

CS322/Project Final Report

Challenges to the architecture of Distributed Sensor Networks (DSNs): with comparison to traditional networks

by

Xuehai Zhang

December 2000

Abstract

Distributed sensor networks (DSNs) will produce high-quality environment information using large numbers of physical sensors with coordination with each other since advances in microelectromechanical system (MEMS) technology enable small and cheap nodes capable of sensing, communication and computation. These new technologies will enable exciting new applications that range from medical and home security to machine diagnosis, chemical/biological detection and other military application. While these applications require high performance from the network, they suffer from constraints caused by the DSNs that do not appear in more traditional networks. So, it is necessary to deeply explore the distinguished features of the DSNs, analyzing and evaluating their effect on the design of the network architecture.

This report does analysis on the distinguished features of the DSNs with comparison with traditional network. Based on these requirements, the report addresses some of the key design considerations for future microsensor system, including hardware, network architecture, communication and routing protocol, naming and addressing mechanism, application architecture and security.

1 Introduction

In recent years, we have seen the design of micropower wireless sensor system has gained importance for a variety of civil and military applications. Many consumer applications will be possible, including home automation, security, and irrigation, etc.

They combine micropower sensor technology with a low power sensor interface, signal processing and weak-inversion RF circuits to implement entire sensor systems.

Intelligent sensor nodes function much like individual ants that, when formed into a network, cooperatively accomplish complex tasks and provide capabilities greater than the sum of the individual parts.

While this new class of networks has the potential to enable a wide range of applications, it also poses serious challenges.

- Distributed sensor networks must provide exception free and unattended operation or be self-configuring
- Distributed sensor networks must operate and respond to very dynamic environments.
- Distributed sensor networks must be data-centric and should be data-concentrated
- Distributed sensor networks should be application-specific
- Distributed sensor networks should be power-aware and energy-efficient.
- Distributed sensor networks should be latency-aware because of the requirement of the real-time sensing.

All of these distinguished characteristics can potentially affect aspects of the system's design. Many of these problems are more challenging than their analogues in the traditional networking space, but there are also new opportunities for solutions.

Some research works have given consideration or solutions towards certain aspect of design challenge for distributed sensor network, but little work has been done to give an analysis from a systematic point of view.

This report tries to solve this problem based on the historical research work towards the analysis of design challenges for distributed sensor networks. We explore the characteristics and requirements of distributed sensor networks and focus on our research upon several general but most important issues while implement a distributed sensor networks. These issues include: hardware, network architecture (layer and protocols), communication mechanism and routing protocols, addressing mechanism and naming protocols, application architecture, data fusion, and security. While considering each aspect of these issues, we compare the requirement and potential solution with those in traditional networks.

We hope this report will help the researchers or designers of distributed sensor network form a clear understanding of the requirements and possible solutions towards the whole aspects of distributed sensor network.

2 Distributed microsensor networks (DSNs) Challenges:

2.1 Should be power-aware and energy-efficient.

In the scenario where the sensors are operating in remote or dangerous territory, it may be impossible or inconvenient to retrieve the nodes in order to recharge batteries. Therefore, the network should be considered to have a certain lifetime during which nodes have energy and can gather, process, and transmit information. This means that all aspects of the node, from sensor module to the hardware and protocols, must be designed to be extremely energy-efficient. Decreasing the energy usage by a factor of two can double system lifetime, resulting in a large increase in the overall usefulness of the system. In addition to reducing energy dissipation, protocols should be robust to node failures in order to maximize system lifetime.

2.2 Should provide exception free and unattended operation or be self-configuring

Since it is required that the microsensor networks be easily deployable, this means that the nodes be able to communicating with each other even in the absence of an established network infrastructure.

In DSNs, the ratio of communicating nodes to users is much greater comparing with the Internet. Each computer on the Internet has at least a user who can resolve or report all manner of the errors and problem. For this reason, the Internet may function with much less robust software. But in DSNs, they will exist with the ration of thousands of nodes per user. So, it is impossible to pay special attention to each of individual node. Another reason is even if it is possible to do so, the sensors are not reachable, either because they are embedded in physical structures, or thrown into inhospitable terrain. These characteristics

2.3 Should operate and must respond to very dynamic environments.

DSNs will be deployed in a very ad hoc manner. They will suffer substantial changes as node fail due to battery exhaustion or accident, new nodes are added, nodes move or are carried. User and environmental demands also contribute to dynamics as what is being sensed moves and what is considered interesting changes.

2.4 Should be data-centric and should be data-concentrated

Unlike traditional networks, the sensor node of the DSNs may not need an identity (such as an IP or an address). The network is not node-centric but data-centric, that means application on the DSNs focus on the data generated by sensor nodes. The applications may decouple the data from the sensors, which produce the data.

Here I bring forward the notion data-concentrated which means that the data from neighboring nodes are highly correlated, making the data redundant and the end-user cares about a higher-level description of the events occurring in the environment the nodes are monitoring.

2.5 Should be application-specific

We know traditional networks are designed to accommodate a wide variety of applications. But in DSNs, they can be tailored to the sensing task at hand. In particular, this means that intermediate nodes can perform application-specific data aggregation and caching, or informed forwarding of requests for data. This is contrast to routers that facilitate node-to-node packet switching in traditional networks.

2.6 Should be latency-aware because of the requirement of the real-time sensing.

Events occurring in the environment being sensed may be time-sensitive. It is often important to bind the end-to-end latency of data dissemination. This requires that the design of the DSNs should therefore minimize overhead and extraneous data transfers.

3 Analysis

We try to explore the design aspects of DSNs, from hardware issues, network architectures and protocols, to software issues such as routing algorithms, data fusion algorithms, to application issues.

3.1 Hardware concerns:

We focus our analysis upon networked microsensor.

The microsensor is the basic component of the DSNs. The microsensors are designed for ease of deployment and to be low cost, compact, lightweight, and disposable. Local and collaborative signal processing across the DSNs enhances sensor nodes primitive sensing function (seismic, acoustic, magnetic).

The microsensors will support a flexible hardware and software architecture allowing them to taken on various roles in the network. The exact function of the microsensor may not be determined until deployment and may change over the course of its mission.

The features of the distributed microsensor:

- Should be small physical size and low power consumption.

As known from the history of the technological evolution, size and power constrain the

processing, storage, and interconnect capability of the basic device. So, reducing the size and power required for a given capability are driving factors in the hardware design.

- Should support concurrency-intensive operation:

The primary mode of the operation for these microsensors is to flow information from place to place with a modest amount of processing on-the-fly, rather than to accept a command, stop, think, and responds. There is little internal storage capacity, so buffering large amount of data between the inbound and the outbound flows is unattractive.

- Limited physical parallelism and controller hierarchy.

The number of independent controllers, the capabilities of the controllers, and the sophistication of the processor-memory-switch level interconnect are much lower than in the conventional systems. The space and power constraints and limited physical configurability on-chip are likely to retain the need to support concurrency-intensive management of flows through the embedded microprocessor.

- Diversity in design and usage

Networked distributed microsensor will tend to be application specific, rather than general purpose, and carry only the available hardware support actually needed for the application. A generic development environment is needed which allows specified applications to be constructed from a spectrum of devices without heavyweight interfaces. It should be natural to migrate components across the hardware/software boundary as technology evolves.

- Robust operation

The networked microsensor will be numerous, and largely unattended, and expected to operational a large fraction of the time. The application of the traditional redundancy techniques is constrained by space and power limitations. So, enhancing the reliability of individual sensor is essential. This reinforces the need for efficient modularity: the components should be as independent as possible and connected with narrow interface.

- Dynamically configurable to support a variety of network functions or roles

We noticed there are some existing approaches relating to our concerns:

- The wireless integrated network sensors (WINS), UCLA.

The WINS NG node architecture was developed in UCLA to enable continuous sensing, signal processing for event detection, local control of actuators, event identification, and communication at low power. The sensor, data convertors, data buffer, and signal processing all operate at micropower levels. Protocols for node operation then determine whether extra energy should be expended for further processing and whether a remote user or neighboring WINS node should be alerted. The WINS node then communicates and attribute of the identified event, possibly the address of the event in an event look up table stored in all network nodes.

- Networked sensor based on Tiny Microthreading Operating system(TinyOS).

- Hardware organization:

The prototype uses a coprocessor beside the processor within the MCU. It

integrates a set of timers and counters which can be configured to generate interrupts at regular time intervals, especially the 3 sleep modes: idle, power down, power save.

The radio is the most important component, which represents an asynchronous input/output device with hard real time constraints. Control signals configure the radio to operate in either transmit, receive or power-off mode. The temperature sensor represents a large class of digital sensors which have internal A/D converters and interface over a standard chip-to-chip protocol.

○ Operating system organization:

TinyOS is an event-driven operating system. It provides good support for efficient modularity and concurrency-intensive operation. The collection of tasks associated with an event is handled rapidly and unused CPU cycles are spent in the sleep state as opposed to activity looking for some interesting event. So, it ensures the energy efficiency.

Here are my consideration and suggestion about the hardware concerns:

- One of the requirements of design the microsensor is that there is little internal storage capacity, so buffering large amount of data between the inbound and the outbound flows is unattractive. But as we discuss before, for the whole network, we should try to make it data-concentrated, which means before data reaching the end-user, the network will do some data aggregation (data fusion) or caching in the source sensors or the intermediate sensors to generate concentrated data. It requires that the microsensor should have storage for the fused-data. We see a contradiction. So, we must find a tradeoff and find the proper sensor storage with consideration of data-concentrated character of DSNs.

- Although several approaches pay attention on the energy-efficiency during the design Work (they try to reduce energy consumption while design each aspect of the sensor architecture), few of them provide mechanisms to monitor the energy performance while in the future deployment. We know, after the sensor being deployed and if it is unattended, its energy (mostly provided by the embedded battery) may reduce continuously during the lifetime, some important mechanism of the whole network depends heavily on each sensor's energy situation and will do self-configuration according to the sensor's energy situation. And since it is very hard to measure the energy situation of the sensor by external means, it is important to provide such energy monitor mechanism inside of sensor. As to how to report the energy situation of it, I suggest each microsensor may add his energy information to the data he will generate and transmit. We can separate the data from a sensor into 2 parts, the data it get from the monitoring object and the data from the sensor itself.

- Still consideration about the energy. In most cases, the energy provider in the sensor is the embedded battery. I think it is based on some basic requirements of sensors such as low-cost, diversity of the deployment. We can invent additional mechanism (device) to provide additional energy by converting some other type of energy such as heat and light. These devices can be easily merged into the architecture of the sensor without much system modification but relatively independent. If the microsensor is deployed into proper

environment, it can have alternative energy supplier.

- We assume that the microsensors employ an embedded operating system to manage and support its application for providing real-time performance. But in order to support a flexible design, the microsensors should be remotely reprogrammable to support new functionality. Since in the lifetime of the microsensors, we may want to change some functions embedded in them, but we can not do it locally due to certain causes but have to deal with remotely.

3.2 Network architectural Concerns:

In traditional network, a set of layers and protocols is called network architecture. Such as the one show in Figure:

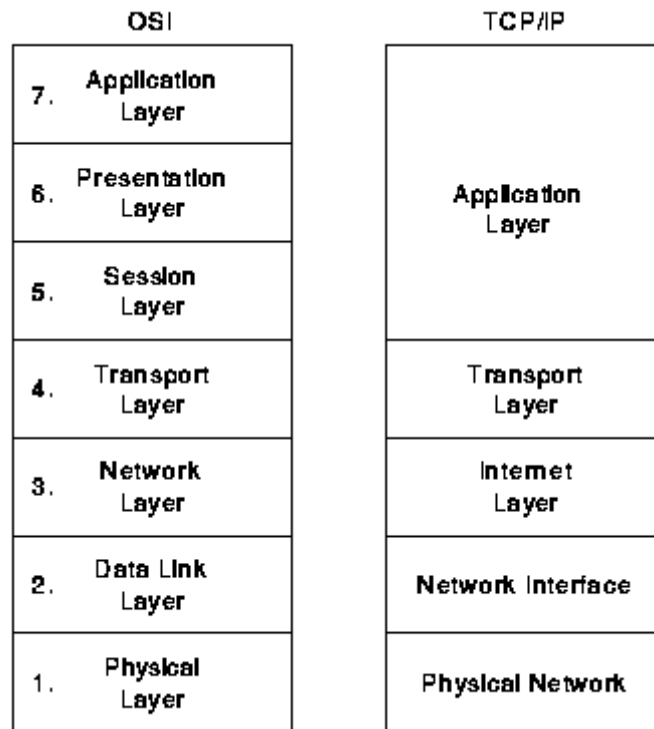


Figure1: layer model of traditional network

The DSN is basically an ad hoc wireless network where the membership of roles of sensor nodes is generally not know until the deployment of the network. And once deployed, the network is self-organizing. Most researches upon DSNs don't deny that the basic architecture of DSNs still follows the traditional network architecture.

3.3 Network layered protocols Concerns:

In conventional network architecture, each layer of the system is designed separately and is independent of the application. A layered approach allows the system design to be broken into smaller pieces that can be developed independently. Protocols designed using such an approach, while reusable by many different applications, are not optimal for any given application.

Rather than using general-purpose protocol architecture, we can foresee in DSNs that the system will be more efficient if they are designed to exploit features of the applications they are supporting. Since the goal of the network design is to maximize the performance of the resident applications and the DSNs has an application-specific nature, so we may see that a cross-layer architecture that exploits features of the application can achieve greater performance than general-purpose protocols.

The protocols should be robust to node failures in order to maximize system lifetime, and it should be fault-tolerant, such that the loss of a small number of nodes does not greatly affect the overall system performance. In addition, it should be scalable such that the addition of new nodes requires low overhead to incorporate the nodes into the existing network.

A battery-constrained sensor node does not have enough energy to support a full TCP/IP stack, perform complex routing algorithms, exchange data at high rates of speed, and constantly poll the system for data and routing updates, so the networking models should take into limited energy into consideration.

Also let us focus on some practical approaches.

- In Jon Agre and Loren clare's research work, they declares that a layered architecture that joins autonomous and cooperative signal-processing elements provides the foundation for large, highly scalable sensor networks. The layered architecture is shown in the following Figure:

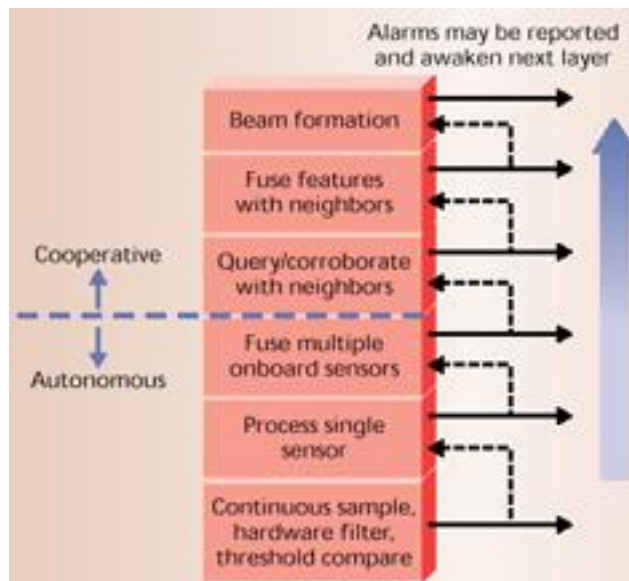


Figure 2:A layered architecture in Jon Agre and Loren clare's research work

The first 3 layers, called the autonomous stack, can be performed independently at any microsensor. At the first layer, a sensor node uses a special low-power circuit to constantly sample its sensors and determine whether a threshold has been exceeded. Once a potential event has been detected, the more powerful signal-processing engine activates on the second layer to perform sophisticated data analysis. At the third layer, data from multiple onboard sensors is fused to improve the classification.

The next three layers, called the cooperative stack, require communication between nodes. The fourth layer involves the simple exchange of information for activities such as voting or comparing arrival times. The fifth layer exchanges the intermediate or feature data. The sixth communicates raw data among a subset of nodes to perform beamforming.

- In NAI lab's research upon DSNs, they suggest a layered protocol stack similar to that show in the following Figure:

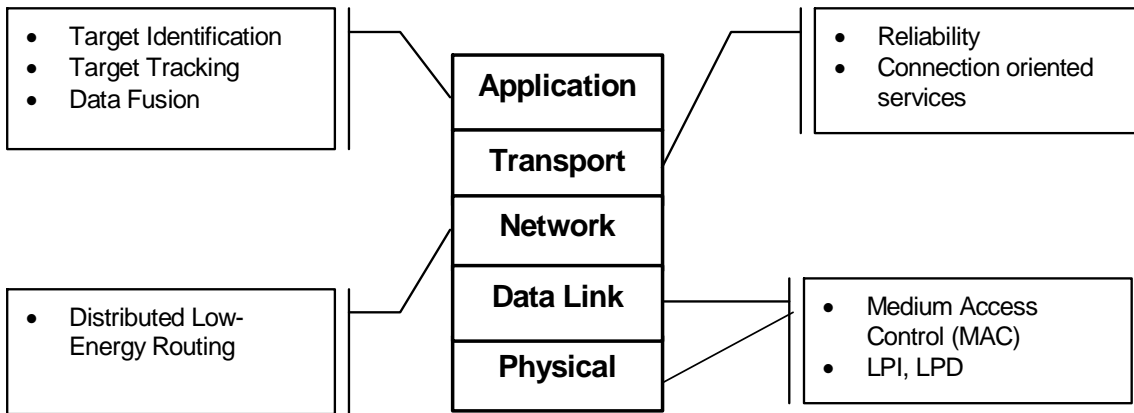


Figure: DSN Communications Layers

- Physical Layer

The physical layer defines the mechanisms for medium access control (MAC) for the wireless sensor network.

- Network Layer

The sensor network utilizes certain-type (maybe a multi-hop bursty) packet based network routing protocol to deliver data throughout the network. The finite energy of the network is the primary design constraint in developing a low-energy routing algorithm (I will discuss more about routing algorithm later) that balances energy throughout the network.

In order to construct the necessary routing information, each sensor node must determine its neighbor nodes and then make a determination of which it will route traffic to. The decision on how to do this energy efficiently is dependent on the

routing algorithm.

- Transport Layer

The transport layer protocols can be designed to provide reliability and session control for sensor node applications.

- Application (Data Fusion) Layer

It is generally more efficient to perform local processing on sensor data rather than transmit the raw data to a centralized point for processing. Sensor nodes will contain signal-processing algorithms specific to their functionality (e.g., acoustic, seismic) to perform local target identification and perform collaborative target tracking. Tracking information received from a group of sensors may be processed by an intermediate fusing node.

3.4 Communication algorithm and routing protocol Concerns:

In DSNs, the microsensors coordinate to establish a communication network. That means sensors will be able to coordinate amongst themselves on a higher-level sensing task. In particular, the coordination can contribute to more scalable behavior as number of nodes increase, improved robustness, and more efficient resource utilization for many distributed sensor tasks. So the communication paradigm used among microsensors which including the routing protocols is very important. Since the DSNs is quite different from the traditional network, the design of communication algorithms and routing protocols should take the features of the DSNs into consideration. They must be energy-efficient, and the can not be affected by the self-configurable feature of the DSNs.

Also we should know that the bandwidth of the wireless links connecting sensor nodes is often limited, it will constrain the inter-sensor communication.

Several current approaches aims to focus on this consideration.

- The SPIN(Sensor Protocols for Information via Negotiation) is a family of protocols to disseminate information among sensors in an energy-constrained wireless sensor network. In SPIN, large data messages are named using high-level data descriptors, called meta-data. Nodes use meta-data negotiation to eliminate the transmission of redundant data throughout the network. Allowing nodes to base routing decisions on application-specific information about the data enables large energy-savings compared with conventional approaches.

- Clustering allows sensors to efficiently coordinate their local interactions in order to achieve global goals. LEACH (Low-Energy Adaptive Clustering Hierarchy) is a clustering-based protocol that utilizes randomized rotation of local cluster base stations to evenly distribute the energy load among the microsensors of the network. LEACH uses localized coordination to enable scalability and robustness for dynamic networks.

- DIRECTED DIFFUSION is a completely different paradigm for microsensor communication. Using direct diffusion, data are named with attribute-value pairs, and interests for certain types of data are disseminate throughout the network. These interests

diffuse to the correct area, setting up gradients that draw events of interest back to the node that originated the request. Good routes are inherently reinforced, enabling low-energy routing of the data.

3.5 Address and naming mechanism Concerns:

In traditional networks, every layer needs a mechanism for identifying senders and receivers. Inherent to the design of most packet networks today is the assumption that each node has a unique address. One of the primary purposes of an address in a conventional network is to provide topological information that can be used to find routes. Address is sometimes also used as names in order to specify a communications endpoint. An important property of addresses in traditional network is that every node has a globally unique one. Addressing therefore has an additional benefit of assigning unique identifiers to nodes. Most addressing schemes can be classified as using either global or local address. Each has its own advantages and disadvantages.

Let consider addressing in DSNs. It is vital to consider the cost of the address (the overhead required inters of network utilization) in DSNs where every bit transmitted can reduce the lifetime of the network. This is by design: the energy cost of communication makes it desirable for nodes to minimize the size and frequency of transmissions by doing as much local processing, summarization, and aggregation of data as possible. So, the cost of an address in an energy-constrained network can be considered high if the address space is underutilized and the address itself accounts for a significant portion of the total bits transmitted. In sensor global unique address would need to be very large compared to the typical size of data attached to them. Therefore, the local addressing seems to be needed.

To maintain local addresses, a sensor network could use a protocol that dynamically assign addresses to nodes based on the addresses of other nodes in the neighborhood. But this scheme will be efficient only as long as the address-allocation overhead is small compared to the amount of useful data transmitted. So, new addressing protocol should be addressed which should provide the small, randomized identifiers that have the advantages of locally unique addresses without the costs usually associated with their maintenance.

Some researchers have been doing works towards this goal. One of the potential solution is the attribute-based naming, originated from the research work upon intentional naming system. The directed diffusion protocol has mentioned such a addressing approach. This kind of data naming is expected to be application-specific. So, such a design effectively moves naming, addressing and even routing from the network layer into the application. This is highly non-traditional compared to existing layered architectures. It departs from Internet-like service models that provide end-to-end packet forwarding. As we explained earlier, factors such as the expected scale and energy constraints indicate the need for highly efficient, application-specific processing inside the network. So, this addressing mechanism seems to work.

3.6 Application architecture and data fusion Concerns:

Application cannot be structured much the same way as traditional Internet application. It should focus on the data generated by sensors. The data-centric feature of the DSN allows for more robust application design: even if certain sensor dies, the data it generates can be cached in other sensor for later retrieval.

As we discussed earlier, depending on the sensor node hardware, it is generally more efficient to perform local processing on sensor data rather than transmit the raw data to a centralized point for processing. So in DSNs, the intermediate nodes can perform application-specific data aggregation and caching. The aspect of the model leverages the application-specificity that is possible in sensor networks. This is quite different from the role of the intermediate nodes in traditional networks. In traditional networks, the intermediate node may not do processing upon the data it transfers. Caching and aggregation can increase the efficiency, robustness and scalability of coordination. Locally cached data may be accessed by other users with lower energy consumption than if the data were to resent end-to-end. Intermediate node storage increases availability of the data, thereby improving robustness.

Another point of view is, sensor data are different from the data associated with traditional wireless networks in that it is not the actual data itself that is important; rather, the analysis of the data, which allows an end-user to determine something about the environment that is being monitored, is the important result of a distributed sensor network. Therefore, automated methods of combining or aggregating the data into a small set of meaningful information are required.

Some research work has been done towards data fusion. One method of aggregating data is called beamforming. Beamforming combines signals from multiple sensors as follows:

$$\mathbf{y}[n] = \sum \sum \mathbf{w}_i[l] \mathbf{s}_i[n-l]$$

where $S_i[n]$ is the signal from the i th sensor, $w_i[n]$ is the weighting filter for the i th signal, N is the total number of sensors whose signals are being beamformed, and L is the number of taps in the filter. The weighting filters are chosen to satisfy an optimization criteria, such as minimizing mean squared error (MSE) or maximizing signal-to-noise ratio (SNR).

Here we must consider the tradeoff between local data processing or caching and storage capability.

3.7 Security Concerns:

Distributed sensor networks will employ communications among large numbers of sensors remotely deployed in irregular patterns to form ad hoc distributed processing networks that can produce high-quality information. Sensors will have limited resources, including memory, computational capability, communications bandwidth, and power. The security properties of distributed sensor networks will be of profound importance.

So, it is very critical to design a communications security architecture that incorporates cryptographic security mechanisms that efficiently support the provision of required integrity, authentication, and confidentiality security services within distributed networks of resource-

limited sensors. Here are some design goals for the security architecture of DSNs Identify practical cryptographic mechanisms and protocols that can be selectively employed by resource-limited sensor nodes; Design communications security architecture suitable for use by distributed networks of Resource-limited sensor nodes;

Also, let us have a look at the attacker model. The attacker can take over any node within a network (but subverting all nodes is not possible). Subverting a single node means that that all nodes within communication radius of that node can be denied receiving any information (they still may be able to send information). So, we should seek to minimize the impact of a subverted node on the rest of the network: in particular, the single node should not grant the attacker the ability to subvert the entire network. Note that in a network with a single gateway (base station) the attacker needs to take over just that one node in order to render the network useless. It is crucial that some redundancy be provided. We should take the microsensor limitation into consideration while designing security concerns. Sensor network nodes are typically quite limited in capability. Only a fraction of processing and memory space can be devoted to cryptographic algorithms, thus it seems that the use of public key encryption is too expensive. Symmetry key cryptography is orders of magnitude cheaper, and may suffice for the sensor network applications; it still remains a challenge to use nontrivial cryptography with such limited resources in an environment where crypto is not the sole application of the device.

4 Conclusions and future work

In this report, we first explore the distinguished features of Distributed Sensor Network (DSNs), including energy-constraint, self-configurable, data-centric, application-specific, and latency-aware, etc. We envision that these characteristics of DSNs will bring great challenges for the design of the DSNs with comparison with traditional network.

Then we discussed in details some key design considerations for DSNs, including hardware concerns, network architectural concerns, communication and routing protocols concerns, addressing and naming concerns, application architecture, data fusion concerns, and security concern etc. From these detailed analysis, we draw a conclusion that the distinguished features of DSNs do badly affect the design considerations. Also, we see there is a rich set of research problems associated with distributed microsensors that require very different solutions than traditional networks. We try to give our own answers to some design considerations, but a lot of left problems still need further exploration.

As to the future work, there is still much work to be done in the area of application architecture for the distributed sensor network. We just get an overview of the requirements for application architecture but there is no concrete answer or solution towards this problem.

While we mentioned the use of beamforming algorithm for data fusion, there is a need of better, faster, and more accurate data aggregation and classification algorithms. In addition to combining data from similar sensor, these algorithms need to handle multi-sensor input. For

example, future sensor networks may contain different type of sensors; they all coexist in the same environment. So, it is important to aggregate such a collection of data.

Finally, more work still needs to be done upon the security issue for distributed sensor network.

Reference

- [1] Andrew S. Tanenbaum. *Computer Networks*. 3rd edition, prentice hall,1996.
- [2] Michael J.Dong, K. Geoffrey Yung, and William J. Kaiser. *Low Power Signal Processing Architecture for Network Microsensors*. In Proceedings 1997 International Symposium on Low Power Electronics and Design, pages 173-177, August 1997
- [3] G.J.Pottie and W.J. Kaiser, *Wireless integrated Network Sensors*
- [4] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, Kristofer Pister. *System Architecture Direction for Network Sensors*
- [5] Scott Shenker. *Fundamental Design Issues for the Future Internet*
- [6] Gregory J. Pottie, *Wireless Sensor Networks*
- [7] Jon Agre and Loren Clare. *An Integrated Architecture for Cooperative Sensing Networks*
- [8] D. Estrin, R. Govindan, J. Heidemann, and S.Kumar. *Next Century Challenges: Scalable Coordination in Sensor Networks*. In Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, pages 263-270, August 1999
- [9] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. *Energy-Efficient Communication Protocol for Wireless Microsensor Networks*.
- [10] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin. *Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks*.
- [11] Atayoun Sohrabi and Gregory J. Pottie. *Performance of A Novel Self-Organization Protocol for Wireless Ad-hoc Sensor Network*
- [12] *Adaptive Protocols for Information Dissemination in Wireless Sensor Networks*
- [13] Saad Biaz, Gavin Holland, Young-Bae Ko, Nitin Vaidya. *Evaluation of Protocols for Wireless Networks*

- [14] Ivan Stojmenovic and Xu Lin, *Power-aware Localized Routing In Wireless Networks*
- [15] Jae-Hwan Chang and Leandros Tassiulas. *Energy Conserving Routing In Wireless Ad-hoc Networks*
- [16] Eremy Elson and D. Estrin. *An Address-Free Architecture for Dynamic Sensor Networks.*
- [17] William Adjie-Winoto, Elliot Schwartz, Hari Balakrishnan, and Jeremy Lilley. *The Design and Implementation of An Intentional Naming System*
- [18] Amin Vahdat, Michael Dahlin. *Active Names: Flexible Location and Transport of Wide-Area Resources*
- [19] Anatha Chandrakasan Rajeevan Amirtharajah, S. Cho, J. Goodman, G. Konduri, J. Kulik, W. Rabiner, A. Wang. *Design Consideration for Distributed Microsensor System*
- [20] D.W. Carman, P.S. Kruus, B. J. Matt. *Constraints And Approaches For Distributed Sensor Network Security*
- [21] Wendi Beth Heinzelman. *Application-Specific Protocol Architectures for Wireless Networks*