

Instilling QoS in Wireless Sensor Networks

¹Kamil Samara, ²Hossein Hosseini

¹*School Department of Computer Science, Illinois Wesleyan University, United States*

²*Department of Electrical Engineering and Computer Science, University of Wisconsin - Milwaukee, United States;*

ksamara@iwu.edu; hosseini@uwm.edu

ABSTRACT

Wireless Sensor Networks (WSNs) have been a desired choice for monitoring and automatic control of remote and unreachable objects and environments due to their low cost. However, such deployment requires quality-of service (QoS) techniques to assure reliable performance. Furthermore, provision of QoS in WSNs is a challenging task due to hardware limitations. A cross-layer approach is a promising option where information from different layers can be used to make QoS decisions. In this paper, we present a routing protocol where information from the application layer is used to make differentiated routing decisions based on data packets classifications. In our case, data packets are classified into: normal, urgent, and critical. Based on this classification each data packet class is treated differently by storing each data packet class in a designated buffer. Different buffers will have different routing priority decided by the protocol designer.

Keywords: wireless sensor networks; cross-layer optimization; QoS; routing.

1 Introduction

Recent advances in technology especially in electronics and communications allowed the emerge of WSNs. A WSN is a collection of sensor nodes.

Sensor nodes have sensing capabilities which make them a suitable solution for sensing and collection data from different environments [1].

Sensor nodes are low-cost nodes with limited computing power, scarce memory, low bandwidth, and most importantly limited energy source. Sensor nodes are usually operated by batteries, which in most cases are not rechargeable or easy to replace [2].

WSNs have a wide range of applications in different fields, but the most common use for WSNs is monitoring. The WSN is usually deployed over a region or structure to monitor one or more phenomenon then reporting the reading collected, by its sensors, to the base station which can convey the aggregated data to a human operator or a central controller for further processing.

A WSN is a collection of hundreds or thousands of wireless sensor nodes that are often deployed in remote areas as shown in Fig. 1, whose job is to collect data wirelessly and deliver it to a base station. Each node contains a sensing component, processor, communication, and storage components [3].

Some common examples of WSNs applications are military, where sensor nodes are used to detect enemy movement, traffic control, where sensor nodes collect data about car jams, or in healthcare sector for monitoring patient's conditions, and in many more applications [1].

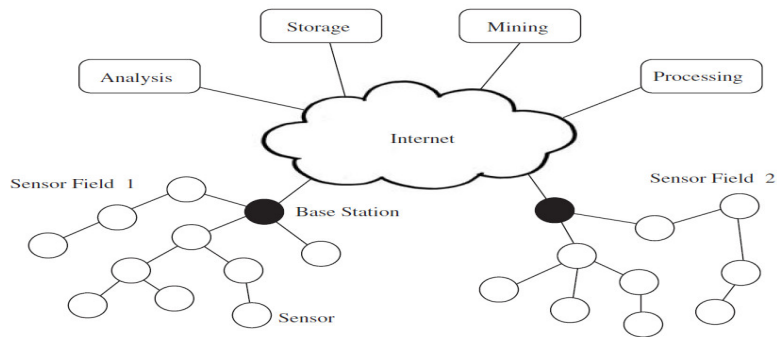


Figure 1 Wireless Sensor Network

Besides resource limitations in WSN nodes, WSNs are usually deployed in unattended and harsh environments implementing crucial applications. These factors emphasize the importance of QoS in WSNs [4].

The special nature of WSNs, mainly the limited energy source in addition to low computation and memory capabilities, makes maintaining QoS through Integrated Services approach unapplicable in WSNs. WSN nodes do not have sufficient resources to establish end-to-end flow connections and manage the information needed for these connections [3].

This work is an extension of our earlier work presented in [5] where all traffic was treated the same without any kind of classification.

In this paper, we are proposing a protocol that provides QoS features through implementing differentiated services, where packets are classified as critical, urgent, and normal. Based on this classification, different packets are assigned different priorities and resources. This results in a more reliable delivery and less delay for urgent and critical packets.

The rest of the paper is organized as follows: In section II we will give a general background about QoS at the network layer. Section III will briefly touch on some of the approaches for achieving QoS in WSNs. Our proposed protocol is described in section IV. Section V is dedicated to simulation results and finally conclusions are presented in section VI.

2 QoS Background

QoS is the overall performance experienced by users when using a networking system. To be able to measure the quality of service, several characteristics are usually measured, such as error rates, bit rate, throughput, delay, availability, and more [6].

Approaches to QoS can be categorized into two main architectures, Integrated Services (IntServ), and Differentiated Services (DiffServ). Differentiated Services is a coarse-grained QoS system. DiffServ provides QoS by classifying network traffic and providing different service according to the packet traffic class [6].

Modern networks carry different classes of data, including voice, video, and text. Each class has its own QoS needs.

DiffServ classifies packets and then routers on the path from the sender to the receiver to implement a per-hop behavior that manages each traffic class differently preferring higher-priority packets [6].

Considering the nature of WSNs, the IntServ approach is not applicable to WSNs. WSN nodes do not have sufficient resources to establish end-to-end connections and manage the information needed for these connections.

Although the QoS requirements differ in WSNs according to the network application, WSN nodes work collectively to achieve the application goals that make the DiffServ a better option to use with WSNs [4].

The most important points to be considered when designing a QoS system for WSNs are as follows:

- QoS must be integrated into all the network layers
- QoS parameters must be decided based on the WSN application
- Resources constraints must be considered

3 QoS Approaches

Routing protocols that ensure shorter paths can increase QoS in the network, because as shown in [7] each node increase on the path between the source and the destination increases the average packet loss ratio by approximately 5-10 %.

Network layer can increase reliability, which is an important QoS aspect, by enforcing multipath routing. WSNs usually have high node density, so the possibility of having more than one path between the source and the destination is high. According to [8 - 10] the delivery ratio on a 14-hop path can be increased from 50% to 75% if there is a second disjoint path.

RAP [11], and SPEED [12] all use geographic forwarding (GF), in which nodes forward packets to their one-hop neighbor that is closer to the sink. This will ensure faster delivery and shorter paths. Multipath Multi-SPEED (MMSPEED) [13] also uses GF but adds the feature of multipath.

JITS (Just-in-Time Scheduling) [14] is a network layer protocol for soft real-time packet delivery. JITS orders packets in a forwarding queue based on their transmission time. Transmission time is calculated by multiplying the average one-hop delay by the number of hops. When a packet's transmission time is reached, it is dequeued from the queue head.

4 Proposed Protocol

This section will be dedicated to describing the design of our proposed protocol.

Routing decisions are made by collecting data about all the available paths from the sensing node to the sink and then use these collected data to decide the best path according to our criteria.

The collected data are total energy on the path, number of nodes on the path (Hop Count), and the lowest energy level of a node on the path to be able to identify critical nodes. Critical nodes are nodes which have been used more than others due to their location, which results in energy drainage.

In our protocol, we classify packets as critical, urgent, and normal as shown Fig. 2. Urgent packets are packets that need to be delivered as fast as possible like real time video packets so that no delay in video streaming is caused. Critical packets are packets that hold sensitive data where reliable delivery is very

crucial to the user like enemy movement and data may not be reproducible once is lost. Normal packets are just regular traffic.

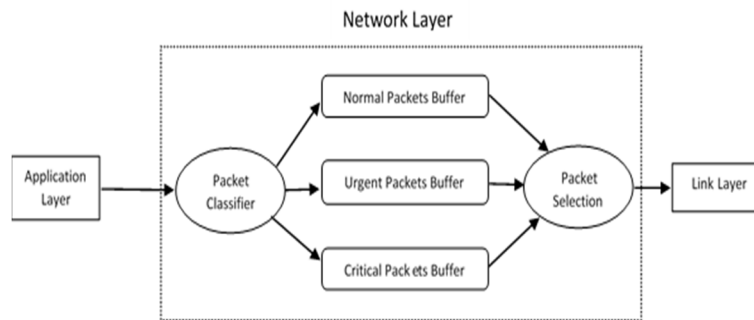


Figure 2 Packets Classification

After the routing paths are established, packets are forwarded per their class. In our protocol, urgent packets are given the highest priority so they are forwarded first, then critical packets, and lastly normal packets. This will result in less delivery delay and highest throughput for urgent and critical packets.

To ensure reliable delivery of critical packets, fault-tolerance is used where sink node will enforce two paths instead of one to deliver critical packets.

The steps of our proposed routing algorithm are shown in Fig. 4 and are described below:

4.1 Interests Propagation

An interest (also called query) is a packet generated by the application layer based on a data request submitted by the network operator. An interest packet will hold the requested data and data class. Data class is decided by the operator and it will be either normal, urgent, or critical.

Interests are flooded through the sensor network. For each active task, the sink will broadcast an interest message shown in Fig. 3 to all its neighbors. Each node that receives an interest message will also broadcast it to all its neighbors.

Query ID	Source Address	Destination Address	Time to Live	Requested Data	Data Class
----------	----------------	---------------------	--------------	----------------	------------

Figure 3 Interest Packet

Every node maintains an interest cache. Each item in the cache corresponds to a distinct interest. Two interests are distinguished by the ID field.

Interest entries in the cache do not contain information about the sink, but just about the immediately previous hop. Also identical interests are aggregated into a single entry.

When a node receives an interest, it checks to see if the interest exists in the cache. If no matching entry exists the node creates an interest entry. This entry has a single gradient (a gradient specifies a direction in which to send events) pointing toward the neighbor from which the interest was received. If an interest entry does exist, but no gradient for the sender of the interest, the node adds a gradient with the specified value.

Finally, if both an entry and a gradient do exist, the node simply updates the attribute fields if they are different. When an interest's entry has expired, the interest's entry is removed from the interests' cache. Not all received interests are resent. A node may suppress a received interest if it recently resent a matching interest.

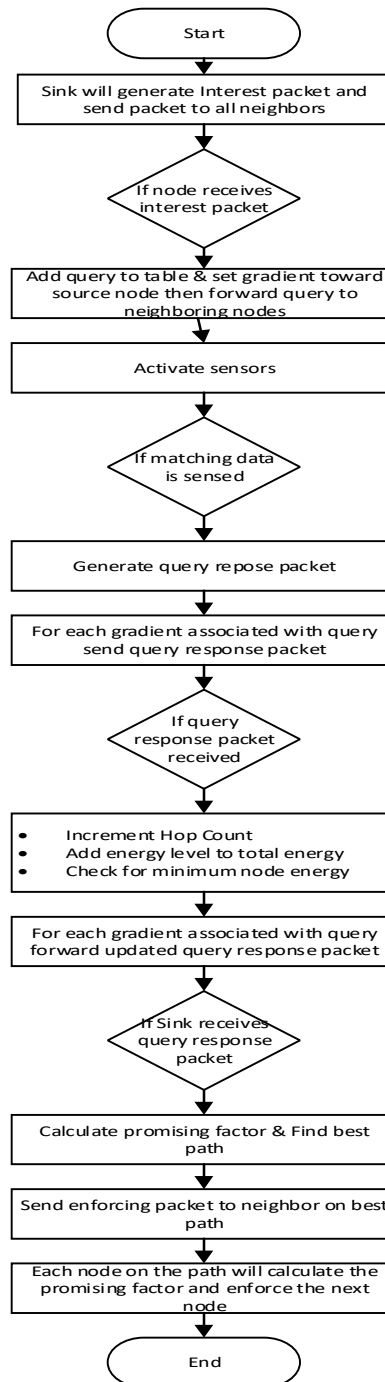


Figure 4 Flow chart of algorithm steps

4.2 Query Response

This stage is which allows us to collect the needed data about each available path and then use these data in enforcing the best path according to our criteria.

As shown in Fig. 5, we are collecting total energy on the path, the number of nodes on the path (Hop Count), and the lowest energy level of a node on the path to be able to identify critical nodes. The set of collected data could be different from one application to another according to what is important to the application in use.

For example, an application that is concerned about fast delivery could collect data about nodes' buffer size to avoid congested nodes.

When a node has at least one active interest, the node will switch on its sensors and start sensing for the requested data.

If the sensing node senses data that matches the requested data by the interest, it will generate a Query Response Packet and send a copy of it to all the gradients associated with the interest.

The base station will receive the Query Response Packet through multiple paths. Then the base station can choose the shortest path and reinforce the source sensors to use the chosen path by using enforcing packets.

Forwarding nodes, on the other hand, could receive the same Query Response Packet flooded by the sensing node from multiple neighbors, but it will only forward one of them.

Query ID	Source Address	Destination Address	Hop Count	Total Energy	Lowest Energy
----------	----------------	---------------------	-----------	--------------	---------------

Figure 5 Query Response Packet

4.3 Reinforcing Paths

When the base station starts receiving Query Response Packets in the reply of an interest that was propagated earlier, it will receive the packets through multiple paths. This is due to the source node sending the Query Response Packet to all the nodes from which it received the interest propagation packet.

Each Query Response Packet received by the base station will hold hop count, total energy, and lowest energy fields about the path it took. Based on that information the base station will choose the best path.

Fig. 6 provides an illustration of the Interest propagation, gradients setup, and data delivery stages.

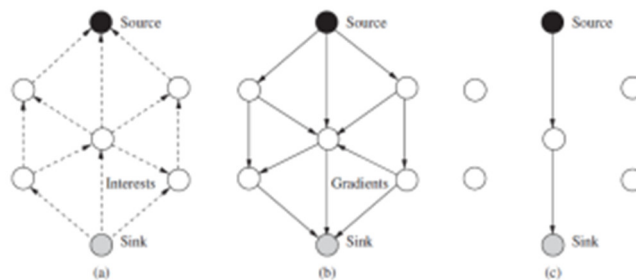


Figure 6 (a)Interest propagation (b)gradients setup (c)data delivery

Choosing the best path is done by calculating the promising factor (PF) for each path from the source (sensing node) to the destination (sink node) using the following formula:

$$PF_{s \rightarrow d} = \frac{\alpha(TE) * \beta(LE)}{\gamma(HC)} \quad (1)$$

Where:

TE: Path Total Energy Ratio

LE: Path Lowest Energy Level Ratio

HC: Path Hop Count Ratio

α, β, γ : are the weights given to each factor in the equation (default value for all weight is 1)

The term TE in equation (1) will basically give preference to paths with higher energy, which is an indicator of the health level of a path. The LE will help us avoid critical nodes, which has been used more than others and their energy level dropped significantly. The term HC will give preference to shorter paths over longer ones which will result in faster delivery.

The Path Total Energy Ratio (TE) is calculated using the following formula:

$$TE \text{ for path } s \rightarrow d = \frac{\text{Path } s \rightarrow d \text{ Total Energy}}{\sum_{\text{All Paths } s \rightarrow d} \text{Total Energy}} * 100 \quad (2)$$

Where:

$$\text{Path } s \rightarrow d \text{ Total Energy} = \sum_{\text{All node on Path } s \rightarrow d} \text{Node Energy Level} \quad (3)$$

The Path Lowest Energy Level Ratio (LE) is calculated the following formula:

$$LE \text{ for Path } s \rightarrow d = \frac{\text{Path } s \rightarrow d \text{ Lowest Energy Level}}{\sum_{\text{All Paths } s \rightarrow d} \text{Lowest Energy Level}} * 100 \quad (4)$$

Where:

$$\text{Path } s \rightarrow d \text{ Lowest Energy Level} = \min_{\text{All nodes on Path } s \rightarrow d} (\text{Node Energy Level}) \quad (5)$$

The Path Hop Count Ratio (HC) is calculated the following formula:

$$HC \text{ for Path } s \rightarrow d = \frac{\text{Path } s \rightarrow d \text{ Hop Count}}{\sum_{\text{All Paths } s \rightarrow d} \text{Hop Count}} * 100 \quad (6)$$

Where Hop Count is the number of nodes on path $s \rightarrow d$

After choosing the best path, the base station will send an enforcing packet to the neighboring node that forwarded the Query Response Packet which resulted in the highest promising factor. In turn each forwarding node on the path of the enforcing packet will forward the enforcing packet to the node it received the Query Response Packet from. The forwarding node will use the interests table to know which node to forward the enforcing packet to.

In case of critical data, base station will enforce two paths instead of one by sending send an enforcing packet to two neighboring nodes with the two highest promising factors.

4.4 Data Propagation

After the reinforcing phase is done, the source nodes know which neighboring node to use to forward the data packets. Every time it senses a matching data to the interest requested data it will generate a data packet and forward the data packet towards the base station using the enforced node.

In addition, priority queueing is used to send packets, where every node along the path will store the received packet in its designated queue according to its priority. Then, packets in the urgent buffer are forwarded first, if there are no packets left in the urgent buffer then packets in critical buffer are forwarded and lastly packets in the normal buffer are forwarded. The forwarding process will continue until the data packet reaches the base station.

This process will continue until the “time to live” field associated with the interest becomes zero. Then this interest will be removed from the table of interests in the source node, and no more data packets will be generated in response to this interest.

5 Simulation and Results

To evaluate the performance of our protocol we implemented it using Castalia simulator. Castalia simulator is a framework that can be used on top of OMNET++ to simulate WSNs, Body Area Networks (BAN) and generally networks of low-power embedded devices. Castalia is an open source simulator which allows researchers to develop and implement their own protocols [15]. Simulation parameters shown in table I.

To compare performance each simulation experiment was conducted twice. Once with differentiated services activated and second without differentiated services being activated; we call the second option single service because all packets’ classes are provided with the same service (stored in the same buffer).

Table 1. Simulation Parameters

Parameter Name	Value
Simulation Time	300 sec
X axis	40 – 180 meters
Y axis	40 – 180 meters
Number of Nodes	25- 256
Sink node	Node 0
Radio Type	CC2420
MAC Protocol	TMAC
% of Normal Packets	60%
% of Urgent Packets	30%
% of Critical Packets	10%

Two simulation experiments were conducted. The first simulation experiment focus was to compare the total number of packets delivered to the sink node. The second simulation experiment focus was to compare the average delay of packets.

5.1 Experiment One: Packets Delivery

The simulation was performed with different number of nodes ranging from 25 to 225 to prove that the same outcome will occur regardless of networks size.

Figures 7, 8, and 9 show the simulation results comparing the total number of delivered packets to the sink in both single service (Single Serv) in orange color and differentiated services (Diffserv) in blue color.

Fig. 7, compares the number of normal packets delivered to the sink in both DiffServ and Single Serv cases. While figures 8 and 9 compare the number urgent and critical packets delivered to the sink node respectively.

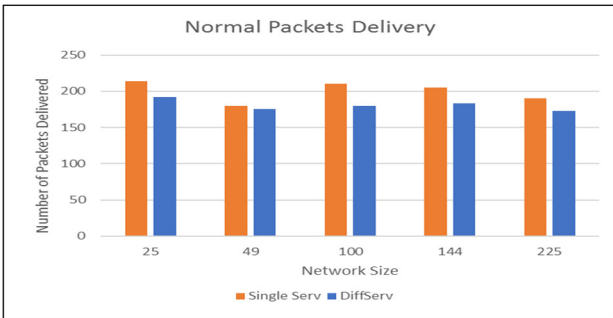


Figure 7 Normal Packets Delivery

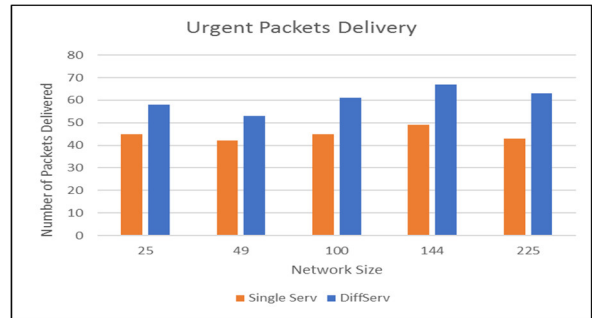


Figure 8 Urgent Packets Delivery

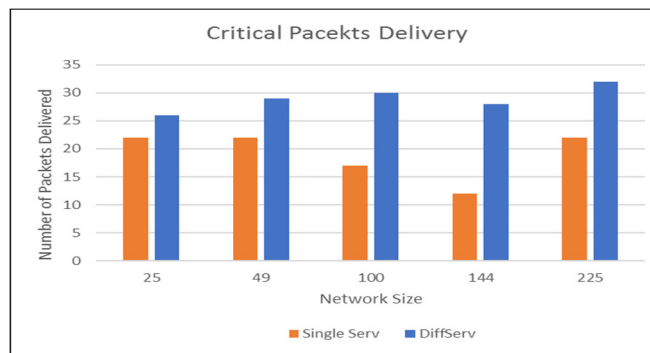


Figure 9 Critical Packets Delivery

As shown in Fig. 7, the number of normal packets delivered to the sink node using Diffserv decreased because normal packets are being stored in their own queue which has lower priority than urgent and critical queues. On the other hand, figures 8 and 9 show the increase in the total packets delivered for urgent and critical packets when using Diffserv due to buffering these types of packets in separate queues from normal packets with higher priority.

After analyzing figures 7, 8, and 9 we can conclude that using differentiated services improved the packets' delivery rate for urgent and critical packets over the expense of normal packets.

5.2 Experiment Two: Average Delivery Delay

The simulation was performed with different number of nodes ranging from 25 to 225 to prove that the same outcome will occur regardless of networks size.

Figures 10, 11, and 12 show the simulation results comparing the average delivery delay in both single service (Single Serv) in orange color and differentiated services (Diffserv) in blue color.

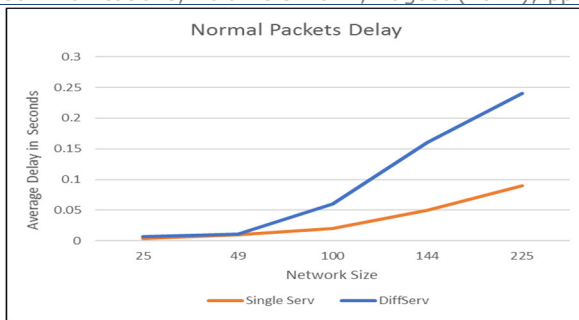


Figure 10 Normal Packets Average Delay

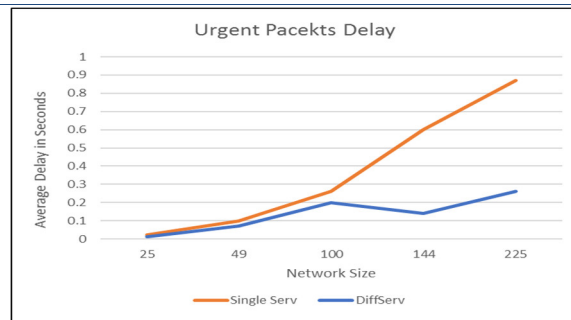


Figure 11 Urgent Packets Average Delay

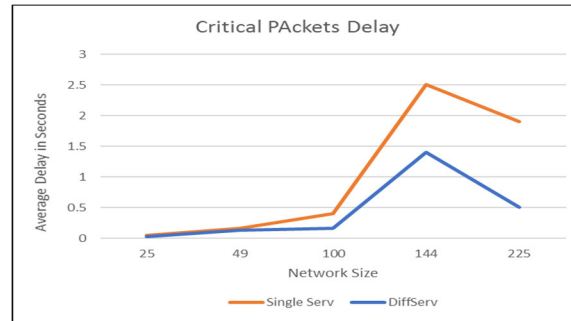


Figure 12 Critical Packets Average Delay

Fig. 10, compares the average delivery delay of normal packets delivered to the sink in both in both DiffServ and Single Serv cases. While figures 11 and 12 compare the average delivery delay of urgent and critical packets delivered to the sink node respectively.

As shown in Fig. 10, the average delivery delay of normal packets delivered to the sink node using Diffserv increased because normal packets are being stored in their own queue which has lower priority than urgent and critical queues. On the other hand, figures 11 and 12 show the decrease in the average delivery delay for urgent and critical packets when using Diffserv due to buffering these types of packets in separate queues from normal packets with higher priority.

After analyzing figures 10, 11, and 12 we can conclude that using differentiated services improved the packets' average delivery delay for urgent and critical packets over the expense of normal packets. We also, can notice how the improvement tends to get higher as the network size gets bigger. This is due to longer paths being used in larger networks.

6 Conclusions

QoS is a crucial feature in WSNs to ensure predictable performance in harsh environments. In this paper, we presented a routing protocol utilizing cross-layer communication where information from the application layer is used to take differentiated routing decisions by the network layer based on data packets classification. In our case, data packets are classified into: normal, urgent, and critical. Each data class is stored in its own designated routing buffer. Urgent packets buffer has the highest priority for transmission, then critical packets buffer, and lastly the normal packets buffer.

The performance of the differentiated services model was compared with the single service model through simulation experiments. Two experiments were conducted. The first experiment compared

packets delivery, the second experiment compared packets' average delivery delay. From these experimental results, it was shown that the differentiated services model performed better than the single service model in all aspects by increasing the packets delivery for urgent and critical packets and decreasing their delivery delays.

REFERENCES

- [1] Nikolaos A. Pantazis, S.A.N.a.D.D.V., "Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey". *Communications Surveys & Tutorials*, IEEE 2012. 15(2): p. 551 - 591.
- [2] Md. Atiqur Rahman, S.A., Md. Ileas Pramanik, Md. Ferdous Rahman. A "Survey on Energy Efficient Routing Techniques in Wireless Sensor Networks". *ICTACT*, 2013. p. 200 - 205.
- [3] Dr. Geoff V Merret, D.Y.K.T., "Wireless Sensor Networks: Application-Centric Design". InTech, 2010.
- [4] B. Bhuyan, H. Sarma, N. Sarma, A. Kar, R. Mall, *Quality of Service (QoS) Provisions in Wireless Sensor Networks and Related Challenges*, *Wireless Sensor Network*, 2010, Vol. 2, pp. 861-868
- [5] Kamil Samara, Hossein Hosseini, "Aware Diffusion: A Semi-Holistic Routing Protocol for Wireless Sensor Networks", *Wireless Sensor Network Journal*, Vol.8 No.3, March 2016, pp. 37-49
- [6] J. Mbowe, G. Oreku, *Quality of Service in Wireless Sensor Networks*, *Wireless Sensor Network*, 2014, 6, pp. 19-26
- [7] N. Ota, D. Hooks, P.Wright, D. Auslander, and T. Peffer., *Application to Demand Response Energy Pricing*, *Proceedings of the First international conference on Embedded Networked Sensor Systems*, November 05-07, 2003
- [8] E. Felemban, C. Lee, and E. Ekici. *MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks*, *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738-754, 2006
- [9] D. Ganesan et al. *Highly resilient, energy efficient multipath routing in wireless sensor networks*, *Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 11-25, 2002
- [10] X. Huang and Y. Fang. *Multiconstrained QoS multipath routing in wireless sensor networks*, *Wireless Networks Journal*, vol. 14, no. 4, pp. 465-478, 2007
- [11] C. Lu et al., *RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks*, In *Proceedings of the Eighth Real-Time and Embedded Technology and Applications Symposium*, IEEE CS Press, Los Alamitos, CA, 2002, pp. 55 - 66
- [12] T. He, J.A. Stankovic, C. Lu, and T.F. Abdelzaher. *SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks*, In *Proceedings of International Conference on Distributed Computing Systems (ICDCS '03)*, Providence, RI, May 2003, pp. 46 - 55

- [13] E. Felemban, C. Lee, and E. Ekici. MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks, IEEE Transactions on Mobile Computing, vol. 5, no. 6, pp. 738–754, 2006

- [14] K. Liu, N. Abu-Ghazaleh, and K.D. Kang. JiTS: Just-in-Time Scheduling for Real-Time Sensor Data Dissemination. In Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM'06), Washington, DC, IEEE Computer Society, Silver Spring, MD, 2006, pp. 42–46

- [15] Boulis, “Castalia user’s manual”, NICTA.