International Research Journal of Computer Science (IRJCS)
Issue 01, Volume 3 (January 2016)

ISSN: 2393-9842
www.irjcs.com

Wireless Sensor Networks Architecture

Ghasem Farjamnia [1] and Yusif Gasimov [2]

[1] *Institute of Applied Mathematics Baku State University, Baku, Azerbaijan*

[2] *Institute of Applied Mathematics Baku State University, Baku, Azerbaijan*

Abstract- Wireless sensor networks (WSNs) have attracted a wide range of disciplines where close interactions with the physical world are essential. The distributed sensing capabilities and the ease of deployment provided by a wireless communication paradigm make WSNs an important component of our daily lives. By providing distributed, real-time information from the physical world, WSNs extend the reach of current cyber infrastructures to the physical world. In this paper, we present a detailed explanation of wireless sensor networks architecture and protocol. Our aim is to provide a contemporary look at the current state of the art in WSNs and discuss the still-open research issues in this field.

Keywords- Wireless Sensor Network, Protocol, Architecture, Layers

I. INTRODUCTION

With the recent advances in micro electro-mechanical systems (MEMS) technology, wireless communications, and digital electronics, the design and development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate untethered in short distances have become feasible. The ever-increasing capabilities of these tiny sensor nodes, which include sensing, data processing, and communicating, enable the realization of wireless sensor networks (WSNs) based on the collaborative effort of a large number of sensor nodes. WSNs have a wide range of applications. In accordance with our vision [1], WSNs are slowly becoming an integral part of our lives. Recently, considerable amounts of research efforts have enabled the actual implementation and deployment of sensor networks tailored to the unique requirements of certain sensing and monitoring applications.

In order to realize the existing and potential applications for WSNs, sophisticated and extremely efficient communication protocols are required. WSNs are composed of a large number of sensor nodes, which are densely deployed either inside a physical phenomenon or very close to it. In order to enable reliable and efficient observation and to initiate the right actions, physical features of the phenomenon should be reliably detected /estimated from the collective information provided by the sensor nodes [1, 2]. Moreover, instead of sending the raw data to the nodes responsible for the fusion, sensor nodes use their processing capabilities to locally carry out simple computations and transmit only the required and partially processed data. Hence, these properties of WSNs present unique challenges for the development of communication protocols [2].

II. WSN ARCHITECTURE AND PROTOCOL

The sensor nodes are usually scattered in a *sensor field* as shown in Figure 1. Each of these scattered sensor nodes has the capability to collect data and route data back to the *sink/gateway* and the end-users. Data are routed back to the end-user by a multi-hop infrastructure less architecture through the sink as shown in Figure 1. The sink may communicate with the *task manager/end-user* via the Internet or satellite or any type of wireless network (like Wi-Fi, mesh networks, cellular systems, WiMAX, etc.), or without any of these networks where the sink can be directly connected to the end-users. Note that there may be multiple sinks/gateways and multiple end-users in the architecture shown in Figure 1. In WSNs, the sensor nodes have the dual functionality of being both data originators and data routers [3].
Hence, communication is performed for two reasons:

- ***Source function:*** *Source nodes with event information perform communication functionalities in order to transmit their packets to the sink.*
- ***Router function:*** *Sensor nodes also participate in forwarding the packets received from other nodes to the next destination in the multi-hop path to the sink.*

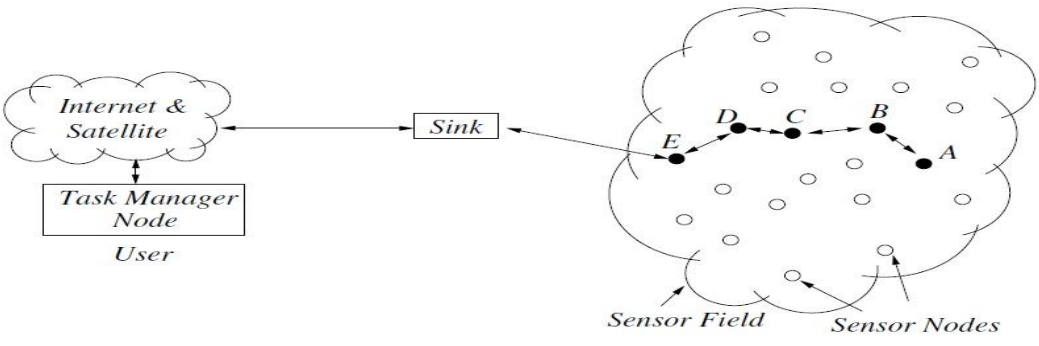

Figure 1- Sensor nodes scattered in a sensor field.

ISSN: 2393-9842
www.irjcs.com

The protocol stack used by the sink and all sensor nodes is given in Figure 2. This protocol stack combines power and routing awareness, integrates data with networking protocols, communicates power efficiently through the wireless medium, and promotes cooperative efforts of sensor nodes. The protocol stack consists of the *physical layer*, *data link layer*, *network layer*, *transport layer*, *application layer*, as well as *synchronization plane*, *localization plane*, *topology management plane*, *power management plane*, *mobility management plane*, and *task management plane*. The physical layer addresses the needs of simple but robust modulation, transmission, and receiving techniques. Since the environment is noisy and sensor nodes can be mobile, the link layer is responsible for ensuring reliable communication through error control techniques and manage channel access through the MAC to minimize collision with neighbors' broadcasts. Depending on the sensing tasks, different types of application software can be built and used on the application layer. The network layer takes care of routing the data supplied by the transport layer. The transport layer helps to maintain the flow of data if the sensor network application requires it. In addition, the power, mobility, and task management planes monitor the power, movement, and task distribution among the sensor nodes. These planes help the sensor nodes coordinate the sensing task and lower the overall power consumption [4].

2.1. PHYSICAL LAYER

The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, modulation, and data encryption. Frequency generation and signal detection have more to do with the underlying hardware and transceiver design and hence are beyond the scope of our book. More specifically, we focus on signal propagation effects, power efficiency, and modulation schemes for sensor networks.

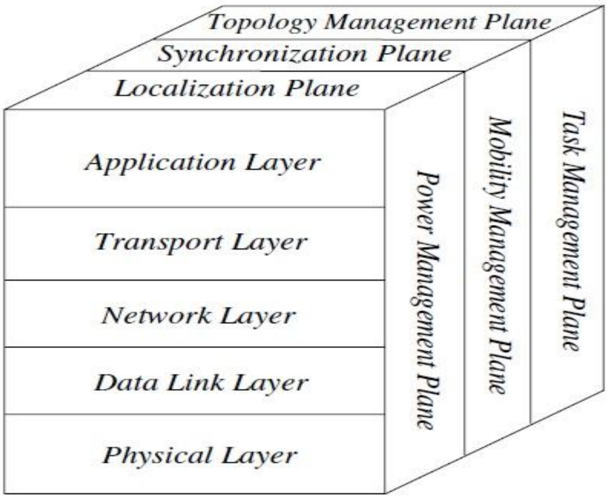

Figure 2- The sensor network protocol stack

2.2. DATA LINK LAYER

The data link layer is responsible for the multiplexing of data streams, data frame detection, and medium access and error control. It ensures reliable point-to-point and point-to-multipoint connections in a communication network. More specifically, we discuss the medium access and error control strategies for sensor networks.

2.3. NETWORK LAYER

Sensor nodes are scattered densely in a field either close to or inside the phenomenon as shown in Figure 1. The information collected relating to the phenomenon should be transmitted to the sink, which may be located far from the sensor field. However, the limited communication range of the sensor nodes prevents direct communication between each sensor node and the sink node. This requires efficient multi-hop wireless routing protocols between the sensor nodes and the sink node using intermediate sensor nodes as relays [5]. The existing routing techniques, which have been developed for wireless ad hoc networks, do not usually fit the requirements of the sensor networks. The networking layer of sensor networks is usually designed according to the following principles:

- *Power efficiency is always an important consideration.*
- *Sensor networks are mostly data-centric.*
- *In addition to routing, relay nodes can aggregate the data from multiple neighbors through local processing.*
- *Due to the large number of nodes in a WSN, unique IDs for each node may not be provided and the nodes may need to be addressed based on their data or location.*

ISSN: 2393-9842
www.irjcs.com

An important issue for routing in WSNs is that routing may be based on data-centric queries.
Based on the information requested by the user, the routing protocol should address different nodes that would provide the requested information. More specifically, the users are more interested in querying an attribute of the phenomenon rather than querying an individual node.

2.4. TRANSPORT LAYER

The transport layer is especially needed when the network is planned to be accessed through the Internet or other external networks. TCP, with its current transmission window mechanisms, does not address the unique challenges posed by the WSN environment. Unlike protocols such as TCP, the end-to-end communication schemes in sensor networks are not based on global addressing. These schemes must consider that addressing based on data or location is used to indicate the destinations of the data packets. Factors such as power consumption and scalability, and characteristics like data-centric routing, mean sensor networks need different handling in the transport layer. Thus, these requirements stress the need for new types of transport layer protocols [6].
The development of transport layer protocols is a challenging task because the sensor nodes are influenced by hardware constraints such as limited power and memory. As a result, each sensor node cannot store large amounts of data like a server in the Internet, and acknowledgments are too costly for sensor networks. Therefore, new schemes that split the end-to-end communication probably at the sinks may be needed where UDP-type protocols are used in the sensor network.
For communication inside a WSN, transport layer protocols are required for two main functionalities: reliability and congestion control. Limited resources and high energy costs prevent end-to-end reliability mechanisms from being employed in WSNs. Instead, localized reliability mechanisms are necessary. Moreover, congestion that may occur because of the high traffic during events should be mitigated by the transport layer protocols. Since sensor nodes are limited in terms of processing, storage, and energy consumption, transport layer protocols aim to exploit the collaborative capabilities of these sensor nodes and shift the intelligence to the sink rather than the sensor nodes

2.5. APPLICATION LAYER

The application layer includes the main application as well as several management functionalities. In addition to the application code that is specific for each application, query processing and network management functionalities also reside at this layer.

The layered architecture stack has been initially adopted in the development of WSNs due to its success with the Internet. However, the large-scale implementations of WSN applications reveal that the wireless channel has significant impact on the higher layer protocols. Moreover, resource constraints and the application-specific nature of the WSN paradigm leads to *cross-layer* solutions that tightly integrate the layered protocol stack. By removing the boundaries between layers as well as the associated interfaces, increased efficiency in code space and operating overhead can be achieved.

In addition to the communication functionalities in the layered stack, WSNs have also been equipped with several functionalities that aid the operation of the proposed solutions. In a WSN, each sensor device is equipped with its own local clock for internal operations. Each event that is related to operation of the sensor device including sensing, processing, and communication is associated with timing information controlled through the local clock. Since users are interested in the collaborative information from multiple sensors, timing information associated with the data at each sensor device needs to be consistent. Moreover, the WSN should be able to correctly order the events sensed by distributed sensors to accurately model the physical environment. These timing requirements have led to the development of *time synchronization* protocols in WSNs [6, 7].

The close interaction with physical phenomena requires location information to be associated in addition to time. WSNs are closely associated with physical phenomena in their surroundings. The gathered information needs to be associated with the location of the sensor nodes to provide an accurate view of the observed sensor field. Moreover, WSNs may be used for tracking certain objects for monitoring applications, which also requires location information to be incorporated into the tracking algorithms. Further, location-based services and communication protocols require position information. Hence, *localization* protocols have been incorporated into the communication stack.

III. CONCLUSIONS

Finally, several *topology management* solutions are required to maintain the connectivity and coverage of the WSN. The topology management algorithms provide efficient methods for network deployment that result in longer lifetime and efficient information coverage. Moreover, topology control protocols help determine the transmit power levels as well as the activity durations of sensor nodes to minimize energy consumption while still ensuring network connectivity. Finally, clustering protocols are used to organize the network into clusters to improve scalability and improve network lifetime.

The integration of each of the components for efficient operation depends on the applications running on the WSN. This application-dependent nature of the WSNs defines several unique properties compared to traditional networking solutions.

Although the initial research and deployment of WSNs have focused on data transfer in wireless settings, several novel application areas of WSNs have also emerged. These include *wireless sensor and actor networks*, which consist of actuators in addition to sensors that convert sensed information into actions to act on the environment, and *wireless multimedia sensor networks*, which support multimedia traffic in terms of visual and audio information in addition to scalar data. Furthermore, recently the WSN phenomenon has been adopted in constrained environments such as underwater and underground settings to create *wireless underwater sensor networks* and *wireless underground sensor networks*. These new fields of study pose additional challenges that have not been considered by the vast number of solutions developed for *traditional* WSNs.

The flexibility, fault tolerance, high sensing fidelity, low cost, and rapid deployment characteristics of sensor networks create many new and exciting application areas for remote sensing. In the future, this wide range of application areas will make sensor networks an integral part of our lives. However, realization of sensor networks needs to satisfy the constraints introduced by factors such as fault tolerance, scalability, cost, hardware, topology change, environment, and power consumption. Since these constraints are highly stringent and specific for sensor networks, new wireless ad hoc networking techniques are required. Many researchers are currently engaged in developing the technologies needed for different layers of the sensor network protocol stack. Commercial viability of WSNs has also been shown in several fields. Along with the current developments, we encourage more insight into the problems and more development of solutions to the open research issues as described in this paper.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, March 2002.
[2] Q. Cao, T. Abdelzaher, J. Stankovic, and T. He. The LiteOS operating system: towards Unix-like abstractions for wireless sensor networks. In *Proceedings of IPSN'08*, pp. 233–244, St. Louis, Missouri, USA, April 2008.
[3] A. Dunkels, B. Grönvall, and T. Voigt. Contiki – a lightweight and flexible operating system for tiny networked sensors. In *Proceedings of IEEE EmNets'04*, Tampa, Florida, USA, November 2004..
[4] P. Levis, N. Lee, M. Welsh, and D. Culler. Tossim: accurate and scalable simulation of entire TinyOS applications. In *Proceedings of ACM SenSys'03*, pp. 126–137, Los Angeles, CA, USA, November 2003.
[5] C. B. Margi, V. Petkov, K. Obraczka, and R. Manduchi. Characterizing energy consumption in a visual sensor network testbed. In *Proceedings of the IEEE/Create-Net International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom'06)*, Barcelona, Spain, March 2006.
[6] G. Mulligan and the 6LoWPAN Working Group. The 6LoWPAN architecture. In *Proceedings of EmNets'07*, pp. 78–82, Cork, Ireland, June 2007.
[7] L. F. W. van Hoesel, S. O. Dulman, P. J. M. Havinga, and H. J. Kip. Design of a low-power testbed for wireless sensor networks and verification. Technical report, University of Twente, 2003.