatric care [3] [4] to military applications [5]. Handling such a wide range of applications is hardly possible to achieve with a single realization of WSN. However, some common challenges exist. One such challenge is to recover network connectivity after a node dies [6]. Loss of network connectivity affects data throughput and may lead to network partition.

The connectivity loss issues are related to disconnected nodes that do not have alternative paths with the same cost to the destination as the broken path. According to sink directed routing protocols, these are nodes which do not have alternative neighbor nodes on the same level as their former successor (parent) nodes. The disconnected nodes are referred to as being poisoned nodes. The only option for reconnecting the poisoned nodes is through nodes further away from the sink. Such reconnection may result in routing loops if not controlled by a proper designed recovery method. Recovery methods may be divided into two classes; global and local recovery. Using local recovery, nodes exchange routing information with local nodes only. The information exchanged and the implemented algorithm for reconnection ensures loop-free paths.

Global recovery, on the other hand, delays the reconnection until the next global update recovers the whole network. One of the main disadvantages of the global method is the relation between the recovery delay and the traffic overhead created by the periodic updates. When the recovery delay is reduced the management overhead increases, which causes increased energy consumption. Routing Protocol for Low-Power and Lossy Networks RPL [1], developed by IETF uses global recovery.

A local repair method is suggested by J.Guo in [2]. The Internet-Draft [7] is based on the idea presented in [2], and is suggested as a local recovery method in RPL. In this method, new loop-free paths are created for the poisoned node by assigning candidate intermediate nodes a new rank. However, predecessors (children) of the poisoned node are not considered as candidate intermediate nodes. Hence, if the only possible new path goes through a former child node, new paths are not found. In addition, other nodes that are poisoned do not participate in developing new available paths. Thus, if the common parent of sibling nodes dies, a race condition occurs, and the available paths may not be found.

We suggest two new local recovery approaches that find all candidate paths. The first method is based on sequence number, abbreviated SeqNum. Each node caches a local-sequence number used to indicate the freshness of paths. Poisoned nodes use the local and global sequence number to discover new feasible paths.

We further suggest a method where the poisoned node reliable informs children nodes about its state, i.e. all child nodes acknowledge the poisoning message. After receiving acknowledgement from all children, the poisoned node may safely connect to nodes further away from the sink without any risk of creating a loop.

The contribution of this article is twofold. We present two new local recovery algorithms. In addition, we present an analysis of the recovery algorithms that contribute to assess different network scenarios most optimal recovery approach.

A. Presentation of RPL

The routing entries in a network running RPL create a Destination Oriented Directed Acyclic Graph (DODAG) that is directed from the nodes toward the sink. Each node participates in the DODAG formation by performing specific rank computation, and parent selection algorithms.

Initially, the sink broadcasts a Destination Information Option (DIO). The DIO includes information about the sink's rank, which is the globally lowest rank in the network. Receiving nodes use this information to choose the lowest rank neighbor as its parent, i.e. the node ad an edge to the DODAG. Their own rank is calculated relative to the chosen parent node. Subsequently, the node transmits DIO including its rank. Receiving nodes perform the same calculations and selection. Thus, the DODAG formation spreads like a wave outward from the sink toward leaf nodes as all nodes adds a new edge. The graphs are acyclic caused by the parent selecting algorithm which prohibits nodes form selecting parents further away from the sink than themselves. The rank of the participating nodes is strictly decreasing toward the sink.

Sequence number is used as a mean to update the DODAG and prevent routing loops in RPL. The sequence number is set by the sink and is part of the information included in the DIO. Only the sink is allowed to change the sequence number. A node has the same sequence number as its preferred parent. Hence, all nodes that are member of the same DODAG have equal sequence number. More recent DODAGs are preferred over older one. Thus, nodes increase their sequence number as DIO with higher sequence number arrive. When a node changes its sequence number, it is required to transmit a DIO. Thus, when the sink increments the sequence number, updated DIOs traverse the network updating the DODAG.

Nodes running RPL are restricted from increasing their rank as long as their sequence number is unchanged. The motivation is to prohibit looping: a node allowed to increase its rank may choose nodes in its own sub-DAG as parent. However, a node may increase its rank when it receives updated information from the sink, as indicated by an increased sequence number. Receiving a message containing an updated sequence number proves that the sender is not part of the receiver's own sub-DAG. Hence, the sender may safely be chosen as the receiver's parent.

The loop avoidance technique of RPL may for some node arrangement introduce recovery delay according to the DAG updating interval. The delay occurs if a disconnected node does not have alternative neighboring nodes with equal or lower rank than its former parent. The disconnected node is then in a poisoned state, and has to wait for next sequence number update to reconnect to the DAG.

RPL has a solution that reduces the delayed repair of a poisoned node. The solution is based on poisoning of the sub-DAG. A poisoned node transmits a poisoning message to inform its children that the path to the sink is broken. Receiving nodes delete the sender from the parent set. Subsequently, the poisoned node may increase its rank to re-attach to the DAG. However, the poisoning message may not reach all neighbors, leaving routing entries in child nodes unaltered. Therefore, there may be nodes with routing entries that are still directed toward the poisoned node. Thus, the poisoned node may create a loop by choosing nodes in its own sub-DAG as a parent.

B. Presentation of the J.Guo- method

The J.Guo loop-free recovery method is based on a rank calculation method that enables nodes to change rank without changing their relative distance to the sink. This is achieved by defining the rank R, as a proper fraction with nominator integer, m, and a denominator integer, n, such that $0 \leq m < n$. The fraction ensures that it is always possible to allocate a rank in between any two former consecutive ranks. Thus, a node may perform a rank decrease without violating its position relative to the sink. The rank change enables former higher level nodes to select the rank decreasing node as a parent without creating loops. The rank is cached as two integers, and the fraction value is only used when performing rank operations.

The rank split operation used to calculate a rank in between two given ranks $R_1 = m / n$ and $R_2 = p / q$ is given by:

$$sp (R_1, R_2) = (m + p) / (n + q) \qquad (1)$$

It is easily shown that if $R_1 < R_2$, then $R_1 < sp(R_1 , R_2) < R_2$.

The root (sink) rank is defined as 0/1 and infinite rank is defined as 1/1. The modification of the J.Guo approach relative to RPL, relates to the rank calculation. The exchange of DIOs is not changed. The nodes choose the minimum rank, $R_M = m_M / n_M$, among its candidate parents nodes, and sets its own rank, R_n > parent rank. $R_n \leq sp(R_M, 1 / 1)$. Subsequently, it transmits DIOs containing its rank information. The process continues throughout the whole network.

The local repair mechanism is based on transmission of two new type of messages, DODAG Repair Request (DR-REQ) and DODAG Repair Reply (DR-REP). A poisoned node broadcasts a DR-REQ containing information such as node identifier, N_q, and rank, $R(N_q)$. A detailed description of the receiver's action is found in [2]. The reply message is a reversed version of the DR-REQ created path.

Based on the exchange of messages, a monotonically increasing rank is made along the path from a replying node toward the requesting node. The intermediate node's rank is calculated as follows: Each node, N_i, that has a rank $R(N_i)$ that is higher than the requesting node's rank $R(N_q)$ decreases its rank to: $R(N_i) = sp(R(N_q), R(N_p))$, where $R(N_p)$ is the sender of the received DR_REP.

In our opinion, the following issues are not thoroughly assessed in the J.Guo approach:

- If the only path available is through one or more child nodes, that path will never be found. The reason is that requests from parent nodes are discarded;
- A race condition occurs if a node dies causing two or more sibling nodes to lose their parent. If the only possible way to reach a node with lower rank goes through other siblings, this path may never be found. The reason is that nodes with pending requests will not respond to request from other nodes; and
- Regular DIOs are sent according to the trickle timer algorithm in network running RPL. This may cause problems to the intermediate nodes on recovery paths. An intermediate member on an alternative path for a poisoned node changes rank according to J.Guo rank-split operation. However, receiving an ordinary DIO, the intermediate node performs an ordinary rank calculation. This rank may be higher in magnitude than the rank calculated participating in local recovery. Hence, the node may need to initialize the local recovery procedure as it is not allowed to increase its rank.

The main benefit of the J.Guo method is that the message distributed is generally limited to one level below and above the poisoning node. This implies a low signaling overhead. An additional benefit is that nodes may rapidly rejoin the DODAG, as a node may immediately re-attach to the network when the replay is received.

The rest of this paper is organized as follows: Section II presents the related works; Section III presents the suggested SeqNum method, while the ACK method is presented in section IV; In Section V we perform simulations to estimate the probability that dedicated recovery methods are needed to mend reconnected nodes in randomly deployed networks; Section VI describes the simulation scenarios used when comparing the recovery methods, and the results of the simulations are discussed in section VII; Finally, the conclusion of our work is presented in Section VIII.

II. RELATED WORK

The ad hoc routing protocols DSDV [8] and [AODV] [9] use sequence number in order to discover new loop-free paths. According to DSDV, new successor nodes are only chosen when a node receives information with updated sequence number, or when it receives information from a node with equal sequence number but shorter path. AODV is an on-demand routing protocol where sources flood the network with route requests to create a path to the destination. The route requests consist of a label containing the source identifier and the current destination sequence number. Only nodes with an unexpired entry for the destination and a sequence number as least as great as the number indicated in the request, can answer the request. If none intermediate nodes have replied, the destination replies the request with its current sequence number. The reply is relayed following the reverse path created by the request.

In the framework presented in [10] Garcia-Luna-Aceves and Rangarajan introduce a sequence number window. Intermediate nodes choose a sequence number inside the window such that the number is higher than each predecessor, hence the number monotonically increases in the path from the source to the destination. Using a window of sequence numbers enables nodes on the path toward the destination to answer route request if they have an active route to the destination with a higher sequence number than the requesting node. In [11] the same authors expand the intermediate nodes ability to answer requests. A node may answer a request if it has a valid route with a fresher sequence number or a shorter path to the destination than all traversed nodes.

ROAM [12] uses two feasibility successor conditions to maintain loop-free routes: passive and active. A passive feasible successor is the next hop node that gives the shortest path to the destination. Active feasible successors are the next hop nodes that report a distance to the destination that is lower than the cost from the source to the destination through the passive feasible successor. Hence, both the path through the passive and active feasible successor is guaranteed to be loop-free. If the link to the passive feasible successor fails, the active feasible successors are chosen as the next hop node. Nodes initialize a diffusing search by broadcasting a query if the available passive feasible successor does not give the shortest possible distance or if the node has no entries for the requested destination.

Studies on fault tolerant ad hoc network as presented in [13] diagnoses the nodes to avoid using faulty nodes as part of the routing path. A disconnected node transmits a request for a new parent. The disconnected node selects its new parent among the fault-free nodes in its neighborhood. Nodes that are child of the disconnected node transmit their own requests. However, the suggested approach does not discuss how to avoid loops that may appear as a result of lost requests.

An on-demand routing protocol using labels to discover loop-free routes is presented in [14]. Split Label Routing (SLR) is applied to calculate the node's labels, and keep a strict hierarchical arrangement of the nodes. Labels are calculated and used in a similar way as in the J.Guo approach. A difference between the J.Guo method and SLR is how children treat requests. A request originated by a receiver's parent is discarded using the J.Guo method. In SLR, there is no parent check, and the request is forwarded carrying a rank label which is the lowest of the received rank label and the forwarder node's own rank. This means that an eligible path going through a child node can be discovered.

Local loop avoidance techniques versus global loop recovery in DAG networks are evaluated in [15]. Based on simulated results it was concluded that the turmoil introduced using local loop avoidance made it improper for general use in WSN. However, limiting network recovery to global updates may not be feasible in all types of networks. Avoiding use of any local recovery methods means that global recovery has to be performed frequently enough to meet the networks' availability demands, i.e. recovery delay demands. Hence, the

overhead may be substantial in network with strict availability demands and should be taken into account when comparing global and local recovery. This issue is not discussed in [15].

In this article we simulate, compare and assess three different loop avoidance approaches. One approach is the proposal of J.Guo [2]. The second is based on sequence number, while the third is based on reliable poisoning of sub-DAG. In addition we discuss global versus local recovery.

III. SEQNUM METHOD

In this section we present our proposed local recovery method based on local-sequence number. The goal is to develop a local path repair mechanism that both avoids loops and enables discovering of all eligible paths, i.e. including paths through children or sibling nodes. It uses local-sequence number to ensure that only fresh, loop-free paths are selected during local recovery.

A poisoned node initiates local recovery by broadcasting a request. The request contains the transmitting node's global sequence number, current local-sequence number increased by one, and a TTL field. Receiving nodes use the received information to decide whether it is eligible to reply to the request.

Nodes that are unable to reply, cache the identifier of the sender, reduce the TTL field, and re-broadcast the request. The cache is used to avoid duplicate transmission of the request and to establish return path for the reply messages. Re-broadcasting is terminated when the request reaches a node qualified to reply, or when the TTL field equals zero.

The request is replied by nodes with equal global sequence number and more recent local-sequence number than the requesting node or nodes with more recent global sequence number. The more recent local-sequence number shows that the associated path does not include any of the traversed nodes. Otherwise, the previous node would have replied to the request. If the sink receives a request, it will update the local-sequence number. The local-sequence number is signaled in the associated reply. Replies are unicasted hop by hop toward the requesting node according to the entries established by the associated request. Traversed nodes save the updated local-sequence number which enables them to answer requests from nodes with older local-sequence numbers. Receiving the reply, the requesting node may increase its rank to join the suggested path as it is guaranteed loop-free. Next, the recovered node transmits a DIO message. The DIO message includes the updated local-sequence number, enabling neighboring poisoned nodes to get reconnected.

Nodes with a more recent global sequence number than the requesting node reply received requests. The more recent number proves that the receiver is connected to the sink through a path not including the poisoned node. The reply includes the updated global sequence number enabling all the nodes on the reverse path, including the requesting node, to become connected to the updated DODAG.

Transmission of ordinary DIO is postponed when node is poisoned to avoid unnecessary DIO transmission before a new stable parent is found.

Some nodes may achieve different rank when local vs. global recovery is performed. The reason is twofold. Transmitted DIOs may be lost due to collisions during regular global network updates. This may leave nodes unaware of their lowest-rank neighbor. However, during local recovery, packets from the lowest-rank node may reach the former unaware neighbors which then change parent and rank. In addition, in dense network, the nature of the trickle timer may prevent certain nodes form transmitting DIO. The reason is that the trickle timer decides that a node should not send DIO if it receives a specific number of consistent DIOs during a specific time interval. A received DIO is consistent if it does not result in any rank or parent change at the receiving node. Hence, the nature of the trickle timer may cause nodes to be unaware of a neighbor that may be a more favorable parent. However, the node may be known at later global or local network updates due to the randomness of the parameters in the trickle timer.

Issues related to the SeqNum approach:

- Requests and/or replies may be lost; and
- The request is relayed all the way to the sink if none of the intermediate nodes have performed any local updates since last global sequence number update. Thus, a poisoned node located far from the sink will result in an almost network wide broadcast depending on the sink location relative to the disconnected node. This may be alleviated by allowing nodes with lower rank than the requesting node to respond to received requests. However, the responding node cannot change the local-sequence number, but may instead include a specific flag in its reply.

The main advantage of the SeqNum method is the assurance that an available path is always found if it exists.

IV. ACK METHOD

Our second proposed method is based on reliable poisoning of the sub-DAG to discover new loop-free paths. To make the poisoning process reliable, nodes need be aware of their children nodes. The ordinary DIO messages are modified to provide this information in a 'sender's parent' field.

The recovery process is initiated when a node enters poisoned mode and transmits a poisoned DIO message. The message informs receiving nodes that the sender has lost connection to the sink. Receivers update their neighbors list according to the poisoning information. All children of a poisoned node acknowledge the reception of the poisoning message. However, children with no alternative new parents enter poisoned state and complete their own poisoning before transmitting the ACK message. The poisoned node keeps track of the responding nodes, and when all its children have acknowledged the poisoning message it may increase its rank and re-attach to the DODAG without any risk of creating a loop.

Timers are used to reduce the probability for route flapping and associated message overhead. Route flapping may occur if nodes choose parent among nodes that are still waiting to transmit a poisoning message or among nodes that entered poisoning state simultaneously with the choosing node. To prevent such flapping situation, all nodes receiving or transmitting a poisoned DIO postpone transmission of ordinary DIO for a pre-defined period of time, Δt. ACK messages and poisoned DIO messages are the only legal management traffic during Δt. Hence, all nodes in the area get overview of neighboring poisoned nodes before selecting their new parent. Remember that all nodes entering poisoning state send poisoning message. When the time Δt expires, each individual node chooses current best parent among its neighbors. However, they do not immediately transmit a DIO; instead, the transmission is delayed once more. This time the delay is based on a timer which magnitude is proportional to the rank calculated when choosing the currently best parent. Thus, the timer for the lowest rank node fires first and the node transmits an ordinary DIO. Receivers of the ordinary DIO perform best parent calculation and update their timers. Then the timer for the node with the next best rank fires and the node transmits its ordinary DIO and so forth. Eventually, all affected nodes have chosen their best parent and transmitted their ordinary DIOs. As the node with the lowest rank starts the transmission, the neighboring nodes get updated with the best available parent before their timer expires.

The time interval for transmission of poisoned DIO, as well as transmission of ACK, is restricted to reduce the collision probability. A poisoning message is transmitted at a time t_p after receiving the poisoning messages. t_p is randomly picked in the interval [0, D1]. ACK messages are transmitted at a randomly picked time in the interval [D1, Δt]. In the simulation D1 and Δt are set to 0.95s and 1.75s respectively.

Issues related to the ACK approach:

- ACK messages may never reach the poisoned node, disabling a parent node from increasing rank and re-attach to the DAG. To reduce the probability of this situation to appear, nodes re-send poisoned DIOs when an expected ACK is not received; and
- The recovery time is negatively affected by the delay time Δt introduced to reduce route flapping and message overhead.

The advantage of the ACK approach is that all eligible paths are found unless several management packets are lost. In addition, the overhead is limited to the vicinity of the nodes that are affected by the path disconnection.

V. PROBABILITY OF POISONING STATE

This section presents the probability that dedicated recovery methods are required to mend disconnections. The results are based on simulations. No dedicated recovery method is needed if feasible paths to the sink are available. According to RPL, a feasible path exists if there are neighboring nodes with equal or better rank than the former parent.

The probability we are investigating is:
Prob (there is *more* than 1 eligible parent node among the node's neighbors | given that there is at least 1 eligible parent node among the node's neighbors).

A. Simulation-dedicated recovery required

The simulation scenario consists of a 700m X 700m area. All nodes have equal transmission range of 141m. The nodes are positioned randomly in the area. The simulation result presented is the average value of 30 runs at each node density value. Each run uses different seed for random deployment of nodes. The node density is defined as the average number of nodes inside a circle with radius equal to the nodes' transmission range.

The routing protocol used is RPL with hop-count as the metric. The simulation is performed using Omnet++ [16] with the MiXiM module. The MAC layer protocol used is CSMA with parameter according to the IEEE 802.15.4 standard [17]. This applies for all presented simulations.

The graph in Fig. 1 shows the probability that dedicated repair methods is *not* needed. As expected, the probability is a function of the node density and increases rapidly at low densities. However, even at high node densities the probability is still significantly lower than one. The reason is that some of the nodes will always select a parent node near the edge of its transmission range. In addition, the grandparent is located close to the edge of the parent node's transmission range. These nodes have the optimal location according to the metric of the routing protocol. However, dedicated recovery methods may be needed as none of the disconnected nodes' other neighbors are at an equal or better rank than the parent node. This result may be a motivation to change the parent selection mechanism to make a more stable network.

According to [18] and its references, eight is the magic number of neighbors ensuring high throughput. Thus, we may assume that an average well designed network has a node density between 8 and 20. Assuming such a well design network, the probability that dedicated recovery methods are *not* needed is between 60% and 85% due to the graph in Fig. 1. In other words, between 15 - 40% of the recovery actions require dedicated recovery methods.

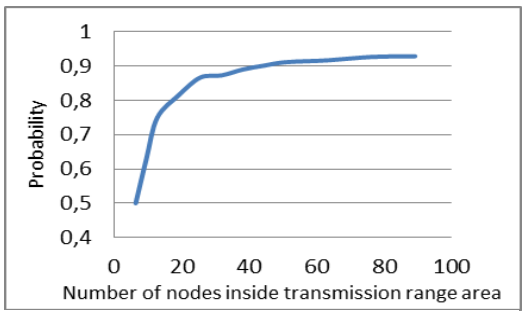

Fig. 1. Probability that dedicated recovery method is *not* needed

VI. SIMULATION – COMPARING RECOVERY METHODS

Most of the random deployment scenarios do not require dedicated recovery methods as shown in Fig. 1 and discussed above. Therefore, we select an artificial topology to ensure that all simulations contribute to the issue under investigation. The disadvantage with this topology is that it does not unveil how nodes more than one hop from the poisoned nodes are affected. However, the main impact of the repair mechanisms is in the vicinity of the poisoned node and the impact decreases as the distance from the dying node increases. The reason is that more alternative paths become available.

A. Simulations scenarios – recovery methods

The node arrangement used in the simulation is shown in the right-hand side of Fig. 2. The left-hand side of Fig. 2 illustrates a general random WSN where the central node named S represent the sink. The two red dashed lines encircling the sink represent areas for different rank nodes. The nodes located between the sink and the inner red circle represent nodes at one-hop distance from the sink, while the nodes lying between the two red circles represent nodes at two-hop distance from the sink. The right-hand side of Fig. 2 represents a segment of the left hand side figure. The segment is illustrated by the black frame in the left hand side figure. The five nodes shown in the right-hand side figure have fixed positions during the simulations. A topology change is enforced by deactivating the node named 'dying'. The location of the dying node is such that at least one of the children, that is n_b, enters poisoned state. A varying number of nodes are randomly scattered at a one-hop distance from the poisoned node, n_b. None of the random scattered nodes are located closer, i.e. at a lower hop distance, to the sink than n_b. Otherwise, the closer located nodes would make dedicated recovery unnecessary. The blue shaded area in the right-hand side figure illustrates the permitted area for the random scattered nodes.

Overhead, convergence time, number of unconnected nodes and number of nodes changing rank due to recovery are investigated. The overhead affects the network energy consumption; thus, it should be as low as possible. The overhead is the sum of the number of management packets transmitted and received when performing dedicated recovery.

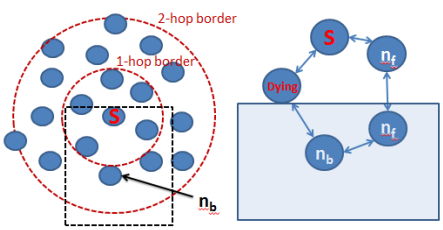

Fig. 2. Node arrangement used simulating the different recovery methods.

Packets sent and received by the dead node and the sink are not included, as they do not affect the energy consumed.

The convergence time is measured as the time from the node dies to the time the last recovery DIO is received.

VII. RESULTS

In this section J.Guo, SeqNum, ACK and standard RPL are compared. The comparison is based on simulating each recovery strategy using the scenario described above. The simulation parameters are equal to the setup used for the 'dedicated-recovery-required' simulations, except that the simulated results are averaged over 60 simulation iterations. The node deployment is as illustrated in Fig 2. The 95% confidence interval for the graphs is shown as horizontal lines above and below the marks with the associated color. The x-axis shows the number of neighbors of the poisoned node which equals the number of nodes inside the blue rectangle in the right-hand side of Fig. 2.

Figure 3 presents the overhead due to local recovery for the three different methods; J.Guo, SeqNum and ACK. As expected, the overhead increases with increasing number of neighbors for all methods. Standard RPL does not include a specific local recovery mechanism; hence, it is not included in the figure.

SeqNum have the highest overhead, while the overhead using J.Guo and ACK are in the same order of magnitude. The reason for the high overhead using SeqNum is due to the artificial simulation topology which makes all nodes participate in the recovery process. Requests from every node entering poisoning mode are broadcasted throughout the network. In addition, all nodes send DIO after recovery. Hence, in small network the SeqNum method may give an increased overhead relative to an iteration of global recovery. However, the relative overhead of the SeqNum method decreases as the size of the network increases.

Figure 4 shows the convergence time. The RPL convergence time is according to the global update period.

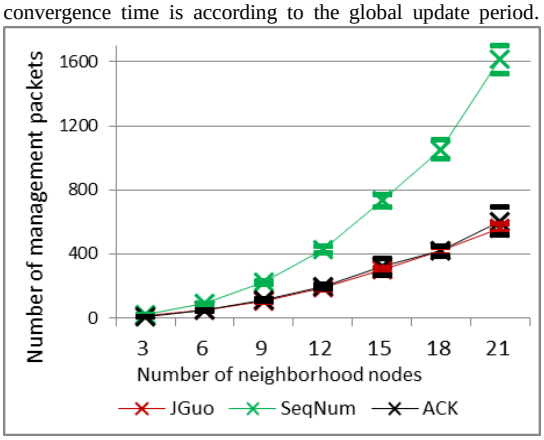

Fig. 3. Number of management packets vs. number of nodes in the network

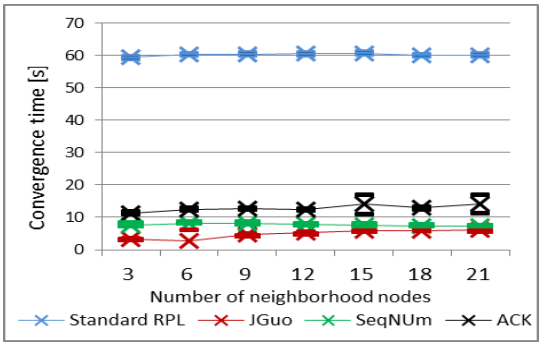

Fig. 4. Convergence time vs. number of negihbors.

Here the updating period is 50s. Increasing the updating frequency will reduce the delay time but increase the

overhead. Hence, delay strict network will have high overhead caused by the high updating frequency and much of these overhead is unnecessary if the network is stable.

As can be seen in Fig. 4, the average convergence time using the ACK method is up to about 2.8 times the J.Guo convergence time. This is due to the period when ordinary DIO is prohibited, as pointed out in the description of the ACK method. The average convergence time using SeqNum is up to about 1.7 times the J.Guo convergence time. The longer convergence time of SeqNum is due to the broadcasting performed by all neighboring nodes.

The graphs in Fig. 5 and associated Table 1 show the number of nodes that is unconnected after recovery. The standard RPL graph is included to demonstrate the number of nodes that are unconnected after the node dies and before any repair mechanism has been applied. In our topology, all nodes are connected following standard RPL global recovery. The difference between the standard RPL graph and the local recovery graphs indicates that all the local recovery mechanisms have a positive impact on network connectivity. However, 59% of the total simulated scenarios results in unconnected nodes when applying J.Guo. This is mainly due to sibling nodes being in poisoning state (they have pending REQ), prohibiting them from answering other requests. In addition, some collisions of request and replying messages may occur. All nodes are connected after local recovery in 99.5% of the simulated scenarios when the ACK method is used. The lack of connection in 0.5% of the scenarios is caused by colliding ACK messages. Finally, the SeqNum

TABLE 1

	Share of simulations resulting in unconnected nodes
Standard RPL	100 %
J.Guo	59,3 %
SeqNum	100 %
ACK	99,5 %

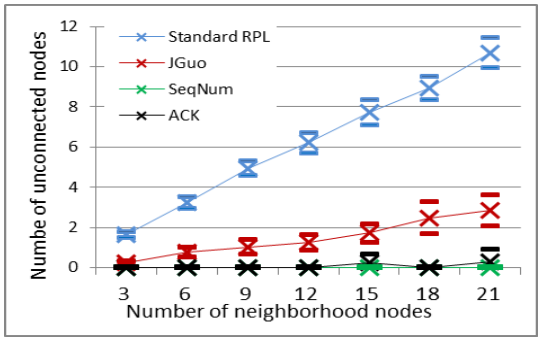

Fig.5. Number of unconnected nodes vs. number of nodes in the network. The table shows theshar of the simulations that results in unconnected nodes.

connect all nodes in the network.

Figure 6 shows the number of nodes that change rank when performing local vs. global recovery. Standard RPL is not included in the figure as it only executes global recovery. Only nodes that are still connected after local recovery are counted. The graphs show that applying J.Guo approach causes the larges fraction of nodes to change rank. These nodes may potentially disturb the network stability as they start transmitting ordinary DIO as explained in the J.Guo-section.

A minor amount of the nodes change rank when applying the SeqNum approach, but the number increases with increasing neighbors. This is due to the increased collisions of the management packets, in addition to the trickle timer issue explained in the SeqNum-section. The ACK approach rarely results in rank change.

VIII. CONCLUSION

Disconnections in WSN should be repaired as fast and energy efficient as possible to avoid data loss, network partitioning and excessive energy consumption. Nodes in loop-prone hierarchical positions require dedicated methods to make appropriate reconnections. According to our simulation, 15-40% of the disconnected nodes in a uniformly distributed

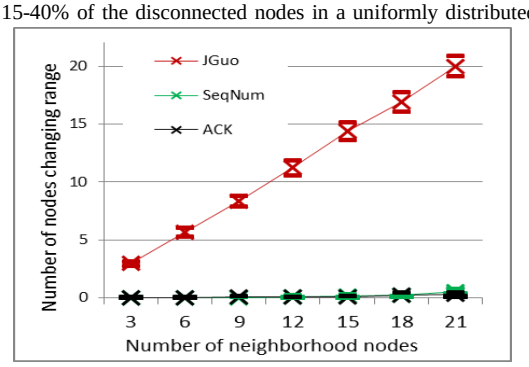

Fig. 6. Number of nodes changing rank when performing local vs. global recovery.

network are located in such loop-prone positions. Hence, these nodes require global or local recovery methods to get properly reconnected.

Network which exclusively relies on global recovery, as for example RPL, requires periodic updates performed often enough to fulfill network availability and recovery requirements. However, many of the updates may be unnecessary as the network topology is stable.

This article presents two new local recovery methods applicable for RPL. The first method we present is called SeqNum and uses local-sequence number to enable nodes to discover new loop-free paths. The second method uses reliable poisoning of the sub-DAG. Further, we discuss and compare the implications of these local recovery methods against global recovery. The discussion also comprises J.Guo's local recovery algorithm which is the basis for the Internet-Draft [7].

The ACK methods reconnect the network in 99.5% of the simulations performed. The ACK and J.Guo methods have overhead in the same order of magnitude. J.Guo has the lowest convergence time. However, J.Guo has lower reconnection success probability than the ACK method. Applying unreliable recovery methods that cannot guarantee connectivity may be of low value as energy is wasted while the network remains disconnected.

The simulation shows that the SeqNum method always succeeds in reconnecting disconnected nodes making it a reliable method. The reliability comes with a cost of high message overhead which increases rapidly with node density. High amount of overhead is unwanted as it causes high energy consumption. However, since the method is reliable it makes global recovery superfluous. Thus, compared to global recovery, the SeqNum method may give a reduced long term overhead and should be considered used in network with strict recovery delay combined with low disconnection probability or fluctuating disconnection probability.

A possible way forward is to reduce the overhead of the SeqNum method. This will be done by limiting the area where request are broadcasted, and make nodes outside the limited broadcast area unicast request toward the sink. Further, overhead reduction may be achieved by allowing lower rank nodes to reply the received requests.

REFERENCES

[1] T. Winter and et.al., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, 2012.

[2] J. Guo, C. Han, P. Orlik and J. Zhang, "Loop-Free Routing in Low-Power and Lossy Networks, SENSORCOMM," International Conference on Sensor Technologies and Applications, 2012.

[3] A. Díaz-Ramírez, F. N. Murrieta, J. A. Atempa and F. Bonino, "Non-intrusive Tracking of Patients With Dementia Using a Wireless Sensor Network," International Conference on Distributed Computing in Sensor Systems, pp. 460 - 465, 2013.

[4] C. Zato, S. Rodriguez, D. Tapia, J. Corchado and J. Bajo, "Virtual Organizations of agents for monitoring elderly and disabled people in geriatric residences," International Conference on Information Fusion, pp. 327 - 333, 2013.

[5] S. H. Lee, S. Lee, H. Song and H. S. Lee, "Wireless sensor network design for tactical military applications: Remote large-scale environments," Military Communications Conference, 2009. MILCOM, 2009.

[6] C. Zhu, C. Zheng, L. Shu and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," Journal of Network and Computer Applications, no. Volume 35, Issue 2, p. 619–632, 2012.

[7] J. Guo, P. Orlik and G. Bhatti, "Loop Free DODAG Local Repair," draft-guo-roll-loop-free-dodag-repair-01, 2013.

[8] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," SIGCOMM, ACM, pp. 234-244, 1994.

[9] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July 2003.

[10] J. Garcia-Luna-Aceves and H. Rangarajan, "A New Framework for Loop-Free On-Demand Routing Using Destination Sequence Numbers," International Conference on Mobile Ad-hoc and Sensor Systems, IEEE, pp.426-435, 2004.

[11] H. Rangarajan and J. Garcia-Luna-Aceves, "On-demand Loop-Free Routing in Ad Hoc Networks Using Source Sequence Numbers," 2005.

[12] J. Raju and J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-Free Multipath Routing," Eight International Conference on Computer Communication and Networks , IEEE, pp. 522-527, 1999.

[13] M. Elhadef, A. Boukerche and H. Elkadiki, "Diagnosing Mobile Ad-hoc Networks: Two Distributed Comparison-Based Self-Diagnosis Protocols," International workshop on Mobility management and wireless access, ACM, pp. 18- 27, 2006.

[14] M. Mosko and J. Garcia-Luna-Aceves, "Loop-Free Routing Using a Dense Label Set in Wireless Networks," International Conference on Distributed Computing Systems, IEEE, pp. 380 - 389, 2004.

[15] W. Xie, M. Goyal, H. Hosseini, J. Martocci, Y. Bashir, E. Baccelli and A. Durresi, "Routing Loops in DAG-based Low Power and Lossy Networks," International Conference on Advanced Information Networking and Applications, IEEE, pp. 888-895, 2010.

[16] "OMNET++ : http://www.omnetpp.org/. (Nov., 2013)".

[17] "Std 802.15.4™-2003. Chapter 6, Digital Object Identifier: 10.1109/IEEESTD.," 2003.

[18] Takagi, H. and Kleinrock, L. "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals," IEEE Transactions on communications, Vol. 32, No. 3, pp. 246-257. (1984).

Paper C:

An Analysis of the Need for Dedicated Recovery Methods and Their Applicability in Wireless Sensor Networks Running the Routing Protocol for Low-Power and Lossy Networks

Anne-Lena Kampen, Knut Øvsthus and Øivind Kure

Proceedings of the 8th International Conference on Sensor Technologies and Applications (SENSORCOMM' 2014), pp. 121-129, 2014, ISBN: 978-1-61208-374-2

An Analysis of the Need for Dedicated Recovery Methods and Their Applicability in Wireless Sensor Networks Running the Routing Protocol for Low-Power and Lossy Networks

Anne-Lena Kampen
Bergen University College, Bergen
ITEM NTNU, Trondheim
Norway
Anne-Lena.Kampen@hib.no
anneleka@stud.ntnu.no

Knut Øvsthus
Bergen University College, Bergen
Norway
knut.ovsthus@hib.no

Øivind Kure
NTNU, Trondheim
Norway
okure@item.ntnu.no

***Abstract*—Wireless Sensor Networks (WSN) functionality depends critically on the network connectivity. The connectivity is generally determined by the node density and the nodes' transmission range. However, the applied routing protocol decides the routing path topology. A failing node may disrupt the current path topology such that dedicated recovery methods are needed to ensure a loop-free reconnection of the disconnected nodes. In this article, we estimate the probability that disconnected nodes need dedicated recovery methods in networks where the nodes are randomly located and which use RPL as routing protocol. We further calculate the success rate and overhead cost for different RPL fitted recovery protocols to better judge where the different methods should be used.**

Keywords-WSN; Recovery; Disconnection; Energy; Looping.

I. INTRODUCTION

A common requirement for Wireless Sensor Networks (WSN) is marginal need for human support during operation. Hence, the networks should be able to autonomously handle common error conditions, such as loss of connectivity due to failing nodes [1]. Connectivity loss negatively affects the data throughput and may lead to network partitioning. Thus, nodes should be reconnected without any unnecessary delay. The reconnection process should further expend limited amount of energy to minimize its influence on network longevity.

Reconnection of nodes located such that several neighboring nodes are at the same routing distance from the sink as a failing next-hop node (parent) introduces insignificant delay and energy consumption. However, nodes located such that all neighbors are at a routing distance further away from the sink than a failing parent node cannot make an immediate reconnection. The reason is that the reconnection process may create routing loops if not controlled. Routing loops are created if the disconnected nodes choose their own directly, or indirectly, connected successors as new parent nodes. Dedicated global or local recovery methods are means to ensure against the formation of routing loops during the reconnection process.

Global recovery processes generally postpone reconnection of nodes in loop-prone topologies until the next global network update. Local recovery processes make nodes in the vicinity of a disconnection communicate routing information to enable fast, loop-free reconnections. The most suitable recovery method is decided by the network characteristics and the requirements of the running application.

Our contribution is twofold and relates to recovery in randomly deployed network running Routing Protocol for Low-Power and Lossy Networks (RPL) [2], which is one of the recommended protocols for WSNs. First, we present calculations and simulations to assess the need for dedicated recovery methods to reconnect disconnected nodes. The result can be used as a base to decide whether to introduce dedicated recovery management in applied networks. We further suggest one on-demand recovery method that combines the global and local approaches. The suggested method, along with two additional local recovery methods, is analyzed to better judge where the different recovery methods should be used.

The rest of the article is organized as follows: the related work is introduced in Section 2. The probability that dedicated recovery methods are needed to mend disconnection is presented in Section 3. An analysis of two local recovery methods are presented in Sections 4 and 5. The on-demand method is presented in Section 6. The methods are compared in Section 7. Section 8 comprises the conclusion.

II. RELATED WORK

Network connectivity calculations are presented in several papers, such as [3][4][5]. Zhu et al. [6] conclude that a

network with satisfying coverage is connected if the communication range is twice the coverage range. Topology controls methods to maintain k-connected networks are investigated and suggested in [7][8] and [9]. Kleinroch [10] discusses network connectivity based on cited works and presents the node degree needed to achieve network connectivity. However, the analysis performed in these papers focuses on connectivity without considering the applied routing protocol or reconnection of disconnected nodes. Our analysis is based on the functionality of the applied routing protocol.

Many of the presented recovery protocols suggest movable nodes to reconnect disconnected nodes [11][12]. Nodes are proactive moved to prohibit disconnections, or reactive moved to mend disconnections. However, we assume that the nodes location is static and the goal is to discover all alternative possible recovery paths.

All recovered paths are required to match the routing protocol's path construction method and avoid formation of routing loops. Global methods fulfil this requirement by making disconnected nodes in loop-prone positions postpone reconnection until the next global update. This approach is used in RPL [2].

The disconnected node initiates the recovery process in local recovery methods. The affected node signals its state to adjacent nodes. Depending on the recovery method used, the signaling may be relayed further on to reach nodes eligible to offer new loop-free paths to the disconnected node. Sequence numbers, as used in [13][14], is a common mean applied in such local recovery methods to discover new paths. Other local recovery processes include avoiding any duplicates in the address field during source routing [15] and caching alternative feasible successors paths in case the current route is broken [16]. The feasible successor paths are guaranteed loop free as they report a distance to the destination that is shorter than the current path from the source to the destination.

III. PROBABILITY OF DEDICATED RECOVERY

This section presents an analysis of the need for dedicated recovery methods to mend routing path disconnections. The routing protocol that is used as the basis for our analysis is RPL.

A. Short presentation of RPL

RPL is a soft state routing protocol that creates routes that are directed toward the sink. The overall topology of the routing entries creates Destination Oriented Directed Acyclic Graph (DODAG). A node's logical location in the routing graph is defined by the nodes rank and selected parent, which are two strongly interrelated properties of a node. A node calculates its own rank based on its selected parent rank and the metric-based cost-of-path between itself and its parent.

To prevent against routing loops are nodes running RPL prohibited from increasing their rank in between global DODAG updates. Performing a rank increase means that

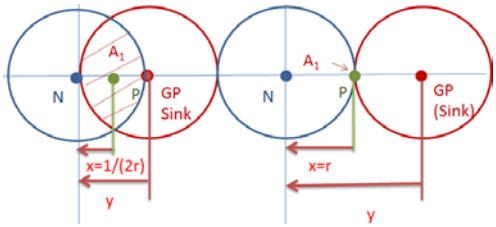

Figure 1.Node arrangement for a node at two-hop distance from the sink.

nodes make a logically move away from the sink in the routing graph, an action which may results in routing loops. Global DODAG updates are initiated by the sink by distributing updated Destination Information Option messages (DIO) that flow like a wave throughout the whole network.

B. Presentation of extreme points for probability calculations

To estimate the probability that a dedicated recovery method is needed is complex and depends on the relative location of all possible parents' next-hop node (grandparent). However, the highest and lowest probability limits may be calculated by studying the difference between dedicated recovery need for nodes located at the extreme points. The extreme location for the nodes is at the border of the routing graph. One of the extreme points is represented by the nodes located at a two-hop distance from the sink. The nodes next-to-the-leaf nodes are the highest rank nodes that may require dedicated recovery, and represent the second extreme point. All other nodes that may require dedicated recovery lie between these two borderline cases, so do their average probability.

1) Extreme location one : Two-hop distance node

Figure 1 illustrates a general node arrangement for nodes at two-hop distance from the sink. The blue dot labeled N and the blue circle is the node under consideration and its transmission distance, respectively. The red dot is the grandparent node, and its transmission range is defined by the red circle. The grandparent node is the sink as the node under consideration, N, is a node at two-hop distance from the sink. The parent node is represented by the green dot labeled P. All nodes choose parents that minimize their own rank. Thus, the node and its grandparent cannot communicate directly. Further, if the parent node dies, the node N needs to find a new parent node to maintain the path toward the sink.

According to the loop-avoiding rules of the RPL method, a node is never allowed to increase its rank unless a global update is performed. Hence, node N needs to maintain or improve its rank if the parent node in Figure 1 dies. The only way node N can improve its rank is to achieve a direct connection to the sink, which it cannot. Hence, node N needs to maintain its rank. It follows that it needs to get connected to a node that is directly connected to the sink, i.e., it must keep the sink as its grandparent node after recovery. Thus, the

alternative new parent node must reside in the overlapping area of the transmission circle of the grandparent and the transmission circle of the node N (area A in the figure).

2) Extreme location two: Node next-to-the-leaf node

Figure 2 is used as a reference to calculate the permitted-area A for a node next-to-the-leaf node. The multiple circles centered at the sink represent the location for nodes at a specific hop distance from the sink. The area between the red dot representing the sink, and the inner red circle, represents the location for the one-hop nodes. The area between the inner red circle and the second inner red circle represent the permitted-area for the two-hop nodes and so forth. As we are considering a node next-to-the-leaf node, we assume that the distance between the sink and the node under consideration is so far apart that the curvature of the sink's h-hop circle line cutting through the node N's circle approaches a straight line.

Figure 3 is a segment of Figure 2. The orange vertical lines in Figure 3 illustrate the sink's outer h-hop circle lines as straight lines based on the explanation above. The red shaded area named 'A' illustrate the permitted-area for a parent node of node N.

To find the probability that dedicated recovery is needed we derive the expression for the expected value of the probability that there exists more than one node inside the area A. If there is more than one node in are A, it means that there exists a recovery node after the current parent node dies. To find the wanted expectation we need an expression for the probability that there is another node in A, as well as an expression of the probability density function for location of node N.

C. Probability that there is a recovery node in area A

We assume a uniform node distribution, thus, the number of nodes in an area is given by the Poisson distribution. λ is defined to be the node density, which corresponds to the expected number of nodes in a circular shaped area with radius equal to the transmission range. All nodes have equal transmission range, r.

The probability that there is another node in A in Figures

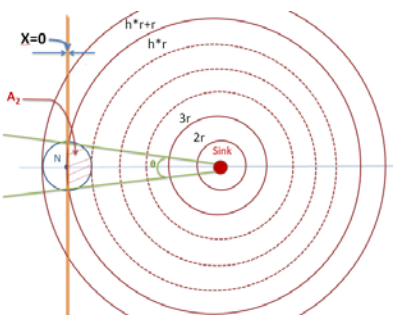

Figure 2. Disconnected node next-to-the-leaf node.

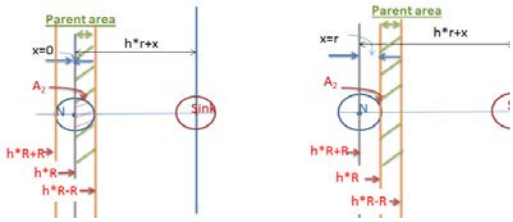

Figure 3. Node arrangement for a node far from the sink.

1 and 2 is given as Prob(more than 1 node in area A | given that there is at least 1 node in area A). Prob(1 or more ∩ at least 1)/ prob(at least 1) = Prob (2 or more)/Prob (at least 1):

$$p(more\ than\ one\ \ in\ A|least\ one\ in\ A) =$$

$$\frac{1 - P(1\ node\ in\ area\ A) - P(0\ nodes\ in\ area\ A)}{1 - P(0\ nodes\ in\ area\ A)} =$$

$$1 - \frac{\lambda A(x)}{e^{\lambda A(x)} - 1} \qquad (1)$$

We assume that node N is at a distance y from a grandparent node, as shown in Figure 1 for the two-hop node. y is in the range r, 2r. The area of A_1 is symmetric around y/2=x. Then the area A_1 as a function of x is:

$$A_1(x) = 2r^2\left(cos^{-1}\left(\frac{x}{r}\right) - \sqrt{1 - \left(\frac{x}{r}\right)^2} * \frac{x}{r}\right) \qquad (2)$$

According to the node next-to-the-leaf node, the permitted-area is one half of the area A_1 in (2), using y=x.

Notice that the area A_2 in Figure 3 is bigger than are A_1 in Figure 1 when the node N is in its closest position to the sink (left hand side of the figures). Hence, with node N in this position, the permitted-area for the recovering node next-to-the-leaf node is bigger than the permitted-area for the two-hop recovering node.

D. Node next-to-the-leaf-node calculations

We use Figure 2 as reference to calculate the probability density function for the node N location. The sector θ in Figure 2 defines the sector where a recovery parent node N may be located. The cumulative distribution function for the node N's location and the probability density function of node N's location are respectively given by:

$$F_1(y) = \frac{\frac{\pi((h*r)+y)^2}{\theta} - \frac{\pi(h*r)^2}{\theta}}{\frac{\pi(h*r+r)^2}{\theta} - \frac{\pi(h*r)^2}{\theta}} = \frac{y(2hr+y)}{(1+2h)r^2} \qquad (3)$$

$$f_1(y) = \frac{2(hr+y)}{(1+2h)r^2} \qquad (4)$$

Using the presented equations, we can derive the expression for the expected value of the probability that there exists a recovery node in area A_2 for the node next-to-the-leaf node. The expression is found by combining (1) with (4) and (2). The expected value of P(there exist a recovery node inside area A_2) is:

$$E[P(y)] =$$

$$\int_r^{2r} \{(P(more\ than\ 1\ inA_2(y)|least\ 1\ inA_2(y))f(y)\}dy$$

$$= \int_r^{2r} \left\{\left(1 - \frac{\lambda A_2(y)}{e^{\lambda A_2(y)}-1}\right) * \frac{2(hr+y)}{(1+2h)r^2}\right\} dy \qquad (5)$$

E. Two-hop node calculations

We will now derive the expectation of the probability that there exists a recovery node in area A_1 for the two-hop node, ref. Figure 1. First we need the expression for the probability density function for the location of node N. This is found using Figure 4. According to Figure 4 are the expression for the cumulative distribution and probability density function of the node N given by:

$$F_2(y) = \frac{\frac{\pi(y)^2}{\theta} - \frac{\pi r^2}{\theta}}{\frac{\pi(2r)^2}{\theta} - \frac{\pi r^2}{\theta}} = \frac{y^2 - r^2}{3r^2} \qquad (6)$$

$$f_2(y) = \frac{2y}{3r^2} \qquad (7)$$

Combining (1), (2) and (7), gives the following expected value of the probability for the two-hop node:

$$E[P(y)] = \int_r^{2r} \left\{\left(1 - \frac{\lambda A_1(y)}{e^{\lambda A_1(y)}-1}\right)\left(\frac{2y}{3r^2}\right)\right\} dy \qquad (8)$$

To summarize; the expression for the expected value of the probability that there exist recovery nodes for a node next-to-the-leaf node given by (5), and the corresponding expression for a two-hop node given by (8). In other words, this is the probability that dedicated recovery methods are *not* needed to mend routing path disconnections. The calculations performed are based on numerically calculations of the equations.

F. Simulations - dedicated recovery

Simulations are conducted in Java to validate the calculated results for the expected value of the probability that dedicated recovery methods are not needed to mend routing path disconnections.

The simulation for the two-hop node is initialized by placing a node N in a fixed position. The next node is randomly paced in the donut shaped area between r and 2*r form the fixed node. The second node becomes the fixed node's grandparent (the sink). A varying number of nodes are subsequently randomly distributed with average density λ inside the simulation area.

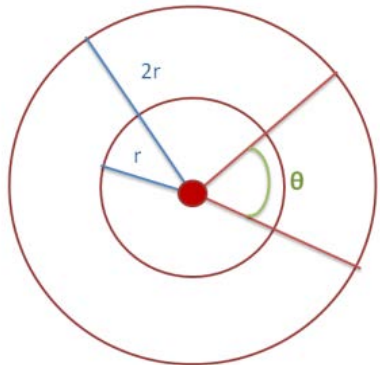

Figure 4. The sector where a possible node N may be located.

The sought probability for the two-hop node is estimated based on the percentage of the simulation runs resulting in two or more nodes located inside the overlapping area defined by the node's transmission range and the grandparent's transmission range. Dedicated recovery is needed if the number of nodes in the overlapping area is less than two. The reason is that if there is only one node in the overlapping area, it is definitely the parent node and there are no nodes left in the area when it dies. Simulation runs resulting in zero nodes inside the area is discarded. The simulated result is averaged over 1000 runs for each node density.

According to the node next-to-the-leaf node the simulations is performed by placing two nodes at a distance h*r + x apart, in the same manner as for the two-hop node. 0<x<r. The two nodes represent the node under consideration, N, and the sink. The number of nodes located both inside node N's transmission range and inside a radius of h*r from the sink are counted. The investigated probability is further performed following the same procedure as when calculating the two-hop node probability.

G. Results – dedicated recovery

In this section, we present and discuss the simulated and calculated results of the expected value of the probability that dedicated recovery methods are *not* needed to mend routing paths. The curves in Figure 5 show the expected value of the probability for the extreme points, i.e., the lowest and highest average probability values. The dashed red curve represents the simulated probability of the nodes at two-hop distance from the sink, and the blue curve shows the calculated probability for the two-hop node. The red and the dashed green curve show respectively the calculated and simulated results for the node next-to-the-leaf node. The curvature of the simulation results conform to the curvature of the calculated results validating each other.

The curves in Figure 5 show that the disconnected nodes next-to-the-leaf nodes have lower need for dedicated recovery than the two-hop nodes.

The difference between the curves in Figure 5 is caused by the unequal characteristics of the two extreme points in the routing graph topology. Both the probability density function for the node N's location and the permitted-area are different in the two extreme points. The probability density function for the location of a leaf node N approaches a uniform distribution as the number of hop gets high. This is illustrated in Figure 6. The reason for the uniform distribution is the straightening of the h*r curvature and the related small difference between the circumference of the h*r and h*r+r circle when h is high. This is easily seen looking at Figure 3. On the contrary, the probability density for the location of the two-hop node N is increasing toward the outer circumference. The reason is the increased circumference which increases available deployment area for the node N. This can be observed in Figure 4. Figure 7 shows how the permitted-area varies with distance between the node N and the sink node. The figure shows that the area decreases with increased distance, and also illustrates the slightly bigger permitted-area of the node next-to-the-leaf nodes. Combining the information given in Figures 6 and 7, shows that the location probability of the two-hop node favors the smallest permitted-area size, while the node next-to-the-leaf node gives equal priority to all permitted-area sizes. A smaller area means that the probability that it contain more than one node is lower. Thus, the probability that it contains a candidate recovery parent node is lower.

As expected, and illustrated by the graphs in Figure 5, is the need for a dedicated recovery method decreasing with

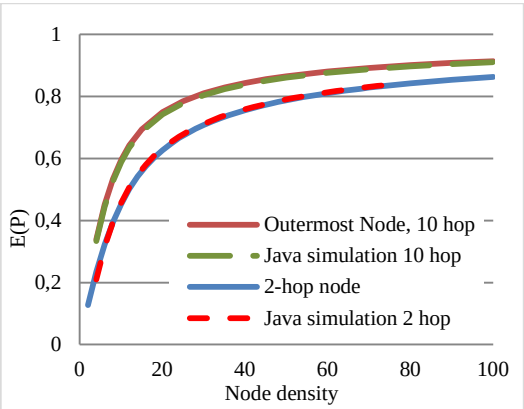

Figure 5. The expected value of the probability that dedicated recovery methods are not needed to mend

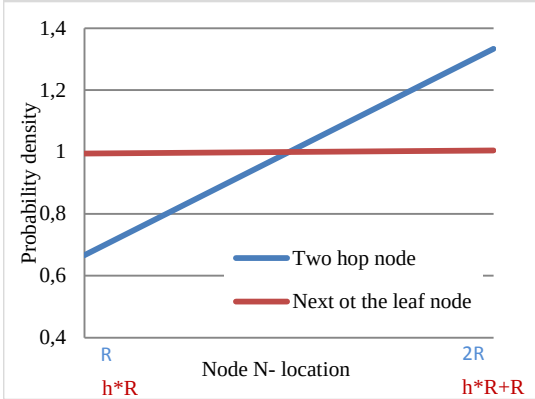

Figure 6. Probability density function of node N's

increased node density. The reason is simply that the probability that more than one node is located inside a defined area increases with node density. However, the probability never reaches 1 although the node density gets high. The reason is due to the explanation given related to Figures 6 and 7: the permitted-area for the recovery nodes is very small for some of the locations of node N. Hence, there is always a probability that some nodes do not have available recovery nodes.

The graphs in Figure 5 show that if all the nodes in a network are required to stay connected some kind of special repair method is needed. According to Takagi and Kleinrock [10], eight is the magic number of neighbors regarding network throughput, and four neighbors are needed to maintain a connected network. Hence, we may assume that an average well design network has a node density between 8 and 20. We define the node density as the number of nodes inside a circular area with radius equal to the transmission range, thus is equals the number of neighbors plus one. The graphs show that the probability that dedicated recovery is not needed is between 40% and 75% at node densities between 8 and 20. Hence, between 25% and 60% of the disconnected nodes needs a dedicated recovery method to get properly reconnected.

IV. Analysis of Guo et al.'s method

This section presents an analysis of a local recovery method suggested by Guo et al. [17]. The method forces intermediate nodes on potential recovery paths to adjust their rank to make the path feasible for a disconnected node.

Their method [17] is activated and runs as follows. A poisoned node, which is a node that needs dedicated recovery to reconnect, initiates the recovery process by broadcasting a request. The request is further relayed to the receivers' parent nodes. The process lasts until the requests reach a node with better rank than the requesting node. Receivers with better rank than the requesting node generate a reply and forward it toward the requesting node using the same path as the

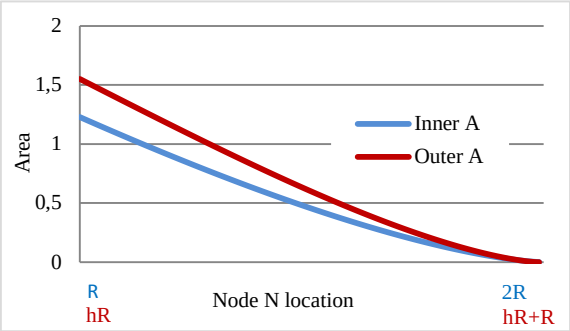

Figure 7. Permitted-area for recovery node.

associated request. The nodes along the path adjust their rank such that a new, valid path for the requesting node is made.

However, the method is not able to find a new valid path for all kind of topologies. The reason is twofold. Requests received form a parent node are silently discarded. Hence, paths pointing through child nodes are never found. In addition, a race condition occurs when siblings of a dying parent node simultaneous enter poisoning state. Nodes with pending requests silently discard received requests. The result may be that paths pointing toward sibling nodes remain undiscovered.

A. Simulation of Guo et al.'s method

Based on the layout in Figures 8 and 9, we simulate the probability that a poisoned node running Guo et al.'s method discovers a new valid path. The red dot is the sink node. The area between the sink and the inner red circle represents the localization of one-hop (rank one) nodes, the two-hop (rank two) nodes are located between the red circles, and so forth. The blue node N represents a node that is poisoned and need dedicated recovery to get reconnected to the DODAG. The blue circle represent node N's transmission range. The figures illustrate the two scenarios that make the [17] method local recovery succeed.

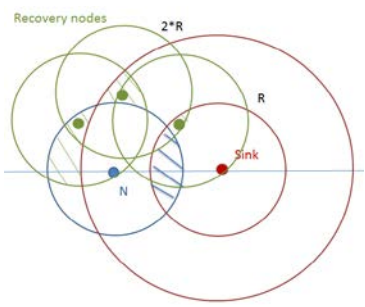

Figure 8.Recovery path goes through higher level node.

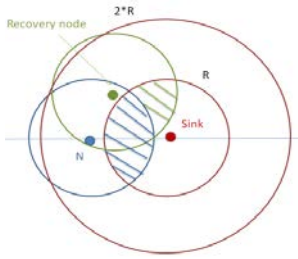

Figure 9.Recovery path goes through sibling node.

Figure 9 illustrates the scenario that a reachable node with equal rank as the poisoned node, has a parent node outside the poisoned node's transmission range. Expressed according to the figure, it means that there exists an equal-rank green node in the overlapping area made of the blue circle and the donut shaped area made of the red circles. Further, this equal-rank-node has a parent inside the green shaded area.

The other scenario that makes [17] succeed is if a lower-level node of N (node lying outside the outer red circle) have a path toward the sink that does not include N, or N's parent. According to Figure 8, it means that the node N has a neighbor in the leftmost green shaded area. This neighbor has a parent in the upper green shaded area, which further has a parent in the rightmost green shaded are.

The simulation is implemented in Java. A varying number of nodes are randomly deployed inside a circle shaped area with radius that is three times the transmission range. All the nodes are supposed to have equal transmission range. 5000 runs with different node densities, and node N locations, are performed. The numbers of runs which satisfy one or both of the scenarios discussed above, and indicated in Figures 8 and 9, are counted. This number is normalized by the number of runs where recovery is needed to reconnect N, i.e., the number of runs where only one node reside inside the blue shaded area.

B. Results on Guo et al.'s method

Figure 10 shows the simulated probability that a poisoned node gets reconnected after performing Guo et al.'s local recovery procedure. The x-axis shows the node density, λ. As expected is the success probability increasing rapidly with node density. When the node density is 4 the probability is about 40%. When the node density approaches 20, the probability approaches 100%. Thus, the approach of [17] works best in high density networks. The probability that a dedicated recovery method is needed to reconnect disconnected nodes is highest at low node densities; Figure 5. Hence, the lowest probability of solving the problem is in the scenarios where the problem is most likely to occur.

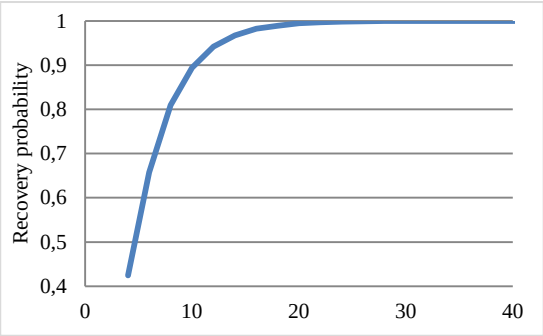

Figure 10. Probability that Guo et al.'s method succeed

V. ANALYSIS OF THE ACK LOCAL RECOVERY METHOD

This section presents a local recovery method that is based on reliable poisoning of successors (sub-DAG) nodes. We call this method the ACK-method. Reliability is achieved by letting nodes be aware of their children, and make all children acknowledge reception of poisoned information transmitted by the poisoned node. Information about children is achieved by making all nodes inform about their parents in regular transmitted DIO messages.

Receiving ACK form all children enables the poisoned node to increase rank to reconnect to the DODAG. No loop is created because sub-DAG nodes with no alternative recovery parent inform about their poisoned state in the transmitted ACK messages.

The ACK method will mend disconnections as long as the poisoned node receives ACK messages from all its children. Hence, the probability of success using this method depends on probability of successful reception of transmitted packets. We name the probability of successful transmission P_{rec}. A poisoned message is retransmitted once if the

poisoned node does not receive ACK from all children. Thus, recovery will not succeed if the two poisoned messages are lost, or if the ACK message is lost.

Thus, assuming one child gives the following success probability of the ACK method: P(ACK succeed one child)=p(First Poisoning messages succeed)*P(ACK succeed)+p(First Poisoning messages do not succeed) *p(Second Poisoning messages succeed)*P(ACK succeed):

$$P(ACK\ succeed\ one\ child) =$$

$$P_{rec} * P_{rec} + [1 - P_{rec}]P_{rec}P_{rec} \quad (9)$$

Assuming that either all or none of the child nodes receive the poisoning message, the expression for the ACK success probability for c child becomes:

P(ACK succeed for c child)= p(First Poisoning messages succeed)*P(ACK succeed)^c+p(First Poisoning messages do not succeed) *p(Second Poisoning messages succeed)*P(ACK succeed)^c:

$$P(ACK\ succeed\ c\ child) =$$

$$P_{rec} * {P_{rec}}^c + [1 - P_{rec}]P_{rec}{P_{rec}}^c \quad (10)$$

A. Results ACK method

The probability that the ACK method succeeds is shown in Figure 11. The x-axis represents the probability that a transmitted packet is received. The blue graph illustrates the success probability for a disconnected node with one child, and the red graph shows the success probability for a disconnected node with five child nodes. As expected is the success probability increasing with increased probability of receiving transmitted messages and with reduced number of child nodes.

VI. ON-DEMAND METHOD

In this section, we present our proposed combination of local and global recovery that may be used to guarantee recovery for all node densities while keeping the network energy consumption as low as possible. We call this method the on-demand method.

The method functions as follows. A node entering the poisoning state broadcasts an increase-sequence-number request. The request is broadcasted throughout the whole network, which means that it will eventually reach the sink if there exists a path between the poisoned node and the sink. When the sink receives the request it initiates the global recovery algorithm.

The message overhead, hence the energy cost, of running one iteration of the on-demand method is about twice the cost of running one iteration of periodic global update. The reason is that the request is broadcasted throughout the whole network in a manner similar to DIO message during global network update.

Running the update only when nodes are poisoned means that the network wide broadcast is run only when needed, and no periodic global network update is in fact ever needed.

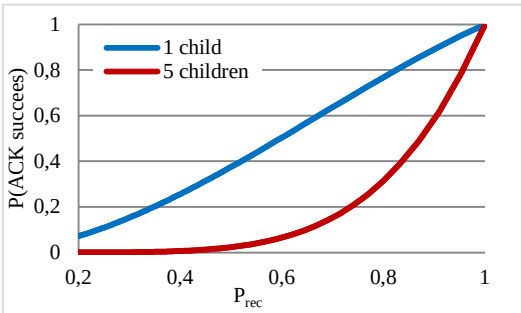

Figure 11. Probability that ACK succeed vs. Prec.

The recovery time using the on-demand method decrease compared to the periodic approach, as the global update is run immediately after the request reach the sink.

VII. ANALYSIS OF RECOVERY OVERHEAD COST

We perform calculations to estimate the overhead cost difference between Guo et al.'s, ACK, on-demand, and periodic recovery methods. The overhead is calculated as the total number of transmitted and received management messages during the recovery process. The overhead cost is proportional to the network energy consumed, which should be as small as possible to limit the recovery process' impact on the network lifetime.

The Guo et al.'s method overhead relates to the transmission of requests and replies. We assume a uniformly distributed network where the average number of neighbors is n. The fraction of neighbors forwarding the request is α, and the fraction of neighbors that replies the request is β. Thus, the number of nodes transmitting the request is (1+ αn). The digit 1 in the expression refers to the poisoning node initiating the request transmission. Each transmitted request is, on the average, received by n neighboring nodes. Hence, the total request cost is (1+ αn)*n. Further, we assume that the request and replies are relayed once. The reply is answered by βn nodes and relayed once by βn nodes. Each transmission is received by n nodes. Hence, the reply cost is (2βn)*n, and the total overhead cost becomes:

$$Guo\ et\ al.'s_{overhead} = 2 * \beta n^2 + (1 + \alpha n) * n \qquad (11)$$

The ACK overhead cost relates to the poisoning message and ACK message transmission. The poisoned node transmits a poisoning message, which is received by all neighboring node giving a total cost of 1+n. Further, we assume that the fraction of neighbors that are child of the poisoning node is Δ. Hence, the ACK messages is transmitted by Δn nodes and all messages are received by n nodes, giving a cost of Δn*n. In addition, all nodes in the neighborhood transmit a DIO concluding the recovery process. The DIO is received by all neighbors giving a cost of n*n. Hence, the total cost of the ACK recovery process is:

$$ACK_{overhead} = (1 + \Delta) * n^2 + n + 1 \qquad (12)$$

The on-demand overhead relates to the total number of nodes in the network, N, transmitting requests and DIO messages. All transmitted messages are received by the average number of neighbors.

$$(On - Demand)_{overhead} = 2 * Nn \qquad (13)$$

The overhead according to one run of global recovery relates to all nodes N transmitting DIO messages, which are received by the average number of neighbors.

$$OneRunGlobalRecovery_{overhead} = N * n \qquad (14)$$

A. Results comparing methods

Figure 12 shows the overhead for the different methods using Δ=0.4 (share of neighbors being child of poisoned node), α=0.6 (share of neighbors relaying request) and β=0.2 (share of neighbors replying request). The value of Δ is chosen looking at the right-hand side of Figure 1: We assume that approximately all nodes located inside an area about the same size as the red shaded area are children of a node N. The rest of the neighboring nodes relay the requests received, hence the value of α = 1 - Δ. β is chosen assuming that only a small fraction of nodes receiving the relayed request are able to answer. These values clearly changes according to the network topology. However, the mutual relation between the parameters will generally remain unchanged. Hence, the information given by the figure is valuable. The total number of nodes in the network is 100.

There is a big difference between the local recovery approach methods' overhead and the periodic update, as shown in Figure 12. However, as the local recovery methods cannot guarantee reconnection they require periodic global update to coexist to guarantee full network connectivity.

The figure shows a substantial overhead cost difference between on-demand method and periodic update. However, the great advantage of using the on-demand method is that the method is only trigged by a disconnection. Thus, the overhead cost will be lower than the periodic update method in network with low disconnection probability.

The significance of our findings is the statistical analysis of the need for dedicated recovery presented in Section 3, in addition to the overhead cost for the recovery methods presented in this Section. The statistical analysis showed that dedicated recovery is needed especially in low density networks. In addition, nodes in the vicinity of the sink are most vulnerable and require dedicated recovery. These nodes are critical for sustaining network connectivity. Our overhead cost findings show that cost analysis should be performed as

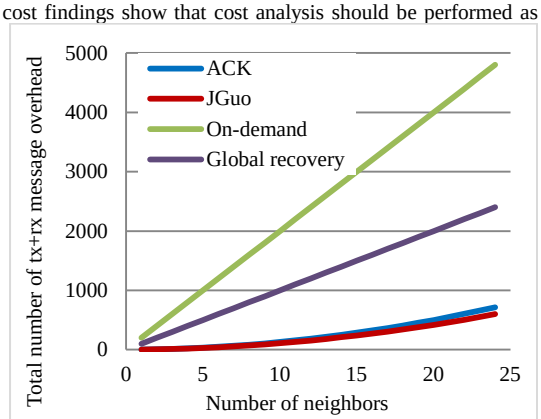

Figure 12. TX+RX message overhead. Number of nodes is 100.

part of real networks' deployment methodology to select an appropriate recovery method. The selected method should either be the on-demand method, or adapting of the periodic global update frequency, as these methods are reliable.

Our results can be used to perform overhead cost calculations for network design. The overhead associated with the on-demand method is calculated combining (13) with the expected number of nodes that need dedicated recovery during a time span. The expected number of nodes is found combining information about the total number of nodes, the nodes failure probabilities, and the probability for disconnected nodes' recovery need, presented in Figure 5.

The overhead cost of adapting the global update frequency is calculated combining (14) with the recovery delay requirement. The delay requirement decides the network update frequency.

VIII. CONCLUSION

Disconnections in WSNs need to be resolved to sustain total network availability and avoid destructive data loss. Whether disconnections needs dedicated methods to regain connectivity depends on the topology in the vicinity of the disconnection.

In this article, we calculated and simulated the probability that dedicated recovery methods are needed to reconnect disconnected nodes in randomly deployed networks. The findings are that dedicated methods are needed in 25% to 60% of the cases when a node is disconnected. These findings demonstrate the significance of including dedicated recovery methods as a part of the network management in real scenarios where the network's availability is crucial. If dedicated recovery is not included, the periodic global update frequency should be adjusted according to the networks' recovery delay requirements. In addition, the findings demonstrate that increased node density may be used as deployment methodology to improve connectivity stability in critical areas of a network.

The findings further show that disconnected nodes close to the sink most often need dedicated recovery. These nodes are critical to sustain network connectivity. Hence, it may be wise to adapt the recovery method according to routing graph location.

The failure frequency increases with network size assuming equal failure probability for the nodes. Hence, the periodic update frequency has to increase with network size in network without any dedicated recovery method. However, increased update frequency increases the nodes' energy consumption causing reduced network lifetime.

Introducing a dedicated recovery method may reduce the load caused by periodically updates. In this article, we calculate the overhead and success rate for two local recovery methods, and one suggested global on-demand recovery method. The two local recovery methods have lowest overhead, but they cannot guarantee reconnection success. The global on-demand method is reliable as reconnections are established if possible. But, it has high overhead. However, using an unreliable recovery method that cannot guarantee connectivity requires a simultaneous periodic global update mechanism to assure the total network connectivity, while a reliable recovery method makes periodic global updates superfluous.

Thus, networks requiring reliable network connectivity should either include an on-demand recovery method, or adjust the global network update frequency. The on-demand method may greatly reduce the long-term network energy consumption. Overhead cost analysis presented in this article may be used for real scenarios to choose between the two methods.

REFERENCES

[1] V. C. Gungor and G. P. Hancke, "Industrial Wireless Sensor Networks: Challenges,," IEEE Transactions On Industrial Electronics, Oct. 2009, pp. 4258-4265.

[2] T. Winter, et. al., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, Mar. 2012.

[3] W. Jia, Y. Fu, and J. Wang, "Analysis of Connectivity for Sensor Networks Using Geometrical Probability," Springer, Lecture Notes in Computer Science, Aug. 2004, pp. 601-611.

[4] Y. Wang, B. M. Kelly, and X. Li, "On the Network Connectivity of Wireless Sensor Networks following a Random and Non-uniform Distribution," IEEE, Sixth International Workshop on Selected Topics in Mobile and Wireless Computing, Oct. 2013, pp. 69-74.

[5] X. Xingt, G. Wangt, J. Wu, and J. Li, "Square Region-Based Coverage and Connectivity Probability Model in Wireless Sensor Networks," IEEE, International Conference on Collaborate Computing, Nov. 2009, pp. 1-8.

[6] C. Zhu, C. Zheng, L. Shu, and G. Han, "A Survey on Coverage and Connectivity Issues in Wireless Sensor Networks," ELSEVIER, Journal of Network and Computer Applications, Mar. 2012, pp. 619-632.

[7] I. Saha, et al., "Distributed Fault-Tolerant Topology Control in Static and Mobile Wireless Sensor Networks", IEEE, 2nd International Conference on Communication System Software and Middelware, Jan. 2007, pp. 1-8.

[8] N. Li and J. Hou, "FLSS: A Fault Tolerant Topology Control Algorithm for Wireless Networks", ACM, Proceedings of the 10th annual international conference on Mobile computing and networking,MobiCom, 2004, pp. 275-286.

[9] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks using Transmit Power Adjustment," IEEE, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, Mar. 2000, pp. 404-413.

[10] H. Takagi and L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals," IEEE Transactions on communications, vol. 32, no. 3, Mar. 1984, pp. 246-257.

[11] K. Vaidya and M. Younis, "Efficient Failure Recovery in Wireless Sensor Networks through Active Spare Designation," IEEE, International Conference on Distributed Computing in Sensor Systems Workshops, June. 2010, pp. 1-6.

[12] M. F. Younis, S. Lee, and A. A. Abbasi, "A Localized Algorithm for Restoring Internode Connectivity in Networks of Moveable Sensors," IEEE, Transactions On Computers, Dec. 2010, pp. 1669-1682.

[13] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July. 2003.

[14] J. Garcia-Luna-Aceves and H. Rangarajan, "A New Framework for Loop-Free On-Demand Routing Using Destination Sequence Numbers," IEEE, International Conference on Mobile Ad-hoc and Sensor Systems, Oct. 2004, pp. 426 - 435.

[15] D. Jhonson, U. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," RFC 4728, Feb. 2007.

[16] J. Raju and J. J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-Free Multipath Routing,", IEEE Eight International Conference on Computer Communication and Networks, Oct. 1999, pp. 522-527.

[17] J. Guo, C. Han, P. Orlik, and J. Zhang, "Loop-Free Routing in Low-Power and Lossy Networks," IARIA, International Conference on Sensor Technologies and Applications, Aug. 2012, pp. 59-66.

Paper D:
Energy balancing algorithms in Wireless Sensor Networks

Anne-Lena Kampen, Knut Øvsthus and Øivind Kure

Proceedings of the 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), Volume 5, pp. 1223-1231, 2015, Electronic ISBN: 978-8-3608-1065-1, DOI: 10.15439/2015F67

Energy Balancing Algorithms in Wireless Sensor Networks

Anne-Lena Kampen
Bergen University College, Bergen
ITEM NTNU, Trondheim
Norway
Anne-Lena.Kampen@hib.no
anneleka@stud.ntnu.no

Knut Øvsthus
Bergen University College, Bergen
Norway
knut.ovsthus@hib.no

Øivind Kure
NTNU, Trondheim
Norway
okure@item.ntnu.no

Abstract—**The energy consumption in Wireless Sensor Networks, WSN, need to be balanced in order to avoid early depletion of nodes. In this paper we use a common context to analyze a broad range of the energy balancing algorithms suggested in literature. In addition we suggest three new algorithms to complete the range. Altogether, nine different balancing techniques are analyzed. We focuses on networks running the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) routing protocol. Our simple change in RPL's parent selection procedure can give a significant balancing effect without any increase in management cost. However, the best balancing algorithm is when the nodes exchange residual-energy information to ensure forwarding through the highest residual-energy next-hop node. The increased information exchange implies increased management cost due to the amount of information transmitted and added computational load.**

I. Introduction

Wireless sensor networks (WSN) generally consist of wireless nodes with a collective objective of gathering measured information at the sink [1]. The monitored area may be large compared to the nodes' transmission range. Hence, the information needs to be relayed to reach the sink. The topology of the relaying paths may create imbalance in the traffic share, and therefore the energy consumption, between the nodes. Energy imbalance results in lifespan variation between the nodes. Observations of real networks in [2] and [3] show that some nodes relay a substantial portion of the traffic, thus they become hot-spot-nodes having a high energy consumption rate.

Depleted or dead nodes make the gathered data incomplete and, more important they may cause network partitioning. Applying energy balancing routing algorithms levels the traffic load, hence lifetime, between the nodes. The ideal situation is long living WSNs where all nodes have equal lifetime. However, this ideal situation not feasible due to the increased traffic density toward the sink in multihop networks. The goal in multihop networks is instead to balance the energy consumption between nodes at equal hop distance from the sink. Network management will be simplified if the nodes at each rank have similar lifetime. Balancing algorithms are the topic of this paper.

Our contribution is threefold. First we present a methodical review of a broad range of energy balancing algorithms. The algorithms range from approaches requiring simple changes of the applied routing algorithms, to approaches that require complex add-ons. Second we use a common context to compare these algorithms. Third, we suggest three new balancing algorithms to complete the collection of balancing algorithms found in the literature. The tree suggested algorithms are random selection of preferred-parent, conserving of Single Point of Failure (SPOF) parent and energy balancing based on eavesdropping.

To get a good estimate of the energy pattern in the network we use the nodes residual-energy. The residual-energy gives the true picture of the energy variation that appears between the nodes. In addition, residual-energy is directly related to the nodes lifetime.

To evaluate the impact of different energy balancing techniques, we use the routing protocol suggested by Internet Engineering Task Force (IETF) for use in WSN, IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [4]. RPL creates routing entries in the nodes which forms an overall destination oriented directed acyclic graph (DODAG) rooted at the sink. The graph is created by broadcasting of DODAG Information Object (DIO) messages. The sink initiates the transmission, and the messages are further broadcasted throughout the whole network. The DIO includes the senders' rank information. The rank indicates a node's distance to the sink. The rank increases as the distance to the sink increases. The sink is at rank 0 and the sink's one-hop neighbor defines the rank-one nodes and so forth. Each node caches a parent-list containing all neighbors that report a rank equal to the lowest rank heard. A preferred-parent is selected among the nodes in the parent-list. The preferred-parent is used as the current next-hop node on the path toward the sink. To maintain the DODAG, the nodes transmit DIO messages periodically at intervals decided by a trickle timer [5].

The rest of the paper is structured as follows. In Section 2 we present related work, in Section 3 introduces the different energy balancing approaches to be analyzed, the simulation is presented in Section 4, and Section 5 comprises the conclusion.

II. Related work

Several energy balancing approaches are suggested in the literature. Energy balancing based on selecting the most energy

optimal path is suggested in [6] - [13]. In all but the two latter of these algorithms are energy information exchanged through DIO messages. Applying DIO messages to exchange energy information means the energy balancing depends on the number of data packet transmitted per transmitted DIO message. Thus, increased energy balance is paid by increased DIO transmission frequency, which means increased average energy consumption. Further, the trickle timer [5] decides the DIO emission frequency such that the emission frequency decreases exponentially with time in converged networks. Hence, the balancing effect of the algorithms discussed in the next paragraph will decline with time.

The object function (OF) for RPL suggested in [6] defines the path cost as the energy level of the node on the path with lowest residual-energy. The node that advertises the highest path cost is preferred as selected parent, and the lower energy nodes are spared. The authors of [8] suggest that the node with the highest remaining energy among the nodes with the lowest expected transmission count (ETX) is chosen as the preferred-parent in network running RPL. Both ETX and node energy is used to select between parent nodes of equal hop-count in [9]. The residual-energy is included as a denominator in the additive distance metric in [10]. Using it as in the denominator makes the cost of a node increase toward infinity as energy approaches zero. Hence, the paths including low energy nodes is avoided due to their high cost. A routing metric that calculate the expected lifetime of the nodes is defined in [11]. The expected lifetime is calculated as the ratio of the node's residual-energy over the total energy spent to transmit data. The paths including the most constrained nodes are avoided by defining the path weight as the minimum expected lifetime along the path. An approach similar to RPL is used to create paths for networks with multiple sinks in [7]. Several equal-rank nodes are cached as potential parent based primary on hop-count metric, secondary the nodes energy metric and third on the highest link-quality-indication. Algorithms where the highest residual-energy path is selected or the lowest residual-energy paths are avoided are part of our analysis.

Energy information is exchange through the ACK packet in the approach presented in [13]. The network run RPL, and the nodes perform a weighted selection to choose among its available next-hop nodes. The selection is weighted between distributing the traffic through the lowest delay path and distributing the traffic to nodes with higher remaining energy. In the routing protocol suggested in [12], the energy information is both piggybacked on data packets and included in the ACK packets. Hop-count is used as the metric to generate parent-list. Data packets are transmitted to the highest energy member of the parent-list. If there is no parent node available, the packet is transmitted to the sibling node with the highest amount of energy. Our analysis includes an algorithm where the ACK message exchange the energy information.

Energy consumption can be balanced by continuously spreading the transmitted data over multiple paths, and such methods are also part of our analysis. Approaches using multiple paths are suggested in [14] [15]. RPL is used as the routing protocol in [14] where the forwarding load is weighted between the members of the parent-list. The weighting is based on the members' residual-energy. The transmission range dynamically adjusted to maintain k parents. Energy information is exchanged between the nodes through ACK and DIO packet. In addition is also hello packets mentioned as possible information carriers. The approach in [15] enables multipath data forwarding through energy-sufficient paths, as opposed to minimum-energy-cost paths. They propose a routing algorithm which makes a hierarchical routing graph similar to RPL. The nodes forward packets through alternate paths to extend the network lifetime. The conditions of the paths are monitored by the sink which re-initiate path search if the number of working paths gets lower than two. Multiple paths are also discussed in the surveys presented in [16] [17]. Survey [16] cites an algorithm presented in [18], which takes both the energy level and hop distance into account to allocate different data rates to multiple disjoint paths. The sink decides the rate of the different paths and assign messages are sent form the sink to the source nodes to inform about the path rates. The top-down survey paper [17] cites an interesting improved cost function used to balance the energy consumption among the nodes [19]. The improved cost calculation algorithm makes the cost increase rapidly with decrease in the nodes remaining energy. Hence, traffic is directed away from hot-spot nodes. The approach requires that the nodes cache several states for each neighbor and that energy information is exchanged periodically. The survey paper [17] also discusses energy balancing by using a few relay nodes with enhanced capabilities. In addition they discuss use of mobile sinks. These algorithms increase the start-up management cost of the networks, and increases the network cost.

Clustering is among the energy efficient algorithm discussed the survey presented in [20] and suggested to improve energy utilization in [26]. The basic idea of energy efficient clustering is to perform energy efficient rotation of the clusterhead assignment and let the clusterhead perform energy efficient management of the local cluster traffic. Clustering is not part of our analysis as it is not very well fitted for RPL running network.

Balancing the energy consumption by making the nodes alternate between direct transmissions to the sink and using multi-hop transmissions is suggested in [21]. The protocol is used as an extension to RPL in [22] which presents a smart/green test-bed of nodes spanning across several smart offices. The findings of [22] shows that the protocol suggested in [21] balances the network energy consumption compared to classic RPL. However, it is more energy expensive giving an overall increased energy dissipation. This algorithm is only considered for one-hop networks, while we are considering multihop networks.

III. BALANCING NETWORK ENERGY CONSUMPTION

In this section we present the different energy balancing algorithms that are analyzed. Our hypothesis is that introducing small changes in the parent selection procedure improves the WSN energy balance, while substantial enhancements come at a cost of increased management complexity and information exchange between nodes. Further, efficient energy balance is achieved when focusing on reducing the load of the hot-spot nodes.

The following text lists nine algorithms. The tree new algorithms that we suggest are A: Randomize parent selection, D: Weighting round-robin based on SPOF-parent energy level and H: weighting round-robin based on eavesdropping.

A. Randomize parent selection

As a first approach to enhance the energy balance in WSN, we suggest a simple change in the preferred-parent selection algorithm. The aim of the suggested algorithm is to reduce the probability of creating hot-spot nodes. The probability is reduced by preventing that several child nodes select the same preferred-parent if other potential parents exist. According to the RPL algorithm, all nodes cache a parent-list containing all candidate parent nodes. A preferred-parent are selected among the parent-list nodes, using a specific parameter as tiebreaker. Hence, nodes with globally good tiebreaker value will be selected by all potential child nodes and may therefore become hot-spot nodes.

Our suggested algorithm creates a small change in the preferred-parent selection procedure to reduce the probability of creating such hot-spot nodes. The nodes randomly select a preferred-parent among the nodes in the parent-list instead of using a preordain tiebreaker parameter value. Hence, the probability that several potential child nodes select the same node as preferred-parent is reduced. The forwarding load is therefore more balanced. The weakness of the algorithm is that it can give energy consumption imbalance if selected parents are located such that they represent single paths for other nodes.

B. Round-robin through multiple paths

Selecting a single preferred-parent may overload some potential parents while leaving some potential parents unused. Thus, our analysis comprises an approach where this imbalance is alleviated by making the nodes transmit data packets to all nodes in their parent-list in a round-robin fashion. The approach shares the forwarding load equally between all members of the nodes' parent-lists. The main weakness of the approach is the load imbalance that is created between nodes with different number of child nodes.

C. Weighted round-robin based on energy information in DIO messages

To level the energy imbalance that may appear using the round-robin approach we implement algorithms in which the nodes exchange energy information during DIO transmission. The information is used to perform a weighted-fair-sharing between the parent nodes. Thus, the nodes share the traffic load among the nodes in the parent-list according to their relative residual-energy level. The energy-balancing effect of the weighted algorithms depends on the freshness of the energy information cached for the nodes in the parent-list. Hence, increased DIO exchange frequency means improved energy balance. However, increased DIO exchange frequency increases the energy consumption in the network. Thus, there is a tradeoff between energy balance and average energy consumption.

Weighted round-robin ensures that the energy depletion rate of the low energy parents is reduced, hence the energy balance is improved. However, the algorithm preserves the existing energy imbalance relationship between parent nodes.

D. Weighting round-robin based on SPOF-parent energy level

The goal of our single point of failure (SPOF) algorithm is to prevent early depletion of SPOF nodes. We define SPOF nodes as nodes that are part of one or more parent-lists containing only one member. In other words, a child that has a SPOF parent is disconnected from the routing graph if the parent node dies. Child of SPOF parent forwards all data through the SPOF parent. Even when the SPOF parent has a very low energy level, the child has no other option than continue forwarding through the SPOF parent. Hence, depleting of the SPOF node is continued.

To reduce the depletion rate of the SPOF nodes we suggest to direct traffic originating from higher rank nodes away from the SPOF nodes. In order to do so, we let nodes with a SPOF parent advertise the energy level that is the lowest of its own and its SPOF-parent's residual-energy level. Thus, traffic is directed away from the paths including the SPOF node.

Directing the traffic away from the SPOF nodes may come at an expense of other low energy nodes on the same rank as the SPOF node. However, child with SPOF parent continue to transmit their own generated data to their SPOF parent, while other nodes only get a weighted amount of traffic from their respective child nodes.

The DIO is used to exchange energy information.

E. Weighted round-robin based on prediction parents energy consumption

The energy information gained through received DIO can be used to predict the energy consumption pattern in between DIO updates. To test such energy prediction algorithms, we implement an algorithm that estimate parents current energy level based on statistics of former energy consumption. The algorithm is as follows. The residual-energy a node advertises in consecutive transmitted DIOs is cached at the receiving nodes. The timespan between the consecutive DIO is further used to estimate the depletion rate of the transmitting node. The current energy level is estimated using the individual parents' energy drain rate and last advertised energy level. The estimated energy level is used to perform weighted-fair-sharing between the parent nodes.

F. Weighted round-robin while avoid lowest energy parent

In order to focus on the hot-spot nodes we suggest a partly weighted algorithm. The data is weighted between the parent nodes. However, no data is transmitted to the parent with the lowest residual-energy. Hence, the load on the hot-spot nodes is reduced. This algorithm requires that the nodes exchange residual-energy information through the DIO message.

G. Use the highest energy parent node

In the multiple path approaches, although weighted, each parent receives data for forwarding from their child nodes. This applies even if the residual-energy level of the parent is low. Hence, if all nodes transmit approximately equal amount of traffic, the nodes that are members of several paths are depleted faster than other nodes.

The depletion pace of low energy nodes is reduced in approaches where the lowest energy parent is avoided such as the approach presented in subsection III.F. However, nodes forces parents with second lowest energy level to forward traffic. Hence, the lowest energy nodes alternates their states with next lowest energy nodes.

A simple solution is to use only the highest energy parent node as the next-hop node. This algorithm is similar to the algorithm used in [8]. We implemented this algorithm and used DIO to exchange energy information.

H. Weighting round-robin based on eavesdropping

Utilizing information conveyed in DIOs may give an incomplete view of the current energy levels of the nodes in the parent-list. The reason is that traffic imbalance, and associated energy consumption imbalance that occurs between DIO transmissions are not taken into account.

In order to predict parents' energy consumption between DIO updates, we suggest that nodes eavesdrop on the traffic transmitted in the area. The algorithm operates as follows. Nodes read the source and destination address information in the eavesdropped traffic. The address is matched against the content in the parent-list of the eavesdropping nodes. When a match is found, the energy level of the associated parent-list entry is reduced according to the eavesdropped information. The energy level of the nodes in the parent-list is then used to perform weighted-fair-sharing.

Eavesdropping does not significantly influence on the nodes energy consumption. The reason is that overheard packets destination address has to be read anyway to determine the intended receiver of the packet. The energy consumption due to overhearing is not taken into account when comparing the different balancing techniques. The reason is that the extent of overhearing energy consumption is mainly decided by the energy saving approach chosen at the MAC layer, while we are concentrating on the routing layer algorithms' impact on energy consumption.

The eavesdropped traffic may not give a complete overview of the parent nodes traffic load. For instance, child nodes of a common parent may not receive each other's packets due to hidden node. Thus, the calculations of the energy consumption of the parent-list nodes may be imprecise.

I. Weighting round-robin based on energy information conveyed in ACK packets

Lastly, we implement an algorithm in which information about the nodes' energy variation in between DIO transmissions is exchanged through ACK packets. ACK packets are sent as a response of received data packet. Hence, the nodes achieve a complete overview of the diverse energy levels of the nodes in their parent-list as each parent relays a packet.

However, the energy information of parents with low residual-energy is less current. The reason is that low energy nodes seldom forward data as they have low weight. In addition, nodes that rarely transmit data can have stale energy information for the nodes in their parent-lists. This may give temporary screwed forwarding load among parent nodes. However, the energy levels are continuously balanced as energy information is updated, smoothing the discrepancy over time.

Weighted-fair-sharing is performed based on the energy information.

IV. SIMULATIONS

We perform simulations to evaluate the different discussed algorithms. The energy consumption in WSN increases toward the sink as the inner nodes are obligated to relay traffic for outer nodes. Thus, we mainly present simulations segregated on the node's rank. Applying an energy balancing algorithm will not change the average energy consumption for the nodes at the different rank since the total number of packet transmitted through each rank is unchanged.

We concentrate on transmitting and receiving energy consumption. Overhearing energy is not taken into account. The reason for omitting the overhearing energy consumption is that we concentrate on energy balancing at the network layer, and overhearing energy consumption is strongly dependent on the energy saving approach applied at the MAC layer. Overhearing may give a small variation in the average energy due to the chosen path. However, the difference between the energy consumed due to overhearing become negligible because the algorithms are compared at given average node density.

As discussed above, the average energy consumption at each rank is consistent regardless of applied balancing algorithm. However, an efficient energy balancing algorithm makes the residual-energy of the most depleted node approach the average residual-energy at the given rank. Hence, we present the average and minimum residual-energy after each node has generated 100 data packets.

Energy information is used to tune the traffic load between parent nodes in some of the evaluated algorithms. In networks running these algorithms, each node caches residual-energy information of its parents. The accuracy of the cached information depends on the update interval. For the algorithms that exchange energy information through DIO messages, the accuracy is improved by reducing the number data packets exchanged per DIO transmitted. The residual-energy information accuracy approaches the accuracy of the ACK algorithms if the DIO emission frequency approaches the data rate. However, increased DIO exchange frequency increases average energy consumption. Further, the DIO exchange frequency is decided by the trickle timer such that the exchange frequency is strongly reduced in converged stable networks. Thus, the energy balance is declining over time when the balance depends on DIO exchanged information.

In our initial simulations, 100 data packets create sufficient network traffic to discriminate the balancing effect of the different category of balancing algorithms. However, each node transmit a total of 16 DIO messages during the simulation runs, hence the number of data packets transmitted for each DIO transmission is low. Thus, to improve the basis of comparison we present additional simulation results for the weighting algorithms. Only two DIO messages are transmitted during the simulation run in these additional simulations, and the number of transmitted data packets is increased to 300. This gives a more fair comparison between the algorithms relying on DIO to exchange energy information, relative to the

algorithms that use additional means to exchange energy information.

We evaluate the different energy balancing approaches by performing simulations in OMNET++ [23], using the MiXiM module for wireless communication. The nodes' energy consumption is calculated based on traffic load. Based on the observations and references in [24], we assume that receiving and transmission of data packets consume the same amount of energy. The different types of packets have different packet sizes. Management packets are assumed to be half the size of data packets, while ACK packets are one tenth of the size of the data packets. These relative values are chosen based on an assumption that data packets never need the maximum allowed packet sizes as they are mainly limited to carry only measured data, while management packets only carries strictly needed information. Maximum data frame sizes and ACK frame sizes information extracted from 802.15.4 datasheet [25].

The nodes are randomly distributed in an 800m times 800m area. The nodes transmission range is 141m. The number of nodes is varied such that the node density changes from 8 to 20. The node density is defined as the number on nodes inside a circle with radius equal to the nodes transmission range. Every simulation point presented represents the average value of 30 simulation runs with different seeds for random deployment of nodes.

As discussed above, the average energy consumption is equal for each rank over all energy balancing algorithms. However, the residual- energy values of the most depleted nodes change with the applied balancing algorithm. The most optimal energy balancing algorithm is the algorithm in which the residual-energy of the most depleted nodes converges to the average value. Therefore, we compare the algorithms with respect to their ability to make the nodes with lowest residual-energy approach the average value of their associated rank. Figure 1 shows simulation results that demonstrate to what extent the different algorithms make the lowest and average values converge. All the algorithms discussed in Section 3 are presented in the figure.

The nodes' average residual-energy, as well as the residual-energy of the most depleted nodes are presented in Figure 1. This is the residual-energy level of the nodes after each node has generated and transmitted 100 data packets. In addition to data, management traffic has been exchanged to make the network converge. Further has periodic DIO updates been transmitted. The circled shaped markers with the corresponding lines show the average residual-energy values. The energy levels for the most depleted nodes are shown as short horizontal markers. In order to clarify the information displayed in the figure, the best and the worst of the residual-energy levels are displayed with solid curves through their associated markers. Hence, the orange curve cut through the markers representing the highest residual-energy level and light green curve cut through the markers representing the lowest residual-energy. The colors of the markers and lines indicate the corresponding energy-balancing algorithm as defined in Table 1. To prevent that the important information gets hidden in an overloaded display, the 95% confidence interval is not shown in the figures. However, the 95% confidence interval is always within 7% of the average values. Simulations performed for node densities of $\lambda = 10$ and $\lambda = 15$ show the same trends as shown for $\lambda = 8$ and $\lambda=20$ in Figure 1.

TABLE 1.

Color-codes used in the figures to define the different energy balancing algorithms

A Randomize parent
B Round-robin
C Weighted - DIO information
D Weighted - SPOF parent
E Weighted - predicted energy
F Weighted - avoid lowest
G Use highest energy node
H Weighted - eavesdropping
I Weighted -ACK information
Native RPL

The simulation results displayed in Figure 1 shows that native RPL creates energy imbalanced networks. The native RPL simulation results are represented by the light green marks and the light green curve. Native RPL gives the overall lowest residual-energy for all ranks and all node densities. The reason is that a fixed parent is used throughout the whole simulation scenario, further is lowest node-id used as a tiebreaker when choosing between potential parent nodes. The latter means that several nodes choose the same preferred-parent node.

The difference between the lowest residual-energy node and the average value increases toward the sink. The reason is the increased traffic density. A given imbalance in traffic share causes an increase in the real traffic load difference as the total traffic increases.

The energy imbalance increases rapidly with node density for approaches where the parent node is fixed. This is observed in Figure 1 where the light green native RPL line rapidly moves away from the average line as the node density increases. The reason is the increased number of neighbors. Increased number of neighbors increases the number of child nodes for the fixed parent.

Based on the discussion above it is clear that some kind of energy balancing techniques should be added to networks running native RPL. Our suggested random preferred-parent selection algorithm presented in subsection III.A, demands a minor change in the RPL implementation. Nevertheless, the residual-energy of the most depleted node is reduce by over 10% compared to native RPL for high density networks. This is seen in the Figure 1 comparing the light green native RPL marks with the dark green marks.

However, the most efficient energy balancing algorithm is the one presented in subsection III.G, in which the parent with the highest residual-energy is selected as the next-hop node. The algorithm is represented by the orange marks in the Figure 1. Using highest energy parent increases the residual-energy of the most depleted node more than 25% compared to the native RPL. The merit of the algorithm is that the residual-energy nodes are avoided. This result corresponds to the results presented in [8] and [6] where ETX is used as a metric to populate the parent-list. However, using the parent with the highest residual-energy means that energy information has to be exchanged between the nodes. Randomizing parent selection, presented in subsection III.A, does not add any overhead.

The weighted share algorithms increase the residual-energy of the most depleted nodes with 15 to 20% compared to native RPL in high density networks. The weighted share algorithms are presented in subsection III.C-F and III.H-I. The improved performance complies with the studies performed in in [14] [15]. However, the poorer performance of these algorithms compared to the highest energy parent algorithm, III.G, is due to the fact that the lowest energy parent is still used, although rarely.

In subsection III.F we suggest the improved weighting algorithm, where the most depleted node is avoided while the traffic is weighted between the other parents. The improved algorithm is represented by the red marks in Figure 1, and shows that the residual-energy of the most depleted node is always distinguishable higher than the general weighting algorithms.

The round-robin approach, represented by the light blue marks, contributes less to balance the energy than the weighting algorithms. The reason is that parent with low residual-energy are loaded with the same amount of traffic as the other parents.

The simulated weighted-fair-sharing algorithms exchange energy information through DIO messages as well as through the algorithm-specific energy information exchange technique. Thus, the energy information update intervals between the different algorithms converge if the DIO exchange frequency

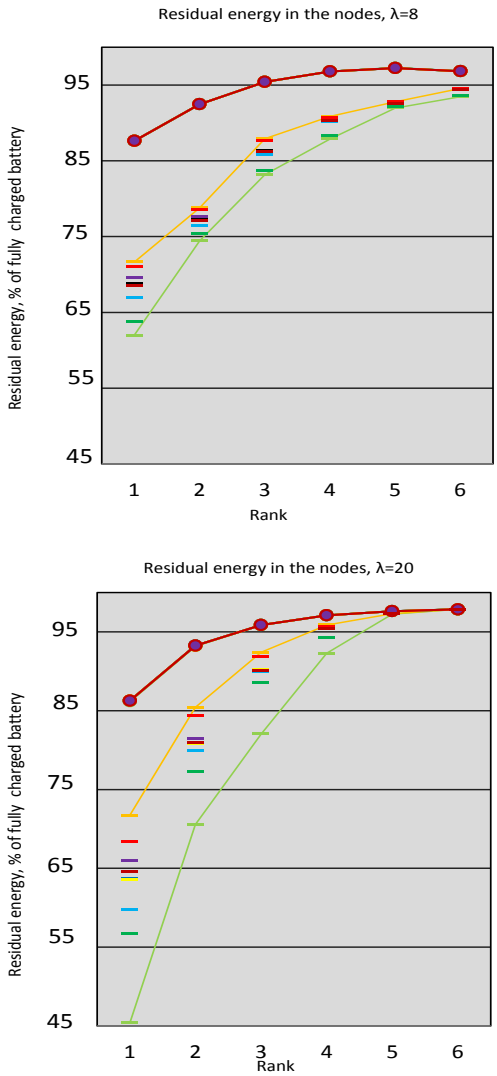

Fig 1. Residual-energy in the nodes after each node has generated 100 data packets.

is high compared to the packet exchange frequency. This phenomenon is demonstrated in Figure 1 as it is difficult to discern between the simulation results for the algorithms that use weighting as balancing technique. To better illustrate the difference between the weighting algorithms we performed simulations where each node generated 300 data packets. The DIO exchange is limited such that each node only generates two DIOs during the whole simulation. The simulation result is shown in Figure 2.

As expected, when the DIO exchange frequency is reduced, the ACK method has an improved balancing performance relative to weighting based on DIO information. The ACK method is presented in subsection III.I and weighting based on

DIO information is presented in subsection III.C. The improved balancing performance is seen in Figure 2 where the blue marker of the ACK method is closer to the average values than the yellow DIO information markers. In Figure 1, the blue ACK markers are actually hidden by the yellow markers. Hence, the ACK and the DIO information performed equally well when the number of data packets per DIO packet is low.

The eavesdrop-algorithm described in subsection III.H and represented with the black marks and line in the Figure 2, has the worst balancing capabilities. This applies especially for the one-hop nodes. The reason is increased traffic density in these areas of the network. High traffic density means that nodes often become hidden terminals preventing them from eavesdropping neighbors' traffic.

Our SPOF-algorithm described in subsection III.D and presented with purple line and marks in Figure 2, gives the best balancing effect. The reason is that higher-rank nodes are encouraged to choose paths that does not include the nodes that act as SPOF. However, SPOF nodes are not completely unloaded from forwarding data since child nodes have to forward all traffic through their SPOF parent nodes.

At higher rank nodes the SPOF-algorithm gives equal or marginally less balance compared to all other weighted-fair-sharing algorithms. However, the most efficient energy-balancing algorithm is the algorithm that focuses on energy balance among the lowest rank nodes, although this may give reduced balancing effect at the higher-rank nodes. The reason is that the lowest rank nodes always consumes the highest average amount of energy.

Increased number of child nodes enlarges energy imbalance between nodes, in particular for algorithms that uses fixed preferred-parent nodes. The number of child nodes increases with increased node density and reduced rank. This is demonstrated in Figure 3 which shows the number of child nodes versus rank for two different node densities, λ=8 and λ=20. This figure supports the findings in Figure 1 related to

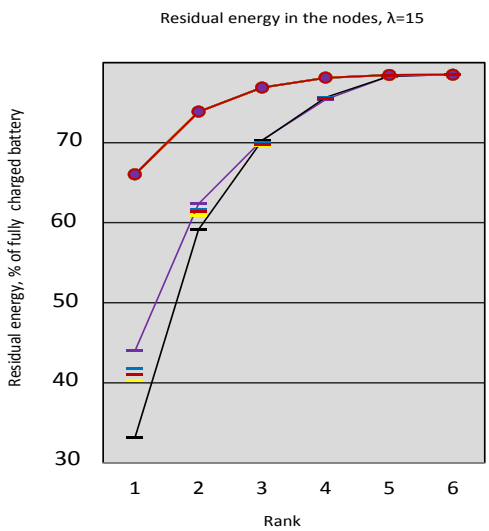

Fig 2. Residual-energy in the nodes after each node has transmitted 300 data packets

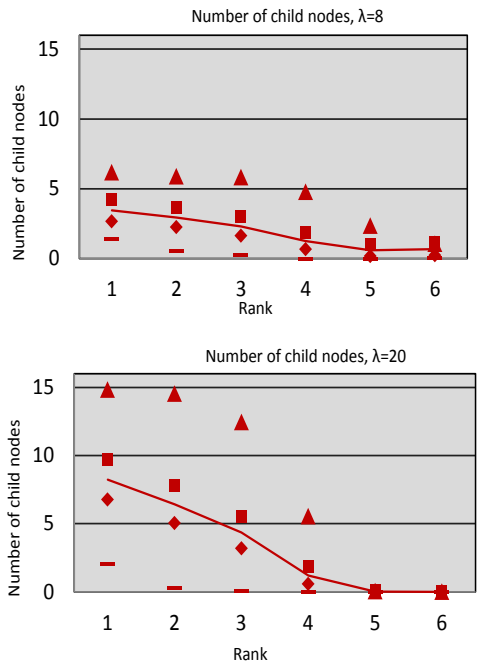

Fig 3. Number of child nodes versus rank for different node densities

the rapidly increase in energy imbalance for increased node density. The circled shaped markers in Figure 3 with the corresponding lines show the average values. The square shaped and the diamond shaped markers shows the 95% confidence interval. In addition, the highest values, averaged over all different-seeds simulations are shown as triangular markers. The lowest values, averaged over all different-seeds simulations are shown as short horizontal lines.

V. CONCLUSION

Simulations presented in this paper show that the energy imbalance is substantial in network running the native RPL routing protocol. Thus, some kind of energy balancing algorithm should be used to prevent premature node depletion.

A total of nine energy balancing algorithms applicable for RPL running networks are analyzed in this paper. Six of the algorithms are based on various approaches suggested in literature. In addition, we suggest three new energy balancing approaches to complement the selection of algorithms. A common context is used to simulate and compare the performance of all the algorithms.

The simplest of our suggested approach is just a tiny adjustment of RPL's parent selection algorithm. Instead of using a preordain tiebreaker parameter, the preferred-parent (next-hop node) is randomly selected among the nodes in the parent-list. The adjustment gives a significant balancing effect. Especially in high density networks, where the residual-energy of the most depleted node is increased more than 10% compared to running native RPL.

The second and the third suggested algorithm use the nodes' residual-energy to weight-balance the transmitted traffic between all available parent nodes. In the second algorithm, a node with a single point of failure (SPOF) parent advertises a residual-energy level equal to the lowest of its own and its parent energy level. The algorithm requires residual-energy information to be exchanged between the nodes during RPL management packet exchange.

In the third algorithm, the nodes eavesdrop on the traffic in their vicinity to estimate neighboring nodes residual-energy level. Increased traffic density degrades the eavesdropping algorithm in the proximity of the sink.

Simulations shows that the SPOF algorithm performs best of all the weighting algorithms. Compared to native RPL is the SPOF algorithm increasing the residual-energy of the most depleted node with over 20%.

However, the best energy balancing is achieved when nodes choose the preferred-parent as the member of the parents list that has the highest residual-energy level. Simulations of the algorithm show that, compared to native RPL, the residual-energy of the most depleted nodes increases by 25%. The merit of this algorithm is that the lowest residual-energy paths are always avoided. On the contrary, weighting the traffic between all potential parent nodes means that also the lowest energy nodes are used, although rarely. However, to select the node with highest residual-energy, the nodes must exchange energy information. Randomly selecting the preferred-parent requires no extra information exchange.

Although the energy consumption is balanced, it is always the nodes closest to the sink that consumed the highest amount of energy. However, balancing the energy consumption of equal rank nodes can give reduced network management cost.

REFERENCES

[1] C. Buratti, A. Conti, D. Dardari and R. Verdone, «An Overview onWireless Sensor Networks Technology and Evolution,» Sensors, Volume 9 Issue 9, pp. 6869-6896, 2009. http://dx.doi.org/ 10.3390/s90906869

[2] Y. Liu, Y. He, M. Li, J. Wang, K. Liu and X. Li, «Does Wireless Sensor Network Scale? Measurement Study on GreenOrbs,» IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, pp. 1983-1993, 2013. http://dx.doi.org/ 10.1109/TPDS.2012.216

[3] K. Heurtefeux, H. Menouar and N. AbuAli, «Experimental Evaluation of a Routing Protocol for WSNs: RPL robustness under study,» IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp. 491-498, 2013. http://dx.doi.org/ 10.1109/WiMOB.2013.6673404

[4] T. W. e. al., «RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,» Request for Comments: 6550, 2012.

[5] P. L. e.al, «The Trickle Algorithm,» Request for Comments: 6206, 2011.

[6] P. O. Kamgueu, E. Nataf, T. D. Ndie and O. Festor, «Energy-based routing metric for RPL,» RR-8208, pp. 1-14, 2013.

[7] G. Xu and G. Lu, «Multipath Routing for DAG-based WSN with Mobile Sinks,» Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering ICCSEE, pp. 1678-1682, 2013.

[8] C. Abreu, M. Ricardo and P.M.Mendes, «Energy-aware routing for biomedical wireless sensor networks,» Journal of Network and Computer Applications, p. 270–278, 2014. http://dx.doi.org/ 10.1016/j.jnca.2013.09.015

[9] L. Chang, T. Lee, S. Chen and C. Liao, «Energy-Efficient Oriented Routing Algorithm in Wireless Sensor Networks,» International Conference on Systems, Man, and Cybernetics (SMC), pp. 3813 - 3818, 2013. http://dx.doi.org/ 10.1109/SMC.2013.651

[10] K. S. Shivaprakasha and M. Kulkarni, «Energy Efficient Shortest Path Routing Protocol for Wireless Sensor Networks,» International Conference on Computational Intelligence and Communication Networks CICN, pp. 333 - 337, 2011. http://dx.doi.org/ 10.1109/CICN.2011.70

[11] O.Iova, F. Theoleyre and T. Noel, «Improving the network lifetime with energy-balancing routing: Application to RPL,» Wireless and Mobile Networking Conference (WMNC), pp. 1 - 8 , 2014. http://dx.doi.org/ 10.1109/WMNC.2014.6878864

[12] S. Chiang, C. Huang and K. C. Chang, «A Minimum Hop Routing Protocol for Home Security Systems Using Wireless Sensor Networks,» Transactions on Consumer Electronics, pp. 1483 - 1489, 2007. http://dx.doi.org/ 10.1109/TCE.2007.4429241

[13] P. T. A. Quang and D. Kim, «Enhancing Real-Time Delivery of Gradient Routing for Industrial Wireless Sensor Networks,» TRANSACTIONS ON INDUSTRIAL INFORMATICS, pp. 61-68, 2012. http://dx.doi.org/ 10.1109/TII.2011.2174249

[14] M. N. Moghadam, H. Taheri and M. Karrari, «Minimum cost load balanced multipath routing protocol for low power and lossy networks,» Wireless Networks,Volume 20, Issue 8, pp. 2469-2479, 2014. http://dx.doi.org/ 10.1007/s11276-014-0753-7

[15] R. Vidhyapriya and P. T. Vanathi, «Energy Efficient Adaptive Multipath Routing forWireless Sensor Networks,» IAENG International Journal of Computer Science, pp. 56-64, 2007.

[16] K. Sha, J. Gehlot and R. Greve, «Multipath Routing Techniques in Wireless Sensor Networks: A Survey,» Wireless Personal Communications, pp. 807-829, 2013. http://dx.doi.org/ 10.1007/s11277-012-0723-2

[17] T. Rault, A. Bouabdallah and Y. Challal, «Energy efficiency in wireless sensor networks: A top-down survey,» Computer Networks, vol. Volume 67, p. 104–122, 2014. http://dx.doi.org/ 10.1016/j.comnet.2014.03.027

[18] Y. M. Lu and V. W. S. Wong, «An energy-efficient multipath routing protocol for wireless sensor networks,» International Journal of Communication Systems, p. 747–766, 2007. doi : 10.1002/dac.843

[19] A. Liu, J. Ren, X. Li, Z. Chen and X.Shen, «Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks,» Computer Networks , p. 1951–1967, 2012. http://dx.doi.org/10.1016/j.comnet.2012.01.023

[20] N. A. Pantazis, S. A. Nikolidakis and D. D. Vergados, «Energy-Efficient Routing Protocols in Wireless Sensor

Networks: A Survey,» Communications Surveys & Tutorials, pp. 551-591, 2013. http://dx.doi.org/ 10.1109/SURV.2012.062612.00084

[21] C. Efthymiou, S. Nikoletseas and J. Rolim, «Energy balanced data propagation in wireless sensor networks,» Journal Wireless Networks , p. 691–707, 2006. http://dx.doi.org/ 10.1007/s11276-006-6529-y

[22] C. M. Angelopoulos, G. Filios, S. Nikoletseas, D. Patroumpa, T. P. Raptis and K. Veroutis, «A Holistic IPv6 Test-Bed for Smart, Green Buildings,» International Conference on Communications (ICC), pp. 6050-6054, 2013. http://dx.doi.org/ 10.1109/ICC.2013.6655569

[23] «OMNET++ : http://www.omnetpp.org/. (Mars 2015)».

[24] A.-L. Kampen, K. Øvsthus, L. Landmark and Ø. Kure, «Energy Reduction in Wireless Sensor Networks by Switching Nodes to Sleep During Packet Forwarding,» The Sixth International Conference on Sensor Technologies and Applications, SENSORCOMM, pp. 189-195, 2012.

[25] C. R. F. H. IEEE P802.15 Working Group, «Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Specifications for Low-Rate Wireless,» IEEE Std 802.15.4™-2006, 2006.

[26] K. K. Gagneja and K. E. Nygard «A QoS based Heuristics for Clustering inTwo-Tier Sensor Networks,» Proceedings of the Federated Conference on Computer Science and Information Systems, FedCSIS, pp. 779–784, 2012.

Paper E:

Modelling the Optimal Link Length in Wireless Sensor Networks for Two Different Media Access Protocols

Knut Øvsthus, Espen Nilsen, Anne-Lena Kampen and Øivind Kure

Sensors & Transducers Volume 185. Issue 2, pp. 21-28, 2015, ISSN: 2306-8515

Sensors & Transducers

http://www.sensorsportal.com

Modelling the Optimal Link length in Wireless Sensor Networks for two Different Media Access Protocols

Knut Øvsthus[1], Espen Nilsen[1] Anne-Lena Kampen[1,2] and Øivind Kure[2]
[1] Dep. Electrical Engineering, Bergen University College, Bergen, Norway
[2] ITEM, NTNU, Trondheim, Norway

Tel.: +47 55 58 75 00, fax: +47 55 58 77 89

E-mail: kovs@hib.no, espen.nilsen.no@gmail.com, alk@hib.no, okure@item.ntnu.no

Received: Accepted: Published:

Abstract: Conservation of energy is one of the main challenges in designing a wireless sensor network (WSN). The reason is that these large scaled networks cannot be arranged, configured, or maintained manually. Thus, automated deployment and configuration are required. One important factor determining the total energy consumption is the network topology. This article evaluates the relation between the maximum distance (link lengths) between the nodes in a WSN and the total energy consumed. The optimal topology for the two most commonly used medium access control (MAC) protocols were found. A WSN based on a Time Division Multiple Access (TDMA) protocol is limited by the maximum available or allowed emitted radio power. Thus, the criterion for optimal link lengths is related to the expected number of transmissions over the links. By including the retransmissions over the links we found an optimal internode distance. A Carrier Sense Multiple Access (CSMA) based WSN, on the other hand, is limited by the consumed energy of the overhearing nodes. In an analysis including only the overhearing nodes, the link lengths should be as short as possible and the connectivity of the network limits the link length used. However, we found that in a sparsely populated WSN, the total energy consumption increased for shorter link lengths as they were decreased from the optimal link length. *Copyright © 2014 IFSA Publishing, S. L.*

Keywords: WSN; Energy Efficiency; Multi-Hop Routing; Hop Length; Network's Life Time

1. Introduction

A typical wireless sensor network (WSN) consists of several battery powered autonomous devices. The devices are equipped with a unit for sensing targeted environmental attributes and a communication unit that enables communication with a designated node, that provide data collection (sink). The communication capability of a sensor node mainly serves two tasks: transmitting the sensor data generated by the node and relaying packets on behalf of other nodes. This article focuses on the transmission and transport of data packets through the WSN.

The characteristics of the radio unit in a WSN differ from traditional radios. Many find it strange that the receiver consumes approximately the same amount of energy receiving (RX) a packet as the sender consumes in transmitting (TX) the packet. The reason is the low power emitted from the sender, as explained in [1][2]. The datasheet of the RF Transceiver CC2420 [3] verifies the statement. Due to the equality in energy consumption, it is very important to reduce the number of overhearing nodes. Overhearing nodes receive the packet from the sender, but they are neither the destination nor next hop node. However, they learn this after receiving the packet and analyzing the packet's Medium

Access Protocol (MAC) address. The conclusion in [4] was that for a WSN using a Carrier Sense Multiple Access (CSMA) protocol the energy consumption in a WSN is dominated by the energy consumption in the overhearing nodes. In a WSN using time division multiple access (TDMA) protocol it was found [4] that the distance between two communicating nodes (the terms internode distance and link lengths are also used) should be as long as feasible. An alternative solution would be to include a relay node. However, this solution would increase the total energy consumption, as more energy is consumed during reception and retransmission.

Our contribution is first the proposed function that relates the packet delivery ratio (PDR) and the distance between two communicating nodes. The function is based on observed results and it is further based on published results. Next, an expression is derived for the expected number of transmissions required along a path. Simulations are used for validating and comparing the statistical result. Finally, we analyzed a 2-D WSN. Following the randomly node deployment, a routing protocol defines the topology. The routing protocol had the core functionality of RPL [5] with an object function [6] that limited the candidate link according to a link quality requirement. We used the expected number of transmissions as the link quality requirement. Finally, the energy consumption in a WSN with different link lengths is found for WSN using CSMA and TDMA. We found that both protocols have an optimum internode distance.

The novelty of the article is a more realistic function relating the packet loss probability and the node distance. Currently, most publications are based on a model where the packet loss probability is zero if the node distance is less than the transmission range, and if the mode distance is greater the losses are 100%. The model is commonly referred to as the disk model. Although this is not a good model, it is used mainly due to a lack of alternatives. The second contribution is that the model is applied for determine the optimum link distance in WSNs using TDMA and CSMA.

The paper first presents related works, and then the model for packet losses is introduced and applied for assessing packet losses over a single path. Next, the model is applied for a real WSN, where energy optimum link lengths are found for a WSN using TDMA and a WSM using CSMA. Finally, the paper presents the conclusion.

2. Related work

Energy saving in WSN is vital for the operation of the network and many publications have assessed this issue. This article revisits the challenges addressed in [4]. In [4], no lower optimum transmission range was found for a WSN using a CSMA MAC protocol. Another result was that in a WSN using a TDMA MAC protocol, the transmission should be as long as possible and no upper limit was found. In addition to the related publications presented in [4], this section supplements related works.

Several publications have addressed the topic of transmission power in WSN. The conclusion found in the survey [7] was that the transmission power should be as low as possible to reduce the energy consumption in overhearing nodes. This conclusion has been the starting point of several publications on transmission power control. Several earlier publications have, however, stated the opposite conclusion, as in [8]. It lists twelve reasons for having long internode distances (using high transmission power). One reason is that longer hops are more efficient as they are closer to the Euclidean distance between source node and sink. The same results are found in [9][10], where sparsely populated random networks were shown to cause long paths. Our simulation confirms these results.

A general observation regarding transmission power and energy consumption in WSN is that the conclusion depends on whether the assessment has included all effects that determine the total energy consumption. The important effects are the physical radio, the link quality, the MAC layer, and the routing layer. Furthermore, the conclusion depends on the technology used, for example which MAC technology that is used. The conclusion of [8] is valid for a TDMA, but as the MAC layer is not included in the assessment the conclusion is not valid for a CSMA. However, the conclusion found in [7] is valid for CSMA as the overhearing nodes dominate the total energy consumption.

Statistical analyses of WSN paths are presented in several publications [11]-[14]. Several of these articles consider path reliability under high traffic loads where the queuing of packets is included in the intermediate nodes. Our analysis is of WSNs with low traffic load, which is a valid assumption for many WSN that reports data at a low duty cycle.

Another contribution of this article is the proposal of applying a Fermi-Dirac function for expressing the relation between PDR and distance between the two communicating nodes. The commonly used disk model is not realistic, even though it is very often used. Using our model we derive an expression for the losses along a path in a WSN. The model was also used in assessing an ordinary randomly deployed WSN. The Fermi-Dirac function was proposed as a function relating the packet losses and Receiver Signal Strength Indicator (RSSI) in [15]. Our proposal is to use this function for relating distance and PDR.

3. Packet loss rate for a link

The performance of the individual radio links determines much of the overall network performance in a WSN. This section presents the fundamental performance issues related to a single radio link. The radio used follows the characteristics of the RF Transceiver CC2420 [3].

Some publications present measured performance of sensor nodes [15]-[18], like the PDR as a function of distance. The observed relation between packet loss and RSSI are presented in [15]. In [15], the *Fermi-Dirac* function is

proposed as a function relating the packet losses and RSSI. Here, we propose to use the same function for relating the PDR and the distance between two communicating nodes:

$$f(x) = \frac{1}{1 + e^{\frac{x - x_0}{x_1}}} \tag{1}$$

Fig. 1 show data extracted form [17] and the approximated function. The internode distance is x, and x_0 and x_1 are fitting parameters. The PDR_1, for a single trial of transmitting a packet between two nodes is: $PDR_1(x) = f(x)$. The packet loss rate (PLR) is $PLR_1(x) = 1 - f(x)$.

Radio links are made more robust by retransmitting packets that are not acknowledged. However, the number of transmissions attempts has to be limited. Without an upper limit, a high number of retransmissions depletes the node's energy and in addition, causes large, unpredictable delays over the links. Therefore, a maximum number, m, of transmission trials are permitted for each packet before the sender discards it. This means that the sender tries to retransmit the packet a maximum of (m-1) times. Based on the PDR_1 for a single packet, it is possible to estimate the PDR(m) of a link:

$PDR(m) = p_1 * Sum[(1-p_1)^i, \{i, 0, m-1\}]$

$$PDR(m) = 1 - (1 - p_1)^m \tag{2}$$

where $p_1 = PDR_1$ and the summation of the series is written using the notation: $Sum[a_i, \{i, 1, m-1\}] = a_1 + a_2 + \ldots + a_{m-1}$. Solving for packet loss rate, PLR(m), over the link: $PLR(m) = (1 - p_1)^m$.

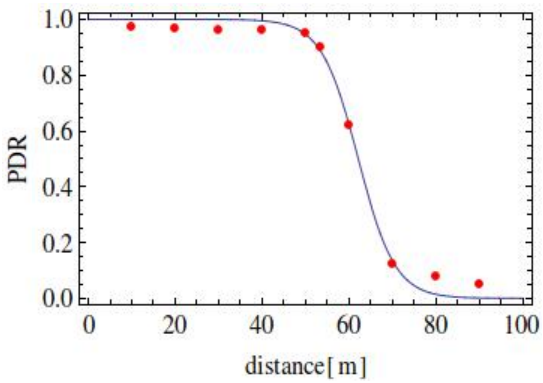

Fig. 1. Packet delivery ratio (PDR) as a function of distance between sender and receiver. The red dots are measured data adopted from [17].

4. Packet error rate for a path

A WSN is a two dimensional (2-D) network as shown in the example in Fig. 2.

Each node, depicted as blue dots, may produce data destined for the sink located in the lower left corner. As can be seen in this example WSN, not all nodes were directly connected to the sink and they required help of neighboring nodes to relay their packets. The pattern of arrows was found by a routing protocol, where the forwarding link was chosen among a set of candidate links. Routing is explained in the next section.

In this section, we present the results of data packet transmission along a single path. This one-dimensional network (1-D network) of N nodes (and one sink) is illustrated in Fig. 3. In our assessment, the data-producing node, $Node_1$, was to the left and the sink (the destination) was to the right. Between these two nodes were a number of relaying nodes. Their only task was to relay the data packet produced by $Node_1$.

Our goal was to forward a data packet using as little energy as possible. The comparison was between different node arrangements using different link lengths. We sought the node arrangement that consumed least energy. The only energy consumption that

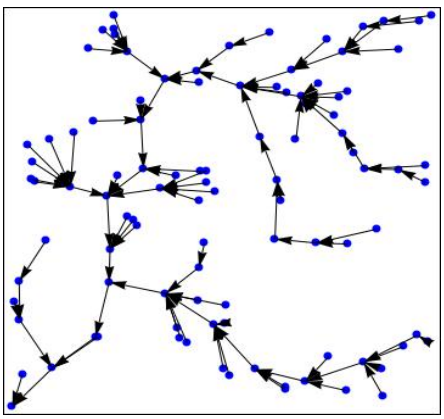

Fig. 2. A WSN with 100 nodes (blue dots), including the sink in the lower left corner. The paths are show as arrows.

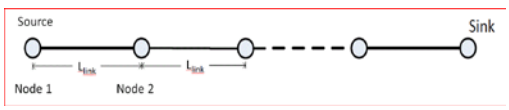

Fig. 3. A path from source node to the sink represented as a 1-D network.

differed between the node arrangements was due to the difference in the number of packet transmitted and received (TX/RX). The network was designed according to common WSN principles where only the communicating nodes (that is the current sender and receiver) were consuming TX or RX energy. Nodes not participating in the communication were in sleep mode. The number of packet TX/RX for a given node arrangement was proportional to the energy difference between the node arrangements. This is a commonly used comparison for example in [11]. The MAC protocol used in the 1-D WSN was TDMA. In TDMA, each pair of communicating nodes was assigned a time slot. The optimal node arrangement was found as the configuration that required least packet treatments. The assessment of the 1-D network was based on both a statistical description of the link performance and a simulated performance. The motivation for using both was to produce two independent comparisons.

In order to illustrate the usability of the theoretical loss function, f(x), two arrangements of the nodes was used. The first arrangement had equal distance between the nodes and equal number of nodes in each calculation. The result, presented in Fig. 4, is for a single path of 50 nodes and one sink. The figure shows the required number of packet transmitted along the path from the first node to the sink. The node distance is altered and its value is presented as ratio to the 50% PDR values given as x_0 in equation (1).

The simulations presented in Fig 4 shows that as the node distance increase the number of transmitted packets starts to increase as the node distances approaches x_0. Clearly, the total number of packet depends on how many transmission attempts are permitted over each link. The effect is clearly visible as the maximum permitted transmission attempts are increased. It is important to notice that as the link lengths increases beyond the distance critical transmission range, the number of transmissions found for the path is equal to the number of transmission attempts over the first link. The explanation is that the first node cannot transmit the packet to its neighboring nodes. However, it tries m times to transmit the packet. Even if this node arrangement is not realistic, it illustrates the important point of not treating each link independent.

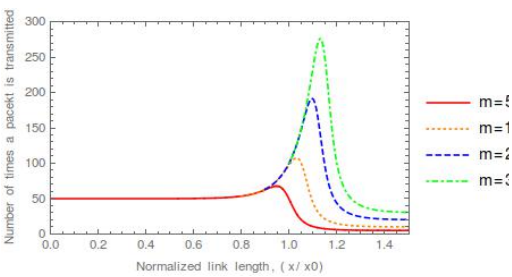

Fig. 4. Total number of packets transmitted from one source node to the sink along a path of 50 hops. The maximum number of transmission attempts (m) is varied from 5 to 30 as shown.

The second node arrangement is presented in this section. It was a path where the distance from the source node to the sink was fixed, thus making it more realistic. The distance from the source node ($Node_1$) to the sink was L_{tot}. L_{tot} was a fixed parameter in this simulation. The distance between the nodes was L_{link}. L_{link} was assumed to apply for all inter node distances, except the last link to the sink, which may be shorter, as explained below. Given the distance between nodes (L_{link}), the number of nodes required to connect the source and sink was found as Ceiling[L_{tot}/L_{link}]. The Ceiling[x] operator returns the smallest integer greater than or equal to x.

The expected performance of the 1-D network depending on L_{link} is: Short internode distances will have few retransmissions at each link, but the path will consists of a higher number of links compared to a path of longer internode distances. However, as the link distances approaches the zone of less quality, retransmission over the individual links starts to limit the gained benefit of reduced hop counts. In the end no packets get through, when the link distance causes disconnected links. Then the sender at Node1 only transmits the packet m times over the first link, before discarding the packet. The description given above is supported in the following statistical derivation of the expected performance.

4.1. Number of transmissions along a path based on our proposed model

The expected number of transmission (ETX) over a single link, allowing maximum m transmissions is: $ETX(m) = (1-q_1)\ Sum[\{n\ q_1^{n-1}\}, \{n, 1, m\}] + m\ q_1^m$, using the notation $q_1 = PLR_1$. The summation gives the following result:

$$ETX(m)=\frac{1-q_1^m}{1-q_1} \tag{3}$$

The importance of not treating each link independently is illustrated in the discussion related to Fig. 4. Therefore, the throughput along the path depends on the probability of a packet reaching the intermediate nodes. The probability that a packet reaches node number z ($Node_z$) is:

$$P[Node_z] = (1-q_1^m)^{z-1} \tag{4}$$

However, the packet losses of the individual links were assumed statistically independent. Thus, the expected number of transmissions along the path is: $ETXpath(N)=ETX(m)*Sum[(1-q_1^m)^{(i-1)},\{i, 1,N-1\}]$

$$ETXpath(N)=\frac{1-q_1^m-\left(1-q_1^m\right)^N}{q_1^m\left(1-q_1\right)} \quad (5)$$

The performance of a 1-D network can be derived from (5). If only high quality links are used, PDR_1 approaches one (q1 approaches zero). Then the number of transmissions required is N-1, which equals the hop count. However, if disconnected links are used, PDR_1 approaches zero (q_1 approaches one). Then it is found from (5) that the number of transmissions equals the maximum transmissions of the first link, which is m. This is in accordance with the results found using simulation as presented in the next section.

Fig 5 illustrates the average number of transmissions required to reach the sink from a node located 500 meters away. The parameter that was changed was the distance between the nodes. Both theoretical and simulated results are presented. The effect of discontinuous changes in the number of links appears as discontinuous changes in the number of transmissions. Fewer transmissions are needed when longer link lengths are permitted since fewer hops are required. However, poor link quality caused retransmission. This can be seen by the increase in number of transmission at link lengths above 70 meters. The theoretical results are given by (5) while the simulation results were produced using the OMNeT++ [20] simulation framework and simulation models available in the MiXiM simulator [21]. To mimic a real world sensor node the theoretical and simulation were according to the CC2420 datasheet [3] and the IEEE802.15.4 standard [19].

5. Optimal topology in a WSN

The previous analyses established some fundamental understanding of the performance of packet transfer along a path subjected to packet errors. In this section, the performance of the individual links are used to find the optimal network configuration in terms of link length that minimize the total energy consumption in a two-dimensional (2-D) WSN. The comparison follows the method presented earlier where the performance is according to the energy consumption for different network configurations. The difference is presented in number of times packets transmitted and received (TX/RX). Here, all nodes, except the sink, generate data. This implies that the nodes are both data generators and some are also relay nodes.

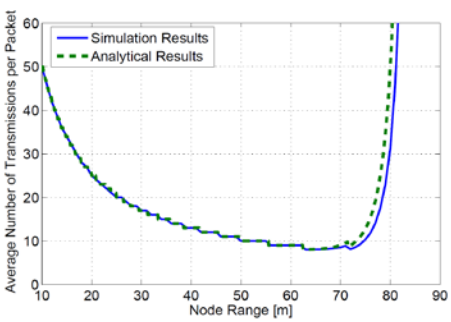

Fig. 5. Number of transmission as a function of node separation.

The simulation was done by randomly deploying the nodes in a 400x400m^2 area. The paths were established using a routing protocol (RPL) with ETX link quality criteria as presented in the introduction and a more detailed presentation will be given. Next, the difference between energy consumption is presented as the difference in the number of times packets had to be transmitted and received in order to get to the sink. The sink was placed, as shown in Fig. 2, in the lower left corner. The motivation was that we wanted to investigate the consequences of long paths. Each node arrangement was repeated at least 200 times in order to gain statistical confidence. The nodes were interconnected as the routing protocol determined which links each node should use to forward their data. However, not all node configurations produced a connected graph. These arrangements were not included in our results. Especially for the short maximum internode distances (producing sparsely WSN), it was difficult to produce a fully connected WSN.

The number of transmission permitted at each link was introduced above as the number m. In our analyses of a single path, it was also pointed out that when link lengths become too large the end-to-end performance is compromised, as the packets do not get through to the sink. A low value for m would produce an optimum network with disconnected links. Therefore we used m=100 in our simulations.

Before the number of TX and RX was found the nodes had to be arranged and interconnected. The nodes were arranged by randomly positioning them in the area. The next step was to interconnect the nodes using a routing protocol. Here, we used the selection criteria of the RPL routing protocol [5], with an object function that selected only links fulfilling the defined EXT requirement [6]. The requirement is given in the following text. The ETX was found from the proposed function relating PDR and internode distances and the ETX of the individual links derived above. Thus, this selection criteria and the random node positioning produced links with different lengths, but with an upper limit with respect to internode distance. This means there was a maximum link length (or internode distance).

Different MAC protocols produce different results. Therefore, the two most popular MAC protocols were evaluated according to their characteristic performance as presented below. In the 1-D network, it was assumed that only the communicating nodes were active. In our analyses of the 2-D WSN we used the same assumption. The first MAC protocol analyzed was a TDMA protocol, where the two communicating nodes were assigned a time slot. Time slots were assumed allocated during network establishment (as in, e.g., WirelessHART™ [22]). The overhead due to network configuration is not included in our assessment. The motivation for omitting this is that the total energy consumption is determined by the long-term operation. Furthermore, several assumptions would have had to be made regarding signaling and initial connections.

The second MAC protocol analyzed was a CSMA MAC protocol. In CSMA, all nodes in the reception area of the sender decoded the packet in order to determine if the packet was destined to them. Since all nodes in the sensing range overheard the messages, it was concluded in [4] that the transmission range (link length) should be as short as possible.

Before the energy consumption results are presented, some observations are given regarding paths length in WSN. The reason for this discussion is that the characteristics of the paths in WSN determine the performance of the network. It was pointed out in [9][10] that short link length in randomly deployed networks results in long paths. This was clearly observed in our simulations. In Fig. 6, the maximum number of hops is presented for a network of 200 nodes.

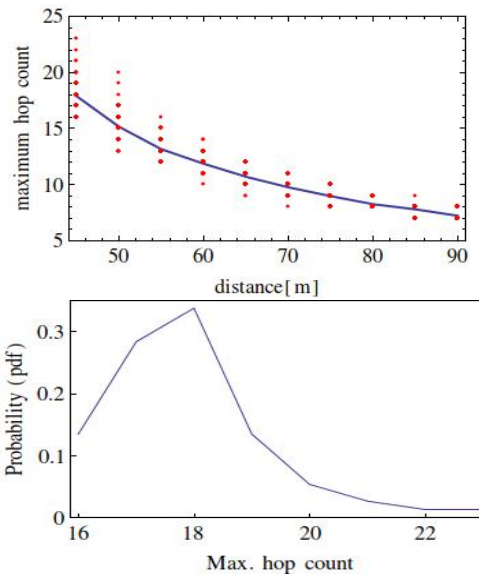

Fig. 6. The upper figure shows maximal hop count as a function of maximal permitted node distance. The lower figure shows the estimated probability distribution function (pdf) of maximal hop for a node distance of 45m

The figure shows that the variation in maximum hop count is larger for networks using short link lengths. In addition, as illustrated in the lower graph in Fig. 6, the distribution of maximum length has a tail towards long paths. These long paths demand many transmissions in order to forward packets to the sink. Clearly this is negative as they consume much energy, and in a CSMA based WSN the long paths will have many overhearing nodes along the paths.

5.1 Optimal topology for TDMA

The TDMA simulations are based on a radio with a fixed transmission power. The motivation was that in TDMA, the link length should be as large as possible, following the conclusion in [4]. However, the emitted power cannot be increased beyond a limit. The limit is determined by the design of the radio and/or the regulations.

In the TDMA simulations, we sought optimal performance with respect to optimum internode distance. The limitation on the distance was the ETX of the individual links. ETX is proposed as an alternative metric to filter out only links fulfilling a defined quality determined by the object function [6]. Our model, f(x), has a direct relation between ETX and the distance between the nodes, as presented above. The challenge of estimating the ETX was thus avoided, and the routing protocol did not have to establish this information based on over the air communications. The assumption was according to the assumption stated earlier that the management traffic was not included.

In Fig. 7, a typical result is presented. Starting with the shortest allowed link length, it can be seen that by allowing longer links the total energy is reduced as the paths get

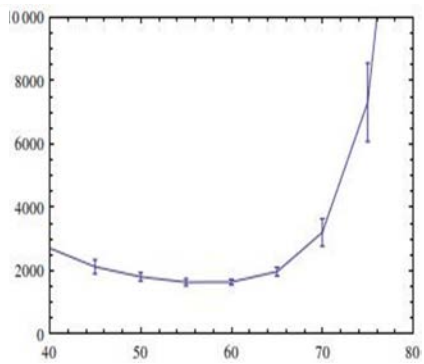

Fig. 7. Number of packets transmitted as a function of maximum allowed link length through a TDMA WSN.

shorter. However, the reduced energy consumption reaches a minimum, in the figure at approximately 60m. Further increase in allowed hop length causes increased total energy consumption. The reason is that longer internode distance causes higher ETX. The probability of retransmission over the links increases and this retransmission consumes energy.

5.2 Optimal topology for CSMA

The CSMA simulations were performed differently from the TDMA simulations. Here, the emitted power of the sender is changed. The reason is that for a CSMA WSN, we seek the optimal link lengths for the radio as in [4]. Change in the TX power causes changes in the parameters x_0 and x_1, in (1). In our simulation, x_0 was increased and the original ratio between x_1 and x_0 determined the new x_1. Thus, the grey zone increased with increasing TX power according to observations [17].

The ETX was used as a criterion for link selection in the CSMA simulations. Only links having a PDR_1 equal to or better than 90% were considered candidates for routing. The derived relations above defined the relation between ETX and PDR_1. Clearly, nodes outside the link length overheard the transmission. These nodes were in the sensing range (gray zone), where they received packets, but most often with bit error. However, the nodes consumed energy in receiving the packets and this energy was included in our analyses. The node's sensing range was set equal to x_0. It might be argued that this is too short, but here we used this as a first approximation.

Fig 8 presents the results for a WSN based on CSMA. The left graph shows all data points in addition to the average result of number of TX/RX. The draw line shows the average number of TX/RX required. The interesting observation is that there is a point where the number of required TX/RX starts to increase if the transmission range is decreased beyond an optimal hop length. The reason for this increase is the long paths that are likely to occur as the network is operated close to its connections limits. A second, interesting observation is the variation in number of TX/RX that is shown for the short internode distances. From the figure, it can be seen that for a maximum link length of 45m, there are some very low values and some very high values for the maximum hop count. From these results it can be concluded that it is advantageous to use long-hop also for CSMA.

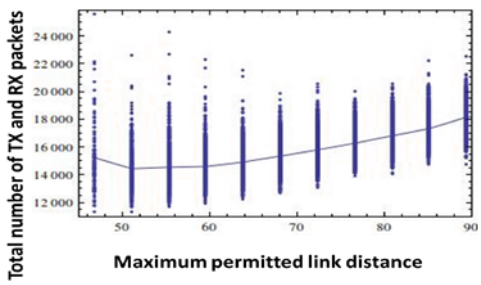

Fig. 8. Total number of TX and RX packets handling as function of maximum link distance. The line shows the average values and the does give each individual result.

7. Conclusions

A *Fermi-Dirac* function was suggested for relating the packet delivery ration (PDR) and the distance between the nodes. The function enables analytical evaluation of WSN. The parameters of the function were found by curve-fitting to published results. Next, the *Fermi-Dirac* function was applied in estimating the total number of transmissions along a path. The estimated number of transmissions showed good correlation to simulated results.

In a WSN with randomly deployed nodes, the link quality of candidate links available for routing was evaluated for a TDMA based WSN. The consequences of having too strict requirement on the links caused higher energy consumption due to the long paths. However, beyond an optimal point the total energy consumption increased rapidly as each link had to retransmit the packet due to the reduced link quality. Thus, it was found that TDMA based WSN has an optimal internode distance.

An optimal node distance was also found for a CSMA based WSN. This is contrary to an intuitive conclusion, where the maximum link length should be as short as possible in order to reduce the unnecessary energy consumption in the overhearing nodes. The increasing total energy consumption for short link lengths was found to be due to the long, none optimal paths that occurred when short link lengths were used.

Our findings can be used as a tool for designing a WSN based on a more realistic description of the relation between the link quality and node distance. The method was used to find the optimum node distance for WSN based on both CSMA and TDMA.

REFERENCES

[1] R. Min and A. Chandrakasan, "Top Five Myths about the Energy Consumption of Wireless Communication", ACM Mobile Computing and Communication Reviews, vol. 7, No 1, Jan. 2003, poster

[2] H. Karl and A. Willig, "Protocols and Architectures for Wireless Sensor Networks", ISBN: 978-0-470-51923-3,John Wiley & Sons, 2007

[3] CC2420, Chipcon Products from Texas Instruments, http://www.ti.com/lit/ds/symlink/cc2420.pdf, visited July 2014

[4] A-L. Kampen, K. Ovsthus, L. Landmark, and O. Kure, "Energy Reduction in Wireless Sensor Networks by Switching Nodes to Sleep During Packet Forwarding" , The Sixth International Conference on Sensor

Technologies and Applications, SENSORCOMM 2012, Italy, 2012, ISBN 978-1-61208-207-3. s. 189-195

[5] T. Winter, et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks”, RFC 6550, Internet Engineering Task Force (IETF), 2012

[6] JP. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks", RFC6551, Internet Engineering Task Force (IETF), 2012

[7] I. Khemapech, A. Miller, and I. Duncan, "A Survey of Transmission Power Control in Wireless Sensor Networks", The 8th Annual Postgraduate Symposium The Convergence of Telecommunications, Liverpool John Moores University, 28th-29th June 2007

[8] M. Haenggi, "Twelve reasons not to route over many short hops", VTC2004-Fall. 2004 IEEE 60th Vehicular Technology Conference , Vol. 5, 2004, pp. 3130 – 3134.

[9] P. Szczytowski, A. Khelil, A. Ali, and N. Suri, "TOM: Topology oriented maintenance in sparse Wireless Sensor Networks", 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011, pp. 548 – 556.

[10] S. Vural and E. Ekici, "On Multihop Distances in Wireless Sensor Networks with Random Node Locations", IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 9, NO. 4, 2010, pp. 540 – 552.

[11] Z. Rosberg, R.P. Liu, T. L. Dinh, Y. F. Dong, and S. Jha, "Statistical reliability for energy efficient data transport in wireless sensor networks", Wireless Netw, 16, 2010, pp. 1913-1927.

[12] R P. Liu, Z. Rosberg, I. B. Collings., C. Wilson, A. Y Dong, and S. Jha, "Energy Efficient Reliable Data Collection in Wireless Sensor Networks with Asymmetric Links", Int J Wireless Inf Networks, 16, 2009, pp- 131-141-

[13] T. Zheng, E. Kamel, and S. Wang, "Node performance model of wireless sensor networks", 2013 9th Asian Control Conference (ASCC), 2013, pp. 1 – 6.

[14] N. Farzaneh and M. H. Yaghmaee, "Probability based hop selection approach for resource control in Wireless Sensor Network", 2012 Sixth International Symposium on Telecommunications (IST), 2012, pp. 703 – 708.

[15] A. Bildea, O. Alphand, F. Rousseau, and A. Duda, "Link quality metrics in large scale indoor wireless sensor networks", IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), 2013, pp. 1888 – 1892.

[16] J. Zhao and R. Govindan, "Understanding Packet Delivery Performance In Dense Wireless Sensor Networks", Proceedings of the 1st international conference on Embedded networked sensor systems, SenSys '03, Pages 1 - 13, ACM New York, USA 2003, pp. 1 – 13.

[17] Y-D. Lee, D-U. Jeong, and H-J. Lee, "Performance analysis of wireless link quality in wireless sensor networks", 5th Int. Conf. on Computer Sciences and Convergence Information Technology (ICCIT), 2010, pp. 1006 – 1010.

[18] L. Siyu and G. Hongju, "Propagation characteristics of 2.4GHz wireless channel in cornfields", IEEE 13th International Conference on Communication Technology (ICCT), 2011, pp. 136 – 140.

[19] Available at http://www.ieee802.org/15/pub/TG4.html

[20] OMNeT++, http://www.omnetpp.org/, visited July, 2014

[21] MiXiM, http://mixim.sourceforge.net/, visited July 2014

[22] WirelessHART™, IEC 62591 ed1.0, http://webstore.iec.ch/webstore/webstore.nsf/ArtNum_PK/43964, visited July 2014