Report Number: WUCSE-2006-61

2006-01-01

Towards a Unified Radio Power Management Architecture for Wireless Sensor Networks

Kevin Klues, Guoliang Xing, and Chenyang Lu

In many wireless sensor networks, energy is an extremely limited resource. While many different power management strategies have been proposed to help reduce the amount of energy wasted, application developers still face two fundamental challenges when developing systems with stringent power constraints. First, existing power management strategies are usually tightly coupled with network protocols and other system functionality. This monolithic approach has led to standalone solutions that cannot easily be reused or extended to other applications or platforms. Second, different power management strategies make different and sometimes even conflicting assumptions about the rest of the system with which they... **Read complete abstract on page 2.**

Towards a Unified Radio Power Management Architecture for Wireless Sensor Networks

Kevin Klues, Guoliang Xing, and Chenyang Lu

Complete Abstract:

In many wireless sensor networks, energy is an extremely limited resource. While many different power management strategies have been proposed to help reduce the amount of energy wasted, application developers still face two fundamental challenges when developing systems with stringent power constraints. First, existing power management strategies are usually tightly coupled with network protocols and other system functionality. This monolithic approach has led to standalone solutions that cannot easily be reused or extended to other applications or platforms. Second, different power management strategies make different and sometimes even conflicting assumptions about the rest of the system with which they need to interact. Without knowledge of which strategies are interoperable with which set of network stack protocols it is dificult for application developers to make informed decisions as to which strategy is most appropriate for their particular application. To address these challenges, we propose a Unified Power Management Architecture (UPMA) that supports the flexible composition of different power management strategies based on application requirements. We envision this architecture to consist of both low level programming interfaces, as well as high level modeling abstractions. These abstractions characterize the key properties of different applications, network protocols, and power management strategies. Using these properties, configuration tools can be created that match each application with the most appropriate network protocol and power management strategy suited to its needs.

Department of Computer Science & Engineering

2006-61

Towards a Unified Radio Power Management Architecture for Wireless Sensor Networks

Authors: Kevin Klues, Guoliang Xing, and Chenyang Lu

Corresponding Author: klueska@cs.wustl.edu

Web Page: http://www.cs.wustl.edu/~lu/upma.html

Abstract: In many wireless sensor networks, energy is an extremely limited resource. While many different power management strategies have been proposed to help reduce the amount of energy wasted, application developers still face two fundamental challenges when developing systems with stringent power constraints. First, existing power management strategies are usually tightly coupled with network protocols and other system functionality. This monolithic approach has led to standalone solutions that cannot easily be reused or extended to other applications or platforms. Second, different power management strategies make different and sometimes even conflicting assumptions about the rest of the system with which they need to interact. Without knowledge of which strategies are interoperable with which set of network stack protocols it is dificult for application developers to make informed decisions as to which strategy is most appropriate for their particular application.

To address these challenges, we propose a Unified Power Management Architecture (UPMA) that supports the flexible composition of different power management strategies based on application requirements. We envision this architecture to consist of both low level programming interfaces, as well as high level modeling abstractions.

Type of Report: Other

Towards a Unified Radio Power Management Architecture for Wireless Sensor Networks

Kevin Klues, Guoliang Xing*, Chenyang Lu
Department of Computer Science and Engineering
Washington University in St. Louis
{klueska, xing, lu}@cse.wust.edu

Abstract

In many wireless sensor networks, energy is an extremely limited resource. While many different power management strategies have been proposed to help reduce the amount of energy wasted, application developers still face two fundamental challenges when developing systems with stringent power constraints. First, existing power management strategies are usually tightly coupled with network protocols and other system functionality. This monolithic approach has led to standalone solutions that cannot easily be reused or extended to other applications or platforms. Second, different power management strategies make different and sometimes even conflicting assumptions about the rest of the system with which they need to interact. Without knowledge of which strategies are interoperable with which set of network stack protocols it is dificult for application developers to make informed decisions as to which strategy is most appropriate for their particular application.

To address these challenges, we propose a Unified Power Management Architecture (UPMA) that supports the flexible composition of different power management strategies based on application requirements. We envision this architecture to consist of both low level programming interfaces, as well as high level modeling abstractions. These abstractions characterize the key properties of different applications, network protocols, and power management strategies. Using these properties, configuration tools can be created that match each application with the most appropriate network protocol and power management strategy suited to its needs.

1 Introduction

A multitude of radio power management protocols have been developed for wireless sensor networks (WSNs). Without a unified framework within which to develop them, however, different solutions have taken different approaches and made different assumptions on how they should interact with the rest of the system. It is often very hard or sometimes even impossible to exchange the use of one protocol for another. This lack of system support for allowing different power management protocols to be flexibly integrated into an otherwise fully operational system has hindered the progress of WSN research and development in more than one respect. First, different applications often benefit more from one type of power management protocol than another. Without the ability to easily select a particular protocol for use by a particular application, developers often waste time fumbling with low level implementation details instead of concentrating on the development of their applications at hand. Second, researchers developing these protocols have never really had a fair way of comparing the power savings achieved by one protocol over another. These comparisons have often been skewed by differences in the systems on which each protocol was being run, resulting in inconsistent energy consumption readings, and confusion over how these results should be interpreted.

To address each of these issues, we propose a *Unified Radio Power Management Architecture (UPMA)*. This architecture aims to support the flexible integration of different power management protocols with a diverse set of applications and platforms. A novel feature of this architecture is that it enables power management protocols existing across multiple layers to be composed together in order to provide a single radio power management solution for an entire WSN system. We envision UPMA to consist of three basic components: *Unified Architectural Abstractions, High-level Modeling Abstractions, and a set of Configuration and Analysis tools*.

- The *Unified Architectural Abstractions* will facilitate the integration of different power management strategies. In contrast to the monolithic approaches adopted by existing solutions, these abstractions separate power management strategies from basic network protocols, enabling them to coordinate across multiple layers as well as work together in the presence of multiple applications.
- The *High-level Modeling Abstractions* characterize the key properties of different applications, hardware platforms, and power management protocols. Their presence allows one to perform systematic analysis and composition of multiple power management protocols together.
- *Configuration and Analysis Tools* automate the integration of power management strategies by: selecting and configuring an overall power management strategy that matches the characteristics of a particular application and network setup, ensuring the compatibility between each power management protocol and the network protocols satisfying any desired network qualities,

*Now with the Department of Computer Science, City University of Hong Kong and jointly optimizing and composing multiple power management protocols together in order to produce an overall integrated power management solution.

The rest of this paper is organized as follows. Section 2 provides an overview of some existing power management protocols as well as some of the open challenges motivating the development of UPMA. Section 3 presents details on the different components that UPMA should include. Section 4 identifies the open questions that exist for the development of these components and provides insight into how they might be solved. Finally, Section 5 concludes the paper.

2 The Case for UPMA

The goal of the SNA project at UC Berkeley has been to create an all encompassing architecture for wireless sensor networks that supports the development of a diverse set of protocols and applications that can be run on a multitude of different hardware platforms. In the original SNA proposal [2], the issues surrounding power management in WSNs were addressed in an abstract manner. The general consensus was that power management should be cross-layer in nature, but there was no formal discussion of how this could or should be done. The ideas presented in this paper complement the SNA project by developing a unified and configurable power management architecture that can be integrated with existing SNA components.

In order to understand how such an architecture might be realized, one must first understand how radio power management protocols themselves actually operate. Most of the energy consumed by a radio takes place while operating in its transmission, reception, or idle listening states. In order to conserve power in each of these states, two different techniques have traditionally been used. *Power control* protocols reduce the transmission power at which individual packets are sent through the network, and *sleep scheduling* protocols schedule nodes to sleep in order to reduce the energy wasted by idle listening. One limitation of the techniques used for sleep scheduling is that they tend to increase the overall communication delay in the network. In order to mitigate this performance penalty, some networks choose to maintain a connected backbone at the cost of increased power consumption. *Backbone maintenance* protocols select a small set of nodes that always remain active in order to forward packets, while all other nodes run sleep schedules in order to conserve energy [3, 19, 20].

Despite the significant effort spent developing these protocols, the following significant challenges still remain in meeting the energy constraints of different applications and varying network conditions.

Flexibility: Current power management strategies often adopt monolithic implementations in which power management is tightly coupled with a particular network protocol stack. As a result, a system is often limited to specific power management strategies that cannot be easily extended or replaced. For example, sleep scheduling is often implemented as part of MAC protocols while power control is often integrated with routing or topology maintenance protocols. The implementations of low-power MAC protocols such as S-MAC and B-MAC could share many of the same underlying radio stack functionality (such as clear channel assessment (CCA)). Furthermore, many routing protocols often use specific power control schemes to compute a set of routing metrics based on transmission power. It would be more flexible to separate the power control functionality from any specific routing protocols that implement them, and simply provide an interface to fetch the values of any cost metrics. Higher level services like TinyDB [13] also employ multiple power management strategies that stretch across multiple layers and are specifically designed to work together. Although each of these monolithic approaches are often times slightly more computationally efficient, they have largely impeded the interoperability of different power management protocols, and the overall synergy between different research efforts.

Configurability: Existing power management protocols are often geared for a particular type of application or platform. A key challenge is to choose and configure the proper power management protocol for whatever scenario exists. Power control only conserves transmission power and hence is effective only when the network workload is so high (or the idle power of the radio is so low) that the transmission energy dominates the overall energy consumption of the network. On the other hand, sleep scheduling is only suitable for applications with low workloads or radios with relatively high idle power. Furthermore, some sleep scheduling protocols [4, 15] have been specifically designed for data collection applications that impose periodic low network traffic, while backbone maintenance protocols are designed for applications (e.g., real-time detection and tracking) in which message delivery latency is extremely important. Most of these protocols assume homogeneous networks that may not be effective when nodes have heterogeneous communication capacity and very low power budgets. Choosing and configuring the right power management strategy often requires careful analysis of the characteristics of the application, the network platform, and the power management protocols themselves.

Composability: While current research efforts mainly focus on the use of a single power management protocol, a unified architechture is needed to effectively compose different protocols together to form a single coherent power management solution. Each individual protocol may be sub-optimal since it only reduces the energy consumption in some subset of its radio states. Power control only reduces the transmission power of nodes, while sleep scheduling reduces the idle power. In order to minimize the total energy consumption of a network, application developers must effectively integrate the use of different power management protocols across different layers. Our experience shows that the optimal integration of different power management protocols requires careful cross-layer consideration of the radio characteristics, routing choices, and network workload imposed by the application. The existence of a unified architecture within which these tasks could be performed would be very benificial.

3 High Level Design of UPMA

Our proposal for a unified radio power management architecture consists of more than just a set of low level architectural abstractions. It also includes the development of a set of modeling abstractions and configuration tools that can be used to facilitate the design and deployment of integrated power management configurations more easily. This section describes each of these components in more detail.

3.1 Architectural Abstractions

The architectural abstractions for UPMA have been designed with the following principles in mind. (1) They should support the development of a multitude of different power management protocols, each having their own independent implementations. (2) They should contain a set of standardized interfaces between all power management protocols and any other components in the system. (3) They should allow components to be integrated into the architecture that are capable of perfroming cross-layer coordination between power management protocols existing at different layers.

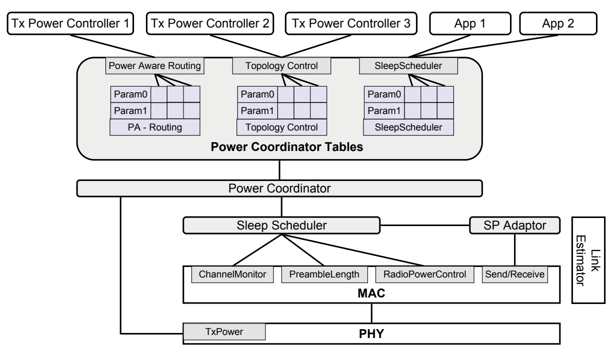

Figure 1. The proposed Unified Radio Power Management Architecture.

As shown in figure 1, our design of UPMA is capable of satisfying each of these requirements. (1) Power management protocols exist as independent entities at both the network layer as well as the data link layer. (2) Communication takes place between these protocols and other components in the system through a standard set of interfaces. (3) Cross coordination can be achieved through the proper implementation of different *Power Coordination Tables* and their corresponding *Power Coordinator* component.

The standard set of interfaces that have been defined were chosen to encapsulate the requirements of the representative power management protocols presented in section 2. Interfaces have been defined for both sleep scheduling protocols at the data link layer as well as power control protocols at the network layer. Sleep scheduling protocols need to be able to (1) turn the radio on and off (*PowerControl*) (2) perform clear channel assessment on the radio channel (*ChannelMonitor*), and (3) set the preamble length of an outgoing packet (*PreambleLength*). Power control protocols need to (1) set the transmission power level that a packet should be transmitted at (*TxPower*), and (2) specify the routing cost for use by network protocols existing in the system (*Cost*)

The Power Coordinator and its corresponding Power Coordination Tables are configured differently based on which power management protocols are being used in the system. These components can be instantiated as necessary to meet the power constraints of any applications running on top of them. As a simple example, consider two applications specifying two different duty cycles for a single underlying sleep scheduling protocol. The Power Coordination Tables store the on and off times required for each duty cycle, and the Power Coordinator combines these values to produce a sleep schedule satisfying the on time requirements of both. A comprehensive evaluation of such a configuration has been performed in [10], with results indicating that flexibility is indeed increased without incurring a significant performance penalty.

A more complicated example might involve some sort of cross-layer optimization. The Minimum Power Configuration Protocol (MPCP) [17] dynamically minimizes the total power consumption of a network by jointly optimizing the sleep schedules and transmission powers of all nodes in a network. When network workload is low, the total power consumption of the network is dominated by the idle listening of nodes. In such a case, MPCP increases the transmission power of some nodes so that fewer active nodes are needed to forward data. Those nodes that do not need to forward data are allowed to follow sleep schedules. Conversely, MPCP reduces the transmission power of nodes when network workload is high, because in this case the majority of power is consumed by data transmission. Based on this functionality, the implementation of MPCP can be broken down into two interleaving components: a sleep scheduling component and a power-aware routing component. When a node starts routing a data flow, the sleep scheduler stops duty-cycling the node. This state transition triggers the power-aware routing component to optimize the transmission power of the node, and assign the node a lower routing cost. As a result, data tends to be routed through only those nodes that are currently active, resulting in less energy wasted by idle listening. Within the proposed architecture, MPCP can be realized by creating appropriate power control and sleep scheduling components, and implementing a cross-layer optimization protocol within the Power Coordinator component. Once the node starts routing a data flow, the Power Coordinator can be triggered to make sure that it modifies the values in the Power Coordination Tables to indicate that a node should always be powered on. At the same time, it can compute any new routing costs based on this change and update this value in the appropriate network protocol through the standard interface defined for this purpose.

3.2 High-Level Modeling Abstractions

The second key component of the UPMA project involves the development of high-level modeling abstractions. These modeling abstractions consist of a set of *profiles* that specify the key characteristics of different power management protocols, hardware platforms, and applications.

The characteristics of each power management protocol will be encapsulated inside of a *protocol profile*. This profile describes the properties of each protocol, as well as its dependency and compatibility with different application requirements, such as message delivery latency and network lifetime. It also specifies which functionality a protocol relies on from other system components as well as which ones it may be incompatible with. For example, adaptive sleep scheduling protocols [16, 22, 23] assume the use of contention-based MAC protocols that can provide clear channel assessment capabilities. On the other hand, time slot based sleep scheduling protocols [5, 8] naturally fit with slot based MAC protocols like the GTS mode of IEEE 802.15.4. Some power control protocols are incompatible with slot based sleep scheduling protocols because dynamic power adjustments change a node's neighborhood frequently. Consequently, many slot scheduling protocols must frequently reassign time slots to ensure that the schedules of neighboring nodes did not conflict with one another, resulting in significant overhead costs. With profiles of this type in place it will be possible to express each of these dependencies so that conflicts can be avoided.

The characteristics of a specific hardware platform will be encapsulated inside of a *platform profile*. This profile essentially describes the power characteristics of any radios existing on the platform. A standard way to model such information is through the use of a power state machine, annotated with the power consumption of each radio state (e.g., transmission/reception/idle/sleep/off), as well as the delay and power consumption associated with each state transition. Additionally, a platfrom profile should also include the types of energy sources supported by the platform (e.g., battery/harvested/wall-plugged), as well as their corresponding capacities.

The characteristics of each application will be encapsulated inside an *application profile*. Profiles of this type include characteristics describing the expected node density, the sampling rate to be used, and the expected workload imposed on the application. These profiles also include various performance requirements specified by the user (i.e. maximum message delivery latency, expected node lifetime, throughput, etc.). Each of these requirements plays a crucial role in determining which power management protocols are most appropriate for a given application.

3.3 Configuration Tools

We propose the development of a set of tools that can examine each of the profiles described above in order to automatically select and configure an appropriate set of power management protocols that best meet the demands of a particular application and hardware platform. Figure 2 illustrates how such tools could be used to perform this operation.

First, the *strategy selector* analyzes the application and hardware platform for which an appropriate power management solution should be generated. It then selects the profiles of a set of compatible power management protocols from a predefined library. These profiles are then fed into a *dependency and compatibility checker* that evaluates if they are able to work with each other and with other network protocols in the system. If the check fails, the process is restarted with the selection of a new set of compatible power management protocols. If the check is successful, the strategy files are passed to the *joint optimizer* and configured for use by the application.

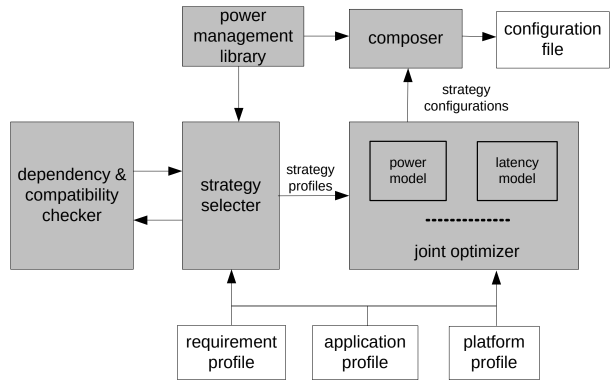

Figure 2. The process of designing an integrated power management configuration. The gray boxes represent the components of the toolset to be developed.

The optimizer first retrieves the performance models from each protocol profile and aggregates them into a single model that captures the impact each of them has on system performance (e.g., power consumption, delivery latency, throughput, etc.). The optimizer then retrieves any necessary information from the application and platform profiles and inputs them into the aggregation models. Based on each aggregate performance model, the optimizer configures each strategy to match the performance requirements of the application. The optimizer may produce configurations at different granularities. Qualitative configurations may be generated to guide the application developer to optimize the run time performance of different power management protocols. Alternatively, quantitative configurations may be generated that expose the key tunable parameters of each power management protocol that has been selected.

The final stage in the configuration process involves the use of a *composer* tool. The composer tool is responsible for retrieving the actual implementations of any protocols described by the protocol profiles that have been selected. It then generates the code necessary for interconnecting these implementations with other components in the system using the architectural abstractions described in section 3.1. This code can be used to assist application developers wishing to incorporate the use of any power management protocols into their design.

4 Research Issues

The architecture and all of the modeling abstractions and configuration tools described throughout this paper only exist as preliminary designs and a number of interesting research questions still remain.

4.1 Cross Layer Coordination

Sometimes it may be appropriate for the *Power Coordinator* to trigger the collection of updated values to be inserted into the *Power Coordination Tables*. Other times, an application or power control protocol may wish to trigger the *Power Coordinator* to reevaluate its coordination policy. For example, the use of a power aware routing protocol may require an update of transmission power levels just before sending a packet, while a sleep scheduling protocol may require an update every time a new value is inserted into the Power Coordination Tables. Topology control protocols, on the other hand, might only want to perform an update of transmission power levels whenever a change in topology is required. An important research task is, therefore, to develop efficient interfaces and mechanisms to support effective coordination between various components in the architecture.

4.2 Modeling Abstractions

A key challenge is to design scalable modeling abstractions that adapt to new applications, platforms, and power management strategies as they continue to emerge. One promising approach is to organize the various system attributes into a well-defined hierarchy so that the new system aspects can be easily modeled by inheriting existing attributes. For example, slot scheduling and power control should be defined as high level classes as the strategies in these two classes are not compatible with each other. The developer of a new slot scheduling protocol can then easily specify the compatibility attributes for the new protocol by inheriting existing strategy profiles of the same class. In addition, hierarchal attributes are easier to analyze, and hence simplify the design of analysis and configuration tools. Another key advantage of hierarchical modeling abstractions is that they allow for automatic analysis and configuration of the same set of power management protocols at different granularities.

Heterogeneous WSNs have shown the promise of improving performance and increasing lifetime [6,12,21]. However, the diversity of the environments and platforms in these networks introduce the challenge of accurately modeling and analyzing their system lifetimes. Therefore, the network topology, as well as the diverse power characteristics of different radios, must be taken into consideration in order to optimally configure power management strategies for heterogeneous WSNs. An important research task involves investigating appropriate models for heterogeneous WSNs and incorporating them into our analysis and configuration tools.

4.3 Joint Optimization

The stringent lifetime requirements of applications necessitate effective composition of multiple power management protocols into a complete power management solution. However, modeling and optimizing multiple protocols simultaneously is challenging due to correlations that exist between different protocols. While there exist analytical models for several specific power management protocols, a key research issue is to develop systematic approaches for composing multiple protocols together. It is also important to investigate the use of empirical models by leveraging on any existing experimental results of individual protocols. An advantage of empirical models is that they often reveal important run-time performance pitfalls that are hard to capture in analytical models.

A promising approach is to develop a set of *power management patterns* that are customized for representative classes of WSN applications such as generic object tracking [11], habitat monitoring [14], and structural monitoring [18]. A power management pattern should be comprised of suitable power management protocols that are carefully composed and optimized for a particular class of applications. Similar to design patterns commonly used in software development, customized power management patterns provide an effective way of documenting and sharing the existing knowledge of power management design for WSNs. Once these patterns have been defined, the configuration and analysis tools that get developed can be seamlessly integrated into any pre-existing end-to-end tool chains for WSN development. Examples of such development environments include Ptolemy [1] and SNACK [7].

5 Conclusion

We have proposed UPMA, a unified radio power management architecture for wireless sensor networks. UPMA is comprised of three innovative components: (1) unified architectural abstractions that facilitate the flexible integration of different power management strategies, (2) high-level modeling abstractions that characterize the key properties of applications, network platforms, and power management strategies, and (3) configuration and analysis tools that generate integrated power management strategies customized for given applications and networks. As the first step toward realizing UPMA, we have developed uniform link-layer abstractions to support flexible sleep scheduling on TinyOS 2.0 [10]. In the future, we plan to integrate UPMA with the Sensor Network Architecture [2] as well as the resource management framework of TinyOS 2.0 [9].

6 Acknowledgements

This work is supported by NSF NeTS-NOSS grant #CNS-0627126.

7 References

[1] P. Baldwin, S. Kohli, E. A. Lee, X. Liu, and Y. Zhao. Modeling of sensor nets in ptolemy ii. In *Proceedings of the third international symposium on Information processing in sensor networks (IPSN)*, 2004.

[2] U. Berkeley. a network architecture for wireless sensor networks. http://webs.cs.berkeley.edu/SNA/.

[3] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *MobiCom*, 2001.

[4] O. Chipara, C. Lu, and G.-C. Roman. Efficient power management based on application timing semantics for wireless sensor networks. In *International Conference on Distributed Computing Systems (ICDCS)*, 2000.

[5] S. C. Ergen. Pedamacs: Power efficient and delay aware medium access protocol for sensor networks. Master's thesis, University of California at Berkeley, 2002.

[6] O. Gnawali, K.-Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, and E. Kohler. The tenet architecture for tiered sensor networks. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 153–166, New York, NY, USA, 2006. ACM Press.

[7] B. Greenstein, E. Kohler, and D. Estrin. A sensor network application construction kit (snack). In *Proceedings of the 2nd international conference on Embedded networked sensor systems (Sensys)*, 2004.

[8] B. Hohlt, L. Doherty, , and E. Brewer. Flexible power scheduling for sensor networks. In *In Proceedings of the Third International Symposium on Information Processing in Sensor Networks (IPSN)*, 2004.

[9] K. Klues, V. Handziski, D. Culler, D. Gay, P. Levis, C. Lu, and A. Wolisz. Dynamic resource management in a static network operating system. Technical Report WUCSE-2006-56, Washington University in St. Louis, 2006.

[10] K. Klues, G. Xing, and C. Lu. Link layer support for flexible radio power management in wireless sensor networks. In *The 6th International Workshop on Information Processing in Sensor Networks (IPSN 2007)*, Cambridge, MA, April 25-27 2007.

[11] D. Li, K. Wong, Y. H. Hu, and A. Sayeed. Detection, classification and tracking of targets in distributed sensor networks. *IEEE Signal Processing Magazine*, 19 (2), 2002.

[12] Y. Ma and J. H. Aylor. System lifetime optimization for heterogeneous sensor networks with a hub-spoke topology. *IEEE Transactions on Mobile Computing*, 03(3), 2004.

[13] S. Madden, M. Franklin, J. Hellerstein, and W. Hong. The design of an acquisitional query processor for sensor networks. In *Proceedings of the 2003 ACM SIGMOD international conference on on Management of data*, pages 491 – 502, 2003.

[14] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *WSNA*, pages 88–97, 2002.

[15] N. Ramanathan, M. Yarvis, J. Chhabra, N. Kushalnagar, L. Krishnamurthy, and D. Estrin. A stream-oriented power management protocol for low duty cycle sensor network applications. In *In Proceedings of the Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*, 2005.

[16] T. van Dam and K. Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *Sensys*, 2003.

[17] G. Xing, C. Lu, Y. Zhang, Q. Huang, and R. Pless. Minimum power configuration in wireless sensor networks. In *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, 2005.

[18] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin. A wireless sensor network for structural monitoring. In *SenSys*, 2004.

[19] Y. Xu, J. Heidemann, and D. Estrin. Adaptive energy-conserving routing for multihop ad hoc networks. Research Report 527, USC, October 2000.

[20] Y. Xu, J. Heidemann, and D. Estrin. Geography-informed energy conservation for ad hoc routing. In *MobiCom*, 2001.

[21] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh. Exploiting heterogeneity in sensor networks. In *IEEE Transactions on Mobile Computing*, 2005.

[22] W. Ye, J. Heidemann, and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM*, 2002.

[23] R. Zheng and R. Kravets. On-demand power management for ad hoc networks. In *INFOCOM*, 2003.