Power-Efficient Adaptable Wireless Sensor Networks

John Lach, David Evans, Jon McCune, Jason Brandon
University of Virginia

Distributed wireless sensor networks are expected to have widespread applications within the coming decades, ranging from military tracking and emergency response to habitat monitoring and environmental tracking. These networks must be capable of adapting to changing environments and requirements. A sensor network application may need to alter its behavior to manage limited resources more efficiently, recover from broken network links, or change its functional behavior in response to commands issued by an operator.

Since sensor nodes (complete with sensing data collection, processing, and transmission) are untethered and typically must run on small batteries, a primary factor in determining the utility of a distributed sensor network is how well it manages energy. Hence, the adaptations considered here focus on allowing a sensor network to adapt energy consumption in ways that increase longevity in exchange for reduced fidelity, increased latency or weakened security.

Currently, most sensor nodes are software based, which provides the flexibility necessary for adaptation. A variety of programs can be stored in a node's local memory, or a base station can wirelessly distribute programs for necessary adaptations. However, a processor executing software is far less efficient (in terms of energy consumption, manufacturing cost per unit, and performance) than a fixed-logic ASIC. On the other hand, an ASIC does not have the flexibility for node-level adaptation.

This paper explores the use of field-programmable hardware in sensor nodes, which provides node flexibility with significantly greater energy efficiency than software. Just as with the software-based approach, node behavior and non-functional properties can be altered using locally stored or broadcasted configuration data. Experimental results reveal that node adaptability is maintained with significant energy efficiency improvements (and therefore increased network lifetime) over processor-based nodes.

Motivational Example

For a simple example of the need for adaptation in a distributed wireless sensor network, consider a network that monitors an environment with cameras. Each node in the network takes pictures of its surrounding area, compresses each image using JPEG compression, and transmits the data to its parent in the network. While each leaf node only transmits its own images, a parent node must also transmit the images of all of its descendents. Hence, a high-level node with many children, grandchildren and great-grandchildren is responsible for transmitting a large amount of data. The nodes closest to the base station (and highest in the routing tree), will have the most data to transmit and will quickly exhaust their available energy. After the highest-level nodes have been depleted, useful information can no longer reach the base station.

Adapting the network and its nodes for energy savings can increase its longevity. At the network topology level, the routing tree could be transformed to take the data forwarding pressure off of nodes with a large number of descendents. In addition, parents could use image processing techniques to aggregate images from their children into a single image or could use filtering to selectively forward only interesting images. At the node level, the image compression can be parameterized for different image qualities and compression ratios to trade off energy consumption and image fidelity. Nodes can remain in a low power mode (highest compression, lowest image quality) until the base station commands a group of nodes to switch to a higher resolution. More details on the benefits of wireless sensor network adaptation will be provided in the full paper.

Much distributed sensor network research uses node models based on the MICA motes developed at UC Berkeley and marketed by Crossbow Technology. Each node contains an Atmega 128L processor, which is a low-power microcontroller running the TinyOS operating system [Col02, Cross03]. These software-based nodes have the necessary flexibility for the JPEG compression adaptation. However, the virtually unlimited flexibility provided by the software comes at a price, as software algorithms executed on processors are less energy efficient than fixed-logic ASICs, which do not possess the flexibility needed for adaptation.

The Case for Adaptable Hardware

We considered a node implemented with a small amount of application-specific field-programmable logic and interconnect (e.g. SRAM-based lookup tables (LUTs), SRAM-gated pass transistors, multiplexors, etc). Given that only portions of a node need be flexible for adaptation, only small amounts of programmability are necessary to achieve the required flexibility. This small-scale reconfigurable (SSR) design approach helps to avoid much of the area, delay, and power penalties associated with general-purpose reconfigurable arrays (i.e. FPGAs). As a result, such nodes provide the flexibility needed for network adaptation with more energy efficiency than FPGAs and significantly greater efficiency than software. More details on SSR are included in the full paper.

Using the wireless network simulator GloMoSim [Zeng98], we simulated a network performing the image-based monitoring application described above for a variety of node implementations. The network starts with 400 nodes that are all able to have their images reach the base station through the routing tree. As nodes run out of energy, they are no longer able to transmit their own or their descendents' images. When a parent's energy is depleted, the routing tree automatically adapts to have its children find different parents to whom to send their images, but a live parent node is not always reachable.

Figure 1 shows the performance of such a network with various implementations of the sensor node. Each node has a constant compression ratio and performs image aggregation at a ratio of $\sqrt{n}$, where n is the number of images at each node. Although this particular application is dominated by transmission energy consumption (which is assumed to be constant across the node implementations), it is clear the energy required for the JPEG compression processing has an effect on the network fidelity, as the software-based node has a reduced fidelity due to the failing of energy-depleted nodes. FPGAs fare better, but SSR is only outperformed by the fixed-logic ASIC, which does not have the flexibility for adaptation.

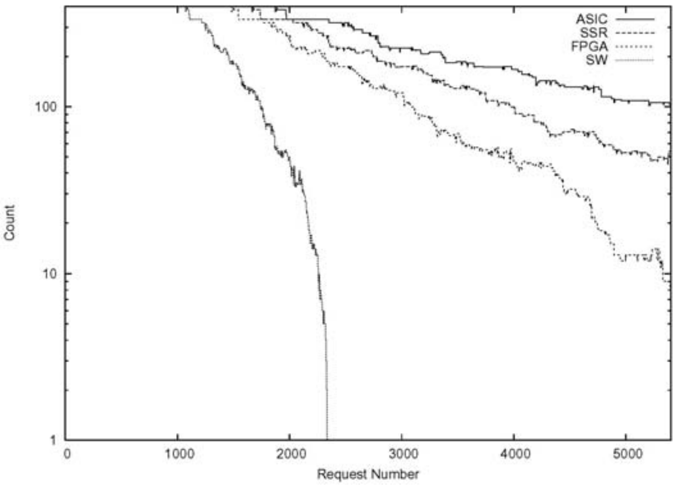

Figure 1. Network fidelity with different node implementations. The vertical axis is the number of nodes whose image reaches the base station after a request (shown on the horizontal axis). The processing energy consumption of SSR-based nodes is normalized to 1, and the ASIC, FPGA, and software node processing energy consumptions are 0.01, 2, and 6, respectively, based on measurements from [Pan02]. Results are the average of 12 simulated executions.

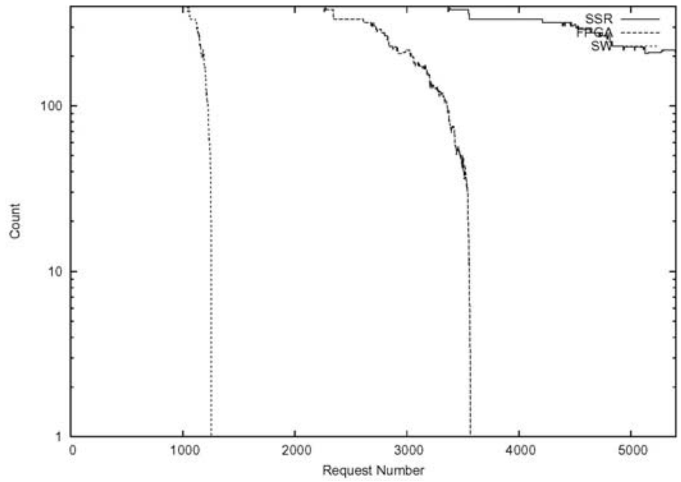

Figure 2. Network fidelity for an application requiring a large amount of computation to perform encryption, but relatively low message sizes. (Energy consumption ratios and simulation approach are the same as in Figure 1.)

The advantages of more efficient node implementations are even clearer for applications that require a significant amount of processing at each node and smaller transmission sizes than JPEG images. For example, consider a network with encrypted communications. Each transmission must be encrypted, but the number of rounds may be made variable to trade off security and processing energy, just as various JPEG compression ratios may be used for different power modes. This also requires that the nodes be adaptable. Figure 2 shows the network fidelity for such an application using different node implementations. The high energy consumption of software-based nodes results in the network collapsing after only 200 requests. As expected, FPGA-based nodes perform significantly better than software but are clearly outperformed by the SSR implementation. While ASIC-based nodes would consume even less energy, they would not have the flexibility to reap the benefits of adaptation discussed above.

From these results, it is clear that a node implementation based on small amounts of programmable logic can greatly increase the fidelity of an application and the lifetime of a distributed wireless sensor network while still providing the means for network and node adaptation. The full paper will examine the details of the SSR-based node implementation and other applications requiring or benefiting from adaptation.

References

[Col02] S. Coleri, M. Ergen, T.J. Koo, “Applications and OS: Lifetime analysis of a sensor network with hybrid automata modeling,” International Workshop on Wireless Sensor Networks and Applications, 2002.
[Cross03] Crossbow Technology, “MICA wireless measurement system datasheet,” 2003.
http: //www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf
[Pan02] D. Panigrahi, C.N. Taylor, S. Dey, “A hardware/software reconfigurable architecture for adaptive wireless image communication,” International Conference on VLSI Design, 2002.
[Zeng98] X. Zeng, R. Bagrodia, M. Gerla, “GloMoSim: A library for parallel simulation of large-scale wireless networks,” Proceedings of the 12th Workshop on Parallel and Distributed Simulations, May 1998.