BirdTracking: A Wireless Sensor Network to Observe Bird Life

Tabish Bari

embedded software

TU Delft Delft University of Technology

BirdTracking: A Wireless Sensor Network to Observe Bird Life

Master's Thesis in Embedded Systems

Embedded Software Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands

Tabish Bari
tabish.bari@gmail.com

18th August 2010

Author
Tabish Bari (tabish.bari@gmail.com)
Title
BirdTracking: A Wireless Sensor Network to Observe Bird Life
MSc presentation
24th August 2010

Graduation Committee

Prof. dr. K.G. Langendoen (chair)	Delft University of Technology
dr.ir. D. H. J. Epema	Delft University of Technology
dr.ir. F. A. Kuipers	Delft University of Technology
ir. O. W. Visser	Delft University of Technology

Abstract

Field biologists and ecologists have started to open a new avenue of inquiry at greater spatial and temporal resolution, allowing them to "observe the unobservable" through the use of wireless sensor networks. Study of birds has helped to develop fundamental knowledge of bird behavior, foraging pattern and migration. The acquired knowledge of birds has contributed to build concepts, like evolution, and applications, such as mitigating risk of bird-strikes and protecting endangered species. Traditional bird life monitoring approaches, like satellite telemetry are not capable to provide the insights in a greater resolution and suffer from large delay to deliver data.

In this thesis we present the communication mechanism of *BirdTracking*, a wireless sensor network to observe bird life for a complete annual bird-cycle. One of the main challenges of a sensor network formed by devices attached to birds is the disrupted connectivity due to mobility and habitat of birds. We propose CHIRP, a routing protocol that utilizes the behavior and mobility of birds to transmit sensor data to a collection point. In order to evaluate CHIRP, we implemented it on a wireless network simulator over a mobility model created by real life traces of a colonial bird (gull) and a territorial bird (honey buzzard). We compare CHIRP against direct transmission and an epidemic routing scheme. Our simulation results show that CHIRP achieves significant improvement in data delivery as compared to direct transmission while consuming less resources than the epidemic routing scheme.

"The greatest achievement was at first and for a time a dream. The oak sleeps in the acorn, the bird waits in the egg, and in the highest vision of the soul a waking angel stirs. Dreams are the seedlings of realities." – James Allen

Preface

Being a student majoring in Embedded Systems I was looking for a research project which fitted into my curriculum. The Embedded Software group at the TU Delft was the first choice of my search for a prospective research project. I was offered the opportunity to work on a new exciting project (BirdTracking) where I had to apply my research skills to identify the problem, design and implement a solution on the sensor device.

My research would not have reached this stage without the help of my supervisors at TU Delft. I would like to thank my professor, Koen Langendoen for his guidance on directing and refining my research. His comments and suggestions have greatly helped me broaden my understanding and put me on right track. I would also like to thank my immediate supervisor, Otto Visser for his patience and help on my trivial issues as well as guiding me to achieve my goal. Both my supervisors gave me sufficient freedom and cooperation to develop my research according to my views while ensuring I do not fly like a free bird with the wings of my ideas to a point of no return. Their comments helped me overcome my lack of formal writing and produce the written report in this form. A word of thanks to my colleagues in the Embedded Software group, Venkat, Andrei, Neils and others for showing interest in my work and helping me whenever needed.

A final word of thanks to Professor Bouten of the University of Amsterdam on providing useful information on birds that helped me to set a right direction of my research.

I owe my achievement to my mother who has always motivated me in every up and down of my life.

Tabish Bari
18th August 2010

Contents

Preface **vii**

1 Introduction **1**
1.1 BirdTracking sensor device . . . 3
1.2 Research approach . . . 4
1.3 Research questions . . . 4
1.4 Organization . . . 4

2 Background and Related Work **5**
2.1 A bird's-eye view of behavior . . . 5
2.1.1 Nesting and breeding . . . 5
2.1.2 Migration . . . 6
2.2 Related work . . . 6
2.2.1 WSN for wildlife monitoring . . . 6
2.2.2 Communication mechanism . . . 8
2.3 Discussion . . . 10

3 Design **13**
3.1 Network model . . . 13
3.2 The CHIRP protocol . . . 14
3.2.1 Assumptions . . . 14
3.2.2 Protocol details . . . 14
3.2.3 Protocol operations . . . 15

4 Implementation **21**
4.1 Link layer . . . 21
4.2 Network layer: CHIRP . . . 21
4.2.1 Timer-based events . . . 22
4.2.2 Network events . . . 23
4.2.3 Packet structure . . . 27
4.2.4 Additional features of CHIRP . . . 27

5 Experiments and Results **29**
5.1 Reference protocols . . . 29
5.2 Simulation setup . . . 29
5.2.1 MiXiM . . . 29
5.3 Experiments . . . 31
5.3.1 Delivery . . . 33
5.3.2 Average delay . . . 36

5.3.3 Resource utilization 36
5.3.4 Multiple sinks 38
5.3.5 Discussion 39

6 Conclusions and Future Work **41**
6.1 Conclusions 41
6.2 Future work 42

Bibliography **45**

Chapter 1

Introduction

The whole animal kingdom is a complex web of inter-linked biological information, which has intrigued scientists for centuries. Nature is a complex perpetual system of which the animal kingdom is a part. Any disruption in natural processes either due to humans or any natural reason can cause a cascading effect and result in animal species getting extinct, which in a way would effect humans too. One class of animals with *feather-covered bodies, a bill, two legs and two wings that are usually adapted for flight* are known as birds, which virtually inhabit every terrestrial habitat on the planet with a global population estimated as 300 billion and 9000 known species. Bird study has a long history starting from the *stone age* and the key concepts have contributed in evolution, behavior and ecology study. For example the famous "Theory of natural selection" by Charles Darwin, which was primarily based on observation of *Galapagos birds*, has changed the way we understand the evolution of not only humans, but also other animals. Researchers are interested to study birds due to the following two major reasons:

Fundamental knowledge. There are many aspects of bird life that are interesting to scientists such as: interaction between birds over the complete annual bird-cycle (i.e., nesting, breeding and migration), movement and foraging patterns, behavioral changes due to environmental factors, flight strategy and motion of birds.

Applied knowledge. The insight knowledge on movements of birds can be used to improve safety of commercial and military aviation by avoiding bird-strikes (mid air collision between aircraft and birds), which is a significant threat to flight safety and has caused numerous accidents with human causalities. Other fields of application are protection of endangered species, research on diseases transferred by birds, like the H1N1 virus (bird flu) and the offshore wind farms risks on birds mortality.

The tools used to observe animal life have evolved from simple manual observation to present day technology-assisted monitoring. In the past decades, due to the advent of microelectronic and communication technologies, it has become possible to obtain the biological and ecological knowledge of animal life in detail. There are several traditional approaches for wildlife monitoring for example, *attach-and-collect*, *Very High Frequency (VHF) radiotelemetry* and *satellite-telemetry*. In attach-and-collect, animals are tagged with sensor devices and released in their habitat. The sensors collect data for a period of time after which the tags are recollected. For example, DTAG [1] tags

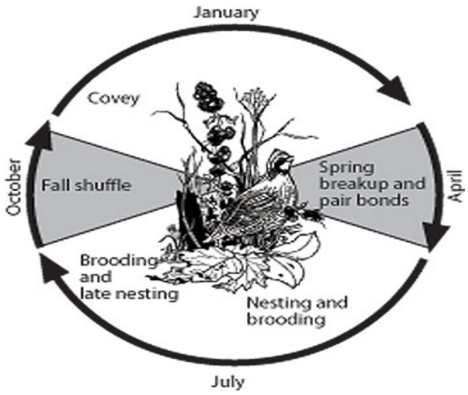


Figure 1.1: Annual bird-cycle

whales with a custom sensor device. The tag automatically detaches itself from the whale's body after a period of time (when the memory is full) and floats to the surface, where it can be collected by researchers. A major drawback of this scheme is that it is not done in real time; data can not be analyzed until the end of the experiment. VHF radiotelemetry is used to track small numbers of animals for a very limited time. The Princeton Wildlife Monitoring Project [2] is an example of automated VHF radio telemetry, which tracks tagged animals in a tropical forest.

Satellite telemetry is the most commonly used method for wildlife monitoring. In this scheme Low-Earth Orbitting Satellites (LEO) like, PASCAT, ARGOS, Global Positioning System (GPS) etc., are used to send data from a sensor device attached to an animal to a remote collection point. These satellites revolve around the earth many times a day and are present at any position on the earth for 5-8 times a day for a duration of 5-10 minutes. A radio receiver (Argos, GPS etc.) is attached to the animal that transmits a signal that can be received by the satellites. If the satellite passes over the location of the animal and a possible communication could be established (usually animal habitats hinder radio communication) then the received signal is transmitted to a remote collection point. Most of the current bird monitoring systems like, Falcon-Trak [3] and Raptor Research [4] use satellite-based telemetry, typically monitoring position of the birds. On the one hand animal monitoring based on satellite telemetry can span a very large observation area, whereas on the other hand it can only provide position information, and data delivery suffers from very large delay.

A new technology paradigm is entering the field of wildlife monitoring known as wireless sensor network (WSN). A WSN is a collection of mini-sensor devices that can organize themselves in an ad-hoc fashion and collaborate to achieve some common goals. Due to its salient features like unattended operation, low power and distributed approach, WSNs have started to gain preference over traditional wildlife monitoring approaches (VHF, satellite telemetry etc.). *BirdTracking* is a wireless sensor network to observe bird life for a complete annual bird-cycle (Figure 1.1), starting with nesting and breeding in spring, going onto migration in winters and returning back in next spring. As far as we know this is the first attempt to use a WSN to observe bird life, wherein sensor devices (tags) are attached to birds. *BirdTracking* is capable of providing fine-grained information (position, temperature, pressure and flight strategy) as

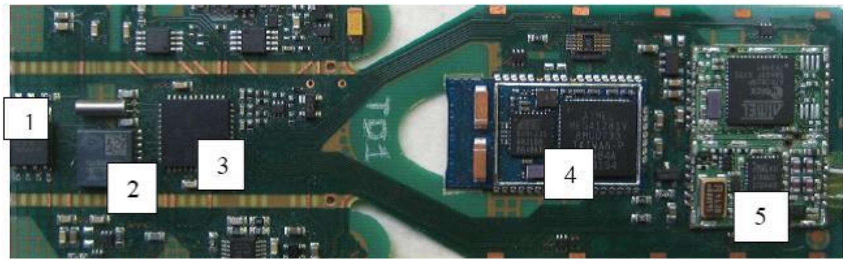


Figure 1.2: BirdTracking sensor device

compared to FalconTrak [3] and Raptor [4] that could only provide position information. It offers exciting new avenues to study birds, which provides valuable information for fundamental scientific research and furthermore utilizing the information for applied knowledge. One of the applications of BirdTracking is to develop the FlySafe Bird Avoidance Model (FlySafe-BAM) [5] and Avian Information Systems (AIS) [6]. These systems mitigate the risk of bird-strikes by developing bird migration prediction systems, which provide automated services to civil and military aviation.

1.1 BirdTracking sensor device

The hardware components are explained in the same sequence as numbered in Figure 1.2.

1. **Flash memory**. A 4 megabytes flash memory is available on the sensor device to store sensor data.

2. **Sensors**. A 3-axis acceleration sensor module that is used to record wing beat frequency and translational acceleration of the bird. There are two other sensors (pressure and temperature) present on board that inform about the local environmental changes, which can be further related to behavioral changes of the bird.

3. **CPU**. A 4 MHz PIC18F4685 microcontroller is used to manage all application and system tasks such as controlling the duty cycle of sensors and GPS, memory operation, radio interface, etc.

4. **Radio**. To transmit data wirelessly the ZigBee radio module (Meshnetics ZigBit-MNZB-24) was chosen. Later in the development of our communication scheme we only use the link layer of the ZigBee protocol stack (IEEE 802.15.4 [7]). The radio module operates in the 2.4 GHz Industrial, Scientific and Medical (ISM) radio frequency band at the maximum bit rate of 250 kbps and maximum transmission power of 3 dBm. The receiver sensitivity is -101 dBm and it consumes 18 mA in transmitting mode and 19 mA in receiving mode.

5. **GPS receiver**. An Atmel u-blox GPS module LEA-4S is used to record the bird's position. The device is capable of achieving one fix per 3 seconds.

6. **Solar cells**. Not shown in the picture, attached at the time of deployment.

1.2 Research approach

A wireless sensor network of tagged birds is expected to function unattended and autonomously for the complete annual bird-cycle. We progress in an incremental fashion dividing the research into the following phases:

In ***phase 1***, we concentrate on collecting data during the nesting and breeding season. The observation area during this time of the cycle spans a few hundred kilometers around the nest, which makes it possible to deliver data in real time. The objective is to devise a communication scheme that can deliver sensor data to a collection point i.e. *the sink*. Additionally, it is also expected by the communication mechanism to be able to deliver control messages (sensing interval updates etc.) from the sink to birds.

In ***phase 2***, we concentrate on collecting data during bird migration. During this time the birds move thousands of kilometers from their nests, which makes it near to impossible to deliver data in real time. Data can only be collected when the birds complete their cycle and return back in spring. Efficient data gathering and in-network processing within a flock of birds could be implemented to gather data for longer periods of time. Some of the approaches envisaged inclulde clustering, assigning rotating duties for data sensing, interpolating and sharing data within a flock.

1.3 Research questions

In this thesis work we focus on phase 1 (nesting and breeding) of BirdTracking. We attempt to devise a communication scheme that can also be utilized in phase 2 (migration) with minimal modifications. The network formed by tagged birds would suffer from long duration partitioning and intermittent links due to the mobility of birds and their habitat (canopy, marshy land, etc.). Furthermore, the sensor device can only have a certain maximum storage and battery resources due to size and weight limitations. Considering these challenges we place the following research questions:

- *How can we deliver sensor data to the sink in a dynamic and intermittently connected network with limited bandwidth, power and storage resources?*
- *How can we utilize the behavior and mobility of birds to improve data delivery?*
- *What is the delay in data delivery? Can we achieve real-time data delivery?*

1.4 Organization

This thesis is produced as a result of work done at TU Delft in collaboration with the University of Amsterdam (partner in FlySafe-BAM project). The remainder of the thesis is organized as follows. The background on birds behavior, a related study carried out on existing WSNs implementations for wildlife monitoring and studied protocols are discussed in Chapter 2. Chapter 3 explains the design details of a new routing protocol. The protocol implementation is mentioned in Chapter 4. The experimental setup and results are discussed in Chapter 5 and finally the conclusion and plans for future work are mentioned in Chapter 6.

Chapter 2

Background and Related Work

This chapter is organized into three broad sections. Firstly, a glimpse on bird behavior is provided that enables to understand the physical space of the problem and derive valuable information useful for our approach. Secondly, insight is provided into the related wireless sensor networks implemented for wildlife monitoring followed by analysis of related routing protocols. Lastly, a comparison of BirdTracking with related WSNs implementation is done followed by the routing strategy that sets our direction of design-space exploration.

2.1 A bird's-eye view of behavior

In this section we study the physical space of the problem and collate essential details, which are used to build upon our network. The main sources of information are Birds of Stanford [8], Ornithology [9] and the biologists from the University of Amsterdam (our partner university). Birds are classified based on various biological and behavioral parameters, for example, feathers, beaks, eggs, mating, migration, cooperative-breeding etc. Furthermore, birds belonging to a specific species show different behavior at different times of the year. We focus on the available intra-species behavioral knowledge and classify according to the seasonal behavior.

2.1.1 Nesting and breeding

During this phase of the bird-cycle, courtship, nesting and breeding happens. This usually starts in spring and continues until the start of winter. During the nesting and breeding season, a bird can travel several kilometers from the nest. This movement could be for search of food, for a new nesting place or for any other behavioral factors.

Territorial birds. Usually most birds show territorial instinct during nesting and breeding season. However, what exactly constitutes a territory is a matter of definition and varies according to the species. An *all-purpose-territory* constitutes a nesting, mating and feeding area, for example with song birds. A *mating-and-nesting* territory is one within which all nesting and breeding occurs, but foraging occurs elsewhere, for example with raptors, hawks, buzzards etc. A nest territory is the smallest notion of a territory, which spans a nest and a small area close to it, for example, with water birds (gulls, albatrosses etc.,). Birds could be extremely territorial while nesting and go to great lengths to expel each other from their respective territories, although these same birds flock to migrate in winter. Due to the dominant territorial behavior social interaction between birds declines during this time. However, some birds like parrots show social behavior during nesting and breeding when they go for collaborative foraging or expelling predators.

Colonial birds. A bird colony is a large congregation of birds of one or more species nesting in close proximity (many single-nest territories). Approximately, about 13% of all bird species are colonial. The colony is usually situated close to an abundant food source and at a distance from predators. The birds forage individually without any collaboration from other colony members. Some examples of colonial birds are gulls, albatrosses etc.

2.1.2 Migration

Many bird species take long distance annual *migrations* and many more take short distance irregular movements during winter. Many species show social behavior during migration when they flock together in foraging groups.

Table 2.1 compares a few example birds that show different behavior at different time of the annual cycle. For example, honey buzzards are territorial in nesting-and-breeding, but they form foraging groups to migrate in winter. On the other hand gulls are colonial in nesting-and-breeding and also migrate long distance in foraging groups. Many smaller birds like chiffchaffs are territorial in spring and do short distance migration with their partner in winter.

2.2 Related work

In the following section we first study the wireless sensor networks that are used for wildlife observation. Second, we perform a study on the communication mechanisms that are related to BirdTracking network.

2.2.1 WSN for wildlife monitoring

Wireless sensor networks have gained the attention of field biologists and ecologists interested in analyzing the biological and eco-diversity systems in greater resolution than was possible with previous techniques (VHF, satellite telemetry, etc.). By using WSNs, researchers can now extend their labs to observation fields. Depending on the way in which sensor nodes are deployed there are two types of WSNs for wildlife and habitat monitoring:

Static network. The sensor nodes are deployed statically near *hot spots* chosen by field biologists. These hot spots are burrows, nests and rendezvous places. Great Duck Island (GDI) [10] is an examples of a WSN deployment under this category.

Dynamic network. Another type of sensor node placement involves attaching devices to the animals under observation often referred to as *tagging* or *collaring*. On the one hand this technique provides greater resolution of data collection and online or

<table>
<tr><th colspan="3">Nesting and breeding</th></tr>
<tr><th></th><th>Territorial</th><th>Colonial</th></tr>
<tr><td>Migration in flocks</td><td>
Honey Buzzard: mating-and-nesting territory, avoid each others territory and migrate in flocks.</td><td>
Gulls: congregation of nests near the sea or abundant food source and the colony migrates long distance in winter.</td></tr>
<tr><td>Individual migration</td><td>
Chiffchaff: all-purpose-territory and short distance migration with partner in winter.</td><td>(All colonial birds do migration in foraging groups.)</td></tr>
</table>

Table 2.1: A classification of some birds based on their seasonal behavior. The columns show the behavior of the bird during nesting-and-breeding and the rows show the behavior of the same bird during winter season.

delayed delivery to sink, whilst on the other hand it imposes challenges on the wireless network due to intermittent connectivity, inconspicuous operations, network longevity and remote management. ZebraNet [11], RatPack [12], Networked-Cows [13], EPSRC WildSensing [14] (Badgers) and Seal-2-Seal [15] (Figure 2.1) are some of the related WSNs for wildlife monitoring.

Table 2.2 compares the sensor network deployments based on application constraints, use cases derived from animal behavior, and sink placement. One of the unique attributes that is found in all example WSN implementations is that the observed animals are mammals. There is some kind of social behavior found in all mammals, which is not a flexible behavior that can be abandoned or adopted over time, as found in birds. The applications vary with species under observation and are usually decided by the domain experts considering what behavioral information is to be studied. The place-

(a) RatPack

(b) Seal2Seal

(c) ZebraNet

Figure 2.1: Wireless sensor networks for wildlife monitoring

ment and number of sinks is very critical in data delivery for dynamic nodes (an animal carrying a sensor device). In ZebraNet, a mobile sink is used, which moves through the observed area to collect data from animals. Multiple sinks are placed at rendezvous points (burrows exits) in Badgers and RatPack or some animals are chosen as sinks in Seal2Seal.

	Application	**Use Case**	**Sink**
ZebraNet	Position (GPS), other sensed data.	Social.	Single and mobile.
RatPack	Vocalization and motion.	Social communities.	Multiple and fixed at the exit of burrows.
Networked-Cows	Position (GPS), actuation with sound.	Single herd.	Single and fixed.
Badgers	Position, selective reprogramming.	Social communities.	Multiple and mobile.
Seal2Seal	Contact data.	Not considered.	Tagged animals as sink.

Table 2.2: A classification of related WSNs for wildlife monitoring

2.2.2 Communication mechanism

The above-mentioned WSNs as well as BirdTracking are mobile ad-hoc networks. Due to the mobile nodes (i.e. tagged or collared animals) data delivery at the sink would be disrupted and delayed. Such a network is abbreviated to *Delay or Disruption Tolerant Network* (DTN) or *Intermittently Connected Mobile Network* (ICMN). Traditional ad-hoc or mobile routing approaches assume there exists at least one route from a source to a destination. However, due to frequent and long partitioning of the network and seldom end-to-end contemporaneous path, such an assumption is not valid for DTN and ICMN.

Protocol Studies

Based on the communication schemes used in related WSN deployments and DTNs, the studied protocols are divided into two classes. One class of schemes utilizes social-based topology information to build upon a routing scheme. The other class of routing schemes do not assume any network topology and devise routing schemes based on a route metric.

Social network. Social behavior is commonly observed in mammals. A social network is a logical structure of individuals linked with each other through some kind of social relationship like friendship or through hierarchy of roles. Social networks show strong clustering [16, 17] and tend to congregate in communities. Members of a community are usually closer to each other than to the members of other communities. In a social community some members are more popular and interact more with other communities thus making them members with high centrality (social power of a node based on how well it "connects" the network). Such members become message carriers from one community to another. Both RatPack and Badgers devise a routing scheme based on social interaction between animals. Social community identification can also be used to disseminate code selectively to target communities as shown by Pásztor et al. [14].

DTN route metric. The routing objective in traditional routing schemes is to minimize certain chosen metrics (e.g. numbers of hops, numbers of retransmissions etc.). For DTN routing the most desirable objective is not immediately obvious. As DTN routing works in a *store-and-forward* fashion, this may lead to packets getting lost in the network due to route loops or insufficient buffer space. One obvious goal is to maximize the probability of message delivery. Unfortunately, there is no precise approach to maximizing message delivery. One class of schemes [18, 19, 20, 21, 22] follow an *epidemic* approach by injecting many replicas of a message in the network to maximize delivery probability. Other approaches [23, 24] claim that minimizing the delay of a message (the time between when a message is injected and when it is completely received) is a good heuristic to maximize the message delivery probability. In *epidemic* routing naive flooding wastes resources and can severely degrade performance. The proposed protocols attempt to limit resource wastage in many ways:

- Seal-2-Seal creates a compact summary of contact data logging and attempts to transfer summaries in a chronological order. It conserves resources by disseminating less data and avoiding any message identifier exchange.
- MaxProp [19], PROPHET [25] and Seek-And-Focus [26] use historic meeting information to limit replication. PROPHET utilizes a probabilistic routing approach by assigning a utility (delivery predictability) to each source and destination pair. Seek-And-Focus is a hybrid routing scheme that starts randomly flooding data with a certain probability and later only forwards data that is based on utility as in PROPHET.

- MaxProp and RAPID [24] purge stale packets (copies of a packet present in the network, although the packet is delivered to the destination) by disseminating acknowledgments of delivered packets .
- SWIM [20] uses a probabilistic mobility model (random directional) to infer delivery. SWIM is built upon a Markov model, which assumes node-to-node and node-to-sink encounters occur at exponentially distributed time intervals and all nodes have the same probability to reach the sink.
- Spray-and-Wait [18] and SWIM bound the number of replicas of a packet in the network. Spray-and-Wait limits the number of packet replicas by adding a maximum hop count field in the packet, whereas SWIM bounds replication by setting a threshold on the maximum number of replicas of a packet allowed in the network.

Jain et al. in [23], propose a forwarding scheme to minimize average delay of a packet using oracles with varying degrees of future knowledge. RAPID [24] claims that even the simplest oracle in [23] would be difficult to implement as transfer opportunities are affected by many factors in practice due to physical environment, interference etc. RAPID [24] is a packet replication scheme that heuristically calculates estimated delay based on future contact opportunities, and replicates packets based on marginal utility. It is furthermore claimed that all other proposed DTN routing schemes are *incidental* , however RAPID is an *intentional* replication scheme. In RatPack, resource parameters like energy level and available storage at the other node are combined with the average delay to create a composite utility function to devise a *Utility Based Forwarding* scheme to transmit content-based data.

2.3 Discussion

With respect to application constraints, *BirdTracking* is close to ZebraNet and RatPack. However, there is no varying priority on gathered data as in RatPack, where initially the location of burrows is of interest and vocalization is deemed more important at a later stage. The communication schemes used in related WSNs cannot be directly implemented in BirdTracking as most of the implementations utilize the social behavior of target animals (mammals). Social behavior in birds is either not observed or appears at certain times of the year (Section 2.1). At this stage of the project we are more interested in observing the behavior during the nesting-and-breeding season. During this time social behavior in birds declines or is completely abandoned.

With DTN routing schemes the choice is between *replication* and *forwarding*. Replication shows better data delivery when no information on future encounters is available, with an overhead of severe resource depletion. If information on future contacts between nodes can be deduced either probabilistically or on a historical basis then a forwarding scheme could be implemented. Furthermore, some of the proposed DTN routing schemes (PROPHET, Epidemic, Spray-and-wait) do not consider bandwidth and buffer limitations and emphasis more on data delivery. Other protocols like MaxProp, RAPID and routing schemes proposed by Jain et al. in [23] route data with finite bandwidth and storage. DTN routing schemes are based on a mobility model, which could be synthetic or build using real life traces. Some of the commonly used mobility models are vehicular DTN traces (MaxProp, RAPID and Jain et al. [23]), community-based (PROPHET, Badgers and RatPack) or synthetic DTN models. Our goal is to study different bird behavior and mobility patterns to develop a routing scheme or a set of routing schemes to transmit data to sink. For *BirdTracking* we could use multiple sinks or one sink with extended reception area by implementing relays. Either of these techniques is used by researchers on a case-by-case basis. The sink is usually statically placed near to a nest or colony.

Chapter 3

Design

In the following chapter we explain the design of our communication scheme. We start with creating a network model (Section 3.1) that could be realized as a network of tagged birds. We propose to devise a DTN routing scheme by taking into account the available behavioral knowledge of birds. The routing protocol is discussed in Section 3.2 mentioning the assumptions, design details and operations. More details on the implementation of the routing scheme are provided in Chapter 4.

3.1 Network model

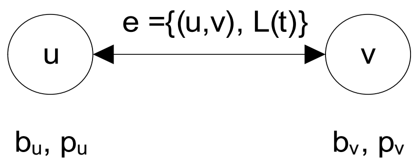


Figure 3.1: Network model.

We model the BirdTracking network as an undirected graph where two nodes (tagged birds) are connected by a single bidirectional link (Figure 3.1) also called contact. In practice the links are usually directional, however in BirdTracking, the MAC layer, which is based on the IEEE 802.15.4 standard (Section 1.1) incorporates an automatic repeat request (ARQ) that informs the network layer about the link asymmetry (through retransmission failures). In other cases the bidirectional link is verified by methods like handshaking. The nodes have finite storage and energy (represented as b and p in Figure 3.1 respectively). In a typical DTN, the link components *capacity*, *quality*, and *propagation delay* vary over time, which results in the link characteristics L(t) as time variant. The network is almost all the time operational outdoor so it is safe to neglect the interference effect on capacity due to other ISM radio band services. For simplicity of our protocol design we assume that the propagation delay is negligible. In literature, there are several ways to incorporate link quality in routing metrics like ETX [27] and LIM [28]. In DTN, the link quality can change due to various practical reasons (habitat etc.) therefore an estimate of link quality at a given instance would not be valid in future encounters. We follow an opportunistic approach rather than a deterministic one to devise our routing scheme. We simplify the time-variant nature of a link by considering L(t) as one i.e. capacity is ideal, no propagation delay and link quality is 100%

when the two nodes are in contact with each other, else it is zero.

$$L(t) = \begin{cases} 1 & \text{if there is a contact,} \\ 0 & \text{otherwise.} \end{cases}$$

3.2 The CHIRP protocol

In the following section we present the *Contact-based, Hybrid and receiver-Initiated Routing Protocol* (CHIRP) for dynamic and sparse networks, which transmits messages opportunistically from one bird to another upon contact. To devise our protocol we have utilized the available knowledge of birds' behavior (Section 2.1) in collaboration with routing metrics to steer routing decisions.

3.2.1 Assumptions

Based on the behavioral information mentioned in Section 2.1 we can deduce the following assumptions upon which CHIRP is based:

1. There is a logical territory that is either a single nest, a group of several nests, or a colony of nests.
2. Two birds that encounter each other at any given time are expected to be in contact again in the future. This assumption is supported by either all or any of the following behavioral reasons:
   - The birds belong to the same territory. Meeting birds could be partners dwelling in the same nest, partners, but with one belonging to several nests (polygamous family) or different birds belonging to different nests that are close enough.
   - The birds are part of a collaborative foraging group.
   - The birds have the same preference for place or direction for foraging.

3.2.2 Protocol details

CHIRP is a receiver-initiated routing scheme. The main idea behind the receiver-initiated scheme is to let the destination node initiate packet exchange. In particular, when a node X periodically wakes up it transmits a beacon announcing that it is awake. The beacon message is broadcast in nature such that any node that is awake and within X's radio range can receive it. Neighboring nodes that receive X's beacon know that X is awake, thus establish a contact with X. Subsequently, the neighbor can decide to send data packet to X. The beacon interval (T_{beacon}) is fixed for all nodes and set at the time of deployment. The sink also broadcasts beacons at the same interval; the only difference being that it is always awake.

CHIRP bases its routing decisions based on contact between nodes. Periodic beacon advertisement is utilized to detect the presence of other nodes within the transmission range. The beacon interval should be chosen suitably to detect the maximum number of contacts without consuming much energy. We are interested in formulating the contact detection in a metric that could imply the number of times birds were co-located. There could be several ways to achieve this, like delivery predictability used in PROPHET [25], contact ratio used in Badgers [14] and total-contact duration [17]. The key idea in PROPHET and Badgers is to augment the metric on each encounter and decay the metric if the nodes stay apart longer, whereas total-contact duration considers how long the contact between two nodes lasts. Contact ratio and total-contact duration are based on social community detection. Contact ratio is better at capturing dynamic behavior of communities whereas total-contact duration shows better insight on stable behavior of communities over time.

We know from Section 2.1 that social behavior in birds is not profound and the maximum possible chances of co-location are when they belong to the same territory. Territories stay stable over the course of time and do not change until the birds decide to migrate. Taking this into consideration we assign a contact probability (CP) to each pair of nodes in the network. The contact probability (CP) is controlled by three functions:

- *IntializeContactProbability*: on meeting a node for the first time an initial probability P_{INIT} is assigned to CP.
- *UpdateContactProbability*: for every contact i.e. when a node receives a beacon from another node, the CP is increased as
  $CP_{new} = CP_{old} \cdot \alpha$, where $\alpha > 1$.
- *DecayContactProbability*: if the nodes stay apart for a long time the CP should be able to refresh itself. This is achieved by decaying CP at every time interval (T_{decay}) as $CP_{new} = CP_{old} \cdot \beta$, where $\beta < 1$.

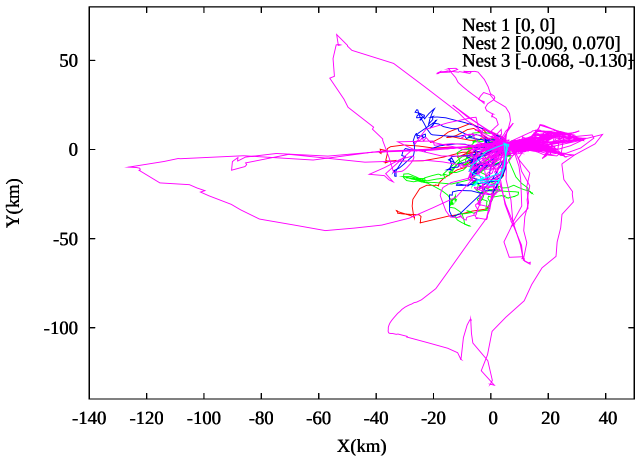
CHIRP is a hybrid routing scheme that utilizes the benefits of both replication and forwarding. CHIRP replicates data packets to members of the same territory (Section 3.2.3) until a suitable path to the sink is established via one or more members of the territory. CHIRP utilizes the available knowledge of foraging patterns of birds to develop a path. Based on our assumptions (Section 3.2.1) we expect that a single or multiple members of a territory would find a direct or indirect route to the sink at some time. When a path to the sink is established a metric, *maximum predicted delay* (MPD) (Section 3.2.3) is assigned to each path. To select a superior route to the sink, the path with the minimum MPD is chosen.

3.2.3 Protocol operations

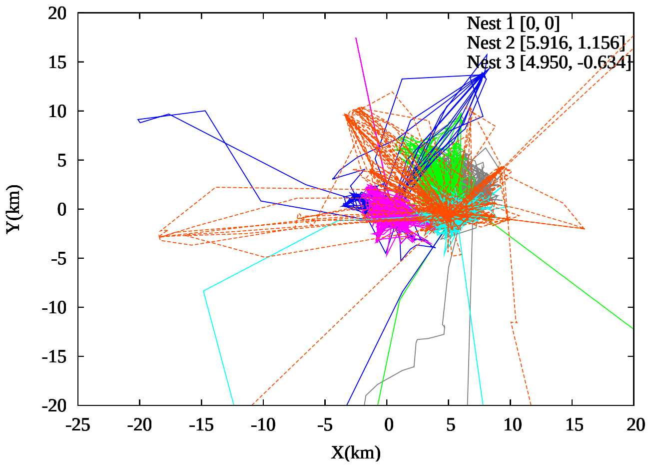
Identifying territories

We define a territory as a set of nodes that have encountered each other for a certain amount of time. Our interpretation of a territory is different from that mentioned in Section 2.1 because our contact estimation is based on radio range, which does not consider the co-location based on physical distance. For example, it is observed in a simulation run over position traces of gulls (scenario 1 in Figure 3.2(a)) that one territory spans several nests whereas for territorial birds a territory is found to be a single nest (scenario 2 in Figure 3.2(b)). Each node in a territory is one hop away from every other node belonging to the same territory, resulting in a single hop cluster.

The protocol starts with all nodes associated to a single-member logical territory with a default identifier (DEFAULT_ID). As time passes, the CP of nodes encountering each other often increases. When the CP reaches a certain threshold $CP_{Territory}$



(a) Scenario 1 (gulls): the inter-nest distance is in range 10-50 m. Each gull forage individually and frequently roosts around the nest.



(b) Scenario 2 (honey buzzards): these birds form a mating-and-nesting territory, where the inter-nest distance is 1-3 km. One of the partner fly around the nest to protect the territory while the other goes for foraging. Usually they tend to avoid each others territory.

Figure 3.2: Two-dimensional movement of a colonial bird (gulls - Scenario 1) and a territorial bird (honey buzzards - Scenario 2). One of the nest location is taken as reference point (0, 0) while converting GPS coordinates to cartesian coordinates.

a make or join territory operation takes place. $CP_{Territory}$ is expected to be defined by the domain experts based on the species under observation. The algorithm TERRITORY_OPERATION (Algorithm 1) explains the territory operation. The beacon message contains the neighbors' territory identifier (*nghbrTID*), territory size (*nghbrTSize*) and list of territory members (*nghbrTMembers*), which is the input to the algorithm. The output generated is the node's territory identifier (*myTID*), territory size (*myTSize*) and members of the territory (*myTMembers*). Initially *myTID* is one, *myTSize* is one and *myTMembers* contains just the node's address. Contact probability is calculated with all members of the neighbor's territory and if CP of the node with all neighbor's territory members is greater than $CP_{Territory}$ then any of the following action takes place depending upon the condition satisfied:

- **makeNewTerritory**: if the node and the neighbor both belong to the default territory a new territory is created with a unique identifier.
- **updateTerritory**: if a node finds a neighbor that has already added it as its territory member then it only updates its territory parameters with the neighbor's.
- **joinNeighborTerritory**: if a node is already part of a smaller territory than the neighbor, and the neighbor has a vacant place (if the territory size is less than MAX_SIZE), the node joins the neighbor's territory.

If any of the above actions are executed then the neighbor's territory members are appended to the territory list.

Algorithm 1 TERRITORY_OPERATION

Input: nghbrTID, nghbrTSize, nghbrTMembers
Output: myTID, myTSize, myTMembers

if $nghbrTSize < MAX\_SIZE$ and $myTSize <= nghbrTSize$ **then**
  if $CPAboveThreshold(nghbrTMembers)$ **then**
    if $myTID == DEFAULT\_ID$ and $nghbrTID == DEFAULT\_ID$ **then**
      makeNewTerritory() {generate a new territory with a unique identifier}
    else
      if $myAddress \in$ nghbrTMembers **then**
        updateMyTerritory() {update myTerritorySize, myTerritoryID}
      else
        joinNeighborTerritory() {join the neighbor's territory with its territory ID}
      end if
    end if
    addNeighborToMyTerritory(myTMembers, nghbrTMembers)
  end if
end if

Metric: minimizing *maximum predicted delay (MPD)*

In CHIRP, like other DTN protocols, the one-hop delivery delay has three components. First, the time a packet has to wait for the next contact to be available (contact wait time). Second, the wait time in queue (queuing delay) due to packets ahead in queue and third the transmission delay. The end-to-end delivery delay largely depends on the contact wait time therefore we devise our metric (MPD) based on contact wait time and ignore queuing delay and transmission delay. We use the beacon advertisement as a utility to predict the delivery delay of a packet.
We argue that in a receiver-initiated transmission scheme each received beacon is an opportunity to transmit data packets. Without loss of generality we assume that the radio link is symmetric and the bird stays within the transmission range after sending the beacon. The MAC layer incorporates an automatic repeat request (ARQ), which

informs the network layer to stop transmitting further packets when the link quality becomes poor or birds move out of communication range. The number of packets transmitted depends on the sender's queue size, link quality, contact duration and node's energy buffer.

Let us assume that in a given time window ($T_{delayCalc}$) a node X received *n* beacons from a node Y, then ignoring the queuing delay, the maximum contact wait time for a packet at X to be transmitted to Y is $\frac{T_{\text{delayCalc}}}{T_{\text{beacon}} \cdot n}$ beacon intervals. In the same way all nodes in the network calculate their one-hop contact-wait time to all their neighbors. Let us for the sake of convenience assume that the network is connected at a given time, then if the contact-wait time for all hops, and for all paths for a given source and destination pair are added then we get a set of paths with multiple end-to-to contact-wait times. Intuitively, the path on which the maximum number of contacts happen is the one with the minimum end-to-end contact wait time, and hence, it is the path over which a data packet can be delivered with minimum delay. Unfortunately, the assumption is not valid for our network (the links would be disconnected at a given time) therefore we rely on the last recorded contact-wait time and predict the maximum delay as *maximum predicted delay* (MPD) for each hop. The end-to-end delay is taken as the minimum of all MPDs for all paths for a source-destination pair.

Due to energy or buffer insufficiency a node may not want to participate in routing despite there being a route through it. Therefore we assign a binary delivery utility *D* ($MPD_{threshold}$, $P_{threshold}$, $B_{threshold}$) where $MPD_{threshold}$ is an upper bound on the maximum allowed delivery delay to be considered as a route, $P_{threshold}$ and $B_{threshold}$ are power and buffer thresholds respectively.

Delivery utility $D_{X,Z}$ of a source node X to a destination Z is one when $MPD_{X,Z}$ is less than $MPD_{threshold}$, and current energy level and buffer space are above the respective threshold, otherwise it is zero. CALCULATE_UTILITY (Algorithm 2) explains how delivery utility *D* for a source node X to destination Z is calculated (here only $MPD_{threshold}$ is used). Each node maintains a neighbor set with all necessary information of which the received number of beacons (*rbCount*) and the last broadcasted MPD of the neighbor is used in CALCULATE_UTILITY. On every neighbor encounter the respective *rbCount* is incremented. CALCULATE_UTILITY resets the counter *rbCount* for all neighbors.

Node roles

To control the hybrid nature of CHIRP we assign roles to nodes, which i sbroadcasted with the beacons. Any node receiving the beacon decides to either replicate, forward or do nothing based on its role and the beacon sender's role.

- **Default.** At initialization all nodes are associated with a default single member territory and a default role. No routing action is done when the node is in this role.

- **Unique.** The node has formed or joined a territory with a unique identifier. The routing action is *replication* in this role. All members belonging to a territory replicate packets each time they detect each other during beacon broadcast. To increase data redundancy in networks formed by extremely territorial birds, which is very sparse, a member of one territory also replicates packets to a member of another territory on contact. This functionality would be enabled at the time of deployment according to the species, we call this variant of our protocol as CHIRP-RoT (Replicate to other Territory) in this case.

Algorithm 2 CALCULATE_UTILITY

Input: neighbors set N containing n number of neighbors. Each element is a tuple <rbCount, MPD> (number of beacons received in $T_{delayCalc}$ (rbCount) and MPD of neighbor to destination Z)
Output: delivery utility $D_{X,Z}$ of source node X to destination Z

Initialization: $MPD_{Z,Z} = 0$
for all neighbors Y in N **do**
  $MPD_{X,Y} = \frac{T_{\text{delayCalc}}}{T_{\text{beacon}} \cdot rbCount_{\text{Y}}}$ {Calculate MPD from X to neighbor Y}
  $MPD_{X,Z_{\text{Y}}} = MPD_{X,Y} + MPD_{Y,Z}$ {Calculate MPD from X to Z via Y}
  $rbCount_{Y} = 0$ {Reset the received beacon count}
end for
$MPD_{X,Z} = \min(MPD_{X,Z_1}, MPD_{X,Z_2}, ..., MPD_{X,Z_n})$
Delivery utility D_{XZ} is calculated as

$$D_{\text{X,Z}} = \begin{cases} 1 & \text{if } MPD_{X,Z} < MPD_{threshold} \\ 0 & \text{otherwise.} \end{cases}$$

- **Path.** The node has found a direct or indirect path to the sink. In this role routing action is switched to *forwarding* if the delivery utility (D) generated as the output of (Algorithm 2) is one. The neighbor node through which the delivery utility is one is chosen as the next forwarder node.

Chapter 4

Implementation

In the following chapter we explain the protocol stack of our communication scheme. We explored available wireless sensor network frameworks, which provide a simulation environment and are able to be ported on our hardware platform. TinyOS [29] and Contiki [30] are commonly used WSN operating systems, but these are not supported on our hardware platform. We have chosen MiXiM [31] as the implementation and simulation framework. MiXiM does not rely on any underlying operating system and the event-driven and layered structure helps to utilize the simulation implementation to build the firmware on the hardware platform. MiXiM is explained in detail in Chapter 5 (Section 5.2). In Section 4.1 the underlying link layer is explained followed by the protocol CHIRP implementation explained in Section 4.2.

4.1 Link layer

The Medium Access Control (MAC) is based on the IEEE 802.15.4 standard. The choice is mainly due to the limitation of the radio module (Section 1.1) and is not a design outcome. We have taken an open source implementation of IEEE 802.15.4 known as OpenMAC [32] and stripped off not required functionalities to convert it into an *non-beacon enabled unslotted CSMA-with-acknowledgment* protocol. A maximum of three retransmissions is possible and if no acknowledgment is received even after this then retransmission failure is reported to the network layer by the event EV_RETRANSMIT_FAIL. If an acknowledgment is received within the maximum retransmission attempts then an event EV_PKT_SEND is sent to the network layer. The network layer sets policy for radio duty cycle and manages synchronization.

The IEEE 802.15.4 standard specifies the maximum physical protocol datagram unit (PPDU) as 127 bytes and the maximum frame overhead (physical and MAC) as 25 bytes, which leaves 102 bytes of MAC service datagram unit (MSDU). CHIRP uses the short addressing (16-bit source and destination address) feature of the standard that increases the size of MSDU to 110 bytes.

4.2 Network layer: CHIRP

CHIRP is explained in Figure 4.1. To implement the receiver-initiated transmission each node wakes up after a fixed time duration T_{beacon} and remains awake for a fixed

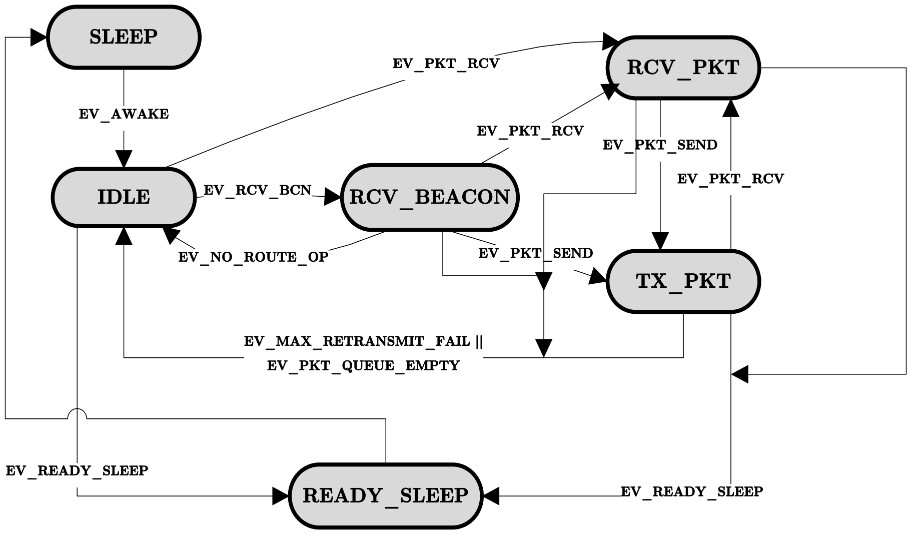


Figure 4.1: CHIRP state machine.

duration $T_{RxWindow}$. After wake-up event EV_AWAKE, each node broadcasts a beacon anticipating that some neighbor will receive the beacon and establish contact with it. It is essential for an opportunistic routing scheme like CHIRP that all nodes wakeup at the same time so that contacts can be established. For this scheme to function we require that all nodes have their local clocks synchronized. Fortunately, the sensor device is equipped with a GPS receiver, which allows the local clocks of nodes to be synchronized with GPS's absolute clock. However, it is not always possible to receive a GPS update because the birds may be within a canopy or there may not be enough power available. To counter this we estimate the worst case clock drift and put a guard intervals at both ends of wakeup and sleep. Another important aspect is the broadcast nature of the beacon, which can be lost due to collisions, yet the sender node would assume that there is no neighbor within communication range. CHIRP does not mitigate lost beacons and relies on the MAC layer's capability to avoid beacon collision.

4.2.1 Timer-based events

Timer-based events trigger independently of network events. Timer-based events are used to calculate protocol parameters and to initiate and halt network communications. CHIRP uses four timers for respective purposes:

$\mathbf{T}_{beacon}$. A periodic beacon timer is set that activates the network layer by triggering an event (EV_AWAKE). After this event the MAC is set to awake state.

$\mathbf{T}_{RxWindow}$. To control the radio duty cycle CHIRP sets a policy for MAC awake and sleep. $T_{RxWindow}$ is the duration of time for which the receiver is allowed to be on.

T_{decay}. The contact probability decay interval is chosen according to the bird under observation. For colonial birds where the chances of contact are higher a shorter decay interval will suffice typically 1/2-1 hour. However, for territorial birds where contacts are less likely, a moderate decay time duration is chosen, typically 2-4 hour.

$T_{delayCalc}$. The delivery utility is calculated only based on MPD. The intuition behind selecting an adequate $T_{delayCalc}$ is to utilize the regular foraging behavior of birds. One of the main reasons for movement of birds is the search for food, which CHIRP utilizes to predict future encounters. Usually diurnal birds forage during daytime and nocturnal birds forage at night. The chances of birds meeting each other is higher during foraging period of the day. $T_{delayCalc}$ should be chosen as the average time a bird takes to go for foraging, return back to the nest and fly again for food. The assumption of collaborative foraging or preferential place or direction for foraging implies that the birds would meet again the next time they go for foraging. For example for colonial birds the average duration between two consecutive departures for foraging is 1-2 hour and for territorial birds it is in the range of 2-5 hour in their respective foraging time of the day.

Network events are not preempted by timer-based events (decayContactPrbability and delayCalc), rather a corresponding flag is set. In the READY_SLEEP state the flags are read and the corresponding functions are executed. Additionally, READY_SLEEP state also performs a post check, it waits for an additional time (based on time required to get a retransmission failure) before going to sleep so that any in progress data packet sending or receiving when the event EV_READY_SLEEP was trigerred, is completed.

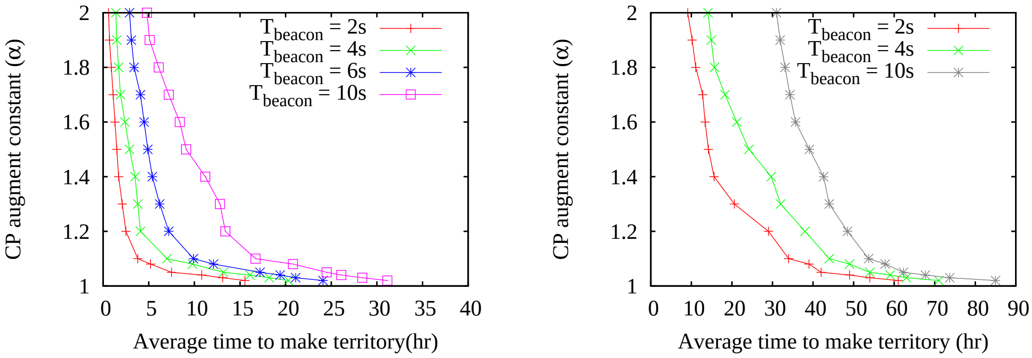
4.2.2 Network events

CHIRP starts with the event EV_AWAKE. After activation, firstly the MAC is set to wake state, secondly the receiver on timer $T_{RxWindow}$ is set and lastly the node broadcasts a beacon. The node waits idly for one of the two events; receive a beacon (EV_RCV_BCN) or receive a data packet (EV_PKT_RCV). On receiving a beacon the sequence of actions taken is:

Add neighbor. Each node maintains a list of all encountered neighbors in a *neighborList* with the following fields:

- *address*: the network address of the neighbor node.
- *CP*: contact probability of the node to neighbor.
- *lastRcvBeaconSequenceNumber*: sequence number of the last received beacon.
- *rcvBeaconCount*: a counter to count the number of beacons received.
- *MPD*: The maximum predicted delay of the neighbor to the sink.

  On every encounter the neighbor list is updated with the neighbor's information. A neighbor is purged from the neighbor list if its CP reaches the minimum threshold $CP_{Threshold}$.



(a) P_{INIT}=0.1, $CP_{Territory}$=0.5, β=0.99, and T_{decay}=1hr (b) P_{INIT}=0.1, $CP_{Territory}$=0.5, β=0.999 and T_{decay}=4hr

Figure 4.2: Effect of α and T_{beacon} on average time to make a territory in a (colonial bird) and b (territorial bird).

Territory operation. The territory operation is implemented as mentioned in Section 3.2.3. The maximum territory size (MAX_SIZE) is significant for colonial birds where the nests are in close proximity and many birds could be in contact with each other, whereas for territorial birds the maximum territory size is governed by the number of birds dwelling in a nest. Figure 4.2 shows the effect of contact probability update constant α and beacon interval T_{beacon} on the average time to form a territory in scenario 1 (colonial birds; Figure: 3.2(a)) and scenario 2 (territorial birds; Figure: 3.2(b)). The territory creation and join operation should not be too opportunistic, sufficient time should be provided for territories to form. Ideally a time duration of a day should be allowed for the territories to stabilize.

The territory parameters (TerritoryID, TerritorySize and TerritoryMembers) required by territory operation are sent within the beacon (Table 4.1). To provide a unique identifier for a newly formed territory, a random value is created from the node's address, the neighbor's address and the current time.

If more than one node receives a beacon and decides to join the territory of the sender simultaneously then more than MAX_SIZE members would join the same territory. As CHIRP identifies territories in a completely distributed fashion such scenarios can happen. CHIRP does not use a leader to manage the territory size and our prime goal of identifying territories is to limit replication therefore we do not attempt to mitigate this at this stage of the project and leave this for future work.

Set node role. The hybrid behavior of CHIRP is controlled by assigning roles to nodes. A routing action is determined dynamically depending on the node's role and the neighbor's role. There are three roles (Default, Unique and Path) electable by a node. Figure 4.3 explains the node-role state machine. Node-role is a bit-field, where the lower order three bits are used as one bit flags for the three roles respectively (starting from the least significant bit). All nodes in the network are initialized with DEFAULT role (Role = 001). After joining a territory the node transits to node-role with unique identifier (UNIQUE) (Role = 011). When a node encounters a neighbor with a path to sink and delivery utility (D) equal to one then the node takes the role (PATH). The role value is set depending on the previous state as either 101 (transition from DE-

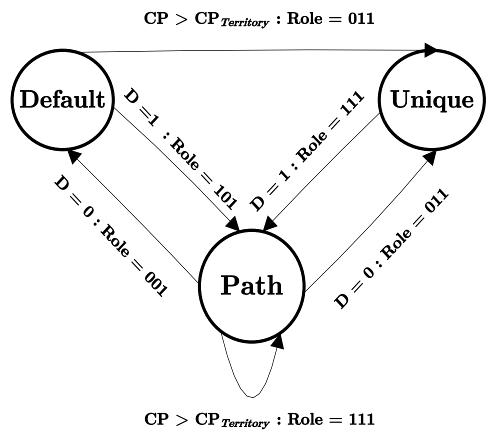


Figure 4.3: Node-role state machine.

FAULT to PATH) or as 111 (transition from UNIQUE to PATH). Based on behavioral information (Section 2.1), once formed, territories stay stable over time and do not change until the birds decide to migrate therefore no transition from state UNIQUE to state DEFAULT is taken. The delivery utility (D) would become zero if the neighbor's prediction is false or the resources are not sufficient therefore the node-role of the node transits back to the role by setting the PATH flag in node-role as false. In this way the node may return to either DEFAULT state or UNIQUE state.

Routing mode. There are three routing modes in CHIRP (replication, forwarding and no operation). The routing mode is decided based on the node's role and the neighbor's role.

- **Replication** (EV_RPL). This mode is selected if the node and neighbor are in UNIQUE state. Subsequently, replication is done within the territory (CHIRP) or with members of other territories (CHIRP-RoT). There are two components in replication that can lead to severe resource usage. Firstly, the number of replicas of a packet present in the network and secondly, the packet identifiers exchanged to inform the sender about the packets that are not available at the receiver. CHIRP attempts to minimize both, the number of replicas of a packet and information exchanged. This is done by the territory identification feature of CHIRP. The logical territory is a single-hop cluster, where a member node can communicate directly to other members. A node replicates data packets to only its members therefore the number of replicas is bounded by MAX_SIZE. However, in implementation it is not always true due to the territory operation where more than MAX_SIZE members can join a single territory. A node is only allowed to replicate its own data packets leading to only one source of packets within the territory. Therefore by just sending the last received sequence number of a received data packet, a source node knows from where to start replicating packets to the neighbor. Figure 4.4 explains the replication process by taking an example territory with 3 members.
  Another issue with replication is stale copies of a packet present in the network although, the packet is already received at the sink. In literature, one of the

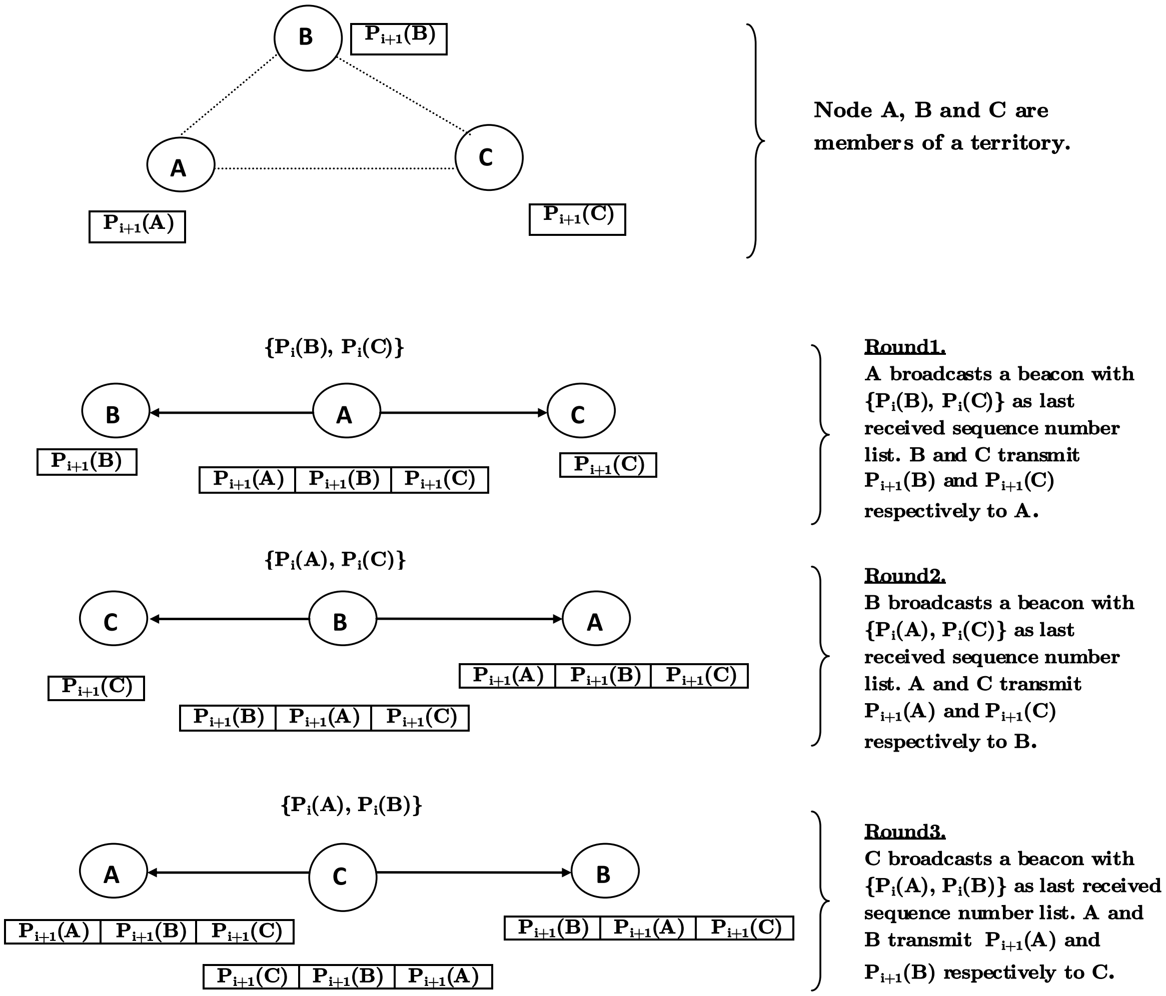


Figure 4.4: Replication process explained in a territory of 3 members (A, B, C). Nodes A, B and C have new packets P_{i+1}(A), P_{i+1}(B) and P_{i+1}(C) respectively. Before starting the beacon rounds it is assumed that packets P_i(A), P_i(B) and P_i(C) have already been replicated to all members. Each node only replicates its source data packet on every EV_RPL action. In this example, in three rounds of beacon reception each member has a single copy of the new packets of all territory members.

techniques commonly used to purge stale packets is by disseminating acknowledgments of received packets. CHIRP attempts to minimize the number of stale packets by disseminating the sequence number of last received in-sequence data packet at the sink. Due to the limitation on beacon size this cannot be done for all nodes in the network.

- **Forwarding** (EV_FWD). This mode is selected if the neighbor and the node satisfy the following conditions respectively:
  The neighbor is in state PATH with its delivery utility (D) set as one.
  The CALCULATE_UTILITY (Algorithm 2) has returned the delivery utility as one for this neighbor node, thus appointing it as the next forwarder node.

- **No operation** (EV_NO_ROUTE_OP). This mode is selected if the node decides not to transmit data packets. This can happen in four cases; first, the node is in DEFAULT role, where no transmission is done. Second, CHIRP only allows data packets to be replicated within its territory and if the neighbor is from another territory then no transmission is done. However, in case of CHIRP-RoT this becomes a replication action. Third, the neighbor node is not the next forwarder node according to delivery utility. Fourth, the neighbor node is the next forwarder node, but it has set its delivery utility to zero in the beacon, which means that its buffer or energy is less than the threshold.

4.2.3 Packet structure

There are two types of network packets, control packets and data packets. In this implementation of CHIRP where we have assumed the link as bidirectional there is only one control message i.e. the beacon. In other cases handshake messages would also fall in this category. The beacon structure is shown in Table 4.1. The beacon is limited to the maximum available MSDU size i.e. 110 bytes. The source address is a 16-bit network address and the destination is the network broadcast address (0xFFFF). The other fields of the beacon packet are command identifier, new value, territory size, identifier, members and list of sequence numbers. The territory member list and the sequence number lists are variable-size fields of the beacon. The territory member list and the respective last received sequence number list varies between 6 (address(2) + sequence number(4)) bytes to (MAX_SIZE $\cdot$ 6) bytes. The rest of the beacon packet can be used to send sequence numbers of last received in-sequence data packets at the sink. One of the improvements we will look into in the future is to compress the list fields (territory members and sequence numbers) of beacons to reduce the size of beacon packets, which would reduce the overhead incurred by CHIRP.
The data packet is shown in Table 4.1. The source and destination addresses are 16-bit nodes' addresses, the sequence number is a 32-bit unsigned integer and the maximum data payload is 100 bytes. The other field in a data packet is the time-to-live (TTL). The sequence number uniquely identifies the data packet generated at a node. It is used by CHIRP for controlling replication and removing stale copies of packets from the queue. The size of the network packet header is 10 bytes (4 bytes of addresses, 4 bytes of sequence number and 2 bytes of TTL)

4.2.4 Additional features of CHIRP

Multiple sinks. Three of the higher-order network address bits are reserved for sink addresses envisaging a need of multiple sinks. The algorithm CALCULATE_UTILITY

(a)

Beacon fields	**Size (bytes)**
SourceAddress	2
DestinationAddress	2
BeaconSequenceNumber	4
CommandIdentifier	1
NewValue	1
MPD	2
DeliveryUtility(D)	1
NodeRole	1
TerritoryID	1
TerritorySize	1
TerritoryMembersList	[2, (2 · MAX_SIZE)]
LastRcvSequenceNumberList	[4, (4 · MAX_SIZE)]
LastAckSequenceNumberList	MSDU - rest fields

(b)

Data fields	**Size (bytes)**
SourceAddress	2
DestinationAddress	2
SequenceNumber	4
TTL	2
DataPayload	100

Table 4.1: CHIRP packets structure . (a) explains the various fields and respective sizes (bytes) of a beacon packet. (b) explains the data packet fields with respective sizes (bytes).

(Algorithm 2) does not change in case of multiple sinks. The algorithm would work as; the multiple sinks are considered as a class of destinations meaning that any of the sinks is a possible destination where any packet can be transmitted. The class address is reserved as (CLASS_SINK_ADDRESS) that all nodes enter as the destination address. All sinks advertise their MPD as zero. As the metric is based on choosing the minimum MPD among all neighbors, CHIRP does not care about path to which of the deployed sinks is chosen. On receiving a data packet, a sink accepts it if the destination address of the packet is set as (CLASS_SINK_ADDRESS). Duplicate packets can be received at multiple sinks, which CHIRP does not care to remove.

TTL. Routing loops are inevitable in a routing protocol and CHIRP is not exempted from this. We do not use any route loop detection technique and adopt a time to live (TTL) based loop mitigation. Each source data packet is initialized with a TTL count that is decremented on every replication or forwarding action. When the TTL reaches zero it means that the packet is trapped in some loop and it is dropped from the network. In traditional routing schemes TTL is chosen based on the network diameter estimate. In BirdTracking, the TTL is set as $(\frac{Network\ size}{MAX\_SIZE} + 1)$, which means that the packet has been replicated or forwarded to at least one member of all territories in the network.

Command update. To dynamically change the sensor intervals a command identifier field and corresponding value field is present in the beacon packet (Table 4.1). The command update is disseminated in the network by the sink. The beacon sequence number informs about the recent command update send by the sink.

Chapter 5

Experiments and Results

To evaluate the performance of CHIRP we want to conduct experiments both in simulation and in the field by implementing it on BirdTracking sensor devices (Section 1.1). Due to lack of time we are performing our experiments only in a simulation environment and leave deployment testing for the future. We have compared CHIRP with two reference protocols, *direct transmission* and *epidemic routing*. In the rest of the chapter the simulation setup is explained with a comparison of protocols on various metrics like yield, delay and resource utilization.

5.1 Reference protocols

We have chosen two reference protocols to compare against CHIRP on various performance metrics. First, a direct transmission scheme, where a source node transmits data packets directly to the sink. Direct transmission is a single-hop data transmission scheme that has been used to get the traces for the two scenarios (gulls and honey buzzards). Second, an epidemic scheme, which is based on replication and propagation of copies of a packet to many mobile nodes. This scheme does not rely on an underlying mobility model and replicates packets irrespective of any knowledge of future contacts. Epidemic routing has shown good data delivery and has been used by most of the DTN routing protocols as a reference. In our comparison we use a scaled-down version of the epidemic protocol, where only the last received sequence number of the data packet is used as exchange information (the actual epidemic scheme uses a bloom-filter for exchanging information).

5.2 Simulation setup

For simulation tests we have chosen the MiXiM wireless network simulator. The simulation environment is setup such that the simulation network resembles the network in a field deployment.

5.2.1 MiXiM

MiXiM [31] is a discrete event-based simulator targeted for mobile and wireless networks. In MiXiM, every event occurs at some discrete moment in time and events are processed in a chronological order. In MiXiM, various layers of the Open Systems

MAC layer parameters	Values
Bit rate	250 kbps
Min backoff exponent	1
Max backoff exponent	6
Max CSMA backoffs	20
Unit backoff period	0.02 s
Max frame retries	3
MAC queue size	1 frame

Table 5.1: MAC layer parameters used in simulation

Interconnection (OSI) model can be modeled as modules that provides services to the upper and lower layers through two types of channels, a *data channel* for data signals and a *control channel* for control signals. Each node is constituted from various components; some that are already part of the framework like the physical channel module, mobility module, application module etc., and custom-built modules like CHIRP. The various modules used in simulation are explained as below:

- **Channel module**. A simple pathloss model with pathloss exponent γ = 3.5 and carrier frequency as 2.4 GHz is used in all simulation runs. The signal decider module decides to receive the signal based on a *signal to noise ratio* (SNR) threshold and a bit-error-rate lower bound (berLowerBound).

- **MAC**. The MAC layer is a beaconless unslotted CSMA/Ack with most of the parameters taken as defined in the IEEE 802.15.4 standard. Some of the interesting parameters are listed in Table 5.1.

- **Mobility model**. The real life position traces of birds for a duration of one month are modeled with the Bonn-motion mobility model of MiXiM. The mobility model only supports two-dimensional mobility, therefore only x and y coordinates are used. In scenario 1 (colonial bird), the positions are recorded every 10 minutes and in scenario 2 (territorial birds) a new position is taken once every 20 minutes. The positions are stored in a vector file with respective timestamps. The Bonn motion model reads a tuple (t_{new}, x_{new}, y_{new}) from the trace file and sets the direction of movement from the current position (x, y) in a straight line towards the target with a speed of $\frac{distance(x_{\text{new}},y_{\text{new}},x,y)}{t_{\text{new}}-t}$. In the simulation each new position is updated at an interval of 0.1 second with the initially calculated speed, which is in the range of 0-15 m/s. This in not the real life movement of birds as they tend to follow a random motion from the prior position; however, our model is close enough to replicate the movement of birds in a simulation environment.
The movement trace is available for only one type of colonial bird (gulls) and one type of territorial bird (honey buzzards). In both cases the number of birds is 6 (3 pairs). We want to conduct our simulation experiments on more birds in both scenarios (colonial and territorial) so that the performance of protocols is tested with adequate network size. A network of birds is not going to be scaled to very large number due to practical limitations, which is catching birds and tagging individual birds. Considering these limitations we generate traces for more birds (30 colonial birds and 18 territorial birds) by using the available traces

while keeping the behavioral nature intact. For example, the inter-nest distance for colonial birds is kept within the range 10-50 meters and for territorial birds the inter-nest distance is between 1-3 kilometers.

- **Traffic**. The sensors on board generate two types of traffic. First, the GPS, pressure and temperature sensors are recorded every several minutes (4-10 minutes) resulting in a non-bursty constant rate traffic. Second, the accelerometer which is typically read every several seconds (3-5 seconds) for several minutes generates a bursty traffic flow. We are not adding accelerometer data into the traffic in this experiment and consider only a non-bursty constant rate traffic generator. In CHIRP, the MAC layer provides a MAC service datagram unit (MSDU) of a maximum of 110 bytes (127 bytes PPDU - MAC and PHY headers), which results in a data payload of 100 bytes (110 bytes MSDU - NETW headers). For our simulation experiments the message size is taken as 100 bytes, which is same as used in the scheme to collect traces (we would improve the data gathering and packetization in later stage of the project). For scenario 1 (colonial bird), the traffic rate is taken as 12 pkts/hr and for scenario 2 (territorial bird) it is taken as 6 pkts/hr.

- **Radio transmission power**. The hardware radio module is capable of transmitting at a maximum transmission power of 3 dBm (Section 1.1). The transmission power is largely responsible for the achieved radio communication range, which affects the performace of CHIRP as well as the reference protocols. Considering a uni-disk model (pathloss exponent as 3.5) and the radio module details (Section 1.1) the radio range is calculated as 70 meters. In our simulation we have kept the transmission power as constant and so the radio range, however the impact of radio range is discussed wherever applicable.

5.3 Experiments

In order to simulate CHIRP we have used the two scenarios; a colonial bird (gulls) and a territorial bird (honey buzzards). There are some parameters that can be varied whilst some others are fixed.

Fixed setup. The simulation mobility model, which is created from real life traces is fixed. This means that in all repetitions of the simulation tests the position of a bird at a given simulation time is the same. Additionally, the model puts a limit on the total number and duration of time two birds are within the radio communication range. The radio module decides whether to establish a radio communication or not based on the channel model. The contact probability constants are kept fixed. For colonial birds the constants are chosen such that the territory is formed in 12 hour whereas for territorial birds the constants are chosen to form a territory in 24 hour. $CP_{Threshold}$ is taken as 0.5 in all scenarios. The $T_{delayCalc}$ is taken as 2 hour for scenario 1 (colonial bird) and 5 hour for scenario 2 (territorial bird). The MAC layer parameters (Table 5.1) are fixed in all simulation runs for all protocols except the duty cycle policy, which is set by CHIRP. The maximum territory size is kept constant at four in all simulations and for all scenarios. The radio transmission power is taken as 3 dBm (radio range of 70 meters) in all experiments.

Variable setup. In simulation, the obtained number of contacts is influenced by the beacon interval T_{beacon}. The available bandwidth is governed by two factors; first the mobility of birds and second the radio-on duration ($T_{RxWindow}$).

In traditional routing schemes congestion in the network is mainly caused by the traffic and the bandwidth. However, in DTN, the communication duration between nodes also plays a role in network congestion. We define the network load as the relative ratio of traffic generated to the contact volume as

$$\text{Load} = \frac{\textit{Total traffic generated}}{\textit{Total contact volume available}}$$

Contact volume is the product of total contact duration (including the contact with the sink) and the capacity available at the network layer. Contact duration is the duration of time when two nodes are in contact with each other. The total contact duration is the sum of all contact durations for all pair of nodes in the network. In this experiment we have taken fixed rate traffic as mentioned in Section 5.2. The physical link bandwidth is 250 kbps, but this is not achieved at the network layer. Based on the MAC parameters (Table 5.1) and IEEE 802.15.4 standards, we calculate the achievable transfer rate as approximately 100 kbps. We have already assumed in Section 3.1 that the transfer rate is the same for all nodes (neglecting the impact due to interference). Therefore the contact volume only depends on the achievable contact duration.

Load is not able to capture the fact that on the one hand a packet may have to travel multiple hops thus reducing the effective usable bandwidth whereas on the other hand there is no traffic demand, though the contacts are available. Inspite of these limitations, the above ratio provides some useful insight on the relationship between scenario parameters and protocol performance. Additionally, load would be used to evaluate the performance of the routing protocols when the accelerometer data is also added to the traffic. This results in the load depending on the two factors, one being the contact duration and other being the traffic demand.

In literature of DTN routing the total contact duration is usually dependent on the mobility of nodes and radio range. In BirdTracking, we go a step further and also add radio duty cycle for power conservation. Therefore total contact duration depends on radio range, mobility of birds and radio-on duration. As far as this experiment is concerned we do not vary the radio transmission power (we take the maximum transmission power provided by the radio module) and hence the achieved radio range is fixed (considering a uni-disk model). To calculate contact duration we perform simulation experiment for both scenarios (colonial and territorial birds). We first start with establishing a contact (a node receives a beacon from a neighbor) and from then the contact duration is calculated using periodic probe messages. The beacon interval (T_{beacon}) influences the number of contacts achieved. The higher the beacon interval, the lower the number of obtained contacts and vice-versa. In Figure 5.1, it is shown that for a given radio transmission power (3 dBm in this case) the average number of contacts per bird per hour decreases as T_{beacon} increases for both scenarios. The radio range would also affect the number of contacts. If the radio transmission power is increased for a given T_{beacon} the average number of contacts would increase. In the same experiment (done to calculate the number of contacts mentioned in Figure 5.1) the achieved contact duration is calculated with varying $T_{RxWindow}$ starting from 1 second till the T_{beacon} (100% radio duty cycle) for various beacon intervals. In particular, after receiving a beacon from a neighbor, a node starts to send probe messages

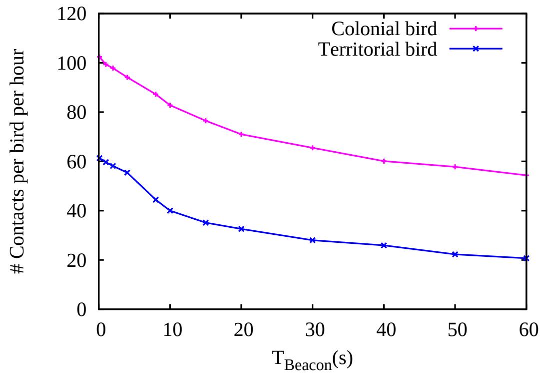


Figure 5.1: Impact of beacon interval on number of contacts per bird per hour.

periodically (the period is chosen based on round-trip-time with maximum number of retransmissions) until either a retransmission failure happens (the neighbor node is out of range now) or $T_{RxWindow}$ expires. It is observed that for a fixed T_{beacon} increasing the $T_{RxWindow}$ results in an increase in achieved contact duration. One of the interesting findings in this experiment was that there is some overlap between total contact durations available with two different values of T_{beacon} and $T_{RxWindow}$. This is possible as sum of contact durations calculated using a smaller T_{beacon} (more contacts) with a smaller $T_{RxWindow}$ could be same as the one calculated using a larger T_{beacon} (fewer contacts) and a larger $T_{RxWindow}$. To argue on which of the two approaches is better we take help of the number of contacts. As T_{beacon} increases the average number of contacts decreases, therefore a smaller beacon interval is better in capturing the dynamics of moving birds.

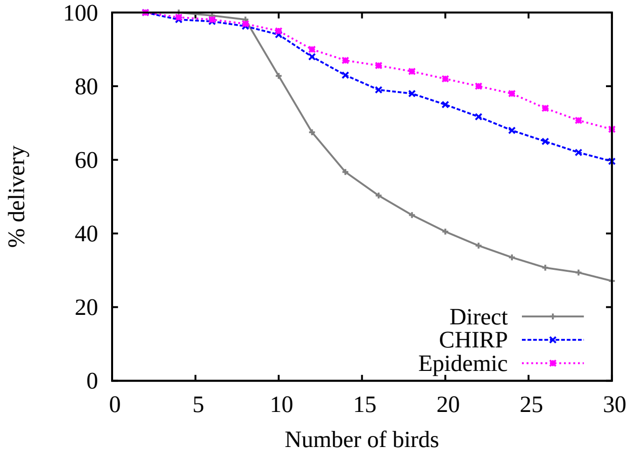
5.3.1 Delivery

We compare CHIRP, direct transmission and the epidemic routing protocols for data delivery against the number of birds and load for both scenarios. Delivery percentage is defined as the percentage of data packets received at the sink at the end of the simulation to the number of data packet generated in the network.

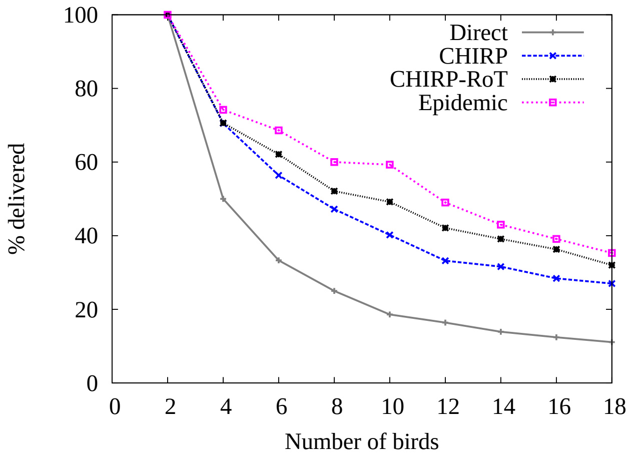
Network size

In this experiment the buffer size is taken as infinite, however bandwidth is restricted due to mobility and radio-on duration.

Colonial bird. We run our simulation experiment on 30 colonial birds. Only one sink is used that is placed near the nest close to the center of all nest locations. Figure 5.2(a) compares delivery ratio at the sink for the three protocols at load = 0.001 (T_{beacon} = 4 s, $T_{RxWindow}$ = 1 s). We increase the number of birds in pairs, where the two birds are partners. The first 8 birds have their nest close to the location of the sink and therefore they are able to deliver almost all ($> 95\%$) packets in all the three protocols. Subsequent birds have their nests out of the radio range of the sink however, in some instances the birds fly close to the sink getting an opportunity to transmit data packets



(a) Colonial bird (gulls)



(b) Territorial bird (honey buzzards)

Figure 5.2: Effect of network size on delivery at load = 0.001 (T_{beacon} = 4 s, $T_{RxWindow}$ = 1 s)

directly. The delivery percentage for direct transmission scheme drops rapidly as the number of birds is increased. Both CHIRP and the epidemic schemes manage to keep the delivery significantly above the direct transmission, but none are able to achieve a 100% data yield due to insufficient contact volume to move data to the sink before the end of simulation. CHIRP is able to achieve a 120% increase in data delivery over direct transmission however, it achieves 15% less delivery than the epidemic scheme.

Territorial bird. We run our simulation experiment on 18 territorial birds making 9 nests with two partner birds dwelling in each nest. The sink is placed near one of the nest locations. According to the behavior of territorial birds no bird enters any other birds' territory (mating-and-nesting territory for honey buzzards). One of the partner always flies around the nest in a range of a kilometer leaving only the other partner to act as a relay node. Therefore the contact opportunities in case of territorial birds are

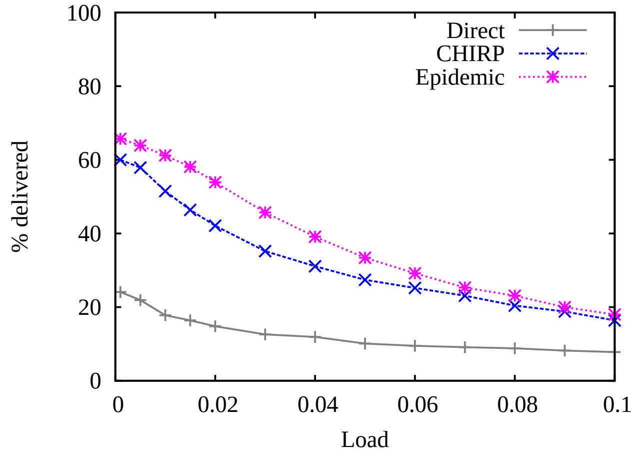


Figure 5.3: Effect of scaling load (by decreasing contact volume) on delivery for colonial birds

less as compared to the colonial birds. Figure 5.2(b) compares the data delivery with the three protocols in this scenario. In this case one nest constitutes one territory and only one partner goes for foraging and the other partner flies around the nest most of the time. CHIRP achieves a 130% increase in data delivery and CHIRP-RoT achieves a 180% increase in data delivery over direct transmission. CHIRP-RoT shows better results than CHIRP as it allows sharing data packets to birds of other territories. The delivery percentage achieved by the epidemic scheme is 14% more than CHIRP-RoT and 29% more than CHIRP.

Load

We now compare the effect of increasing load on data delivery. The load is increased by decreasing the total contact duration. None of the protocols are able to achieve 100% data delivery due to insufficient contact volume to move data to the sink before the end of the simulation. CHIRP achieves a significant improvement in data delivery against the direct scheme and remains close to data delivery achieved by the epidemic scheme.

Colonial bird. In this experiment the number of birds is taken as 30 with the same sink placement as in the last experiment. Figure 5.3 compares CHIRP, epidemic and direct transmission for various loads in this scenario. The experiment is done from load 0.0008 (T_{beacon}=1s) to 0.1. As data delivery drops below 20% at load = 0.1, subsequent experiments are not done. As expected, the data delivery drops as the load increases (decreasing contact duration) for all the three protocols. The delivery for the epidemic and CHIRP drops significantly beyond a load of 0.01. The rate of delivery drop for the epidemic and CHIRP is larger than that of the direct scheme. This is intuitive, as the load is based on the total contact duration for all birds however, the individual birds which are in direct contact with the sink still manage to push data even at a higher load.

5.3.2 Average delay

Due to the physical limitation of not having enough contact durations, none of the protocols is able to achieve 100% data yield therefore delivery delay is only calculated for the packets which are received at the sink. The number of packets received in all the three protocols are different however, the idea is to evaluate the effect on delay in all cases against various loads. The delay of a packet is calculated as the difference between the time the packet is injected into the network and the time when it is received at the sink. The direct scheme is not used in comparison as in this scheme a packet injected into the network means it is transmitted to the sink instantly.

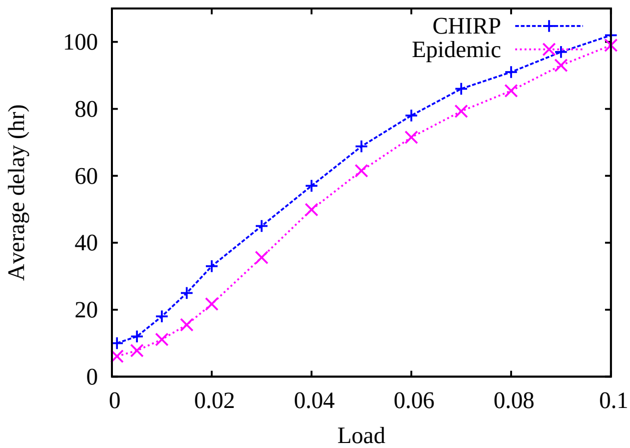


Figure 5.4: Effect of scaling load (by decreasing contact volume) on average delay for colonial birds

Colonial bird. Figure 5.4 shows the effect of increasing load on the average delay in scenario 1 (colonial birds). Due to decreasing contact durations (increasing load) the data packets are queued for long time and it takes more time to reach the sink, therefore the delay increases. There is an inverse relationship between average delay and delivery percentage. For higher loads (> 0.01) the data delivery decreases quite rapidly and therefore the average delay increases equally.

5.3.3 Resource utilization

The cost of the protocols is compared by means of the resources they use to achieve the performance. In DTN, we typically observe the bandwidth, storage and power with the protocol performance.

Bandwidth. For a given load we compare CHIRP and the the epidemic scheme to find out which of the two schemes better utilizes the available bandwidth. It is observed that in case of scenario 1 (colonial bird) at a load of 0.001, a 15% increase in the delivery percentage is shown by the epidemic with a 110% increase in the total number of transmissions (including retransmissions at MAC and network layer) against CHIRP.

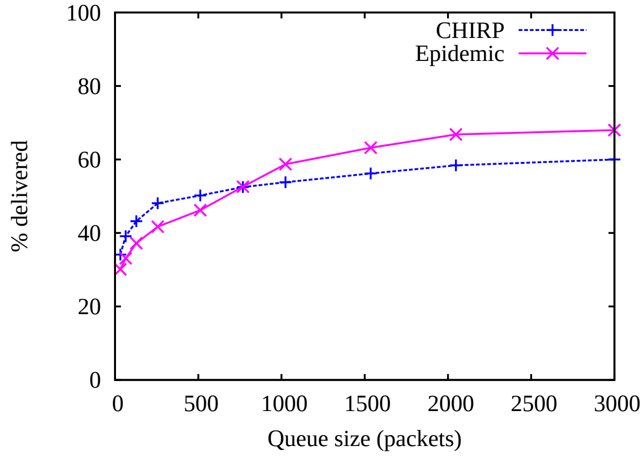


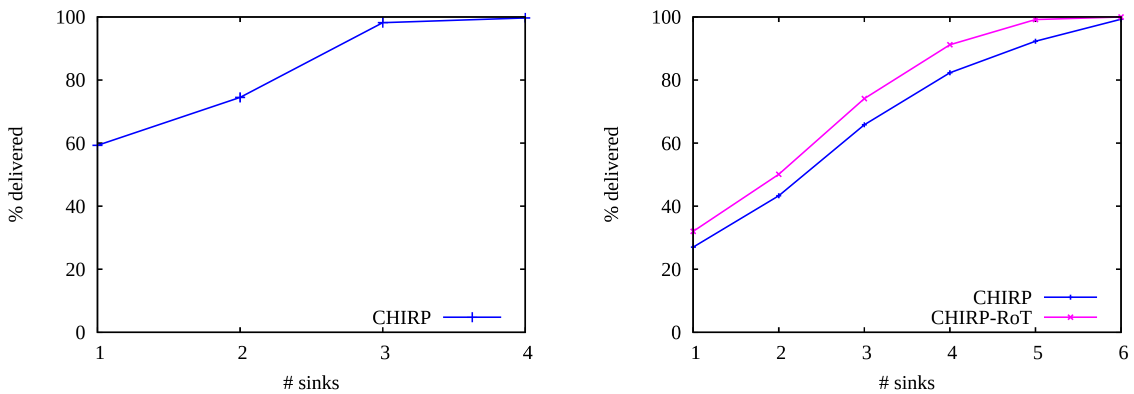
Figure 5.5: Effect of queue size on delivery in case of colonial birds

Buffer. All the three protocols lack knowledge about buffer limitations. Therefore, we expect to see packet loss when buffer space is limited. To explore this we vary the maximum buffer capacity at each node to see how the protocols perform with respect to packet delivery. We only compare the epidemic and CHIRP against buffer space requirement. The smallest queue size is taken as 32 packets (based on the available RAM). Figure 5.5 compares the delivery obtained by the epidemic and CHIRP on scaling queue size. The epidemic scheme is quite sensitive to storage requirement and needs significant buffer space to achieve the desired delivery. In the following experiment a head-drop (the packet at the front of the queue is dropped) policy is used based on the intuition that a packet at the front of the queue would be buffered for the longest time in the queue and possibly other replicas are present in the queue of some other node. CHIRP shows better delivery for smaller queue sizes (less than 500 packets) than epidemic scheme. However, as the queue size increases (larger than 1000 packets) the delivery percentage for epidemic scheme increases. It is obvious from the figure that the ratio of delivery gain with queue size is more for epidemic scheme than CHIRP. Flash memory would be required to be used to queue additional data packets, which means that the space for data telemetry is reduced. Data telemetry is the only form of gathering sensor data if the wireless communication failed due to any undesirable reasons and therefore we would like to use the flash memory for queuing packets as minimum as possible. From our experiment in Figure 5.5 we can say that 200 kilobytes to 300 kilobytes (about 5-7 %) of the flash memory would be adequate to be used for packet queuing for CHIRP.

Power. In our simulation we have assumed that the nodes are synchronized and all nodes stay awake for a fixed duration of time where all sending and receiving happens for all the three protocols. This scheme is not the best one when it comes to power utilization in sensor networks as it leaves room for idle listening. We are aware of this issue and would work on a energy simulation in future to find how much power is wasted on idle listening. For our experiments the power consumed in transmitting and receiving is almost the same (19 mA in TX and 18 mA in RX as mentioned in Section 1.1) therefore the contribution of number of transmissions in overall power consumption is marginal. The radio-on duration is the major power consumer as far as the communication is concerned. However, due to the requirement of a large buffer to store packets another power consumer emerges. As there is a limited available space in RAM to buffer relay data packets, which in our case is 32 data packets, is not sufficient to achieve the desired delivery. We do not want to loose packets therefore the additional relay packets are written to the flash memory. Read and write access from the flash memory are expensive as it consumes significant power, typically 10 mA (according to the data sheet of the serial flash memory). The hardware module is equipped with a 4 megabytes flash. To preserve nodes' own source data and make space for additional packets a queue drop policy would be used for relay packets written in flash memory. In the epidemic protocol the requirement for a larger queue size (Figure 5.5) is critical in obtaining the data delivery and the average queue size is much larger than CHIRP therefore epidemic protocol would consume more power than CHIRP due to more flash read and write operations.

5.3.4 Multiple sinks

In CHIRP, a maximum of eight sinks can be deployed to collect sensor data. Deployment of multiple sinks would increase the data yield with increasing overhead due to duplicate data packets transmissions and post removal of redundant packets. Figure 5.6 shows that the delivery percentage is increased in both scenarios (colonial and territorial) by deploying multiple sinks. In both cases more sinks are placed adjacent to different nests locations. The load in this experiment is 0.001 in both cases. As the data delivery increases the average delay of a packet would decrease. Usage of multiple sinks is also possible for the direct transmission and the epidemic scheme. In the direct transmission one sink near a nest (for territorial birds) or nests (for colonial birds) within the range is ideal. The number of sinks required scales linearly with the number of nests. This is not desirable both in terms of resource requirement and also it is not always possible to locate all nests of the target birds (specially in case of territorial birds).



(a) Colonial bird (gulls)

(b) Territorial bird (honey buzzards)

Figure 5.6: Effect of scaling number of sinks on delivery

5.3.5 Discussion

Our simulation experiments in this chapter show that CHIRP is able to achieve a significant data delivery as compared to the direct scheme (>100%) in both scenarios. Some of the key points found in the experiments are:

- We have seen that the colonial birds are more in contact with each other as compared to territorial birds and hence the data delivery is better in former case than the latter. However, even in the case of colonial birds we are not able to achieve a 100% data yield due to insufficient contacts available to push all data before the end of the simulation. It is shown that in the case of territorial birds that CHIRP-RoT improves the data delivery by allowing replication to birds of other territories. The epidemic scheme is shown to perform better than CHIRP at the cost of consuming more resources. At one end the direct scheme consumes less resources, but performs poor on delivery and suffers from very large delay. On the other end the epidemic scheme is able to achieve decent data delivery with an average delay of 6 hours, but at the cost of enormous resource usage. CHIRP follows a middle path by achieving a delivery of just 15% less than the epidemic scheme with an average delay of 10 hours. CHIRP uses less buffer and power than the epidemic scheme to achieve the desired performance.

- The average delay is quite high (10 hours for colonial birds at minimum load) as compared to the traffic load therefore we cannot achieve real-time data delivery just by using one sink in both scenarios (colonial and territorial). Using multiple sinks increases the data delivery and consequently reduces delay.

- The requirement of buffer space (more than what is available in the RAM) leads to use of the flash memory as an additional buffer space for relay packets. This consumes significant power, leaves less space for the node's own data and incurs additional delay in flash read and write operation. CHIRP is aware of this issue and therefore attempts to limit flash operations by limiting replication and switching to forwarding whenever possible so that the packets do not stay long in the queue.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This research work explores the challenges in developing a wireless sensor network to observe bird life wherein sensor devices are tagged on birds. The mobility and habitat of birds cause disrupted links and network partitions, which results in most of the traditional routing schemes being unsuitable. We identify the BirdTracking network as a delay tolerant network and develop a routing scheme based on assumptions made by studying the behavior of birds. We propose CHIRP, a routing protocol to deliver data to the sink during the nesting and breeding season utilizing the behavioral knowledge of birds. Our findings on birds' behavior suggest that during nesting season birds nests can be clustered in logical territories where each bird shares data with every member of the territory. Furthermore, we believe that the foraging behavior of birds can be used to predict communication opportunities.

We have performed experiments on two scenarios (one colonial bird and one territorial bird) created out of real-life traces of birds. We have compared the protocols on a metric (load), which is better in capturing the congestion in DTN by also considering the contact durations between nodes. The mobility of birds, radio range and the radio-on duration impact the achieveable contact duration, which is related to the protocol performance on delivery and delay metrics. In both scenarios CHIRP outperforms direct transmission scheme by achieving more than 120% increase in data delivery. It is shown that CHIRP is not able to achieve a 100% data delivery because of insufficient contacts to deliver all data packets before the end of the simulation. However, we were aware of such outcomes and therefore provide a facility to deploy multiple sinks. By using multiple sinks data delivery is increased significantly (about 15-20% increase in delivery percentage by deploying one additional sink) and consequently the average delay is reduced. Our experiments also show that the epidemic scheme is able to achieve a better delivery as compared to CHIRP, but at the cost of large storage and power requirements. Our protocol is capable to achieve a delivery that is close to the epidemic scheme while consuming less resources. Power saved by using CHIRP can be used to generate more data, but as the contact volume is not sufficient enough we need help of multiple sinks to increase data delivery. Utilization of flash memory for storing additional relay packets is less in CHIRP when compared to epidemic scheme, therefore more space is allocated to store the node's own data.

We have been able to solve our research questions and demonstrate the results through our simulation experiments. We have shown that using the behavior information of birds multi-hop packet transmission is possible that improves the data delivery at the sink. Furthermore, it also helps to reduce the resource wastage as done by an epidemic routing scheme. We have seen that due to the limitation imposed by the mobility of birds we are not able to achieve sufficient contacts to push data to the sink in less than 10 hours. Therefore our expectation of real-time data delivery cannot be satisfied by just using one sink. Deploying more sinks would reduce the delay in delivering a packet to the sink, but due to the physical challenges (locating nests) it cannot be scaled to large number of sinks.

CHIRP is developed using behavioral information of birds, though it is not limited to only network formed by birds. CHIRP uses replication and forwarding (when contact opportunities are predictable) to route data to the sink, which in this special case is utilizing the behavioral information of birds.

6.2 Future work

In this thesis work we have only focused on the first phase (nesting and breeding) of BirdTracking. However, there are still some open questions to be dealt with in phase 1. We group our future work in the following categories:

Simulation tests. As CHIRP utilizes the behavior of birds to develop the routing strategy we need to perform experiments on more bird species in both categories (colonial and territorial). There are not too many real-life mobility traces available therefore one obvious future step is to create a synthetic mobility model for birds. To achieve this we need to understand the movements of birds in more details. There are various modes of a bird mobility which are characterized by speed, randomness and halt duration. For example in search for food, halt at a food place, revisiting the food place, roosting, migration etc. One of the sensors, the accelerometer, makes the traffic flow bursty. In the present experiment we have assumed the traffic flow as non-bursty, which is neglecting the accelerometer data. This also contributes in increasing the load that would affect the delivery of CHIRP. The territory operation requires some improvement. There could be some instances where the current distributed approach may create a territory of more than MAX_SIZE members. To improve this we need to devise a better scheme by electing a leader of the territory who manages the territory operation. We also need to simulate the energy consumption of CHIRP and attempt to minimize the idle listening caused due to fixed radio-on duration.

Phase 2: migration of birds. After phase 1 (nesting and breeding) we move on to the phase 2 (migration of birds). We know that data delivery at this time of the bird cycle is not possible until the birds complete their trip and return back. We target birds forming collaborative foraging groups in this phase as it provides greater opportunity to do in-network data aggregation and assign rotating duties. The territory operation in CHIRP that was based on co-location of partners or nearby nests would change during migration as now the whole flock is in contact with each other most of the time. One approach is to remove the restriction of the maximum territory size so that all birds in the flock tend to form a single large cluster. Another approach could be to retain the maximum territory size restriction and make a hierarchical cluster.

Firmware development and field testing. We want to test CHIRP in the field by implementing it on the BirdTracking sensor device and tagging on birds under observation. The link layer of the radio module is converted to a non-beacon enabled CSMA/Ack protocol. CHIRP is based on this link layer, but if in future the link layer is changed i.e. if there is a separate MAC used then CHIRP can be modified without much rework. If the MAC layer does not support acknowledgments then these functionalities would be delegated to CHIRP. We need to also do efficient sensor reading and compression to use a minimum number of bytes in a message. Finally a field test would expose the system to a real-world scenario where inputs gathered will be analyzed for a possible improvement.

44

Bibliography

[1] M. P. Johnson and P. L. Tyack, "A digital acoustic recording tag for measuring the response of wild marine mammals to sound," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 1, pp. 3–12, 2003.

[2] "Princeton Wildlife Monitoring Project." `www.princeton.edu/~wikelski/research`.

[3] "FalconTrak." `www.ccb-wm.org/vafalcons/falcontrak/falcontrak_falcontracking.htm`.

[4] "Raptor Research Project." `www.raptor-research.de`.

[5] "FlySafe Bird Avoidance Model." `http://public.flysafe.sara.nl/bambas/index.php`.

[6] J. Shamoun-Baranes, W. Bouten, L. Buurma, R. DeFusco, A. Dekker, H. Sierdsema, F. Sluiter, J. Van Belle, H. Van Gasteren, and E. Van Loon, "Avian information systems: developing Web-based bird avoidance models.," *Ecology and Society 13(2): 38*, 2008.

[7] "IEEE 802.15.4." `www.ieee802.org`.

[8] "Birds of Stanford." `www.stanford.edu/group/stanfordbirds/`.

[9] "Ornithology." `www.people.eku.edu/ritchisong/ornitholsyl.htm`.

[10] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *First ACM Int. Workshop on Wireless Sensor Networks and Application (WSNA)*, (Atlanta, GA), pp. 88–97, September 2002.

[11] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet," in *ASPLOS-X: Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*, (San Jose, California), pp. 96–107, 2002.

[12] J. Link, K. Wehrle, O. Osechas, and J. Thiele, "Ratpack: Communication in a sparse dynamic network," *SIGCOMM'08 Poster*, 2008.

[13] Z. Butler, P. Corke, R. Peterson, and D. Rus, "Networked cows: Virtual fences for controlling cows," in *WAMES 2004*, (Boston, USA), 2004.

[14] B. Pásztor, L. Mottola, C. Mascolo, G. Picco, S. Ellwood, and D. Macdonald, "Selective Reprogramming of Mobile Sensor Networks through Social Community Detection," *Wireless Sensor Networks*, pp. 178–193, 2010.

[15] A. Lindgren, C. Mascolo, M. Lonergan, and B. McConnell, "Seal-2-Seal: A delay-tolerant protocol for contact logging in wildlife monitoring sensor networks," in *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008*, pp. 321–327, 2008.

[16] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, (Montreal, Quebec, Canada), pp. 32–40, 2007.

[17] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, (Hong Kong, Hong Kong, China), pp. 241–250, 2008.

[18] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, (Philadelphia, Pennsylvania, USA), pp. 252–259, 2005.

[19] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," in *Proc. IEEE Infocom*, vol. 6, 2006.

[20] T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, (Philadelphia, Pennsylvania, USA), pp. 260–267, 2005.

[21] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Multiple-copy routing in intermittently connected mobile networks," *Department of Electrical Engineering, University of Southern California, Tech. Rep. CENG-2004-12*, 2004.

[22] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," tech. rep., Technical Report CS-200006, Duke University, 2000.

[23] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, (Portland, Oregon, USA), pp. 145–158, 2004.

[24] A. Balasubramanian, B. Levine, and A. Venkataramani, "Dtn routing as a resource aladdress problem," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 373–384, 2007.

[25] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks. SIGMOBILE Mob," *Comput. Commun. Rev*, vol. 7, no. 3, pp. 19–20, 2003.

[26] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," in *IEEE SECON 2004*, pp. 235–244, October 2004.

[27] D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.

[28] D. LaI, A. Manjeshwar, F. Herrmann, E. Uysal-Biyikoglu, and A. Keshavarzian, "Measurement and characterization of link quality metrics in energy constrained wireless sensor networks," in *IEEE Global Telecommunications Conference, 2003. GLOBECOM'03*, vol. 1, 2003.

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

[30] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki-a Lightweight and Flexible Operating System for Tiny Networked Sensors, in proc. of 29th IEEE International Conference on Local Computer Networks (LCN04)," *Tampa, FL, USA*, pp. 455–462, 2004.

[31] A. Köpke, M. Swigulski, K. Wessel, D. Willkomm, P. T. K. Haneveld, T. E. V. Parker, O. W. Visser, H. S. Lichte, and S. Valentin, "Simulating wireless and mobile networks in omnet++ the mixim vision," in *Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, (Marseille, France), pp. 1–8, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[32] "OpenMAC." `www.meshnetics.com/opensource/mac/`.