Using Local and Global Knowledge in Wireless Sensor Networks

Christopher Gwilliams
Cardiff University

November 11, 2013

Contents

1 Introduction **1**
- 1.1 The Local Knowledge Problem . . . 2
- 1.2 Motivation . . . 3
- 1.3 Research Contributions . . . 4
- 1.4 Thesis Structure . . . 4

2 Background **5**
- 2.1 Wireless Sensor Network Issues . . . 5
  - 2.1.1 Fault Tolerance . . . 5
  - 2.1.2 Hardware Constraints . . . 6
  - 2.1.3 Energy Constraints . . . 6
  - 2.1.4 Transmission Medium . . . 7
  - 2.1.5 Environment . . . 7
- 2.2 Routing Protocols . . . 8
  - 2.2.1 Flat . . . 8
  - 2.2.2 Data-centric . . . 9
  - 2.2.3 Hierarchical . . . 10
  - 2.2.4 Location-based . . . 11
- 2.3 Sensor Middleware . . . 12
  - 2.3.1 Issues . . . 13

List of Figures

List of Tables

Abstract

To Write...

Chapter 1

Introduction

A wireless sensor network (WSN) consists of a collection of heterogeneous nodes with sensing and, typically, wireless capabilities. These sensing nodes can be extremely complex and powerful devices with the ability to sense multiple phenomena simultaneously, or they can be simple motes that have limited processing power and are tasked with sensing one thing in their environment.

Upon deployment, these nodes use their wireless capabilities to form links with their neighbours, where a neighbour is any node that is within transmission range. The way that nodes discover, and communicate with, their neighbours is defined by their routing protocol. Routing protocols vary based on the purpose of the WSN, the requirements of data transmission as well as the characteristics of the nodes. Communication between nodes is expensive and drains the available power faster than any other action that a node performs. For example, if a WSN is deployed in a building with consistent power available, then the routing protocol does not need to be adapted to ensure the nodes maximise their battery life by transmitting as little as possible. However, not every WSN has unlimited resources at their disposal and these protocols, as well as the underlying structure of the network, are used to ensure the network is able to perform well for as long as possible.

Each WSN is different and each will have different constraints, a WSN that monitors traffic along a busy road may experience memory limitations, whereas a WSN that is deployed in the middle of a desert may experience power issues. Typically, however, all WSNs do the same thing: sense one or more characteristics of their environment and forward that data on to a specified endpoint.

1.1 The Local Knowledge Problem

The majority of WSNs do not know what data they are sensing, or have any knowledge of their environment. This means that, unless fixed by a routing protocol or human that deployed the network, data is delivered on a chronological basis and is then filtered at the base station, usually manually. Some WSNs store all of the data on the node and users of the network must use a 'pull' model to query for data from nodes, but this requires some technical knowledge and, while it does increase the battery life of the nodes, it is a manual process again.

The environment that a WSN is deployed is usually rich and the data sensed often contains patterns that can be used to improve the performance of the network. For example, if a node knows that it is has only been triggering between the hours of 6pm and 5am for the past few weeks, it can enter a deep sleep outside of those hours or use that time to transmit data it has been storing while it knows it will be inactive. Alternatively, this knowledge can be used to prioritise data throughout the network so that the most important data is received first, instead of the most recent. An example of this could two camera nodes deployed facing the entry and exit of a building, tasked with looking for intruders between 5pm and 8am. If the camera facing the exit is triggered at 5:01pm and the camera on the entrance is triggered at 5:05pm, then the knowledge that the security guard leaves through the exit between 5:01pm and 5:08 pm will allow the entrance camera to prioritise its capture as more important, as it is an irregular occurrence.

This knowledge can be categorised into *local* and *global*. Local knowledge (LK) is the knowledge of an area that has been gained through experience or experimentation and global knowledge (GK) is knowledge that is generally available to everybody. An example of this is someone who has been tasked with deploying a WSN in the Amazon rainforest would use readily available sources, such as the Internet or prior research, to determine the humidity and weather patterns in order to use a node that could withstand such conditions. This would be classed as GK. However, a native to the Amazon may know that three of the locations that the nodes are to be deployed in are flooded for two weeks of the year, rendering their readings useless for that time period and increasing their risk of failure. This is LK, as it cannot be gained without experiencing the flooding in that area, or experimenting with water levels.

We believe that the use of this knowledge can increase the efficiency of the network, as well as prioritise sensed data by its value instead of the time it was recorded. To show this, we have developed a network architecture for WSNs that utilises knowledge from the data it senses, as well as its deployed environment. It is called the Knowledge-based Hierarchical Architecture for

Sensing (K-HAS) and this thesis will show how K-HAS addresses the problem of delivering the most important data first and improving the overall efficiency of the network.

1.2 Motivation

Throughout this thesis, we focus on a scenario motivated by our collaboration with Cardiff University School of Biosciences, who run a research centre in the Malaysian rainforest, in Sabah, known as Danau Girang (DG) . Located on the banks of the Kinabatangan river, DG has been running for more than six years and holds Masters, PhD and Undergraduate students from around the world, studying the ecology and biodiversity of the unique region.

The reason that the rainforest that DG is set in is so unique is that the area was heavily logged until the late 1970s and now serves as a corridor, between large Palm Oil plantations, connecting two separate rainforest lots. The area is now secondary rainforest (rainforest that has grown since being destroyed) and is experiencing a large variety of wildlife using the area as a habitat, or as a path. Some of this wildlife is unique to this area of the world and DG has had sightings of animals that have not been seen in many years.

There is a variety of research projects currently underway in the field centre, looking into fish population, crocodile attacks, hornbill habitats or the movement patterns of small mammals. One project that has been running almost since DG opened, is the *corridor monitoring programme*, a programme that consists of dozens of wildlife cameras deployed in various areas around DG and triggered whenever an animal triggers a break in their infrared (IR) sensor.

The Kinabatangan is a very humid place, with thick forest, making it very difficult to walk through and even more difficult for hardware to survive the conditions. Cameras are placed along the river and up to 1km into the forest, recording triggers onto SD cards. These SD cards are collected and stored at the field centre, where the images are manually collated and processed. The cameras are designed to have a battery life of three months but, due to the humidity, a battery life of 3 weeks is more realistic. In 2010, twenty cameras were deployed and half of them were inspected every two weeks, on a rotating basis. In that time, each cameras can record more than a thousand pictures and the dynamic nature of the rainforest, such as the sun through leaves, falling trees and reflections in the water can cause the camera to trigger when an animal has not walked past; we call these *false triggers*. False triggers can make up to 70

We have used this scenario to test our hypothesis and implement a WSN that automates the collection, transmission, processing and storage of images, using LK to classify the data and prioritise the flow of information through the network, making more efficient use of the limited power and bandwidth available.

1.3 Research Contributions

1.4 Thesis Structure

The rest of this thesis is structured as follows. Chapter 2 provides some background on wireless sensor networks and the use of knowledge. Chapter 3 explains some technical decisions we made and the findings when running experiments in the Malaysian rainforest. Chapter 4 introduces the K-HAS architecture we have developed and explains the purpose of each tier. Chapter 5 details the ontology we have developed to support K-HAS and shows how current ontologies do not sufficiently cover all of the concepts involved with a *scientific observation*. Chapter 6 shows how we use rules to determine how valuable a piece of sensed data is and to explain how K-HAS is able to use these rules, and human feedback, in order to inform future classifications. Chapter 7 highlights the technical limitations of implementing K-HAS today and shows our simulations of it running as it was designed. Chapter 8 then concludes this thesis and summarises our contributions and findings, as well as highlighting work that could be undertaken to take this project further.

Chapter 2

Background

Using knowledge in a WSN is related to existing research into sensor networks that utilise context-awareness in order to improve their efficiency or adapt their sampling rate.

This chapter is split into the following sections. Section 2.1 outlines the issues surrounding WSN design and deployment. Section 2.2 details relevant existing routing protocols for sensor networks. Section 2.3 highlights popular sensor middleware in use today. Section **??** shows some examples of existing WSNs that are related to our motivating scenario. Section **??** introduces research into local and global knowledge and Section **??** shows some related work into WSNs that utilise knowledge or context to prioritise data and/or improve efficiency.

2.1 Wireless Sensor Network Issues

WSNs have been used in a number of domains, for a range of different purposes, from habitat monitoring [28] to military purposes [23] and healthcare [22]. While these applications are vastly different, the technology behind each is very similar. Each requires the use of nodes with sensors attached and each node requires a power source and storage devices.

According to [3], there are eight factors that affect the design of sensor networks, but we focus on a subset that are the most relevant to our research problem. The following points must be considered when designing a WSN:

2.1.1 Fault Tolerance

WSNs typically contain a large number of nodes and each can fail for various reasons, from a lack of power, filling its storage capacity, to factors of the environment causing the hardware to fail. While the sensor nodes that are used in WSNs typically consist of the same platform, the variation between each deployment means that the device itself must be adapted to its environment, [19] used a custom protective casing for their nodes so that they were able to survive being in the open while ensuring that the transmission range was not affected.

2.1.2 Hardware Constraints

A sensor node typically consists of: a platform that contains the memory and processing power, a sensor (or sensors) and a transceiver that uses a wireless standard, such as Wi-Fi or Zigbee. Cost and size are the most common barriers to entry when designing a WSN. [14] mentions that the expectation of a sensor node is a matchbox-sized form factor. While the research is over ten years old, the original focus on sensor node was *smart dust* [16], small, inexpensive, disposable nodes that can transmit until their power reserve is depleted, [4] mentions that it is a requirement for the nodes to cost less than USD10. In [8], it is noted that, a decade on from the first WSN papers, smart dust has not been realised and the focus has instead been on larger, more powerful nodes that have reduced in cost and grown in power.

The Gartner Hype Cycle for 2013 [**?**] shows that smart dust is still in early innovation stages and may not be fully commercialised for another ten years. To counter this, research has been focussed on using software solutions to maximise the battery life in these more powerful, more expensive nodes, accompanies with the use of renewable energy sources.

2.1.3 Energy Constraints

The majority of sensor nodes do not have access to a constant power supply and must run on a battery that is, generally, a similar size to the node itself; or smaller. This means that the nodes must be as efficient as possible, knowing when to transmit data and when to sleep. The lifetime of a sensor network is extremely dependent on the battery life of each node and, unlike other mobile devices, they cannot typically be recharged [3]. Much work has been done on power efficient routing protocols, as well as the control of which attached devices are active [25, 13, 24].

The limited resources on the nodes mean that the sensing devices, and transceivers, attached must consume as little power as possible. Some routing protocols implement turning off wireless radios and scheduling a wakeup across the network [30], but the cost of turning off a device can waste just as much energy as leaving it on and sampling at a lower rate; if not more [10].

The use of energy in a node is dependent on how active the sensor(s) are, how much it transmits and receives, the transmission medium used as well as the environment it is in.

2.1.4 Transmission Medium

Common transmission media, such as Wi-Fi, are viable solutions in WSNs when a high data rate is required and power is readily available. However, research has shown that Wi-Fi is extremely power-hungry and [18] shows that Wi-Fi consumes almost 9 times more energy, while transmitting, than other standards, such as Zigbee. Bluetooth is a more power efficient standard that is becoming increasingly popular for sensing devices that are [art of the *Quantified Self* movement [27], with wearable devices that report measurements, such as heart rate, steps taken and calories burned. With the advent of the new low-power Bluetooh 4.0 , also known as Bluetooth Low Energy (BLE), this standard is supposed to allow months of continuous use on a coin-cell battery [**?**]. However, the range is limited to 100m and, using the same frequency, as Wi-Fi (2.4GHz) means that it is as susceptible to path loss and reduced transfer rates. [33] shows that a 2.4GHz Wi-Fi antenna is capable of transmitting up to 350m, while a considerably lower frequency of 41MHz was able to achieve links of 10km. The use of 2.4GHz frequencies in wet conditions have been shown to reduce the performance by up to 28New low-power, low-frequency standards have emerged in recent years and allow for a considerably longer range and increased battery life, at the cost of transmission speeds. Digmish is an example of this and, while it can achieve 250kb/s using 2.4GHz, it has much slower speeds of 125kbps when using the 900MHz spectrum. However, it does offer a range of, up to, 64Km [5].

2.1.5 Environment

The environment that a node is deployed in can have a great impact on almost all aspects of a WSN, such as range and battery life. Harsh environments that are not easily accessible make it difficult to place nodes and protected environments may limit where nodes can be placed. In [21], nodes were deployed within glaciers and had to survive extreme temperatures, lasting without human intervention, for months at a time. [15] attached collars to Zebras that had to withstand high speed movement, dust and high temperatures. The deployment of any node requires extensive research as to the environment that it will be deployed in and adjustments must be made to ensure it is able to survive for extended periods without continued maintenance. Section 2.1.4 also shows that environment does not simply affect the hardware, but humidity can reduce the transmission range significantly, as well as moisture collecting on wireless antenna can reduce the range for days at a time.

2.2 Routing Protocols

Routing protocols specify how nodes in a WSN are organised, as well as how they transmit data throughout the network. In [2], the more popular routing protocols are surveyed and split into three of the main identified categories: data-centric, hierarchical and location-based. We use the aforementioned categories, as well as flat, to highlight some of the key protocols that are relevant to our work. The protocols have the task of ensuring that a network is performing at its best, providing the best lifetime and ensuring reliable and consistent delivery of data. This must reduce *flooding*, where nodes send every message to every link, aside from themselves, effectively flooding the network with unnecessary messages, and find a way to deliver data to the endpoint using the most efficient path possible.

2.2.1 Flat

Initially, this was the most common structure for a WSN, dozens of nodes spread out over a geographical area, with one or more neighbours, sending observations to a single endpoint.

MCFA

The Minimum Cost Forwarding Algorithm (MCFA) is an flat routing protocol that works by assigning costs to each node, based on how many hops they are from the base station [6].

Each node has a path-estimate of the cost of transmission from itself to the base station. The base station sends out a broadcast message and it is received by all nodes in range. The message contains a cost from the base station (initially zero) while every node has their cost set to infinity. If the cost in the message, plus the link it was received, is less than the current cost. If yes, the estimate is update on both the node and the message; the message is then passed on to other nodes in range.

This approach allows for dynamic reconfiguration of the network, as well as a reduced overhead due to not having to maintain a global routing table on each node. The assumption with MCFA, however, is that the direction of routing is always towards a fixed endpoint.

2.2.2 Data-centric

Data-centric routing protocols are not like traditional WSNs where nodes are given addresses; they use a method that involves the advertising, or querying, of the data that has been sensed and those with the relevant data can respond to the request.

SPIN

Sensor Protocols for Information via Negotation is one of the first data-centric protocols and attempts to address the issue of flooding the network whenever new data is sensed by addressing the data through metadata [12]. SPIN works on three messages passed between nodes:

1. ADV - A message sent by a node when it has sensed new data, advertising what it has recorded.

2. REQ - Sent by nodes that received an ADV to request the data.

3. DATA - Message containing the sensed data.

When a node has sensed data, it sends an ADV message to all nodes within range. If any of those nodes are interested in the data, then they respond with a REQ message, at which point the DATA message is sent to nodes that responded.

SPIN eliminates the need for a global view of the network topology, as nodes only need to know their single hop neighbours. However, SPIN does not guarantee equal diffusion of data throughout the network as a node may be interested in the data sensed at the other edge of the network, with only nodes that are not interested in between. This would mean that those nodes would not request the data or pass it on.

SPIN-IT

An extension to SPIN, SPIN-IT uses a slightly different approach to receiving data and is developed solely for the transfer of images [31].

Nodes use the existing message structure of SPIN, but REQ messages are used as queries, sent to all nodes in transmission range. The receiving nodes keep these requests and generate a new REQ message, thus allowing nodes to store temporal paths. When a REQ reaches a node that has the desired data, it responds with a ROUTE-REPLY message. This message is used because images are large and resource-constrained WSNs would have a much shorter lifetime if a lot of unnecessary transmissions were made. The

ROUTE-REPLY is used in case multiple nodes, in range of the requesting node, have the data it has requested and it can then choose the optimal route. As each node keeps a history of REQ messages, these can be used to trace the requested data back through the network, to the originating node, without the overhead of maintaining a global routing table.

COUGAR

A slightly different data-centric approach is the proposed COUGAR protocol, viewing the network as a distributed database. While similar to SPIN due to the fact that it does not forward data as soon as it is sensed, COUGAR uses a query language that abstracts the underlying network structure and uses that query to generate a plan that utilises in-network processing to provide an answer [32].

Within the network, a *leader* is selected and this node is used to aggregate the data from nodes that were able to fulfil all, or some, of the query. At risk of failure, each query should result in a *leader* being dynamically selected and it must have sufficient resources to be able to satisfy the request. This protocol was only proposed, and much of the technical detail has yet to be completed, but the concept of treating the network as a distributed database is a novel idea and this is one of the first protocols to suggest the use of a query language that could be used by people without specific domain knowledge.

2.2.3 Hierarchical

Hierarchical networks are WSNs that contain nodes of different classes; nodes at the end of the network are typically clustered into groups and served by a gateway node. This gateway could be in charge of aggregating the data, processing the data, or simply forwarding it to an endpoint. Clusters of nodes allow the network to be spread out over a wider geographical area and gateway nodes can use a different transmission method to provide long distance links to the base station. Gateway nodes serving a cluster of nodes means that the network can scale easily as well, simply by adding a new cluster to the network.

TEEN

The Threshold sensitive Energy Efficient sensor Network (TEEN) protocol is designed for reactive sensor networks, networks that require instant reactions to changes sensed in their environment. TEEN recognises that transmission is the most power hungry action for a node so each node is coded with a hard and soft threshold. The hard threshold is a value that makes nodes transmit the reading to their cluster head. Similarly, the soft threshold is a small change in the value of the sensed attribute that causes further transmissions.

During the initialisation of the network, the base station sends information about the thresholds and sensing attributes to all cluster heads in the network; the cluster heads then forward this on to all nodes in their cluster. When a node senses data over the hard threshold, it transmits to the cluster and only transmits again when new sensed values are greater than the hard threshold and the difference between the current sensed value and the previous is greater than the soft threshold [20].

Clusters are assigned for a period of time and then new clusters are selected by the base station, at which point new attributes and thresholds are broadcast to all nodes. This kind of protocol allows the network to be dynamic after deployment and allows user input based on the data that has been sensed in the previous cluster times.

For example, a network could be tasked with sensing humidity in a rainforest but the thresholds have been set such that nodes are transmitting readings that are not of interest. A user can change these thresholds and they will be pushed out to the nodes at the time that the next clusters are chosen, without any need to visit the node or configure them individually.

2.2.4 Location-based

Instead of using the physical addresses of nodes, or the data they store, location-based protocols are based on the region that nodes are deployed in. **FLESH OUT**

Span

Span is a protocol where nodes are selected as *coordinators* based on their positions. A node can decide to be a coordinator based on the amount of energy it has and the number of neighbouring nodes it would benefit if they were able to use it as a bridge [7].

An example of this would be node B placed between node A and C. C and A are unable to communicate directly so, when node B wakes up, it decides whether it should become a coordinator. It knows that it has sufficient energy levels and it can provide connectivity for a previously disconnected area of the network, so it chooses to become a coordinator, staying awake and routing sensed data to other coordinators, which form the backbone of the network.

Results showed that using Span, in a system that transmits using 802.11, provides an network lifetime increase of more than a factor of 2 over networks that just use the 802.11 protocol.

GEAR

The Geographic Energy-Aware Routing protocol (GEAR) is similar to SPAN in that it makes routing choices based on both energy-awareness and location. Each node maintains an *estimated cost* and a *learning cost* of forwarding a packet through its neighbours. The estimated cost is calculated using the distance to the packet destination and the energy remaining on the node whereas the learning cost is the estimated cost that takes holes in the network into consideration [**?**].

GEAR is designed to perform in two phases: forwarding a packet towards a region and disseminating a packet within a region. When sending a packet towards a destination, GEAR either sends a packet on to the node in range that is closest to the destination or, if such a node does not exist, then a hole is identified. If a hole is identified then the node that minimises a cost is selected.

To disseminate a packet within a geographic area, uses algorithms based on the density of the network. Recursive geographic forwarding is typically used but this can result in an endless loop if the density of the network means that the region is unable to contact the destination. In that case, restrictive flooding is used.

Provide a concluding section for routing protocols here?

2.3 Sensor Middleware

Acting as a bridge between the hardware and the user, sensor middleware is software that abstracts the underlying network from the user and provides a means of accessing sensed data and administrating how the network performs. These middlewares must not be specific to a single network and provide support for as many different sensor nodes as possible. In [**?**], a middleware is said to provide standardised services to many applications and perform operations that make effective use of limited system resource.

In [**?**], a middleware should include four major components: programming abstraction, system services, runtime support and Quality of Service (QoS) mechanisms. In this section, we will discuss the challenges surrounding middlewares for WSNs and highlight some existing middleware that are particularly relevant to our research problem and/or motivating scenario.

2.3.1 Issues

WSNs present a range of new challenges to existing middleware, due to their resource constraints, deployment environments and more. However, there has been research into the key issues that must be addressed in order for middleware to be considered suitable. While there has been a number of surveys into these challenges [?, ?, ?], we will detail those that we believe to be most relevant to our work.

Resource Constraints

It is rare that nodes in a WSN would have a constant power source, unlimited memory and a casing that can survive a harsh environment without decaying. In order to ensure that the lifetime of nodes is mazimised, middleware needs to offer a power scheduling system that makes efficient use of the hardware on the node, typically turning off the radio at particular times.

Ideally, a middleware will be able to coordinate nodes through wireless communication, making efficient use of transmissions and dynamically modifying sleep schedules based on the power remaining.

Heterogeneity

Not every node in a network will have the same capabilities, manufacturer or hardware. WSN middleware needs to provide a standard interface to add data, regardless of the format it is recorded in.

Bibliography

[1] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, May 2005.

[2] I Akyildiz. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, March 2002.

[3] IF Akyildiz. A Survey on Sensor Networks. *Communications ...*, (August):102–114, 2002.

[4] Daryoush Bayat, Daryoush Habibi, and Iftekhar Ahmad. Development of a Wireless Sensor Node for Environmental Monitoring. *... International Conference on Sensor ...*, (c):1–5, 2012.

[5] A. Chen. A scalable solution to minimum cost forwarding in large sensor networks. In *Proceedings Tenth International Conference on Computer Communications and Networks (Cat. No.01EX495)*, pages 304–309. IEEE, 2001.

[6] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless Networks*, 8(5):481–494, 2002.

[7] Peter Corke, Tim Wark, and Raja Jurdak. Environmental wireless sensor networks. *Proceedings of the IEEE*, 98(11), 2010.

[8] D Estrin and L Girod. INSTRUMENTING THE WORLD WITH WIRELESS SENSOR NETWORKS. *Acoustics, Speech, and ...*, 2001.

[9] S. Hadim and N. Mohamed. Middleware: Middleware Challenges and Approaches for Wireless Sensor Networks. *IEEE Distributed Systems Online*, 7(3):1–1, March 2006.

[10] WR Heinzelman, J Kulik, and H Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. *... of the 5th annual ACM/IEEE ...*, 1999.

[11] Mark Hempstead, Nikhil Tripathi, Patrick Mauro, Gu-Yeon Wei, and David Brooks. An Ultra Low Power System Architecture for Sensor Network Applications. *ACM SIGARCH Computer Architecture News*, 33(2):208–219, May 2005.

[12] C Intanagonwiwat, R Govindan, and Deborah Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. *Proceedings of the 6th annual ...*, pages 56–67, 2000.

[13] Philo Juang, Hidekazu Oki, Yong Wang Ý, Margaret Martonosi, Li-shiuan Peh, Daniel Rubenstein , Óññùò ØÓò, Ú Ó Ñ Ò Ö Ò ý, ÑôóöøÒø, Ö Ú Ö Ó Ñ Ò Ý Ò Û Óñôùø, Ò Ô Ô ØÓò, Ø Ó Ò Òúóú, Ò Ùøóòóñóù Ù Ó Óñôùø, Ò Û Ö Óññùò ØÓò, Ú Ó Ö Óø, Ôôý Ò Û Ö Ô Öøóô Ö Ò Øûóö Ò ØÒ Õ Ù Ò, Ñ Ó Ò Ó Ö Ò Øûóö, Ö ÆØÝ ØÑ Ò Ù Ù Øóñ, Ø Ö Ò Ó, Ó Ö Ý Û Ö Øö, Ò Ú Ö Ò Ñ Èí, Ò Øý, Ó Ú Ö Ò ØÖ Ó Ò Û Ö Ò Ñ Ö Øù, Ó Ñ Ò Òøö, Ôöóøó Ó, Ò Ò Ö Ý ØÖ Ó Ôö, Ò Ó ØÖ Û Ö Ò Ò Óö, Ô Ô ØÓò, Ò ØÒ Õ Ù óû, and Ø Ñ ØÓ Û Ô Ò Øóö. Energy-Efficient Computing for Wildlife Tracking : Design Tradeoffs and Early Experiences with ZebraNet. *Structure*, 2002.

[14] JM Kahn, RH Katz, and KSJ Pister. Next century challenges: mobile networking for Smart Dust. *... on Mobile computing and networking*, pages 271–278, 1999.

[15] Jin-Shyan Lee, Yu-Wei Su, and Chung-Chou Shen. A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. *IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society*, pages 46–51, 2007.

[16] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications - WSNA '02*, page 88, 2002.

[17] a. Manjeshwar and D.P. Agrawal. TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. *Proceedings 15th International Parallel and Distributed Processing Symposium. IPDPS 2001*, 00(C):2009–2015, 2001.

[18] K. Martinez, R. Ong, and J. Hart. Glacsweb: a sensor network for hostile environments. *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.*, pages 81–87, 2004.

[19] Chris Otto and A Milenkovic. System architecture of a wireless body area sensor network for ubiquitous health monitoring. *Journal of Mobile . . .*, 1(4):307–326, 2006.

[20] Diego Pizzocaro, Fangfei Chen, Thomas La Porta, and Matthew P Johnson. System Architectures for Multi-Sensor Task Allocation. *Network*.

[21] Atiqur Rahman. Middleware for wireless sensor networks : Challenges and Approaches Sensor Network application : 3 Middleware for Sensor Network : Network :. *Security*, pages 2–6.

[22] C. Schurgers and M.B. Srivastava. Energy efficient routing in wireless sensor networks. *2001 MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force (Cat. No.01CH37277)*, pages 357–361.

[23] Michael Segal. Improving lifetime of wireless sensor networks. *Network Protocols and Algorithms*, 1(2):48–60, 2010.

[24] Melanie Swan. Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self 2.0. *Journal of Sensor and Actuator Networks*, 1(3):217–253, November 2012.

[25] R. Szewczyk, Eric Osterweil, Joseph Polastre, Michael Hamilton, Alan Mainwaring, and Deborah Estrin. Habitat Monitoring With Sensor Networks. *Communications of the ACM*, 47(6):34–40, 2004.

[26] N.H. Vaidya. A wakeup scheme for sensor networks: achieving balance between energy saving and end-to-end delay. *Proceedings. RTAS 2004. 10th IEEE Real-Time and Embedded Technology and Applications Symposium, 2004.*, pages 19–26, 2004.

[27] M.M. Wang, J.N. Cao, J. Li, and S.K. Dasi. Middleware for wireless sensor networks: A survey. *Journal of computer science and technology*, 23(3):305–326, 2008.

[28] E. Woodrow and W. Heinzelman. SPIN-IT : A DATA CENTRIC ROUTING PROTOCOL FOR IMAGE RETRIEVAL IN WIRELESS NETWORKS. In *Proc. IEEE ICIP*. Citeseer, 2002.

[29] Yong Yao and Johannes Gehrke. The cougar approach to in-network query processing in sensor networks. *ACM SIGMOD Record*, 31(3):9, September 2002.

[30] Yan Yu and Ramesh Govindan. Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks. *Technical Report, UCLA-CSD TR-01-0023*, 2001.

[31] Yang Yu, Bhaskar Krishnamachari, and VK Prasanna. Issues in designing middleware for wireless sensor networks. *Network, IEEE*, pages 1–7, 2004.

[32] Marco Zennaro and Antoine Bagula. Planning and deploying long distance wireless sensor networks: The integration of simulation and experimentation. *... and Wireless Networks*, pages 191–204, 2010.