ADAPTABILITY IN SENSOR NETWORKS

A Thesis

in TCC 402

Presented to

The Faculty of the

School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Computer Engineering

by

Jonathan McCarrell McCune

April 8, 2003

On my honor as a University student, on this assignment I have neither given nor received unauthorized aid as defined by the Honor Guidelines for Papers in TCC Courses.

_____

Approved _____ (Technical Advisor)

David Evans (Signature)

Approved _____ (TCC Advisor)

Patricia Click (Signature)

Table of Contents

Table of Figures

Glossary of Terms

Application Layer – The application layer is the uppermost layer in the OSI network model. It is commonly defined as providing network services to end users. With respect to sensor networks, the base station is the end user for the sensor nodes, and the human operator of the sensor network is the end user for the base station.

Base Station – The base station is a master node. Data sensed by the network is routed back to the base station. Sensor networks in this thesis contain a single base station.

Fidelity – With respect to sensor networks, fidelity is a measure of the quality of data received at the base station. Fidelity can change as a result of adaptations made in the sensor network, and it can degrade due to the failure of sensor nodes.

Hop – With respect to wireless networks, a hop is a communication link between two nodes, without any intermediate nodes to forward data messages.

Longevity – With respect to sensor networks, longevity is a measure of the time required for sensor network performance to degrade to some specified threshold. Time can be measured in seconds, in requests issued by the base station, or in responses received by the base station.

Node – A node is a sensing device, containing a complete computer system with a processor, memory, radio data link, and one or more electronic sensors. Nodes are typically battery powered.

Open System Interconnect (OSI) Network Model – The OSI model is the International Standard Organization's (ISO) standard model for networking protocols and distributed applications. It defines seven network layers: physical, data link, network, transport, session, presentation, and application.

Sensor Network – A sensor network is a collection of communicating sensing devices, or nodes, with a base station. All of the nodes are not necessarily communicating at any particular time, and nodes can only communicate with a few nearby nodes.

Abstract

Stringent energy constraints restrict the practical applications for sensor networks, as battery technology lags behind microelectronic system fabrication technology. Traditional sensor networks are built with general purpose processors (GPPs) because processing power consumption is insignificant compared to radio power consumption. As applications for sensor networks become more sophisticated, processor power utilization becomes significant. Successful sensor networks must adapt to changing conditions and requirements in order to maintain energy-efficient operation. This thesis considered a combination of two approaches to adaptability: parameterizeable algorithms and hardware implementation.

Parameterizeable algorithms allow sensor devices to tailor their operation to specific conditions and requirements. Examples of parameterizeable algorithms include JPEG image compression and most symmetric ciphers. GPPs are popular because of their ability to perform any computable function (limited only by time, energy, and memory constraints) and the ease with which different programs can be executed. Sensor networks implemented with more efficient hardware designs offer improved performance, as required levels of adaptability can be achieved on simpler and more efficient hardware.

Two sensor network applications – JPEG image compression and encryption – were analyzed to determine the impact of adaptability on fidelity and longevity. Even in applications where transmission costs dominate, such as JPEG image compression, energy savings obtained from using a more efficient processing implementation are

significant. In applications where processing costs dominate, such as encryption, improvements of well over 100% in terms of network longevity can be gained by switching from GPPs to small scale reconfigurable (SSR) hardware. SSR hardware is shown to be an optimal design choice because algorithms can be implemented with efficiency approaching that of application specific integrated circuits (ASICs) while maintaining adaptability.

1. Introduction

Wireless distributed sensor networks have emerged in the past five years as a result of improvements in microelectronic system fabrication. Battery technology has not kept pace with microelectronic system-design technology, resulting in stringent energy constraints for sensor networks. A critical feature of successful sensor networks is their ability to adapt to changing conditions and requirements, thereby maintaining power-efficient operation.

Traditional sensor network devices are built with general purpose processors (GPPs). Programs executing on sensor devices with GPPs can be adapted to perform an enormous range of operations, limited only by energy, time, and memory constraints on the sensor node. Unfortunately, the power cost per operation for GPPs is high compared to a fixed-logic hardware implementation of the same operation. Fixed-logic hardware implementations, such as application specific integrated circuits (ASICs), offer excellent power-efficiency, but the operations are fixed at design time, and cannot be adapted in the field.

One alternative is the field programmable gate array (FPGA). FPGAs can be reconfigured many times, even while in the field, to implement many algorithms in hardware. Unfortunately, their general-purpose nature results in higher energy consumption than fixed logic implementations. A new type of hardware, small scale reconfigurable (SSR) hardware, offers an alternative to FPGAs and ASICs.

This thesis investigated adaptability in sensor networks, with a focus on the advantages that can be gained from SSR hardware. A simulation environment was

developed to investigate the impact of adaptability and SSR hardware on power consumption in sensor networks. Two applications were considered: image compression and encryption.

1.1. Sensor Networks Defined

A sensor network is a collection of communicating sensing devices, or nodes. All of the nodes are not necessarily communicating at any particular time, and nodes can only communicate with a few nearby nodes. The network has a routing protocol to control the routing of data messages between nodes. The routing protocol also attempts to get messages to the base station in an energy-efficient manner.

The base station is a master node. Data sensed by the network is routed back to a base station. The base station is a larger computer where data from the sensor network will be compiled and processed. The base station can be thought of as a controller for the sensor network. It is the source of instructions concerning the type of phenomena to be sensed, and it collects all results. Human operators controlling the sensor network send commands and receive responses through the base station.

1.2. Sensor Network Applications

Battery-powered and wireless sensor devices are small, expendable, and inexpensive. Practical applications for networks of these sensors range from military surveillance to environmental monitoring to corrosion detection in large structures.

Deployment of sensors on a battlefield can reduce the need for soldiers to put themselves in danger. Bridge inspectors will no longer need to climb to dangerous heights to examine corrosion, since the sensors will be able to report conditions (Warnke).

1.3. Problem Definition

Adaptability has become one of the principle design features of successful sensor networks. Lach and Evans classify the stages of a sensor network's lifetime as device design time, application design time, scenario design time, deployment, and operation (Lach and Evans, 2003). As a sensor network progresses further along its lifetime, it becomes increasingly difficult to incorporate adaptability. Eliminating the use of general purpose hardware, while maintaining operation-time adaptability, is a very challenging problem.

Today's sensor networks make use of radio data links for communication. Energy consumed by the transmission and reception of radio messages quickly depletes existing batteries. Research to date has focused on ways to minimize radio energy usage on sensor devices to maximize the useful lifetime of sensor networks. In existing sensor networks, energy consumed in data processing is less significant than energy used in data transmission. As future sensor networks gather more sophisticated data, such as frames from a video camera, data processing will have a more significant impact on energy consumption.

1.4. Rationale

Today's sensing devices employ general purpose computing and radio hardware requiring high energy consumption. An optimal sensing device would have the ability to power down all components not integral to the present task, while maintaining the ability to adapt to changing conditions and requirements. Devices could reconfigure to sense other kinds of data or perform other computations on sensed data.

SSR hardware is designed to bridge the gap between general-purpose and application-specific devices. While not as efficient as application-specific hardware, these reconfigurable devices can be adjusted to tradeoff network fidelity for longevity, thereby optimizing for the requirements of a specific application. Figure 1 illustrates the energy-efficiency and flexibility tradeoffs of common hardware implementations used for sensor networks.

Figure 1 - Progression of sensor network hardware platforms.

This thesis was based on the assumption that future sensor networks will need to transmit and process significantly more data than current sensor networks. For example,

sensor nodes equipped with video cameras can adapt the resolution of recorded images to desired fidelity levels. As power is consumed on the nodes, the resolution may again need to be adapted to ensure network longevity.

Sensor networks can leverage hardware adjustments and additional computation done by individual nodes to enhance network fidelity and longevity, and to enable a tradeoff between the two. The implementation hardware platform, as well as the particular algorithms used for computation, can dramatically affect network fidelity and longevity.

1.5. Scope and Method

This thesis – the work of a Computer Engineering major – was a collaborative effort between the Computer Science (CS) and Electrical and Computer Engineering (ECE) Departments. ECE Professor John Lach and CS Professor David Evans co-advised this project. Jason Brandon, an electrical engineering graduate student at UVA, researched many of the applications and power statistics that were incorporated into the simulations for this thesis. Together, the four of us mapped out the objectives for the overall project and broke down the problem: the simulation which formed the basis for this thesis, and Jason Brandon's research into the specifics of several hardware platforms and the targeted adaptive sensing applications.

The simulations for this thesis were run based on two adaptive sensing applications. The first was an adaptive video sensing network, where each node's sensing device is a video camera. The second is a secure data transmission application,

where nodes encrypt their data before transmission. Simulations were run to verify the advantages of adaptability in software for these applications. Further simulations were then run to emphasize the additional gains that could be made by using reconfigurable hardware.

The simulation environment created for this thesis is built on version 2.02 of GloMoSim (Zeng, 1998), the Global Mobile Information System Simulator, and it was run under the version 7.2 of the Red Hat Linux operating system. GloMoSim is a freely available simulation framework for wireless computer networks.

1.6. Overview

Chapter 2 provides the reader with an improved background in the field of sensor networks. Chapter 3 lays out the methods and activities used in this thesis, including the contributions from other team members but focusing on the extensions of GloMoSim. Chapter 4 presents the argument for adaptability in software on sensor nodes. Chapter 5 extends the argument to include reconfigurable hardware. Chapter 6 concludes with a summary of the key lessons from this project. Recommendations for future work are made in the relevant technical sections.

2. Background – Sensor Networks

Electronic sensing devices have been an active research area for many years. Research started in the direction of wireless sensor networks as power and size requirements for sensing devices decreased. Chalermek Intanagonwiwat investigated communication between battery-powered sensor devices in "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks" (2000). The devices used in Intanagonwiwat's work were approximately the size of a matchbox. Figure 2 shows an example of a matchbox-sized device, known as a mote, developed at UC Berkeley. A year later, Brett Warnke et al. published "Smart Dust: Communicating with a Cubic Millimeter Computer" as an investigation into the potential capabilities of such a tiny device.

The latest advances in microelectronic fabrication have resulted in the creation of a new field in the past five years: distributed sensor networks. These networks are composed of large numbers of battery-powered and wireless sensor networks that are small, low-cost, and expendable. Brett Warnke et al. list some of the potential applications of their Smart Dust sensor devices: rapid deployment of defense networks by unmanned aerial vehicles or artillery; tracking the movements of birds, small animals, and insects; monitoring environmental conditions that affect crops and livestock; managing inventory control; constructing smart-office spaces; and providing interfaces for the disabled (44).

Moore's Law dictates that the numbers of transistors per square inch of integrated circuit will double every 18 months (Moore). Battery technology has lagged

considerably, with performance increases on the order of 5% per year. As a direct result, power consumption issues plague distributed sensor networks. Message transmission and reception dominate power usage on sensor nodes, with the microprocessor using significant amounts of power during periods of intense computation (Krishnamachari, 2002). Encryption is one example of a computationally intense application.

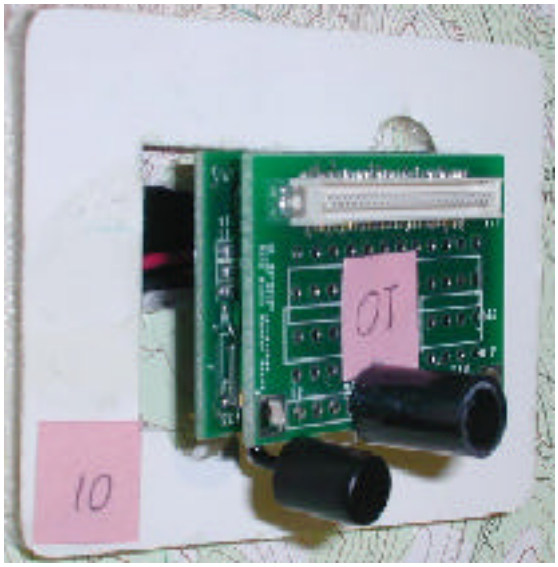

Figure 2 – A *Mote*, UC Berkeley's two-board wireless sensor platform. The processor and radio module are soldered on the lower board. The upper sensor board allows for additional sensor circuitry. The black cylinder is the antenna. The board measures 1.5 inches by 1 inch (Liu, 2002).

2.1. Computation – Data Aggregation and Security

Processing tasks that individual sensor nodes perform include data aggregation and data encryption. Data aggregation occurs when one node receives sensor readings from other nodes and consolidates that data in some way to reduce the amount of data that it must transmit back to the base station. Examples of data aggregation are duplicate

suppression, minimum, maximum, and average (Krishnamachari, 2002). Samuel Madden et al. have implemented an aggregation service for TinyOS in their paper "TAG: A Tiny AGgregation Service for Ad-Hoc Sensor Networks" (2002). TAG offers an approach for efficient distribution of simple, declarative queries to nodes in a sensor network. TAG and TinyOS currently run on matchbox-sized devices, like those in Figure 2.

Encryption algorithms are computationally intense with respect to the amount of data involved in cryptographic operations. Many common encryption algorithms require multiple rounds of calculations to be performed on each unit of data. Sasha Slijepcevic, in "On Communication Security in Wireless Ad-Hoc Sensor Networks," proposes an encryption algorithm that varies the number of rounds of encryption depending on the sensitivity of the data being encrypted. Slijepcevic's work is based on the assumption that data can easily be classified as either mobile code, locations of sensor nodes, or application specific data, with sensitivity decreasing respectively (2002).

2.2. Radio Power Consumption

The greatest consumption of energy in wireless sensor nodes is by the radio circuitry. Jason Brandon has estimated that common sensor-node radios use 50% of their transmission power even while sitting idle, listening for incoming messages. Consequently, many researchers are considering communications protocols where messages can only be sent during certain time windows, so that the radios can be powered down most of the time.

The uAMPS (micro-Adaptive Multi-domain Power aware Sensors) group at the Massachusetts Institute of Technology has been working to develop a set of enabling technologies for distributed sensor networks (Min, 2000). In "Dynamic Voltage Scaling Techniques for Distributed Microsensor Networks," Rex Min asserts that systems allowing a tradeoff between quality and energy savings are crucial to long system lifetimes (Min, 2000). This concept of a quality-energy tradeoff is shared by W. R. Heinzelman, et al in "Energy-Efficient Communication Protocol for Wireless Microsensor Networks."

The stringent power conditions under which sensor networks operate dictate that sensor devices capable of generating a large amount of output data cannot operate in high-fidelity mode all the time. Adaptable nodes that can operate in low-fidelity mode until interesting phenomena are observed and then switch to high-fidelity mode to gather more information offer the best solution to power limitations. This thesis considered a combination of two approaches to adaptability: parameterizeable algorithms and hardware implementation. There remains a great deal of work to be done on the power characteristics of specific hardware configurations. This thesis implemented a simulation of SSR hardware designed in the ECE Department at the University of Virginia.

3. Simulation Framework

The goal of this thesis was to run simulations to confirm the advantages of adaptability in sensor networks. First, parameterizeable algorithms were analyzed, and then the advantages of dynamically reconfigurable hardware in sensor networks were

examined. In order to run simulations, it was first necessary to extend a network simulation environment to include a power model for several hardware platforms and to construct an application that could run on the simulated nodes and perform sensing tasks. GloMoSim was the simulation framework chosen to achieve these objectives. Once the simulator was completed, it was necessary to do a large number of simulation runs to acquire data on sensor networks built on various hardware platforms. Input data for the simulations came from Jason Brandon.

3.1. GloMoSim

GloMoSim, the Global Mobile Information System Simulator, is a simulation framework for wireless networks and is freely available for academic use. It is built on top of Parsec, a parallel discrete event simulator (http://pcl.cs.ucla.edu/projects/glomosim/). GloMoSim was written in Parsec C, a modified version of the C programming language, and compiled with the Parsec C compiler. GloMoSim is highly configurable – one of the features that made it attractive for this project. It was necessary to write additional C code to model the power consumption of the particular hardware platform to be simulated and to cause the simulated nodes to behave like sensor nodes.

3.2. GloMoSim Extentions

One solution to managing the complexity of a networked computing environment is to break up the networking functionality into different layers. GloMoSim implements a slightly altered version of the OSI seven-layer network architecture, which simplifies development by modularizing networking functionality. Table 1 lists the GloMoSim layers that were used in the simulations for this thesis. Each layer is configurable, and the options that were selected for this thesis are underlined. Almost all of the work that was done to extend GloMoSim for this thesis was done in the application layer, with the addition of Sensor Network functionality to the application layer.

Layer	**Option**
Radio Propagation	Two ray and Free space
Radio Model	Noise Accumulating / Noise Free
Packet Reception Models	SNR bounded, BER based with BPSK/QPSK modulation
Data Link (MAC)	CSMA, IEEE 802.11 and MACA
Network (Routing)	IP with AODV, Bellman-Ford, DSR, Fisheye, LAR scheme 1, ODMRP, WRP, None
Transport	TCP and UDP
Application	CBR, FTP, HTTP and Telnet, Sensor Network

Table 1 - Modified OSI Network architecture used by GloMoSim (http://pcl.cs.ucla.edu/projects/glomosim/).

3.2.1. Power Model

Power consumption by individual nodes in the sensor network was modeled using two parameters: processor power consumption and data size. For each piece of sensed data, a certain amount of processing was done. Examples of the purpose for this processing include data encryption, data compression, and image manipulation. The data size parameter represented the post-processing data that then needed to be sent over the network.

3.2.2. Ad hoc Network Routing Tree Discovery

The final positions of the individual sensor nodes are unknown until they are actually deployed into the environment to be sensed. Even post-deployment, physical positions are not known, but each node can send and receive radio messages to the other nodes within the range of its radio equipment. Each node must participate in an ad hoc network setup algorithm. The algorithm used for this thesis is designed to set up a simple tree-hierarchy, where the base station is level 0, and the nodes within radio range of the base station are level 1. Remaining nodes within range of level 1 nodes become level 2 nodes, and so on, so that every node has a level. A node's level corresponds to its distance from the base station, in terms of hops. A hop is a message sent from one node to another, without any intermediate nodes forwarding the message.

To set up the routing tree, the base station broadcasts a tree-setup beacon with the same transmission power as the radios on the regular nodes. This beacon message includes the sending-node's unique identifier and the sending-node's level in the routing

tree. The goal of the beacon is to provide every node in the tree with a parent and a path to the base station. Upon reception of a beacon message, nodes will compare the level of their current parent (no parent is equivalent to maximum possible level) to the level of the received beacon. If the new level is lower (i.e., fewer hops to the base station), the node will adopt the sender of the beacon as its new parent.

Figure 3 shows two examples of routing trees formed from random deployments of 100 nodes. A critical difference between these two trees is the degree to which they are balanced. The first tree has poor balancing, since over half of the level 1 nodes (those connected to the base station by a single hop) have no children. The second tree has good balancing, since almost all of the level 1 nodes have children. The performance of a sensor network depends heavily on the degree to which the routing tree is balanced. An unbalanced routing tree will cause some nodes' batteries to be depleted more rapidly than others, prematurely reducing sensor network fidelity and longevity.

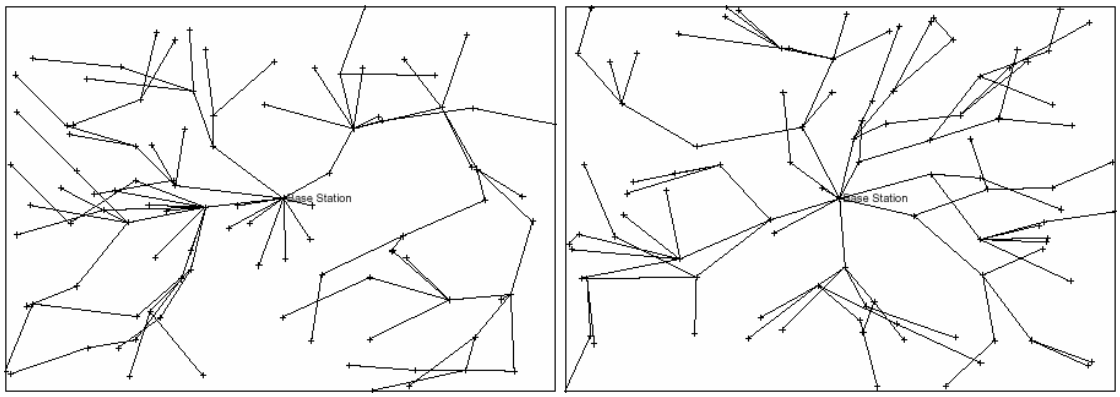

Figure 3 - Routing trees formed from random deployments of 100 nodes. Observe the centrally located base station. Nodes are represented by the + symbol, while hops between nodes are represented by lines.

Aside from conditions affecting radio performance, the most important factor affecting routing tree balance is the algorithm used to set up the initial routing tree. The algorithm used in this thesis was primitive. Any nodes that received tree-setup messages from multiple parents chose between them by selecting the one with the unique id numerically closest to its own. Tree-balancing algorithms exist, but the communication overhead is unacceptable in sensor networks' power-constrained environment. The development of more sophisticated tree-balancing algorithms that do not require a large communication overhead is an area for future research.

3.2.3. Message Broadcast

Whenever messages were sent in a sensor network, every node within radio range of the sending-node could receive the message. Child nodes could listen to their parent to verify that any sensed data was forwarded up the routing tree, towards the base station. If a child node observed that its parent failed to forward its data multiple times, the child could assume a status of orphaned and send out a message requesting a new parent. Any node hearing a request for a new parent would respond, and the orphaned node could select the new parent with the fewest hops to the base station.

3.3. Power Model

Simulations were run for sensor networks using cryptographic and image processing applications. For each application, ECE graduate student Jason Brandon

provided this thesis with processing power and data size parameters for several hardware platforms:

- General Purpose Processor (GPP)
- Field Programmable Gate Array (FPGA)
- Small Scale Dynamically Reconfigurable Hardware (SSR)
- Application-Specific Integrated Circuit (ASIC)

The GPP and FPGA platforms offer increased flexibility and decreased efficiency, while the ASIC platform maximizes efficiency and does not provide any flexibility. SSR hardware attempts to bridge the gap between FPGA and ASIC hardware. Jason Brandon also provided data on reasonable amounts of energy that the batteries of a sensor network might be able to provide.

3.4. Future Work

The execution times for the simulations in this thesis were inconveniently long. Individual simulator runs with a single random seed finished in only a few minutes. To get reliable data, a large number of runs must be executed and the results need to be averaged. This requirement lends itself to parallelization and distribution to multiple machines for execution. A framework for parallelizing simulation runs would enable a much faster turnaround for results, and fine-tuning network parameters could proceed at a much faster rate.

4. Applications

In order to understand the gains in sensor network longevity and fidelity made by incorporating SSR hardware, an analysis of the target applications is necessary. The applications considered for this thesis were JPEG image compression and encryption. Algorithms for JPEG image compression and encryption are naturally adaptable. JPEG compression can sacrifice image quality to reduce processing and the resulting data size. Cryptographic applications can be adapted to perform fewer rounds of encryption, trading off the desired level of security for the amount of computation required.

JPEG image compression and cryptography were chosen because of the different kinds of strain they place on sensor networks. JPEG image compression works with large amounts of data, while encryption uses small amounts of data but performs a disproportionately large amount of computation. Analysis of these two applications allowed identification of the relative impact of the processor power use on large-data and small-data applications. Applications that do not require significant processor power will not benefit appreciably from more efficient processing hardware.

4.1. JPEG Image Compression

JPEG image compression is a parameterizeable algorithm. Figure 4 shows the impact of different JPEG quantization levels on network fidelity and longevity. The vertical axis shows the number of nodes whose image reaches the base station in response to each request. JPEG 8 provides the best image quality and the worst data compression.

JPEG 1 provides the worst image quality and the best data compression. The sensor network started with 400 nodes randomly scattered over an area, all of which were able to transmit their images to the base station via the routing tree described in section 3.2.2. As parents' batteries are depleted, the routing tree automatically adapts to find a new parent who can forward messages to the base station. A new parent is not always available, however, and network fidelity suffers.

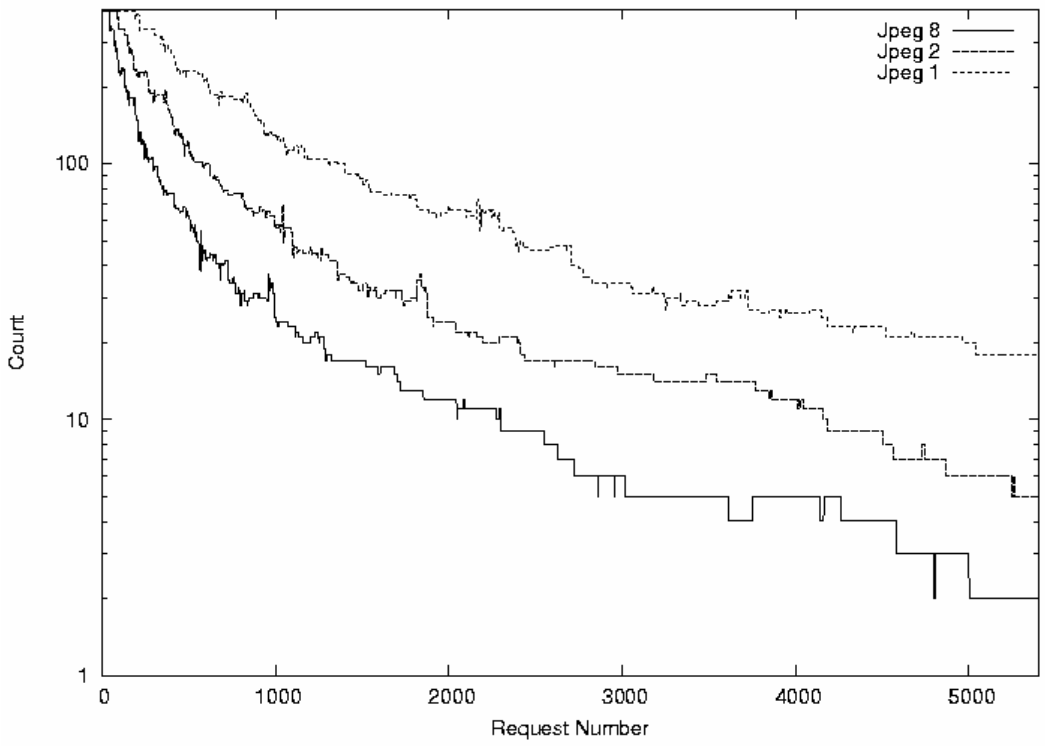

Figure 4 - Network fidelity for different JPEG image compression parameters. Results are the average of 12 simulated executions. Normalizing the compressed transmitted image size at 1 for JPEG 1, JPEG 2 and JPEG 8 were of size 2.34 and 4.68, respectively (Panigrahi, 2002).

Since data transmission dominates in this application, sensor networks using JPEG 1 compression make smaller transmissions and keep a greater number of nodes

alive over time. Using 100 images received at the base station as a threshold for required network fidelity, the JPEG 1 algorithm lasts for 1300 requests, while JPEG 8 is only able to serve 300 requests. Adaptable sensor networks can use JPEG 1 until something interesting is observed, and then the base station can command only the nodes observing the interesting phenomena to switch to JPEG 4 or 8, improving fidelity while minimizing the impact on longevity.

4.2. Encryption

Most modern symmetric ciphers, such as the Advanced Encryption Standard (AES), can be parameterized to control the number of encryption rounds (Garrett, 159). However, increasing security requires additional time and energy to perform additional rounds of encryption. Figure 5 shows the result of adjusting the number of encryption rounds on the longevity of a sensor network application. Cryp64-n represents encryption with 2^{n+1} rounds. For example, Cryp64-4 is 32 rounds.

The behavior of the cryptographic application in Figure 5 is quite different from the JPEG application in Figure 4. In the JPEG application, data transmission dominates power consumption, so that nodes nearest the base station have to transmit significantly more data than nodes farther away. Thus, nodes closest to the base station deplete their power supplies first. In the cryptographic application, processing dominates power consumption, so that all nodes exhaust their power supplies at nearly the same time. This is especially clear with Crypt64-4, where only 20 requests are served between the time of the first and last node's batteries being depleted. As the strength of encryption decreases,

the amount of power consumed by processing becomes proportionately less. Cryp64-1 illustrates this; the fidelity of the sensor network drops off gradually at first as nodes nearest the base station fail due to significant power consumption by both transmission and processing.

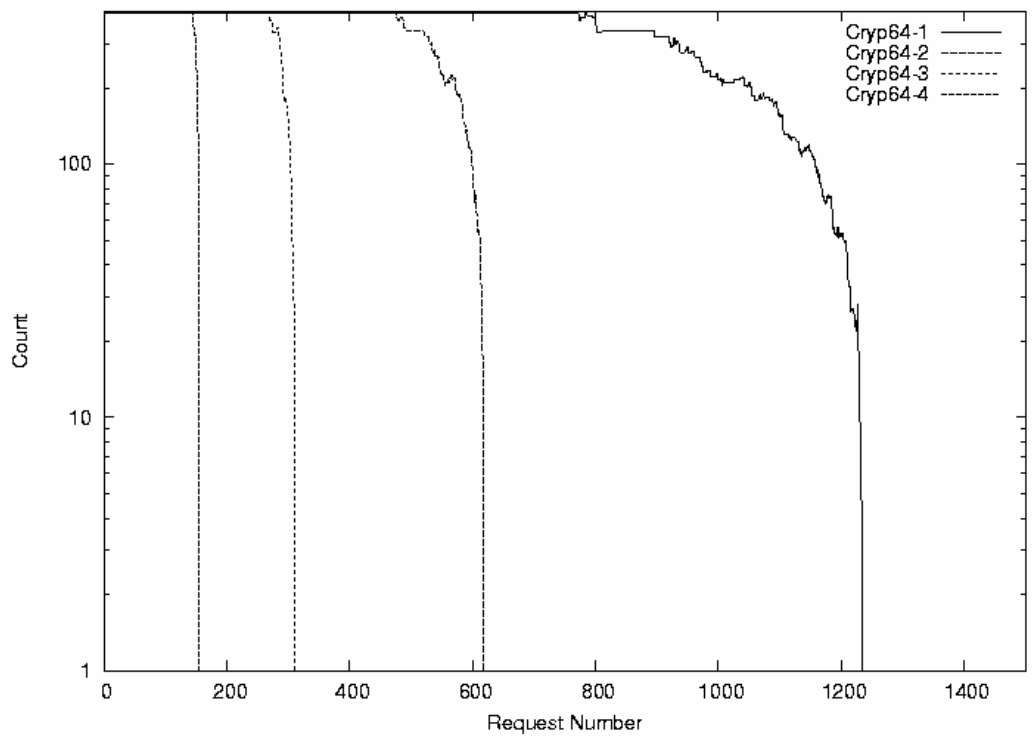

Figure 5 - Effect of encryption strength on sensor network longevity.

5. Implementing Adaptability

To further increase network fidelity and longevity, sensor nodes can be implemented with more efficient hardware designs. For JPEG image compression and cryptographic applications, the sensor node hardware can be specialized to maximize the efficiency of a sensor network while maintaining adaptability.

5.1. JPEG Image Compression

In Chapter 4, the advantages of adapting the JPEG image compression algorithm are clear. Implementing this application in hardware can yield significant advantages if adaptability can be maintained. Figure 6 shows the performance of a sensor network implementing the JPEG application on various hardware platforms. Although sensor networks executing the JPEG application are dominated by transmission power, it is apparent that the processing cost of image compression cannot be ignored. The network fidelity is reduced on the less efficient hardware and hardware / software implementations.

The ASIC implementation, while excellent in terms of network lifetime, cannot adapt to change power modes at all. The GPP and FPGA implementations had the worst performance, since they include excessive flexibility. The SSR platform came close to achieving the efficiency of the ASIC, with the significant difference that the SSR hardware is fully capable of adapting to change power modes.

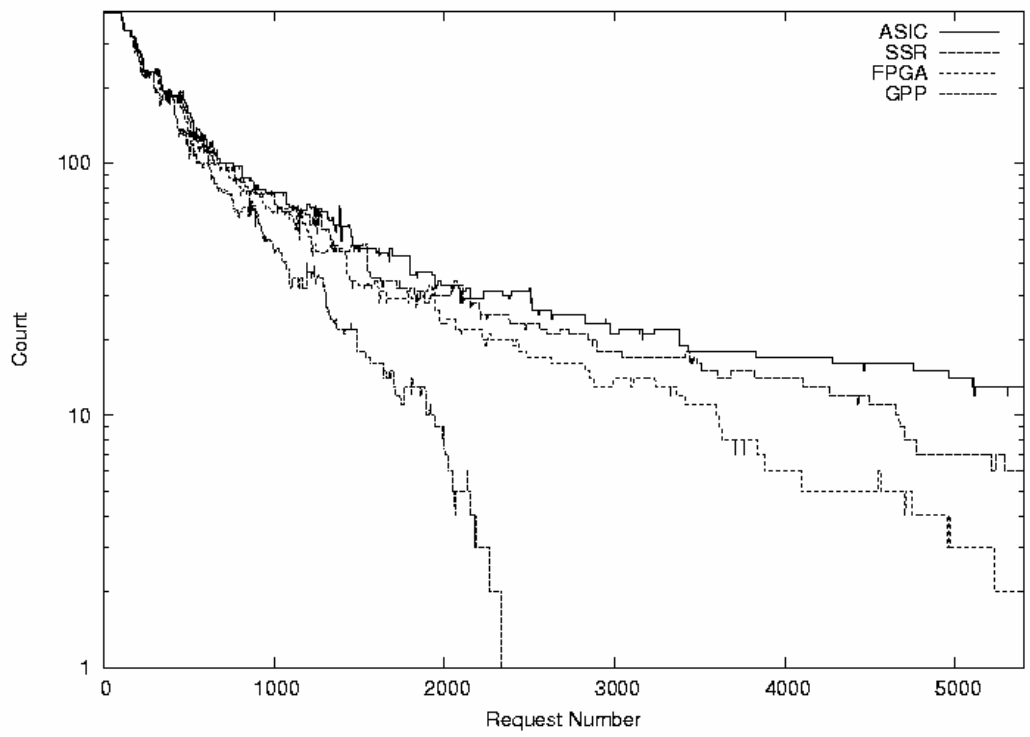

Figure 6 - Sensor network fidelity with different node implementations for JPEG image compression.

5.2. Cryptography

Applications that require significant processing and have smaller transmission sizes benefit dramatically from more efficient node implementations. Figure 7 shows the effects of sensor node implementation on network performance for a cryptographic application requiring a large amount of computation but with relatively low message sizes. The ASIC implementation was not included because it does not decrease in the range in the figure.

Using 100 readings received at the base station as a threshold for acceptable network performance, SSR, FPGA, and GPP return results for approximately 200, 500, and 1000 base station requests, respectively. This is a noteworthy result for SSR

hardware, as it offers a full 100% improvement over the FPGA hardware adaptable implementation.

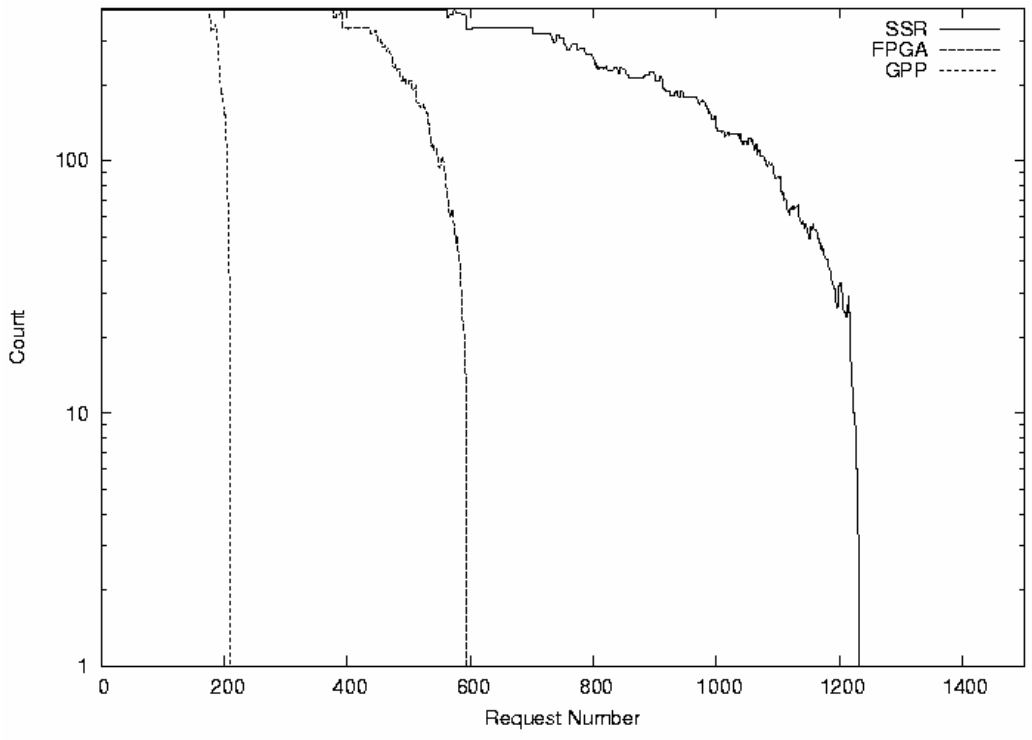

Figure 7 - Sensor network fidelity with different node implementations for encryption application.

5.3. Future Work

Additional research into the specific energy requirements of the hardware modeled in this thesis will enable more accurate simulation results. Sensor nodes actually need to be built with SSR hardware so that real – not just simulated – data can be gathered on their performance.

6. Conclusion

Sensor networks must take advantage of adaptability to be successful. On sensor nodes with GPPs, applications can take advantage of parameterizeable algorithms to minimize necessary computation. Further improvements can be made by implementing the algorithms in reconfigurable hardware. Even in applications where transmission costs dominate, such as JPEG image compression, energy savings obtained from using a more efficient processing implementation are significant. In applications where processing costs dominate, such as encryption, improvements of well over 100% in terms of network longevity can be gained by switching from GPPs to SSR hardware. SSR hardware is an optimal design choice because algorithms can be implemented with efficiency approaching that of ASICs while maintaining adaptability.

Additional experiments need to be run to gain a better understanding of the exact relationship between processing and transmission power for most potential sensor-network applications. The simulation framework created for this thesis is capable of providing results for these other applications if appropriate input data is available.

The simulations created for this thesis project demonstrated with a very high degree of confidence that adaptable sensor networks built with SSR hardware can extend the longevity of sensor networks by adapting fidelity. The myriad of practical applications for sensor networks provides a strong impetus to continue research into more efficient designs. Although sensor networks have been plagued by power limitations to date, the work presented above demonstrates that dramatic improvements can be made.

SSR hardware is an emerging technology, and there is every reason to believe SSR hardware will continue to improve.

Works Cited

Denning, Peter J. (Eds.). The Invisible Future. New York: McGraw-Hill, 2002.

Garrett, Paul. Making, Breaking Codes: An Introduction to Cryptology. Upper Saddle River, New Jersey: Prentice Hill, 2001.

Heinzelman W. R., Chandrakasan A., Balakrishnan H. "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", Hawaii International Conference on System Sciences Jan. 2000. 11 Oct. 2002 <http://www.mtl.mit.edu/research/icsystems/uamps/pubs>.

Intanagonwiwat, C., Govindan, R., & Estrin, D. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks." Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'2000). Boston, Massachusetts: MOBICOM, Aug. 2000.

Krishnamachari, B., Estrin, D., Wicker, S. "Impact of Data Aggregation in Wireless Sensor Networks." International Workshop on Distributed Event-Based Systems 2002. 9 Oct. 2002. <http://www.krishnamachari.net/papers/DataAggregationSensorNetworks2.pdf>

Lach, J., Evans, D. "Dynamically Adaptable Distributed Sensor Networks." NSF Proposal, Sensors and Sensor Networks. University of Virginia, March 2003.

Liu, J., Cheung, P., Guibas, L., and Zhao, F. "A Dual-Space Approach to Tracking and Sensor Management in Wireless Sensor Networks." ACM International Workshop on Wireless Sensor Networks and Applications Workshop. Atlanta: September 2002.

Loizeaux, J. "Building the Swarm: A Review of Three Areas Necessary for Swarm Computing." Undergraduate Thesis, University of Virginia, Apr. 2001.

MacKenzie, D., Wajcman, J. (Eds.). The Social Shaping of Technology. Philadelphia: Open University Press, 1985.

Madden, S., Franklin, M. J., Hellerstein, J., & Hong, W. "TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks". Symposium on Operating Systems Design and Implementation. December 2002.

Min, R., Furrer, T., Chandrakasan, A. "Dynamic Voltage Scaling Techniques for Distributed Microsensor Networks." IEEE Workshop on VLSI 2000. April 2000 <http://citeseer.nj.nec.com/min00dynamic.html>

Moore, Gordon E. "The Continuing Silicon Technology Evolution Inside the PC Platform." 5 April 2003. <http://www.intel.com/update/archive/issue2/feature.htm>.

Panigrahi, D., Taylor, C. N., Dey, S. "A Hardware / Software Reconfigurable Architecture for Adaptive Wireless Image Communication." International Conference on VLSI Design, 2002.

Perrig, A., Szewczyk, R., Wen, V. Culler, D., & Tygar, J. D. "SPINS: Security Protocols for Sensor Networks." Mobile Computing and Networking. Rome, Italy: MOBICOM, 2001.

Slijepcevic, S., Potkonjak, M., Tsiatsis, V., Zimbek, S., & Srivastava, M. "On Communication Security in Wireless Ad-Hoc Sensor Networks." Proceedings of the Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02). IEEE, 2002.

Takai, M., Bajaj, L., Ahuja, R., Bagrodia, R., Gerla, M. "GloMoSim: A Scalable Network Simulation Environment," Technical report 990027, UCLA, Computer Science Department, 1999.

Warneke, B., et al. "Smart Dust: Communicating With a Cubic-Millimeter Computer." Computer 34 (2001): 44-51.

Zeng, X., Bagrodia, R., Gerla, M. "GloMoSim: A Library for Parallel Simulation of Large-scale Wireless Networks." Proceedings of the 12th Workshop on Parallel and Distributed Simulations. May 1998.