Leveraging Spatial and Temporal Correlation in Wireless Sensor Networks

Thesis submitted for the degree of
Doctor of Philosophy
By

Bracha Hod

Submitted to the Senate of
The Hebrew University of Jerusalem
June 2012

This work was carried out
under the supervision of:

Prof. Danny Dolev and Dr. Tal Anker

Acknowledgments

This thesis owes its existence to many people.

First, I would like to express my gratitude to my supervisors, Prof. Danny Dolev and Dr. Tal Anker. I have been extremely fortune to have Danny as an advisor. With his immense knowledge, rich experience and uncompromising quest for excellence, he provided me with the tools to develop into an independent researcher. He has been a constant source of inspiration and infinite patience. I am also grateful for his help in finding funding for my work on this thesis. Working with Tal has been a unique intellectual experience. He has been a valuable and important resource - providing useful insights and directions for my work. Tal has inspired me with his motivation, encouragement and enthusiasm.

Very special thanks also goes to Dr. Danny Bickson, who gave me the motivation to get through during the first stages of my research. Danny also introduced me to the area of graphical models and belief propagation, which was a significant part of my work.

Dr. Gaddi Blumrosen was my colleague in the practical work with the sensor nodes. I thank him for the knowledge he shared with me on signal processing and wireless communication, enabling me to gain important insights into sensor nodes in the real world. I also thank him for his friendship and support during the joint work.

It was a great pleasure to work with Dr. Ami Wiesel. I am grateful to him for guiding me with his wide knowledge in statistical learning and in robust statistics. His expertise, understanding, and wealth of ideas, combined with patience and enthusiasm, significantly contributed to the last part of this thesis.

I owe deep thanks to Mrs. Amy Fredj and Mrs. Susan Goodman for their patient

correcting of my English and for giving me important writing skills.

I would like to thank the members of my committee: Dr. Dana Porrat and Prof. Shay Kutten for the assistance they provided at all levels of the research.

I thank all my friends in the university lab and those at Intel for their encouragement and support. I benefitted so much from the discussions we had together and the advice they gave.

Finally, I would like to thank my family and especially my parents. They have all supported me in every possible way whenever I needed them. In particular, I want to express my gratitude to my husband and best friend, Achikam, for the infinite patience, endless encouragement and amazing love that he gives me every day. This thesis is dedicated to him and to our wonderful daughters: Ortal, Meital and Hadar.

Abstract

Wireless Sensor Networks (WSNs) have been a major research topic over the last decade, attracting a great deal of attention from scientists and researchers, due to their great potential and the challenges they pose. WSNs are a promising technology because they make possible a wide range of applications. A WSN, composed of a group of sensor nodes with sensing, data processing and communication, can be deployed for environmental monitoring, health-care applications, security and safety. However, the limited resources and capabilities of the sensor nodes make the data generated by the sensor nodes unreliable and inaccurate, and make the design of efficient and effective networks a real challenge.

This thesis proposes to leverage spatial and temporal correlation of sensor nodes in order to improve various aspects of WSN performance. Spatial correlation enables data aggregation that combines similar observations, and removes redundant information. This combined information is more reliable, the amount sent through the network is significantly reduced, and the energy consumption is therefore decreased. Temporal correlation allows reconstruction of faulty measurements using correct measurements. The robustness and reliability of the reported observations are therefore increased. Spatio-temporal correlation combines both spatial and temporal correlation and thus it is powerful in many aspects, including an improved detection and estimation of processes and of events observed by the sensor nodes.

The research is divided into three parts. Each part making a specific contribution to its own domain. In the first part, we designed a distributed inference scheme based on graphical models that statistically capture the spatial correlation. We explored this inference scheme by applying it to the clustering problem of grouping sensor nodes

into an efficient hierarchical model. In the second part, we focused on a tracking application in short-range areas using signal-strength measurements. High transmission rate combined with signal processing techniques, such as interpolation and filtering, were used to effectively exploit the temporal correlation between consecutive measurements to get an improved tracking accuracy. In the third part, we investigated detection of anomalous and events when the data contains outliers. Spatio-temporal analysis with robust statistics principles allows the distinction between local faults of a node and global events in space and time domains.

The main contribution of this thesis is by providing realistic solutions to problems that are state-of-the-art in the field of WSNs. The solutions use techniques from different disciplines that were adapted to the unique characteristics of WSNs. The proposed solutions were evaluated through simulations and real-time experiments and were proved to be efficient, robust and powerful.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Wireless Sensor Networks

Wireless Sensor Networks (WSNs) [1, 2, 3] are wireless networks which are capable of providing powerful applications in many fields [4, 5, 6]. A WSN can number from just a few, up to several thousand sensor nodes. Sensor nodes are tiny devices with three major capabilities: sensing, data processing and communicating. They usually have one or several sensors, which can be related to physical aspects of the environment or to biological and chemical processes [7, 8]. The sensor nodes also have low-cost embedded processors and low-power radios, but they do have limited computation capabilities, bandwidth and energy supplies. These limitations mean that each stand-alone sensor node is useless. It is when they cooperate that a wide range of applications becomes feasible [9].

The applications used in WSNs are categorized into two types: event detection and continuous data collection. For example, a WSN can be deployed over an area to detect an enemy intrusion [10] or the presence of dangerous gases [11]. Sensor nodes can be integrated into a wearable network for providing early detection of medical conditions [12]. In addition, WSNs can continuously control light, temperature and humidity in greenhouses and other buildings [13, 14]. In all these applications, the nodes make observations of the area of interest and collect measurements from the environment. These measurements are further processed, aggregated and transmitted

to a control center using communication protocols.

The communication protocols have been especially designed to deal with the challenges arising from the distinctive characteristics of WSNs [15, 16, 17]. The limited power sources of the battery-powered sensor nodes means that energy efficiency is a major design goal. Their self-organization requires a distributed methodology because a central management does not always exist. In addition, the wireless medium is characterized by interference, collisions and poor link quality, which cause noisy measurements and continuous changes in the network topology. Robust protocols are necessary in order to deal with this medium and with faulty nodes. Some WSNs, e.g. networks for military applications, are expected to have thousands of sensor nodes. Large networks may lead to scaling problems. For example, network protocols are more complicated since the paths are long; information is propagated slowly and involves many nodes that can breakdown.

1.2 Leveraging Spatial and Temporal Correlation

This thesis proposes to leverage spatial and temporal correlation of sensor nodes for fulfilling the special requirements of WSNs [18]. Spatial correlation in WSNs usually means that observations of proximate sensor nodes are statistically dependent and it is possible to predict a sensor node's observations from observations of its nearby neighbors. Spatial correlation is established because the nodes are deployed close to each other and, presumably, sense the same events. Temporal correlation means a statistical correlation between consecutive observations of a single node and implies the ability to predict some observations of a single sensor node based on its preceding or subsequent observations. Temporal correlation is expressed because the events monitored by the sensor nodes are mainly constant over a short time period or occasionally tend to repeat themselves.

Leveraging spatial and temporal correlation can improve various aspects of WSN performance. Spatial correlation enables data aggregation [19, 20, 21, 22] that combines similar observations and removes redundant information. The combined information is more reliable because faulty observations can be identified and discarded.

In addition, the amount of data sent through the network is significantly reduced and therefore the energy consumption is decreased. Temporal correlation allows reconstruction of lost or noisy observations using correct measurements. And hence the robustness and reliability of the reported observations are increased. Spatio-temporal correlation combines correlation in both space and time domains and thus it is a powerful tool in many aspects, including an improved detection and estimation of processes and events observed by the sensor nodes.

The research is divided into three parts:

- In the first part, we developed a scalable and practical implementation of the belief propagation (BP) algorithm for WSNs [23]. Belief propagation (BP) is a message passing algorithm for performing inference on graphical models. Our scheme maps a WSN into a graphical model by constructing multiple trees which are based on spatial correlation between the nodes. This graphical model can be used for inference tasks for a large set of applications. We examined the constructed model by applying it to the problem of clustering [24]. Clustering is the grouping of sensor nodes into a hierarchical model. The clusters provide means to exploit spatial correlation by enabling aggregation techniques.

- In the second part, we focused on a tracking application in short-range areas [25, 26, 27]. The tracking application continuously estimates a mobile node's location from signal-strength measurements received on the radio links in the network. With an adequate transmission rate, we can use the knowledge about the continuity of the movement, so temporal correlation between consecutive measurements is exploited to interpolate missing measurements and filter the noise from the data. In a continuous movement, consecutive RSSI measurements refer to proximate locations of the mobile sensor node and can compensate for channel interference distortion. This application was designed for Body Sensor Networks (BSNs) rather than for the traditional WSNs that were explored in the first part.

- In the third part, we investigated the challenge of efficient outlier detection in

the network. Outlier detection techniques identify anomalous observations either to eliminate them or to report on an occurrence of events. We suggest a spatio-temporal analysis with robust statistical principles that allow an effective estimation of statistical measures that are used for outlier detection. A robust mean is estimated using a polynomial regression which employs the temporal correlation between consecutive observations. Spatial correlation among neighboring nodes established an efficient local and distributed estimation of the inverse covariance matrix using a Gaussian graphical model. These measures are further used to identify noisy observations and to detect events in the network.

1.3 Thesis Outline and Summary of Contributions

We begin with theoretical work on large-scale WSNs that is verified using simulations. We then continue with practical work involving real sensor nodes in a small scale. Finally, we combine theory with real experiments on a larger scale. The main contribution of this thesis is to provide realistic solutions to problems that are cutting edge in the field of WSN. The solutions use techniques from different disciplines that are adapted to the unique characteristics of WSNs.

The first part of the thesis is presented in Chapter 2 and Chapter 3. Chapter 2 starts with a brief background on the BP algorithm and the Min-Sum implementation. It then elaborates on the challenges of integrating the algorithm in WSNs and finally a practical scheme is presented. Due to limitations of space, experimental results that show the properties of the scheme are omitted from this thesis [23]. Some evaluation of the proposed scheme is combined with the evaluation of the clustering algorithm in Chapter 3. The main contribution of this part is to provide a general scheme for distributed inference in WSNs that can be used for a wide range of applications. The scheme is fully distributed and localized, asynchronous and robust. It introduces a minor and consistent cost in communication and overhead, regardless of the size of the network. Chapter 3 uses the BP scheme presented in Chapter 2 to construct an efficient clustering framework. The chapter includes a formalization of

the clustering problem in WSNs and a description of the BP-based algorithm. The algorithm is evaluated by simulations and a final conclusion closes this chapter. The significant contribution of this chapter is by introduction of a new algorithm for efficient clustering that considers not only local properties of a node, such as residual energy or degree, but also takes into account joint characteristics of a group of nodes, such as link quality and topology information. Utilization of all available data, while maintaining small constant message and time overhead, leads to considerable increase in network performance and balanced power consumption among the nodes.

The second part is described in Chapter 4, where a different direction is studied. The chapter presents a new RSSI-based tracking system that is designed for small networks, such as BSNs. The chapter begins with an introduction, then it describes the system model and the problem formulation. The data processing and the calibration process are also described. The experimental setup and results are presented. The chapter ends with a conclusion and discussion about future work. This work makes a two-fold contribution. Firstly, we employ a new RSSI-based tracking system that exploits an a priori knowledge about the system settings to improve the tracking accuracy. The calibration technique, the transmit power selection and the filtering were optimized for a small scale that fits BSN, resulting in a high accuracy in a scale of few centimeters. Existing tracking systems for an indoor environment are usually designed for general applications and therefore present an accuracy in the scale of one meter. The other contribution of this work is by the presentation of the capability of RSSI to estimate body part placements. The experiments showed that RSSI can provide a clear pattern of the movement without drift and regardless of the channel conditions. Classification of patient activities and movement can be therefore extended from this work.

Chapter 5 elaborates on the third part of the thesis that deals with outlier detection. The chapter starts with an introduction and a formulation of the outlier detection problem as a spatio-temporal statistical problem. A robust estimation of the statistical measures for outlier detection is proposed. Then, the properties and the advantages of our proposed solution using synthetic simulations and experiments with real world data are presented. A discussion and directions for future work close

this chapter. The contribution of this work is the introduction of a novel method for effective outlier detection using a combination of robust statistic tools and graphical models. The method uniquely addresses the robustness and efficiency requirements for outlier detection in WSN environment. The proposed method succeeds in identifying anomalies and events in the network under noisy and faulty conditions, and without any prior information. The outlier detection is fully distributed, localized and requires only a small number of observations and messages. The method is general and suitable for a wide range of applications, including time-series processes.

Chapter 6, the concluding chapter, presents a general discussion of the work done and indicates directions for future work.

Chapter 2

Belief Propagation in WSNs

This chapter starts with a brief background on the Belief Propagation (BP) algorithm and the Min-Sum implementation. It then elaborates on the special characteristics of WSN and the challenges of integrating the BP algorithm in WSNs. Finally, a practical scheme for using BP in WSN environment is presented.

2.1 Belief Propagation

WSNs are generally designed to gather and process information from their environment. This information can be huge, erroneous and inappropriate for direct use by the applications. Information processing, including combination, estimation and inference of the sensor nodes' observations are therefore important.

Statistical methods for distributed inference often use graphical models. Graphical models [28] are useful formalism of statistical structure using graphs. In graphical models, each vertex in a graph corresponds to a random variable and each edge represents a conditional dependency between the variables. An absence of an edge in the graph implies that the variables are conditionally independent.

Belief propagation (BP) [29, 30], is a message passing algorithm for performing inference on graphical models. BP is presented in the literature as an effective and useful inference method for a wide range of communication applications and network topologies, including WSNs [31, 32]. The idea behind the BP algorithm is that the marginal or posterior probabilities can be efficiently computed in a distributed

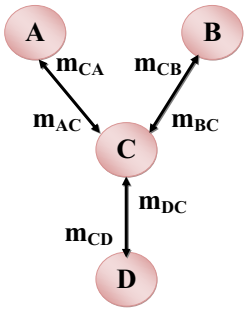


Figure 2.1: Distributed Computation of the Posterior Probability

manner, using Bayes' theorem and by means of local message-passing. Consider, for instance, the four events as illustrated in Figure 2.1. It is possible to globally compute the probability of event D, using 8-sum calculation:

$$p(x_D) = \sum_{x_C, x_B, x_A} p(x_D, x_C, x_B, x_A) \ .$$

Alternatively, by a local computation, only 6-sum calculation is required:

$$p(x_C) = \sum_{x_B, x_A} p(x_C|x_B, x_A)p(x_B)p(x_A) \ .$$

$$p(x_D) = \sum_{x_C} p(x_C) \ .$$

More formally, let us denote $m_{ij}(x_j)$ as a message from node i to node j about the state that node j should be. Node i calculates the massage using previous messages it receives from its adjacent neighbors $N(i)$. The message update rule performed by a node i in round t is:

$$m_{ij}(x_j)^t = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}(x_i)^{t-1} \ .$$

The update rule referring to state x_j of node j is a sum over all the possible states x_i of node i. On each state, three elements are incorporated together: the local prior

information $\psi_i(x_i)$, the joint function $\psi_{ij}(x_i, x_j)$ and the direct neighbors information $m_{ki}(x_i)^{t-1}$.

Upon termination, after round $\bar{t}$, the belief at a node i (the marginal of the variable) is the product of the local evidence together with all the incoming messages and a normalization constant α [33]:

$$b_i(x_i) = \alpha\psi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i)^{\bar{t}} \ .$$

The BP algorithm for trees is an exact inference algorithm, which means that the belief converges to the correct marginal values in a finite number of iterations equal to the diameter of the tree.

2.1.1 Min-Sum Algorithm

For energy efficiency, a variation of the original BP algorithm, known also as the Min-Sum (MS) [34], can be used. This algorithm uses only addition and subtraction operations, so it works well with integer values and saves the overhead of floating-point calculations. Additionally, the algorithm uses broadcast messages [35], in order to preserve communication resources.

The MS algorithm computes inference in the negative log domain, which can be equivalently viewed from the physics point of view as an energy, or cost minimization. Taking this into consideration, the goal of the MS algorithm is to minimize the overall cost over all the nodes in the network, based on the local cost functions and the constraints between the nodes. The algorithm is intuitive. Each node transmits to its neighbors a message with its local and joint costs. Each neighbor that receives the message updates its own belief accordingly and transmits the new belief, so gradually the information is propagated through the network until the nodes converge to a common belief. This convergence point minimizes the overall cost in the network. The algorithm, in its broadcast form has three basic steps:

1. *Message Passing*

   Each node i transmits its local evidence on the initial round, and transmits its

belief, based on incoming messages, on the successive rounds. Every broadcast message m_{i*} from node i includes a combined information for all its neighbors, replacing multiple unicast messages. The receivers extract the information intended for them.

$$m_{i*}(x_i)^0 = \psi_i(x_i) \ ,$$
$$m_{i*}(x_i)^t = \psi_i(x_i) + \sum_{k \in N(i)} m_{ki}(x_i)^{t-1} \ .$$

2. *Message Update Rule*
   Upon a reception of message $m_{j*}(x_j)^t$ from node $j \in N(i)$, node i updates its local belief by extracting the unicast information from the broadcast message of node j, using the following calculation:

$$m_{ji}(x_i)^t = \min_{x_j}\{\psi_{ij}(x_i, x_j) + m_{j*}(x_j)^t - m_{ij}(x_j)^{t-1}\} \ .$$

   The value of every message at round $t < 0$ is 0.

3. *Belief Calculation*
   At the end of round $\bar{t}$, where $\bar{t}$ can be chosen to be the network diameter or any other predefined limit, node i determines its final state x_i to be the one which minimizes the total cost.

$$b(x_i) = arg\min_{x_i}\{\psi_i(x_i) + \sum_{k \in N(i)} m_{ki}(x_i)^{\bar{t}}\} \ .$$

2.2 WSN Model

The WSN model referred in this chapter and in the next chapter, is a large network of hundreds to thousands of sensor nodes over large geographical regions. The sensors nodes are battery-powered and they may be deployed in a way that makes it difficult, or not worth, recharging them. The nodes divide their operation time between active (transmission, reception, listening) and sleep modes in order to save power. The wake-up nodes communicate among themselves asynchronously, organize themselves into a multi-hop wireless network and coordinate to perform common tasks. As some nodes

may die over time, and some may move to different locations or may join the network, so the network topology changes over time. The network characteristic leads to the focus on energy-efficiency and network lifetime rather than on latency and bandwidth utilization which are primary concerns in traditional networks. Another important aim of this network is the scalability with changes in network size, topology, and density.

2.3 Challenges

The set of unique constraints and requirements on such WSN demands significant improvements and modifications to traditional algorithms. In particular, the BP algorithm cannot be embedded into WSNs in its original form. Several issues should be addressed to enable its efficiency in WSNs.

2.3.1 Mapping WSN to Graphical Model

Mapping of a WSN into the graphical model appears to be a real challenge. The mapping of the network can be either to a tree or to a cyclic network.

Several papers, such as [31], have shown good results of the Loopy BP (LBP) in practical applications, mainly decoding. [36] discusses the convergence of the BP algorithm in general networks with single or multiple loops. The LBP worked well in these cases mostly because the cycles in the graph were large, so the effects of the cycles faded after only a few iterations. WSNs are associated with short cycles made by the broadcast range, so LBP is not an appropriate method for many applications in such networks. The use of the Min-Sum algorithm for energy efficiency also implies some limitations upon operating in a cyclic network. The convergence problem of the Min-Sum algorithm is similar to the convergence of the distance-vector routing protocol [37] and it is guaranteed only in acyclic networks. The split horizon rule cannot be applied in a BP algorithm because of the algorithm structure.

Alternative methods remove the loops by replacing the cyclic network with trees. The junction tree is the most common such method and it is based on two properties: (1) every clique of the original graph is contained in some clique of the junction tree;

and (2) for each node of the original graph, the cliques and all the edges containing it form a connected subtree of the junction tree.

Paskin and Guestrin in [38], discuss the need for a special architecture for distributed inference. Their major claim is that an optimized junction tree may reduce the overall communication cost. Additionally, the junction tree is more stable, flexible and does not depend on the network layer. The key disadvantage of the junction tree method is the large communication overhead it requires. Construction and maintenance of such a tree, when the network is continuously changing, incurs considerable overhead by the nodes.

2.3.2 Robustness against Failures

BP has a rapid convergence property, but when too many errors are involved, it is likely that the convergence will be slower and the nodes will converge to an incorrect value. WSNs are exposed to a fairly large amount of failures. Apart from the ordinary failures in WSNs, such as packet errors and loss due to interferences and poor link quality, BP is especially vulnerable to broadcast message-passing, synchronization problems and topology changes during the message transmission.

Broadcast Communication

The wireless medium allows transmission of a single one-to-many message instead of multiple one-to-one messages. However at the same time, it imposes larger constraints on the shared medium, such as collisions and contention.

Message transmission in a broadcast manner reduces the communication volume, but at the same time it is much more sensitive to synchronization problems and is more error-prone than the original algorithm. Instead of receiving a unicast message $m_{ji}(x_i)^t$ from node j, node i is required to extract the relevant information from the broadcast message, by a subtraction of its own information from the previous round's, and by calculating it on its own [39]:

$$m_{ji}(x_i)^t = \min_{x_j}\{\psi_{ij}(x_i, x_j) + m_{j*}(x_j)^t - m_{ij}(x_j)^{t-1}\} \ .$$

The separation of a single update rule into two rules performed by different nodes, increases the potential errors and the synchronization issues that are involved with this method. Consider a situation where node j sends a message $m_{j*}(x_j)^t$. Upon reception of this message by node i, it subtracts its last message $m_{ij}(x_j)^{t-1}$, assuming that it was included in node j's message. In the event of message loss, when $m_{i*}(x_i)^{t-1}$ was not received by node j and was not included in its broadcast message, node i's belief will be wrong and this error may be propagated through the network to other nodes. The nodes may ultimately converge to a common belief, but there is no guarantee that they will converge to the correct value. In the original protocol such a scenario will not occur, since the entire calculation is an atomic operation by a single node. In case of message loss, the nodes may synchronize in a subsequent iteration.

Synchronization

Perfect synchronization among the nodes in WSNs is difficult to achieve in practice, because of clock drifts. Therefore, message-passing would be better off if performed asynchronously, upon message reception from other nodes or upon external events.

The nodes' duty cycle is another factor to consider in the context of asynchronous operation; the message-passing algorithm should take into account cases where nodes wake up only in the middle of a process.

The general BP algorithm enforces some message ordering in each of the message-passing iterations. In the asynchronous method, there are no sequencing constraints and the messages may be transmitted arbitrarily during an iteration. Every node stores the received messages and computes them at the end of the iteration. The lack of synchronization thus introduces the additional cost of storage, and adds even further cost because some messages may be recomputed and retransmitted several times.

Topology Changes

WSNs are usually defined as semi-static networks because the nodes are not mobile in the sense of mobile networks. However, the nodes may nevertheless be repositioned by external factors, such as wind. Most commonly, the topology might change because

the wireless links are not stable and sensor nodes are prone to failure. Therefore, the message-passing algorithm must not assume static topology during its invocation, and scenarios such as link break must be taken into consideration during the message-passing. A link break between some key node and its descendant may harm the convergence of the entire network, as the connectivity may be broken into separate components. Even when the message-passing tree is re-constructed, the synchronization between the nodes may not be restored. Therefore, it is very important to build a stable tree to minimize the effect of topology changes on the message-passing process, while managing such common scenarios.

2.3.3 Scalability

Since scalability is a main concern in WSNs, localized algorithms [15] are used as the building blocks in these networks. These localized algorithms are distributed and only a subset of the nodes participate. The nodes interact with each other only in a restricted vicinity, thus using only a limited amount of communication, computation, and storage resources - all crucial for energy efficiency in WSNs. While this approach seems to promise scalability, the design of such algorithms under the constraints of WSNs is not a trivial undertaking. Following this paradigm and the self-organization property of WSNs [40], the key challenge is to find localized behavior rules that may lead to the desired global property or at least approximate it, when applied by all the nodes.

Although BP is based on local message-passing, it is not inherently limited to a small region, and most of the proposed inference approaches based on BP are not localized. Localizing BP means that the algorithm is required to involve only part of the network and have a constant number of iterations, independent of the network diameter. Consequently, this decreases the number of transmitted messages and the time to deliver them, as well as resulting in low latency, regardless of the size of the network.

2.4 A Practical Approach

In light of the challenges presented in the previous section, this section describes the BP scheme for distributed inference in WSN.

2.4.1 Graphical Model based Spatial Correlation

Our scheme maps a WSN into a graphical model by constructing multiple trees which are based on spatial correlation between the nodes. Each tree includes several sensor nodes that are close to each other. Proximity between the nodes can be obtained using some routing primitives, such as hop count and link quality, and by sharing common neighbors. Spatial correlation between the nodes guarantees that the proximate nodes have related information. Therefore, each tree assumes to hold application data that is correlated.

A construction of multiple trees is not the ideal mapping of the network to a single tree. However, some properties of the original graphical model are preserved by connecting nodes which are highly correlated to the same tree and disconnecting nodes which do not have spatial correlation. Nodes that are far from each other are usually not correlated and so in the graphical model they won't be connected. This approximation is simple and provides an appropriate approximation to the global and centralized approach.

Relying on spatial correlation has advantages similar to the data-centric approach developed for WSNs. Data centricity [15, 41] is a basic term in WSNs, which refers to the greater reliance upon the information content. Concentration of the data content enables design of a more robust application, and outperforms idealized traditional schemes.

We improve our message-passing tree by considering the information content that the nodes hold. Every tree is created on-the-fly using a single message that contains routing information, including parent and hop count, in addition to application-specific information. The fact that the tree is dynamically and locally created without any maintenance requirements, means that it scales and is efficient. The node that

starts the inference process (i.e. with no prior information from its neighbors) operates as a root, by setting its hop count to zero. The nodes that receive the message can either select the sender as a parent, or wait a random short period (limited by a timer) in search of a better candidate. To be selected, a parent must fulfill the routing requirements and reside in close to the other nodes in the tree. High correlation of the data is also necessary. If a node does not find any parent after a given period, it operates as a root.

Each time the node selects a parent, it increments its hop count, This mechanism is used to detect and break cycles in the graph. Once every node is either designated as a root or has a parent, the trees are defined and it is now possible to perform the entire Min-Sum algorithm.

2.4.2 Robustness against Failures

The overall robustness of the algorithm has been presented. It should be noted that it is not possible to totally overcome the algorithm's sensitivity to failures, such as malformed messages and message loss. However, it is possible to reduce the occurrence of failures by using several heuristics:

1. The asynchronous nature of the sensors can be overcome by means of a 'round' field in each message. This field designates the time interval in which messages are grouped together. Messages that arrive too early can be stored in a buffer and messages that arrive too late can be ignored.

2. The 'round' field in each message can also be used for detection and reproduction of message loss. Reproduction of the last message is performed by processing the last message that was received by this node, as if it had been received in the current round. Reproduction of the message keeps the nodes synchronized and enables convergence in later iterations.

3. The Min-Sum algorithm computes cost information by subtracting previous messages, under the assumption that the cost cannot decrease from one round to another. We use a broadcast version of this algorithm, which can cause errors

in the subtraction operation, in that a value greater than the current value may be subtracted, resulting in an (incorrect) negative value. Some of these errors may be detected, because the application assumes that these values fall within some range, so that any deviation from this range will signify an error. The wrong information is ignored in this case.

4. Link breaks between neighbors that are not mapped in the graphical model as a parent and its descendant, do not affect the message-passing process. When the link break affects the graphical model, a node may either select a new parent and try to synchronize with it, or it may become a root.

2.4.3 Scalability

The BP algorithm is not inherently localized and requires global processing of all the groups of nodes in order to achieve a global optimum. Construction of a fully localized algorithm is similar to the general scheme with a few salient differences: (1) The localized algorithm operates locally, and therefore tends to create multiple trees, instead of a single global tree. (2) Flooding control is managed by a 'propagation limit' field in each message, which determines the diameter of the message-passing tree. This field may be set to any desired small value, so only nodes within this vicinity are able to participate in the message-passing process. (3) Scalability is also achieved by defining, a priori, the number of rounds until termination, resulting in a constant message and time overhead, regardless of the size of the network. This limit is necessary not only for reducing the processing and the communication overhead, but also to ensure the termination of the process. This is due to the fact that convergence is not guaranteed in an asynchronous environment with failures and errors.

The BP algorithm that we constructed is based on spatial correlation between the nodes. The natural application which is suitable for the graphical model that was constructed, is clustering.

Tree Construction:

(1) Upon a triggering event or a timer:
  (1.1) If no BP messages with positive *propagationLimit* have been received, start the process as a root by setting *hop* to zero and the localized predefined value of *propagationLimit*;
  (1.2) Otherwise, select the best possible parent and start the process with the parent's *propagationLimit* decreased by one; The parent is defined as 'final' if it meets all the requirements;

(2) Upon reception of a first-round BP message from other nodes:
  (2.1) If already in the message-passing process:
    (2.1.1) If the current parent meets both the routing tree requirement and the data information correlation requirement $\rightarrow$ process the message if it originates from this node's parent or descendant;
    (2.1.2) If the current parent is not final: if the message's sender meets the requirements and also has a positive *propagationLimit*, then replace the parent with the message sender and process its message;
  (2.2) If not in the message-passing process:
    (2.2.1) If the message's sender meets all the requirements, and also has a positive *propagationLimit*, select that node as a parent and start the process with the given *propagationLimit* decreased by one;
    (2.2.2) Otherwise, set a timer to start the process at a later time.

Figure 2.2: Sketch of the Tree Formation

Chapter 3

Clustering with Belief Propagation

This chapter uses the BP scheme presented in Chapter 2 to construct an efficient clustering framework. The chapter begins by giving some background overview. It then provides a formalization of the clustering problem in WSNs and a description of the BP-based algorithm. The algorithm is evaluated by simulations and a final conclusion closes this chapter.

3.1 Clustering in WSNs

Organization of large multi-hop wireless networks into clusters is essential for achieving basic network performance. In WSN, the clustering is primarily characterized by data aggregation by each cluster head, which significantly reduces the traffic cost. The hierarchial model requires two main methods: (1) periodic selection of cluster heads (CHs); and (2) assignment of each node to one or multiple clusters.

Optimal clusters' selection is equivalent to the minimum dominating set problem which is an NP-complete problem. The literature is extremely rich with many approximation algorithms based on several heuristics. The reader is referred to [42] and [43] for a review of previous work.

While most efforts thus far have focused on an energy-efficient clustering scheme [44, 45, 46, 47, 48], the attention to the performance of the multi-hop network was quite limited. An energy-efficiency algorithm may select a few CHs for energy-saving,

but if these CHs do not have good connectivity or if they are not stable, the retransmission and the dropped packets may significantly degrade the network performance and the total energy wasted may end up to be higher. Therefore, taking reliable communication into account is essential for any clustering algorithm which aims to reduce the energy consumption in a network [49].

Moreover, the network lifetime should be measured not only by the time that the first or the last node dies, but also by the period of time that the network is available for providing services and operating appropriately. Since the network is usually dense and many nodes are redundant, the death of a few nodes does not affect the network. Thus, network lifetime is tightly coupled with the network performance.

We uniquely address the clustering problem in multi-hop networks with a special focus on network performance, using the BP algorithm. The main advantage of this method over existing algorithms for clustering [44, 45] is that BP considers not only local properties of a node, such as residual energy or degree, but also takes into account joint characteristics of a group of nodes, such as link quality and topology information. The spatial correlation between proximate nodes that is used in the proposed BP algorithm increases the effectiveness of the formed clusters. Therefore, utilization all available data, while maintaining small constant message and time overhead, leads to considerable increase in network performance and balanced power consumption among the nodes.

3.2 Problem Formulation

We model the sensor network as a directed graph $G = (V, E)$, where V is a set of nodes, where each one is assigned a local unique identifier. E is a set of wireless links connecting two adjacent nodes. Nodes are defined as adjacent if and only if they are within each other's transmission range. The links may be asymmetric. A special node, v_0, is defined to be the base station (BS). The BS is distinguished from other nodes by its unlimited energy supply. The network is multi-hop, where nodes closer to the BS relay traffic of other remote nodes and probably consumes much more energy [47]. There are no assumptions about the distribution of the nodes,

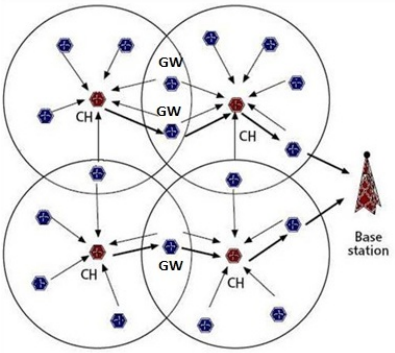


Figure 3.1: Clustered network with gateways.

their homogeneity, location information etc.

The challenge of a clustering scheme is to efficiently form and maintain connected disjointed groups of nodes in a local and distributed manner. Each group contains a single leader and several ordinary nodes.

The connectivity requirement may be achieved using one of two basic methodologies: (1) an adaptive transmit power, where the CH increases its transmission power to reach the next CH; (2) an assignment of a set of nodes that are connected by several CHs to be gateway nodes. These gateways nodes are responsible for passing traffic among clusters (shown in Figure 3.1). In this work, the second approach is used. This approach is more general because it does not assume any distribution of the nodes and it also takes into consideration interferences in the area.

An efficient scheme is used to select CHs that: (1) minimize the total transmission power aggregated over all nodes in the selected path; (2) balance the load among the nodes to prolong the network lifetime. These two requirements may contradict; e.g. a long path that consumes more energy than a short path may be selected in order to avoid battery depletion at some nodes. The network performance itself is obtained, in part, by the first requirement, where minimizing the total transmission cost results in a decrease of retransmissions as well as the data transmission time.

In order to achieve a scalable and feasible framework, the overhead of the scheme should have a constant message and time complexity per node, with low maintenance cost. Additionally, it should work well under constraints such as topology changes, asynchronous environment, failures and duty cycle.

3.2.1 Cost Metrics

Basic metrics for energy-efficient and reliable communication are formulated in [50] for minimum energy path and maximum lifetime. Their analysis shows that an incorporation of the link error rates is required for reliable packet delivery, in both constant-power and variable-power scenarios. Using a similar method, two cost functions are defined. These cost functions consider residual energy, degree, topology and link quality, distance from BS (in terms of hops) and overall transmission cost, as shown in the following.

A self cost of a potential CH is denoted by $C_i = \frac{E_i}{B_i}$, which is basically defined by the expected energy consumption in a period E_i and it's residual battery power B_i. The expected energy consumption is an estimation of the power used in the routing if that node becomes a CH. The estimation is based on the network topology: the degree of the node determines the expected reception and transmission; the distance from the BS in terms of hop count estimates the further transmission cost to the BS.

Transmission cost among two nodes or along a path is a function of the radio power level and the number of transmitted bits. Previous work [51, 52] has shown that the overall transmission cost cannot be estimated by the distance between the nodes, e.g. because of interferences, nor can be estimated by the received signal strength indicator (RSSI), due to in-correlation between low RSSI and reception rate. Link quality can evaluate the expected number of transmissions along the path. Each node estimates the quality of the links by observing packet success and loss events. Accordingly, the transmission cost between two neighbors $C_{ij} = \frac{E_{ij}}{B_i}$ is defined as a function of the energy consumption over the link E_{ij} and the remaining battery power of the transmitting node B_i.

3.3 Algorithm Description

Let x_i be a CH candidate of node i, i.e. $x_i = i$ or $x_i \in N(i)$ and x_i has a valid route to the BS and appropriate link quality.

We define $\psi_i(x_i)$ to be a local cost function of connecting node i to CH x_i.

$$\psi_i(x_i) = \begin{cases} C_i & for\ x_i = i \\ C_{ij} & for\ each\ x_i = j \in N_i \ . \end{cases}$$

$\psi_{ij}(x_i, x_j)$ represents the constraints between two neighbors i and j to eliminate improper assignment of CH association. The constraints are: (1) two neighbors cannot be both CHs; (2) a node can select another node to be its CH only if that node announces that it is a CH.

$$\psi_{ij}(x_i, x_j) = \begin{cases} \infty & one\ of\ the\ constraints\ is\ applied \\ 0 & otherwise \end{cases}$$

Cluster selection is possible at each node after a period of initialization, when a route to the BS is constructed. The process is asynchronously triggered by two events: (1) when a regular node does not find a CH among its neighbors, e.g. because of topology changes; and (2) periodically, by a CH, to balance the power among the nodes in a local area. The second event also ensures that the number of CHs will not be too large, by preventing a CH from assuming that role if it is not re-selected.

The message passing algorithm is performed on a tree structure, which is a sufficient condition for convergence. The algorithm is executed in a restricted region of a 1-hop neighborhood, and as a result, it requires a constant number of messages. It stabilizes when the entire network is not affected by local changes anymore. The tree is a subtree of the general routing tree that is used in the network. In the first event, once a node triggers a clustering process because of no CH, it announces itself as a temporary CH and its 1-hop neighbors, which get its message and find it as an appropriate CH, selects it as a parent and performs the message passing on the resulting 1-hop tree. In the second event, the node is already a CH, so the message passing tree is already constructed, where all the children of that node participate in the message passing.

Each node i starts the process by broadcasting the message $m_{i*}(x_i)^0$. This message contains its cost for being a CH (infinite if it is not a valid CH) and the cost to connect other CH candidates among its neighbors. These costs are transmitted as 16-bit integer numbers together with 16-bits of identification.

The rest of the packet processing is performed according to the MS algorithm described above, where unordered messages are stored in a buffer until computation. The timer between the rounds is large enough to support asynchronous operation, but not too large, for not to adversely impact effective operation. Topology changes during the message passing are taken into consideration as follows: (1) Cost of new neighbors is not added in the middle of the message passing operation; (2) Node who loses its parent during the message passing cannot converge with its new parent, so all its messages are ignored. The node should wait until the end of the process to find out a new CH; (3) Link breaks are marked by updating the joint cost to be infinite. A node determines which of its neighbors are in its routing subtree by inspection the messages of its parent and its descendants. A node discards cost information of nodes that are not in its subtree, because it does not have complete information about them. Messages with errors or those which are not synchronized with the messages of the node, are discarded as well. One round before termination, a node calculates the belief about its final state - a CH or an ordinary node, and attaches the appropriate announcement to the message. After the last round a node operates according to its announcement; If it has previously announced itself as a CH, it becomes a CH. Otherwise, it joins the cluster that minimizes the overall cost, according to the information it holds. In case of errors or a convergence problem, it is possible that no node would declare itself as a CH. In such a scenario, nodes that do not have any alternative CH in their area start the clustering process again.

In contrast to the cost messages, which are propagated over the routing tree to avoid loops, the decision of a selected cluster is made by the information spread in the entire 1-hop neighborhoods, i.e. a node can select a CH that does not appear in its current subtree. Each node updates its clusters map according to all the broadcast messages it gets.

Once the clustering process is done, the routing tree is changed, where CHs operate as parents of the nodes who join them. Using the gateway approach to connect two clusters, a CH may choose a regular node to be its parent, if it does not have any CH that could operate as its parent. The hop metric is used to detect and avoid cycles, so after the process there is a new routing tree.

Main

(1) If CH and timer expires or if ordinary node with no CH
  (1.1) Start clustering process with propagation limit of 1;
(2) Upon reception a first-round BP message from parent or from CH candidate
  and when the propagation limit is 1
  (2.1) Update your parent to be the sender node for the message passing;
  (2.2) Start clustering process with propagation limit of 0;

Clustering Process

(1) Compute local cost function and joint cost function of all the neighbors;
(2) Run the MS algorithm with the following rules:
  (2.1) Unordered messages will be stored in a buffer until computation;
  (2.2) Upon topology changes update the cost;
  (2.3) Messages with errors or synchronization problems are discarded;
(3) One round before termination attach the belief about final state to the message;
(4) Ending steps:
  (4.1) Set the power level according to the final state and update timers;
  (4.2) Select a parent: if ordinary node, select CH that minimizes the cost;
    if CH, select other CH if possible, otherwise choose an ordinary node
    as a gateway.

Figure 3.2: Sketch of the Algorithm

3.3.1 Convergence Time

BP has a fast convergence property, but when too many errors are involved, it is likely that the convergence will be more slow and go into a wrong value. WSN are exposed to a large amount of communication and node failures, so the convergence to a correct state is not guaranteed. Therefore, in order to avoid impact of the physical and the MAC layers as well as other environmental factors, we limit by design the number of rounds to be a predefined small fixed value prior to termination. In an ideal environment, the convergence of the algorithm to a common belief, not including the CH announcement, is 2 rounds, equal to the diameter of the 1-hop vicinity graph. Actually, the predefined round number was set to 5. This value is robust against some synchronization and packet loss, and it is sufficient in most of the cases to reach a convergence via three steps: detection of the nodes in the routing tree for correct

cost calculations, computation of the belief based on cost functions, and publication of the CH announcement. This number of rounds is very small in compared to other schemes and is not affected by the network size, therefore providing a scalable solution in large networks. Moreover, the limitation on the number of messages means low delay and small message overhead.

3.4 Performance Evaluation

To evaluate the performance of clustering using BP, it has been compared with the clustering process of HEED [45], in a network model that uses gateway nodes to connect between the clusters, when two CHs cannot communicate directly.

In HEED, a node initially sets its probability to become the CH according to its residual energy. During each iteration, a node arbitrates among the CHs announcements it has received to select the lowest cost CH. If it has not received any announcements, it elects itself to become a CH with the probability it has. If successful, it sends an announcement indicating its willingness to become CH. The node then doubles its probability, waits for a short iteration interval, and begins the next iteration. A node stops this process one iteration after its probability reaches the value of 1. Simulation results have shown that HEED is effective in prolonging the network lifetime and in supporting scalable data aggregation.

3.4.1 Simulation Model

TOSSIM, TinyOS simulator [53], was used for the analysis of the clustering algorithm. Link Estimation and Parent Selection (LEPS) [52] was used as the routing protocol in the multi-hop network. In this method, each node monitors all traffic received within the single hop range, including route updates from neighbors. Using shortest path heuristic, it manages the nearest available neighbors and decides the next hop. The Surge application was used for data aggregation, where every node periodically takes light sensor readings and sends them over the network to the BS. The simulator provides an environment which includes realistic properties of a network, like interferences and collisions, asymmetric links, changes in the link quality, nodes death and

failure etc.

Evaluation of the communication cost, as well as the estimation of the remaining energy, were done based on the power information about Berkeley Mica2 mote [54] and using the credit point system, proposed by [55]. In this system, every node is assigned some number of points that reflect its residual energy. Each packet reception or transmission reduces points from the node, based on the packet size and the transmission power level.

Every plot was taken as an average of 27 different runs. In all the experiments, 250 nodes including a single BS were run. The simulated time was 20000 seconds, to observe the network in a stable state until it collapses when the major of the nodes die.

Every node starts with a random residual energy, ranges from 250 to 500 thousand points. The power level of a regular node was -20 dBm and the power level of a CH was -13 dBm. A timer of 540 seconds was set for periodic cluster selection triggered by each CH or by each node in BP and HEED, accordingly, and a timer of 11 seconds was used between the rounds of the message passing. Both the power levels and the timers are the default parameters used by HEED in TinyOS. We adapted the transmission rate and the aggregation rate to the network size, so the transmission rate by the application was increased to 6144 milliseconds. Every CH that receives the packets aggregates them and transmits them every 3 minutes. The other parameters are taken to be the defaults defined in TOSSIM.

3.4.2 Network Performance

We first study the network performance of the two algorithms, in terms of data packets received by the BS. Each node constantly transmits data points to its CH, which aggregates all points into a single packet and forwards them toward the BS.

As one can see in Figure 3.3, clustering with BP achieves more than 40% higher throughput than HEED, where the data points received by the BS are significantly greater. This higher throughput is expressed by both data collection rate and time.

The trend of the data rate during the network lifetime is shown in Figure 3.4. In Figure 3.4(a), there is an increase in the data rate over time, both because the network

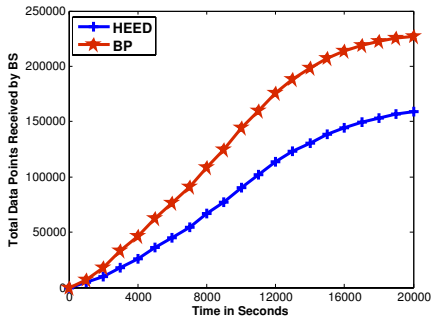
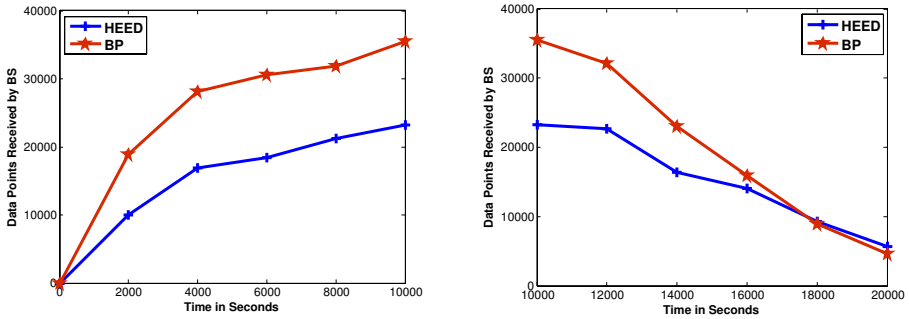


Figure 3.3: Data collection time



(a) Data collection rate (b) Deterioration of data collection rate

Figure 3.4: Data collection rate during the network lifetime

becomes more stable and also because nodes start to die, so the network experiences less interferences. The number of live nodes in the system decreases to about 150 nodes at time 10000, but the network is still well connected and only the nodes' redundancy is removed. From this time, the nodes die quickly, so the connectivity of the network and its coverage rapidly decrease. Since the data rate of BP is larger than HEED, the deterioration is steeper.

The advantage of BP can be explained by several network parameters, which are all a result of the fact that BP selects CH better than HEED. The non-optimized routing of HEED can be shown by the average hop count of HEED, as presented in Figure 3.5 which is larger than BP. This means that the number of transmissions in the network may increase, so the number of interferences and the dropped packets increase as well.

Better deployment and network stability may be another reason for the advantage

Time (s)	HEED	BP
2000	5.11	3.04
4000	4.11	2.83
6000	3.61	2.73
8000	3.93	2.67
10000	3.88	2.55
12000	3.39	2.51
14000	3.81	2.44
16000	3.58	2.42
18000	3.46	2.38
20000	3.51	2.36

Figure 3.5: Average hop count

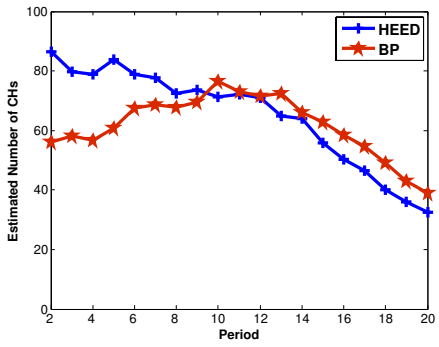


Figure 3.6: Estimated number of CHs

of BP over HEED. The estimated number of CHs in the system during each period of time is presented in Figure 3.6. Each period is about 540 seconds, with a single periodic clustering process. The figure shows the network state from the beginning with 250 nodes, until about 150 nodes are left, at which point (in periods 11-13) nodes start to die. At the beginning, BP has less CHs in the system which implies better aggregation, less transmission cost and interferences. Once nodes start to die, the number of CHs selected by BP in the system increases proportionally to the number of nodes that are alive and to the number of CHs which are selected by HEED. The intersection of BP and HEED in periods 11 and 12 is a result of the decrease in the number of CHs in HEED and the increased number of CHs, in proportion to the number of alive nodes, by BP. The increased number of CHs achieves better coverage and deployment and improves the network connectivity compared to HEED. The network with BP performs better even under conditions of topology changes, so as a result, less clustering processes are performed and less route failures exist, as it shown in Figure 3.7 and Figure 3.8.

The number of clustering processes that are triggered in HEED increases in period 11, which can be explained by the fact that nodes start to die, and consequently some of the nodes lose their CHs. Nonetheless, with the exception of that increase, during most of the duration, the number of clustering processes that are triggered is quite similar, even during the periods when there are much fewer nodes alive. This means that the network has proportionally more clustering processes and that it is not in a

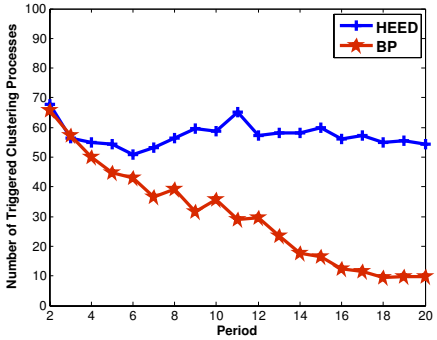


Figure 3.7: Triggered clustering processes

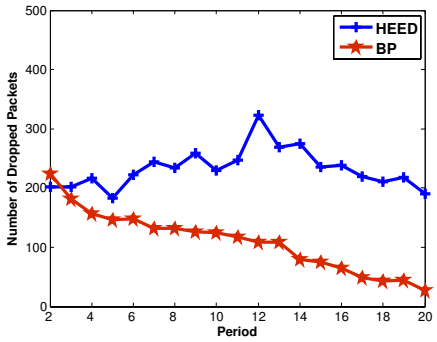


Figure 3.8: Dropped packets

stable state. On the other hand, the number of clustering processes that are triggered in the BP scheme decreases over time, which shows better stability even when nodes die. The number of packets that are dropped because of no route, correlates to number of clustering processes that are triggered because of no CHs, and presents the same trend.

It is important to note that no retransmission is done in the simulation. When retransmission is performed, HEED is expected to perform much worse than BP, since retransmission means more interferences and more energy consumption.

3.4.3 Clustering Overhead

Although BP and HEED have both a constant and consistent number of rounds in the clustering process, BP suffers from more overhead during the clustering process. This is because the messages of BP are larger than HEED. BP messages, at the extreme, might reach up to 74 Bytes (17 cost entries with identification of total 4 Bytes plus header of about 6 Bytes), while HEED message have size of 29 Bytes at most. In fact, BP messages are usually not that long, and do not reach that limit, but still the messages are longer than HEED, so the transmission cost is higher.

Figure 3.9 shows that at the start of the simulation, the overhead of BP is about double the HEED overhead. Later, when the network becomes more stable, BP performs less re-clustering than HEED. HEED performs more re-clustering processes because nodes die, so this difference significantly decreases.

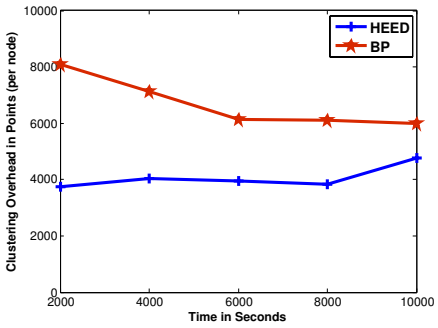


Figure 3.9: Clustering process overhead

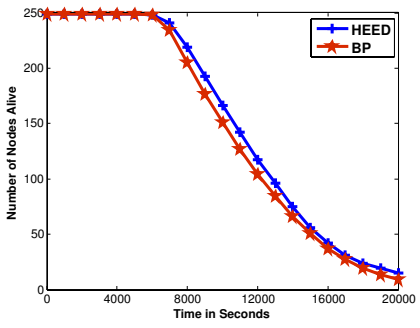


Figure 3.10: Network lifetime

3.4.4 Energy Characteristics

Network Lifetime

BP achieves better network performance and reduces the transmission cost as well. However, the network lifetime, measured by the number of live nodes of BP and HEED are quite similar, with a small advantage of HEED over BP, as presented in Figure 3.10. This results from the fact that the total number of packets that are forwarded in the network is significantly greater in BP than HEED. This implies a higher total transmission cost. As a result of the CHs' selection, BP pays for transmission of a single packet much less than HEED pays, but over the network lifetime the overall transmission cost is similar.

When measuring the network lifetime as the time that the network is available for providing services, we can see in Figure 3.4(b) that BP succeeds in achieving better performance than HEED, until very close to the end. Only from time 18000, HEED has a slight advantage in the throughput, but this has no real meaning because there are about 20 nodes in the network and anyway the network does not operate appropriately. Therefore, from service availability point of view, BP has better overall network connectivity than HEED and thus better network lifetime.

Power and Load Balancing

In multi-hop communication, the nodes closest to the BS usually tend to be burdened with a heavy relay traffic load and to die first. This is the hot-spot problem and many clustering algorithms suffer from it. To verify that this problem does not occur in

Average hop count	Average number of nodes	Average initial energy (points)	Avgerage lifetime (seconds)
1	13	374933.48	11673.37
1.5	14	394017.97	12727.69
2	57	388017.87	13047.62
2.5	81	388938.62	12546.50
3	76	335469.51	9587.23
3.5	8	292071.86	7800.46

Figure 3.11: Energy information about the nodes in BP

BP, we analyze the energy characteristics of the nodes based on their distance from the BS.

A node with some physical distance from the BS can have different hop distances over time. For example, a node with distance 1.5 from the BS, can sometimes be connected directly to the BS and sometimes connected via a CH. The different hop count is mostly a result of link quality, which is affected by many network parameters.

We explore the general concepts that arise from Figure 3.11 and not from the specific values, since the nodes start with a random initial energy, which definitely affects the network lifetime, even when power balancing takes place.

As shown, both nodes that are very close to the BS, with distance 1-1.5, and more remote nodes, with distance 2-2.5 (that start with comparable initial energy), have similar lifetimes. This means that the BP method succeeds in achieving power-balancing in the core of the network and it does not suffer from the hot-spot problem.

It is interesting to see that more remote nodes (distance 3-3.5) not only start with significantly less residual energy, but their lifetime is shorter. The reason for the initial low energy is that nodes with low residual energy usually would not be selected as CH, and this means that their average distance is larger because they are constantly connected to a CH one hop farther away. The explanation for the short lifetime of those nodes, in general, is that they are located at the edge of the network. Nodes at the edge, usually have less neighbors and less chance for having CHs around, so they experience more topology changes and usually perform further frequent clustering processes, which result in more overhead as well. This overhead has a considerable effect on the nodes' lifetime.

3.5 Conclusion

In Chapter 2 we introduced an efficient and practical BP framework for distributed inference based on spatial correlation between the nodes. The BP framework that has been proposed is a feasible and realistic inference scheme, and can be effective for many other applications. The special attention to energy constrains and the fact that no assumptions were made regarding the network topology or size, differs this framework from other schemes for WSN that are based on BP, and makes it more practical and scalable to large networks with their dynamics.

We used this inference scheme later in Chapter 3 to present a novel distributed inference scheme, for efficient clustering in multi-hop WSN. This inference scheme selects CHs that minimize the overall transmission cost and at the same time balances the power among the nodes, achieving a longer network lifetime. Utilization of all available information, is more optimal than current solutions, and leads to a significant improvement in the network performance.

Using simulations, we show that the BP algorithm succeeds in improving the data transmission time and rate, so at the same network lifetime as the HEED scheme, the overall throughput of BP is increased.

Chapter 4

Tracking in Body Sensor Networks

This chapter presents a new RSSI-based tracking system that is designed for small networks, such as Body Sensor Networks (BSNs). BSNs are a specific category of WSNs, aim at operating in a pervasive manner for on-body applications. Much of the theory relating to general WSNs is also applicable to BSNs, but their focus is sometimes different. In BSNs only a few proximate sensor nodes which can be easily charged are deployed, meaning that energy efficiency and scalability are only a secondary objective. Reliability and data accuracy are key design parameters in these networks as they are essential in health-care analysis. This chapter shows that a tracking system that is designed especially for BSN and takes the advantages of this network, can lead to improved reliability and data accuracy.

The chapter begins with a brief background on tracking in general WSNs. It then describes the motivation to apply tracking in BSNs using Received Signal Strength Indication (RSSI), and sketches the stages in building that tracking system. With reference to the tracking system itself, we first describe the system model and the problem formulation. The data processing and the calibration process are also described. The experimental setup and results are presented. The chapter ends with a conclusion, and discussion about future work.

4.1 Introduction

Precise motion and location tracking in an indoor environment using WSNs plays an important role in sports, medicine, and many other fields. In sports, the tracking system can give information about the movement of different body parts during activity. In medicine, a precise tracking system can be deployed at the clinic or at patients' homes for analyzing abnormalities in gait or in motion, and for rehabilitation of injured persons [56, 57]. Applications in these fields estimate body parts' displacements over time and use it to classify motion patterns with different pathologies. The performance of such applications depends on the accuracy of the tracking system. Consequently, much effort has been taken recently to improve the tracking resolution to the scale of centimeters.

Systems designed for indoor object tracking based on WSNs [58, 59, 60, 61] can be classified by technology, measurement metrics, and processing methods. The main technologies are inertial sensors technology, infra-red technology, ultrasound technology and radio technologies. Common measurement metrics are: Angle-of-Arrival, Time-of-Arrival, and Received Signal Strength Indicator. The main processing methods are based either on statistical approaches or on geometrical techniques of triangulation or trilateration.

WSN tracking systems that utilize radio technology can be based on Received Signal Strength Indicator (RSSI) measurements. RSSI is a measurement of the signal power on a radio link [62] and can be used for localization, link quality estimation and power control. It is part of the IEEE 802.11 protocol family and the 802.15.4 standards and it is supported by most of the existing transceiver chipsets with no extra cost. RSSI measurements are very simple to use and have a lower power consumption compared to other methods. Therefore, RSSI is widely used in various applications, including tracking.

RSSI-based tracking systems usually consist of several mobile nodes and a set of static nodes, referred to as anchor nodes. The tracking algorithm tries to continuously estimate the mobile nodes' location from the RSSI measurements [63, 64, 65, 66]. RSSI-based tracking algorithms [67] are usually composed of two-steps. In the first

step, RSSI measurements are used to estimate the range between pairs of nodes using known channel model characteristics or by calibration methods used off-line. The calibration methods either find channel model parameters or produce a conversion table among RSSI measurements and distances, e.g., [68, 69, 70]. In the second step, statistical or geometrical methods are applied to obtain the instant location from the range estimation [71].

The design of RSSI-based tracking systems has numerous challenges that are related to the nature of RSSI. First, RSSI measurements are highly affected by the variation in the wireless medium [72]. Reflections of the transmitted signal from walls or from scatterers in the medium result in severe multi-path interference at the received signal. The interference causes range estimation errors that can lead to large tracking errors. Second, an accurate conversion of RSSI measurements to range requires a precise calibration process, which reflects the channel model. In many cases, changes in the medium during tracking with a calibration process that does not reflect the dynamic channel can cause severe conversion errors. Third, RSSI measurements are very sensitive to interference at relatively high distances. The RSSI measurements attenuate in distance with a power decay factor [73]. This implies that when the distance between the nodes is not proximate, i.e. in range of a few meters, the range approximation is sensitive to small interference, which can result in high range estimations errors.

The sensitivity of RSSI measurements to the medium and to distance can be mitigated using a priori knowledge. Prior knowledge about the target application or the relevant environment settings leads to improved tracking accuracy. For example, prior information about the environment dimensions enables an optimized transmission power setting of the nodes and a better conversion of RSSI measurements to range, as non-relevant range can be truncated. Information about the range of possible velocities enables an adjustment of the transmission rate. An appropriate transmission rate implies effective use of the correlation among consecutive RSSI measurements in order to reduce noisy measurements. Existing RSSI-based tracking systems usually do not exploit such a priori knowledge and so their accuracy is measurable in the scale of a meter [74, 75, 71, 76].

In this chapter we present a novel idea of a RSSI-based tracking system that exploits existing a priori knowledge. The tracking system is tailored to the area of Body Sensor Networks (BSNs) [12] and can be used in many sport and health-care applications. In BSN applications, the set of sensor nodes is deployed over a human body in proximate Line-of-Site (LOS) conditions and in a distance of less than one meter from each other. The position estimation and tracking are performed relative to the anchor nodes. The anchor nodes can be either placed on reference locations in the room or attached to the relatively static torso. The mobile nodes can be attached to any other body part, such as the legs or arms, for relative motion analysis. The sensor nodes can be charged every few days to enable continuous tracking during daily life activities.

Applying the a priori knowledge about the target application and the relevant environment settings leads to an improved tracking accuracy. The proximate LOS condition, i.e. where there are no massive scatterers between the nodes, results in a relatively static channel which promises that the calibration process will be relevant to the real-time scenarios. In close proximity, where the distance between the nodes is small, e.g. around a meter, we can adopt an adequate transmission power level so the RSSI dynamic range is higher and the conversion of RSSI measurements to range is more accurate. With an adequate transmission rate, we can further exploit the knowledge about the continuity of the movement. In a continuous movement, there is a temporal correlation between consecutive RSSI measurements, which refer to proximate locations of the mobile sensor node. Using this correlation, channel interference distortion can be compensated.

The proposed RSSI-based tracking first derives a constrained Minimum Mean Square Error (MMSE) criterion for tracking based on the RSSI measurements. The constraints are tailored to the a priori knowledge about the system settings. In order to solve the criterion, a transmission power level that maximizes the RSSI dynamic range is used. An advance calibration scheme that uses prior knowledge about the channel conditions and the environment dimension is applied. Effective processing of the RSSI measurements that exploits their temporal correlation are then performed. The processing stage includes pre-processing of the RSSI measurements, estimation

of the range between each pair of nodes, mobile node location estimation based on the range estimation, and advanced filtering based on the constraint is applied to the location estimations. The localization algorithm used in this system tries to derive the mobile node's location from the RSSI measurements using maximum likelihood estimation that is based on geometrical properties. This implementation might not be optimal in MMSE sense for all channel realizations, but it is simpler than other statistical algorithms, such as Kalman filter [77] or Particle filter [78]. This algorithm clearly demonstrates the exploitation of the constraint to improve tracking accuracy.

A series of experiments were conducted in a real-world indoor environment for performance evaluation of a tracking system. Two anchor nodes were deployed in known locations and a single mobile node was moving continuously in a range of less than a meter from the other two nodes. In a first set of experiments, the mobile node was attached to a toy car that was moving along different pre-determined tracks. The first set evaluates the proposed processing methods and algorithms. The properties of the system are demonstrated using manual measurements as a reference. In a second set, the mobile node was attached to a human hand that was moving randomly on a 2D plane. The second experiment set presents the capability of the system to give location estimations in real environment with dynamic channel conditions. Body reflections affect the RSSI measurements, and therefore the location estimation is more challenging. The hand was arbitrarily chosen, and the experiments can be extended in the future to other body parts and to 3D plane. Experiment results show that the accuracy of our proposed tracking scheme in terms of location, mean error, and standard deviation is in the range of a few centimeters.

4.2 System Model

4.2.1 System Description

The basic system consists of a single mobile node with a location of $L_0 = (x_0, y_0, z_0)$ in Cartesian coordinates and N static nodes, referred to as anchor nodes, placed at $L_1 = (x_1, y_1, z_1), L_2 = (x_2, y_2, z_2), .., L_N = (x_N, y_N, z_N)$, respectively. The goal of our work is to continuously estimate the mobile node's location $L_0^t = (x_0^t, y_0^t, z_0^t)$ at any given

time t. The mobile node transmits a data packet with a known transmission power to the anchor nodes every T ms. The anchor nodes, located in the transmission range of the mobile node, calculate the received power values $P^t_{r_1\,dBm}, P^t_{r_2\,dBm}, .., P^t_{r_N\,dBm}$. Each transmitted packet is labeled with a time stamp which is used for recovering possible packet loss. No synchronization is assumed among the nodes.

4.2.2 Wireless Channel Model

The most common wireless channel model is the channel path-loss model [79]. The received power for anchor node i at time t in channel path-loss model is:

$$P^t_{r_i\,dBm} = P_{tdBm} + G_{r_i} G_t + A - q10\log_{10} d^t_i + \alpha^t \,, \tag{4.1}$$

where G_{r_i}, G_t are the receive and transmit antennas gains and A is a constant that is a function of the transmission wave length [80]; q is the channel exponent that varies between 2 (free space) and 5 (indoor with many scatterers); d^t_i is the distance between the anchor node i and the mobile node; and α^t is additive noise that accounts for the random effect of multi-path and for channel model inaccuracy.

The interference factor α^t, which accounts for the random effect of shadowing and for channel model inaccuracy, is sometimes not stationary due to the random effect of multi-paths and shadowing. Some channel models divide the interference into two components: a Gaussian distributed random variable, with zero mean and standard deviation σ, and a component that reflects the strong reflections from walls [81]. The component of the reflection factor can sometimes be estimated and partially filtered over time with advanced processing [82]. With many scatterers in the medium and no dominant reflectors or in LOS conditions, α^t can be modeled just by its Gaussian part, with zero mean and standard deviation σ.

Each antenna has a distinct radiation pattern. The antenna can radiate greater power in one direction compared to its other directions. An antenna that radiates power uniformly in one plane with a directive pattern shape in a perpendicular plane is called an omnidirectional antenna [83]. In this work we will use omnidirectional antennas for the anchor and mobile nodes.

For the case of omnidirectional antennas and a constant transmit power, the received power in (4.1) can be written as a function of only three parameters:

$$P^t_{r_i dBm} = B_i - q10\log_{10} d^t_i + \alpha^t \ , \tag{4.2}$$

where B_i is the power offset that consists of the transmit power P_t, the receive and transmit antenna gain factors G_{r_i}, G_t, and the system constant A.

4.2.3 Problem Formulation

In order to track the mobile node, we need to continuously estimate the location of the mobile node, using the set of N power measurements, the location of the mobile node. The Minimum Mean Square Error (MMSE) optimal transformation of the measurement matrix Pr can be obtained by solving the following criterion:

$$\hat{f} = argmin_f E(L_0 - f(Pr))^2 \tag{4.3}$$

$$s.t.\ |L_0^{t+1} - L_0^t| < \delta \ ,$$

where L_0 consists of M consecutive coordinates of the mobile node; M refers to the size of a frame; Pr is the $N \times M$ power measurement matrix that contains the N anchor nodes power measurements over M measurements; f is a transformation of the power measurements to location; $E[\cdot]$ is the expected value over all stochastic sources; and δ is a bound on the difference between consecutive location estimations which are a function of transmission rate and mobile node velocity. With high RSSI transmission rate or low mobile node velocity, consecutive RSSI measurements imply proximate locations.

4.3 Methods

The goal of the processing methods is to estimate from the RSSI measurements the instant mobile node location. Our RSSI-based tracking algorithm is composed of two phases: an offline phase, which includes finding the optimal transmit power and

calibration process, and a tracking phase in which we track the mobile node location in real time.

In the offline phase, we perform a set of offline tests of the system that attempt to match system parameters to the experiment's environment by using different RSSI measurements at predetermined locations. First, we find the optimal transmission power for the environment. Then we perform a calibration process in which we find the channel model parameters for translating the RSSI measurements to range estimations.

The tracking phase is comprised of the following four stages:

1. Pre-processing of the RSSI measurements to obtain the received power. This stage includes conversion of the RSSI measurements to power measurements, interpolation of missing samples, and filtering out the channel noise.

2. Range estimation between the mobile node and each anchor node according to the power measurements and calibration.

3. Combination of the information from all the nodes and MMSE estimation of the mobile node's location.

4. Filtering out estimation errors with statistical methods based on the continuity of the mobile node's movement.

4.3.1 Transmit Power Selection

Distance estimation accuracy is highly affected by the transmission power level [84]. Insufficient transmission power may lead to high packet loss while high transmission power can lead to saturation of the RSSI measurements and distort the distance estimation. As an increase in the RSSI measurements dynamic range for a given environment can improve the distance estimation accuracy [73], we would like to maximize the dynamic range of the RSSI measurements. Adaptive transmission power has been investigated in several papers, such as [85] and was considered in [86] and [87] in the context of distance estimation. A transmit-power adjustment is feasible for most existing off-the-shelf transceivers in WSNs, e.g. CC2420 [88].

We suggest a different, simpler method to find the optimal transmit power level. The new method can be incorporated into the calibration process and therefore does not require any additional overhead. In our method, the mobile node is placed at different locations along the indoor tracking environment. For each location, the mobile node transmits different packets with different transmission power levels, and the corresponding RSSI measurements are stored in a table. RSSI measurements with high packet loss, e.g. more than 10 percent, are excluded from the table. For each transmission power level between each pair of nodes, we calculate the RSSI quality and the RSSI dynamic range. RSSI quality can be estimated by the inverse of RSSI standard deviation, $\sigma(Pt)$. The criterion to obtain the optimal transmit power level which tries to maximize RSSI dynamic range on both anchor nodes under the constraints of low packet loss and small RSSI standard deviation is:

$$\tilde{P}_t = argmax_{P_t}(\Delta P_r(P_t)), \tag{4.4}$$

$$\text{s.t. } \sigma(P_t) \leq T_{\sigma_{RSSI}} \text{ and } PL \leq T_{PL},$$

In this equation, $\Delta P_r(P_t)$ is the RSSI dynamic range of the transmitted power P_t in the environment dimensions and is calculated by $\Delta P_r(P_t) = P_{max}(Pt) - P_{min}(P_t)$. $T_{\sigma_{RSSI}}$ is a threshold over σ_{RSSI} and has typical values between 0.5 - 2 dBm. T_{PL} is a threshold over packet loss and has typical values of 10%.

The optimal power level can be found by numerical methods such as a binary search. In a binary search, the range of transmission power is divided at each iteration into two sections. For each interval the two boundary transmission power levels are calculated according to (4.4). A transmission power level with high packet loss rate can be excluded from the search.

4.3.2 Calibration

Calibration is required to find the system parameters for translating the power levels' measurements between each pair of sensor nodes to the corresponding distance. Calibration schemes [68] often use an a priori knowledge about the channel or perform offline measurements at a grid of points in the area of interest. The result of

the calibration process is either a mapping table between RSSI measurements and distances or an estimation of the channel propagation model parameters. Inaccurate calibration or using a channel propagation model that does not reflect the channel may lead to range and tracking estimation errors.

For multi-dimensional location tracking, multiple sensors deployed in different planes are necessary. The calibration process should be performed between the mobile node and each of the other anchor nodes. In the case where the sensor nodes' antennas are not fully isotropic, the RSSI measurements can vary in different planes. As a result, the calibration is not uniform in space and should be performed in a multidimensional grid [72]. The calibration procedure usually requires human intervention and is sometimes tedious and inaccurate. In order to avoid this procedure [89, 90], self-calibration methods using other sensors' data or by an advanced online processing of all node information were proposed recently.

In this work, we use a calibration scheme that is based on log fitting of the RSSI measurements and approximating the power offset and the channel exponent using the a priori knowledge of the environment physical dimensions and the range of the channel exponent values. The range of channel exponent values can be determined by the channel conditions in the environment. This method is an extension to the methods of logarithmic fitting in [25]. This calibration method can provide a relatively accurate distance estimation with few RSSI measurements.

4.3.3 Processing Stages

Pre-processing of RSSI Measurements

Pre-processing of the RSSI measurements is necessary for synchronizing the RSSI measurements obtained at the different anchor nodes, to recover lost packets, and to exclude the channel noise from the measurements.

Synchronization between the measurements can be obtained by two key methods. The first one is clock synchronization of all nodes in the system. In clock synchronization e.g. [91, 92] all measurements are measured at the same time. Clock

synchronization algorithms in WSN have overhead in computation and battery consumption and suffer from clock drift. This may affect the synchronization. The second synchronization method is obtained by a global time reference. In our system, it can be implemented by using data frames and time-stamps of the mobile node (which is the global time reference in each anchor node). The synchronization is performed at the beginning of each frame. The frame size is in the range of one second. The delay induced to the system is measured in the scale of the frame size. This delay is acceptable for most tracking applications.

After the synchronization of the different nodes, lost packets can be recovered using interpolation that relies on the temporal correlation obtained by the high sampling rate and the continuous movement of the mobile node [93]. When the packet loss is caused by a long burst due to inefficient power or shadowing, the interpolation will only partially recover the signal distortion.

Following the nodes frame synchronization and packet loss recovery, we can filter the channel interference with Low Pass Filter (LPF). The effectiveness of the LPF derives from the high temporal correlation between the nodes. The signal after preprocessing is:

$$\tilde{P}^t_{r_i} = P^t_{r_i} * h \ , \tag{4.5}$$

where $*$ denotes the convolution operation, and h is a LPF that smoothes the additive noise and eliminates the fast-fading. The LPF filter can also be implemented by Auto Regression Moving Average or with Kalman smoother [77].

Range Estimation

A continuous estimation of the distance between the mobile node and an anchor node i can be derived analytically from the filtered received power according to (4.1):

$$\tilde{d}^t_i = 10^{\frac{P_t + A - \tilde{P}^t_{r_i}}{10q}} \ . \tag{4.6}$$

This range approximation requires a-priory knowledge of channel parameters, channel exponent value, and receive and transmit antenna gains that determine the exponent offset. Using the common channel exponent for an indoor environment in

a range of $2-4$ will not provide accurate results and will not compensate for specific channel conditions such as shadowing.

Location Estimation

Let us denote by $\tilde{D}$ the $N \times M$ matrix of approximated distances between the mobile nodes and the N anchor nodes over M measurements.

The following criteria can estimate the mobile node's location:

$$\hat{g} = argmin_g E(L_0 - g(\tilde{D}))^2 \tag{4.7}$$

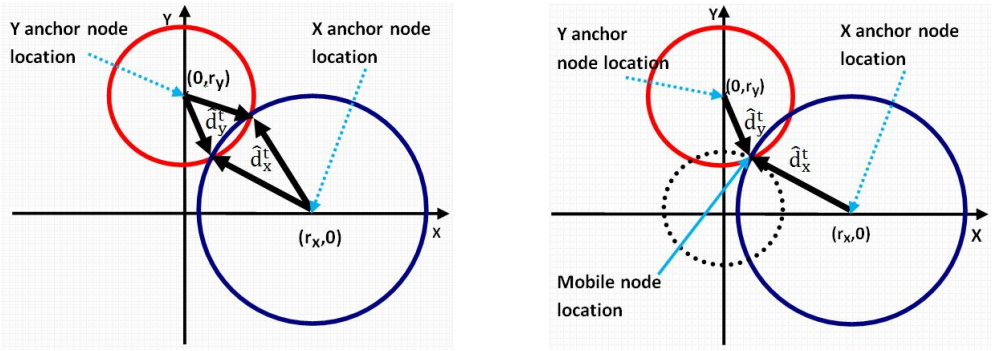
$$s.t.\ |L_0^{t+1} - L_0^t| < \delta\ .$$

There are several methods for solving (4.7). The most common one is trilateration. Trilateration is a positioning technique [94] which estimates the mobile node's location by the intersection of the circles, each centered on the anchor node position with a radius equal to the estimated distance between the mobile node and the anchor node. The number of anchor nodes required for localization is $N = p + 1$ anchor nodes in a p dimensional space. The estimated location is defined by the center of the region formed by the intersection of the circles. Another approach [95] utilizes only $N = p$ anchor nodes and estimates the location by only one of the intersection points. It records several intersection points in consecutive times, and estimates the intersection location by the closest distance.

We choose a variant of [95] to estimate the mobile node's location using the Maximum A Posteriori (MAP) criterion. Assuming the range estimations have the same statistical distribution and the mobile node location has Gaussian distribution, the MAP criterion coincides with the MMSE criterion, [96]. The solution is performed with the following steps:

1. Derive the intersection of the circles formed by the estimated distance for each anchor node.

2. Choose the intersection that minimizes the MAP criterion.



(a) Intersection points of two anchor nodes' circles.

(b) Selection of the intersection point that is related to the mobile node's location.

Figure 4.1: The two circles centered on the anchor nodes' positions.

An elaboration on these steps can be found in [26]. Figure 4.1 illustrates an example of two estimated intersection points and the MAP selection.

Post Processing

We use additional filtering in order to smooth the results and to exclude location estimations that are not likely due to the continuous movement. The filtering at this stage is more effective as it is performed on the combined estimation obtained by all anchor nodes, unlike the filtering in the pre-processing stage which is performed on each anchor node separately. Furthermore, the locations are continuous, and the constraint in (4.3) can only be applied at this stage. Mean filter and median filter [97] are both effective filtering techniques that exploit statistical information of the data. Mean filter [98] is a variant of a moving average that operates only on measurements in which their standard deviation is above a predetermined threshold. The median filter is a non-linear filtering technique that uses the median value instead of the mean. The median filter and the mean filter performances depend on the window size and threshold value. With zero thresholds, the mean value coincides with the LPF simple moving average. With Gaussian noise, the mean and median filters display a similar performance [99]. When the data is corrupted by spike noise, the median filter can

exclude the spikes better than the mean and the LPF filter [100]. We use the median filter to apply the continuity constraint on location estimations which can be caused either by error in combining the anchor nodes distance estimations or by imperfect compensation for packet loss due to burst noise. We exploit the temporal correlation by using the LPF that averages proximate location estimations.

The filters' length should reflect the temporal correlation of the RSSI measurements. It can be small with high measurement correlations due to low mobile node velocity, high transmission rate, or small channel deviations. The filters' length should be higher with an increase in the mobile node velocity, a low transmission rate, or distortion of the signal due to multi-path fading and packet loss burst.

4.4 Experimental Setup

Two sets of experiments in a small-scale indoor environment were conducted. The first set tracked a toy car moving over a plastic trail. The second set of experiments tracked the movement of a human hand. In both experiments the setup included two anchor nodes, a mobile node, a base station, and a notebook. RSSI measurements were derived at the anchor nodes located at known separate locations and then sent through the base station to a notebook for further analysis. For the first experimental set, a reference to the tracking application was obtained by interpolating predetermined points marked in advance on the plastic trail. The reference was synchronized to the tracking according to a starting point using a relatively constant car velocity. For the second set, an optical real-time motion tracking system Polaris (Northern Digital Inc.) [101] was used as a reference. The Polaris tracking system provides an accurate orientation and positioning information with an update rate of up to 60 Hz and accuracy of around 0.35 mm. The Polaris affective coverage is limited to 1 square meter. The system model of the mobile car tracking and the hand movement is shown in Figure 4.2, Figure 4.3 and Figure 4.4.

The two anchor nodes and the mobile node included a BSN node [102] and a dipole antenna. A BSN node includes a processing unit (TI MSP430) and a transceiver for the wireless communication (Chipcon CC2420) [88]. The transceiver has a built-in

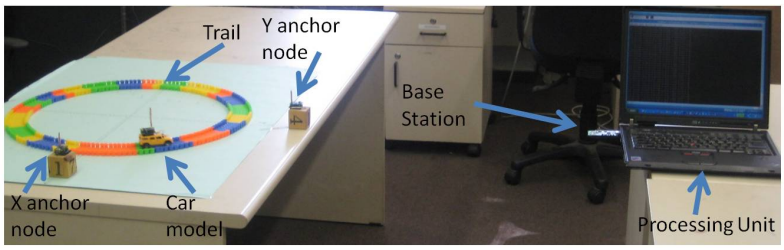


Figure 4.2: The basic experiment setup used in the first set. In addition to the circular trail shown here, two more shapes were used.

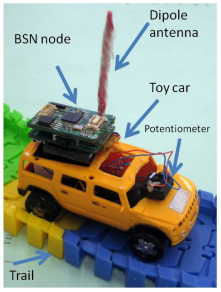


Figure 4.3: The mobile node on the toy car.

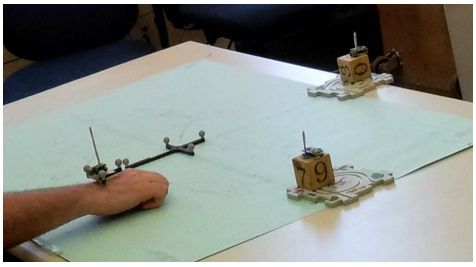


Figure 4.4: Hand tracking setup.

RSSI that provides a digital value in the range of -127 to 128 dBm. The RSSI value is always averaged over 8 symbol periods (128 μs). The conversion of the RSSI measurements to the received power is done by an addition of -45 dBm. The transceiver offers an additional mechanism for transmission power selection in a desired range from -25 dBm to 0 dBm. Using the TinyOS operating system [103], we set the transmission power in real-time, according to the power selection algorithm. An omnidirectional antenna was added to each node to increase the transmission range. The antenna was made by bending and winding together a 10 cm wire and forming a dipole antenna of 5 cm, which is equivalent to half the length of the 802.15.4a wave length. The antenna was connected to a dedicated connector in the BSN node's board. The anchor nodes were placed at x and y axes and attached to a wooden cube giving them the same height as the mobile node and forming a 2-D plane that maximizes performance. The mobile node was attached to the car or to the human hand. The base station was implemented by a sensor node (TelosB) [7]. The communication between the base station and the anchor nodes was implemented by 802.15.4. The

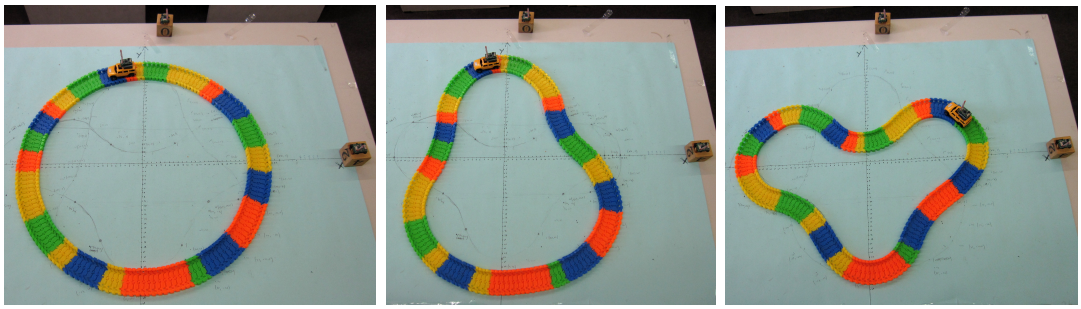


Figure 4.5: Trails used in the experiments.

notebook processed the data (IBM T43) and was used to program the sensors.

The first set of experiments was performed in an indoor environment in stationary conditions without significant reflection from walls and without metal reflectors. There was a direct path between the sensor nodes approximating LOS conditions. The maximal range among the sensor nodes was less than a meter. The anchor nodes were located in the x and y axes, in coordinates of (40;0) and (0;40) centimeters respectively. We examined three different trails in the shapes of a circle, a pear and a heart, as shown in Figure 4.5. The circular trail was with radius of 26 cm and the two other trails were in the range of 10 to 68 cm from the origin. Each trail had a reference of marked points. Using an interpolation process could have provided a description of the trail with 1 cm error. To compare the tracking results with the reference, we matched each reference point with a time stamp. For the circular trail, the matching was accurate with a resolution of 2 cm due to the nearly constant velocity of the toy car.

The calibration between the mobile node and each of the anchor nodes was performed at 12 different locations along the trail. The two anchor nodes received packets transmitted from the mobile node with a known power level. They calculated the received power level and then sent it to the processing unit for analysis. For the optimal transmit power level selection we used the two extreme locations and found the optimal transmit power for each sensor node as described in (4.4). The 32 different transmit power levels of Chipcon CC2420 were changed in a loop by commands sent

from the base station. The packet loss ratio and the RSSI measurements were continuously recorded and stored in the notebook. For the tracking phase, the car model traveled over the trail having a constant velocity of 0.33 m/s. The sensor node was attached to the top of the car and periodically transmitted data packets with time stamps in a cycle of 20 ms. Each anchor node computed the received power and transmitted it to the processing unit via the base station.

The second set of experiments was performed in the same environment as in the first set, but with a presence of a human body. With the human body, the channel was not fully LOS as the body reflected part of the transmitted signal to the medium and, due to body movements, the channel was not fully stationary. The calibration process between the mobile node and each of the anchor nodes was performed at 8 different locations. The measurements were at a distance of 20 cm to 85 cm from the nodes. During the tracking phase, the hand arbitrarily moved in the area at different speeds.

4.5 Experiments' Results

We conducted a series of experiments with the different trails. The experiments for each trail had a separate calibration process. We analyzed the tracking system performance accuracy and the effect of the processing stages.

4.5.1 Car Tracking Offline Phase

The mapping of the power measurements to distance was performed on 12 different locations along the pear shape path. The RSSI measurements were converted to power levels and then sorted according to their distance levels.

Each anchor node has its own unique calibration process to reflect a possible variation in the antenna dimensions, orientation, and other manufacture's chipset parameters that affect the transmit and receive power. The mapping of the received power to distance of the two anchor nodes for the pear shape trail is shown in Figure 4.6. Node x received power is almost 2 dBm lower than node y. This demonstrates the need for the separate calibration process for each pair of nodes.

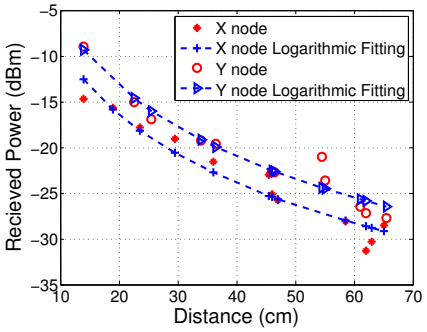


Figure 4.6: The mapping of the measured power to distance.

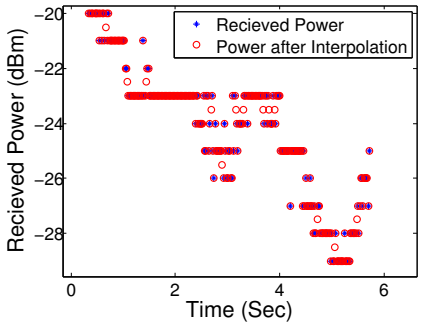


Figure 4.7: Interpolation of the RSSI measurements.

The optimal power was obtained to minimize the criterion in (4.4). The packet loss threshold was selected to be 5% and the standard deviation threshold was selected to be 2 dBm. The transmit optimal power was selected to be -11 dBm.

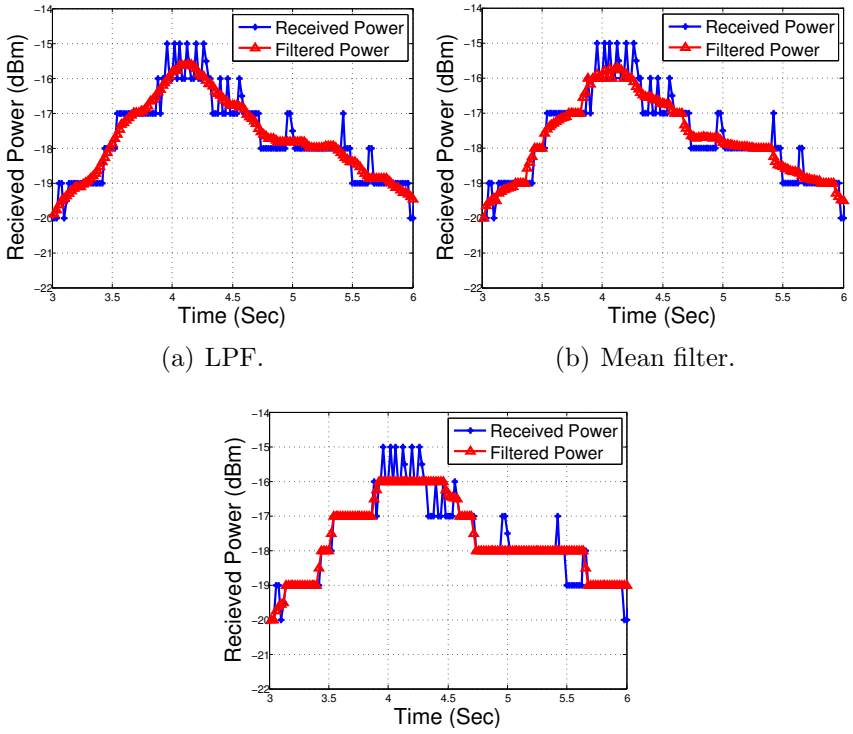
The dynamic range, as obtained from Figure 4.6, was around 16 for the environment dimension of 55 cm. This dynamic range represents the coarse sensitivity of the tracking system. For a standard deviation of 1 dBm, the corresponding tracking resolution per dimension would be around 4 cm.

4.5.2 Car Tracking Phase

Pre-processing

The received power measurements are stored in a synchronized frame. For each frame we performed interpolation of the measurements to recover missing measurements and to synchronize between the anchor nodes' data. Figure 4.7 demonstrates the received power in x and y nodes on a frame with a packet loss ratio of 12% and 10%, respectively.

After the nodes frame synchronization and packet loss recovery, we filtered the channel interference by a LPF. We examined 3 types of filters for the received power - a standard moving average window, which is equivalent to a LPF with equal coefficients, and mean and median filters. Figure 4.8 shows the received power for node x before and after the filters. The mean and the median filters threshold was 0.6. The window size that was used for the filters was 15 measurements which is equivalent to a time



(a) LPF.

(b) Mean filter.

(c) Median filter.

Figure 4.8: Filtering in the pre-processing stage.

duration of nearly 0.3 sec and to a distance of about 10 cm. The window size for each of the filters is a function of the received power standard deviation and the ratio between the mobile node velocity and the transmission rate. For example, for a double transmission rate, the window size must be doubled. The optimal window size can be obtained with an autocorrelation function of the RSSI measurements. The mean and the median filters replace values that divert from the mean value by more than the standard deviation with the mean and median values, respectively. These filters sometimes distort the signal and exclude fine and delicate parts of the signal. Since the RSSI interference in our system is not characterized by spikes and the received power level standard deviation is low (in the range of 1 dBm), we use a LPF filter. This filter suitably exploits the correlation between successive measurements in time and space and is commonly used when there is no additional statistic knowledge on

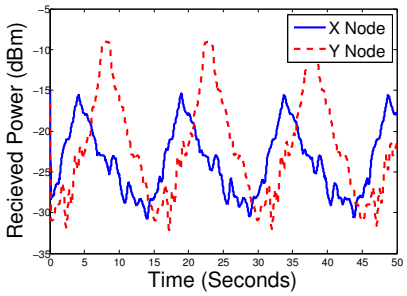


Figure 4.9: Received power after filtering.

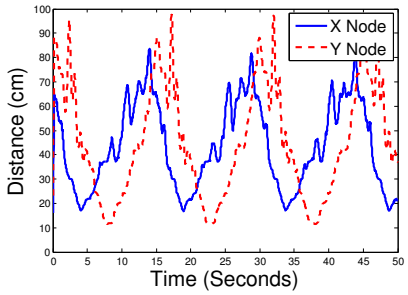


Figure 4.10: Mapping power to distance.

the signal.

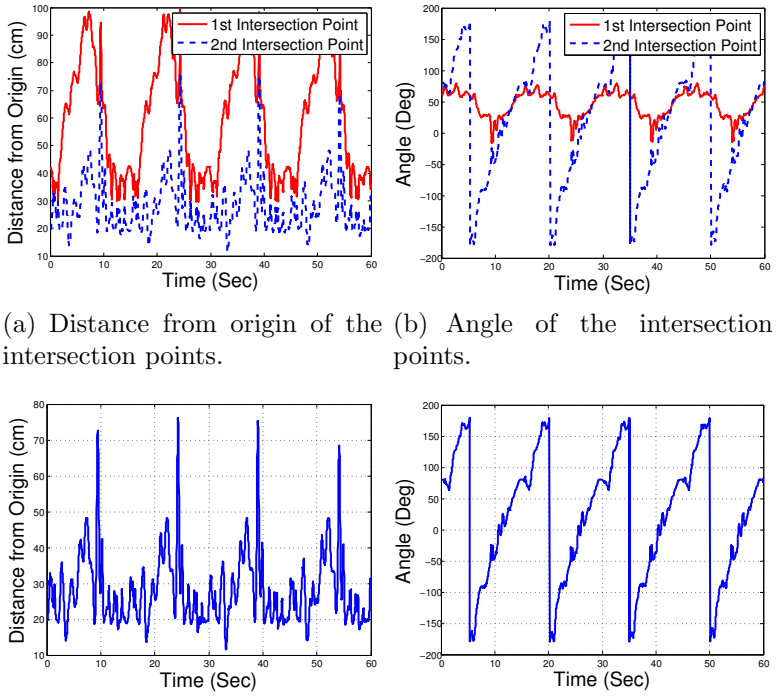
Figure 4.9 shows the filtered received power at the anchor nodes for 3 cycles. The fluctuations, which are caused by bumps over the trail or by small variations in antenna orientation, are mostly removed by the filtering operation. The peak power in the node y is about 7 dBm higher than the power in node x. This may be a result of the asymmetry of the path for the nodes x and y or from the difference in channel conditions of the two nodes, in particularly a difference in the antennas' gain. As the x and y axes are orthogonal, the phase shift between x and y is nearly 90%.

Range Estimation between the Sensor Nodes

The estimated range between the mobile node and the two anchor nodes in the pear shape trail is presented in Figure 4.10. The distance estimation closely represents the balance between excluding the noise and not filtering out part of the real signal. Additional filtering is applied after the location estimation where mutual information from the x and y nodes is combined. Subsequently, discontinuity in localization estimation is more easily distinguished and easier to filter out.

Location Estimation

We found the intersection points of the circles, which are formed by the distances between the anchor nodes and the mobile nodes, according to Section 4.3.3. Figure 4.11 describes the intersection of the two circles in polar coordinates. We select the intersection of the mobile node's location that minimizes the MAP criterion. Since the



(a) Distance from origin of the intersection points.

(b) Angle of the intersection points.

(c) Chosen distance from origin.

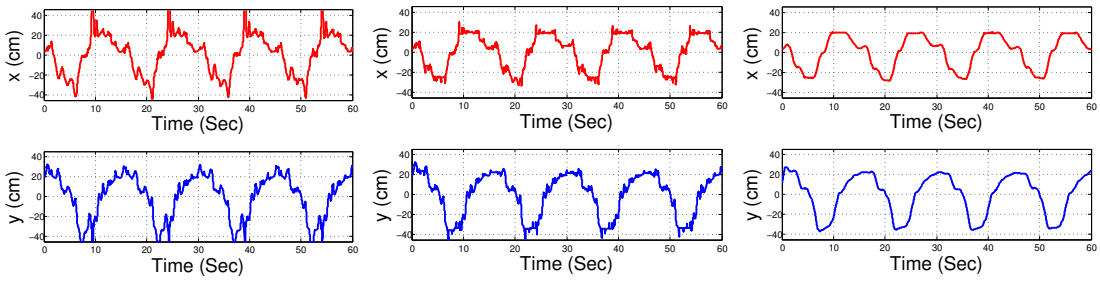
(d) Chosen angle.

Figure 4.11: The intersection points and the chosen intersection point values.

mobile node in our experiment is inside the range defined by the two sensors, the second intersection point is usually selected.

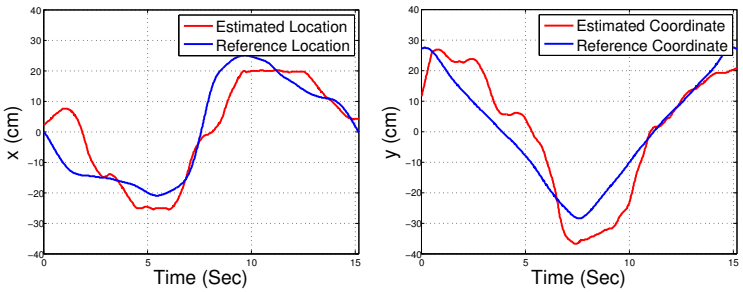
Post-processing

Figure 4.12(a) shows the location estimation for the x and y axes before the post-processing stage. The location estimation is noisy and some of the location estimations are not continuous and have large deviations. To apply the constraint of continuity in (4.3), we use a median filter that excludes irregular location estimations. The median filter window size must be tailored to the specific channel similar to the pre-processing filters. For example, if the irregularity is due to a packet loss burst, the window size must be twice the size of the burst. The standard deviation threshold should be set with the window size. The threshold level should be adopted to the signal. A too high threshold might filter out significant information while a



(a) Estimated location before post-processing. (b) Estimated location after applying median filter. (c) Estimated location after LPF.

Figure 4.12: Estimated location of the mobile node at the post-processing stages.

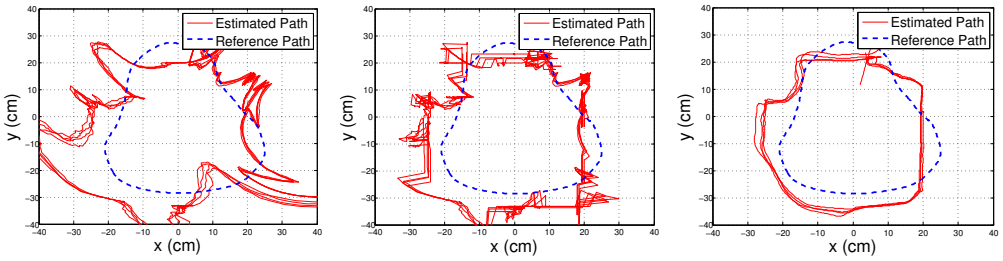


(a) Location estimation of x. (b) Location estimation of y.

Figure 4.13: Estimation of location coordinates of one cycle compared to the reference.

too low threshold might not filter part of the noise well. Figure 4.12(b) shows the location estimation after applying a median filter with a window size of 50, which is equivalent to 1 second, and a standard deviation threshold of 0.9. The correlation in time can be further exploited by a LPF, which averages the signal and smoothes the location estimation. Figure 4.12(c) shows the location estimation after filtering with a LPF implemented by a moving average with a window size of 30 measurements.

The mean error and standard deviation of the approximated location before the post processing were 7.23 cm and 6.2 cm for the x axis, and 7.19 cm and 5.77 cm for the y axis. After the median filter, the results were reduced to 5.72 cm and 4.74 cm for the x axis, and 6.79 cm and 4.56 cm for the y axis. After the LPF filter, the mean error and standard deviation were further reduced to 5.35 cm and 4.3 cm for the x



(a) Location estimation before post-processing.

(b) Location estimation after applying median filter.

(c) Location estimation after median filter and LPF.

Figure 4.14: Location estimation for x and y axes during post-processing stage.

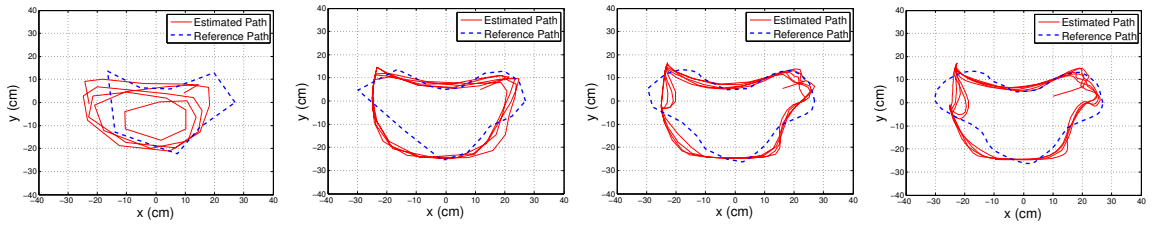
axis, and 6.14 cm and 4.06 cm for the y axis. The estimation quality at different locations for x and y coordinates is shown in Figure 4.13.

The post-processing stages are demonstrated in Figure 4.14. The 2-D location estimation is compared with the reference path, which was obtained by markers on the trail. The location mean error and standard deviation before post processing were 11.21 cm and 7.09 cm. After the median filter, it was reduced to 9.75 cm and 5.22 cm. After the LPF, it was further reduced to 8.90 cm and 4.73 cm.

Transmission Rate Effect

The transmission rate should be high enough in order to track the object motion. With a relatively high transmission rate, diverse measurements can be used to exploit temporal correlation as in (4.3) and to enhance system tracking accuracy. In a static environment, the main factor that determines an adequate transmission rate is the toy car velocity. For a velocity in the range of 10 cm/sec, a transmission rate of 10 Hz allots one measurement per centimeter.

Figure 4.15 describes the tracking accuracy achieved in the heart shape trail with various transmission rates. The heart shape trail demonstrates the significance of the transmission rate better than other trails, such as the circle, because of small variations along the trail that require precise tracking. Figure 4.15(a) shows the tracking results with a transmission rate of 1 Hz. The tracking is not consistent and there are variations in the different cycles along the trail. These variations can be



(a) 1Hz (b) 2Hz (c) 5Hz (d) 10Hz

Figure 4.15: Transmission rate effect on tracking accuracy.

explained by the lack of diverse measurements. Tracking with a transmission rate of 2 Hz, with nearly a sample per 5 cm is shown in Figure 4.15(b). The tracking accuracy is significantly higher than the previous one. Still, the scheme cannot detect details of the trail. With 5 and 10 Hz, the tracking can detect finer details of the trails as shown in Figure 4.15(c) and Figure 4.15(d). It seems that a transmission rate of 5 Hz is sufficient for our environment, forgoing a slight amount of accuracy. Nonetheless, a higher transmission rate can improve the results and is necessary when the environment is dynamic with scatterers.

4.5.3 Hand Movement Offline Phase

The human body affects the channel model in that the signal transmitted from the mobile node is partially absorbed in the body and partially reflected back to the medium. The reflection from the body and the body movement result in non-stationary channel conditions during the calibration process. Figure 4.16 shows the receive power measured at nodes x and y when a human body is present. A considerable difference in gain and dynamic power range can be seen between the two nodes. This difference can be explained by the strong reflection from the body in the direction of node y that enhanced the received power and increased the power level. Due to motion of the human body, the variability of the measurements at node x is much higher in comparison to the measurements used for the first experiment set, as shown in Figure 4.6. This variability demonstrates the non-stationary channel conditions in the direction of node x.

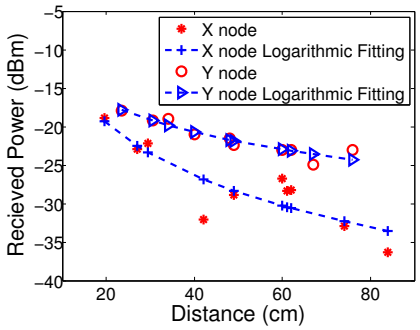


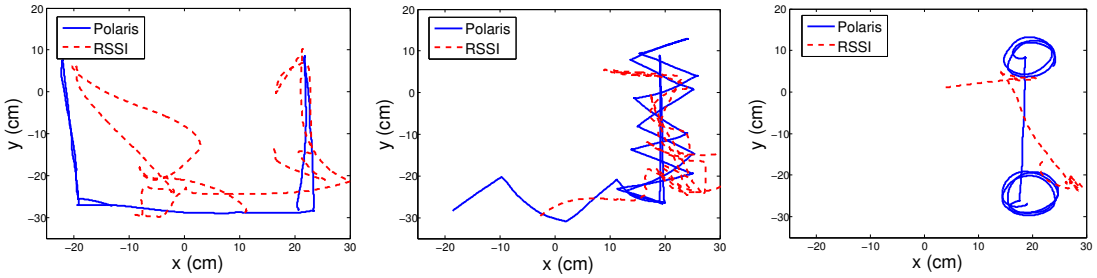
Figure 4.16: Power levels measured in the hand movement tracking.

4.5.4 Hand Movement Tracking

During tracking, the hand moved in an arbitrary path in the 2D space determined by the two anchor nodes. Unlike the first experiment set, where the mobile node velocity was relatively constant and the channel conditions were close to LOS, in this experiment set, the mobile node moved at different velocities and the channel was not stationary due to the human body effect. The Polaris tracking system captured the hand movement during the experiment period. The RSSI measurements and the tracking system were synchronized by correlating the RSSI location estimation to the one of the tracking system over time. Figure 4.17 illustrates the tracking results of three experiments: a hand moving in horizonal movement, in a zig-zag pattern, and in a combination of horizontal movement and a circular pattern.

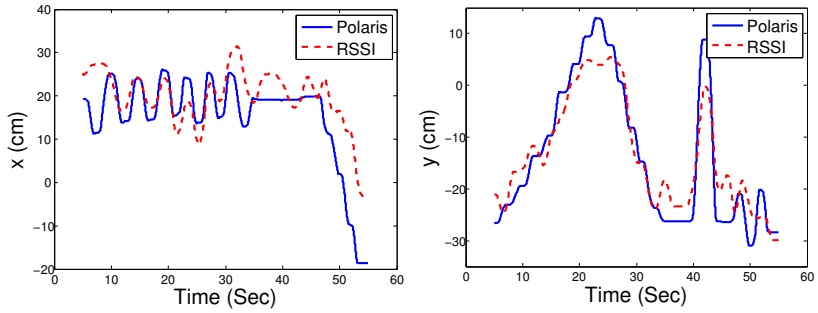
The location estimation mean errors were 10.77 cm, 7.88 cm and 9.07 with standard deviations of 5.69 cm, 4.77 cm and 3.0 cm, respectively.

The distribution of the location estimation error is not uniform along the path. The non-uniform distribution can be explained by the statistical nature of the calibration, which is performed in one dimension for each sensor node, and cannot capture all of the channel variations in $2D$. For example, in Figure 4.17(a), the right area has a lower estimation error compared to the left area. The low estimation error in the right area can be explained by the coincidence of the calibration points at a distance of $10-20$ cm from the anchor node x with the channel path-loss model (the logarithmic fitting). The high variations at a higher distance of $30-60$ cm caused



(a) Hand moving straight. (b) Hand moving in a zig-zag pattern. (c) Hand moving in a circular pattern.

Figure 4.17: Location estimation of a hand movement.



(a) Location estimation at x axis. (b) Location estimation at y axis.

Figure 4.18: Location estimation of hand moving in a zig-zag pattern.

an estimation bias. As a result, the calibration accuracy and the overall location accuracy decreases in areas where the calibration varies from the channel model.

Figure 4.18 shows the tracking results of the hand moving in a zig-zag pattern for axes x and y. The tracking results for each axis have a mean error of 4.33 cm and 0.19 cm, and standard deviation of 6.74 cm and 4.55 cm, for axes x and y respectively. The experiment results vary more slowly compared to the reference. This can be explained by the estimation bias and by the filters used in the data processing. Even with the relatively high estimation error due to the non-stationary channel conditions, the pattern of the movement is well captured by the location estimations. As a result, RSSI-based tracking can be useful for tracking a movement pattern even in a dynamic environment.

4.6 Conclusion

We proposed a new RSSI-based tracking system that utilizes commonly available a priori knowledge about the environment. This new tracking scheme is optimized for networks like BSN, where the sensor nodes are deployed in close proximity and operate with an adequate power level and with a high transmission rate. A MMSE criterion for tracking, based on the coarse RSSI measurements and their temporal correlation, was derived. The sensor node transmission power level was chosen to maximize the RSSI dynamic range for better tracking accuracy. Effective processing techniques for efficient solution of the criterion were used. For performance evaluation, we conducted a series of experiments in an indoor environment for tracking a mobile node's location on various paths with two anchor nodes. The first experiment set was designed for tracking a toy car moving on a plastic trail. The second experiment set was designed for tracking the movement of a human hand in a dynamic environment. Experiments results' show location mean errors and standard deviations of few centimeters.

Extension of the tracking system to a 3D environment is challenging and is planned to be carried out in the future. Tracking in 3D requires additional sensor nodes and adaptation of the localization algorithm. Systems that consist of isotropic antennas, where the antenna has the same intensity of radio waves in all directions, can use similar calibration and tracking phases as in 2D. When the antennas are not fully isotropic and have directionality, the RSSI measurements can vary inconsistently in different planes, e.g. the RSSI measurements may be affected by the reflection of the signal from the ground. Therefore, the calibration process should be changed and also include additional information regarding sensor orientation, e.g. the height of the mobile node or its orientation.

Information from external devices about channel conditions, sensor nodes' location, or orientation can further improve the accuracy of RSSI-based tracking systems. For example, a feedback from a camera, inertial sensors or other devices can be used to adopt the adequate transmission power level during the tracking. External information can be also used to adopt the RSSI measurements to the dynamic channel. Combining the external information requires efficient synchronization of the RSSI

measurements with the external information. Another prospective challenge is to combine the entire information obtained from different sensor nodes by using advance statistical methods, such as Kalman filtering, to mitigate over estimation bias and improve the location estimation accuracy.

The tracking system presented can be used as a robust and economical solution for indoor tracking applications in areas such as BSN. The system is also beneficial for capturing the pattern of the movement for diagnosis.

Chapter 5

Robust Outlier Detection

The chapter starts with an introduction and a formulation of the outlier detection problem as a spatio-temporal statistical problem. A robust estimation of the statistical measures for outlier detection is proposed. Then, the properties and the advantages of our proposed solution using synthetic simulations and experiments with real world data are presented. A discussion, and directions for future work close this chapter.

The WSNs referred in this chapter are assumed to be similar to those described in Chapter 2 and Chapter 3: large networks composed of battery-powered sensor nodes that operate autonomously. The sensor nodes communicate with their 1-hop neighbouring nodes in a distributed manner. In contrast to Chapter 3, this chapter focuses on the data-processing theory rather than on a detailed energy consumption model and memory requirements - these remain for future work.

5.1 Introduction

WSNs are prone to outliers, observations that significantly deviate from the normal pattern of the observations. Outliers may be the result of noisy sensor readings, lack of power, accumulated errors from other sensor nodes, events that cause changes in the environment, or malicious attacks. Outlier detection techniques identify any kind of anomalous observations either to eliminate them or to report on an occurrence of events, and include fault detection, event detection and intrusion detection. Outlier

detection is therefore a fundamental and critical task in WSNs to ensure the reliability and accuracy of the sensor nodes' observations [104].

Statistical approaches to outlier detection are commonly used in WSNs. Some approaches focus on identifying only one type of outliers, e.g. event detection [105, 106]. These approaches have a trigger condition or a special semantic for certain events. More general approaches do not have such prior knowledge. They usually use spatial and temporal correlation to distinguish between several kind of anomalous [107, 108]. The distinction between outliers is possible because noisy observations and faults are usually independent among sensor nodes, where events are likely to be correlated.

The statistical techniques are often based on probabilistic data models [109], where outliers are considered as those observations which deviate from the model distribution. In some cases, the underlying distribution is assumed and the data is evaluated with respect to how well it matches the model. In other cases, when the distribution is not available a priori, the WSN autonomously learns the behavior of the network in order to construct an appropriate statistical model. The statistical techniques can effectively identify outliers and provide a score that is associated with a confidence interval. The key disadvantage of the statistical techniques in unsupervised settings is that the probability distribution model may be wrong due to outliers, resulting in a poor outlier detection.

Robust statistics [110, 111] derives the probability data model fitted by the majority of the data, and therefore the model is less affected by outliers than the simple statistical model. Detection of the deviations from the model allows effective detection of the outliers in the data. Robust estimates of the data model usually involve high computation overhead and therefore they are extremely challenging in WSN. Foundations for statistical tools used in robust estimation in WSNs are presented in [112, 113, 114]. The work in [115] developed robust statistical methods to make localization attack-tolerant. In [116], robust statistics is used for a detection of an insider attacker in the network.

This chapter introduces a unique approach to robust outlier detection by leveraging the spatial and temporal correlation among sensor node observations. We first

model the problem of outlier detection as a spatio-temporal statistical problem. We then offer a robust estimation of statistical measures that are used for outlier detection. A robust mean is estimated using a polynomial regression which employs the temporal correlation between consecutive observations. The robust mean is calculated independently at each node. In time-series data, the regression is also used for a robust detrending. A robust covariance estimation is derived based on the Orthogonalized-Gnanadesikan-Kettering (OGK) principle. Spatial correlation among neighboring nodes established an efficient local and distributed estimation of the inverse covariance matrix using a Gaussian graphical model. We demonstrate the advantages of the proposed method using both synthetic and real data. Performance analysis shows that online outlier detection using a small amount of observations achieves high detection accuracy and low false alarm rate.

The contribution of this work is to offer a novel approach for effective outlier detection using a combination of robust statistic tools and graphical models. This approach uniquely addresses the robustness and efficiency that are required for outlier detection in WSN environment. The proposed method succeeds in identifying anomalies and events in the network under noisy and faulty conditions, and without any prior information. The outlier detection is fully distributed, localized and requires only a small number of observations and messages. The method is general and suitable for a wide range of applications, including time-series processes.

5.2 Problem Formulation

We model the topology of a sensor network by an undirected graph, which refers to the communication graph. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote an undirected graph with a set of p vertices $\mathcal{V}$ connected by a set of edges $\mathcal{E}$. Each vertex represents a sensor node that is associated with a t-dimensional measurements vector. The set of measurements X of the sensor nodes is modeled as a $p \times t$ matrix

$$X = M + Q, \tag{5.1}$$

where M refers to a deterministic temporal component defined as:

$$M = \begin{bmatrix} m_1^T \\ m_2^T \\ \vdots \\ m_p^T \end{bmatrix} ; \tag{5.2}$$

and Q refers to a stochastic spatial component:

$$Q = [q_1 \ q_2 \ \cdots \ q_t] \ . \tag{5.3}$$

For each sensor node $i = 1, .., p$, the vector m_i is based on a simple k-th order polynomial structure:

$$m_i = A\theta_i, \tag{5.4}$$

where A is a known matrix that relates to the unknown parameter and defined as:

$$A = \begin{bmatrix} 1 & a_1 & \cdots & a_1^k \\ 1 & a_2 & \cdots & a_2^k \\ \vdots & \vdots & & \vdots \\ 1 & a_n & \cdots & a_n^k \end{bmatrix} ;$$

and θ_i is the polynomial coefficient vector of size k, which is unknown. We assume a slow time variation process and therefore a small k.

While m_i is composed of highly temporal correlated measurements, q_i is a $p \times 1$ independent and identically distributed (i.i.d.) normal vector with a zero mean and a covariance matrix Σ, such that:

$$q_i \sim_{iid} \mathbb{N}(0, \Sigma) \ . \tag{5.5}$$

Due to the spatial structure, we assume that Σ follows a Gaussian graphical model [117, 118, 119] based on the topology of the network. Graphical models represent conditional dependencies between variables by a graph. In graphical models, each vertex in a graph corresponds to a random variable and each edge represents a conditional

dependency between the variables. An absence of an edge in the graph implies that the variables are conditionally independent. When the variables are jointly Gaussian, sparsity of the WSN graph means sparsity of the inverse covariance matrix. Thus, the conditional independence assumptions are then expressed as:

$$[\Sigma^{-1}]_{i,j} = 0, (i,j) \notin \mathcal{E} \ . \tag{5.6}$$

In practice, we expect to obtain noisy observations with different failures, both in the time and the spatial domains. Thus, we observe

$$Y = X + V, \tag{5.7}$$

where V is a sparse matrix. The entries of V which are not zero represent the outliers in the system. An event at time j is expressed by a column vector $V_{:,j}$ which is not sparse. Those entries $V_{i,j} \neq 0$ which have no clear pattern in time or space connecting them (both $V_{i,:}$ and $V_{:,j}$ are sparse), are considered as local faults.

Our goal is to estimate M and Σ^{-1} given Y. As a byproduct, we would also like to detect the non-zero elements in V and localize the faulty sensor nodes and/or time samples.

One motivation to find M and Σ^{-1} is for applying the Mahalanobis distance metric at each sensor node i for event detection [120]:

$$d_i = \sqrt{(y_i - m_i)\Sigma^{-1}(y_i - m_i)^T}. \tag{5.8}$$

In WSN, when the sensor nodes observations are constant over a time period or occasionally tend to repeat themselves, we can use only a small set of observations in order to estimate M and Σ^{-1}. Then, we can apply the Mahalanobis distance metric on the entire data, including future data.

5.3 Robust Spatio-temporal Estimation

5.3.1 Mean Estimation

Estimation of M can be performed independently at each node $i = 1, .., p$ by computing m_i.

Coefficients Estimation

The coefficients matrix θ_i can be estimated using a polynomial regression [121]. In a polynomial regression, the least-squares (LS) method is often used for the regression analysis. The LS method finds its optimum when the sum s_i of squared residuals is a minimum:

$$s_i = \sum_{j=1}^{t} (r_{ij})^2 \ . \tag{5.9}$$

A residual is defined as the difference between the actual value of the sensor node observation (dependent variable) and the value predicted by the model, such as:

$$r_{ij} = y_{ij} - A\hat{\theta}_i \ . \tag{5.10}$$

The maximum likelihood (ML) estimate of θ_i given y_i corresponds to the LS method and provides a unique solution:

$$\hat{\theta}_{iML} = (A^T A)A^T y_i \ . \tag{5.11}$$

The LS estimates for regression models are highly non-robust to outliers, as the assumptions on the underlying data distribution are violated. Robust regression methods are designed to be less sensitive to outliers in a presence of such violation. Due to the fact that large residuals have a great impact upon the LS criterion, alternative functions have been developed. The M-estimator estimates θ_i by solving:

$$\hat{\theta}_i = \arg\min_{\theta_i}(\sum_{i=j}^{t} \rho(r_{ij})), \tag{5.12}$$

where r_i is a t dimension vector of residuals and $\rho(\cdot)$ is a function corresponding with

some M-estimator, such as the Huber. The ρ function gives less weight to data points which deviate greatly from the parameter and therefore leads to a robust solution.

Iteratively reweighted least squares (IRLS) [122] methods solve the problem in three steps:

1. Select initial estimates of θ_i^0, e.g. LS estimate.
2. At each iteration l calculate the residuals r_i^{l-1} and associate weights w_i^{l-1} from previous iteration.
3. Solve the new weighted least square estimates until convergence:

$$\hat{\theta_i}^l = (A^T w^{l-1} A)^{-1} A^T w^{l-1} y_i \ . \tag{5.13}$$

The termination condition for IRLS can be either by defining a threshold for the estimates difference between successive iterations or by limiting the number of iterations. The computation of θ_i consists of multiplications of A, w_i, y_i and their inverse. In a slow time variation process, the regression model uses a sliding window and the coefficients are calculated with respect to the time described by the window. This resulted in small k and t, providing a small computation overhead.

A byproduct of the polynomial regression is the vector of weights w_i^l. A small weight means that the corresponding measurement deviates greatly from the other measurements and is thus considered as an outlier. We can therefore remove the outlying measurements in order to get $\tilde{Y}$ as a robust estimator of Y.

Detrending

Time-series observations are measured at successive time in predefined time intervals. In some cases, the observations are changed over time in a constant manner (e.g. temperature during day and night) or suffer from a drift over time that may change the real mean. In order to factor out the influence of the time and to work with a zero mean which makes computations simple, a stochastic detrending operation is made to subtract a moving average from the observations. When this moving average is affected by the outliers, the entire window of measurements is influenced and as a

result the amount of outliers in the detrended data is significantly increased. Using a regression for robust estimation of $\tilde{Y}$ is therefore essential to derive a detrending process that is not affected by the outliers.

5.3.2 Inverse Covariance Matrix Estimation

The computation of the inverse covariance matrix Σ^{-1} is usually not practical. The covariance matrix can be high-dimensional because of a large number of observations or a large number of sensor nodes. Therefore, estimation of the covariance matrix by a small number of samples is a fundamental problem. The inherent limitations of WSNs, including restricted power sources, bandwidth and computation capabilities, make this problem to be more challenging. Efficient operation in WSNs requires also a distributed methodology, meaning that the computation should be divided among the sensor nodes in order to decrease the number of data transmissions and the latency in the system.

A distributed and lightweight inverse covariance estimation based on Gaussian graphical models is proposed in [123]. That paper presents an alternative to the global ML [124] and suggests implementation of the ML as an aggregation of decoupled ML estimators computed at each node independently. Each node i estimates its own inverse covariance matrix Σ_i^{-1} with its neighboring nodes N_i using a simple message passing, such that:

$$\Sigma_i^{-1} = (S_{[iN_i],[iN_i]})^{-1} \ , \tag{5.14}$$

where S is the sample covariance given by

$$S = \frac{1}{t} \sum_{j=1}^{t} Y_j Y_j^T \tag{5.15}$$

The first row of the local covariance estimation Σ_i^{-1} for each node i is equivalent to the corresponding i row in the global inverse covariance matrix $\hat{\Sigma}^{-1}$. Therefore, $\hat{\Sigma}^{-1}$ is a $p \times p$ matrix having the first rows of Σ_i^{-1} for $i = 1, ..., p$ in it's $(i, [iN_i])$

positions and having zero padding elsewhere,

$$\hat{\Sigma}^{-1} = \begin{bmatrix} {\Sigma_1^{-1}}_{1,:} \\ {\Sigma_2^{-1}}_{1,:} \\ \cdots \\ {\Sigma_p^{-1}}_{1,:} \end{bmatrix} . \tag{5.16}$$

This local ML method is not necessarily symmetric or positive definite. Moreover, it is not robust in a present of outliers. We therefore propose using the Orthogonalized Gnanadesikan-Kettering (OGK) estimator introduced by Maronna and Zamar [125]. The OGK estimator guarantees a positive semi-definite matrix and it also enables a robust estimation. In addition, it is pairwise and therefore naturally distributed and suits for WSN environment. The OGK is based on the identity:

$$cov(y_i, y_j) = \frac{1}{4}[\sigma(y_i + y_j)^2 - \sigma(y_i - y_j)^2] , \tag{5.17}$$

where σ is the standard deviation. The OGK guarantees a positive semi-definite matrix that can be inverted by making the following steps:

1. Compute the correlation matrix U, such that

$$u_{ij} = \frac{1}{4}(s(y_i + y_j)^2 - s(y_i - y_j)^2).$$

2. Decompose U to $U = EDE^T$, where Q is the $e \times e$ matrix whose columns are the eigenvectors of U, and D is the diagonal matrix composed of the eigenvalues of U.

3. Compute the diagonal d = diag(D) and replace small (negative) values with a predefined threshold.

4. The matrix $\tilde{U} = EdE^T$ is the robust estimation of the covariance matrix, which is positive definite and invertible.

Using robust scale s instead of σ [126] forms a robust covariance matrix. We suggest two approaches for computing s:

1. A naive approach is to use $\tilde{y}_i$ and $\tilde{y}_j$ that were computed during the mean estimation phase.

2. Another approach is to define a robust variance estimator such as the median absolute deviation (MAD) or the interquartile range (IQR).

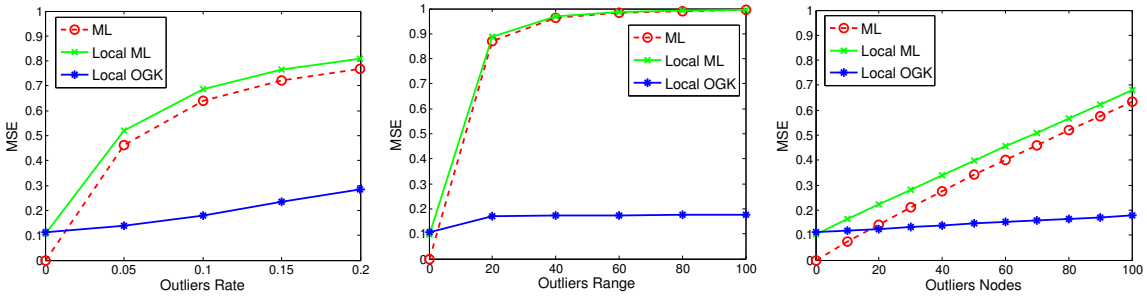
The combination of OGK with Gaussian graphical models means that each node i broadcasts its set of t measurements y_i or $\tilde{y}_i$ and compute $\tilde{U}_i$ based on the input from its neighbors. The matrix $\tilde{U}_i$ is small, proportional to the number of neighbors of the node. Σ_i^{-1} is derived from the inversion of $\tilde{U}_i$. The computation of $\hat{\Sigma}^{-1}$ is therefore performed distributively by a simple message passing.

5.4 Performance Evaluation

We analyzed the proposed outlier detection method throughout three data sets: synthetic data, temperature measurements, and RSSI measurements.

5.4.1 Synthetic Data

The synthetic data relies on purely statistical data and is used to demonstrate the superior performance of the proposed robust inverse covariance estimation. In the synthetic simulation, we generated a network of 100 nodes with an average of 5 neighbors for each node. Each node had 1000 samples that were randomly selected in the range of $[0, 1]$, forming a multivariate gaussian distribution with zero mean. The effect of detrending is therefore not shown here and no robust regression has been performed. The general settings included 10 % outliers at each node, ranging from $[1, 10]$. We changed the outliers rate, the outliers values and the number of outliers node, to study the effect of these parameters on the ML and the OGK approaches. The median absolute deviation (MAD) was used as a robust scale estimator in this case. The centralized ML estimation in graphical model was implemented as described in [118] as a reference to the global ML estimate. Each point in the graph represents an average of 100 monte carlo runs. We used the total mean squared error (MSE) in terms of Frobenius norm, as described in [123], as a metric for evaluation.



(a) Outliers rate effect. (b) Outliers values effect. (c) Outlying nodes effect.

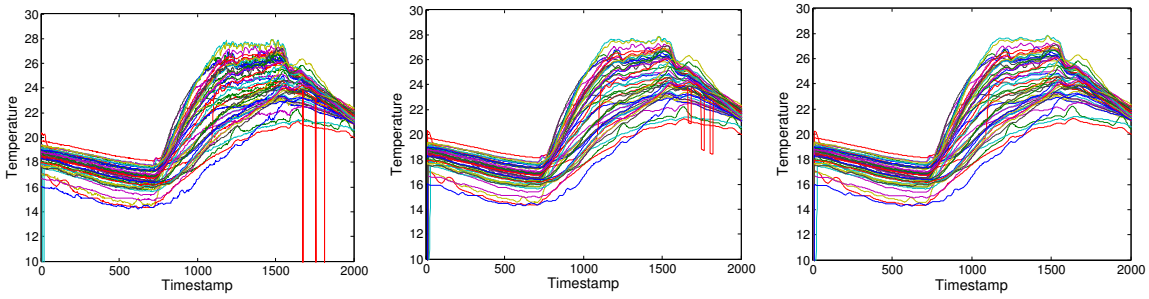
Figure 5.1: Outliers' effect on the estimated inverse covariance matrix accuracy.

Figure 5.1 shows the robustness of the local OGK estimator in compared to the global and local ML solution. While the ML estimations are highly affected by the outliers values and the number of outlying nodes, the OGK estimator provides a consistent and small estimation error. The breakdown point of the OGK in the task of inverse covariance estimation is around 0.2.

5.4.2 Temperature Estimation

In this simulation, we used a real data set collected from Intel Berkeley Research lab [127] between February 28th and April 5th, 2004. We used 2000 consecutive samples of temperature readings from 50 nodes. Because of the sun affects on temperature at different times during the day and at different places, the measurements contain variations over time. However, spatial and temporal correlation still exists in this case.

The main purpose of using this data set is to demonstrate the advantage of robust detrending when outliers exist in the data. Figure 5.2(a) shows the raw readings after the interpolation. As one can see, part of the readings are erroneous and can be considered as outliers. The trend of the temperature is clearly shown and was computed using a local window of 20 samples. Figure 5.2(b) and Figure 5.2(c) shows the trend and the robust trend, respectively. For the robust trend, we applied the polynomial regression with a 2-degree polynomial and the Huber function.



(a) The temperature reading. (b) The trend of the data. (c) The robust trend.

Figure 5.2: Temperature estimation of 50 nodes.

5.4.3 Motion Detection using RSSI

Anomaly detection techniques [128] are important for many applications including motion detection of human activity. Smart home and buildings apply these techniques for automatic lighting and heating. Security and emergency applications, which monitor restricted areas, also use such techniques. Video cameras or radio transmitter tags are not always useful. For example, the view of the camera is limited and cameras do not work well in a smoky environment. Radio transmitter tags require all people of interest to be wearing a radio tag and this is not practical. WSNs can be an alternative for existing systems. With easy deployment within large facilities, WSNs are capable of providing anomaly detection services using radio channel characteristics. The terms device free localization [129] or sensorless sensing [130] are often used to emphasize that this technology does not require carrying devices or using expensive sensors.

Radio signal fluctuation or radio irregularity arises in the presence of human activity [131]. The level of the received signal strength (RSSI) between each pair of transmitter and receiver nodes is changed because the human body causes signal absorption, reflection or scattering. These changes in the signal are usually referred to as multipath fading and shadowing. The absolute signal levels are highly dependent on the surroundings [132], but the interference of the human body is consistent in any environment and effective even through walls [133]. In order to exploit the effects

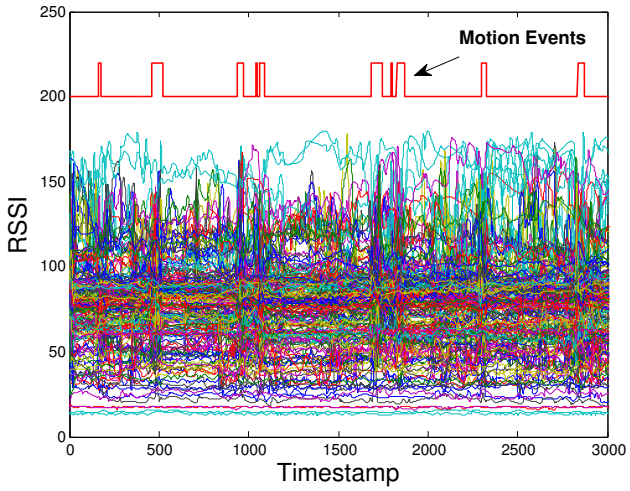
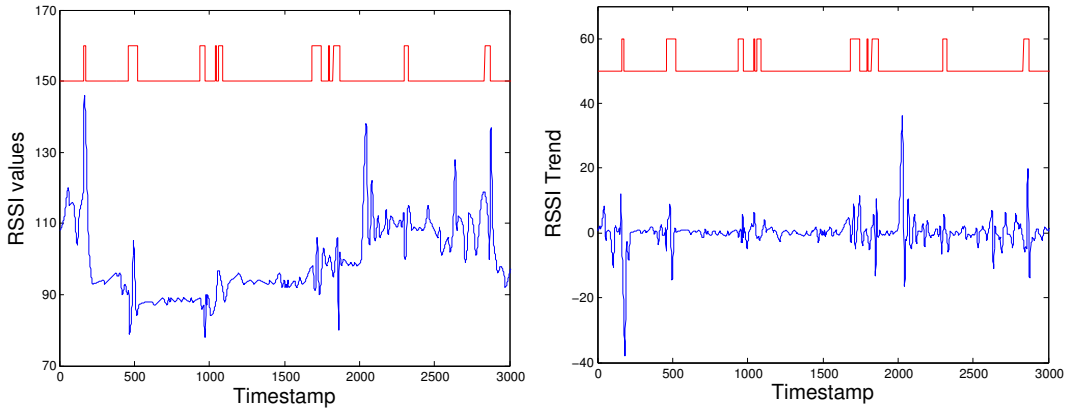


Figure 5.3: RSSI measurements of 182 pairs of sensor nodes.

of radio irregularity, the set of RSSI measurements among pairs of nodes should be collected and processed to infer the anomalies.

Motion detection using RSSI is extremely challenging in the presence of outliers, as it is necessary to distinguish between the anomalies that indicate events of motion and outliers that refer to noise or errors. The last data set we used was taken from a motion detection experiment directed by Prof. Neal Patwari and performed at the University of Michigan [134]. In this experiment, 14 sensor nodes were deployed randomly and transmitted packets every 0.5 seconds. The RSSI values were recorded for each pair of transmitting and receiving nodes, resulting in $14 * 13 = 182$ pairs of RSSI measurements. Persons that walked at random times caused anomalies in the RSSI measurements that were recorded also by a web camera for a ground truth. Figure 5.3 presents the RSSI measurements and the motion events captured by the camera.

The graphical model was formed using the connectivity information of the sensor nodes. Each node had 4-5 neighbors that were selected among the nodes with the highest connectivity rate. The RSSI measurements were detrended before processing in order to obtain a data with zero mean and to remove drifts over time. The detrending used a window size of 10 measurements. The robust method first used a

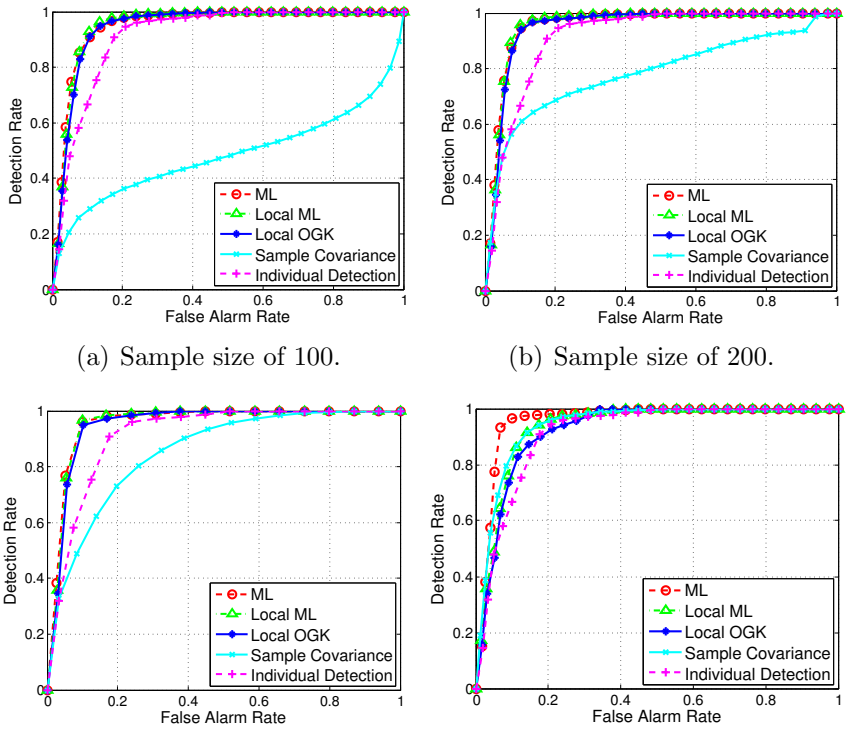


(a) RSSI measurements between one pair. (b) Detrended RSSI measurements.

Figure 5.4: RSSI measurements before and after detrending.

moving median, which is equivalent to polynomial regression with degree 1, to form a robust data. The sliding window was small due to the high variability of the RSSI measurements. Figure 5.4 demonstrates the need for the detrending process. The original RSSI measurements suffer from some bias over time. This bias can be a result of the temperature or other environment effects that are changed during the time.

The distinction between outliers and events was performed on the detrended data in two phases. In the first phase, a random small sample was taken from the measurements in order to estimate the inverse covariance matrix. The inverse covariance matrix was computed using our proposed local OGK method, the global and local ML estimates, and the sample covariance. In the second phase, the Mahalanobis distance metric was applied, using a zero mean and the inverse covariance matrix that was calculated on the entire data set in order to detect the events in the system. We also implemented a purely distributed method, where each sensor node decides based on its own data on event occurrence (inverse covariance equals to the identity matrix). The event detection performance was evaluated using the receiver operating characteristic (ROC) curve averaged over 50 independent 50 Monte-Carlo trials. The curve shows the tradeoff between the detection rate (true positive) and false alarms (false positive).



(a) Sample size of 100.

(b) Sample size of 200.

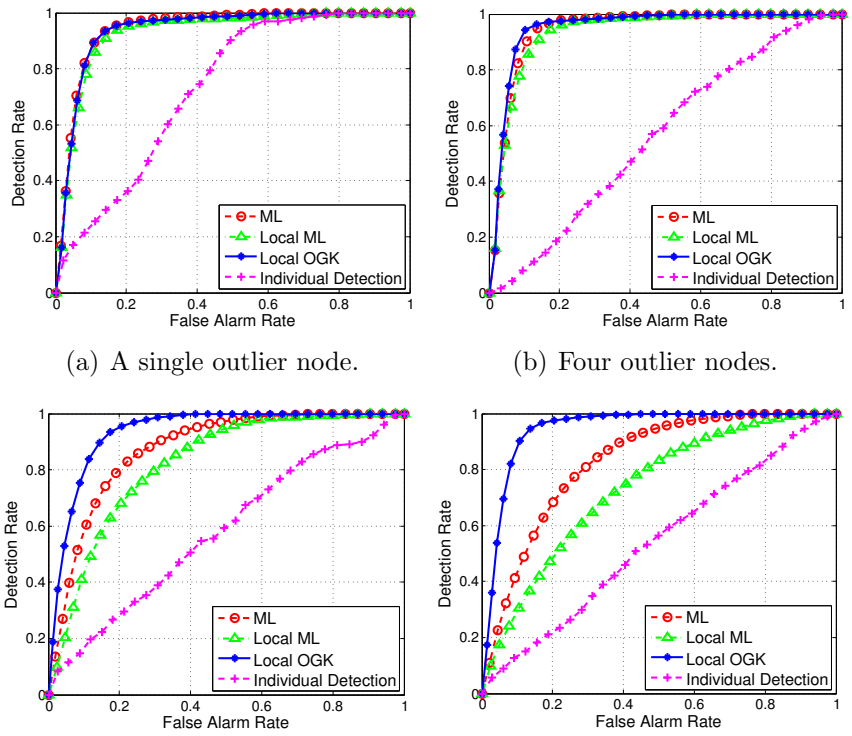
(c) Sample size of 500.

(d) Sample size of 500.

Figure 5.5: Motion detection rate.

Figure 5.5 shows the detection performance on the original data with sample size of 100, 200, 500 and 1500 measurements. As one can see, the detection performance using the sample covariance is highly dependent on the number of samples, while the other methods provide high detection rate with low false positives even when the number of samples is low. The ML, local ML and local OGK approaches are better than individual detection with few samples because they exploit spatio-temporal correlation, while the individual detection uses only temporal correlation. With 1500 samples, which are around 50% of the data, most approaches provide similar performance. The ML approach, which uses the entire information from all the nodes in the network, therefore presents the best detection performance.

The original data set was performed in a controlled environment without outliers that may be caused because of lack of battery and other errors. We simulated one



(a) A single outlier node.

(b) Four outlier nodes.

(c) Seven outlier nodes.

(d) Eight outliers node.

Figure 5.6: Motion detection in a presence of outliers.

percent of outliers that were selected randomly in a range of -180 to 180 in $1, 4, 7$ and 8 nodes in order to examine their effect on the detection performance. We used a sample size of 100 measurements in order to perform the motion detection. Figure 5.6 shows the advantage of the local OGK method over the other methods. In the presence of outliers, the individual detection becomes completely ineffective. The ML and local ML performance becomes worse as the number of outliers increase, with some advantage of the global solution over the local ML.

5.5 Conclusion and Future Work

Outlier detection is essential for increasing network reliability and robustness. This chapter suggested a novel approach for effective outlier detection using a combination of robust statistic tools and graphical models. The problem of outlier detection

was modeled as a spatio-temporal statistical problem and several robust statistical techniques tools were proposed in order to estimate the statistical measures locally and distributively. We demonstrated the advantages of our method using synthetic and real data. Performance evaluation presents a superior event detection of the proposed robust solution than existing solutions. The proposed method is fully distributed, localized and requires for the robust statistics measures only a small number of observations among the entire data.

This chapter provided the foundations for both robust estimation of mean and variance, but the performance evaluation did not use both measures at the same time. In the first data set, the mean was assumed to be zero and only a robust variance was calculated using MAD. In the second and third data sets, the polynomial regression that was performed in order to estimate a robust mean removed the outliers from the data, so the calculation of a robust variance was skipped. We did not address the problem of joint robust estimation of mean and variance, that could have increased the accuracy of the solution. We also did not elaborate on malicious and security attack that may require more extensive handling. Furthermore expenditure optimizations were not discussed. These issues remain directions for future work.

Chapter 6

Conclusions and Future Work

Wireless Sensor Networks have shown astonishing advance in the last few years. Large-scale deployments have been made, for example [135], and commercial products have been developed [136]. The future with WSNs is promising. At the same time, many challenges still exist in various areas. The research summarized in this thesis contributes to increasing the reliability, robustness, efficiency, and lifetime of these networks. This thesis presents several areas in WSN research where leveraging spatial and temporal correlation leads to an improved network performance.

In the first part, utilization of spatial locality between proximate sensor nodes was shown to enable efficient network graph formation for inference. The immediate product from this inference method was a clustering scheme which is aware of the capabilities and the limitations of sensor nodes in the area, and forms a robust hierarchical structure used for efficient networking operations. The first part covered some of the first research experience in WSNs at the Hebrew University, and naturally it focused on WSN characteristics and constraints. It tried to bridge the gaps between the theory graphical models theory and WSNs by proposing a practical inference method. The inference method was designed and developed using TinyOS simulation, which resulted in attention to small details making this method extremely efficient and practical. The advantages of this approach can be further explored in future applications.

The second part of the thesis was concerned with more practical aspects and

focused on tracking using RSSI. Exploitation of temporal correlation between consecutive RSSI measurements led to a more reliable position estimation than other tracking systems. This research was one of the earliest to propose the idea of combining RSSI in BSN environment and it demonstrates the feasibility and the limitations of RSSI measurements for future health applications. The possibilities suggested by this work for the development of accurate tracking are considerable. For example, the RSSI measurements do not suffer from drift like inertial sensors, and therefore can be used for feedback to inertial systems in order to mitigate their drift [137]. The calibration process necessary for accurate tracking required a tedious overhead, even after simplification. An immediate consequence of this work is to use RSSI in ways that do not require exact distance results but just patterns. For example, RSSI measurements can be effective for estimating the placement of body parts, e.g. for studying the walking cycle of a person. This requires only relative distance estimation.

The third part of the thesis directly followed from the insights gained about RSSI in the second part and dealt with investigation of RSSI for motion detection. The outliers and the difficulties in the data processing led to the development of a novel robust method for outlier and event detection. We combined this with the graphical model field studied in the first part in order to increase the efficiency of the computation. We further generalized the method to work with various data streams, including time-series applications, by proposing the robust detrending phase. While the robust statistics methods are not widely used in WSNs, we showed the feasibility and the performance of some tools in WSN environment. This work can be further explored. Investigation of the robust simultaneous estimation of the mean and variance has not been fully investigated in WSN and could be an area for future research. The third part was based on real experiments and data, but the proposed method was not fully implemented in the sensor nodes. We believe that some tradeoffs and new insights will be discovered during implementation of the method in real sensor nodes.

Bibliography

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.

[2] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[3] I.F. Akyildiz and M.C. Vuran. *Wireless sensor networks*. John Wiley & Sons Inc, 2010.

[4] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97, New York, NY, USA, 2002.

[5] T. Arampatzis, J. Lygeros, and S. Manesis. A survey of applications of wireless sensors and wireless sensor networks. In *Intelligent Control, 2005. Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation*, pages 719–724. Ieee, 2005.

[6] Hans-Joachim Hof. *Applications of Sensor Networks*. 2007.

[7] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 364–369. Ieee, 2005.

[8] A. Burns, B.R. Greene, M.J. McGrath, T.J. O'Shea, B. Kuris, S.M. Ayer, F. Stroiescu, and V. Cionca. Shimmer–a wireless sensor platform for noninvasive biomedical research. *Sensors Journal, IEEE*, 10(9):1527–1534, 2010.

[9] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 4, pages 2033–2036. IEEE, 2001.

[10] G. Simon, M. Maróti, Á. Lédeczi, G. Balogh, B. Kusy, A. Nádas, G. Pap, J. Sallai, and K. Frampton. Sensor network-based countersniper system. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 1–12. ACM, 2004.

[11] X.R. Wang, J.T. Lizier, O. Obst, M. Prokopenko, and P. Wang. Spatiotemporal anomaly detection in gas monitoring sensor networks. In *Proceedings of the 5th European conference on Wireless sensor networks*, pages 90–105. Springer-Verlag, 2008.

[12] G.Z. Yang and M. Yacoub. *Body sensor networks*. Springer-Verlag New York Inc, 2006.

[13] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, et al. A macroscope in the redwoods. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 51–63. ACM, 2005.

[14] N. Wang, N. Zhang, and M. Wang. Wireless sensors in agriculture and food industry–recent development and future perspective. *Computers and electronics in agriculture*, 50(1):1–14, 2006.

[15] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: scalable coordination in sensor networks. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. ACM Press, 1999.

[16] 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*, pages 10–pp. IEEE, 2000.

[17] K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie. Protocols for self-organization of a wireless sensor network. *Personal Communications, IEEE*, 7(5):16–27, 2000.

[18] M.C. Vuran,
"O.B. Akan, and I.F. Akyildiz. Spatio-temporal correlation: theory and applications for wireless sensor networks. *Computer Networks*, 45(3):245–259, 2004.

[19] L. Krishnamachari, D. Estrin, and S. Wicker. The impact of data aggregation in wireless sensor networks. In *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on*, pages 575–578. IEEE, 2002.

[20] R. Rajagopalan and P.K. Varshney. Data-aggregation techniques in sensor networks: a survey. *Communications Surveys & Tutorials, IEEE*, 8(4):48–63, 2006.

[21] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi. In-network aggregation techniques for wireless sensor networks: a survey. *Wireless Communications, IEEE*, 14(2):70–87, 2007.

[22] S. Pattem, B. Krishnamachari, and R. Govindan. The impact of spatial correlation on routing with compression in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 4(4):24, 2008.

[23] T. Anker, D. Dolev, and B. Hod. Belief propagation in wireless sensor networks-a practical approach. *Wireless Algorithms, Systems, and Applications*, pages 466–479, 2008.

[24] T. Anker, D. Bickson, D. Dolev, and B. Hod. Efficient clustering for improving network performance in wireless sensor networks. *Wireless Sensor Networks*, pages 221–236, 2008.

[25] G. Blumrosen, B. Hod, T. Anker, D. Dolev, and B. Rubinsky. Continuous close-proximity rssi-based tracking in wireless sensor networks. In *2010 International Conference on Body Sensor Networks*, pages 234–239. IEEE, 2010.

[26] G. Blumrosen, B. Hod, T. Anker, D. Dolev, and B. Rubinsky. Enhanced calibration technique for rssi-based ranging in body area networks. Submitted, 2012.

[27] G. Blumrosen, B. Hod, T. Anker, D. Dolev, and B. Rubinsky. Enhancing rssi-based tracking accuracy in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 2012.

[28] M.I. Jordan. *Learning in graphical models*. Number 89. Kluwer Academic Publishers, 1998.

[29] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. Technical Report TR-2001-22, Mitsubishi Electric Research Laboratories, 2002.

[30] M. I. Jordan and Y. Weiss. *The Handbook of Brain Theory and Neural Networks*, chapter Probabilistic inference in graphical models. MIT Press, 2002.

[31] C. Crick and A. Pfeffer. Loopy belief propagation as a basis for communication in sensor networks. In *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence*, 2003.

[32] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for self-calibration in sensor networks. *IEEE Journal of Selected Areas in Communication*, 2005.

[33] Y. Weiss and W.T. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *Information Theory, IEEE Transactions on*, 47(2):736–744, 2001.

[34] N. Wiberg. *Codes and Decoding on General Graphs*. PhD thesis, Dept. of Electrical Engineering, Linköping, Sweden, 1996.

[35] D. Bickson, D. Dolev, and Y. Weiss. Modified belief propagation algorithm for energy saving in wireless and sensor networks. Technical report, The Hebrew University of Jerusalem, 2005.

[36] Y. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Comput.*, 12(1):1–41, 2000.

[37] M. Chiang and N. Bambos. Distributed network control through sum product algorithm on graphs. In *Global Telecommunications Conference GLOBECOM'02*, 2002.

[38] M. A. Paskin, C. Guestrin, and J. McFadden. A robust architecture for distributed inference in sensor networks. In *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks IPSN'05*, 2005.

[39] T. Anker, D. Bickson, D. Dolev, and B. Hod. Efficient clustering for improving network performance in wireless sensor networks. In *Proceedings of the 5th European conference on Wireless Sensor Networks EWSN'08*, 2008.

[40] C. Prehofer and C. Bettstetter. Self-organization in communication networks: principles and design paradigms. *Communications Magazine, IEEE*, 43(7):78–85, 2005.

[41] B. Krishnamachari, D. Estrin, and S. Wicker. The impact of data aggregation in wireless sensor networks. *Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops, 2002.*, pages 575–578, 2002.

[42] O. Younis, M. Krunz, and S. Ramasubramaian. Node clustering in wireless sensor networks: Recent developments and deployment challanges. *IEEE Network Magazine*, 2006.

[43] J. Y. Yu and P. H. J. Chong. A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys & Tutorials*, 7(1), 2005.

[44] W. R. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4), 2002.

[45] O. Younis and S. Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3(4), 2004.

[46] M. Qin and R. Zimmermann. VCA: an energy-efficient voting-based clustering algorithm for sensor networks. *Journal of Universal Computer Science*, 13(1), 2007.

[47] C. Li, M. Ye, G. Chen, and J. Wu. An energy-efficient unequal clustering mechanism for wireless sensor networks. In *Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems MASS'05*, 2005.

[48] Y. He, Y. Zhang, Y. Ji, and X. (Sherman) Shen. A new energy efficient approach by separating data collection and data report in wireless sensor networks. In *Proceeding of the 2006 international conference on Communications and mobile computing IWCMC'06*, 2006.

[49] Y. Kiri, M. Sugano, and M. Murata. On characteristics of multi-hop communication in large-scale clustered sensor networks. *IEICE Transactions on Communications*, E90-B, 2007.

[50] S. Banerjee and A. Misra. Energy efficient reliable communication for multi-hop wireless networks. *CM/Kluwer Journal of Wireless Networks (WINET)*, 2005.

[51] A. Ault, E. Coyle, and X. Zhong. K-nearest-neighbor analysis of received signal strength distance estimation across environments. In *Proceedings of the First Workshop on Wireless Network Measurements WiNMee'05*, 2005.

[52] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings Of The 1St International Conference On Embedded Networked Sensor Systems SenSys'03*, 2003.

[53] TinyOs. http://www.tinyos.net/.

[54] V. Shnayder, M. Hempstead, B. Chen, G. Werner Allen, and M. Welsh. Simulating the power consumption of large-scale sensor network applications. In *Proceedings of the second international conference on Embedded networked sensor systems SenSys'04*, 2004.

[55] O. Younis and S. Fahmy. An experimental study of routing and data aggregation in sensor networks. In *Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems MASS'05*, 2005.

[56] E. Jovanov, A. Milenkovic, C. Otto, and P.C. De Groen. A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation. *Journal of neuroengineering and rehabilitation*, 2(1):6, 2005.

[57] H. Alemdar and C. Ersoy. Wireless sensor networks for healthcare: A survey. *Computer Networks*, 54(15):2688–2710, 2010.

[58] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1067–1080, 2007.

[59] Y. Gu, A. Lo, and I. Niemegeers. A survey of indoor positioning systems for wireless personal networks. *Communications Surveys & Tutorials, IEEE*, 11(1):13–32, 2009.

[60] F. Seco, A.R. Jimenez, C. Prieto, J. Roa, and K. Koutsou. A survey of mathematical methods for indoor localization. In *Intelligent Signal Processing, 2009. WISP 2009. IEEE International Symposium on*, pages 9–14. IEEE, 2009.

[61] F. Xia, Z. Yang, L. Yao, and W. Zhao. Localization Technologies for Indoor Human Tracking. In *Future Information Technology (FutureTech), 2010 5th International Conference on*, pages 1–6. IEEE, 2010.

[62] T. Rappaport. *Wireless communications: principles and practice*. Prentice Hall PTR Upper Saddle River, NJ, USA, 2001.

[63] M. Sichitiu and V. Ramadurai. Localization of wireless sensor networks with a mobile beacon. *Proceedings of MASS*, pages 174–183, 2004.

[64] M. Bertinato, G. Ortolan, F. Maran, R. Marcon, A. Marcassa, F. Zanella, M. Zambotto, L. Schenato, and A. Cenedese. RF Localization and tracking of mobile nodes in Wireless Sensors Networks: Architectures, Algorithms and Experiments. *University of Padua, Italy*, 2007.

[65] W.Y. Chung. Enhanced RSSI-Based Real-Time User Location Tracking System for Indoor and Outdoor Environments. In *Proceedings of the 2007 International Conference on Convergence Information Technology*, pages 1213–1218. IEEE Computer Society Washington, DC, USA, 2007.

[66] H.J. Lee, M. Wicke, B. Kusy, and L. Guibas. Localization of mobile users using trajectory matching. In *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 123–128. ACM, 2008.

[67] J. Wang, R.V. Prasad, X. An, and I.G.M.M. Niemegeers. A study on wireless sensor network based indoor positioning systems for context-aware applications. *Wireless Communications and Mobile Computing*, 2010.

[68] M. Helen, J. Latvala, H. Ikonen, and J. Niittylahti. Using calibration in RSSI-based location tracking system. In *Proc. of the 5th World Multiconference on Circuits, Systems, Communications & Computers (CSCC20001)*, 2001.

[69] C. Alippi and G. Vanini. A RSSI-based and calibrated centralized localization technique for Wireless Sensor Networks. In *Proc. IEEE Int. Conference on Pervasive Computing and Communications Workshops (PERCOMW)*, pages 301–306, 2006.

[70] N. Patwari and P. Agrawal. Calibration and Measurement of Signal Strength for Sensor Localization. *Localization Algorithms and Strategies for Wireless Sensor Networks*, page 122, 2009.

[71] G. Zanca, F. Zorzi, A. Zanella, and M. Zorzi. Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks. In *in REALWSN'08*, 2008.

[72] T. Stoyanova, F. Kerasiotis, A. Prayati, and G. Papadopoulos. Evaluation of impact factors on rss accuracy for localization and tracking applications. In *Proceedings of the 5th ACM international workshop on Mobility management and wireless access*, pages 9–16. ACM, 2007.

[73] S. Hara, D. Zhao, K. Yanagihara, J. Taketsugu, K. Fukui, S. Fukunaga, and K. Kitayama. Propagation characteristics of IEEE 802.15. 4 radio signal and their application for location estimation. In *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, volume 1, pages 97–101. IEEE, 2005.

[74] E. Elnahrawy, X. Li, and R.P. Martin. The limits of localization using signal strength: A comparative study. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 406–414. IEEE, 2004.

[75] K. Lorincz and M. Welsh. Motetrack: a robust, decentralized approach to rf-based location tracking. *Personal and Ubiquitous Computing*, 11(6):489–503, 2007.

[76] M. Saxena, P. Gupta, and B.N. Jain. Experimental analysis of RSSI-based location estimation in wireless sensor networks. In *Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on*, pages 503–510. IEEE, 2008.

[77] A.S. Paul and E.A. Wan. RSSI-Based Indoor Localization and Tracking Using Sigma-Point Kalman Smoothers. *Selected Topics in Signal Processing, IEEE Journal of*, 3(5):860–873, 2009.

[78] V. Seshadri, G.V. Zaruba, and M. Huber. A bayesian sampling approach to in-door localization of wireless devices using received signal strength indication. In

Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on, pages 75–84. IEEE, 2003.

[79] E. Miluzzo, X. Zheng, K. Fodor, and A.T. Campbell. Radio characterization of 802.15. 4 and its impact on the design of mobile sensor networks. In *Proceedings of the 5th European conference on Wireless sensor networks*, pages 171–188. Springer-Verlag, 2008.

[80] G. Mao, B. Fidan, and B. Anderson. Wireless sensor network localization techniques. *Computer Networks*, 51(10):2529–2553, 2007.

[81] J.B. Andersen, TS Rappaport, and S. Yoshida. Propagation measurements and models for wireless communications channels. *Communications Magazine, IEEE*, 33(1):42–49, 2002.

[82] I. Guvenc, C.T. Abdallah, R. Jordan, and O. Dedeoglu. Enhancements to rss based indoor tracking systems using kalman filters. In *GSPx & International Signal Processing Conference*, pages 91–102. Citeseer, 2003.

[83] N. McDonald. Omnidirectional pattern directivity in the presence of minor lobes: revisited. *IEEE Transactions on Antennas and Propagation Magazine*, 41(2):63–68, 1999.

[84] D. Lymberopoulos, Q. Lindsey, and A. Savvides. An empirical characterization of radio signal strength variability in 3-d ieee 802.15. 4 networks using monopole antennas. *Lecture Notes In Computer Science*, 3868:326, 2006.

[85] S. Lin, J. Zhang, G. Zhou, L. Gu, J.A. Stankovic, and T. He. ATPC: adaptive transmission power control for wireless sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 223–236. ACM, 2006.

[86] J. Blumenthal, F. Reichenbach, and D. Timmermann. Minimal transmission power vs. signal strength as distance estimation for localization in wireless sensor networks. In *Sensor and Ad Hoc Communications and Networks, 2006.*

SECON'06. 2006 3rd Annual IEEE Communications Society on, volume 3, pages 761–766. IEEE, 2007.

[87] H. Ren and M.Q.H. Meng. Power adaptive localization algorithm for wireless sensor networks using particle filter. *Vehicular Technology, IEEE Transactions on*, 58(5):2498–2508, 2009.

[88] AS Chipcon. CC2420 2.4 GHz IEEE 802.15. 4/ZigBee-ready RF Transceiver. *Chipcon AS, Oslo, Norway*, 4, 2004.

[89] H. Lim, L.C. Kung, J.C. Hou, and H. Luo. Zero-configuration, robust indoor localization: Theory and experimentation. *work*, 2005:1818, 2005.

[90] P. Barsocchi, S. Lenzi, S. Chessa, and G. Giunta. A Novel Approach to Indoor RSSI Localization by Automatic Calibration of the Wireless Propagation Model. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1–5. IEEE, 2009.

[91] N. Patwari, JN Ash, S. Kyperountas, A.O. Hero, RL Moses, and NS Correal. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE*, 22(4):54–69, 2005.

[92] F. Sivrikaya and B. Yener. Time synchronization in sensor networks: a survey. *Network, IEEE*, 18(4):45–50, 2004.

[93] A. Jain and E.Y. Chang. Adaptive sampling for sensor networks. In *Proceeedings of the 1st international workshop on Data management for sensor networks: in conjunction with VLDB 2004*, pages 10–16. ACM, 2004.

[94] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61. ACM New York, NY, USA, 2004.

[95] W.H. Liao and Y.C. Lee. A lightweight localization scheme in wireless sensor networks. In *ICWMC'06*, 2006.

[96] H.L. Van Trees. *Detection, estimation, and modulation theory: Detection, estimation, and linear modulation theory*. Wiley-Interscience, 2001.

[97] J.W. Tukey. Exploratory data analysis. 1977.

[98] L. Wu, M.Q.H. Meng, Z. Lin, W. He, C. Peng, and H. Liang. A practical evaluation of radio signal strength for mobile robot localization. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pages 516–522. IEEE, 2010.

[99] H. Liu, J. Li, Z. Xie, S. Lin, K. Whitehouse, J.A. Stankovic, and D. Siu. Automatic and robust breadcrumb system deployment for indoor firefighter applications. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 21–34. ACM, 2010.

[100] J. Bednar and T. Watt. Alpha-trimmed means and their relationship to median filters. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 32(1):145–153, 2003.

[101] NDI Polaris. Northern digital polaris tracking system. 2004.

[102] B. Lo, S. Thiemjarus, R. King, and G.Z. Yang. Body sensor network–a wireless sensor platform for pervasive healthcare monitoring. In *The 3rd International Conference on Pervasive Computing*, volume 13. Citeseer, 2005.

[103] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, et al. Tinyos: An operating system for sensor networks. *Ambient Intelligence*, pages 115–148, 2005.

[104] Y. Zhang, N. Meratnia, and P. Havinga. Outlier detection techniques for wireless sensor networks: A survey. *Communications Surveys & Tutorials, IEEE*, 12(2):159–170, 2010.

[105] B. Krishnamachari and S. Iyengar. Distributed bayesian algorithms for fault-tolerant event region detection in wireless sensor networks. *Computers, IEEE Transactions on*, 53(3):241–250, 2004.

[106] M. Ding, D. Chen, K. Xing, and X. Cheng. Localized fault-tolerant event boundary detection in sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 2, pages 902–913. IEEE, 2005.

[107] W. Wu, X. Cheng, M. Ding, K. Xing, F. Liu, and P. Deng. Localized Outlying and Boundary Data Detection in Sensor Networks. *IEEE Transactions on Knowledge and Data Engineering*, 19(8), 2007.

[108] L. Bettencourt, A. Hagberg, and L. Larkey. Separating the wheat from the chaff: practical anomaly detection schemes in ecological applications of distributed sensor networks. *Distributed Computing in Sensor Systems*, pages 223–239, 2007.

[109] V. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.

[110] P.J. Huber, E. Ronchetti, and MyiLibrary. *Robust statistics*, volume 1. Wiley Online Library, 1981.

[111] P.J. Rousseeuw and M. Hubert. Robust statistics for outlier detection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):73–79, 2011.

[112] M. Rabbat and R. Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 20–27. ACM, 2004.

[113] C. Guestrin, P. Bodik, R. Thibaux, M. Paskin, and S. Madden. Distributed regression: an efficient framework for modeling sensor network data. In *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, pages 1–10. IEEE, 2004.

[114] L. Xiao, S. Boyd, and S. Lall. A scheme for robust distributed sensor fusion based on average consensus. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 63–70. Ieee, 2005.

[115] Z. Li, W. Trappe, Y. Zhang, and B. Nath. Robust statistical methods for securing wireless localization in sensor networks. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 91–98. IEEE, 2005.

[116] F. Liu, X. Cheng, and D. Chen. Insider attacker detection in wireless sensor networks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 1937–1945. IEEE, 2007.

[117] T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning. 2001.

[118] Trevor. Hastie, Robert. Tibshirani, and JH (Jerome H.) Friedman. *The elements of statistical learning*. Springer, 2009.

[119] J. Whittaker. Graphical models in applied multivariate statistics. 2009.

[120] P.C. Mahalanobis. On the generalized distance in statistics. In *Proceedings of the National Institute of Science, Calcutta*, volume 12, page 49, 1936.

[121] D.C. Montgomery, E.A. Peck, and G.G. Vining. *Introduction to linear regression analysis*, volume 3.

[122] P.W. Holland and R.E. Welsch. Robust regression using iteratively reweighted least-squares. *Communications in Statistics-Theory and Methods*, 6(9):813–827, 1977.

[123] A. Wiesel and AO Hero. Distributed covariance estimation in gaussian graphical models. In *Submitted to IEEE Trans. on Signal Processing*. IEEE, 2011.

[124] S.L. Lauritzen. *Graphical models*, volume 17. Oxford University Press, USA, 1996.

[125] R.A. Maronna and R.H. Zamar. Robust estimates of location and dispersion for high-dimensional datasets. *Technometrics*, 44(4):307–317, 2002.

[126] R. Gnanadesikan and J.R. Kettenring. Robust estimates, residuals, and outlier detection with multiresponse data. *Biometrics*, pages 81–124, 1972.

[127] P. Bodik, W. Hong, C. Guestrin, S. Madden, M. Paskin, and R. Thibaux. Intel lab data. *Online dataset*, 2004.

[128] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3):1–58, 2009.

[129] N. Patwari and J. Wilson. Rf sensor networks for device-free localization: Measurements, models, and algorithms. *Proceedings of the IEEE*, 98(11):1961–1973, 2010.

[130] K. Woyach, D. Puccinelli, and M. Haenggi. Sensorless sensing in wireless networks: Implementation and measurements. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, 2006 4th International Symposium on*, pages 1–8. IEEE.

[131] M. Nakatsuka, H. Iwatani, and J. Katto. A study on passive crowd density estimation using wireless sensors. In *The 4th Intl. Conf. on Mobile Computing and Ubiquitous Networking (ICMU 2008)*. Citeseer, 2008.

[132] P.W.Q. Lee, WKG Seah, H.P. Tan, and Z. Yao. Wireless sensing without sensors - an experimental approach. In *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*, pages 62–66. IEEE, 2009.

[133] J. Wilson and N. Patwari. See through walls: Motion tracking using variance-based radio tomography networks. *IEEE Transactions on Mobile Computing*, 2010.

[134] Y. Chen, A. Wiesel, and A.O. Hero. Robust shrinkage estimation of high-dimensional covariance matrices. *Signal Processing, IEEE Transactions on*, 59(9):4097–4107, 2011.

[135] L. MO, Y. HE, J. WANG, M. LI, K. LIU, W. DONG, and W. XI. Long-term large-scale sensing in the forest: recent advances and future directions of greenorbs. 2010.

[136] Spinwave Systems. http://www.spinwavesystems.com/.

[137] Ami Luttwak. Tracking and orientation estimation using inertial sensors and rssi measurements. Master's thesis, The Hebrew University of Jerusalem, December 2011.