# PRACTICAL MOBILE AD HOC NETWORKS FOR LARGE SCALE CATTLE MONITORING

By Bartosz Wietrzyk, MSc.

Thesis submitted to the University of Nottingham

for the degree of Doctor of Philosophy

September 2007

# Abstract

This thesis is concerned with identification of realistic requirements for the cattle monitoring system and design of the practical architecture addressing these requirements. Automated monitoring of cattle with wireless monitoring devices mounted on the animals can increase efficiency of cattle production, decrease its reliance on human labour and thus increase its profitability. Multi-hop ad hoc wireless communication has the potential to increase battery life of the animal mounted devices, decrease their size and combat disconnections. This thesis reveals that no current approach sufficiently addresses energy constrains of the animal mounted devices and potential disconnections.

We propose a delay tolerant store and forward architecture that provides data retention, detecting custom events, issues notifications, answers remote and in-situ queries, based on requirements identified during field experiments we conducted. This architecture utilizes fixed infrastructure but also works in ad hoc infrastructureless conditions. The core of the proposed architecture, Mobile Ad Hoc Network (MANET) communication, provides offloading data for long term storage by sending data to farm servers via sinks that are a part of MANET and handles in-situ queries issued by users collocated with the animals. The proposed MANET routing algorithm addresses high mobility of nodes and disconnections. It provides lower and more balanced energy usage, shorter delays and increased success ratio of delivering answers to in-situ queries than more generic existing approaches.

Problems of large scale deployment of the envisaged system are also addressed. We discuss the necessary configuration process performed during the system installation as well as pervasive mobile and home access to the target system. We propose cost efficient strategies for sinks installation and connecting sinks to farm servers, adaptive to different requirements, estates layout, available infrastructure and existing human and vehicle mobility. We also propose a cost efficient security model for the target system based on public key cryptography.

*To my parents*

# Acknowledgements


I would like to offer my deepest gratitude to Dr. Milena Radenkovic and Prof. Chris Greenhalgh for all their help, support and kindness during the course of my PhD. Our collaboration and discussions were very important for me and for my work, and most enjoyable. I would especially like to thank Dr. Milena Radenkovic for her invaluable input and feedback, which challenged me and encouraged me from the early stages of my PhD. Also, thanks to EPSRC for sponsoring my PhD.

I would like to thank people from the School of Biosciences, especially Prof. Anthony Flint, Mr. Ivaylo Kostadinov, Mr. Nigel Armstrong and Mr. Neil Saunders for their help and support in performing field experiments presented within this thesis. I am also grateful to Prof. Jon Crowcroft and Dr. Tristan Henderson for their precious feedback concerning content of this thesis.

Many thanks go to my colleagues in School of Computer Science & IT for their friendship over the years.

# Contents

# List of Figures

# List of Tables

# 1. Introduction

In many countries animal rearing has been an important constituent of agriculture. Particularly cattle production still strongly depends on the human labour which increases its costs and decreases its profitability. Livestock producers have always needed to 'see' their animals as frequently as possible to detect oestrus and animal diseases such as mastitis and other infection diseases, metabolic diseases, lameness and eating disorders. Inattention to the wellbeing of the animals, whether it be a health or welfare issue can lead to reduced productivity and death of valuable stock [1]. It is often the case that the farmer has neither time nor resources to 'observe' the animals regularly and even when she does, she may not be in a position to identify reproductive state of the individuals or some of their more deep-rooted health problems. Contrary to humans who can describe their situation to a doctor, an animal is usually unable to communicate symptoms to an observer. In the past animal husbandry was a labour intensive human activity, which involved tending animals on individual basis.

The recent progress in automation of agriculture due to advances in sensing technology allowed construction of larger dairies including milking robots, which resulted in large number of animals per stockman. Due to the increased automation of agriculture in the UK 200,000 farms disappeared between 1966 and 1995, 17,000 farmers and farm workers left agriculture in 2003 [2]. The consolidation has resulted in less personalized care and attention to the animals. This finally caused faster spreading of animal diseases such as foot-and-mouth disease (FMD) and bovine spongiform encephalopathy (BSE, also known as a mad cow disease). This has had a devastating effect on economy and has endangered health of both animals and people.

In particular eating meat from animals affected by BSE is believed to cause Creutzfeldt-Jakob disease, which is incurable and lethal to humans.

Automated monitoring of animal activity profile including walking, eating and milking allows quick detection of the oestrus as well as quick detection and prevention of the animal health disorders such as mastitis and other infection diseases, metabolic diseases, lameness and eating disorders. Prompt detection of animal diseases such as FMD and BSE can prevent spreading of these diseases and limit their devastating influence on economy and human health [3]. Timely and reliable detection of oestrus can become an efficient method of detecting pregnancy, much cheaper than currently used rectal palpation, which must be performed by a veterinary [4]. Detection of pregnancy and oestrus is essential for breeding the animals in particular when the modern farming technologies such as artificial insemination are applied [5]. The ability to be informed of changes of the animals' health, reproductive status and welfare can reduce the current reliance on manpower and improve the decision making process. High ratio of detecting oestrus can increase the conception rate of the cattle and allow wider application of artificial insemination, which improves the genetic quality of the calves [4]. These factors can considerably increase profitability of cattle farming [4, 6] and improve the quality of farmer's life [6]. The simple automated sensing systems for detecting oestrus are already commercialized and have proven their importance for the increasing profitability of farming [6].

## *1.1. Motivation*

The current sparse sensing strategy does little to capture the dynamic variability of particular animals. The current sensing systems for monitoring cattle rely on monitoring milk parameters, animal identification as well as measuring walking and mounting intensity [6-10]. They typically monitor only wet dairy cows (i.e. cows yielding milk) or have limited reliability. Animal identification is performed by utilization of RFIDs that are read over a short distance (up to 0.4 m) and only when animal's speed is limited (up to 3 m/s) [9]. This decreases the reliability of the monitoring process. The pedometers are usually read at the milking robots which limits their scope only to wet dairy cows [10]. Monitoring of milking parameters is characterised by the similar limitation. The systems that are already commercialized often rely on monitoring of a single behavioural parameter such as mounting activity [6].

Current Wireless Sensor Networking (WSN) approaches for monitoring cattle behaviour and metabolism are largely pragmatic proofs of concepts [11]. More precisely they use GSM telephony for most of their wireless communication [1], which is expensive and thus financially infeasible, or they have only single-hop communication [6]. Multi-hop communication is much more appropriate for this application. When animals are kept in pastures multi-hop communication allows animal mounted devices with shorter transmission range and fewer sinks (devices forwarding data from the animal mounted nodes to the farm servers) offering lower vulnerability to disconnections, i.e. splitting of the topology into separated islands of connectivity. Decreasing disconnections improves reliability of the monitoring process and thus increases probability of detecting animal diseases such as FMD or BSE as well as oestrus. Animal mounted devices with shorter transmission range are either smaller or characterised by longer battery life, which decreases labour intensity of their maintenance and thus makes the monitoring financially more feasible. When animals are kept in a barn multi-hop communication allows circumventing the obstacles in radio waves propagation and combat the effect of animal bodies absorbing the radio waves [2, 12]. In this way the disconnections are decreased and the reliability improved.

WSN multi-hop ad hoc routing algorithms [13-15] typically address mostly static nodes. Mobile Ad Hoc Networking (MANET) approaches are more suitable for large scale cattle monitoring because they combine advantages of multi-hop ad hoc communication with supporting high mobility of the nodes [16] and high scalability of the topology. There is a number of existing MANET approaches (e.g. [17-29]) but they are typically theoretical and driven by simulations utilizing random movement patterns of the mobile nodes. Recent studies [30] show the increasing need for the research in this area to be driven by field experiments, real data sets and emulations. We focus on realistic requirements and emulation based evaluation utilizing data from the field experiments we performed.

Within this thesis we identify realistic requirements for a scalable cattle monitoring system. Then to satisfy these requirements we design the delay tolerant architecture that provides data retention, detecting events related to animals' health, welfare and reproductive state, issuing custom notifications concerning these events, answering remote and in-situ (i.e. issued by users collocated with animals) queries. The core of this architecture is the practical MANET routing algorithm, that provides offloading data for long term storage by sending data to farm servers via sinks that are a part of MANET and handles in-situ queries issued by users collocated with the animals. We define 'practical' as being based on real requirements but also minimizing deployment and maintenance costs. The proposed routing algorithm decreases maintenance costs by extending battery life of the animal mounted devices and improves reliability of the monitoring system by combating disconnections.

## *1.2. Focus of this Thesis*

The specific focus of this thesis is: *How can we design the realistic cattle monitoring system such that:*

- It is driven by the realistic user and environmental requirements.
- It comprises a delay-tolerant architecture that provides retention of data, detecting custom events, issuing notifications, and answering remote queries so that it satisfies the identified realistic requirements.
- It has a suitable practical energy efficient wireless routing algorithm that sends data for retention and handles in-situ queries.

In the following sections we will consider these challenges in a greater detail and show how they are addressed by the novel delay tolerant architecture for the scalable cattle monitoring system and its core - the novel MANET routing algorithm proposed within this thesis.

### 1.2.1. Challenges in Identifying Requirements for a Cattle Monitoring System

In order to be optimized for a particular application the MANET approach must be informed by its realistic requirements. This constitutes user and environmental requirements. User requirements refer to the usability and interactivity of the system identifying

- which data the users need to access;
- how, where from and how often they have to access this data.

Environmental requirements refer to constraints enforced by the environment where the target system will be deployed including the movement characteristics of the mobile nodes.

In order to collect user requirements we distributed a questionnaire to the farm personnel. It showed which functionality of the system is most essential and which are the tolerable delays in reporting each type of detected events.

In order to identify environmental requirements we were tracking the movement and behaviour of the animals. We were logging and analyzing data from the GPS receivers and pedometers installed on the animals and filming these animals. This showed the typical walking speeds of the cattle and considerable differences in preferred walking speed among the particular animals. This also showed patterns of their day and night activity.

### 1.2.2. Challenges in Designing an Architecture for a Cattle Monitoring System

The research challenges concerning the practical, delay tolerant architecture for a cattle monitoring system include:

- Where to perform the processing necessary to detect oestrus, animal diseases, reduced efficiency of pastures and user defined events? Should it be centralised or distributed among sinks or animal mounted devices?
- How to provide long term data retention and remote queries?
- How do we handle in-situ queries?
- How to minimize involvement of personnel in the cattle monitoring process?

- How to improve the reliability of the target cattle monitoring system?

To address these challenges we propose a delay tolerant architecture that provides data retention, detecting custom events, issuing notifications, answering remote and in-situ queries. The proposed architecture utilizes infrastructure and ad hoc, opportunistic networking. It combines peer-to-peer and client-server paradigms.

### 1.2.3. Challenges in Designing a MANET Routing Algorithm for a Cattle Monitoring System

A practical MANET routing algorithm that supports offloading data for long term storage and answering in-situ queries is a major building block of the target application because of limited battery capacity and high mobility of the wireless nodes as well as disconnections. In order to design such an algorithm it is necessary to answer the following research challenges:

- How to deal with high mobility of animal mounted devices?
- How to deal with disconnections (i.e. splitting of the topology into separated islands of connectivity)?
- How to minimize energy utilization of animal mounted devices?
- How to balance energy utilization to prevent a subset of animal mounted devices from exhausting their remaining battery capacity sooner than the rest?
- How to evaluate the proposed routing algorithm?

To address these challenges we propose a practical energy efficient MANET routing algorithm. This algorithm provides offloading data for long term storage by sending data to the farm servers via sinks being a part of MANET and handles in-situ queries issued by users collocated with the animals. It addresses high mobility of nodes and disconnections. The proposed algorithm utilizes in a novel way heterogeneity of nodes' mobility. It also utilizes the low intensity of data traffic to save energy on route discovery for the cost of energy efficiency of data traffic. This is achieved by a novel combination of the energy efficient unicast MANET algorithm ESDSR [27] with the MANET broadcasting PCDI [31]. The disconnections are addressed using the *cooperative detection of route availability*.

## 1.3. Thesis Overview and Contributions

In the process of designing and evaluating the delay tolerant architecture and the energy efficient MANET routing algorithm for the target cattle monitoring system, we have organized this thesis in the following way:

Chapter 2 identifies criteria important for providing the scalable cattle monitoring system, in particular its wireless communication infrastructure. This chapter then reviews the relevant existing work in animal monitoring, WSNs, MANETs, DTNs, system design and management according to these criteria.

Chapter 3 describes field experiments we performed to gather the realistic requirements for the target cattle monitoring system. They are divided into qualitative and quantitative parts. The qualitative part constitutes of the questionnaire distributed to the farm personnel. It shows which functionality of the system is most essential and which are the tolerable delays in reporting each type of detected events. The quantitative part constitutes of tracking the movement and behaviour of the animals. It shows the typical walking speeds of the cattle and considerable differences in the preferred walking speed among animals. It also shows patterns of their day and night activity. Quantitative experiments produced data utilized for the evaluation of the proposed MANET routing algorithm.

Chapter 4 proposes the delay tolerant architecture that provides data retention, detecting custom events, issuing notifications, answering remote and in-situ queries. It is based on the requirements defined in Chapters 2 and 3. The proposed architecture utilizes infrastructure and ad hoc, opportunistic networking. It combines peer-to-peer and client-server paradigms in a unique way. In this chapter we also define the input and output of the cattle monitoring system and discuss the installation, usage and maintenance scenarios for this system.

Chapter 5 proposes a practical energy efficient MANET routing algorithm that supports the architecture proposed in Chapter 4 and is based on requirements defined in Chapters 2 and 3. This algorithm provides offloading data for long term storage by sending data to the farm servers via sinks being a part of MANET and handling in-situ queries issued by users collocated with the animals. It addresses high mobility of nodes and disconnections, which is crucial for improving reliability of the animal monitoring process. The proposed algorithm utilizes in a novel way heterogeneity of nodes' mobility. It also utilizes the low intensity of data traffic to save energy on the route discovery process for the cost of energy efficiency of data traffic. This is achieved by a novel combination of the energy efficient unicast MANET algorithm ESDSR [27] with the MANET broadcasting PCDI [31]. The disconnections are addressed using the proposed *cooperative detection of route availability*.

Chapter 6 evaluates the practical energy efficient MANET routing algorithm proposed in Chapter 5. This evaluation comprises emulation that utilizes animal movement patterns collected over the course of our field experiments described in Chapter 3. In particular these movement patterns are utilised to realistically emulate movement of a larger number of animals. Movement patterns from this emulation are then utilised for the packet level emulation of the proposed algorithm. We demonstrate that the proposed MANET routing algorithm provides lower and more balanced energy usage of animal mounted nodes, shorter delays of answering in-situ queries as well as increased success ratio of delivering answers to in-situ queries than the classic, non-energy aware Dynamic Source Routing (DSR) [18] and the more generic existing energy aware routing algorithm Energy Saving Dynamic Source Routing (ESDSR) [27]. This results in increasing the battery life of the animal mounted nodes and thus decreasing maintenance costs of the animal monitoring process. The most important impact on saving and balancing energy has utilization of Passive Clustering with Delayed Intelligence (PCDI) [31] to optimize broadcasting of route discovery packets and in-situ queries. PCDI allows achieving almost constant energy utilization regardless of the number of nodes. PCDI also reduces the delays in the case of in-situ queries and improves success ratio of sending replies to in-situ queries. More precisely PCDI is responsible for lower deterioration of success ratio and delays with the increasing number of nodes thus improving the scalability. Finally we argue how the proposed architecture and routing algorithm address the criteria defined in Chapter 2.

Chapter 7 discusses practical management issues of the envisaged cattle monitoring system considering the criteria defined in Chapter 2. We discuss necessary configuration performed during the system installation as well as providing mobile and home access to the target system. We propose strategies for sinks installation, connecting sinks to farm servers and improving security of the system. We also discuss necessary maintenance activities and interoperability issues.

Chapter 8 concludes the work presented in this thesis and draws out the main contributions of the thesis. It also discusses other possible applications of the proposed approach and promising directions for the future work.

# 2. Related Work

This chapter reports and classifies the existing work related to the research presented in this thesis. Section 2.1 defines and motivates the set of criteria used for reviewing existing work. Section 2.2 reviews the existing Wireless Sensor Network (WSN) approaches to animal monitoring and the energy saving techniques for WSNs. Section 2.3 presents the existing Mobile Ad Hoc Network (MANET) approaches and discusses their applicability to handling in-situ queries and sensing data to sinks. Section 2.4 reviews the existing approaches from Delay Tolerant Networks (DTNs) and discusses their applicability to handling in-situ queries, sensing data to sinks and forwarding data from sinks to farm servers. Section 2.5 reviews existing approaches to the design of wireless monitoring systems. Section 2.6 reviews existing approaches to network management. Finally, Section 2.7 discusses the methods of wireless communication which could be integrated in the design of the cattle monitoring system.

## *2.1. Criteria*

This section presents criteria used for reviewing the existing research. They can be divided into satisfying user requirements and addressing environmental constrains. Satisfying user requirements includes:

- Increasing reliability
- Managing delays
- Increasing scalability
- Lowering costs

Addressing environmental constrains includes:

- Handling high mobility

- Addressing lack of mobility supporting communication
- Handling dynamics of topology

Further within this section we discus each of these criteria in a greater detail.

### 2.1.1. Increasing Reliability

Reliability is an important requirement affecting the usefulness of the monitoring system. The important challenge from the perspective of reliability in the case of utilizing animal mounted monitoring devices is that each node monitors a different animal and each animal is important. In this way we do not have any redundancy so data from each node should be delivered to the sink and be available for querying.

In a system utilizing wireless ad hoc networking reliability is typically limited to the best effort level due to the nature of this type of communication. However we target to increase the reliability by applying the appropriate techniques such as extending range of transmitters with multi-hop communication, utilizing redundant data storage and feedback. Due to lower time constraints it is easier to increase reliability of sending data for retention and delivering notifications about detected events than answering in-situ queries.

### 2.1.2. Managing Delays

Different types of traffic have different time constrains. According to the users' requirements the acceptable delays for sending data from animal mounted devices to farm servers via sinks depend on the type of data. The urgent data includes for example information about the detected oestrus or an animal disease. Such events should be reported as quickly as possible. Non-urgent data is for example a periodic update necessary for detecting the reduced efficiency of pastures. Reduced efficiency of pastures should be reported within 48 hours. Delays for answering in-situ queries should allow the users to work interactively.

### 2.1.3. Increasing Scalability

The target system can comprise several MANETs. Each MANET can comprise from several up to approximately hundred of animal mounted devices. We consider scalability in terms of the number of MANETs in the overall topology, of the number of animal mounted nodes within each of the MANETs and of the density of the topology of a single MANET. The system should maintain the required parameters such as delays and energy efficiency within the viable range of topology size and density.

In the case of lower densities of the topologies the major challenge are disconnections. The transmitters of the nodes participating in a MANET have a certain maximal range, which is limited by their size and energy constrains. If density of the topology is low the topology can split into separated islands of connectivity. This may happen when e.g. an animal becomes ill or injured or the herd splits into separated groups. Such disconnections are challenging for the wireless communications because the multi-hop path between a pair of nodes does not necessarily always exist. Handling disconnections means thus detecting the existence of the multi-hop path and when it exists performing necessary data exchanges or routing the data in the store and forward manner or caching data and answering queries within the network partition.

In the case of higher densities of topologies or higher numbers of nodes the major challenge is combating network congestion. The congestion is usually caused by broadcasts. Therefore the most important approach to combating congestion is optimization of broadcasts by differentiating the roles of nodes in rebroadcasting packets.

### 2.1.4. Lowering Costs

Costs within this thesis refer to financial and labour costs of installation and maintenance of the target cattle monitoring system. The price of memory, microcontrollers, embedded wireless transceivers and sensors is continually decreasing. Therefore, we concentrate mostly on the costs of utilization of the third party communication services such as GSM or satellite telephony and human labour.

The major constituent of maintenance costs of the target system is replacing batteries of the animal mounted nodes. To address this, minimizing and balancing energy utilization is essential. Minimizing energy utilization means within this thesis decreasing the amount of energy utilized by animal mounted nodes. More precisely energy utilized for wireless communication because the progress in the energy efficient microcontrollers with high computation power made the energy utilized for data processing negligible [32]. Balancing energy utilization means within this thesis making the energy utilization of nodes more even. That will prevent a subset of nodes from exhausting their remaining battery capacities much sooner than other nodes due to their typical location or mobility. Replacing a battery even of a single device involves physically going to the area occupied by the animals, locating the appropriate animal, approaching it etc. This can be as labour intensive as replacing batteries of a larger number of devices.

### 2.1.5. Handling High Mobility

Most of the nodes participating in the MANET are animal mounted, others include devices carried by users and sinks. Only sinks can be stationary, which means that most of the nodes can move any time, which must be accounted for in the routing algorithm. The maximal speed of the nodes is limited in comparison to other scenarios where for example nodes are mounted on vehicles. However in the road traffic scenarios cars move with similar speed in the same direction so they are almost static in relation to each other. This is not the case with the animals, which move in a far less organised fashion. The mobility of animals is also far less predictable than in the case of people or cars.

Within this thesis high mobility refers to frequent changes of topology. This results in data about the structure of the topology, which is necessary for routing, becoming quickly obsolete. Handling high mobility thus means using soft state topology data which is collected in the demand driven way, i.e. when there is data to be routed.

### 2.1.6. Addressing Lack of Mobility Supporting Communication

In numerous applications of wireless networking technologies mobile nodes are carried by people who have incentives to move in the way supporting connectivity. This usually means standing or moving in close proximity as well as approaching a spotted individual that is believed to carry a compatible wireless node. On the contrary animals carrying the monitoring devices are not going to move in the way supporting the wireless connectivity. Therefore the routing algorithm should work in spite of their mobility or try to utilize their typical, hard to predict, mobility. Users performing in-situ queries may try to approach the animals of interest. However a user may only know that the animal of interest is located in the specific pasture without any easy way of identifying it. In such case we can only expect that she will approach the biggest or random aggregation of the animals.

### 2.1.7. Handling High Dynamics of Topology

In our scenario the topology of the sensor nodes is very dynamic and can rapidly change from very sparse to very dense. This can be attributed to behaviour of animals such as fleeing caused by fear but also different environments where animals are kept and moved between such as barns and pastures. In a sparse topology the multi-hop path between a sensor node and the sink may be available only sporadically. In a dense topology there is a risk of network congestion especially in the case of broadcasts – broadcasts storms. In order to address these challenges the routing algorithm should automatically adapt to the changing topology in the self organised fashion.

## *2.2. Wireless Sensor Networks*

### 2.2.1. Application to Animal Monitoring

This section discusses existing Wireless Sensor Networks (WSNs) for animal monitoring. The WSNs [33] consist of hundreds to thousands of inexpensive wireless nodes, each with some computational power and sensing capability, operating in an unattended mode. The hardware technology for these networks are low cost processors, miniature sensing and radio modules. Sensor data includes continuous sensor readings of physical phenomena, audio and video streams.

#### 2.2.1.1. Stationary Wireless Sensor Networks

The initial WSNs were purely stationary. The sensor data was archived in a powerful server geographically collocated with the sensors (usually referred to as a base station) that was usually fully replicated on the pre-determined powerful servers in the labs. Users could query the databases to get information about sensor data.

An example stationary WSN was the WSN deployed on the Great Duck Island [34] to monitor the ecology of Leach's Storm Petrel. It used single-hop communication and had a multi-layer architecture as shown in Figure 2.1. The fist layer consisted of multiple sensor networks that were deployed in dense patches that were widely separated and measured various physical phenomena and had cameras and microphones. Each sensor patch had sensor motes that were capable of various forms of filtering, sharing and combining sensor measurements. Sensor motes transmitted sensor data to the second layer that is referred to as a gateway. A gateway was then responsible for transmitting the packets to the third level referred to as the base station and some further data processing. The base station in the third level provided full database services and connectivity to the database replicas across the Internet. Fourth layer usually refers to services that provide multi-user access to sensor data including services for supporting analysis, visualization and web content. Once deployed, most base stations are intended to remain stationary and in a densely packed configuration.

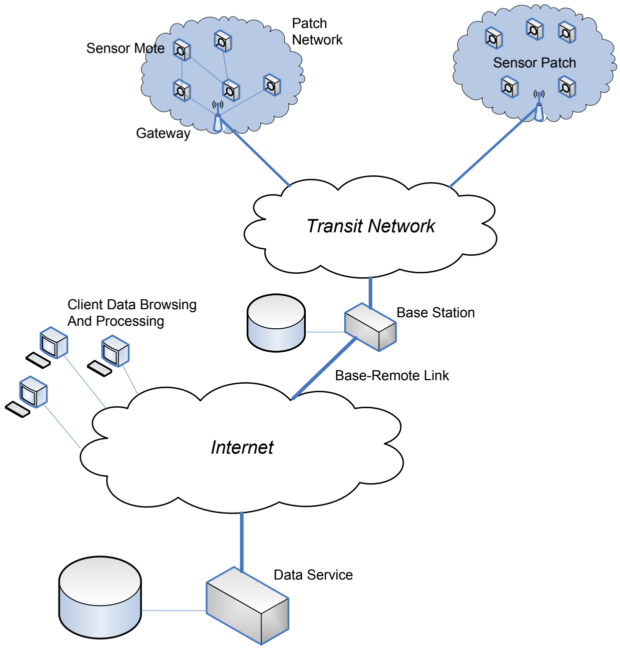


**Figure 2.1. WSN deployed on Great Duck Island**

WSN deployed on the Great Duck Island comprised 43 sensor nodes and its maintenance was characterized by low labour intensity. Its stationary character allowed simplification of the routing and avoiding problems with mobility and disconnections. The simple routing and lack of disconnections helped in avoiding problems with energy saving. Lack of disconnections and problems with energy saving allowed short delays. This approach because of its stationary character does not apply to our scenario.

### 2.2.1.2. Animal Mounted

In a typical animal mounted WSN mobile nodes send measurements to a centralized server over a GSM or satellite network. Alternatively the measurements are collected by a mobile base station carried by a human or mounted on a vehicle and then manually processed.

The oldest form of animal mounted wireless sensors are radio tags, which send VHF beacons [35]. Their measurements are retrieved by a base station which can be fixed, carried by a human or mounted on a vehicle. This approach is not optimal for our scenario because using fixed base stations is expensive in the case of covering larger areas. Using base stations carried by humans or mounted on vehicles is very labour intensive. In both cases potentially data from only a subset of tagged animals can be retrieved. The more recent variant of this method [35] is using satellite telephony instead of VHF beacons. This is much less labour intensive and more reliable but also very expensive and energy inefficient.

One of the first examples of animal mounted WSNs was ZebraNet [36]. Its objective was to monitor locations of zebras in the Mpala Research Center in Kenya. ZebraNet consisted of animal mounted collars collecting and exchanging GPS locations, which were retrieved by a mobile base station. The collars were opportunistically exchanging all stored measurements with all encountered nodes. This addressed disconnection but limited scalability – the maximal envisaged number of the deployed animal mounted nodes was 30. Due to the Delay Tolerant Networking (DTN) type of communication the delays could be long but it was not against the requirements as the urgency of data was limited. The required labour intensity of the system maintenance was increased by the fact that the base station was mounted on a vehicle driven by humans. This approach does not address our requirements because of low scalability and high labour intensity.

The authors of [1] mounted various sensors on a single steer to monitor temperature inside its rumen, location, acceleration, as well as external temperature, humidity and pressure. The measurements from the sensors were transmitted to the gateway mounted on the animal, which forwarded them on via GPRS. The presented approach was expensive and not energy efficient because of extensive utilization of GPRS. Low energy efficiency increased the labour intensity of its maintenance. Authors proposed using renewable energy sources such as solar cells but this does not apply to animals kept in barns. The GSM telephony can have limited coverage in rural areas where the cattle is kept [1]. Therefore, utilizing this type of communication is not reliable. This approach does not address our requirements because due to heavy utilization of GSM it has high costs and low energy efficiency.

Butler et al. [12] proposed using animal mounted devices to force bovine animals to move or stay within virtual fences. Whenever an animal was close to the border of the designated area the device was producing a deteriorating sound. The animal mounted devices communicated over 802.11 with the base station in the multi-hop manner but the emphasis of the referenced work is on the automatic control aspect of the proposed approach rather than on wireless communication. Authors do not address the energy efficiency of the wireless communication and thus do not address our requirements.

Researchers at CSIRO [2, 37] performed research on utilizing animal mounted sensor nodes to cattle monitoring. They fitted 13 cows with collars containing accelerometers, GPS receivers and wireless networking interfaces. Their objective was to examine reliability of the communication and usability of the data collected by GPS receivers and accelerometers. The authors do not give details about the utilized routing algorithm and do not consider the energy efficiency in their approach. The maximal lifetime of an animal mounted device was 4.3 days. The later work of these researchers [38] concerns using animal mounted devices to prevent bulls from fighting with each other. The animal mounted collars have GPS receivers, wireless network interfaces and are capable to apply electric shocks to the animals wearing them. Whenever a bull is trying to attack another one this fact is detected by its collar and the electric shock is applied. The detection of the approaching fight is performed on the base of the GPS readings from the local and neighbour collars, the latter received over the wireless connections. The utilised wireless communication is a simple single-hop one without considering energy efficiency. Therefore, this approach is not optimal for our scenario.

Small et al. [39-41] proposed using whale mounted sensors to collect data about whales and their habitat. They utilized a combination of the *Infostation* [42, 43] paradigm and a DTN approach similar to Gossiping [44]. In the *Infostation* model, users can connect to the network in the vicinity of *Infostations*, which are geographically distributed through the area of network coverage. In the Shared Wireless Infostation Model (SWIM) proposed by Small et al., similarly as in Gossiping [44] nodes exchange packets with some predefined probability whenever they meet. When any of the nodes meets one of the *Infostations* it offloads all its carried data – generated by itself and forwarded. The *Infostations* are buoys mounted at the whales' feeding areas, they can be also carried by sea birds. In other words, the SWIM paradigm is similar to communication utilized by the ZebraNet [36] but SWIM limits the probability of forwarding data to other nodes. In our scenario animal mounted devices form a much denser topology then in the case of whale monitoring. Therefore, gossiping would increase the network overhead and thus affect energy efficiency.

### 2.2.2. Energy Efficiency

Minimizing and balancing energy utilized by animal mounted nodes is very important for minimizing maintenance costs of the target cattle monitoring system. Therefore, in this subsection we present the energy saving techniques for WSNs and discuss their applicability for sending data from animal mounted nodes to sinks and handling in-situ queries.

There are many approaches addressing energy efficiency of WSNs. Typically they consider only stationary nodes. One of them is reducing network overhead by data-centric routing. In Directed Diffusion [13, 45] the sink broadcasts the interest message across the network. Such a message contains information about the type of events the sink wants to be informed about – initially at a low interval. This initial broadcast sets up gradients which are used to forward exploratory messages from the nodes which detect the events to the sink. The sink then reinforces the paths which generate events with the lowest delay. Then the events are sent at a higher rate along the reinforced paths. The gradients are soft state, i.e. they expire after a timeout. Broadcasting of the interests and exploratory events can be optimized [46] by utilization of Passive Clustering [47, 48] described in the Subsection 2.3.5.5. In general Directed Diffusion is not optimal for our scenario because it does not handle high mobility of sensor nodes and assumes their high redundancy. In our approach each animal is a source of valuable measurements and is associated with a single node in the sensor network. In this way we do not have redundancy.

Another approach to energy saving in WSNs is controlling transmitter power. Kubisch et al. [49] proposed several algorithms, local and centralized, which aim at achieving number of neighbours from a given range or a fully connected network. The authors assume that the nodes are stationary and thus the transmitter power is set during the network's initialization phase. Although we apply controlling transmitter power in our approach, algorithms proposed by Kubisch et al. are not appropriate for our scenario because they do not address mobility of the nodes.

Aslam et al. [50] proposed three routing algorithms which balance energy utilization and find routes consuming up to $z$ times more power than the possible minimum. These algorithms consider minimal power required to send data between neighbours and remaining battery capacity. The first one is fully centralized, the second is centralized but hierarchical and the third is fully distributed. The authors proposed an adaptive method of calculating $z$ but it is centralized. These approaches are not appropriate for our scenario because they only considered stationary nodes.

Guo proposed an energy efficient broadcasting for WSNs [51]. Nodes rebroadcast non duplicated packets with the delay proportional to their remaining battery capacity. In this way, nodes with higher remaining battery capacity are more likely to rebroadcast, which balances energy utilization. The energy is saved by not rebroadcasting duplicated broadcasts. This approach is similar to the Passive Clustering with Delayed Intelligence [31] described in Section 2.3.5.5, which we adapted to our scenario.

## *2.3. MANET Routing Algorithms*

Animal mounted nodes and users issuing in-situ queries are mobile. Therefore, it is relevant to review here the existing routing algorithms for Mobile Ad-hoc Networks

(MANETs) and discuss their applicability to sending data from animal mounted nodes to sinks and handling in-situ queries.

### 2.3.1. Proactive Routing

MANETs [16] consist of mobile and stationary wireless nodes. Each of them can forward packets sent by other nodes. This section discusses proactive MANET routing algorithms. Proactive MANET routing algorithms continuously maintain the routes to all possible destinations.

One of the first such algorithms and MANET routing algorithms in general is DSDV [17]. In DSDV each node maintains the routing table comprising an entry for each possible destination. Such entry comprises the address of the next hop in the route, number of the route's hops and the sequence number. Each mobile node periodically broadcasts its full routing table (*full dump*). If necessary the incremental changes to the last full dump can be broadcasted before the next full dump. Routes with more recent sequence number are preferred. If the sequence number is the same then the routes with the lower number of hops are preferred. The necessity to continually maintain routes to all the destinations affects the energy efficiency, particularly when the data exchanges are infrequent as in our scenario. Regular broadcasts can lead to broadcast storms in the larger topologies which affects scalability. In the conditions of higher mobility of nodes the broadcast storms can be also caused by the incremental updates to the full dumps resulting from the broken routes. This routing algorithm is not optimal for our scenario because of its low scalability and energy efficiency.

### 2.3.2. On-demand Routing

Dynamic Source Routing (DSR) [18] was the first on-demand MANET routing algorithm. The paths are discovered whenever a node needs to exchange data with a node to which it does not have a path in its cache. In such case the node broadcasts a route request packet (RREQ). Each node rebroadcasts it at most once and upon rebroadcasting appends its address. If the recipient of the RREQ packet is the destination node or has a valid route to the destination in its cache instead of rebroadcasting, it responds with the route reply packet (RREP). The RREP packet contains a path from the source to the destination and is sent back to the source node. Nodes can also collect paths from forwarded packets. The routes are removed from the cache after a timeout or whenever any of their hops is broken. In the latter case a node that discovers a broken hop propagates this information back along the route. DSR is not optimal of our scenario in its original form because it does not address the energy efficiency. The optimality of the discovered routes is achieved at the cost of extensive broadcasting of RREQ packets. This approach is suboptimal in our scenario, where the amounts of data are limited.

Ad-hoc On-demand Distance Vector Routing (AODV) [19] is another ad hoc routing algorithm. It works similarly to DSR but instead of accumulation of the path details in RREQ packets, in particular the addresses of intermediate nodes, AODV establishes route table entries at intermediate nodes. In this way, in the case of routes with a higher number of hops the control traffic is lower than in the case of DSR [18]. From the standpoint of our scenario this routing algorithm has similar limitations as DSR.

Temporarily-Ordered Routing Algorithm (TORA) [20] is an on-demand multipath routing algorithm for MANETs. It is an adaptation of a 'link-reversal' type of algorithms to the MANET environment. For each required destination TORA builds a directed acyclic graph (DAG) rooted at the destination. As a multipath routing algorithm TORA is more optimized for throughput rather than energy efficiency. More precisely it requires extensive broadcasting to build and maintain the DAG – separate for each destination. In the cattle monitoring scenario the amounts of data are low so the single path routing is sufficient. Energy efficiency is much more important than throughput. TORA also requires precise synchronization of clocks. As the animal mounted devices in the target system do not have GPS receivers, clock synchronization would require additional network overhead, which is expensive in terms of battery usage.

### 2.3.3. Geographic Routing

There is a body of work in the area of MANETs concerning geographic routing [52-54]. In geographic routing route discovery or packet forwarding utilizes the physical location of nodes. For example in Location-Aided Routing (LAR) [52] the route discovery is similar to DSR [18] and AODV [19] with the following exception. The source node approximately knows the current location of the destination node, in particular the *expected zone* where the destination node is likely to be found. Basing on the *expected zone* source node defines the *request zone*. In the process of route discovery, route requests are only rebroadcasted by the nodes located in the *request zone*.

Zone-based Hierarchical Link State (ZHLS) [53] routing algorithm is proactive and hierarchical. The hierarchy has two levels: inter zone and intra zone. Zones are geographical and the nodes establish their zone membership from GPS readings. Using the periodic broadcasts nodes maintain routing tables with routes to all nodes within their zones and the connections between zones.

GRID [54] is an on-demand MANET routing algorithm similar to AODV [19] with the following differences. The topology is divided into grids and their membership is assigned basing on the GPS readings. In each grid one of the nodes is dynamically elected as a gateway. Only gateways are responsible for forwarding packets. Similarly as in LAR [52], broadcasting of route discovery packets is geographically limited.

In Greedy Perimeter Stateless Routing (GPSR) [55] nodes are assumed to know their location in relation to other nodes. This allows forwarding the packets to nodes which are physically closer to the destination. This greedy approach does not allow circumventing void spaces in the topology, where a packet can reach a local minimum of proximity to the destination. Such void spaces, when encountered, are circumvented using the right-hand rule, i.e. the packet is routed around the void in the counter-clockwise direction.

In general geographic routing algorithms allow reducing network overhead or even decreasing the soft state nodes have to store. They typically require knowledge about the relative physical location of the nodes. This makes geographic routing algorithms not suitable for our scenario, where the mobile nodes do not know their location. In order to know the location they would have to have GPS receivers, which consume considerable amounts of energy. Alternatively we would need dedicated static nodes knowing their location to triangulate locations of the mobile nodes. This would make the deployment much more expensive.

### 2.3.4. Energy Efficiency

Minimizing and balancing energy utilized by animal mounted nodes is very important for minimizing maintenance costs of the target cattle monitoring system. Therefore, in this subsection we present the energy saving techniques for MANETs and discuss their applicability for sending data from animal mounted nodes to sinks and handling in-situ queries.

The initial work in energy efficiency of MANET routing concerned route selection metrics. The metric in Minimum Total Transmission Power Routing (MTPR) [21] calculates total transmission power for the route:

$$P_l = \sum_{i=0}^{D-1} P(n_i, n_{i+1}), \tag{2.1}$$

where $D$ is the total number of hops and $P(n_i,n_{i+1})$ is the minimal power necessary to transmit a packet between subsequent hops. MTPR selects the route with the minimal value of $P_l$. In this way the overall energy consumed by the transmission of a packet is minimized.

In the next paper the same authors notice that MTPR can lead to unfair utilization of the nodes' batteries. In particular a subset of the nodes is likely to utilize their battery capacity sooner. The authors propose the Minimum Battery Cost Routing (MBCR) [22], where the route metric is calculated using the following formula:

$$R_j = \sum_{i=0}^{D_j - 1} \frac{1}{c_i^t}, \tag{2.2}$$

where $D_j$ is the total number of hops for the route $j$ and $c_i^t$ is the battery capacity for the $i$th node on the route at the time $t$. MBCR selects a route with the minimal value of $R_j$. This metric considers the remaining battery capacity of the nodes but because of the summation some of the nodes can still be overused if they are combined in one route with nodes with large battery capacity. Therefore, in the same paper [22] an alternative metric, Min-Max Battery Cost Routing (MMBCR), was proposed. It is calculated according to the following formula:

$$R_j = \max_{i \in route\_j} \frac{1}{c_i^t}. \tag{2.3}$$

The route with the minimal value of $R_j$ is selected. In this way, nodes with the lowest remaining battery capacity are avoided.

Toh noticed [23] that because the MMBCR metric does not consider transmission power, very suboptimal routes in terms of energy utilization can be selected. He proposed Conditional Max-Min Battery Capacity Routing (CMMBCR) [23]. In CMMBCR if there is a set of routes, where all nodes between the source and the destination have remaining battery capacity higher than a predefined threshold, a route from this set is selected according to the MTPR metric [21]. Otherwise a route is selected according to the MMBCR metric [22]. In this way the routes with lower energy cost are preferred but also the nodes with very low battery capacity are avoided.

Dongkyun Kim et al. noticed [24] that all the previously described schemas consider battery capacity only before data exchange starts. They proposed a new schema, Minimum Drain Rate Mechanism (MDR) [24], for monitoring battery capacity during the data transfer. If an alternative route becomes more optimal it is used to finish the transfer – this is particularly useful for longer connections. They propose the following metric:

$$L_j = \min_{i \in route\_j} \frac{RBP_i}{DR_i}, \tag{2.4}$$

where $RBP_i$ is the remaining battery capacity of the node $i$ on the route $j$ and $DR_i$ is the drain rate of the node $i$. Drain rate means the energy consumption per time unit. The route with maximal value of $L_j$ (i.e. lifetime) is selected. $DR_i$ can be calculated using the exponential weighted moving average:

$$DR_i = \alpha \times DR_{old} + (1-\alpha) \times DR_{sample}, \tag{2.5}$$

where $\alpha$ is the weight e.g. 0.3, $DR_{old}$ is the previous average and $DR_{sample}$ is the newly measured value. Such measurement can be taken every predefined time $T$, e.g. 6s. MDR does not consider transmission power but this can be addressed by CMMBCR [23] using MDR instead of MMBCR [22].

Maleki et al. proposed a DSR based Power-aware Source Routing (PSR) [25]. This routing algorithm had a new metric for route selection:

$$\begin{aligned} C(\pi,t) &= \sum_{i \in \pi} C_i(t) \\ C_i(t) &= \rho_i \cdot \left( \frac{F_i}{R_i(t)} \right)^{\alpha}, \end{aligned} \tag{2.6}$$

where $\rho_i$ is the transmission power of the node $i$, $F_i$ is the full-charge battery capacity of the node $i$, $R_i(t)$ is the remaining battery capacity of the node $i$ at time $t$ and $\alpha$ is a positive weighting factor. The authors also propose a mechanism for calculating this metric. The cost of the route is carried in the RREQ packets and each rebroadcasting node increments it by the calculated $C_i(t)$. Each node waits for the time $T_r$ before rebroadcasting a RREQ packet and then rebroadcasts the one with the lowest total cost ($C(\pi,t)$). An intermediate node forwarding the traffic sends the route error to the source node when the following condition is met:

$$C_i(t) - C_i(t_0) \geq \delta\,, \tag{2.7}$$

were $t$ is the current time, $t_0$ is the discovery time and $\delta$ is the predefined threshold. In this way too quick depletion of an intermediary node's battery capacity is prevented.

In the later paper [26] Maleki et al. proposed Lifetime Predication Routing (LPR), similar to PSR but with a different metric for the route selection. In particular each node that receives RREQ calculates its predicted lifetime $T_i$ using the formula:

$$T_i(t) = \frac{E_{r,i}(t)}{\frac{1}{N-1}\sum_{k=i-N+1}^{i} R_k(t)}\,, \tag{2.8}$$

where $E_{r,i}(t)$ is the remaining energy at the $i$th packet sent or forwarded, $R_k(t)$ is rate of energy depletion of the current node when the $k$th packet was sent, calculated as the ratio of difference between residual energy during sending packets $k$-1 and $k$ and the difference of arrival time of these packets. $N$ is the length of the history used for the calculation. Then if $T_i$ calculated by the node is smaller than $T_i$ in the header of the broadcasted RREQ packet, the value in the RREQ packet is replaced by the calculated $T_i$. After waiting for the time $T_r$ since the arrival of the first RREQ packet for a given route discovery the node rebroadcasts one with the maximal $T_i$ in the header. An intermediate node forwarding the traffic sends the route error to the source node when the following condition is met:

$$T_i(t_0) - T_i(t) \geq \delta\,, \tag{2.9}$$

where $T_i$ is predicted lifetime of the node $i$, $t$ is the current time, $t_0$ – route discovery time and $\delta$ – a predefined threshold.

Tarique et al. proposed Energy Saving Dynamic Source Routing (ESDSR) [27] based on DSR [18]. ESDSR controls transmitter power in the following way. Senders record transmission power in packets they send. Receivers can then tell senders the minimum power level required for communication. Therefore, senders can adaptively adjust transmission power levels to suit the current need, rather than using a fixed level. More precisely, nodes upon receiving of the packets can calculate minimal energy necessary to reach their single-hop neighbours (these associations are stored in the power table) using the following formula:

$$P_{min} = P_{tx} - P_{recv} + P_{threshold} + P_{margin}, \tag{2.10}$$

where $P_{min}$ is the minimal required power for the sender to use, $P_{tx}$ is the current transmit power, $P_{recv}$ is the current received power, $P_{threshold}$ is the threshold power level for the application, and $P_{margin}$ is the margin to safeguard against changes such as channel fluctuation and mobility. All the values are in dBm.

ESDSR uses the route with maximum remaining lifetime. Remaining lifetime of a node in a route is defined as remaining node energy divided by power required to transmit a packet to the next node in the route. Remaining lifetime of a route is then minimum of remaining life of nodes in the route. Following the notation used in [27],

$$\begin{gathered} C(R,t) = \max_j R_j(t) \\ R_j(t) = \min_i E_i(t) / P_{ij}(t) \end{gathered}, \tag{2.11}$$

where $t$ is time, $E_i(t)$ is the remaining energy of the node $i$ assumed to be known from hardware and $P_{ij}(t)$ is the transmit power of the node $i$ in the route $j$ as stored in the received packet.

Similar power control mechanism as in ESDSR has been proposed by Bergamo et al. [28] and called Distributed Power Control (DPC). Contrary to ESDSR [27] they apply it to AODV [19] not DSR [18]. For selecting routes they only consider transmission power of all nodes along the route and not their battery capacity. In particular the total power of the route is saved in RREQ and RREP packets. Subsequent RREQ for the same route discovery are rebroadcasted if they represent a route with a lower energy cost.

In general, in all the routing algorithms presented above the energy efficiency is achieved by application of the route selection metrics or, as in ESDSR [27] or DPC [28], by controlling transmitter power. Concerning route selection metrics it is always assumed that the amounts of exchanged data are high and most of the energy is consumed by data traffic not the control traffic for the route discovery. This is not the case in our scenario, where amounts of exchanged data are small. Due to this assumption these approaches are not optimal for our scenario. Controlling transmitter power gives good results regardless of the amounts of data traffic so we utilize this technique in the proposed approach.

Another approach to energy conservation in MANETs is turning off wireless receivers to save energy consumed by the receiver in the listening or idle state. This can be done on the MAC level like in PAMAS [21], where nodes turn off their receiver after detecting that the packet they receive is addressed elsewhere.

This technique can be also applied at the routing level, in which case the energy efficiency is increased at the cost of longer delays. Such approach has been proposed by Ya Xu et al. [29]. The authors propose two energy conserving algorithms. In the first of them, Basic Energy-Conserving Algorithm (BECA), a node sleeps for the time $T_s$ and then starts listening. In the listening phase it forwards the traffic as requested by other nodes. If there is no traffic for the time $T_l$ a node comes back to the sleeping phase for the time $T_s$. This approach may cause route discovery process to fail so failed route discovery is repeated $R$ times every time $T_0$. In particular $T_l$ is equal to $T_0$ and $T_s$ is a multiple of $T_0$. The second algorithm proposed by the authors, Adaptive Fidelity Energy-Conserving Algorithm (AFECA) is similar to BECA with the following difference. During the listening phase a node counts its single-hop neighbours. The sleeping phase then does not last for $T_s$ but for

$$T_{SA} = Random(1, N) \times T_s, \tag{2.12}$$

where $N$ is the number of single-hop neighbours.

The technique of switching receivers off described above relies on the assumption that the energy saved by keeping the receiver switched off is greater than the energy cost of switching it on and that time of switching the receiver on is much shorter than the sleeping phase. This does not have to apply to all wireless platforms. BECA and AFECA algorithms also assume no disconnections, i.e. splitting of the topology into separated islands of connectivity. A disconnection would force a node to repeat frequently route discovery for the $R$ times, which would be expensive in terms of energy usage. For these reasons we do not utilise this technique in our approach.

### 2.3.5. Broadcast Techniques in MANETs

Due to the limited amounts of exchanged data in our scenario, route discovery and broadcasting of in-situ queries are important constituents of energy utilization of the animal mounted nodes. Therefore improving energy efficiency of these processes is an important part of our work. As these processes are forms of broadcasting it is relevant to briefly summarize here the existing techniques of broadcast optimization in MANETs and show which of them can be utilized in our scenario. Our categorization of the broadcast optimization methods is the extension of the one proposed in [56].

#### 2.3.5.1. Simple Flooding

The most basic and the most expensive in terms of network overhead method of broadcasting is simple flooding [18]. The source broadcasts a packet to all neighbours, which then rebroadcast it, typically only once.

#### 2.3.5.2. Probability Based Methods

One type of possible optimizations to the simple flooding are probability based methods. The most basic Probabilistic Scheme [57] is similar to the simple flooding but nodes rebroadcast with certain probability. This approach decreases the network overhead but decreases reachability (i.e. fewer nodes will receive the broadcasted packet) and does not consider the dynamic density of the topology. This is why it is not optimal for our scenario.

In a more advanced approach, Counter-Based Scheme, nodes wait a random delay before rebroadcasting [57] and then rebroadcast with the probability reversely proportional to the number of received duplicates. The Counter-Based Scheme considers local density of the topology but still can decrease the reachability of the broadcast and some of the available potential is underutilised. In particular the duration of the delay can convey some information for better optimization of broadcasting. Therefore we do not use this technique in our approach.

Gossiping in wireless ad hoc networks [44] is another form of probabilistic method. In the basic mode, GOSSIP1($p,k$), the packet is broadcasted for $k$ hops with probability 1, and then further with probability $p$. A more advanced scheme proposed by the same authors, GOSSIP2($p_1$, $k$, $p_2$, $n$), works similarly to the previous one but if a node has fewer than $n$ neighbours it rebroadcasts with probability $p_2$ instead of $p_1$. GOSSIP3($p,k,m$) is a modified version of GOSSIP1($p,k$), in the way that if according to probability $p$ a node decides not to rebroadcast but receives less than $m$ duplicates of the broadcasted packet it rebroadcasts the packet.

### 2.3.5.3. Area Based Methods

In area based methods nodes try to cover with the broadcast as large area as possible with the minimum number of broadcasted packets. In the Distance-Based Scheme [57] nodes wait a random delay from receiving the first packet for a given broadcast. If any of the packets received during this delay is transmitted over a distance shorter than the predefined threshold the node does not rebroadcast. The distance can be measured by geolocating (e.g. GPS) or measuring signal attenuation. This approach has still a considerable degree of randomness introduced by random duration of the delay, which could convey some information. The predefined distance is difficult to estimate. If it is too short this method gives little advantage over simple flooding. If the distance is too long, then it may be difficult for a broadcast to traverse the gaps in the network topology. More precisely, the nodes on the edge of the gap will not rebroadcast due to receiving the broadcasted packets from nodes which are too close to them. For these reasons we do not use this approach.

Another area based method is Location-Based Scheme [57]. In this method nodes must have some means of finding their geographic location. Whenever a node broadcasts or rebroadcasts a packet it stores the geographic coordinates in this packet. At the end of the random delay a node waits after receiving the first packet for a given broadcast, it then calculates the area it can cover with its broadcast which has not been covered by the packets it has received. Then it rebroadcasts only when the size of this area is larger than the predefined threshold. We do not apply this technique to our approach because our animal mounted nodes do not have the means of geolocating.

### 2.3.5.4. Proactive Neighbourhood Knowledge Methods

The proactive neighbourhood knowledge methods optimise broadcasting basing on the single-hop and two-hop neighbourhood knowledge acquired from the exchange of periodic 'Hello' packets. Some of them (e.g. these from [58]) incorporate also probabilistic techniques.

The simplest from these methods is Flooding with Self Pruning [59]. The nodes know their 1-hop neighbours from the periodic 'Hello' packets. Whenever a node broadcasts or rebroadcasts a packet it stores in this packet the list of its neighbours. Whenever another node receives such packet it rebroadcasts this packet only if it has 1-hop neighbours not listed in this packet.

In Scalable Broadcast Algorithm (SBA) [60] nodes put the list of their 1-hop neighbours into the periodic ‘Hello’ packets. In this way each node has a list of its single-hop and two-hop neighbours. After receiving a first packet of the given broadcast a node waits a random delay and then decides if the broadcasted packet should be rebroadcasted. Knowing the ids of the single-hop neighbours that sent the packets it received and having the knowledge about its single-hop and two-hop neighbourhood it can find out which other nodes also received these packets. Then the node rebroadcasts the packet if it is going to be received by nodes not covered by the packets it received. The random delay $t_d$ in SBA is calculated using the following formula:

$$t_d = \frac{d_{N\,max}}{d_{me}}, \tag{2.13}$$

where $d_{me}$ is the number of 1-hop neighbours of the rebroadcasting node and $d_{Nmax}$ is the maximal number of 1-hop neighbours of any 1–hop neighbour of the rebroadcasting node.

In Dominant Pruning (DP) [59] the knowledge of 2-hop neighbourhood is acquired similarly as in SBA. Contrary to SBA nodes proactively put the list of the nodes that should rebroadcast the packet into the broadcasted or rebroadcasted packets. In particular when a node receives such packet and is included in the list, it uses Greedy Set Cover (GSC) algorithm to establish the next subset of nodes that should rebroadcast this packet. GSC recursively chooses 1-hop neighbours which cover the largest number of 2-hop neighbours and recalculates the cover set until all 2-hop neighbours are covered.

Multipoint Relying [61] is similar to DP, differing only in the algorithm for selecting the nodes that should further rebroadcast the packet – called here Multipoint Relays (MPRs). This algorithm is following:

1) Find all 2-hop neighbours that can only be reached by one 1-hop neighbour. Assign these 1-hop neighbours as MPRs.
2) Determine the resulting cover set (i.e. the list of 2-hop neighbours that will receive the packet from the current MPRs).
3) From the remaining 1-hop neighbours find the one that would cover the maximal number of 2-hop neighbours outside the cover set.
4) Repeat steps 2-4 until all 2-hop neighbours are covered.

The Lightweight and Efficient network-Wide Broadcast (LENWB) [62] also uses 'Hello' packets to collect knowledge about 2-hop neighbourhood. The decision about rebroadcasting the packet is taken considering which single-hop and two-hop neighbours are going to rebroadcast this packet. This knowledge comes from the information about which nodes received the packet from the common source and which are the priorities of neighbours. The priority of a node is proportional to the number of neighbours. A node can calculate if all its lower priority neighbours will receive a broadcasted packet from the higher priority ones and if this is not the case it rebroadcasts the packet.

Cartigny and Simplot proposed [58] several schemes which combine probabilistic methods with proactive neighbourhood knowledge methods. We categorise them as proactive neighbourhood knowledge methods because they utilise periodic 'Hello' packets. Their first scheme, Density Aware Probabilistic Flooding (mode 2), is very similar to Probabilistic Scheme [57]. The only difference is the exchange of 'Hello' packets and the rebroadcasting probability calculated as:

$$f_2(n) = \frac{k}{n}, \quad (2.14)$$

where $n$ is the number of neighbours and $k$ is a predefined constant.

In the next scheme, Border Retransmission Based Probabilistic Flooding (mode 3), the probability of rebroadcasting depends on the distance between the rebroadcasting node and the previous node which sent the broadcasted packet. The metric $\mu$ for the distance is the difference in 1-hop neighbourhood and is calculated using the formula:

$$\mu = \frac{N_b}{N_a - N_c}, \quad (2.15)$$

where $N_a$ is the number of 1-hop neighbours of the source node, which are not 1-hop neighbours of the destination node; $N_b$ is the number of 1-hop neighbours of the destination node which are not 1-hop neighbours of the source node; $N_c$ are the common 1-hop neighbours of source and destination nodes. The rebroadcasting probability is then:

$$f_3(\mu) = 1 - \left(\frac{\mu}{M}\right)^{\sigma}, \quad (2.16)$$

where, $M$ is a constant representing maximal value of $\mu$ and σ is a constant called by authors coefficient of convexity (e.g. 0.5, 1 or 2).

Density Aware and Border Node Retransmission Based Probabilistic Flooding (mode 4) is a combination of modes 2 and 3. In particular the rebroadcasting probability is calculated using the following formula:

$$f_4(\mu,k) = \frac{\frac{k}{n}-1}{M^{\sigma}}\mu^{\sigma} + 1, \tag{2.17}$$

where the meaning of symbols is the same as in Formulas 2.14 – 2.16.

Finally, the authors proposed Density Aware and Border Node Retransmission Based Probabilistic Flooding with Neighbour Elimination (mode 5). Its objective is increasing reachability of the mode 4 scheme. Each node includes the list of its 1-hop neighbours in the broadcasted or rebroadcasted packets. Then a node which receives such packet, if it does not rebroadcast this packet according to the probabilistic function from Formula 2.17, stores in its broadcast table (BT) an association between the broadcast id and the list of its 1-hop neighbours. For each received broadcasted packet it removes the sender of the packet and the 1-hop neighbours of the sender listed in the packet from the list of nodes associated with the broadcast id in BT. After the time *T* from receiving the first packet of a given broadcast the node checks if the list of nodes in BT associated with the id of the broadcast is empty. If not, the node rebroadcasts the packet. Time *T* is calculated according to the formula:

$$T = T_{max} + T_{alea} * x, \tag{2.18}$$

where $T_{max}$ and $T_{alea}$ are predefined constants and $x$ is a random real number from 0 to 1.

To summarize, all the presented proactive neighbourhood knowledge methods [58-62] require exchange of 'Hello' packets between wireless nodes to provide them with knowledge about their neighbourhood. When the nodes are highly mobile and the data exchanges are relatively infrequent as in our scenario the 'Hello' packets can negatively influence battery lifetime or fail to provide up to date information about the topology. For this reason we do not use these methods in our approach.

### 2.3.5.5. Passive Neighbourhood Knowledge Methods

The passive neighbourhood knowledge methods optimise broadcasting basing on the single-hop neighbourhood knowledge. This knowledge is acquired from the data piggybacked onto the broadcasted packets. In this way no periodic ‘Hello’ packets are required. This energy saving comes usually at the cost of longer delays required for the nodes to receive the information about neighbourhood sufficient to make a decision about rebroadcasting. Typically these delays only affect the first broadcast of a temporally framed series because after the first broadcast from the series the nodes can reuse the topology data they collected.

Passive Clustering (PC) [47, 48] creates soft-clusters by determining clusterhead nodes without complete neighbourhood knowledge. A clusterhead is a node responsible for forwarding packets to all of its neighbours. All 1-hop neighbours of a clusterhead can not be clusterheads themselves. Gateway nodes link multiple clusterheads together. If a node is neither a gateway nor a clusterhead, it does not rebroadcast and is classified as an ordinary node. Nodes revert to the initial state if no traffic has been seen for a defined period. More complete rules for the alteration of node states are given in [47].

With PC, clusterheads are selected using the first declaration wins principle, which dictates that the first node to broadcast itself as a clusterhead automatically becomes the clusterhead. All other nodes within the broadcasting radius of the first clusterhead declaration broadcast must eventually declare themselves as gateways or ordinary nodes, depending on subsequent node declarations. A node monitors the number of neighboring clusterheads (NC) and neighboring gateways (NG). It can declare itself a gateway when

$$\alpha \times \mathrm{NC} + \beta > \mathrm{NG} \tag{2.19}$$

$\alpha$ and $\beta$ can be local parameters ($\alpha \geq 0$, $\beta \geq 0$), unique to each node, that can be adjusted based on density, channel usage, etc. This formula is designed to limit the number of gateway nodes that link clusterheads together.

Deliberately delaying a retransmission for a length of time (much greater than the network propagation delay) can convey information without adding additional messages, which is profitable for energy saving. In the Delayed Intelligence (DI) strategy [31] a node delays its retransmission according to the received signal power and its local remaining energy.

To maximize efficiency of transmissions using DI, an additional transmission delay is proportional to the power of received broadcasted packets. In dense networks, the physically close neighbours receive and record high transmission power. These nodes would not cover as many new nodes as those nodes toward the edge of the broadcasting radius of the initiating node and are penalized most heavily before responding. To maximize longevity, each node is also penalized with a delay inversely proportional to its local remaining energy. Thus, the higher the local energy of a node, the higher is the probability that a node responds first and undertakes a broadcasting role. If the node is ordinary or is already in one of the aforementioned states, no delays are introduced. The wait time $W$ is calculated as

$$W = \delta \times \frac{receivedPower}{localEnergy}, \tag{2.20}$$

where $\delta$ is a constant. We use Passive Clustering with Delayed Intelligence (PCDI) in our approach. We modified PCDI in order to make it consider heterogeneity of the nodes' mobility - these modifications are described in Chapter 5.

Note that there is a body of other work concerning routing in MANETs relying on formation of virtual infrastructures such as clusters [63-72] or connected dominating sets [73-75]. But apart from Passive Clustering these forms of routing require proactive exchange of control packets to maintain the virtual infrastructure. This makes them less optimal for our approach than Passive Clustering.

## *2.4. Delay Tolerant Networks*

Delay Tolerant Network (DTN) approaches concern wireless networks with disconnections, in particular the path between the source and the destination may never exist so store and forward strategy is typically utilized. In the case of sending data from animal mounted nodes to sinks and handling in-situ queries we anticipate disconnections, especially when animals are located on pastures. Therefore it is relevant to summarise here the current work in the area of DTNs and discuss if it can be used to address disconnections in our scenario. We also envisage utilization of DTN techniques to connect sinks to farm servers when the constant wireless or wired connectivity is not available or too expensive.

### 2.4.1. Reactive Approaches

The reactive DTN approaches concentrate on utilising existing node's mobility for message forwarding. The initial work in this area, before DTNs were formally defined [76], addressed MANETs affected by disconnections. One of the first such papers [77], written by Xiangchuan Chen and Amy L. Murphy concerned exchanging data between nodes within disconnected MANETs. A source node forwards the message to a node within its partition (i.e. island of connectivity) that is most likely to deliver the message to the destination. The authors proposed a metric called *utility* that allows choosing the best candidate for forwarding the message. Utility consists of several components:

- Most recently noticed – when the potential forwarding node noticed the target for the last time
- Most frequently noticed – how frequently the potential forwarding node was noticing the destination
- Future plans – future plans of the potential forwarding node's carrier e.g. data from a calendar application
- Source defined utility – a function defined by the source node

The source decides about the weights given to the utility components. Sending data to the forwarding node has the following stages:

1) Utility probe – the source broadcasts the query concerning the utility of all nodes within the sources' island of connectivity
2) Utility collection – the source collects the replies to the query
3) Message redistribution – the source sends the message to the node with the highest utility

Authors envisage only one disconnected hop within the path. We do not utilize this approach for sending data from animal mounted nodes to sinks because we do not have sufficient data to assume that cattle exhibit as much regularity in their social behaviour as humans. We are also interested in detecting animal diseases that can influence mobility and social interactions of the animals. Collecting contact patterns would require considerable network overhead decreasing the battery life. This approach is not optimal for sending data from sinks to farm servers because here we have a more classical DTN case with infrequent node encounters rather than disconnected dense MANETs.

In Epidemic Routing [78] the path can comprise more disconnected hops. In particular nodes exchange the forwarded messages stored in their buffers whenever they meet. All the messages are exchanged subject to the storage space available to the nodes and their local preferences. Each message has a maximal hop count, which is the maximal number of hops it is allowed to traverse.

A very similar approaches were proposed in [79-81]. In particular these papers concentrate on utilization of existing mobility of people, vehicles or even animals to exchange data between stationary devices such as sensors, WAN connected access points, message boards etc. The Epidemic Routing approaches are very expensive in terms of network overhead because of high redundancy in forwarding data. This, due to energy constrains, makes it not optimal for forwarding data from animal mounted nodes to sinks. In the case of forwarding data from sinks to farm servers too high redundancy can result in long data exchange times.

Kumar Viswanath and Katia Obraczka proposed [82] a MANET broadcasting approach, Adaptive Flooding Protocol that addresses disconnections. Nodes exchange periodic ‘Hello’ packets which contain list of 1-hop neighbours, speed and movement direction. Basing on this velocity data and level of MAC collisions nodes choose locally one of the broadcasting strategies: scoped flooding, plain flooding (i.e. simple flooding) and hyper flooding. All the broadcasted packets contain list of neighbours of the broadcasting nodes. In scoped flooding a node does not rebroadcast if it shares at least 85% of the neighbours with the sender of the packet. In the hyper flooding a node repeats the broadcast whenever it receives a packet (broadcasted packet or a ‘Hello’ packet) directly from a node outside its neighbour list.

This approach was later extended by Khelil et al. [83] and called Hypergossiping. The authors modelled the topology as a constantly partitioned MANET with nodes moving from one partition to another. Nodes exchange periodic ‘Hello’ packets and rebroadcast with the probability *p* dependant on the number of neighbours. If a new neighbour is detected nodes append to the ‘Hello’ packets a list of most recently received or locally originated broadcasts – Last Broadcast Received (LBR). If a node receives a ‘Hello’ packet with LBR sufficiently different from its own, it starts rebroadcasting the non expired broadcasts it received or originated. In this way the broadcasts are exchanged between partitions in a store and forward manner.

Adaptive Flooding Protocol [82] and Hypergossiping [83] require periodic exchange of ‘Hello’ packets to maintain knowledge about the topology. The cattle can move at any time so for sending data from animal mounted nodes to sinks and handling in-situ queries exchange of ‘Hello’ packets would result in high network overhead not acceptable due to the energy constrains. Forwarding data from sinks to farm servers is a unicast or a very selective multicast rather than broadcast so there is no need to forward data to each node in the network.

Davis et al. developed [84] dropping strategies for Epidemic Routing [78] to address the limited storage space of the nodes. They consider following strategies: Drop-Random, Drop-Least-Recently-Received, Drop-Oldest and Drop-Least-Encountered. They find the Drop-Oldest and Drop-Least-Encountered to produce the best results. Drop-Least-Encountered utilizes the per-node metric for estimating likelihood of meeting each other node. The messages addressed to the nodes which are least likely to be met are dropped. In particular each node (e.g. node A) computes the probability of meeting each other node (e.g. node B) considering the collocated nodes (e.g. node C) according to the following formula:

$$M_{t+1}(A,C) = \begin{cases} \lambda M_t(A,C) & \textit{if none colocated} \\ \lambda M_t(A,C)+1 & \textit{if } C=B \\ \lambda M_t(A,C)+\alpha M_t(B,C) & \textit{for all } C \neq B \end{cases}, \quad (2.21)$$

where $M_t(A,C)$ is the value for meeting between A and C at time $t$ (initially 0), $\alpha$=0.1 and $\lambda$=0.95 are predefined constants. In practise this approach offers only marginal improvement over the basic Epidemic Routing [78] and for the same reasons as Epidemic Routing is not optimal for our scenario.

A similar metric was proposed by Lindgren et al. [85] in the Prophet algorithm. Contrary to Davis et al. [84] they use this metric not for the dropping strategy but for taking the decision about forwarding data to an encountered node. In particular a node forwards the data only if an encountered node has a higher probability of successful forwarding data to the destination node. We do not utilize this algorithm for sending data from animal mounted nodes to sinks because we do not have sufficient data to assume the regularity of cattle social patterns. We are also interested in detecting unusual conditions such as animal disease that can change typical animal behaviour. However this approach is useful for sending data from sinks to farm servers.

Jain et al. proposed [86] several routing algorithms for DTNs. They are divided into following classes according to knowledge available to the algorithms: zero knowledge, complete knowledge and partial knowledge. In the proposed zero knowledge algorithm, First Contact, subsequent hops are chosen randomly similarly to the Epidemic Routing [78]. Partial knowledge algorithms are modifications of the Dijkstra's Algorithm and differ in consideration of temporal dynamics of link costs and message queues. In the case of complete knowledge, routing is formulated as a Linear Programming problem. The zero knowledge algorithm is not optimal for our scenario for the same reasons as Epidemic Routing. The remaining algorithms require global knowledge which is not available both in the case of sending data from animal mounted nodes to sinks and from sinks to farm servers.

Pan Hui and Jon Crowcroft proposed [87] a DTN routing algorithms that are, according to the data from the field experiments, congruent with the human socializing patterns. In particular each node is assigned a set of labels describing the communities its carrier belongs to. Each node also measures its centrality – the number of distinct nodes met during a time unit (authors select 6 hours). Centrality is measured on the global and local levels. Local centrality refers only to encountered nodes belonging to a particular community the measuring node belongs. Global centrality refers to all encountered nodes. According to the first algorithm if a forwarding node does not belong to the same community as the destination, it forwards the data to nodes with the higher global centrality and to nodes from the same community as the destination. A node from the same community as the destination forwards the message to a destination or a node with higher local centrality. The second algorithm is similar to the first one but the forwarding node from outside the community of the destination deletes the message after forwarding it to a node from the destination node's community. These approaches are strongly human oriented so they are not directly applicable for animal mounted nodes. For the case of sending data from sinks to farm servers the algorithms are targeted at too large numbers of nodes from different communities. Not all the farmers in a village are going to participate in this form of data distribution. Some are going to use different means of connecting sinks to farm servers, potentially available to them, such as constant wired or wireless connectivity. DTN approach is a worst case scenario. Even if all the farmers participated in this form of data exchange the number of nodes and variety of communities is going to be too small to justify this approach.

### 2.4.2. Proactive Approaches

In proactive DTN approaches nodes adjust their mobility in order to support the message forwarding [88, 89]. Certain approaches utilize the dedicated message couriers to forward the data [89].

First of the proactive approaches were proposed by Qun Li and Daniela Rus [88]. In the first approach each node has full knowledge about trajectories of all other nodes and in order to send a message selects the optimal path. The forwarding nodes may have to adjust their trajectories as well. The second approach assumes that this knowledge is not available and nodes only have the periodic location data about their spatial neighbours. The first approach is not applicable to our scenario because both in the case of animal mounted nodes and couriers transporting data from sinks to farm servers the exact trajectories are not globally known. The second approach requires additional long range communication channel to exchange node positions, which can be easily avoided in our scenario by utilization of other approaches – both in the case of sending data from animal mounted nodes to sinks and from sinks to farms servers.

Wenrui Zhao et al. [89] proposed utilisation of specialised ferries for transporting messages. They are distinguished from nodes, which are sources or destinations of the messages. The authors propose two schemes. In the first one, Node-Initiated Message Ferrying (NIMF), ferries move according to designated permanent routes and are approached by nodes which want to forward a message. In Ferry-Initiated Message Ferrying (FIMF) ferries can be summoned by nodes and adjust their trajectories to deliver messages from a source to a destination. In the case of sending data from animal mounted nodes to sinks and handling in-situ queries both approaches are not applicable. In our scenario animals will not purposefully approach any ferries for the purpose of data transfer (NIMF). There are also no potential ferries that can be summoned by animal mounted nodes (FIMF). Let us now consider the case of sending data from sinks to farm servers. Neither stationary nor animal mounted sinks can approach a ferry (NIMF). A ferry (pedestrian or vehicle) could be summoned to approach a sink but this would required a secondary communication channel (e.g. SMS), which could be used to send the actual data. Note that we do not expect a high intensity of data traffic.

## *2.5. Design of Wireless Monitoring Systems*

Most of the existing work concerning design of wireless monitoring systems (e.g. [36, 90]) presents experiences from the performed research rather than proposing any formalized methodologies. The reported experiences are typically very application specific and thus not directly applicable to our scenario.

The notable example here is [91] which discusses methodologies for the design of embedded systems. The referenced work does not consider wireless networking issues but is applicable to the design of wireless monitoring systems which typically comprise embedded devices. Typically in computer systems hardware and software are separated and very different methodologies are applied to designing them. Hardware designers compose the system from the interconnected, parallel building blocks which form a static structure. Software designers use sequential building blocks composed in the dynamically changing structure. The authors claim that the design of embedded systems requires a holistic approach, which integrates paradigms from hardware and software design and control theory.

The authors divide system design methodologies into critical and best effort. Critical methods try to guarantee system safety at all costs even in the extreme conditions. Best effort methods try to optimize system's performance when the system operates in the expected conditions. In our case utilization of ad hoc wireless networks enforces best effort approach which we utilize.

## *2.6. System and Network Management*

System and network management are potentially an important issue in the cattle monitoring system. Therefore it is relevant to review the existing work from this area. Traditionally network management is performed through the manager-agent model [92]. Agent is a software component running on the managed device, which monitors and controls this device. Agent can be contacted by the managers to retrieve monitored parameters or control managed devices. Agents can also contact managers asynchronously in the case of occurrence of predefined events. This model was utilizes in the Simple Network Management Protocol (SNMP) [93].

The manager-agent model can be extended by introducing the third layer – mid-level managers (MLM) [94], which can execute tasks closer to agents and thus lower bandwidth consumption. The more recent paper proposes a peer-to-peer manager-agent architecture [92]. Another research direction in network and system management is policy-based management [95]. Policies represent the externalized logic that determines the behaviour of the managed systems. In this way the operation of network and computing resources can be guided to follow certain rules and dynamically configured so that they can achieve certain goals and react to their environment. This approach has been utilised to manage large scale sensor networks [90].

As shown later in Section 4.4 the server part of the proposed cattle monitoring system has limited scale. The utilised networking technologies are mostly of the ad-hoc nature. In this way the management of the proposed system is minimized and thus we do not utilise the approaches discussed above.

## *2.7. Discussion*

This chapter has reviewed some of the most relevant techniques and approaches according to the criteria given in Section 2.1. This section summarizes all the approaches reviewed in this chapter according to the list of selected criteria. In the case of connecting animal mounted nodes with sinks and mobile users we argue that there is no single approach that satisfies all the criteria and we need some kind of combined and extended approach that would provide the best support for all the criteria. In the case of connecting sinks to farm server we show which current approach is the most optimal for this task.

### 2.7.1. Connecting Animal Mounted Nodes, Sinks and Users

The existing Wireless Sensor Networks (WSN) for animal monitoring are either stationary (e.g. [34]), too expensive [1, 35], target small scale deployments (e.g. [36]), do not consider energy efficiency [1, 2, 12, 37, 38] or consider very different types of animals and environments [39-41]. The classical MANET routing algorithms [17-20] do not target energy efficiency and do not address disconnections. Georouting approaches [52-54] require spatial positioning not available in our scenario. The energy efficient approaches from WSN area consider static scenarios [13, 45, 46, 49-51], whereas energy efficient approaches from MANETs assume high data traffic and thus optimize energy efficiency of the data traffic not route discovery or broadcasting [21-28]. Some other approaches propose switching off transmitters to save energy [21, 29] – we do not use them because their gain strongly depends on the characteristics of a given hardware. We utilise however in our approach energy control of transmitters [27, 28].

In our scenario we have limited amounts of data traffic so most of the energy is potentially spent on the route discovery and broadcasting of queries. Therefore we need to make broadcasts energy efficient. The probability based methods of broadcast optimization limit reachability [44, 57], which would affect negatively reliability in our scenario. The proactive neighbourhood knowledge methods rely on the topology knowledge obtained from the periodic 'Hello' packets [58-62]. Such messages due to high mobility of our scenario would have to be frequently sent which would limit energy efficiency. Area based methods often require means of geolocating not available in our scenario (e.g. [57]). The passive neighbourhood knowledge methods are best suited to our scenario, in particular offering best performance Passive Clustering with Delay Intelligence [31], which we adapt to our system. It is superior to the georuting methods, which can rely on signal attenuation instead of geolocating [57] because they use delays of random length, not conveying any information. There is a body of work other than Passive Clustering concerning routing in MANETs relying on formation of virtual infrastructures such as clusters [63-72] or connected dominating sets [73-75]. However they require proactive exchange of control messages to maintain the virtual infrastructure, which makes them less optimal for our scenario than Passive Clustering.

There is also no available single approach to address disconnections in our scenario. The current approaches addressing disconnections in MANETs [82, 83] typically require exchange of 'Hello' packets which, as discussed earlier, is not advisable in our scenario. From the DTN area Epidemic Routing [78-81, 84, 86] would be very expensive in terms of energy efficiency due to high redundancy. The approaches utilising nodes movement pattern either assume more knowledge than is available in our scenario [86] or rely on the regularity of movement of nodes [77, 85, 87]. We do not have enough data to assume that contact patterns of the animals are as regular as in the case of people. Moreover, we want to detect unusual conditions such as animal diseases that can have impact on the behaviour of the animals. We cannot use any proactive DTN approaches [88, 89] because the animal mounted nodes do not control mobility of the animals and there are no candidates for dedicated message ferries.

### 2.7.2. Connecting Sinks with Farm Servers

Using pedestrians or vehicles to connect sinks to farm server is a typical DTN problem. Therefore the approaches addressing disconnected MANETs [77, 82, 83] are not directly applicable. The Epidemic Routing approaches [78-81, 84] can pose too much overhead due to high redundancy. The approaches proposed by Jain et al. [86] are either similar to Epidemic Routing or require too much knowledge about mobility of nodes. Pocket switched network routing algorithms proposed by Pan Hui and Jon Crowcroft [87] are targeted at too large numbers of nodes from different communities – such scale is not realistic in our scenario. The best suited for this scenario is then probabilistic routing [85].

Proactive routing algorithms proposed by Qun Li and Daniela Rus [88] require more knowledge about the mobility of the nodes than is available in our scenario. Considering message ferrying approach [89] we cannot use in our scenario Node-Initiated Message Ferrying because sinks cannot approach a ferry (they are stationary or animal mounted). Ferry-Initiated Message Ferrying requires a secondary communication channel to summon a ferry, which in our scenario could be used to send the actual data as we do not expect high intensity of data traffic.

# 3. Field Experiments

In this chapter we describe field experiments we performed at the University of Nottingham's Dairy Centre in collaboration with School of Biosciences. The purpose of these field experiments was collection of realistic data sets and requirements necessary to design, develop and evaluate the delay tolerant architecture and the energy efficient MANET routing algorithm for the cattle monitoring system. These experiments included the quantitative and qualitative parts described further in this chapter.

### 3.1.1. Quantitative Experiments

#### 3.1.1.1. Experiment setup

Quantitative experiments comprised cattle movement and behaviour monitoring in order to gather the realistic environmental constrains. In the first field experiment we monitored two of the cows located in one of the divisions of a modern dairy intended for about 100 animals, shown in Figure 3.1. Cows could move freely in the area with feeder, water tank, resting bays and milking robots available 24 hours a day. We mounted on the monitored cows two collars, each comprising a neck strap and an aluminium instrument enclosure containing a Bluetooth GPS and a Bluetooth enabled mobile phone. Mobile phones were logging data from the GPS receivers including positions and timestamps. All the cows in the dairy were wearing pedometers. Their measurements were automatically collected by milking robots whenever a cow was milked (see Figure 3.2). One of the cows became spooked after installing the collar and the collar had to be placed on a different cow. Cows that were finally wearing the collars did not pay any attention to them and did not try to remove them. Some of the cows were licking the aluminium enclosures attached to the collars worn by other cows but without any attempt to destroy them or force them open. The data collection started at 11:10. Both GPS receivers worked until around 14:05, when we switched them off. Some of the collected measurements suggested that cows moved with speeds impossible for them, which suggested GPS errors. Concurrently we were filming the part of the dairy where the monitored cows were kept. We placed the camera on the ramp above this area. This location offered the most complete view but some parts of the area were obscured. GPS receivers and filming were utilized only for the purpose of our field experiments. Their utilization is not intended for the target monitoring system.

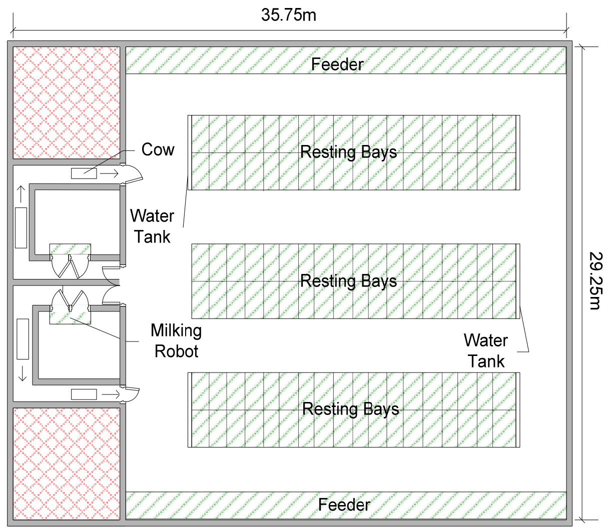


**Figure 3.1. Layout of the dairy division**

We repeated the previous experiment with five collars mounted on animals (see Figure 3.3 and Figure 3.4) and two cameras located at two different ramps to get a more complete view of the area where the monitored cows were kept. We had GPS receivers with better batteries than before and we were logging data about the precision of logged locations. Monitoring started at 11:10. The collars were removed at 18:10. GPS receivers worked until 18:24 (manually turned off), 12:23 (probably jammed), 18:51 (manually turned off), 15:09 (exhausted battery), 15:33 (exhausted battery). We received the plan of the dairy and then captured the coordinates of the characteristic locations on the plan using a handheld GPS receiver.

**Figure 3.2. Leg mounted pedometer**


**Figure 3.3. Collar with a GPS receiver and a mobile phone**

**Figure 3.4. Cow wearing the collar**

### 3.1.1.2. Results

Our field experiments show that cows typically react well to the animal mounted collars weighting 1075g. This is very promising for the practical feasibility of the target cattle monitoring system. In the Figure 3.5 based on the pedometer data we can see that walking intensity of the monitored cows was sufficiently different to detect oestrus and animal diseases but too similar to influence routing. In the Figure 3.6 based on the GPS data we can see that at the same time the preferred momentary walking speed significantly differs between cows, which can be utilized in the wireless routing algorithm. The same figure shows that they rarely move faster than 0.8 m/s. We can see that the preferred walking speeds have gamma distribution [96] with $\Theta = 2.0$ and $k = 2$ (Cow 1) or $k = 1$ (Cow 3).

The pedometer data presented in Figure 3.7 demonstrates that cows are active all the day and night including walking and milking. Their walking activity tends to be less intensive in two periods: 0-6 and 15-18. These periods can be utilized for scheduled data exchanges. Video footage and analysis of the GPS data (see Figure 3.8) show no other predictable behaviour patterns useful for design of the wireless routing algorithm.

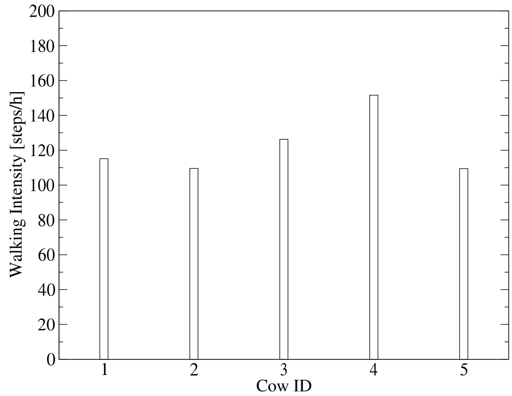


**Figure 3.5. Walking intensity (pedometers, 2nd experiment, 4:00-22:19, 13th August)**

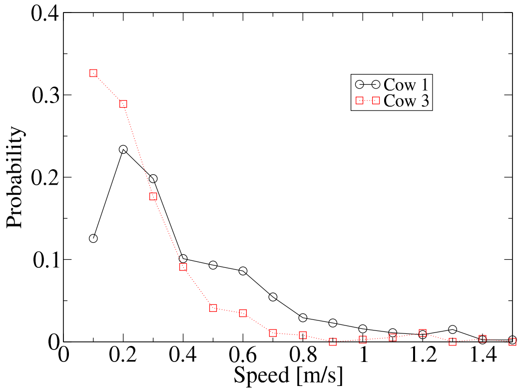


**Figure 3.6. Probability distribution of animal speed (GPS, 2nd experiment)**

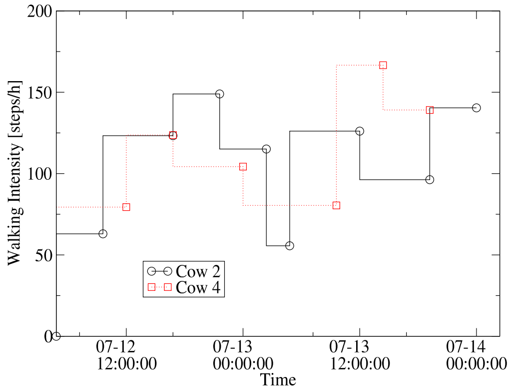


**Figure 3.7. Activity over the day (pedometers, 2nd experiment on 13th August)**

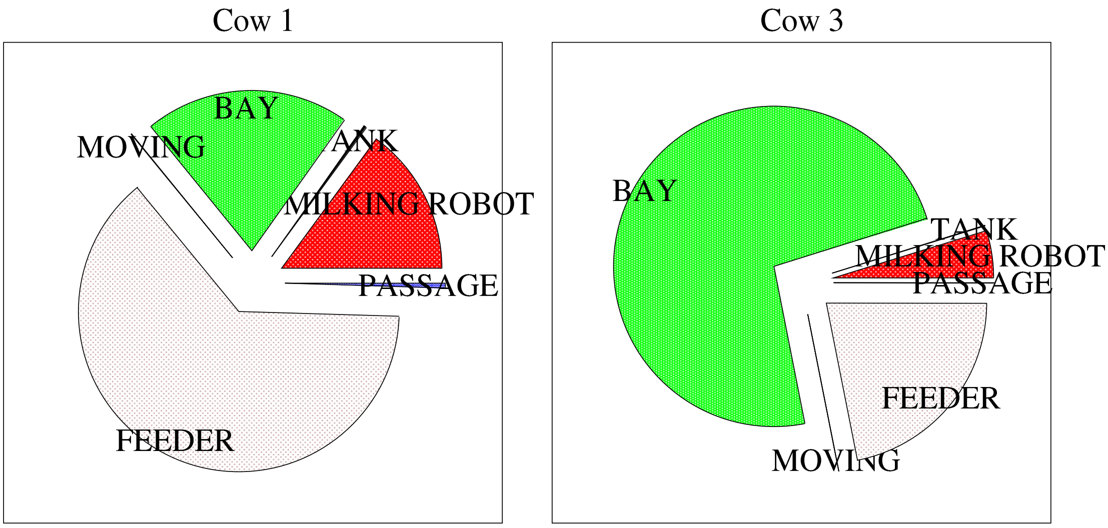


**Figure 3.8. Activity diagrams (GPS, 2nd experiment)**

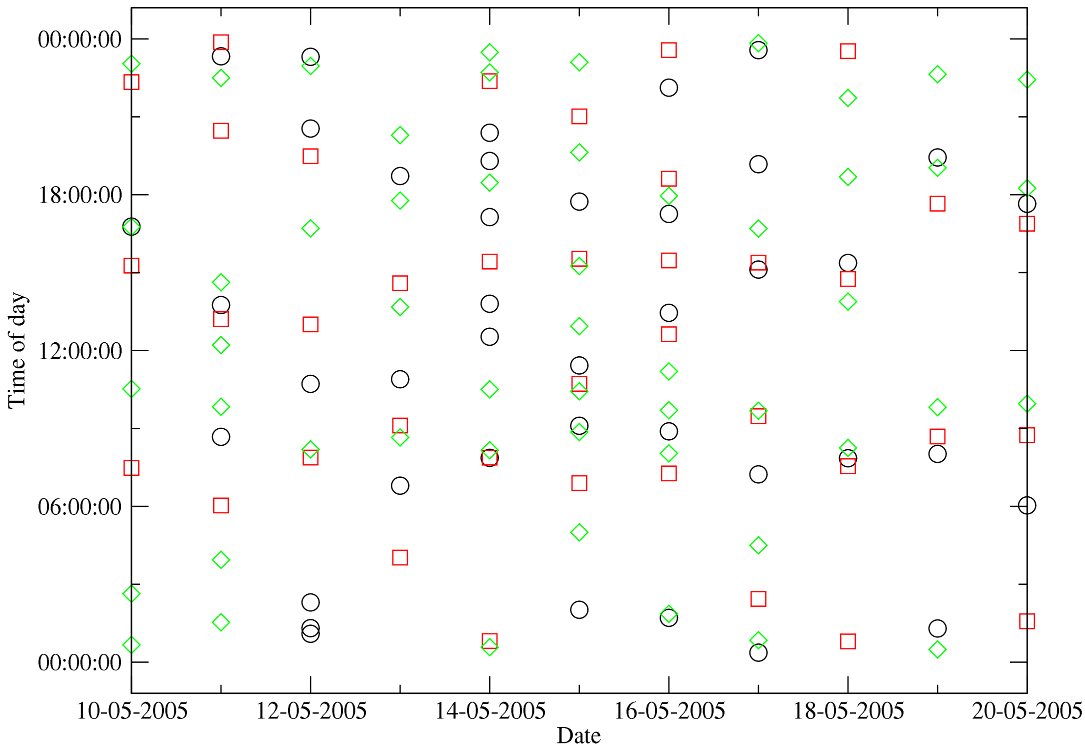


**Figure 3.9. Milking times**

Figure 3.9 presents historical pedometer data received from the farm personnel. Each symbols presents milking and different symbol shapes refer to different cows. We can see that milking can happen at any time of the day and night and particular animals do not show any constant preference or frequency.

The quantitative experiments were performed in the dairy but this is only an example deployment scenario of the target monitoring system. The target monitoring system is also intended to monitor beef cattle animals kept continuously on the pastures even all the year. Such cattle may never be taken to the farm buildings.

### 3.1.2. Qualitative Experiments

#### 3.1.2.1. Experiment Setup

The objective of the qualitative experiments was gathering of the realistic user requirements. They comprised distributing an anonymous questionnaire to the farm personnel and researchers working on the farm. The content of the questionnaire can be found in Appendix 1. We received four filled questionnaires. One of them was filled by a regular herdsman, one by the head herdsman (farm manager) and two by researchers working on the farm.

#### 3.1.2.2. Results

From the performed questionnaire we learnt that the most required functionality of the system is detection of oestrus, pregnancy and animal diseases. Users have to be informed about oestrus and a newly detected disease as quickly as possible. The pregnancy should be reported within 48 hours from detection. Detection of reduced efficiency of pastures is less essential and less urgent – it should be reported to users within 24 hours from detection.

In order to inform users about the detected oestrus and animal diseases as quickly as possible, animal mounted nodes should be able to detect oestrus and animal diseases on their own and send this information over the sink as soon as it is detected. When no particular event is detected, data from collars should be transmitted via sink at least every 24 hours to allow server its aggregation and detection of reduced performance of pastures.

The users recognize sending notifications to their mobile phones as very useful and have to receive them any time, not only when they are collocated with the animals. This requires sending the notifications using the GSM network as e.g. SMS messages. The users need to perform in-situ queries up to several times a day. This means that energy saving is relevant not only for sending data to sinks but also in-situ queries.

The head herdsman recognized also as useful measuring body temperature of the animals. This however requires using sensors mounted inside animal body because externally mounted sensors do not provide reliable measurements [1]. The regular herdsman recognized as useful detection of calving but feasibility of this requires further research in animal physiology.

# 4. Architecture of the Cattle Monitoring System

## *4.1. Functional Overview*

This chapter proposes architecture for the scalable cattle monitoring system fulfilling realistic requirements identified in Chapters 2 and 3. The work presented in this chapter has been published in [97, 98]. As shown in Figure 4.1, in order to clarify the scope of the target application the work presented within this thesis considers monitoring walking, feed intake intensity and location of cattle in order to detect oestrus, pregnancy, animal diseases such as mastitis or lameness and reduced efficiency of pastures and paddocks (i.e. fenced parts of pastures). Location of animals refers to the association between a paddock, pasture or a barn and the animals located there. The animals can be moved between barns, pastures and paddocks, which makes their location not always obvious for the farm personnel.

The envisaged system is targeted at a farming enterprise, which can possess several pastures and barns where animals are kept. The users of this system are stockmen, farm managers and veterinaries. The typical number of users can be estimated from a study about the labour use on UK farms [99]. In 1994 the average numbers of workers of a farming enterprise (including part-time, casual and contract) was 8.54. This average for very small (from 0 to 8 ESU) was 3.14. For very large (200 ESU and over) this was 19.99.

The stockmen can use this system in order to detect disorders of the animals and to take better care of them. In particular detecting oestrus can inform the stockmen when the animals can be successfully impregnated. Monitoring efficiency of pastures will let the stockmen know when to move the animals to a different pasture, which increases the efficiency of rearing and improves health and welfare of the animals. Finally this system can inform them where an animal of interest is currently located. Farm managers can use the target system to monitor work of stockmen and asses efficiency of the farm. They can use the data about efficiency of pastures to develop their policy concerning utilization of land, buying or selling properties and changing the number of livestock. Veterinaries can use the system to better understand the conditions of the ill animals. The selected data collected by the system can be provided to the business counterparts to give them the opportunity to evaluate the animals before the purchase.

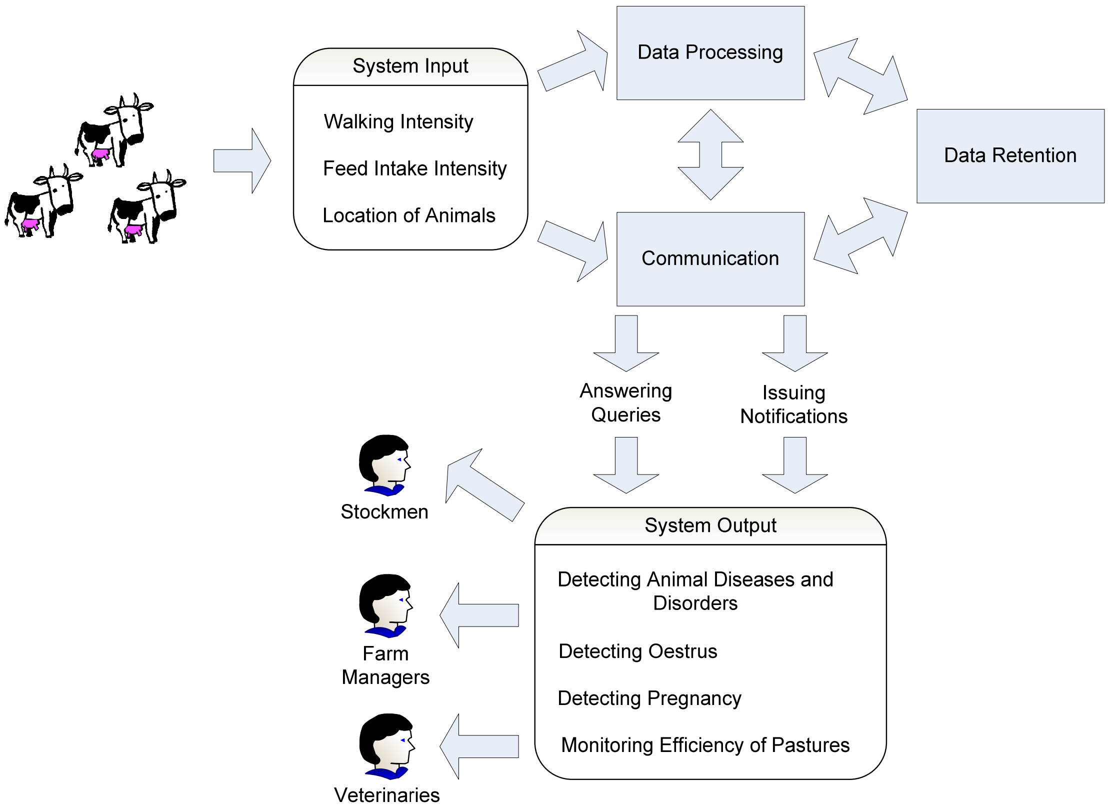


**Figure 4.1. Functional overview of the cattle monitoring system**

The average number of cattle held by a British enterprise in 1998 was 72 [100], which gives an idea about the required scale of the monitoring system. In extreme cases an enterprise can have many more animals than this. The cattle is kept either all the year continuously on the pastures or all the year in the barn but the most common practice is to keep them on the pastures in the warmer half of the year and indoors in the other [100]. The proposed system can be used to monitor animals regardless if they are kept continuously on the pastures or continuously in a barn and regardless if they currently yield milk or not. Depending on the conditions stocking density of the pastures can range from 2 to 7 animals per ha [100], which directly projects onto density of a wireless topology. The target system should then monitor animals kept both in the barns and on the pastures, within this range of the stocking density.

Walking intensity is typically measured using a pedometer mounted on the animal’s leg [7, 9] and feed intake by an accelerometer mounted on the animal’s neck [1]. Detecting oestrus and animal diseases requires historical data, which is also useful for the veterinary examination. Therefore, both the raw and processed data should be safely retained as long as the animal stays on the farmer’s hold.

In order to satisfy requirements identified in Chapter 3, the access to the collected data should be provided by queries and notifications (see Figure 4.1). Users should be able to query the real-time and historical data, raw and processed. To make the data collected by the system more accessible for all types of users, querying should be possible from the farm terminals, remotely over Internet, from mobile phones, home and office PCs. According to the identified requirements users, even these working currently on the field, should be informed about detection of diseases and oestrus as soon as possible. Therefore depending on the enterprise’s financial constrains and the preference of the users notifications should be issued to the users' mobile phones and/or displayed on the message boards mounted in the farm buildings. Querying should be also possible at the place where animals are kept using a handheld device. A user (i.e. a farmer, a stockman or a veterinary) should be able to query all the currently ill animals or animals which are or soon going to be ready for insemination. It should be possible for a user to retrieve data about a certain animal such as current and past diseases, oestrus, pregnancy, average frequency of oestrus and predicted next oestrus. A more advanced user such as a veterinary or a skilled stockman should be able to see graph of a particular factor of a particular animal together with its deviation from the average.

## *4.2. Usage, Installation and Maintenance Scenarios*

This section presents usage, installation and maintenance scenarios for the envisaged system. The system is targeted at various groups of users each of them interacting with it in a different way. Stockmen can define custom events and are notified about their occurrence. These events concern oestrus, pregnancy, illness of an animal or shortage of feed on the pasture. They are delivered to smart phones carried by stockmen or displayed on the message boards mounted in the places visible to stockmen such as living areas or offices. These notifications can trigger certain actions performed by stockmen such as artificial insemination, isolating an animal, calling a veterinary or moving animals to different pastures. Stockmen can also query data concerning animals in order to monitor the results of their activities such as hormonal synchronization or artificial insemination. They can also identify a pasture where an animal of interest is kept or query the historic processed and unprocessed data about an animal to better understand its current conditions. The general usage scenario of the system is presented in Figure 4.2. The stockmen and/or the envisaged system refer to the particular animals using their ear tag ids, custom nicknames or even pre-loaded photos of the animals. Querying data about animals is possible from a terminal in a farm building to check if a farming activity is required and to plan such activity. Querying is also possible on the pasture using a smart phone or PDA to get an instant information supporting performed farming activities.

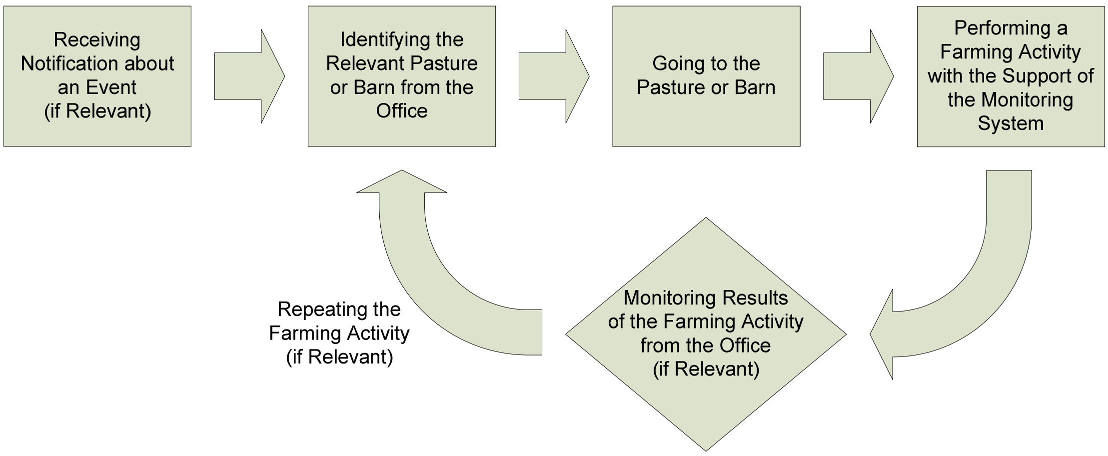


**Figure 4.2. Stockmen's interaction with the system**

We can envision following example usage scenarios of the proposed system. Let us assume that the stockmen want to perform the hormonal synchronization in order to artificially inseminate cows in gross. They begin by identifying non pregnant and mature enough cows using terminals in the farm office. This stage is very important because in the case of using prostaglandins synchronization of pregnant cows can cause abortion [100]. Synchronizing too young or pregnant cows can also cause wastage of the medicines used for synchronization. Stockmen identify barns and pastures where the cows to be synchronized are located. Then they visit these places and make the necessary injections or insert implants. Two days later they check which cows successfully entered oestrus and similarly as in the case of synchronization they use the system to locate and identify the appropriate animals to isolate them as a preparation for the visit of the inseminator. After 21 days stockmen use the system to monitor the pregnancy of cows. The cows entering oestrus again (i.e. not pregnant) can be inseminated again. If the farm does not practice hormonal synchronization, stockmen can still utilize the system to detect the 'natural' occurrence of the oestrus (e.g. by receiving an appropriate notification). In particular they act similarly as in the scenario described above to prepare cows having oestrus for the visit of the inseminator and to monitor the results of insemination.

Another anticipated scenario of utilization of the system is monitoring grazing process of the animals. If the stockmen receive the notification about reduced efficiency of a pasture or a paddock, they can check the availability of water there, provide animals with pasture supplement or move them to another pasture, paddock or even a barn. Monitoring efficiency of pastures is important to provide animals with sufficient amount of fodder. Concentrate supplements are preferred by cattle over herbage. In contrast, silage supplements are only eaten when the availability of herbage is limited. If the silage supplement is left to long in the pasture it can spoil [100]. Therefore, monitoring efficiency of pastures can improve the optimality of utilization of herbage and pasture supplements, which has always been challenging for farmers.

When stockmen receive a notification about an animal disease such as mastitis or lameness they can see the animal using the system to locate and identify it and then perform necessary actions such as calling a veterinary or isolating the affected animal.

Veterinaries, when they are called by stockmen, use the system differently. In particular they can examine the current and historic walking and feed intake intensity of the animals to better understand the animal's conditions. They can also find out in which paddock or barn the animal of interest is located. They can see a bigger picture of a potential disease by issuing a query from an office or query data about the currently examined animal using a smart phone or a PDA.

Let us envisage a scenario when the veterinary is called. She starts with identifying animals requiring her attention using the office terminal. Then with the stockmen she identifies the pastures or the barns where these animals are kept. They go there and examine the affected animals using the system to obtain additional relevant information.

Farm managers use the envisaged system for different purposes. In particular they monitor efficiency of stockmen in moving the animals between pastures and barns. This can be calculated as the length of time when animals were staying in pastures having efficiency below a certain threshold (e.g. 10%). Using the historic data about efficiency of paddocks they make decisions about buying or selling pastureland or changing the utilization of the land e.g. building new farm buildings on the low quality pastures. They can also use the historic data to examine if the low conception rate is caused by infertility of cows or poor skills of stockmen. If necessary they can decide to cull infertile cows. Typically farm managers query data using terminals in their offices and are not subscribed to any notifications.

The installation of the system should be as easy as possible to limit its cost. Ideally most if not all of the work should be manageable for stockmen with minimal technical skills. The farm servers can be available with all the necessary software preinstalled. Therefore the deployment of the farm infrastructure will involve only placing the devices and connecting them to the existing infrastructure such as LAN or Internet with minimal configuration. The pasture infrastructure can be deployed in a similar plug and play fashion. Installation of the animal mounted devices most probably involves most of the work. Each of the devices has to be individually mounted on an animal and the mapping between the hardware id of the device and the ear tag id of the animal must be provided. A basic data about each animal such as sex also has to be entered to the system. Optionally, possibly over the course of using the system, farm staff can enter more data about animals such as custom nick names or photos of the animals.

Similarly to installation, maintenance of the system should be as minimal as possible. Certain maintenance activities such as backing up data from farm servers, mounting devices on new born or newly acquired animals are unavoidable but should be maximally simplified so they do not require involvement of skilled technicians. In the case of replacing damaged devices, a new device should automatically regain any data and functionality of the old one. Maintenance of the animal mounted devices such as replacing batteries generally should be minimized. Note that any such maintenance is much easier when the animals are kept indoors (i.e. during the colder part of the year) than outdoors. Ideally the batteries should at least be able to suffice for the worm part of the year. On the other hand devices should be sufficiently small and light otherwise the animals may be provided with an incentive to remove or destroy them [35].

## *4.3. Input and Output of the Cattle Monitoring System*

This section defines input and output of a cattle monitoring system. The system's input data can be divided into time dependant and time independent data about the animals. The time dependant data about an animal can be presented as a following vector:

$$I_{ID,t} = (w, f), \tag{4.1}$$

where *ID* is an ear tag id of an animal to which the measurements refer, *t* is a timestamp of the measurements, *w* is walking intensity (i.e. number of steps made by an animal per time unit), *f* is feed intake intensity (i.e. the number of times an animal rose its head per time unit). The time independent data about the animal can be presented as a following vector:

$$I_{ID} = (s, N), \tag{4.2}$$

where *ID* is an ear tag id of an animal to which the measurements refer, *s* is the sex of an animal (as it is not relevant to detect oestrus for bulls or steers) and *N* is a set of custom names given to an animal.

The output data of the system comprises the time dependant data about an animal and a pasture/paddock. The output data about an animal can be presented as a following vector:

$$A_{ID,t} = (d_1, d_2, f_1, \ldots, f_n, l), \tag{4.3}$$

where *ID* is an ear tag id of an animal to which the measurements refer, *t* is a timestamp, $d_1$, $d_2$ are dates of previous and next oestrus $f_1,\ldots,f_n$ are flags indicating detection of pregnancy, mastitis, lameness etc. and *l* is the location of the animal i.e. the id of the pasture, paddock or barn where the animal is located. The output data about a pasture/paddock can be presented as a following vector:

$$P_{ID,t} = (e, A), \tag{4.4}$$

where *ID* is an id of a pasture/paddock, *t* is a timestamp, *e* is an integer percentage of the pasture/paddock efficiency (0% meaning no feed or water – animals do not graze), *A* is a set of animal ids placed on the pasture/paddock.

A user should be able to query data about a particular animal or paddock. The most important and potentially the most often queried are the most recent results of processing ($A_{ID,t}$ and $P_{ID,t}$). A more advanced user (veterinary or a skilled stockman) can query historical input data concerning a particular animal (intensity of walking and feed intake). In certain circumstances, historical results of processing concerning a particular pasture can be utilized. They may help in assessing value of a farmland to plan purchases and sale of the properties. A user should be able to refer to an animal using its ear tag or a custom name.

The users should be able also to make 'regular queries'. The input of such query would be a logical condition referring to the most recent output data concerning animals or paddocks. The output would be a set of ids or custom nicknames referring to paddocks or animals. The example queries would be: 'Which animals are ill?', 'Which cows will have oestrus tomorrow?' or 'Which paddocks occupied currently by animals have efficiency below 10%?'.

The system should be able to detect user-definable events and notify users about these events. Example definitions of the events are: 'If a new animal becomes ill send a relevant text message to every stockman and manager and display a notice on the message board in the farm', 'If a cow will have oestrus tomorrow send a text message to every stockman', 'If the efficiency of a paddock currently occupied by animals falls below 10% send a message to every stockman'. A definition of an event *i* can be formalized as:

$$E_i = (s, e, n), \tag{4.5}$$

where *s* is a scope of the notification: either all animals or all pastures. *e* is a logical function describing the circumstances, where the event happens. The input parameters of this function are variables from the most recent $A_{ID,t}$ or $P_{ID,t}$ . *n* defines a method of notification delivery. It is a set comprising *(u, c)* pairs where *u* is an id of a user or predefined group of users and *c* is a communication channel such as GSM text message, e-mail, notice board etc. An event detected according to definition $E_i$ can be formalized as:

$$V_i = (t, ID), \tag{4.6}$$

where *t* is a timestamp of the event and *ID* is an id of an animal or paddock that generated the event.

## *4.4. System Architecture*

The architecture for the proposed system is shown in Figure 4.3. An animal mounted device has the form of a collar with a built-in accelerometer measuring the intensity of feed intake. Walking intensity is measured by a pedometer mounted on the animal's leg. Measurements from the pedometer are acquired by the collar over wireless communication. As shown in Figure 4.4 measurements from the pedometer and accelerometer are stored and processed by the collar in order to answer in-situ queries (i.e. queries issued by users collocated with the animals) and to trigger sending data to the farm servers.

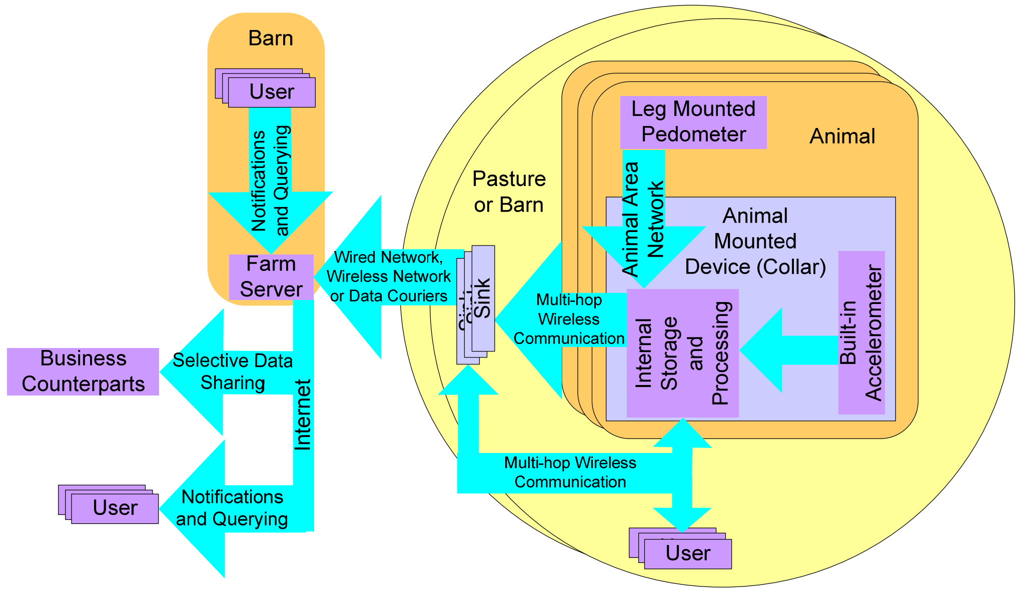


**Figure 4.3. Architecture of the cattle monitoring system**

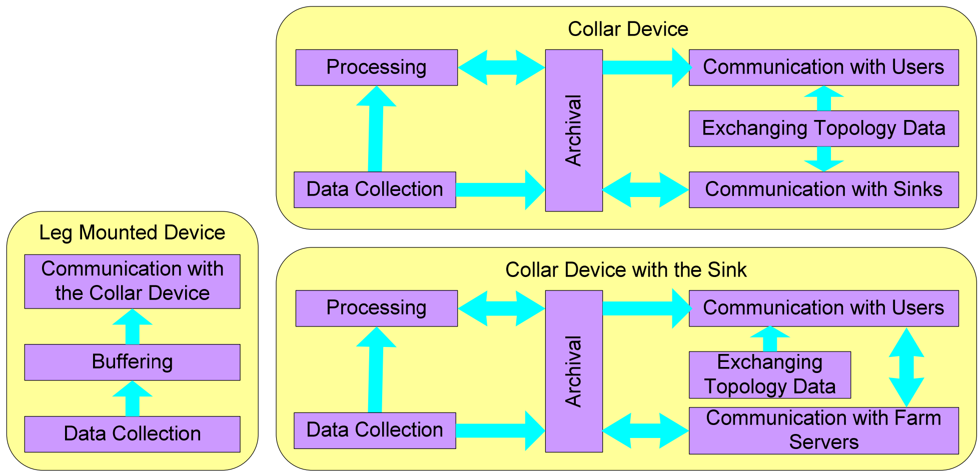


**Figure 4.4. Animal mounted sensor node – functional diagram**

Both the collar and the leg mounted pedometer are battery powered. Using renewable energy sources is difficult in this scenario. Solar batteries are not efficient when animals are kept indoors, may require manual cleaning and require rechargeable batteries with limited lifetime [101]. Swinging weights can compromise animals' welfare [36] and piezoelectric power sources are difficult to mount on animals [36].

Data processing performed by animal mounted devices aims to detect oestrus, pregnancy, animal diseases etc. They have wireless network interfaces and transmit raw and processed data to the farm servers over the sinks. The sinks are members of the MANET which forward the data collected and processed by animal mounted devices to farm servers. Sinks are stationary or animal mounted devices. If they are stationary and mounted outdoors they can be solar or wind powered. It is possible to deploy more sinks per a single barn or pasture in order to increase reliability of the system by decreasing probability of the animal mounted devices being outside of the multi-hop range of the sinks.

Sinks are associated with the pasture or barn where they are located and attach their identifiers to the data forwarded from animal mounted devices. This allows associating animals with the pasture or paddock where they are currently located, which is necessary for monitoring the efficiency of pastures and supports locating of a particular animal in the case of enterprises having numerous barns and/or pastures.

## *4.5. Communication Architecture*

In order to satisfy user requirements animal mounted collars inform the servers about detection of the oestrus or an animal disease. Every 24 hours they send the raw and processed data to the farm servers even when no such event was detected in order to allow server to evaluate efficiency of pastures and to inform the farm servers about detected pregnancies, which according to the users is less urgent (see Subsection 3.1.2.2.). This regular update works also as a heartbeat message allowing servers to detect broken or lost animal mounted devices. The typical amount of data sent from animal mounted devices to sinks is 32B.

Sinks can be connected to farm servers over a wired or wireless network connection or data couriers. As shown in Figure 4.5 the farm servers store the real-time and historic data, detect the user defined events and issue notifications about these events. The detection of reduced efficiency of pastures is performed by farm servers by aggregation of data from multiple animal mounted devices. This can be done only by the farm servers because only the farm servers have the data concerning all animals on the pasture. The sinks may not have this complete data when more than one is deployed at a single barn or pasture. The selective access to the data stored on the farm servers can be provided to business counterparts to give them the opportunity to evaluate the animals before the purchase.

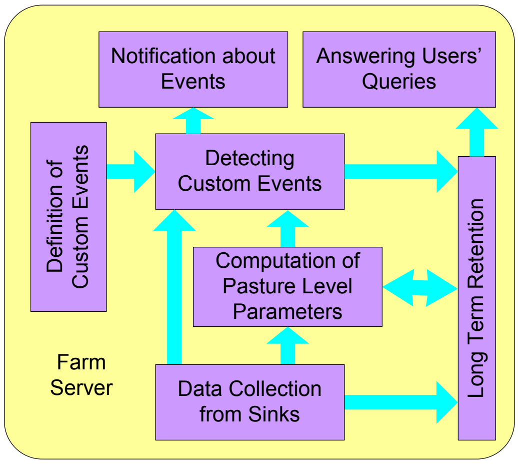


**Figure 4.5. Farm server – functional diagram**

In order to make delivery of data from animal mounted devices to sinks more reliable, reception of each portion of data is acknowledged by a sink. Animal mounted nodes keep track of these acknowledgements to resent the data if no acknowledgement has been received within a timeout. A loss of an acknowledgement can result in duplicate data being delivered to a sink. Because we allow deployment of multiple sinks per single barn or pasture, such duplicates can only be reliably detected by farm servers.

Considering the capabilities of modern database management systems this should not impose any excessive overhead.

The users can query the data stored on the servers, including raw and processed data, either locally at the farm or remotely over the Internet. Users located in a pasture, barn or in their close proximity may need to query data about the animals located there. This can be achieved by querying the data from a PDA or a smart phone connecting in the multi-hop manner to the animal mounted devices or to farm servers via sinks. The multi-hop route between a user and a sink may be temporarily unavailable so the sensor nodes should be able to answer a query using data available within their island of connectivity.

## *4.6. Deployment*

The example deployment of the system is shown in Figure 4.6. The data is regularly transferred from the animal mounted nodes to the sinks over the wireless communication. Where possible the sinks can be connected to farm servers over a wired network connection as in Pasture 1 and Barn 2. If a pasture is located far from the farm server it is possible to use GSM telephony to connect pastures to farm servers as in Pasture 2. In that case, a sink can be stationary or animal mounted. When no wired or wireless connection is available the sink can be connected to the farm servers using data couriers.

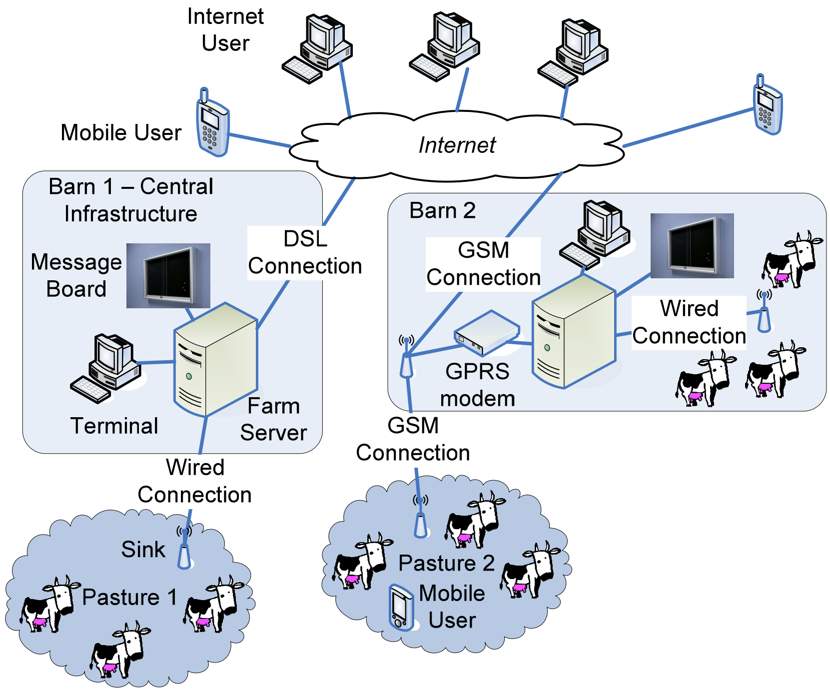


**Figure 4.6. Example deployment**

It is also possible to deploy target system without any fixed infrastructure, i.e. without sinks, farm servers, farm terminals etc. This limits the functionality but considerably decreases the deployment costs. In such case the only way for the users to access the input and output data of the system would be in-situ queries. The aggregation of the data in order to asses the efficiency of the pasture would be performed by the user's mobile device.

# 5. Energy Efficient Route Discovery

This chapter proposes a realistic, energy efficient MANET routing algorithm, Energy Efficient Route Discovery (EERD), for the cattle monitoring system. It concerns sending data from animal mounted nodes to sinks and performing in-situ queries. This chapter is divided into three sections. Section 5.1 presents the design space. Section 5.2 gives an overview of the proposed routing algorithm. Section 5.3 presents in detail the proposed techniques for limiting and balancing energy usage of the animal mounted nodes. Section 5.4 discusses how the proposed routing algorithm handles disconnections, i.e. splitting of the topology into the separated islands of connectivity. Section 5.5 considers sending data from farm servers to animal mounted nodes.

## *5.1. Design Space*

In order to allow extending coverage while preserving energy efficiency (i.e. low transmission power) and to allow circumventing of obstacles in radio propagation we need the multi-hop ad hoc connectivity between mobile nodes. This can be achieved by a MANET routing algorithm. Due to characteristics of our scenario such algorithm should be optimized for energy efficiency and handling disconnections.

The design space for the energy efficiency of the routing algorithm is shown in Figure 5.1. The Broadcast Optimization axis represent saving energy on broadcasting queries and route discovery control packets. The relevant approaches here include Passive Clustering with Delayed Intelligence [31] and utilization of heterogeneity of nodes' mobility we propose. The Route Selection Axis represents proposed selecting routes which potentially have the maximal lifetime. The vertical axis, Transmitter Power Control represents saving energy by minimizing transmitter power. The relevant approach here is similar to the transmitter power control utilized in Energy Saving Dynamic Source Routing (ESDSR) [27] or Distributed Power Control (DPC) [28]. The proposed routing algorithm is a combination of these techniques.

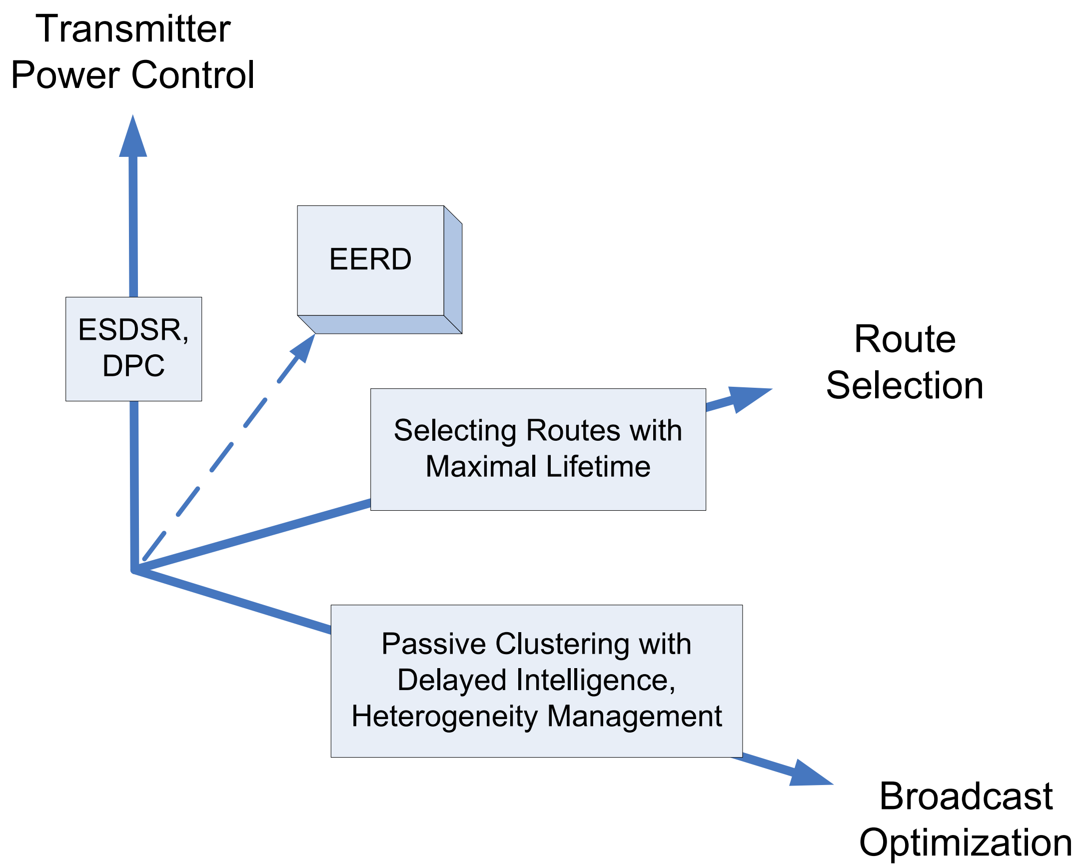


**Figure 5.1. Energy efficiency design space**

### *5.2. Overview*

Energy Efficient Route Discovery (EERD), for cattle monitoring system minimizes and balances energy consumption in the face of low data traffic and high mobility of nodes. As shown in Figure 5.2, it decreases energy spent on route discovery and in-situ queries by utilization of the tailored PCDI broadcasting. The number of necessary route discoveries is decreased by utilization of heterogeneity of nodes' mobility, selecting routes with longest lifetime and opportunistic route discovery. This algorithm also deals with disconnections by cooperative detection of route availability. It is based on the established MANET routing algorithm, DSR [18]. DSR was selected instead of Ad-hoc On-demand Distance Vector Routing (AODV) [19] because due to the application of PCDI the duration of the route discovery is difficult to estimate, which collides with expiry times of AODV dynamic routing table entries. Too long expiry time of these entries would highly increase the amount of soft state maintained by the nodes. In contrast too short expiry time would prevent routes with higher number of hops from working. The only advantage of AODV over DSR are shorter control packages in the case of routes with higher number of hops [19], which were not experienced in the evaluation reported in Chapter 6.

Figure 5.3 shows that EERD balances and saves energy on routing data by monitoring average speed of the nodes, remaining battery capacity of the local node, energy attenuation of the received and overheard packets, as well as acquiring routes from overheard and forwarded packets.

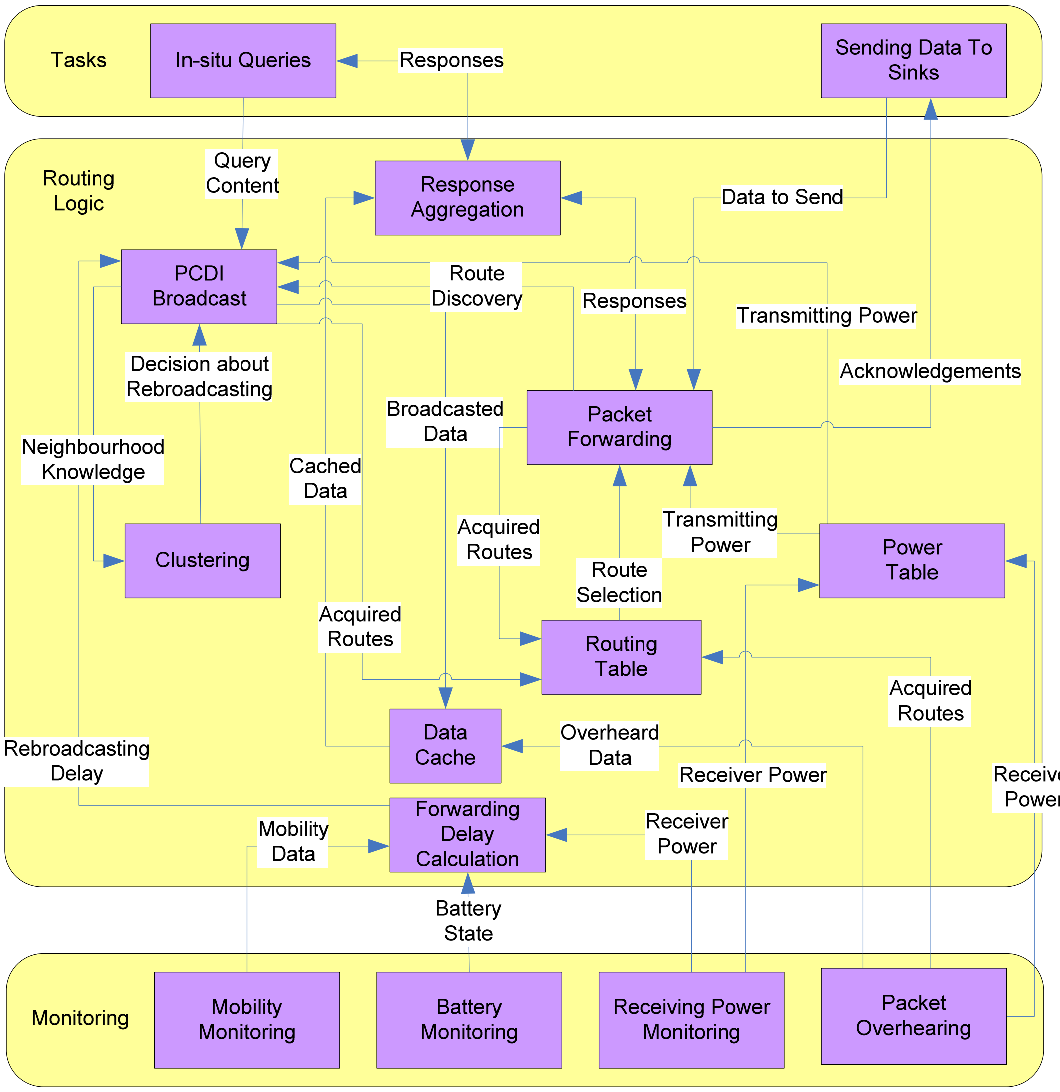


**Figure 5.2. Routing algorithm for data off-loading and in-situ queries**

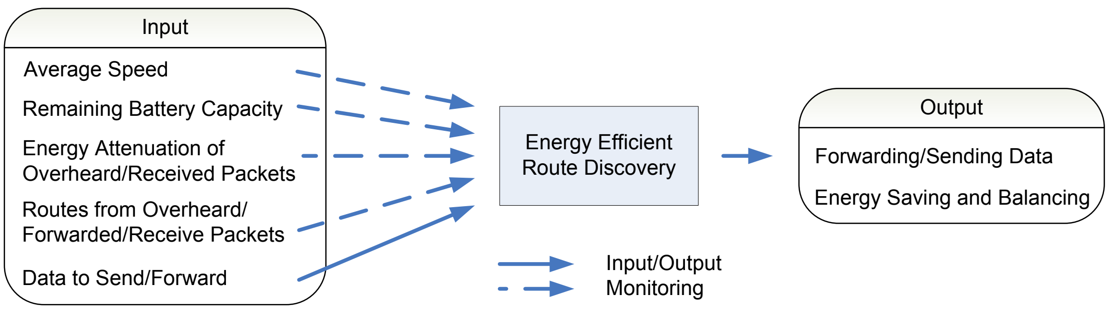


**Figure 5.3. Input and output of the Energy Efficient Route Discovery**

## *5.3. Energy Saving Techniques*

### 5.3.1. Decreasing and Balancing Energy Spent on Route Discovery

As in ESDSR [27] nodes put the utilized transmitter power in the packets so that each node can track power necessary to contact its single-hop neighbours using the following formula:

$$P_{min} = P_{tx} - P_{recv} + P_{threshold} + P_{margin} \quad (5.1)$$

where $P_{min}$ is the minimal required power for the sender to use, $P_{tx}$ is the current transmit power, $P_{recv}$ is the current received power, $P_{threshold}$ is the threshold power level for the application, and $P_{margin}$ is the margin to safeguard against changes such as channel fluctuation and mobility. All the values are in dBm. Note that only route requests and other broadcasted packets are sent using the maximal power of the transmitters.

The proposed algorithm minimizes and balances energy spent on route discovery control traffic at the cost of the energy efficiency of data traffic. This is promising because the amount of exchanged data is low and power spent on sending data packets is minimized by limiting the transmitter power. The latter is possible because the power necessary to send data over each hop is known from monitoring power attenuation between neighbours. The power of route discovery broadcasts cannot be similarly decreased because it would decrease the probability of finding any route.

As shown in Figure 5.2, energy spent on route discovery is minimized and balanced by applying Passive Clustering with Delayed Intelligence (PCDI) [31] to route request broadcasts. Note that broadcasted packets are sent using maximal transmitter power so power of the received broadcasts can still be utilized to calculate PCDI waiting time (see Formula 2.20). In PCDI nodes with higher battery capacity are more likely to route broadcasted packets so discovered routes lead through these nodes. This results in more fair energy utilization of data traffic.

### 5.3.2. Decreasing Number of Route Discoveries

#### 5.3.2.1. Utilizing Heterogeneity of Node's Mobility

The field experiments reported in Chapter 3 show that there are considerable differences between typical movement speeds of animals carrying wireless nodes. The proposed algorithm decreases chances that faster wireless nodes become members of the route by delaying their rebroadcasting of PCDI broadcasts (see Figure 5.2). In this way the lifetime of the discovered routes is extended so repeated sending of data, route failure packets and route discovery broadcasts can be minimized.

Each mobile node stores the 24 hour time series of its momentary speed received from the pedometer – expressed as number of steps per time unit. An average speed is calculated over this time series discarding time when an animal did not move. The 24 hour time period is motivated by limited resources of the nodes and the 24 hour movement pattern cycle of the animals. In particular, animals' mobility fluctuates within 24 hour cycles (see Figure 3.7), so using 24 hour time series gives more stable average values. E.g. a cow is not considered slower because she is eating at the moment. Each transmitted packet has a piggybacked maximal and minimal average speed of a node. These values are updated and stored by the forwarding nodes. Each node resets these stored values after a timeout to account for the changing conditions. This data allows nodes to asses their mobility in relation to other nodes. In EERD the

PCDI formula calculating waiting time (Formula 2.20) is extended by taking into account the average speed of the node in relation to average speeds of other nodes:

$$W = \delta \times \frac{receivedPower}{localEnergy} + \varepsilon \times \frac{V_L - V_{MIN}}{V_{MAX} - V_{MIN}}, \qquad (5.2)$$

where $\delta$ and $\varepsilon$ are constants adjusted for the particular hardware, $V_L$ is the average speed of the local node, $V_{MIN}$ and $V_{MAX}$ are minimal and maximal average speeds of the neighbourhood nodes. In this way, relatively faster nodes wait longer to rebroadcast PCDI broadcasts so their probability of becoming PCDI clusterheads or gateways and later forwarding data traffic is smaller.

#### 5.3.2.2. Selecting Routes with Longest Lifetime

The number of route discoveries is further minimized by selecting routes with potentially longest lifetime. Because of the high mobility of the nodes the life of a route is typically terminated not by the exhausted battery capacity but by the change of the topology.

As shown in Figure 5.2, utilizing received, forwarded and overheard packets a node monitors how the energy attenuation changes between the one hop neighbours. In this way a node can count how many links within the multi-hop route are increasing their energy attenuation (deteriorating). In particular each forwarded route request and acknowledgement packet contains a counter of deteriorating hops. The size of the counter is only one byte so it does not considerably increase the network overhead. This counter is incremented by the forwarding nodes which received such packet over a deteriorating link. Note that it is possible to measure changes of the energy attenuation for such link because the measuring node hears twice from the other end of the link: first time when the other end is forwarding the route request or data packet and second time when it is sending route response or acknowledgement.

Finally, as shown in Figure 5.4, a node selects routes which have (1) *the least number of hops*. For routes with the same number of hops, a node chooses these with (2) *the least number of deteriorating links*. If this is equal one with (3) *the minimal total power* (i.e. sum of the transmitter power necessary to send data over each hop) is selected. The rationale behind (1) is that on average the fewer nodes are required to take part in routing the longer it takes before one of them moves out of the wireless range of its neighbours. (2) is used to avoid routes comprising hops between nodes moving away from each other. (3) is motivated by assuming that the power attenuation between two nodes is in most cases proportional to square distance between them. Therefore, selecting routes with minimal total power tends to select the routes leading through nodes which are closer together. Such nodes are likely to need more time to leave each other's range.

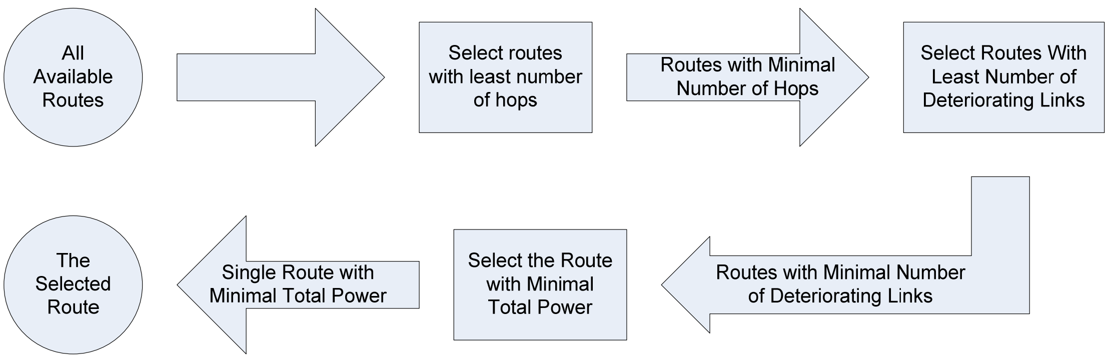


**Figure 5.4. Route selection algorithm**

Note that selecting a more optimal route does not involve exchanging additional packets. The selection of a route is performed in two cases. The first case is when a node wants to send data and finds multiple routes to the target node – one of them can have been acquired from a route discovery and the rest from forwarding or overhearing packets. The second case is when a node, which is due to rebroadcast a route request, finishes waiting enforced by Delayed Intelligence [31]. If it receives multiple route requests originating from a single source but arriving over different routes, it rebroadcasts the one which has travelled minimal number of hops and over links with minimal power. Number of deteriorating links is not considered at this stage because nodes are unlikely to have recent data about them. Nodes not performing PCDI delay, rebroadcast only the first packets they receive from each broadcast.

Overall power of a route is calculated incrementally by adding the power necessary for sending data over subsequent hops. The partial result is carried by packets such as route requests, route replies and acknowledgements. In the case of route requests this is necessary for selecting the optimal route for further forwarding. In the case of route replies and acknowledgements this is necessary for opportunistic route acquisition from forwarded and overheard packets. A node rebroadcasts more than one route reply for a single route discovery attempt only if subsequent replies contain better routes.

Route replies contain a timestamp indicating when the route was discovered. The target node sets this to the current time when replying to a route request packet. Nodes store this value in their Routing Tables (RTs). When a node replies to the route request with a route from its RT, it inserts the route timestamp from its RT. This timestamp is used to drop routes from the RT which exceed their validity period. This requires nodes to have real-time clocks but their required precision is limited to seconds.

#### 5.3.2.3. Opportunistic Route Acquisition

An important way of limiting the number of route discoveries is collecting routes from overheard and forwarded packets such as route replies and data traffic (see Figure 5.2). The gain from overhearing depends on the utilised wireless networking interface, in particular how much the power consumed by transmitting is greater than the power consumed by receiving and what is the difference in power consumption between promiscuous and non-promiscuous mode.

The sink always acknowledges receiving data. In order to account for possible disconnections, if no acknowledgement is received delivery is repeated after a timeout. In this way it is possible to opportunistically collect routes not only from forwarded or overheard route replies but also acknowledgements. For that purpose acknowledgements similarly to route replies carry aggregated power of the route and the counter of deteriorating links.

### 5.3.3. Saving Energy on Broadcasts in In-situ Queries

A mobile user collocated with the animals can issue both regular queries and directed queries. The answer to a regular query is a group of animal ids (or their custom nicknames) that fulfil a given logical condition (e.g. all animals which are sick). The user broadcasts the query using PCDI with the proposed optimizations (see Figure 5.2). All the nodes that know any partial answer to the query send the answer back to the user, together with the timestamp of the data based on which answer was generated. The answer is sent back along the route traversed by the query. Nodes that forward the queries assemble and filter these answers according to their timestamps in order to reduce redundant traffic. The final assembly is performed by the user's device.

Directed queries concern data about a particular animal (e.g. predicted date of the next oestrus). To receive the answer to such a query a user's device sends a broadcast using PCDI with the proposed optimizations to retrieve the route and the hardware address of the node that has the most recent data about the animal of interest if the user's device does not already have this information in its cache. This node could be a device that produced or caches the required data, or a sink which can retrieve this data from a server. Then the user's device sends the query along the discovered route selected according to the cost metric proposed above. Finally, the queried device sends the answer back along the same route.

## *5.4. Handling Disconnections*

The proposed MANET routing algorithm handles disconnections, which within this thesis mean splitting of the network topology into separated islands of connectivity. In the case of sending data to sinks the data is sent only when the multi-hop path between an animal mounted node and any of the sinks exists. It is detected using the proposed *cooperative detection of route availability*, shown in Figure 5.5. More precisely, if the route discovery is unsuccessful it is repeated after a certain timeout with a small random delay. The purpose of the random delay is preventing the broadcast storm caused by multiple nodes initiating route discovery at the same time. In order to save energy on the repeated unsuccessful route discoveries if the route discovery is unsuccessful the node that initiated it broadcasts a negative acknowledgement. In this way all the nodes within its island of connectivity know that the route to the sink is not available and the route discovery should be repeated no sooner than after the predefined timeout. Otherwise if a node receives a route request packet but no negative acknowledgement, this means that a route to a sink exists so the node can try to discover it. The negative acknowledgements are preferred here over positive ones to save energy in circumstances when no disconnections take place – e.g. animals are located in a barn.

When a sink receives data from an animal mounted node it sends an acknowledgement. If no acknowledgement is received the animal mounted node resend the data over a different path and if it does not know any alternative path it initiates route discover.

In order to answer the in-situ queries in the face of disconnections the animal mounted nodes should be able to answer the query within the island of connectivity (network partition). To achieve that, nodes cache data sent to sinks which they forward or overhear. This caching is performed according to their available storage space. The proactive caching, i.e. the proactive exchange of the data for the purpose of caching, is not advisable here due to the energy constrains [102]. If the sink is connected to the farm server over an expensive third party connection such as GPRS, it may also cache the data forwarded to farm servers. In this way the sink can support answering in-situ queries without the need to query the farm server.

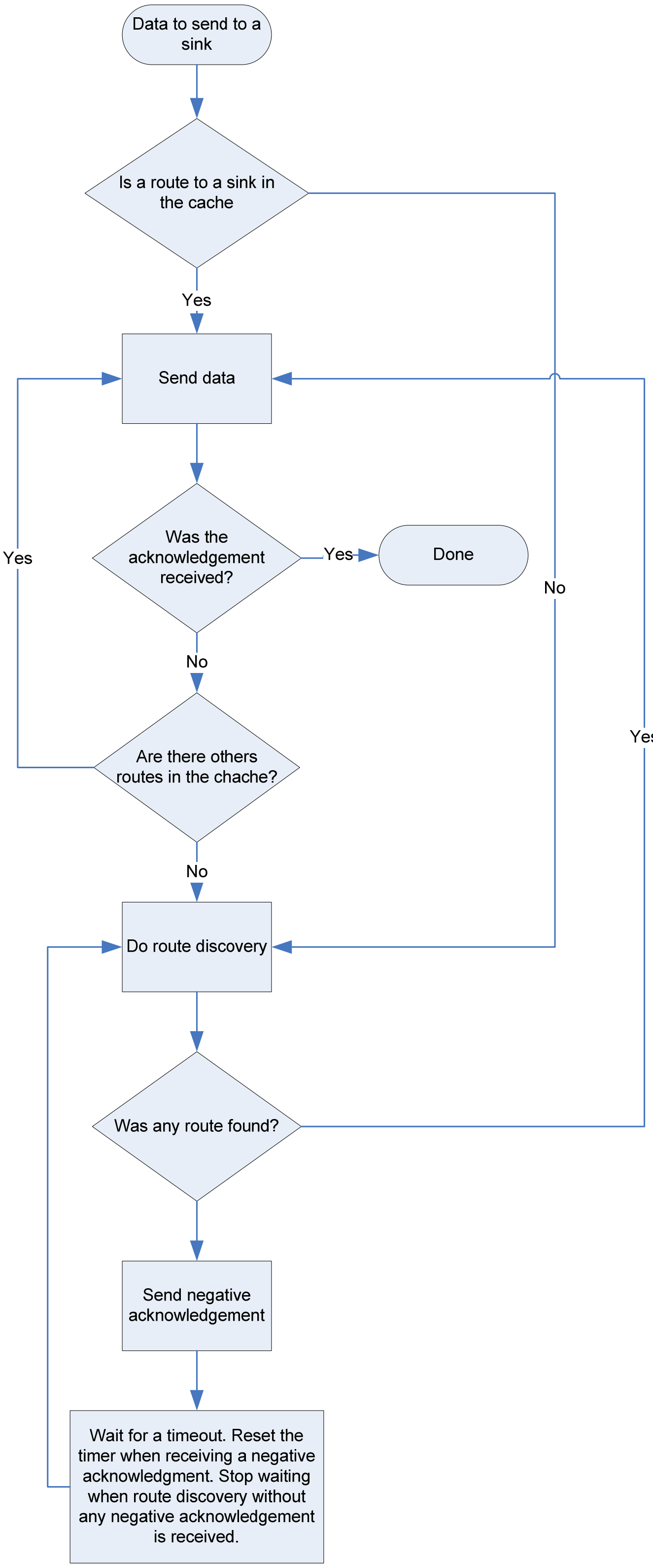


**Figure 5.5. Cooperative detection of route availability**

Nodes receiving an in-situ query answer it whenever they have at least a partial answer to this query. This answer can come from locally produced or cached data. If the in-situ query is received by the sinks, the sink may answer it after fetching appropriate data from the farm server or its local cache. In the case of direct queries nodes forward the answers to the queries only when the answer was based on the data which is newer than in the case of answers already forwarded.

## 5.5. Sending Data from Farm Servers to Animal Mounted Nodes

Sporadically the farm servers may need to send data to the selected or all animal mounted nodes. This data can be for example a firmware or configuration update. Such communication is similar to sending data from animal mounted nodes to farm servers. In particular the farm servers keep track of associations between the animals and pastures or barns where they are kept so they know to which sinks data should be forwarded. After receiving this data a sink performs route discovery (similarly as an animal mounted node) and sends the data to the given animal mounted nodes. If a route does not exist it retries after a timeout.

If there is more than one sink collocated with the target animal mounted node the first sink which manages to successfully deliver data to the target node can inform about this other collocated sinks so that the target node does not receive duplicates. To prevent duplicates being delivered at the same time each sink can wait a random delay before sending the data. If the instant communication between sinks is not possible e.g. they are connected to farm servers by data couriers or GSM then the animal mounted nodes may receive duplicates but this type of communication is sporadic so these duplicates do not make a considerable difference for energy efficiency.

# 6. Evaluation

This chapter reports on the evaluation of the proposed architecture for the cattle monitoring system, and its core, MANET routing algorithm - EERD. As the method of evaluation we selected emulation, i.e. simulation utilizing data from the real experiments. This approach offers a satisfactory compromise between realism, variety of examined conditions, number of observed parameters and utilization of resources. In particular it offers higher realism than the purely stochastic simulation. Whereas in comparison to purely experimental evaluation emulation offers higher variety of examined conditions and more observed parameters for the same constraints (i.e. time and funds). We compared the proposed routing algorithm with DSR [18] – a classical MANET routing algorithm and ESDSR [27] – an example energy efficient routing algorithm. We emulated the communication scenarios, which are realistic for the proposed cattle monitoring system but also sufficiently challenging for the emulated algorithms to demonstrate benefits of the proposed algorithm. In order to increase realism of the simulation the movement patterns of mobile nodes are emulated utilizing data from the field experiments instead of utilization of generic stochastic models such as Random Waypoint Model [18] or Reference Point Group Mobility Model [103]. These models were devised to simulate mobility of people and it is very difficult to adjust their parameters to make them reflect mobility of bovine animals.

This chapter is organized in the following way. Section 6.1 presents the emulator of bovine movements developed for the purpose of emulation. Section 6.2 compares the proposed MANET routing algorithm, EERD, with the existing approaches, namely DSR and ESDSR. Section 6.3 evaluates the specific techniques introduced or utilized in the proposed routing algorithm. Section 6.4 discusses how the proposed architecture and its core, the EERD routing algorithm, satisfy the criteria identified in Chapter 2.

## *6.1. Bovine Movement Emulator*

In order to make a realistic packet level emulation involving up to 100 nodes we implemented an emulator of bovine movements. This emulator is informed by field experiments described in Section 3 and utilizes animal movement data from these experiments.

The emulation area is similar to the dairy where the field experiments were performed (see Figure 3.1). Each of the emulated cows is for most of the time in one of three states: (1) resting in a bay, (2) eating/drinking, (3) being milked. These states are associated with groups of locations within the division of the dairy and transitions between the states are connected with moving between locations. Selecting the next state is restricted in the following way. There is a minimal period allowed between milkings and a cow goes to a milking robot only when any of them has a queue shorter than three animals. If after reaching the robot the queue is longer than two animals, the cow changes the target state.

Speeds which the emulated cows randomly select were acquired from the GPS data. This makes the emulated cows move with similar distribution of speeds as the real animals, whose preferred speeds have gamma distribution [96] with $\Theta = 2.0$ and $k = 1$ or $k = 2$ (see Figure 3.6). Speeds higher than 1.5 m/s were filtered out under assumption that they were unavailable to the bovine animals [104] and were recorded because of GPS drift. Two different speed profiles utilizing real speeds from two different real cows were utilized (see Figure 3.6). These profiles are distributed evenly between the emulated cows.

The times a cow stays at any of the locations were acquired from the video footage. These are randomly selected for the cows during the emulation to achieve the distribution close to reality. GPS data is only utilized for acquiring resting times because in other cases the accuracy of GPS data is too low in relation to the distances between different types of locations such as feeder, water tanks, milking robots and bays. The patterns of eating and drinking and the times the cows spent performing these activities were also determined from the video footage. These patterns are also randomly selected during the emulation. The minimal period between milkings for a cow we calculated from the timestamps of the pedometer readings taken during the milkings.

## *6.2. Comparison with Existing Approaches*

### 6.2.1. Emulation Setup

This section describes how the proposed MANET routing algorithm was compared with the existing approaches including DSR [18] and ESDSR [27]. DSR was selected as a classical MANET routing algorithm and ESDSR as an example energy efficient MANET routing algorithm.

The proposed algorithm was evaluated using the ns-2 [105] network simulator, best suited for the MANET character of our scenario. The algorithm was implemented in C++ as a wireless routing agent [106]. In order to allow processing of the packets overheard by nodes the tap function was enabled.

As shown in Figure 6.1, Bovine Movement Emulator (BME) described in Section 6.1 was utilised to generate mobility traces for ns-2. Ns-2 generated wireless traces which were then processed using Python scripts to measure the observed parameters. Then in the case of one of the emulated scenarios, which were emulated in several iterations to average the results, statistics from all the iterations were aggregated.

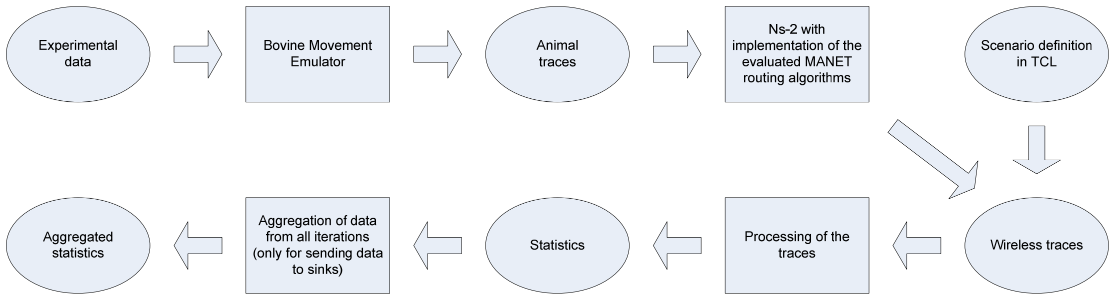


**Figure 6.1. Emulation environment**

We emulated two scenarios which reflect realistic communication patterns within the proposed cattle monitoring system and are sufficiently challenging for the simulated routing algorithms to demonstrate differences in their performance: (1) animal mounted nodes sending data to a sink, (2) one stationary user querying animal mounted nodes. In both cases data traffic starts after 1 hour to let the emulated animals leave their initial positions. At the beginning of simulation the animal mounted nodes already know their average speed in relation to the maximal and minimal average speed of the other nodes.

In the first scenario animal mounted devices try to send once 32B of data to the stationary sink, which models the regular daily update sent to the farm servers (see Section 4.5). 32B reflects the amount of data from animal mounted pedometer, accelerometer and results of processing made by animal mounted nodes such as detected animal diseases, date of last oestrus etc. They start after 1 hour, randomly distributed over 5s to take advantage of passive acquisition of routes. They perform the route discovery if they do not already have a route to the sink in their cache (from overheard or forwarded packets). The whole emulation lasts for 3 emulated hours. In this scenario for each set of parameters we repeat the emulation 5 times with different random values for BME and ns-2 and then average the results. In the second case the user broadcasts 20 queries. Each node replies to the query with probability 0.25 with 32B of data. This emulates range queries. Each subsequent query is submitted 10s after receiving the last answer to the previous query.

To evaluate the scalability of the evaluated routing algorithms the number of animals was altered. The observed parameters include:

- number of transmitted packets over the time
- minimal, average and maximal energy usage per node over the course of the emulation and its standard deviation (we consider only the animal mounted nodes)
- number of nodes with exhausted battery capacity at the end of emulation
- minimal, maximal and average delays and their standard deviation
- success ratio

Delays mean here in the case of sending data to sinks the time from the moment when data is sent until successful receiving of the acknowledgement by an animal mounted node. In the case of in-situ queries delays mean time from sending the query to receiving the answer. Success ratio means in the case of sending data to a sink the fraction of nodes that successfully delivered data to sinks. In the case of in-situ queries we measured two different success ratios. Success ratio for queries is calculated using the following formula:

$$SR = \frac{N_{RQ}}{N_Q \cdot N_A}, \tag{6.1}$$

where $N_{RQ}$ is the number of receptions of a query by an animal mounted node, $N_Q$ is a number of issued queries and $N_A$ is a number of animal mounted nodes. If the animal mounted node receives the same query more than once only the first case is considered. Success ratio of responses is the fraction of responses that were successfully returned to the user. The standard deviation was calculated using the following formula:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}, \tag{6.2}$$

where $N$ is the number of samples, $x_i$ is the sample value and $\bar{x}$ is the value of the arithmetic average.

The maximal power of the transmitter is 0.85872mW (i.e. power consumed by the transmitter and power of the transmitted signal), which gives the maximum transmission range of 40m. According to [45] this gives parameters closer to those found in sensor radios. Since the receiving power is constant and a fixed amount of energy is dissipated when a node receives a packet, receiving power is ignored (modelled as zero). The authors of ESDSR made a similar assumption [27]. At the beginning of emulation the sink and the user has 1000J (effectively infinite energy) and animal mounted nodes have 1J. $P_{margin}$ in Formula 5.1 is 1. We use the following EERD parameters: $\alpha$=1, $\beta$=1, $\delta$=10000s, $\varepsilon$=0.5s (see Formulae 2.19 and 5.2), reverting to the initial state and discarding received states of neighbours after 60s. The route validity period is 60s and waiting for route replies lasts 1s.

### 6.2.2. Emulation Results

Emulation results are shown in Figure 6.2, Figure 6.3, Figure 6.4 and Figure 6.5. In Figure 6.3 and Figure 6.5 points and lines show average values per node or standard deviation. Error bars show minimal and maximal values. In each examined case no node exhausted its battery capacity.

Figure 6.2 shows the time graphs for transmitted packets in the case of mobile nodes sending data to the sink. The initial high network traffic is caused by route discovery. Later more nodes have valid routes in their routing tables so intensive broadcasting can be avoided. This demonstrates how important it is to optimize route discovery traffic.  We can see that this objective was achieved by the proposed protocol EERD. The difference is particularly significant for high number of mobile nodes (100) but also visible for low number of nodes (10). The visible optimization of route discovery can be attributed to utilization of PCDI with proposed extensions.

Figure 6.3a shows energy utilized by animal mounted nodes for sending data to the sink. EERD considerably decreases average energy usage in comparison to DSR and ESDSR (48%-75%). The proposed algorithm considerably balances energy utilization compared to DSR and ESDSR. Figure 6.3b shows that EERD has standard deviation of energy utilization by 76%-91% smaller than DSR and ESDSR. These improvements can be attributed to PCDI with proposed optimizations and proposed metrics for selecting routes.

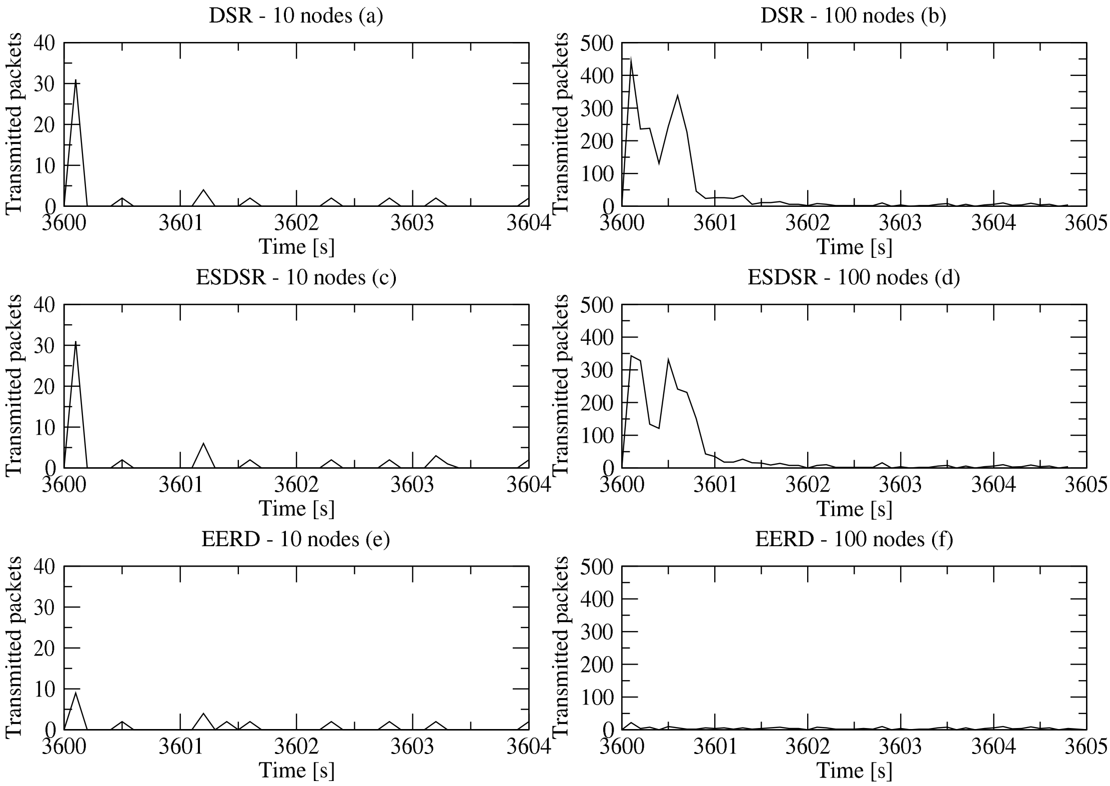


**Figure 6.2. Comparison with existing approaches – time graphs for sending data to sinks**

Figure 6.3c shows delays for sending data to the sink. We can see that in the case of DSR and ESDSR the delays grow with the number of nodes, whereas in the case of EERD the delays are almost constant. Figure 6.3d shows the average deviation of delays. In the case of DSR and ESDSR it grows much faster with the increasing number of nodes than in the case of EERD. This means better scalability of the EERD in comparison to DSR and ESDSR, which can be attributed to reduced network overhead achieved by utilization of PCDI.

Figure 6.3e shows the success ratio (SR) for delivering data to sinks. We can see that in all examined cases the SR is very high. Nodes do not repeat failed attempts otherwise the SR would be even higher. In the case of DSR and ESDSR SR drops slightly for the higher numbers of nodes (to 0.94 and 0.95 respectively for 100 nodes).

It is not the case with EERD. This can be attributed to avoiding congestion achieved by utilization of PCDI.

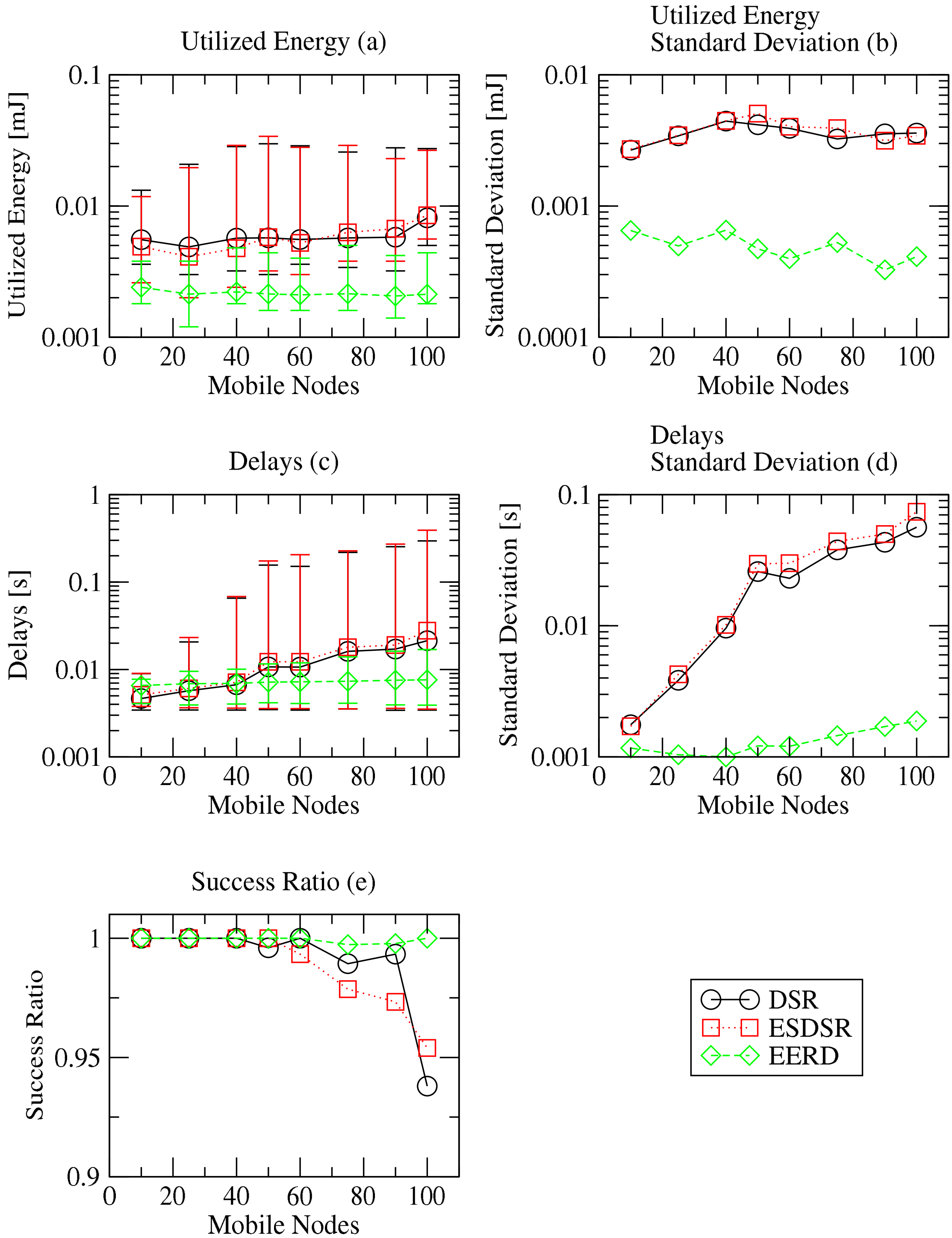


**Figure 6.3. Comparison with existing approaches – statistics for sending data to sinks**

Figure 6.4 shows the time graphs for answering in-situ queries. Each peak represents a single query issued by the user. We can clearly see that the traffic generated by EERD is much lower than traffic generated by DSR and ESDSR. The difference is visible in the case of high (100) and low (10) number of mobile nodes.

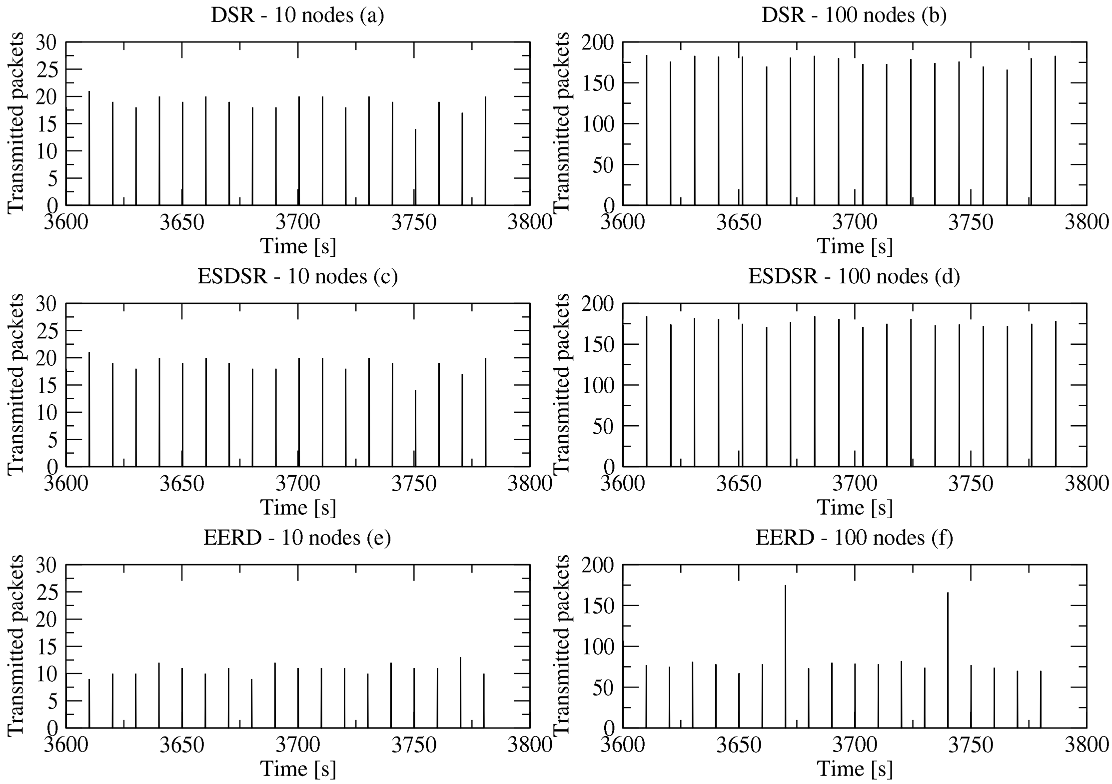


**Figure 6.4. Comparison with existing approaches – time graphs for in-situ queries**

Figure 6.5a shows the energy utilized by animal mounted nodes for answering in-situ queries. The amount of utilized energy is comparable to the case of communication with the sink, which justifies optimization of this type of communication. The amount of utilized energy is almost constant for each of the protocols regardless of the number of nodes. The considerable decrease of average energy utilized by the proposed algorithm in relation to the existing routing algorithms (by 77-82%) is achieved by optimization of broadcasting queries. Figure 6.5b shows that the standard deviation of utilized energy is much higher for EERD than for other compared algorithms for the very sparse topology (10 nodes). Then the EERD's standard deviation drops sharply for 25 nodes and stays almost constant. In contrast the standard deviation of DSR and ESDSR grows with the number of nodes. This demonstrates better scalability of EERD in terms of energy usage achieved by optimized broadcasting.

Figure 6.5c shows delays in answering in-situ queries. The delays grow linearly with the number of animals. In the case of the proposed routing algorithm this increase is lower, which means better scalability. This can be attributed to the decreased network congestion caused by the proposed optimization of broadcasting. For 100 mobile nodes EERD achieves up to 57% of decrease in average delays and up to 29% in maximal delays. The decrease of delays in answering in-situ queries is very important as this improves the usability of the system. Figure 6.5d shows the average deviation of delays. It grows linearly with the number of nodes but this growth is much higher in the case of DSR and ESDSR than in the case of EERD. This gain is achieved by utilization of Passive Clustering and is very important for scalability.

For all examined number of nodes and routing algorithms the in-situ queries were delivered to all mobile nodes. The success ratio of delivering answers to the user's device is shown in Figure 6.5e. We can see that this success ratio decreases almost linearly with the increasing number of animals which can be attributed to the network congestion. The proposed algorithm offers however a higher success ratio for higher numbers of animal mounted nodes. This is due to the decrease in network traffic achieved by utilization of Passive Clustering. For 100 nodes the proposed algorithm has success ratio higher than DSR by 22% and higher than ESDSR by 19%.

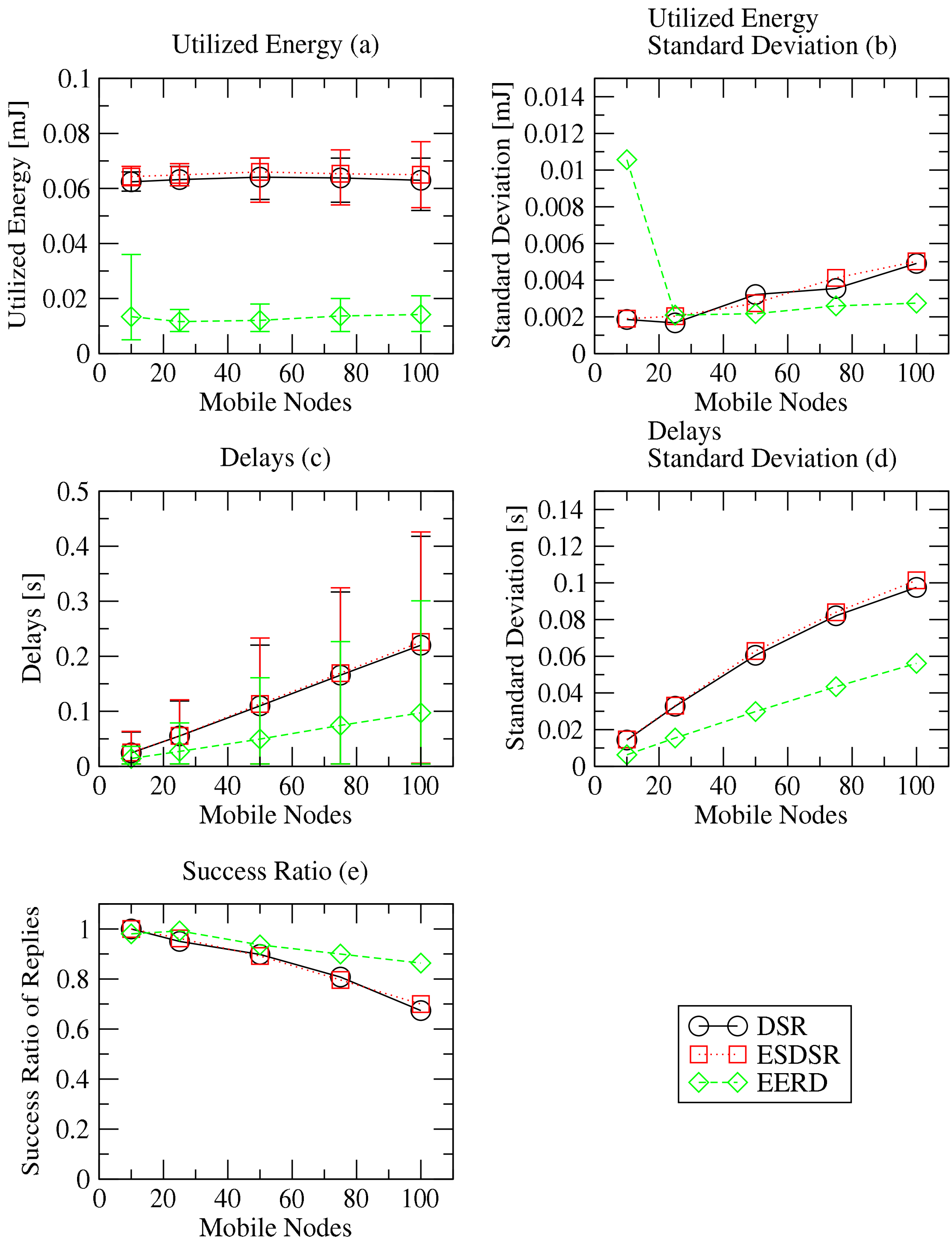


**Figure 6.5. Comparison with existing approaches – statistics for in-situ queries**

To summarize, the proposed MANET routing algorithm has lower and more balanced utilization of energy than the other compared routing algorithms. In the case of in-situ queries it also offers better scalability in terms of delays and success ratio.

## *6.3. Evaluation of the Specific Techniques Utilized in EERD*

### 6.3.1. Emulation Setup

In order to better understand the influence of the specific techniques utilized in the proposed MANET routing algorithm on its performance, the algorithm was emulated with certain its features disabled. More precisely the simulation setup was similar to the one described in Section 6.2. Instead of comparing the proposed algorithm with the existing approaches, the performance of the fully functional algorithm was compared with the performance of the same algorithm with certain its features disabled, including:

- Power control – all packets are sent with the maximal power
- PCDI [31] – the flooding similar as in DSR [18] is utilised instead
- Utilization of heterogeneity of the nodes' mobility – the original PCDI formula [31] (Formula 2.20) for calculating delays is utilized instead of the one we propose (Formula 5.2)

### 6.3.2. Emulation Results

Emulation results are shown in Figure 6.6, Figure 6.7, Figure 6.8 and Figure 6.9. In Figure 6.7 and Figure 6.9 points and lines show average values per node or standard deviation. Error bars show minimal and maximal values. In each examined case no node exhausted its battery capacity.

Figure 6.6 shows time graphs of transmitted packets for sending data from animal mounted nodes to sinks. We can see that application of PCDI has greatest impact on reducing route discovery (initial high peak) due to optimizing route discovery broadcasts. Transmitter power control increases number of the transmitted packets.

This is because nodes transmit packets with lower power so they are overheard by smaller number of nodes, which then limits acquisition of routes from overheard packets.

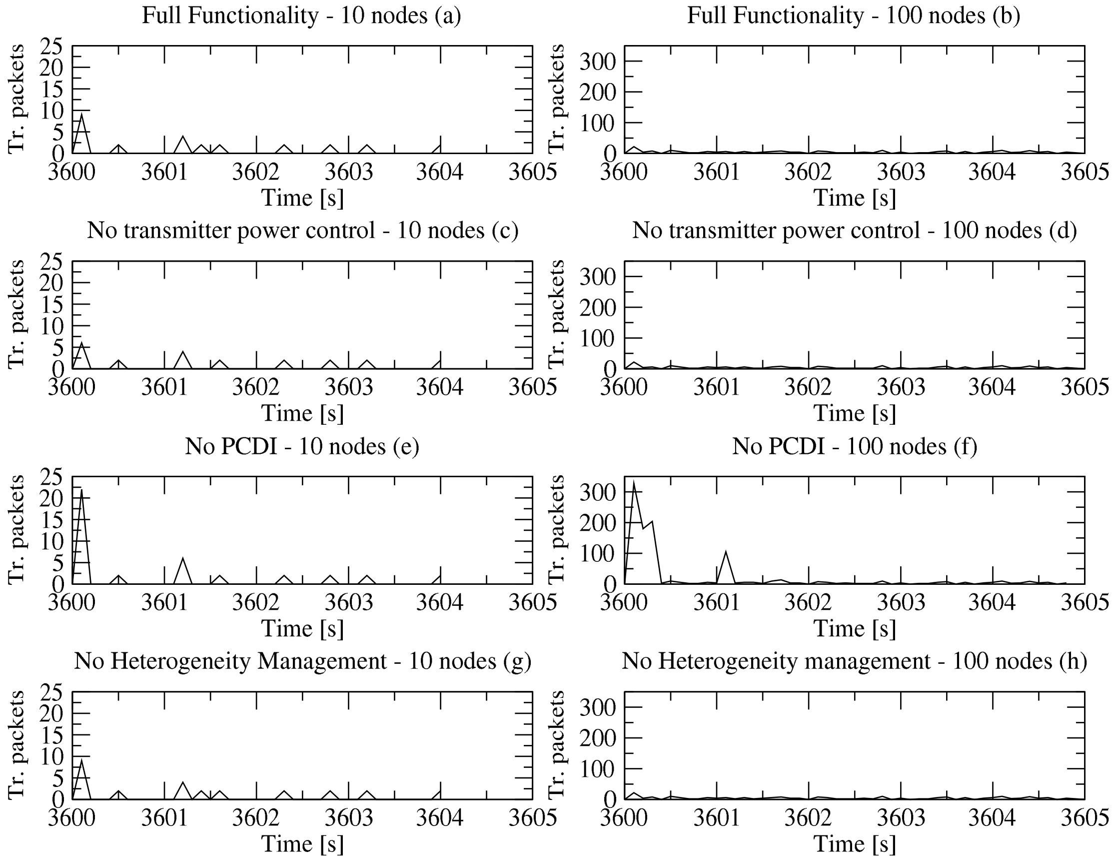


**Figure 6.6. Evaluation of the utilized techniques – time graphs for sending data to sinks**

Figure 6.7a shows energy utilized by animal mounted nodes for sending data to the sink. We can see that without PCDI the utilized energy grows with the number of nodes. Utilization of PCDI makes the energy consumption almost independent of the number of nodes. This can be attributed to the energy saved on route discovery broadcasting. Utilization of transmitter power control gives the constant advantage in average utilized energy of 22% to 31%. Heterogeneity management extension to PCDI does not make any considerable difference here. Figure 6.7b show standard deviation of the average utilized energy. Utilization of PCDI and transmitter power control increases standard deviation. This is the cost of achieving lower average energy usage.

Figure 6.7c shows delays for sending data to sinks. Without utilization of PCDI delays grow slightly with the number of nodes otherwise their average is almost constant. Figure 6.7d shows standard deviation of delays. We can see that utilization of PCDI make it grow slower with the number of nodes.

Figure 6.7e shows the Success Ratio (SR) for delivering data to sinks. The SR is very high. The nodes do not repeat failed attempts otherwise the SR would be even higher. We can see that utilization of PCDI slightly improves SR.

Figure 6.8 shows time graphs of transmitted packets for in-situ queries. We can see that only PCDI has any visible influence on reducing network overhead. This is due to the fact that most of the energy in the case of in-situ queries is spent on broadcasting queries and PCDI optimizes this type of traffic.

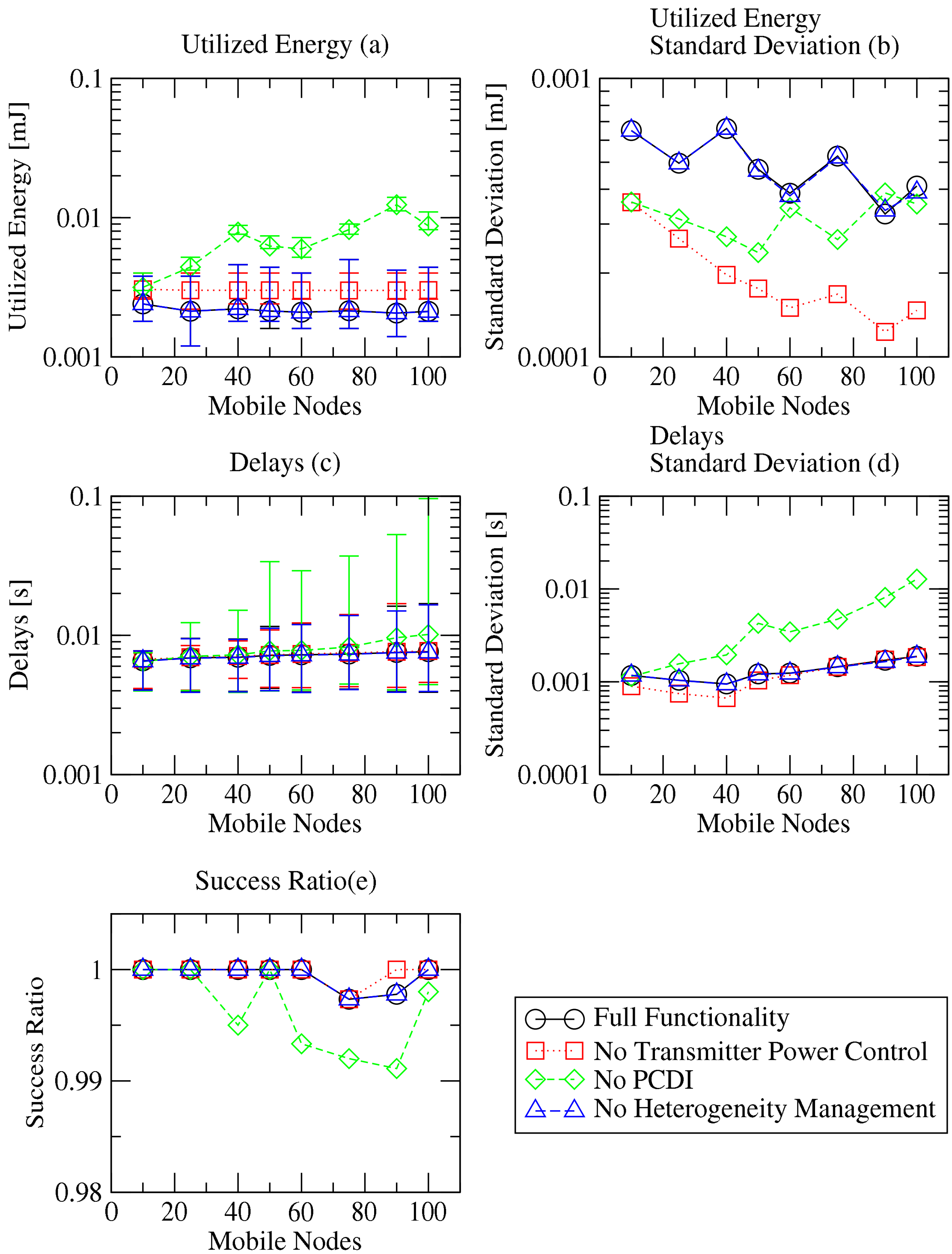


**Figure 6.7. Evaluation of the utilized techniques – statistics for sending data to sinks**

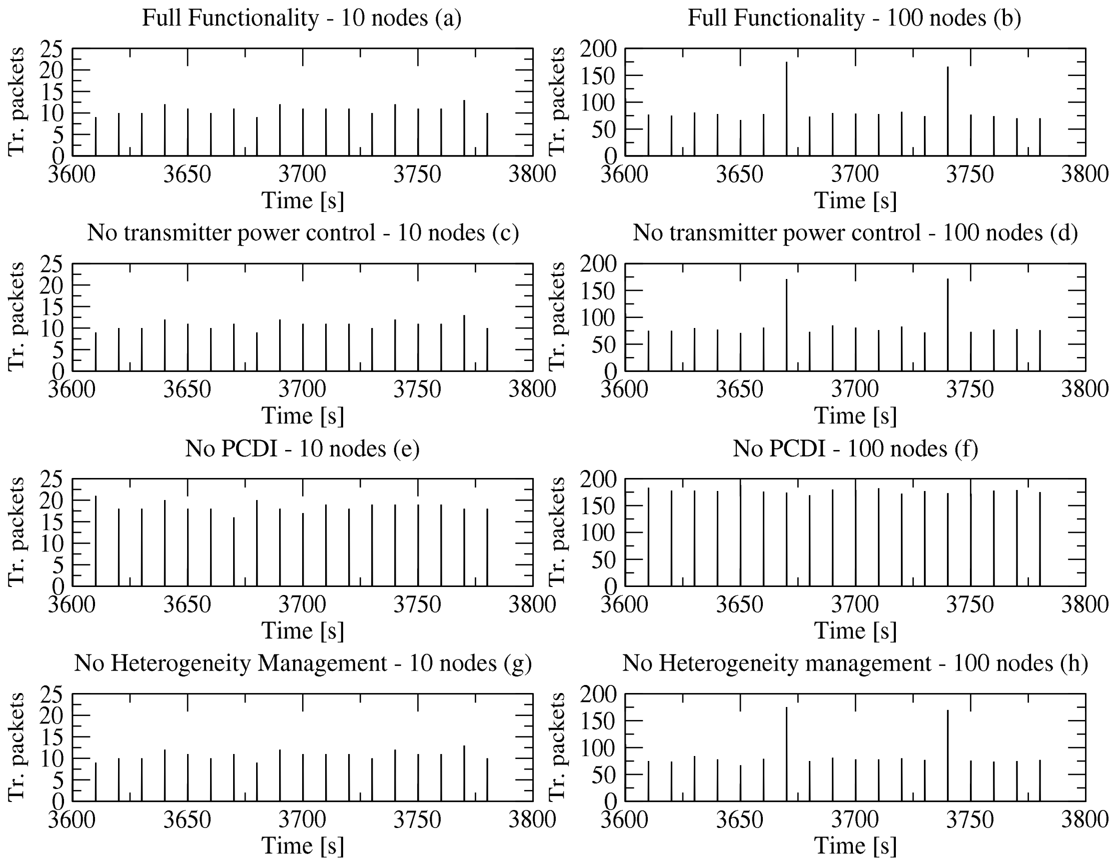


**Figure 6.8. Evaluation of the utilized techniques – time graphs for in-situ queries**

Figure 6.9a shows the energy utilized by animal mounted nodes for in-situ queries. For all the examined cases the amount of utilized energy hardly depends on the number of mobile nodes. We can see that the most important decrease of energy utilization results from using PCDI and transmitter power control. PCDI decreases average energy utilization by 57-64% and transmitter power control by 33-40%. Figure 6.9b shows the standard deviation of the energy utilised for answering in-situ queries. We can see that PCDI highly increases this deviation for small topologies (10 nodes) but decreases it for topologies of medium size (25-75 nodes). The reason of high influence of PCDI on limiting utilized energy is decreasing network overhead caused by broadcasting of queries (see Figure 6.8).

Figure 6.9c shows the delays of answering in-situ queries. We can see that they grow linearly with the increasing number of nodes but utilization of PCDI makes this growth smaller. Figure 6.9d shows that PCDI also decreases the standard deviation of the delays. These gains can be attributed to the decreased network congestion resulting from optimization of broadcasting.

For all examined cases the in-situ queries were delivered to all mobile nodes. The success ratio of delivering answers to the user's device is shown in Figure 6.9e. We can see that this success ratio decreases with the increasing number of animals which can be attributed to the network congestion. Utilization of PCDI decreases the network congestion and thus improves the success ratio.

To summarize, the proposed MANET routing algorithm EERD provides lower and more balanced energy usage than the classic, non-energy aware DSR and the more generic existing energy aware routing algorithm ESDSR. In the case of in-situ queries EERD makes success ratio and delays deteriorate slower with the increasing number of nodes thus improving the scalability in comparison to DSR and ESDSR.

The most important impact on saving and balancing energy has utilization of PCDI to optimize broadcasting of route discovery packets and in-situ queries. In the case of in-situ queries PCDI provides shorter and more stable delays as well as higher success ratio.

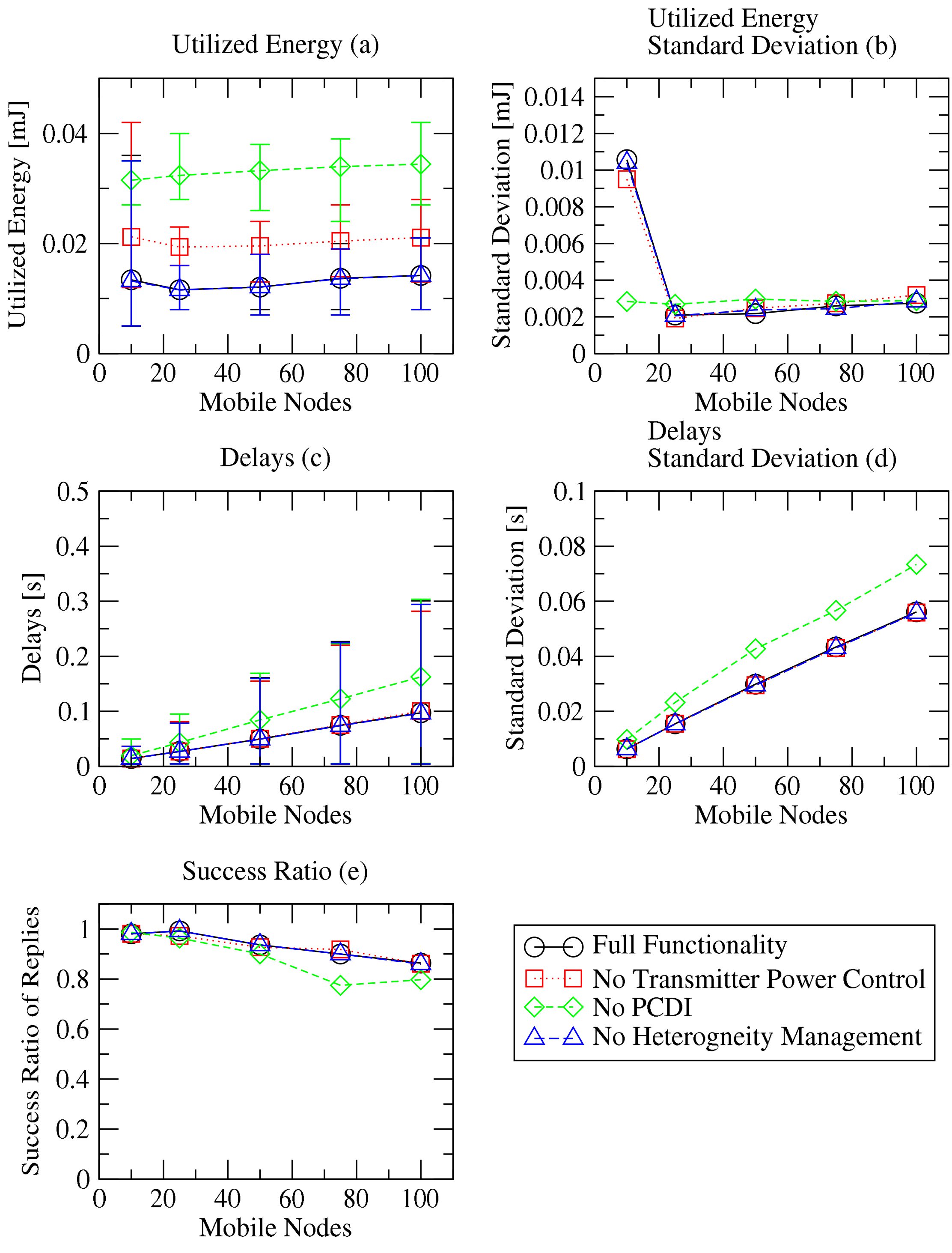


**Figure 6.9. Evaluation of the utilized techniques – statistics for in-situ queries**

## *6.4. Conclusions*

In this section we discuss how the proposed architecture and its core component, the EERD MANET routeing algorithm, satisfy the criteria identified in the Chapter 2.

### 6.4.1. Increasing Reliability

The proposed routing algorithm increases reliability of the cattle monitoring system by extending the network coverage of sinks, mobile users and animal mounted sensor nodes using multi-hop ad hoc communication and decreasing network congestion which can be responsible for delays in data delivery. Congestion is decreased by optimization of route discovery and broadcasting in-situ queries. The reliability is further increased by addressing disconnections. In the case of delivering data to sinks the disconnections are addressed using *cooperative detection of route availability*, which allows detecting and utilizing the availability of a multi-hop route to a sink. In the case of in-situ queries the disconnections are addressed by caching the data when it is sent to sinks. In this way, a query can be answered within the island of connectivity of the user issuing the query using the cached data even when the addressees of the query are outside this island of connectivity.

### 6.4.2. Managing Delays

The proposed routing algorithm manages the delays in the case of delivering data to farm servers in the following way. When the sensor node detects oestrus or an animal disease, if the multi-hop route to a sink is available the data is sent immediately. Otherwise the data is sent when such route is detected. Delays in delivering data from sinks to farm servers depend on the available infrastructure – this issue is further discussed in Chapter 7. Every 24 hours (subject to availability of the route to a sink) animal mounted nodes send the raw and processed data to the farm servers even when no event was detected in order to allow server to evaluate efficiency of pastures, detect non-functional animal mounted devices and be informed about the detected pregnancies. According to the users (see Subsection 3.1.2.2) this information is less urgent and around 24 hours delay is acceptable. In the case of answering in-situ queries, caching of the data sent to sinks allows immediate answering of the query within the user's island of connectivity.

### 6.4.3. Increasing Scalability

The scalability is mainly increased at the MANET routing algorithm level. As demonstrated in the Section 6.2, for the proposed routing algorithm the following parameters degrade less with the increasing number of nodes than in the case of more generic existing approaches such as ESDSR [27] and DSR [18]:

- Energy utilized by animal mounted nodes for sending data to sinks
- Delays in answering in-situ queries
- Success ratio of delivering answers to in-situ queries

As demonstrated in the Section 6.3 this was achieved mainly by utilization of the PCDI.

### 6.4.4. Lowering Costs

Installation costs are lowered by utilization of multi-hop wireless communication which, by extending network coverage, allows utilizing smaller number of sinks. Maintenance costs are lowered by minimizing utilization of expensive third party wireless services such as GSM as well as decreasing and balancing energy utilization of the animal mounted nodes. Third party wireless services are utilized only in certain cases to connect sinks to farm servers, when no other option is available, and to deliver notifications to users' mobiles. Decreasing and balancing energy utilization is very important because the animal mounted devices are battery powered and replacing batteries is very labour intensive. As demonstrated in the Section 6.2, the proposed MANET routing algorithm allows saving energy on sending data to sinks and in-situ queries. Section 6.3 shows that this was achieved mainly by utilization of PCDI and transmitter power control.

### 6.4.5. Handling High Mobility

Handling high mobility concerns the MANET routing algorithm as this is the part of architecture responsible for collecting data from animal mounted nodes. High mobility is addressed by on-demand character of the MANET routing and using only soft state topology information (i.e. recently collected). This is due to the fact that the topology is changing frequently. Therefore, all the collected topology data becomes quickly outdated and maintaining routes proactively would be very expensive. To increase lifetime of the routes the shortest routes, which are less likely to break, are preferred.

### 6.4.6. Addressing Lack of Mobility Supporting Communication

The animals carrying sensor nodes are not going to support wireless communication. Therefore the proposed MANET routing algorithm does not rely on their support but aims at utilizing the observed characteristics of their mobility. More precisely, the routing algorithm monitors mobility of the animals and selects for forwarding nodes carried by slower animals. The routes with the hops that offer non degrading connectivity are preferred. Degrading means here requiring increasing transmitter power.

### 6.4.7. Handling High Dynamics of Topology

In the cattle monitoring scenario density of the topology is very dynamic and can rapidly change from very sparse to very dense. In a sparse topology the multi-hop path between a sensor node and a sink may be available only sporadically. In the case of sending data to sinks we detect the availability of a route to a sink in the energy efficient manner utilizing *cooperative detection of route availability*, described in Section 5.4. The in-situ queries are answered within the user's island of connectivity thanks to caching of the data sent to sinks. In particular, nodes cache data sent to sinks which they forward or overhear.

In a dense topology there is a risk of network congestion especially in the case of broadcasts. We optimize broadcasting by utilization of Passive Clustering with Delayed Intelligence [31]. In this way only a subset of nodes rebroadcasts packets, which alleviates congestion.

# 7. System and Network Management

This chapter focuses on the management of the cattle monitoring system proposed within this thesis. The goal of this chapter is to examine management issues of the cattle monitoring system considering the criteria defined in Chapter 2, in particular user requirements including increasing reliability, managing delays, increasing scalability and lowering costs. The fundamental question is whether the deployment of the proposed architecture is desirable or even feasible in the agricultural environment. The considerable impact on preventing spread of animal diseases such as foot-and-mouth disease (FMD) or bovine spongiform encephalopathy (BSE) can be achieved only after the widespread adoption of the cattle monitoring presented within this thesis, which strongly depends of its feasibility and incentives it offers to the farmers. In this chapter we discuss necessary configuration process performed during the system installation, providing mobile and home access to the target system and maintenance action necessary for continuous functionality of the system. We propose strategies for sink installation, connecting sinks to farm servers and improving security of the system.

This chapter is organized in the following way: Section 7.1 discusses deployment of the central infrastructure which includes farm servers, message boards and terminals. It presents necessary configuration of this part of the system and discusses providing home and mobile access to the system. Section 7.2 discusses deployment of the animal mounted nodes including necessary configuration and issues related to the animal body area network. Section 7.3 discusses deployment of the sinks. It proposes the strategy for sink installation and connecting sinks to farm servers. Section 7.4 discusses maintenance activities necessary for continuous functionality of the proposed cattle monitoring system. Section 7.5 discusses interoperability issues of the target system. Section 7.6 proposes strategy for increasing security of the target system.

## *7.1. Deployment of the Central Infrastructure*

### 7.1.1. Installation

As shown in Figure 4.6 central infrastructure proposed in this thesis includes farm servers, message boards and terminals. As proposed in Chapter 4, a message board displays notifications defined by farm personnel concerning detection of animal diseases, oestrus, pregnancy etc. This can be an embedded system comprising a microcontroller, an LCD or LED display and an Ethernet or 802.11 interface. Alternatively it can be an off-the-shelf PC with an Ethernet or 802.11 network interface and a large LCD screen. Message boards are optional and their deployment depends on the budget available to the enterprise.

Installation of the central infrastructure should be as easy as possible to increase feasibility of its adoption. Ideally this should involve connecting the off-the-shelf components to network and mains. Farm personnel should also enter basic data about the enterprise such as list of the barns and pastures it possesses. The integration of the servers with the networking infrastructure should be simplified using self configuration services such as DHCP [107] to minimize or even avoid involvement of IT technicians and thus minimize installations costs.

The message boards should be placed at visible locations in barn rooms frequently visited by the farm personnel such as an office or an area where the animals are kept.

This way the herdsmen and farm managers can notice them without additional effort. In the later stage of the installation the farm personnel can define custom notifications.

### 7.1.2. Mobile and Home Access to the Monitoring System

In the near future the home is going to become the centre of people's activities [108]. People will work from home whenever possible. This trend will not avoid agriculture and its animal production sector and can be supported by the proposed cattle monitoring system. More precisely farm workers will be able to stay at home and perform the farming activities only whenever 'something happens' or 'needs to be done' such as in the case of detected oestrus or an animal disease. In this way they will have more free time and be available for the farm work outside the typical working hours. The latter decreases probability of missing an important event such as oestrus or an animal disease. The farm estates can include multiple barns and pastures which span large geographical areas. It is important that the farmer can maintain remote animals.

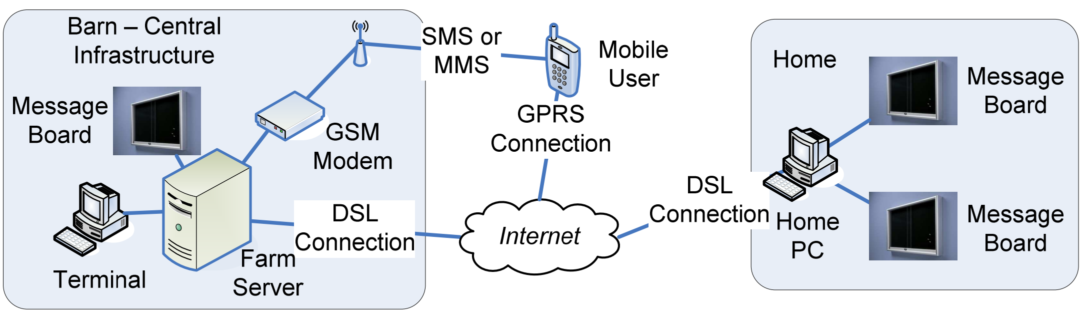


**Figure 7.1. Mobile and home access to the monitoring system**

In order to provide the environment for supervising the farm from home, the monitoring system has to be integrated with the home facilities to provide seamless access from home to the data collected and produced by the system and to efficiently deliver notifications. As shown in Figure 7.1 the data collected and produced by the system can be accessed from home PCs. The notifications can be displayed on a message board mounted inside the house or instant messaging software.

Another aspect of delivering notifications is handling the mobility of the farm personnel. In particular the notifications can be delivered to the farmer's mobile as text or multimedia messages (see Figure 7.1) so she can handle them while e.g. performing other obligations on the farm or interacting with business partners. Contrary to notifications displayed on the message boards at the farm or home the notifications delivered to the farmer's mobile should be more selective in order to make it less interfering with her current activities. The delivered content can be location dependant so that more notifications can be delivered when the farmer is at home or at the farm. The location can be established from a GPS receiver or the current GSM cell. The farmer can also query the data stored on the farm server using GPRS or UMTS.

## *7.2. Deployment of the Animal Mounted Devices*

### 7.2.1. Installation

Animal mounted devices proposed in Chapter 4 include a leg mounted pedometer and a collar. Both of them are custom made embedded systems. The pedometer comprises a battery, an energy efficient microcontroller (e.g. an Atmel picoPower microcontroller [109]), an inertial sensor (e.g. [110] from STMicroelectronics) and a wireless interface such as Bluetooth (e.g. [111] from STMicroelectronics) or 802.15.4 (e.g. [112] from Freescale Semiconductor). The collar comprises an energy efficient microcontroller (e.g. an Atmel picoPower microcontroller [109]), a battery, an inertial sensor (e.g. [110] from STMicroelectronics) and one or two wireless interfaces. One wireless interface is for communicating with the pedometer and is analogical to the one mounted in the pedometer. Another is for MANET communication with other collars, sinks and mobile users. This can be 802.11 (microcontrollers with 802.11b/g interfaces are available from Atheros, e.g. [113]) or 802.15.4 (e.g. [112] from Freescale Semiconductor). It may be possible to use one wireless interface for both these objectives.

Animal mounted devices are deployed during installation of the system and after a new animal is born or purchased. The minimal entered configuration data includes:

- the association between animal's ear tag id and hardware addresses of the devices mounted on the animal such as leg mounted pedometer and the collar
- unique enterprise id
- sex of the animal.

The optional data includes custom nicknames of an animal or even its photos. The enterprise id is important to prevent the animal mounted nodes from reporting the measurements to the system of a neighbour enterprise. The sex of the animal is necessary as it makes no sense to detect oestrus for bulls and steers. These parameters can be entered to the system during installation of the animal mounted devices from a handheld device, from a farm terminal or even over the Internet. In the two latter cases the necessary data is transmitted from the farm server to the animal mounted device via a sink.

It is possible that an animal mounted device is destroyed or stolen. In such case the device can be replaced by a new one and the system has only to be informed about the association between the hardware address of the new device and the ear tag id of the animal. The essential data required by the animal mounted device for detecting oestrus or animal diseases can be automatically downloaded from the server. This way the labour intensity of the maintenance is decreased.

### 7.2.2. Animals' Body Area Network

The important aspect of the envisaged system's deployment is connectivity between the leg mounted pedometer and the collar shown in Figure 4.4, i.e. body area network (BAN). BANs have been intensely utilized for the medical applications [114] and more recently for animal monitoring [1]. The most typically utilized communication platforms for this communication [114] is IEEE 802.15.4.

In order to save the battery of the leg mounted pedometer, which has smaller dimensions than the collar, it is necessary to use a different communication for the body area networking than for the communication between collars, sinks and users. The power of transmitters for body area networking should allow extending the battery life of the devices. However, if the transmitter power is too low, the absorption of RF signal by animal bodies can cause considerable packet loss in the case of certain body positions of the animals. The battery life of a body mounted sensor can range 5-6 days in the case of continuous activity to 20 years in the case of constantly staying in the sleeping mode [115]. The battery life is inversely proportional to the frequency of data exchanges between the pedometer and the animal mounted collar. For the purpose of walking intensity monitoring it is necessary to send the walking intensity from the pedometer to the collar several times a day. However as discussed in Section 5.3.2.1 data from the pedometer can be utilized to optimize communication between collars, sinks and mobile users. For this purpose walking intensity should be sent from the pedometer to the collar from every ten seconds to every one minute. This frequency is a compromise between the battery life of the pedometer and battery life of the collar. In particular more frequent sending data to the collar decreases battery life of the pedometer but potentially increases battery life of the collar.

## *7.3. Sink Deployment*

### 7.3.1. Installation

The sinks described in Chapter 4 are custom embedded systems comprising a microcontroller, a power source, a wireless interface for communicating with animal mounted nodes and another networking interface for communicating with the farm servers. It is possible that one interface serves both objectives. Sinks can be static or mobile (i.e. animal mounted). The animal mounted sinks have to be battery powered and require manual changing of the batteries, which is very labour intensive. However they can be convenient when animals are frequently moved between pastures. In this way the need for the fixed infrastructure or moving the static sinks is avoided.

Static sinks can be powered from mains or solar cells, which limits the labour intensity of their maintenance. Static sinks can be permanent or temporary. Temporary sinks are deployed only when the pasture or barn is in use and then moved to a different pasture. Temporary sinks are beneficial for the pastures which are used temporarily. They allow avoiding theft as well as tear and wear of the sinks when a pasture is not in use. They can also allow saving costs by decreasing the number of devices because they are required only on the subset of pastures which are currently in use. Permanent sinks are better for barns or pastures that are in constant use.

Deployment of a sink either permanent or temporary requires entering to the system an association between the sink's hardware address and the pasture or barn where it is located. Similarly as with the animal mounted devices this information can be entered form a handheld device, a farm terminal or over Internet. This association allows assessing efficiency of a pasture and identifying a pasture or barn where a given animal is located.

In the case of static sinks they should be optimally deployed in a location providing best coverage or frequently visited by all the animals. A place with the best coverage would be the middle of an area occupied by the animals – e.g. the middle of a pasture or the middle of a part of a barn where animals are kept. Places most frequently visited by animals are water tanks and feeders. Milking robots can be problematic if not all the animals kept in the particular area are wet (i.e. yielding milk).

Another important issue to be considered in placing static sinks is if they are going to be damaged or destroyed by the animals. This depends on the placement and housing of the sinks as well as temperament of the animals. It is better to install the sinks at places offering worse coverage but safer. They can be for example installed outside the fencing surrounding the area available to the animals.

If a static sink is connected to the farm server over wireless communication it is important to locate it in the place which is not a black spot. This refers to the local wireless communication and third party wireless communication services such as GSM or UMTS. If the data from sinks is delivered to the farm servers by data couriers, then the sinks should be installed close to the routes of these couriers e.g. in close proximity to a public road.

It is possible to deploy more than one sink per a single barn or pasture. This applies to both static and mobile sinks. The higher number of sinks increases the deployment costs but decreases the delays in delivering data from animal mounted nodes to farms servers. It is particularly beneficial in the case of larger pastures not divided into paddocks, where installation of multiple sinks increases probability of availability of a route to a sink. The higher number of sinks per a single paddock or pasture also increases reliability by providing redundancy useful when a subset of the deployed sinks is damaged. In such case a damaged sink should stop accepting data from animal mounted nodes to allow other sinks to take over this task.

### 7.3.2. Connecting Sinks to Farm Servers

There are multiple ways in which sinks can be connected to the farm servers including:

- Wired communication
- Local wireless communication
- Third party wireless communication services
- Delay Tolerant Networking
- No connection

Each of these options has its advantages and disadvantages in terms of installation and maintenance cost, delays as well as feasibility. These are discussed in the greater detail below and summarised in Table 7.1

**Table 7.1. Connecting sinks to farm servers**

| Type | Installation Costs | Maintenance Costs | Reliability | Potential Problems | Applicability |
|---|---|---|---|---|---|
| Wired Comm. | High | Low, high in the case of failure | High | High costs of installation and repair | Barns and pastures close to barns |
| Local Wireless Comm. | Low | Medium | Medium | Interference with other systems, limited range | Barns and pastures close to barns |
| Third Party Comm. Services | Low | High | Depending on location | High maintenance costs, limited coverage | Pastures far from farm building |
| Delay Tolerant Networking | Low | Low | Low | Long delays and low reliability | Pastures far from farm buildings |
| No Connection | None | None | Low | Limited functionality | Pastures far from farm buildings |

## 7.3.2.1. Wired Communication

Wired communication such as Ethernet offers best reliability. More precisely it is not affected by interference from other devices and its continuous functionality does not require administration tasks such as changing utilized wireless channels. The prices of necessary hardware are low. However, its installation is very labour intensive and requires skilled technicians, which increases installation costs. A possible failure such as cable damage is difficult to repair and requires appropriate equipment and a skilled technician. Wired network connections are much easier to install indoors than outdoors. The existing infrastructures such as cables, hubs etc. can be also utilized. Therefore the wired connection can be attractive in the case of sinks mounted indoors or at the pastures placed in the close proximity to barns.

#### 7.3.2.2. Local Wireless Communication

Local wireless communication offers easy and cheap installation without need for skilled technicians. The probability of a persistent failure such as hardware damage is low. However the hardware is typically more expensive than in the case of wired network. The important challenge is the relatively short range. In particular the off-the-shelf IEEE 802.11b/g hardware typically offers around 100m range in open space [116]. This can be insufficient in the agricultural conditions but can be leveraged using repeaters. The interferences between different systems using the same wireless technology can be also a problem. Therefore the local wireless communication is optimal for the sinks mounted indoors or in the pastures located within the limited distance from the farm buildings. It is particularly attractive when the farm estates occupy a limited compact area.

#### 7.3.2.3. Third Party Communication Services

Third party communication services such as GSM or UMTS are cheap to deploy. The installation does not require any skills and hardware is typically cheap. Theoretically they can offer very good coverage. In practice this coverage may not be as good in the agricultural estates as in cities or villages [1] because agricultural areas are typically sparsely populated and thus less attractive for the mobile networks operators. Other problems are the operator's charges. In the best case it is a monthly fixed fee but the operator can charge per unit of transmitted data or connection time. This can have a major impact on the maintenance costs of the whole system. Such communication requires also additional administration effort for managing the cooperation with the network operator, accounting the charges etc. Change of the operator's ownership and pricing policy is a considerable risk. The prices of the service and the model of charging for the service can change even with the short notice.

To summarize, the utilization of third party communication services is expensive in maintenance and relatively risky as an investment but in certain circumstances it is difficult to avoid. This includes connecting remote pastures to the farm serves, particularly in the case when the farm buildings are separated from the monitored pasture by the properties not belonging to the enterprise using the system.

#### 7.3.2.4. Delay Tolerant Networking

In the Delay Tolerant Networking (DTN) approaches (e.g. [85]) humans or vehicles are utilized to carry data from one place to another. In the case of automated cattle production it is typically not profitable to use humans or vehicles for the sole purpose of moving data. The labour costs would be most likely higher than the installation of the network infrastructure or utilization of the third party communication services. However the financially feasible application of this approach is utilization of the existing movement of people or vehicles. For example the data could be automatically collected and uploaded by devices carried by people who commute to work, children going to school etc. The major advantage of this scenario is low cost of hardware and maintenance. The major risk here is the limited reliability introduced by the human factor. For instance the massage carrier may not need or be unable to make her daily route on particular occasions such as a holiday or an illness. This problem could be overcome by redundancy but it may not always be possible due to the limited number of people or vehicles available for utilization. Other disadvantages of DTNs are increased delays, which may decrease the efficiency of the system for example by giving the farm personnel not enough time to handle the detected oestrus. To summarize, DTNs offer a cheap alternative to the constant wireless or wired networking connectivity but this approach is not always available and has a risk of limited reliability and longer delays.

#### 7.3.2.5. No Connection

In certain cases it may be impossible to connect sinks to farm servers and thus the deployment of the sinks may be infeasible. For example, when a pasture is far away from farm buildings, the coverage of third party communication services is not available or they are too expensive and there is no human mobility to utilize. In such cases it is possible to completely avoid deploying sinks. This severely limits the functionality of the envisaged system. In particular the following functionality is not available:

- safe long term storage,
- delivery of notifications to mobile phones or message boards,
- querying collected data from terminals in the office or home,
- identifying a pasture where a particular animal is kept.

## *7.4. Maintenance*

This section discusses maintenance activities necessary for continuous functionality of the proposed cattle monitoring system. This includes the following activities.

### 7.4.1. Maintenance of Animal Mounted Nodes

The farm personnel have to replace the discharged batteries of the animal mounted nodes and replace the lost or broken animal mounted devices. The animal mounted devices must be installed on newly born or purchased animals and configured as descried in Section 7.2.1.

### 7.4.2. Management of Sinks

After the initial deployment, the sinks have to be managed on everyday basis. The farm personnel has to remove and install temporary stationary sinks, replace batteries in animal mounted sinks and also replace any broken sinks. In order to maintain the required quality of service provided by the sinks the quality of service has to be automatically monitored and its potential deterioration has to be indicated to the users. They have to react by appropriate corrective action such as deploying additional sinks.

### 7.4.3. Management of the Connection between Sinks and Farm Servers

In the case of all types of connections between sinks and farm servers discussed in Section 7.3.2 the farm personnel has to occasionally replace broken devices. Some types of connections require more frequent maintenance activities. In the case of wired connections the network cable can be damaged, which requires a corrective action performed by a skilled technician. In the case of local wireless communication it is necessary to periodically reassign channels used by wireless devices to solve the interference problems. This is a relatively easy task that can be performed by stockmen. In the case of data couriers it is necessary to monitor if they are frequently enough moving along the specified routes carrying the appropriate devices. If due to changing circumstances couriers stop providing their service it is necessary to appoint different couriers or utilize different means of connecting sinks to farm servers.

### 7.4.4. Backups

The configuration and historical data stored on the farm servers has to be regularly backed up in order to safeguard against potential hardware failure. This task can be performed automatically but the farm personnel have to make sure that sufficient storage media are continuously available. It is a non challenging task, which can be performed by the farm personnel with minimal technical knowledge.

### 7.4.5. Management of User Accounts

For the security reasons discussed in Section 7.6 users who are allowed to query the data produced by the system have to be associated with the accounts defined in the system's directory. In order to access the system users have to provide credentials such as the username and password. In order to maintain this functionality users' accounts have to be managed on continuous basis in order to remove the accounts of former employees and create accounts for new ones. This task requires assigning a person that would be responsible for it. Due to the nature of this task no particular technical skills are required.

## *7.5. Interoperability*

There are following aspects of the interoperability of the proposed cattle monitoring system:

- Interoperability between components of the system
- Interoperability between other systems used on the farming enterprise

In the following subsections we will look closer at these aspects.

### 7.5.1. Interoperability between the Components of the System

If all the components of the system are to be manufactured by the same institution this aspect of interoperability is trivial. However if the components can be manufactured by different institutions and several institutions can manufacture the same class of components then defining communication interfaces between components becomes crucial. Similar approach was successfully utilized in the GSM network [117].

### 7.5.2. Interoperability between Different Systems Used by an Enterprise

The envisaged system can be integrated with other systems used by the enterprise. For example the reliability of detecting oestrus and animal diseases can be improved by integrating the proposed system with a system monitoring milk yield and milk parameters [8, 10]. However, providing seamless integration can be challenging. It can be provided by exposing the clearly defined interfaces to the existing systems during the design of the proposed cattle monitoring system. Alternatively a mechanism of the extensions (*plugins*) can be provided to allow integration with the newly developed systems. Another approach is to provide generic clearly defined interfaces to allow developers of other systems to provide integration with the proposed system.

## *7.6. Security Considerations*

### 7.6.1. Overview

The security considerations are outside the main focus of this thesis but in this section we present a brief overview of the security issues related to the target cattle monitoring systems in order to consider possible security problems and propose a cost efficient security model for the target system. The major envisaged attacks on the target system are:

- illegitimate access to the data collected or processed by the system,
- spoofing of the data collected and processed by the system.

The possible targets of an attack include:

- sending data from the pedometer to the collar,
- sending data from animal mounted collars to farm servers,
- sending data from farm servers to animal mounted nodes,
- in-situ queries,
- access to data stored on the farm server and delivery of notifications.

In the following subsections these targets are discussed in the greater detail including the potential threats and precautions. These precautions are summarized in Table 7.2.

**Table 7.2. Security precautions**

| Communication Type | Possible Precautions | Cost of Increasing Security |
| --- | --- | --- |
| Sending Data from the Pedometer to the Collar | Signing and encrypting transmitted data using public-key cryptography | Need for more powerful animal mounted devices (pedometers and collars), more complex installation process |
| Sending Data from Animal Mounted Collars to Farm Servers | Signing and encrypting transmitted data using public-key cryptography | Need for more powerful animal mounted collars, more complex installation process, not possible to cache data sent to farm servers for answering in-situ queries |
| Sending Data from Farm Servers to Animal Mounted Nodes | Signing transmitted data using public-key cryptography | Need for more powerful animal mounted devices |
| In-situ Queries | Signing queries, signing and encrypting replies using public-key cryptography, public key infrastructure issuing user certificates | Need for more powerful animal mounted devices (collars) and user devices, more complex installation process |
| Accessing Data Stored on the Farm Servers | SSL or TLS, verifying users' credentials | Need for more powerful servers |
| Delivery of Notifications | Signing notifications sent to message boards using public-key cryptography, checking phone numbers from which picture or text messages are sent | More complex message boards, more complex software or usage of mobile devices |

### 7.6.2. Sending Data from the Pedometer to the Collar

Walking intensity data is sent from the pedometer to the animal mounted collar using short-range radio frequency. This transmission can be intercepted or spoofed by an attacker. The transmission range is short but the attacker can use a unidirectional antenna to intercept it or replace the pedometer with a spoofed one, which produces bogus data.

The possible precaution would be to use the public-key cryptography [118] to sign and encrypt the transmitted data. The amount of data is small which makes it feasible from the aspect of computational intensity. The generation and exchange of public keys between the collar and the pedometer could take place during installation triggered in a way which is not accessible to the attacker e.g. by the user connecting to the collar from a handheld device and providing administrative credentials.

### 7.6.3. Sending Data from Animal Mounted Collars to Farm Servers

Data can be intercepted or spoofed by the attacker when it is transmitted from the collar to the farm server. This is much more likely than in the previous case because the data is more meaningful so potentially more valuable to the attacker. The transmission range is longer including transmission from animal mounted devices to sinks and from sinks to a farm server, so there are more opportunities for spoofing or interception.

The attacker can overhear the transmission from animal mounted devices to a sink or even pretend to be an animal mounted device and forward the data intercepting or even changing it. The attacker can also send the spoofed data pretending to be one of the existing animal mounted devices. The data can be also spoofed or intercepted on the way from a sink to a farm server.

Here the feasible precaution is also the public-key cryptography [118]. The animal mounted collars sign and encrypt data sent to a farm server. The amounts of data are small so the computational intensity is feasible. Private and public keys can be generated during the installation. At this point also the exchange of the public keys can also take place. Then the farm server knows the public keys of each animal mounted collar and each collar knows the public key of the server. The only problem here is that if the public-key encryption is utilized it is no longer possible to cache the data sent to farm servers for the purpose of answering in-situ queries. In this way there is a trade-off between security and reliability in answering in-situ queries.

### 7.6.4. Sending Data from Farm Servers to Animal Mounted Nodes

Sporadically, as envisaged in Section 5.5 farm servers may need to send data to animal mounted nodes. Data is send via sinks to the animal mounted collars, which then can forward it on to the pedometers. This data can include firmware and configuration updates. A potential attacker can send spoofed configuration or firmware updates to disable or intercept animal mounted devices.

The possible precaution would be to use public-key cryptography [118]. In particular servers sing the data sent to animal mounted nodes, which then can evaluate authenticity of this data. Each animal mounted node would have to be informed about the farm servers' public keys during the installation process.

### 7.6.5. In-situ Queries

In the case of in-situ queries an attacker can access the data stored by the animal mounted devices or on the farm server (via a sink). A malicious devices provided by the attacker can answer queries issued by the user with bogus data. Finally, the attacker can overhear the queries issued by the user and the answers to them.

Similarly as in previous cases the public-key cryptography [118] can be used to address these threats. The user signs her queries. The animal mounted nodes and the sink check if the user has rights to query the data. The answer is then signed and encrypted by the animal mounted devices answering the query.

The exchange of the public keys is not as straightforward as in the previous cases. The user can download from the farm server the public keys to all the animal mounted nodes but the animal mounted nodes may not know the public keys of all user devices. In particular a subset of user devices may be purchased after the installation of the animal mounted nodes so it is not enough to store the public keys of the user devices at the animal mounted nodes during their installation. Animal mounted nodes can obtain these keys from farm servers via sinks but this is unreliable as the multi-hop path between the animal mounted nodes and sinks does not have to be always available. A better approach is to utilize the Public Key Infrastructure (PKI). In particular there is a certification authority (e.g. a farm server) with the public key known to the animal mounted nodes, which they acquire during the installation. Then the certification authority is used to issue the user certificates that verify users' public keys and their right to access data from the monitoring system. Such certificate is attached to in-situ queries issued by a user. These certificates should have limited validity period in order to limit necessity of revoking certificates of compromised user devices. In this way the animal mounted nodes do not have to store long lists of revoked certificates.

### 7.6.6. Accessing Data Stored on the Farm Servers and Delivery of Notifications

In the case of accessing data stored on the farm server an attacker can obtain unauthorised access to the data or pretend to be the server to mislead a user. An attacker can also issue fake notifications.

The connection between the user and the server can be secured using SSL or TLS. In this way the identity of the server can be verified and interception of the information exchange between a user and the server prevented. User identity can be verified simply by username and password authentication.

Prevention of issuing fake notifications depends on the method of notification delivery. In the case of emails or message boards the public-key cryptography [118] can be used to sign the notifications. In the case of GSM text messages it may be sufficient to verify the number from which the message was sent.

# 8. Conclusions

This chapter is divided into four sections. Section 8.1 summarises the work presented in this thesis. Section 8.2 presents the main conclusions and contributions of this work. Section 8.3 discusses other applications of the proposed MANET routing algorithm. Section 8.4 presents the promising future directions of the presented work.

## *8.1. Summary*

This thesis is concerned with the design and realisation of a large scale adaptive cattle monitoring system that supports data retention, detecting custom events, issuing notifications, answering remote and in-situ queries.

Chapter 1 motivated the work presented within this thesis by introducing the problem of scalable cattle monitoring. Currently cattle production is an important and labour intensive sector of agriculture. Automated monitoring of cattle can increase efficiency of cattle production, decrease its reliance on human labour and thus increase its profitability. Prompt detection of animal diseases such as foot-and-mouth disease (FMD) and bovine spongiform encephalopathy (BSE) can prevent their spreading and thus limit their devastating impact on economy and human health. High ratio of detecting oestrus can increase profitability of cattle farming [4, 6] and improve quality of farmer's life [6].

The current sensing systems for monitoring cattle rely on monitoring milk parameters, animal identification as well as measuring walking and mounting intensity [6-10]. Animal identification is performed by utilization of RFIDs that are read over a short distance and only when animal's speed is limited [9]. This seriously affects reliability of the cattle monitoring process. The pedometers are usually read at the milking robots which limits their scope only to monitoring wet dairy cows [10]. Monitoring of milking parameters is characterised by a similar limitation. The systems that are already commercialized often rely on monitoring of a single behavioural parameter (e.g. [6]). Current Wireless Sensor Networks (WSN) approaches for monitoring cattle behaviour and metabolism are largely pragmatic proof of concepts [11]. More precisely they use GSM telephony for most of their wireless communication [1], which is expensive and decreases financial feasibility of the target system or have only single-hop communication [6]. Multi-hop communication is much more appropriate for this application. When animals are kept on pastures multi-hop communication allows animal mounted devices with shorter transmission range and fewer sinks offering lower vulnerability to disconnections. Limiting disconnections improves reliability of the monitoring process and thus increases probability of detecting animal diseases such as FMD or BSE as well as oestrus. Animal mounted devices with shorter transmission range are either smaller or characterised by longer battery life. This way the labour intensity of maintenance is limited making cattle monitoring more financially feasible. When animals are kept in a barn multi-hop communication allows circumventing the obstacles in radio waves propagation and combat the effect of animal bodies absorbing the radio waves [2, 12], which reduces disconnections and thus increases reliability. WSN multi-hop ad hoc routing algorithms [13-15] typically address mostly static nodes. Routing algorithms for Mobile Ad Hoc Networks (MANETs) consider topologies comprising mostly mobile nodes [16] so they are better suited for this application than the typical WSN approaches. There is a number of existing MANET approaches (e.g. [17-29]) but they are typically theoretical and driven by simulations utilizing random movement patterns of the mobile nodes. In contrast, the work in this thesis focuses on realistic requirements and emulation based evaluation utilizing data from the field experiments.

Chapter 2 gave an overview of the existing work related to the content of this thesis. Firstly a series of criteria were chosen, described and motivated. These criteria were important for designing the scalable cattle monitoring system, in particular its wireless communication infrastructure. They were divided into satisfying user requirements and addressing environmental constrains. Satisfying user requirements included increasing reliability, managing delays, increasing scalability and lowering costs. Addressing environmental constrains included handling high mobility and addressing lack of mobility supporting communication. Then the relevant work in animal monitoring, WSNs, MANETs and DTNs were reviewed according to these criteria. We showed that there was no single approach addressing sending data from animal mounted nodes to sinks and answering in-situ queries. We identified the most optimal existing DTN approach for delivering data from sinks to farm servers by data couriers. We also reviewed the existing work related to the design and management of wireless monitoring systems.

Chapter 3 described field experiments performed to gather the realistic requirements for the design of a scalable cattle monitoring system. They comprised qualitative and quantitative parts. Qualitative part constituted of the questionnaire distributed to the farm personnel. It showed which functionality of the system was most essential and which were the tolerable delays in reporting each type of detected events. Quantitative part constituted of tracking the movement and behaviour of the animals. It showed the typical walking speeds of the cattle and considerable differences in preferred walking speed among animals. It also showed patterns of their day and night activity. Quantitative experiments produced data utilized for the evaluation of the proposed MANET routing algorithm.

Chapter 4 proposed the novel delay tolerant architecture that provided data retention, detecting custom events, issuing notifications, answering remote and in-situ queries. It offered high scalability and limited the costs of installation and maintenance. This was achieved by utilization of infrastructure as well as ad hoc, opportunistic networking and Delay Tolerant Networking. The proposed architecture combined peer-to-peer and client-server paradigms in a unique way. In this chapter we also defined input and output of the cattle monitoring system and discussed the installation, usage and maintenance scenarios of this system.

Chapter 5 proposed a novel practical energy efficient MANET routing algorithm that supported the architecture proposed in Chapter 4 and was based on requirements defined in Chapters 2 and 3. This algorithm provided offloading data for long term storage by sending data to the farm servers via sinks being a part of MANET and answering in-situ queries issued by users collocated with the animals. It addressed high mobility of nodes and disconnections, which was crucial for improving reliability of the animal monitoring process. The proposed algorithm utilized in a novel way heterogeneity of nodes' mobility. It also utilized the low intensity of data traffic to save energy on route discovery for the cost of energy efficiency of data traffic. This was achieved by a novel combination of the energy efficient unicast MANET routing algorithm ESDSR [27] with the MANET broadcasting PCDI [31]. The disconnections were addressed using the *cooperative detection of route availability*.

Chapter 6 evaluated the practical energy efficient MANET routing algorithm proposed in Chapter 5. The evaluation comprised emulation that utilized animal movement patterns collected over the course of our field experiments described in Chapter 3. In particular these movement patterns were utilised to realistically emulate movement of a larger number of animals. Movement patterns from this emulation were then utilised for the packet level emulation of the proposed algorithm. We implemented the proposed algorithm, a classical MANET routing algorithm DSR [18] and an existing energy efficient MANET routing algorithm ESDSR [27] as wireless routing agents [106] for ns-2 [105] which we used for the packet level emulation. We demonstrated that the proposed MANET routing algorithm provided lower and more balanced energy usage, shorter delays of answering in-situ queries as well as increased success ratio of delivering answers to in-situ queries than the classic, non-energy aware DSR [18] and the more generic existing energy aware routing algorithm ESDSR [27]. This was important for increasing battery life of animal mounted nodes and thus decreasing maintenance costs of animal monitoring. The most important impact on saving and balancing energy had utilization of PCDI [31] to optimize broadcasting of route discovery packets and in-situ queries. PCDI allowed achieving almost constant energy utilization regardless of the number of nodes. PCDI also reduced the delays in the case of in-situ queries and improved success ratio of sending replies to in-situ queries. More precisely PCDI was responsible for lower deterioration of success ratio and delays with the increasing number of nodes thus improving the scalability. Finally we showed how the proposed architecture and routing algorithm addressed the criteria defined in Chapter 2.

Chapter 7 discussed the practical management issues of the envisaged cattle monitoring system considering the criteria defined in Chapter 2, in particular user requirements including increasing reliability, managing delays, increasing scalability and lowering costs. We discussed necessary configuration performed during the system installation and providing mobile and home access to the target system. We proposed strategies for sinks installation, connecting sinks to farm servers and improving security of the system. We also discussed necessary maintenance activities and interoperability issues.

## *8.2. Contributions*

This section draws out the main contributions of this thesis:

- This work identifies realistic requirements for the cattle monitoring system that extend our understanding of the role, incentives and challenges of agricultural monitoring. These requirements are based on the quantitative and qualitative field experiments we performed as well as the relevant farming and computer science literature. An early version of these requirements was published in [97] and [98].

- The cattle movement data from the experiments we performed has been submitted to the Community Resource for Archiving Wireless Data At Dartmouth (CRAWDAD) [119]. CROWDAD is an international repository of real wireless data for wireless network research community. Our data will become publicly available when our publication describing field experiments described in this thesis is accepted.

- This work proposes a novel application of the energy efficient MANET routing to combat disconnections, limit and balance energy utilization. This thesis concentrates on the cattle monitoring system but the proposed approach can be adapted to any scenario where the wireless monitoring devices are mounted on the monitored animals or people. A part of this was published in [98].

- This work formally defines input and output of the proposed cattle monitoring system. An early version of these analysis were published in [98].

- This work presents a novel realistic architecture for the cattle monitoring system. It combines the MANET wireless, wired and DTN communication. Such architecture has a potential to increase profitability of agriculture exceeding the boundaries of its cattle production sector to for example monitoring other types of animals. This architecture was published in [97].

- This work extends the concept of energy efficient routing by application of Passive Clustering with Delayed Intelligence (PCDI) [31] to route discovery. In this way the energy utilization for route discovery is minimized and balanced for the cost of energy efficiency of data traffic. This approach is also suitable to other MANET applications with tight energy constrains and limited data traffic such as other monitoring applications.

- This work extends the PCDI concept by utilizing the heterogeneity of mobile nodes' mobility. In this way data is more likely to be forwarded by the nodes moving with lower speeds, which form more reliable routes. This technique limits the delays of sending data to sinks. This approach demonstrates how the PCDI concept can be adapted to utilize application specific characteristics of the environment such as heterogeneity among the mobile nodes.

- This work proposes a novel technique for handling disconnections - *cooperative detection of route availability*. This technique minimizes and balances energy spent on detection of availability of a multi-hop path to a sink. It can be used for any wireless multi-hop application with temporary disconnections, in particular when mobility of the nodes does not support connectivity.

- This work presents a novel MANET routing algorithm that provides lower and more balanced energy usage, shorter delays of answering in-situ queries as well as increased success ratio of delivering answers to in-situ queries than the existing more generic approaches. It is suitable for cattle monitoring but can be adapted to any other monitoring application, where nodes are mounted on monitored people or animals.

- This work presents the detailed evaluation of the proposed MANET routing algorithm and general feasibility of applying a MANET routing to animal monitoring. It involves detailed emulation utilizing data from the field experiments we performed. This evaluation demonstrates which techniques utilised in the proposed routing algorithm are most beneficial for cattle monitoring.

- In this thesis we discuss in detail management of the large scale deployment of the envisaged cattle monitoring system. We discuss necessary configuration performed during the system installation and providing mobile and home access to the target system. We propose strategies for cost efficient sinks installation and connecting sinks to farm servers as well as a cost efficient security model for the target system based on public key cryptography. In this way we demonstrate that the cost effective and secure deployment of the system is feasible in various realistic conditions.

## 8.3. Other Applications of Energy Efficient Route Discovery

The MANET routing algorithm proposed within this thesis has potentially more applications than cattle monitoring. It can be used for monitoring applications with the similarly limited data traffic, where the sensors are carried by sufficiently large mobile agents having the speed of similar order of magnitude as cattle and moving on a constrained area. This includes other types of animals (e.g. sheep, goats or horses) or people. In the case of people the potential applications include monitoring health parameters of sportsmen [120], hospital patients or elderly and disabled people in general [121]. Health monitoring of sportsmen is promising because they operate in extreme conditions, which can seriously affect their health e.g. overheating [120]. Monitoring elderly people is important due to aging of the population. Automating healthcare of elderly people can potentially keep its costs at a feasible level.

## 8.4. Future Work

The results from the experiments presented in this thesis are encouraging, validating the efficiency of the proposed routing algorithm in terms of energy utilization, delays and success ratio. However there are still issues that have not been fully addressed. They are identified below.

Some of the conclusions drawn in the Chapter 3 concerning the mobility patterns of the cows were based only on two animals. More precisely this includes the conclusion concerning the preferred walking speeds, walking intensity over the day and locations over the day. This should be validated by monitoring mobility of a larger number of animals not only in the dairy environment but also outdoors. The behaviour of the animals in the pastures is different than in the dairy, where the field experiments where performed [12, 100]. For example the density of the topology in such circumstances is going to be lower [100].

The proposed MANET routing protocol EERD, can utilize multicast to optimise sending the same data from sinks to multiple animal mounted nodes.

The efficiency of the proposed MANET routing algorithm can be further validated by the real world large scale deployment of the devices utilising this algorithm. This will allow considering some parameters that were not considered during emulation such as absorption of radio frequency waves by animal and human bodies [2, 12] or propagation of radio waves in relation to position of the animal and its collar.

The movement patterns used for simulation were based on real data and observations and thus are close to reality. We tested the correctness of the protocol implementation by analyzing the simulation traces. The potential weak point of our simulation is the validity of the utilized radio propagation model (i.e. two-ray ground reflection model from ns-2 [122]) for the dairy environment. In particular all simulation models make simplifying assumptions about radio propagation, which do not have to apply for all types of environment [123]. The typical method of validating these assumptions and simulation models in general for the given type of environment is comparing the parameters of radio propagation form the model with the measurements from the real environment [123, 124]. Such measurements have to be performed to validate our simulation.

The proposed architecture can be extended by utilizing more animal mounted sensors. The other relevant sensors include a thermometer mounted inside the animal's rumen [1] and a pressure sensor mounted on the animal's rump [6] to detect mounting activity, which is related to oestrus.

Another promising direction for the future work is testing and adapting the proposed MANET routing algorithm for other application described in Section 8.3. Such evaluation is necessary as the other animals and people have different movement patterns than cattle, occupy different types of areas and form different densities of topologies. The amounts of collected data can also be different.

# Appendix 1: Questionnaire

Imagine an automatic cattle monitoring system. Its objective is to detect

- oestrus,
- pregnancy,
- animal diseases such as mastitis, lameness, BSE, FMD etc.,
- reduced efficiency of a pasture (meaning cattle located there do not graze for example because of lack of foliage).

You can access the collected data in all of the following ways:

- from home over the Internet,
- from the PC in the farm office,
- at the area where the animals are kept (a part of dairy or a pasture) using a mobile phone or a palmtop you carry.

It is also possible to set up notifications about newly detected events. They can be delivered to your mobile phone as text messages or displayed on a message board located at a convenient location within the farm.

Could you rate the usefulness of the following features of such system? Please circle the relevant number.

Detecting oestrus:
Useless 1 2 3 4 5 Very useful

Detecting pregnancy:
Useless 1 2 3 4 5 Very useful

Detecting animal diseases:
Useless 1 2 3 4 5 Very useful

Detecting reduced efficiency of pastures:
Useless 1 2 3 4 5 Very useful

Querying data at the area where the animals are kept:
Useless 1 2 3 4 5 Very useful

Notifications about newly detected events delivered to your mobile phone or displayed on a message board:
Useless 1 2 3 4 5 Very useful

Receiving these notifications (delivered to a mobile or a palmtop you carry) only at the moment when you enter the area occupied by the affected animals:
Useless 1 2 3 4 5 Very useful

How often would you do the following activities? Please mark one option of each activity.

Query the system from home or office:

1) Never
2) Once a month
3) Once a week
4) Several times a week
5) Once a day
6) Several times a day

Query the system at the area where the animals are kept:

1) Never
2) Once a month
3) Once a week
4) Several times a week
5) Once a day
6) Several times a day

How immediately would it be necessary to be informed about the following events (either by notifications or from querying the system)? Please select one option for each of the following:

Oestrus:

1) As soon as possible
2) Within an hour
3) Within 4 hours
4) Within 24 hours
5) Within 48 hours

Animal disease:

1) As soon as possible
2) Within an hour
3) Within 4 hours
4) Within 24 hours
5) Within 48 hours

Pregnancy:

1) As soon as possible
2) Within an hour
3) Within 4 hours
4) Within 24 hours
5) Within 48 hours

Reduced efficiency of a pasture:

1) As soon as possible
2) Within an hour
3) Within 4 hours
4) Within 24 hours
5) Within 48 hours

Which other animal related events you think it would be useful to detect (if any):

…………………………………………………………………………………………
…………………………………………………………………………………………
…………………………………………………………………………………………
…………………………………………………………………………………………

Your position at the Dairy Centre:

…………………………………………………………………………………………

In general your duties at the Dairy Centre:

…………………………………………………………………………………………
…………………………………………………………………………………………

**Thank you very much for your participation!**

# References

[1] K. Mayer, K. Ellis, and K. Taylor, "Cattle Health Monitoring Using Wireless Sensor Networks", In *Proc. of CCN*, Cambridge, Massachusetts, USA, 2004.

[2] T. Wark, et al., "Transforming Agriculture through Pervasive Wireless Sensor Networks", *Pervasive Computing, IEEE*, April-June 2007, 6 (2), pp. 50-57.

[3] D. P. Davis, "Emerging animal diseases: global markets, global safety : a workshop summary", the National Academies' Board on Agriculture and Natural Resources of the Division on Earth and Life Studies, 2002.

[4] J. L. Albright and C. W. Arave, *The Behaviour of Cattle*, CAB INTERNATIONAL, 1997.

[5] B. Lowman, "Triple synchronisation of suckler cows", *In Practice*, 31 January 2003, 25 (1), pp. 38-43(6).

[6] "HeatWatch® Estrus Detection System", CowChips, LLC, Available: http://www.heatwatch.com/.

[7] A. Frost, et al., "A review of livestock monitoring and the need for integrated systems", *Computers and Electronics in Agriculture*, 1997, 17, pp. 139-159.

[8] R. Firk, E. Stamer, W. Junge, and J. Krieter, "Automation of oestrus detection in dairy cows: a review", *Livestock Production Science*, July 2002, 75 (3), pp. 219-232(14).

[9] W. J. Eradus and M. B. Jansen, "Animal identification and monitoring", *Computers and Electronics in Agriculture*, 1999, 24, pp. 91–98.

[10] K. Maatje, R. M. d. Mol, and W. Rossing, "Cow status monitoring (health and oestrus) using detection sensors", *Computers and Electronics in Agriculture*, 1997, 16 (3), pp. 245-254(10).

[11] A. Chaintreau, et al., "Pocket Switched Networks: Real-world mobility and its consequences for opportunistic forwarding", Computer Laboratory, University of Cambridge, Cambridge, UK, Technical Report 617, 2005.

[12] Z. Butler, P. Corke, R. Peterson, and D. Rus, "Dynamic Virtual fences for Controlling Cows", In *Proc. of ISER*, Marina Mandarin Hotel, Singapore, 2004.

[13] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed Diffusion for Wireless Sensor Networking", *IEEE/ACM Transactions on Networking*, 2003.

[14] S. Ratnasamy, et al., "Data-Centric Storage in Sensornets", In *Proc. of SIGCOMM*, 2002.

[15] S. Ratnasamy, et al., "GHT: A Geographic Hash Table for Data-Centric Storage", In *Proc. of WSNA*, 2002.

[16] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc mobile wirelessnetworks", *IEEE Personal Communications*, 1999, 6 (2), pp. 46-55.

[17] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", In *Proc. of SIGCOMM*, 1994, pp. 234–244.

[18] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks": Kluwer Academic, 1996.

[19] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing", In *Proc. of WMCSA*, 1999.

[20] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", In *Proc. of INFOCOM*, 1997, pp. 1405-1413.

[21] S. Singh and C. S. Raghavendra, "PAMAS - Power Aware Multi-Access protocol with Signalling for Ad Hoc Networks", *ACM SIGCOMM Computer Communication Review*, 1998, 28 (3), pp. 5-26.

[22] S. Singh, M. Woo, and C. S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks", In *Proc. of MobiCom*, Dallas, TX, USA, 1998, pp. 5-26.

[23] C.-K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks", *IEEE Communications Magazine*, June 2001.

[24] D. Kim, J. J. Garcia-Luna-Aceves, K. Obraczka, J.-C. Cano, and P. Manzoni, "Power-Aware Routing Based on The Energy Drain Rate for Mobile Ad Hoc Networks", In *Proc. of ICCCN*, 2002.

[25] M. Maleki, K. Dantu, and M. Pedram, "Power-aware Source Routing Protocol for Mobile Ad Hoc Networks", In *Proc. of ISLPED*, Monterey, California, USA, 2002.

[26] M. Maleki, K. Dantu, and M. Pedram, "Lifetime Prediction Routing in Mobile Ad Hoc Networks", In *Proc. of IEEE Wireless Communications and Networking (WCNC)*, 2003, pp. 1185-1190.

[27] M. Tarique, K. E. Tepe, and M. Naserian, "Energy Saving Dynamic Source Routing for Ad HocWireless Networks", In *Proc. of WIOPT*, 2005.

[28] P. Bergamo, et al., "Distributed Power Control for Energy Efficient Routing in Ad Hoc Networks", *Wireless Networks*, 2004, 10 (1), pp. 1572-8196.

[29] Y. Xu, J. Heidemann, and D. Estrin, "Adaptive Energy-Conserving Routing for Multihop Ad Hoc Networks", USC/ISI, Research Report 527, 12 October 2000.

[30] Y. I. Jerschow, B. Scheuermann, C. Lochert, and M. Mauve, "A Real-World Framework to Evaluate Cross-Layer Protocols for Wireless Multihop Networks", In *Proc. of RealMAN*, Florence, Italy, 2006.

[31] M. R. Gosnell, R. Albarelli, M. X. Cheng, and B. McMillin, "Energy Balanced Broadcasting Through Delayed Intelligence", In *Proc. of ITCC*, 2005, pp. 627-632.

[32] E. J. Riedy and R. Szewczyk, "Power and Control in Networked Sensors", 2000.

[33] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks", In *Proc. of ICDCS*, 2002, pp. 575-578.

[34] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler, "Lessons from a Sensor Network Expedition", In *Proc. of EWSN*, 2004.

[35] R. E. Kenward, *A Manual for Wildlife Radio Tagging*, Academic Press, 2001.

[36] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, "Hardware Design Experiences in ZebraNet", In *Proc. of SenSys*, Baltimore, Maryland, USA, 2004.

[37] P. Corke and P. Sikka, "Results from the Farm", In *Proc. of EmNets*, Harvard University, Cambridge, MA, USA, 2006.

[38] T. Wark, et al., "The design and evaluation of a mobile sensor/actuator network for autonomous animal control", In *Proc. of International Conference on Information Processing In Sensor Networks (IPSN)*, Cambridge, Massachusetts, USA, 2007, pp. 206-215.

[39] T. Small and Z. J. Haas, "The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)", In *Proc. of MobiHoc*, Annapolis, Maryland, USA, 2003.

[40] T. Small, Z. J. Haas, A. Purgue, and K. Fristrup, "A Sensor Network for Biological Data Acquisition", *Handbook on Sensor Networks*, 2004.

[41] Z. J. Hass and T. Small, "A New Networking Model for Biological Applications of Ad Hoc Sensor Networks".

[42] A. L. Iacono and C. Rose, "Infostations: New Perspectives on Wireless Data Network", in *Next Generation Wireless Networks*, vol. 598: Springer Netherlands, 2002, pp. 3-63.

[43] D. J. Goodman, J. Borras, N. B. Mandayam, and R. D. Yates, "INFOSTATIONS: a new system model for data and messaging services", In *Proc. of Vehicular Technology Conference*, Phoenix, AZ, USA, 1997, pp. 969-973.

[44] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing", *IEEE/ACM Trans. Netw.*, 2006, 14 (3), pp. 479-491.

[45] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", In *Proc. of MOBICOM*, Boston, MA, USA, 2000, pp. 56-67.

[46] V. Handziski, A. Köpke, H. Karl, C. Frank, and W. Drytkiewicz, "Improving the Energy Efficiency of Directed Diffusion Using Passive Clustering", In *Proc. of EWSN*, Berlin, Germany, 2004, pp. 172-187.

[47] M. Gerla, T. J. Kwon, and G. Pei, "On demand routing in large ad hoc wireless networks with passive clustering", In *Proc. of IEEE WCNC*, 2000.

[48] T. J. Kwon and M. Gerla, "Efficient flooding with Passive Clustering (PC) in ad hoc networks", *ACM CCR*, 2002, 32 (1), pp. 44-56.

[49] M. Kubisch, H. Karl, A. Wolisz, L. C. Zhong, and J. Rabaey, "Distributed Algorithms for Transmission Power Control in Wireless Sensor Networks", In *Proc. of IEEE INFOCOM*, Anchorage, Alaska, USA, 2001.

[50] J. Aslam, Q. Li, and D. Rus, "Three power-aware routing algorithms for sensor networks", *Wireless Communications and Mobile Computing*, 2003, 3 (2), pp. 187-208.

[51] X. Guo, "Broadcasting for network lifetime maximization in wireless sensor networks", In *Proc. of IEEE SECON*, Santa Clara, CA, 2004, pp. 352-358.

[52] Y.-B. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) in mobile ad hoc networks", *Wireless Networks*, 2000, 6 (4), pp. 307–321.

[53] M. Joa-Ng and I.-T. Lu, "A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad Hoc Networks", *IEEE Journal On Selected Areas In Communications*, 1999, 17 (8), pp. 1415-1425.

[54] W.-H. Liao, J.-P. Sheu, and Y.-C. Tseng, "GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks", *Telecommunication Systems*, 2001, 18 (1-3), pp. 37-60.

[55] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks", *Mobile Computing and Networking*, 2000, pp. 243–254.

[56] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks", In *Proc. of International Symposium on Mobile Ad Hoc Networking & Computing*, Lausanne, Switzerland, 2002, pp. 194-205.

[57] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network", *Wireless Networks*, 2002, 8 (2-3).

[58] J. Cartigny and D. Simplot, "Border Node Retransmission Based Probabilistic Broadcast Protocols in Ad-Hoc Networks", *Telecommunication Systems*, January 2003, 22 (1-4), pp. 189-204.

[59] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks", In *Proc. of MSWIM*, Boston, Massachusetts, United States, 2000, pp. 61-68.

[60] W. Peng and X.-C. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks", In *Proc. of MOBIHOC*, Boston, Massachusetts, USA, 2000, pp. 129-130.

[61] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying: An efficient technique for flooding in mobile wireless networks", INRIA - Rapport de recherche, Technical Report 3898, 2000.

[62] J. Sucec and I. Marsic, "An Efficient Distributed Network-Wide Broadcast Algorithm for Mobile Ad Hoc Networks", Rutgers University, CAIP Technical Report 248, September 2000.

[63] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network", *Wireless Networks*, 1995, 1 (3), pp. 255-265.

[64] C. R. Lin and M. Gerla, "Adaptive Clustering for Mobile Wireless Networks", *IEEE J-SAC*, 1997, 15 (7), pp. 1265–1275.

[65] T. J. Kwon and M. Gerla, "Clustering with power control", In *Proc. of MILCOM*, Atlantic City, NJ, USA, 1999, pp. 1424-1428.

[66] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for multihop, mobile wireless networks", *Cluster Computing*, 1998, 1 (2), pp. 187-196.

[67] S. Basagni and E. Jonsson, "Distributed and mobility-adaptive clustering for multimedia supportin multi-hop wireless networks", In *Proc. of VTC*, Amsterdam, Netherlands, 1999.

[68] R. Krishnan, R. Ramanathan, and M. Steenstrup, "Optimization Algorithms for Large Self-Structuring Networks", In *Proc. of INFOCOM*, 1999.

[69] A. B. McDonald and T. F. Znati, "A Mobility-Based Framework for Adaptive Clustering in Wireless Ad Hoc Networks", *IEEE J-SAC*, 1999, 17 (8), pp. 1466-1487.

[70] P. Basu, N. Khan, and T. D. C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks", In *Proc. of ICDCS*, Phoenix (Mesa), Arizona, USA, 2001.

[71] I. I. ER and W. K. G. Seah, "Mobility-based d-Hop Clustering Algorithm for Mobile Ad Hoc Networks", In *Proc. of WCNC*, Atlanta, Giorgia, USA, 2004.

[72] S. Sivavakeesar, G. Pavlou, and A. Liotta, "Stable Clustering Through Mobility Prediction for Large-Scale Multihop Intelligent Ad Hoc Networks", In *Proc. of WCNC*, Altanta, Giorgia, USA, 2004.

[73] J. Wu and H. Li, "A Dominating-Set-Based Routing Scheme in Ad Hoc Wireless Networks", *Telecommunication Systems*, 2001, 18 (1-3), pp. 13-36.

[74] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating Sets and Neighbor Elimination-Based Broadcasting Algorithms in Wireless Networks", *TPDS*, 2002, 13 (1), pp. 14-25.

[75] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks", *Mobile Networks and Applications*, 2004, 9 (2), pp. 141-149.

[76] K. Fall, "A delay-tolerant network architecture for challenged internets", In *Proc. of Applications, Technologies, Architectures, and Protocols for Computer Communication*, Karlsruhe, Germany, 2003, pp. 27-34.

[77] X. Chen and A. L. Murphy, "Enabling Disconnected Transitive Communication in Mobile Ad Hoc Networks", In *Proc. of POMC*, Newport, Rhode Island, USA, 2001.

[78] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks", Duke University, Technical Report CS-200006, 2000.

[79] A. Beaufour, M. Leopold, and P. Bonnet, "Smart-tag based data dissemination", In *Proc. of International Workshop on Wireless Sensor Networks and Applications*, Atlanta, Georgia, USA, 2002, pp. 68-77.

[80] N. Glance, D. Snowdon, and J.-L. Meunier, "Pollen: using people as a communication medium", *Computer Networks*, 2001, 35, pp. 429-442.

[81] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks", Intel Research Seattle, Seattle, Technical Report IRS-TR-03-001, 2003.

[82] K. Viswanath and K. Obrazcka, "An Adaptive Approach to Group Communications in Multi Hop Ad Hoc Networks", In *Proc. of ISCC*, 2002, pp. 559.

[83] A. Khelil, P. J. Marrón, C. Becker, and K. Rothermel, "Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks." In *Proc. of The 2005 Conference on Communication in Distributed Systems (KiVS)*, 2005.

[84] J. A. Davis, A. H. Fagg, and B. N. Levine, "Wearable Computers as Packet Transport Mechanisms in Highly-Partitioned Ad-Hoc Networks", In *Proc. of ISWC'01*, 2001.

[85] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic Routing in Intermittently Connected Networks", In *Proc. of MobiHoc 2003*, Annapolis, MD, USA, 2003.

[86] S. Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant Network", In *Proc. of SIGCOMM*, 2004.

[87] P. Hui and J. Crowcroft, "Bubble Rap: Forwarding in small world DTNs in ever decreasing circles", University of Cambridge, Computer Laboratory, Cambridge, Technical Report 684, 2007.

[88] Q. Li and D. Rus, "Sending Messages to Mobile Users in Disconnected Ad-hoc Wireless Networks", In *Proc. of ACM MobiCom*, 2000.

[89] W. Zhao, M. Ammar, and E. Zegura, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks", In *Proc. of MobiHoc'04*, Roppongi, Japan, 2004.

[90] L. B. Ruiz, et al., "On the Design of a Self-Managed Wireless Sensor Network", *IEEE Communications Magazine*, 2005, 43 (8), pp. 95-102.

[91] T. A. Henzinger and J. Sifakis, "The Discipline of Embedded Systems Design", *Computer*, October 2007, 40 (10), pp. 32-40.

[92] L. Z. Granville, et al., "Managing Computer Networks using Peer-to-Peer Technologies", *IEEE Communications Magazine*, October 2005, 43 (10), pp. 62-68.

[93] D. Harrington, R. Presuhn, and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", RFC 3411, December 2002.

[94] G. S. Goldszmidt and Y. Yemini, "Distributed management by delegation", In *Proc. of 15th International Conference on Distributed Computing Systems*, 1995, pp. 333-340.

[95] D. Agrawal, K.-W. Lee, and J. Lobo, "Policy-Based Management of Netowrked Computing Systems", *IEEE Communications Magazine*, October 2005, 43 (10), pp. 69-75.

[96] R. V. Hogg and A. Craig, *Introduction to Mathematical Statistics*, Macmillan, New York, 1978.

[97] M. Radenkovic and B. Wietrzyk, "Mobile Ad Hoc Networking Approach to Detecting and Querying Events Related to Farm Animals", In *Proc. of ICNS*, Santa Clara, California, USA, 2006, pp. 109-115.

[98] B. Wietrzyk and M. Radenkovic, "Energy Efficiency in the Mobile Ad Hoc Networking Approach to Monitoring Farm Animals", In *Proc. of ICN*, Sainte-Luce, Martinique, 2007.

[99] M. Turner and M. Fogerty, "Labour use on UK farms: a pilot study", Agricultural Economics Unit, University of Exeter, Exeter, UK, 24, 1994.

[100] C. J. C. Phillips, *Principles of cattle production*, CABI Publishing, Oxon, UK, 2001.

[101] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, "Design Considerations for Solar Energy Harvesting Wireless Embedded Systems", In *Proc. of IPSN*, 2005.

[102] M. Radenkovic and B. Wietrzyk, "Wireless mobile ad-hoc sensor networks for very large scale cattle monitoring", In *Proc. of ASWN*, Berlin, Germany, 2006, pp. 47-58.

[103] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks", In *Proc. of 2nd ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, Seattle, Washington, United States, 1999, pp. 53-60.

[104] C. J. C. Phillips, *Cattle behaviour*, Farming Press, Ipswich, UK, 1993.

[105] L. Breslau, et al., "Advances in network simulation", *IEEE Computer*, 2000, 33 (5), pp. 59-67.

[106] F. J. Ros and P. M. Ruiz, "Implementing a New Manet Unicast Routing Protocol in NS2", Dept. of Information and Communications Engineering University of Murcia, 2004.

[107] R. Droms, "Dynamic Host Configuration Protocol", Bucknell University, RFC 2131, March 1997.

[108] H. Rannou and I. Orlova, "Workshop on “Research priorities for Home and Extended Home Networking”", European Commission, DG Information Society and Media, Available: ftp://ftp.cordis.europa.eu/pub/ist/docs/ka4/au_ws280606_rannou_en.pdf.

[109] Atmel, "Atmel AVR 8-Bit RISC - picoPower Technology ", Available: http://www.atmel.com/products/avr/picopower/.

[110] STMicroelectronics, "LIS3L02AL MEMS INERTIAL SENSOR: 3-axis - +/-2g ultracompact linear accelerometer", Available: http://www.st.com/stonline/books/pdf/docs/11668.pdf.

[111] STMicroelectronics, "GS-BT2416C2 Bluetooth class 2 module", Available: http://www.st.com/stonline/products/literature/ds/13002/gs-bt2416c2.pdf.

[112] FreescaleSemiconductor, "MC13192 2.4 GHz Low Power Transceiver for the IEEE® 802.15.4 Standard", Available: http://www.freescale.com/files/rf_if/doc/data_sheet/MC13192.pdf.

[113] Atheros, "AR6001GL Embedded 802.11b/g Solution for Mobile and Battery-Operated Devices", Available.

[114] E. Jovanov, A. Milenkovic, C. Otto, and P. C. d. Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation", *Journal of NeuroEngineering and Rehabilitation*, 2005, 2 (6).

[115] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "CodeBlue: An Ad Hoc Sensor Network Infrastructure for Emergency Medical Care", In *Proc. of Workshop on Wearable and Implantable Body Sensor Networks (BSN)*, Imperial College London, United Kingdom, 2004.

[116] M. S. Gast, *802.11 Wireless Networks: The Definitive Guide, Second Edition*, O'Reilly Media, Inc., Sebastopol, CA 95472, USA, 2005.

[117] ETSI, "GSM 06.51 version 4.0.1", Available: http://www.3gpp.org/ftp/Specs/archive/06_series/06.51/0651-401.zip.

[118] A. J. Menezes, P. C. v. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 2001.

[119] D. Kotz and T. Henderson, "A Community Resource for Archiving Wireless Data At Dartmouth", Available: http://crawdad.cs.dartmouth.edu/.

[120] W. D. Jones, "Taking Body Temperature, Inside Out", *IEEE Spectrum*, 2006, 1, pp. 11-13.

[121] G. Záruba, F. Kamangar, M. Huber, and D. Levine, "CONNECT: A Personal Remote Messaging and Monitoring System to Aid People with Disabilities", *IEEE Communications Magazine*, 2005, 43 (9), pp. 101-109.

[122] K. Fall and K. Varadhan, *The ns Manual*, 2008.

[123] D. Kotz, et al., "Experimental Evaluation of Wireless Simulation Assumptions", In *Proc. of MSWiM*, Venezia, Italy, 2004.

[124] J. Liu, et al., "Simulation Validation Using Direct Execution ofWireless Ad-Hoc Routing Protocols", In *Proc. of 18th Workshop o Parallel and Distributed Simulation (PADS'04)*, 2004.