Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 627689, 18 pages
http://dx.doi.org/10.1155/2014/627689

Research Article

ZEQoS: A New Energy and QoS-Aware Routing Protocol for Communication of Sensor Devices in Healthcare System

Zahoor Ali Khan,[1] Shyamala Sivakumar,[2] William Phillips,[1] and Bill Robertson[1]

[1] *Internetworking Program, Faculty of Engineering, Dalhousie University, Halifax, NS, Canada B3H 4R2*
[2] *Saint Mary's University, Halifax, NS, Canada B3H 3C3*

Correspondence should be addressed to Zahoor Ali Khan; zahoor.khan@dal.ca

Received 2 November 2013; Revised 18 February 2014; Accepted 19 February 2014; Published 5 June 2014

Academic Editor: Christos Verikoukis

This paper proposes a novel integrated energy and QoS-aware routing protocol with the considerations of energy, end-to-end latency, and reliability requirements of body area network (BAN) communication. The proposed routing protocol, called ZEQoS, introduces two main modules (MAC layer and network layer) and three algorithms (neighbor table constructor, routing table constructor, and path selector). To handle ordinary packets (OPs), delay-sensitive packets (DSPs), and reliability-sensitive packets (RSPs), the new mechanism first calculates the communication costs, end-to-end path delays, and end-to-end path reliabilities of all possible paths from a source to destination. The protocol then selects the best possible path(s) for OPs, RSPs, and DSPs by considering their QoS requirement. Extensive simulations using OMNeT++ based simulator Castalia 3.2 demonstrate that the performance of the proposed integrated algorithm is satisfactory when tested on a real hospital scenario, and all data types including OPs, DSPs, and RSPs are used as offered traffic. Simulations also show that the ZEQoS also offers better performance in terms of higher throughput, less packets dropped on MAC and network layers, and lower network traffic than comparable protocols including DMQoS and noRouting.

1. Introduction

Various advanced and valuable state-of-the-art applications of body area networks (BANs) help enhance the patient's healthcare monitoring and their quality of life. The BAN devices are used to monitor the patients' health related concerns such as changes in blood pressure (BP), heart rate, or body temperature. In BAN communication, the body implanted and wearable sensors send their data to a central and computationally more powerful device known as the coordinator. The coordinator also behaves like a router in BAN networks. The BAN sensor nodes are, typically, required to use an extremely low transmission power to reduce health concerns and avoid tissue heating [1]. The low transmit power restricts the BAN transmission range to few meters (approximately three meters) [2]. One of the BAN features is to facilitate the physical mobility of the patient; this means that now the patients are not required to stay in the hospital at all times. Routing protocols are required to route a patient's data towards the required destination even when a patient moves. Routing is an issue for the sensor nodes due to the limited availability of resources including ultralow computation power, lower memory, and reduced energy source. The radio frequency (RF) portion of the sensor nodes in BAN plays a major role in the consumption of energy. MAC protocols can reduce the energy consumption by controlling the duty cycle of the RF part. MAC protocols are also helpful in effectively controlling the other sources that are the cause of energy waste, such as collision, idle listening, overhearing, and packet overhead. In short, an ideal MAC protocol increases error-free data transmission, maximum throughput, and medium access management and minimizes transmission delay, thereby increasing network lifetime. Despite the fact that MAC protocols are helpful in resolving many problems, the issues of end-to-end packet delivery, logical-physical address mapping, frame fragmentation, addressing techniques, and route determination methods are not in the scope of MAC protocols. These issues can

be more easily handled by the network layer. As a result, it is important to consider the network layer routing protocols to resolve these issues [3].

The challenges and features of BAN are different than WSN due to the specific needs of the wireless environment on the human body. The development of an efficient routing protocol in BAN requires more careful considerations than WSN. Some of the important factors to consider for the BAN routing protocols are their limited bandwidth, node and link heterogeneity, energy efficiency, coverage area, data aggregation, quality of service (QoS), transmit power, and mobile flexibility [3, 4].

The effect of fading, noise, and interference plays important role to reduce the effective bandwidth. The bandwidth available for BAN also varies due to these effects. The routing protocol can have only limited network control. The placement of sensor nodes during the formation of BAN is possible by a manual process. The nodes are placed manually on the predefined locations of the body where the data transmission is minimally disturbed by noise or interference. Ideally each node sends its own data and forwards the data received from other nodes towards the required destination. But in case of BAN, the implanted sensors, due to their tiny size and limited energy resources, only send the data to the central node or coordinator. The coordinator and other wearable nodes are capable of multihop communication, which helps route the data towards the desired destination. With the consideration of these facts, the routing protocol should be able to find and manage alternate routing paths in case of node failure.

Most of the nodes used in BAN are heterogeneous in terms of their capabilities including available energy, computational power, and communication capability. An example of heterogeneous nodes in BAN is the use of different wearable sensors to monitor body temperature, blood pressure, and other important vital signs of a patient. The link speeds of different implanted and wearable sensor nodes are not similar. The heterogeneity of the nodes should be considered by routing protocols.

The sensor nodes are placed on a human body that can be in motion. The node functionality may be affected due to mobility of the patient. This is because, the sensing capability of the mobile node can place increased energy demands on an application in different scenarios, for example, vital sign monitoring of a mobile patient indoor in the hospital is different than a patient in the outside environment of the hospital. With the mobility of the nodes, the routing protocol should be able to provide a suitable solution for the reliable communication.

Quality of service is one of the important factors in BAN communication. The reliability of associated algorithms improves the successful delivery of critical reliability-sensitive data from sensor nodes to the base station. The routing protocols fulfill the QoS demand of different BAN applications by using the delay-control algorithms. These QoS-aware protocols help monitor the patient's health during a critical situation [5, 6]. Our proposed routing protocol based on ordinary, delay-sensitive, and reliability-sensitive data is for the indoor hospital environment with the enhanced capability of handling mobile node communications.

The paper is organized as follows. Section 2 provides the motivation of this protocol. Section 3 explains the proposed routing protocol (ZEQoS). Sections 4 and 5 provide the MAC and network layer modules, respectively. Section 6 discusses the performance evaluation of ZEQoS. Section 7 demonstrates the superior performance of ZEQoS when compared with DMQoS and noRouting, and Section 8 summarizes this paper.

2. Related Work and Motivation

The consideration of quality of service (QoS) is an important but challenging task for the designers of BAN routing protocols. An ideal BAN routing protocol should provide an efficient and reliable path to route the patient's ordinary and critical data. The two important QoS routing protocols are reliability and delay-tolerant based protocols. The reliability-aware routing protocols ensure the delivery of maximum data packets to the destination. The transmission delay is not an issue for the reliability packets' delivery. For achieving the maximum throughput, data packets are sent on multiple redundant paths in some of the techniques used in reliability-aware protocols.

The delay-tolerant based routing protocols deal with the packets that are required to be delivered within a deadline. The route determination for the traffic of video streaming is one of the examples of this kind of routing. The end-to-end packet delay must be less than a specific delay; otherwise, the quality of overall data monitoring will be affected. Many routing protocols are proposed by researchers to address this issue. Researchers have proposed different energy and QoS-aware based routing protocols [7–16]. Some of the important QoS-aware routing protocols such as QoS-aware framework [9], RL-QRP [11], LOCALMOR [13], and DMQoS [17] are briefly discussed below.

In [9], a QoS-aware routing service framework for biomedical sensor networks is proposed based on a cross layered modular approach. The metrics considered for the determination of routes are wireless channel status, packet priority level, and sensor node's willingness to behave as a router. The proposed framework contains four main modules: an application programming interfaces (APIs) module, a routing service module, a packet queuing and scheduling module, and a system information repository module. The APIs module works as an interface between the user application and the routing service module. The components of APIs are QoS metrics selection, packet sending/receiving, packet priority level setting, and admission control and service level control. The QoS metrics are end-to-end delay, delivery ratio, and power consumption. The sensed data sent by user application for sink or other nodes is received by the packet sending/receiving component of APIs. These data packets contain destination ID, source ID, priority level, and payload. The data packets are received from the network layer. The payloads are forwarded to the user application for aggregation after separation from the data packets. The QoS-aware framework [9] is based on a modular technique that addresses QoS related issues for BAN. The newer routing techniques that consider the geographic location of neighbor

nodes prove very effective. The benefits of using geographic based routing include scalability, routing decisions based on neighborhood information, and being adaptive to dynamic environments. These protocols are also effective for mobile nodes. In this paper, the proposed protocol uses a similar modular approach but with the additional enhancements of location and energy aware routing.

RL-QRP [11] is a reinforcement learning based routing protocol with QoS support for biomedical sensor networks. The protocol focuses on two types of QoS requirements: packet delivery ratio and end-to-end delay. The machine learning approach used in this protocol uses optimal routing policies. These optimal routing policies can be found through experiences and rewards without the requirement of keeping precise network state information. RL-QRP [11] considers the neighborhood node's Q-values and location information for the determination of a QoS route. Energy is one of the major constraints in sensor nodes. The drawback of RL-QRP [11] is not considering energy at all. The proposed routing protocol, in this paper, considers the residual energy and geographic location of the next hop node, which helps improve the node lifetime.

LOCALMOR [13] is a QoS based BAN routing protocol that relies on the traffic diversity of biomedical applications and guarantees differentiated routing, based on using QoS metrics. The three different QoS requirements: (1) energy efficiency, (2) reliability, and (3) latency are considered in this protocol. The data traffic of biomedical applications is divided into four classes: regular, reliability-sensitive, delay-sensitive, and critical. A modular approach used in LOCALMOR consists of four modules: a power-efficiency module, a reliability-sensitive module, a delay-sensitive module, and a neighbor manager. Hello packets are used to update the neighbor's information in the neighbor table. The neighbor manager module is responsible to send/receive the Hello packets and manage the update of information of neighbors. The data from body sensor nodes transfer to the primary and secondary sinks via routers. LOCALMOR [13] provides a QoS-aware modular solution for different packet types. A data-centric multiobjective QoS-aware routing protocol (DMQoS) [17] outperforms the LOCALMOR [13]. The modular based architecture of DMQoS [17] provides the different routing modules to fulfill the QoS services for different packet classes. The reliability and delay control modules introduced in [17] result in better performance than several state-of-the-art approaches [11, 12, 15, 18–22] in terms of lower bit error rates, traffic load, and operation energy overload. The purpose of proposed energy and QoS-aware routing protocol (ZEQoS) is the reliable and energy-efficient routing similar to LOCALMOR and DMQoS. In this paper, the proposed routing protocol uses a similar modular approach and same packet classification as discussed in LOCALMOR and DMQoS. However, the mechanism of Hello protocol and calculation used for end-to-end path delays and end-to-end path reliabilities improves throughput and reduces the network traffic load. The simulation results prove that our protocol, ZEQoS, performs better than these protocols.

An energy-aware peering routing protocol (EPR) discussed in [23] is used to choose the best next hop for only ordinary packets (OPs) by considering the energy availability and geographic information of the devices. EPR has an overall lower energy consumption than comparable protocols [12, 15, 17, 20, 21] and provides better results in terms of reduced overall network traffic, reduced number of packets forwarded by intermediate nodes, and higher successful data transmission rates. In [5], the QPRD was extended to consider delay-sensitive packets (DSPs) as well as OPs. The resulting QPRD proposed an algorithm to route DSPs in addition to OPs. The redundant paths with the help of end-to-end path reliabilities are used in QPRR, discussed in [6], to ensure the reliable transmission of reliability-sensitive packets (RSPs) and OPs. These proposed routing protocols EPR, QPRD, and QPRR are not capable of handling OPs, RSPs, and DSPs simultaneously. For real-time display of patient data in the hospital environment, an energy and QoS-aware routing protocol is required that can handle all three data types (i.e., OPs, DSPs, and RSPs) simultaneously. With the integration of EPR, QPRD, and QPRR, in a unified BAN routing protocol, the ZEQoS provides a reliable solution for the transmission of OPs, RSPs, and DSPs and displays real-time BAN data.

3. Proposed Energy and QoS-Aware Routing Protocol (ZEQoS)

The proposed Zahoor energy and QoS-aware routing protocol (ZEQoS) is intended to be associated with the indoor hospital ZK-BAN peering framework [23]. To summarize, the ZK-BAN peering framework categorizes hospital devices into three types with the consideration of their energy levels. Figure 1 shows a general BAN communication framework. This hierarchical model has three communication tiers [24]. The sensor devices connected to body send data to the BAN coordinator (BANC) in tier 1. The BANC behaves like a cluster-head in WSNs. In tier 2, the possible next hop of a BANC is a BANC, medical display coordinator, nursing station coordinator, or a cellular device as shown in Figure 1. The tier 2 communication devices with the exclusion of the BANC forward the BAN data to tier 3 communication devices.

The device directly connected with the power source is considered as type 1 device such as nursing station coordinator (NSC). Devices with replaceable batteries (e.g., medical display coordinators (MDCs)) and nonreplaceable batteries (e.g., body area network coordinators (BANCs)) are counted in type 2 and type 3 devices, respectively, and this is illustrated in Table 1.

According to ZK-BAN peering framework [23], the information of BANCs and their respective peer MDCs are stored at the NSC. This framework uses a hybrid communication mode that can be in one of two modes, a centralized mode or a distributed mode as appropriate. Hybrid communication helps increase privacy and save energy consumption. In centralized mode, the BANCs get the information of its respective peer from the NSC. In distributed mode, BANCs send the data reliably to their peer MDC in order to achieve the purpose of real-time display of patient data. The detailed

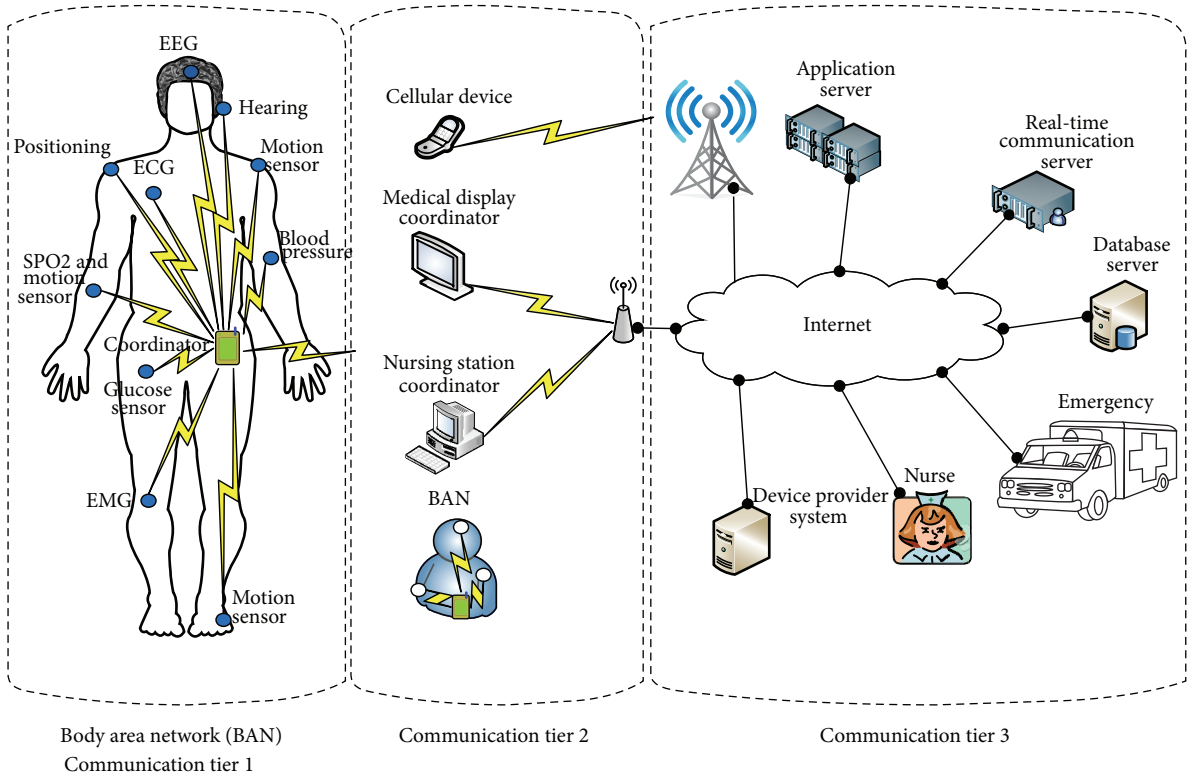


Figure 1: General BAN communication system.

Table 1: Classification of devices in hospital environment.

Class	Device name	Power source	Channels	MAC protocol	Mobility
1	NSC	Directly connected	2	IEEE 802.15.4 IEEE 802.11	No
2	MDC	Replaceable batteries	2	IEEE 802.15.4 IEEE 802.11	Yes
3	BANC	Limited energy available	1	IEEE 802.15.4	Yes

discussion of ZK-BAN peering framework can be found in [23].

ZEQoS calculates the best next hops for OPs, DSPs, and RSPs with the help of different modules and algorithms. The next hop for OPs is denoted by NH_E. The selection of NH_E is based on the communication cost (C_i) which is calculated with the consideration of geographic and energy information of the neighbor nodes. The ZEQoS employs a Hello protocol, discussed in [23], that is used to broadcast the important information of a node to the other nodes. For DSPs, ZEQoS calculates the node delay and end-to-end path delays of all possible paths from source to destination and then chooses the next hop (i.e., NH_D) device based on the lowest end-to-end path delay. For RSPs, ZEQoS (1) computes the end-to-end path reliabilities of all possible paths, (2) selects the three most reliable paths for each destination, (3) determines the degree of path redundancy, and (4) chooses the next hop device(s) based on the most reliable end-to-end path(s) from the source node to the destination. ZEQoS improves the reliability with the help of redundant paths. The architecture of proposed ZEQoS routing protocol is shown in Figure 2 and notations used in this protocol are given in Table 2.

The modules used in ZEQoS are spread into two layers: MAC layer and network layer. MAC and network layer modules are discussed below.

4. MAC Layer Modules

The MAC layer contains four modules: MAC receiver, reliability module, delay module, and MAC transmitter. The data or Hello packets from other nodes (i.e., BANC, MDC, or NSC) are received by MAC receiver of the node i. MAC receiver checks the MAC address of the packets and only forwards the packets, which contain the broadcast address or MAC address of the node i as destination address, to the network layer. The reliability module of node i on MAC layer calculates the numbers of packets sent to neighbor node j and the number of acknowledgements received

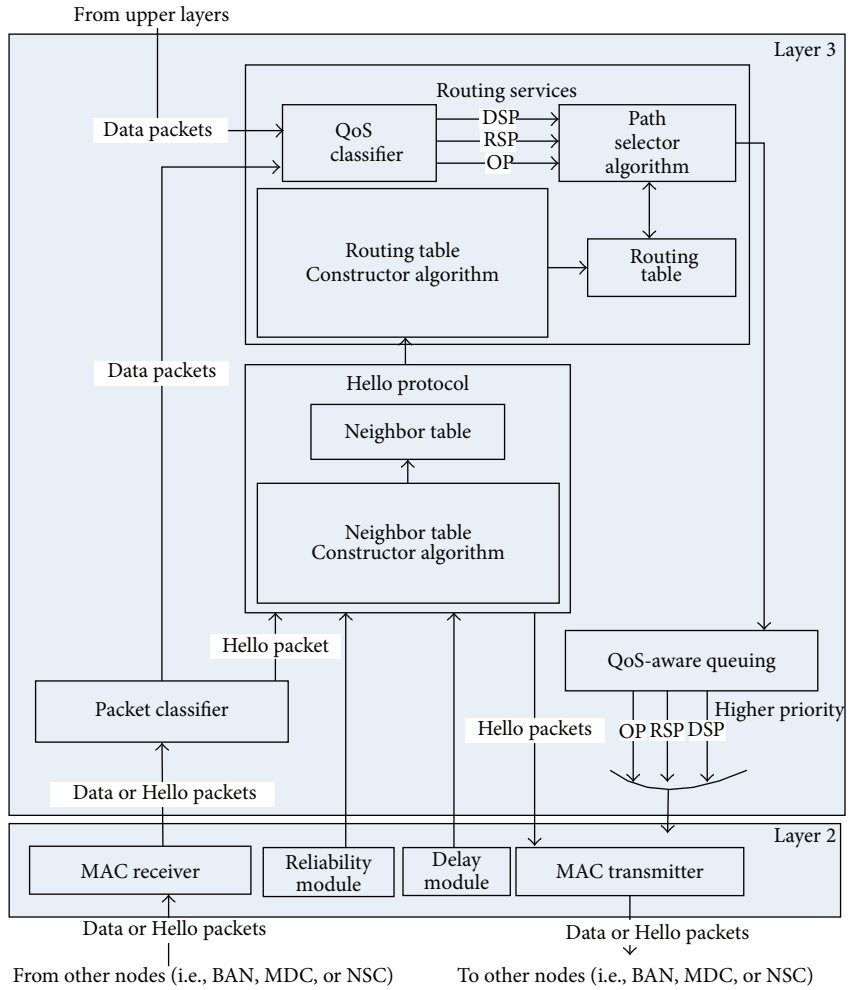


Figure 2: ZEQoS routing protocol architecture.

ID_{Dst}	L_{Dst}	ID_j	L_j	$D_{(j,Dst)}$	E_j	T_j	$R_{path(j,Dst)}$	$DL_{path(j,Dst)}$

Figure 3: Hello packet structure.

from neighbor node j. The delay module monitors the time required to capture the channel ($DL_{channel(i)}$), MAC layer queuing delay ($DL_{MAC\_queue(i)}$), and transmission time ($DL_{trans(i)}$) of a packet. The delay and reliability modules send their information to the Hello protocol module of the network layer. The neighbor table constructor algorithm in Hello protocols module uses this information to calculate the node delay ($DL_{node(i)}$) and the link reliability between the node i and the neighbor node j ($R_{link(i,j)}$).

The data and Hello packets from the network layer are received by the MAC transmitter submodule which stores these packets in the MAC layer queue. The MAC layer queue works in a first-in-first-out (FIFO) fashion. MAC transmitter uses CSMA/CA algorithm to send the data when the channel is captured.

5. Network Layer Modules

Network layer consists of four modules: packet classifier (PC), Hello protocol module (HPM), routing services module (RSM), and QoS-aware queuing module (QQM). The detailed discussion of these modules is given below.

5.1. Packet Classifier. The packet classifier receives data and Hello packets from the MAC receiver module of the MAC layer. The job of packet classifier is to differentiate and forward the data packets and Hello packets to the routing services module and Hello protocol module, respectively.

5.2. Hello Protocol Module (HPM). According to the Hello protocol, type 1 and type 2 devices (NSC or MDCs) send Hello packets periodically and the BANCs broadcast their Hello packets only at the reception of other nodes' Hello packets which contain the NSC or MDC information. The Hello packet fields of node j are shown in Figure 3. The possible destination (Dst) can be a NSC, MDC, or BANC. The Hello packet contains the information about the destination device

Table 2: Notations for the proposed algorithm.

Field ID	Description
Node i	Source node
Node j	Neighbor node of source node
Node Dst.	Destination node (i.e. NSC, MDCs, and BAN)
$\mathrm{ID}_{\mathrm{Dst}}$	Destination ID
L_{Dst}	Destination location
ID_j	Neighbor node j ID
L_j	Neighbor node j location
$D_{(j,\mathrm{Dst})}$	Distance between neighbor node j and destination Dst.
E_j	Residual energy of node j
C_j	Communication cost
T_j	Device type of node j
$R_{\mathrm{path}(j,\mathrm{Dst})}$	Path reliability between neighbor j and destination
$D_{(i,j)}$	Distance between node i to neighbor node j
$R_{\mathrm{link}(i,j)}$	Link reliability from node i to neighbor node j
$R_{\mathrm{path}(i,\mathrm{Dst})}$	Path reliability from node i to destination Dst.
$\mathrm{NH}_{(i,\mathrm{Dst})}$	Next hop between node i and destination Dst.
NH_E	Energy-aware next hop
NH_{R1}	1st reliable next hop
NH_{R2}	2nd reliable next hop
NH_{R3}	3rd reliable next hop
NH_D	Next hop for delay-sensitive packets
$\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$	Path delay from node i to destination Dst.
$\mathrm{DL}_{\mathrm{node}(i)}$	Time delay within the node i
$\mathrm{DL}_{\mathrm{req}}$	Required path delay for delay-sensitive packets
R_{req}	Required reliability of reliability-sensitive packets
$R_{\mathrm{option1}(i,\mathrm{Dst})}$	1st option reliability for sending reliability-sensitive packets
$R_{\mathrm{option2}(i,\mathrm{Dst})}$	2nd option reliability for sending reliability-sensitive packets
$R_{\mathrm{option3}(i,\mathrm{Dst})}$	3rd option reliability for sending reliability-sensitive packets

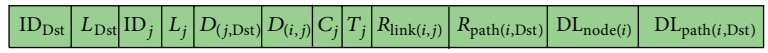


Figure 4: Neighbor table structure.

ID ($\mathrm{ID}_{\mathrm{Dst}}$), destination location (L_{Dst}), sender's ID (ID_j), residual energy (E_j), device type (T_j), distance ($D_{(j,\mathrm{Dst})}$), path reliability ($R_{\mathrm{path}(j,\mathrm{Dst})}$), and path delay ($\mathrm{DL}_{\mathrm{path}(j,\mathrm{Dst})}$). The subscript (j, Dst) means from sender node j to the destination.

The node i receives the Hello packet. The information received from the reliability module, delay module, and Hello packets of the MAC receiver module is used by the neighbor table constructor algorithm to construct the neighbor table. The neighbor table constructor algorithm of node i calculates its own $\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$ and $R_{\mathrm{path}(i,\mathrm{Dst})}$ based on the information in the Hello packets. Node i updates the values of Hello packet fields and broadcasts it to the other nodes. The mechanism of Hello protocol used in ZEQoS is described briefly in the following paragraph and is described in [23].

The neighbor table and neighbor table constructor algorithm are the two submodules of the Hello protocol module. In addition to Hello packet fields, the neighbor table contains fields for both hop-by-hop delay ($\mathrm{DL}_{\mathrm{node}(i)}$) and reliability ($R_{\mathrm{link}(i)}$) and end-to-end delay ($\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$) and reliability ($R_{\mathrm{path}(i,\mathrm{Dst})}$). Neighbor table also uses communication cost (C_i) instead of residual energy (E_i). The neighbor table structure of node i is shown in Figure 4.

5.2.1. Neighbor Table Constructor Algorithm. The neighbor table constructor algorithm updates the values of the neighbor table fields periodically after receiving every new Hello packet. Neighbor table constructor algorithm calculates the values of the additional field used in neighbor table such as $\mathrm{DL}_{\mathrm{node}(i)}$, $R_{\mathrm{link}(i)}$, $\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$, $R_{\mathrm{path}(i,\mathrm{Dst})}$, C_i, and $D_{(i,j)}$. The terms rm, hp, dm, and nt used in Algorithm 1 stand for reliability module, Hello packet, delay module, and neighbor table, respectively.

INPUT: Hello Packet, at each node i.

(1) $\overline{X}_i = \dfrac{N_{\text{Acks}}(\text{rm})}{N_{\text{Trans}}(\text{rm})}$
(2) $\rho_r \leftarrow 0.4$
(3) $R_{\text{link}(i,j)} = (1 - \rho_r) * R_{\text{link}(i,j)} + \rho_r * \overline{X}_i$
(4) $R_{\text{path}(i,\text{Dst})} = R_{\text{link}(i,j)} + R_{\text{path}(j,\text{Dst})}(\text{hp})$
(5) $\rho_d \leftarrow 0.2$
(6) $\text{DL}_{\text{queue+channel}} \leftarrow$ First packet delay
(7) $\text{DL}_{\text{queue+channel}} = (1 - \rho_d) * \left(\text{DL}_{\text{MAC\_queue}}(\text{dm}) + \text{DL}_{\text{channel}}(\text{dm}) + \text{DL}_{\text{Net\_queue}}\right) + \rho_d * \left(\text{DL}_{\text{MAC\_queue}}(\text{dm}) + \text{DL}_{\text{channel}}(\text{dm}) + \text{DL}_{\text{Net\_queue}}\right)$
(8) $\text{DL}_{\text{node}(i)} = \text{DL}_{\text{trans}(i)}(\text{dm}) + \text{DL}_{\text{queue+channel}} + \text{DL}_{\text{proc}}$
(9) $\text{DL}_{\text{path}(i,\text{Dst})} = \text{DL}_{\text{node}(i)} + \text{DL}_{\text{path}(j,\text{Dst})}(\text{hp})$
(10) $D_{(i,j)} = \sqrt{\left(X_i - X_j\right)^2 + \left(Y_i - Y_j\right)^2}$
(11) $C_j = \dfrac{\left(T_j(\text{hp}) * D^2_{(i,j)}(\text{hp})\right)}{E_j(\text{hp})}$
(12) $D_{(i,\text{Dst})} = \sqrt{\left(X_i - X_{\text{Dst}}\right)^2 + \left(Y_i - Y_{\text{Dst}}\right)^2}$
(13) **if** $\left(D_{(j,\text{Dst}}(\text{hp}) < D_{(i,\text{Dst})}\right)$ **then**
(14) (add a new record for the Dst's information in the neighbor table)
(15) $\text{ID}_{\text{Dst}}(\text{nt}) \leftarrow \text{ID}_{\text{Dst}}(\text{hp})$
(16) $\text{ID}_j(\text{nt}) \leftarrow \text{ID}_j(\text{hp})$
(17) $L_j(\text{nt}) \leftarrow L_j(\text{hp})$
(18) $D_{(j,\text{Dst})}(\text{nt}) \leftarrow D_{(j,\text{Dst})}(\text{hp})$
(19) $D_{(i,j)}(\text{nt}) \leftarrow D_{(i,j)}$
(20) $C_j(\text{nt}) \leftarrow C_j$
(21) $T_j(\text{nt}) \leftarrow T_j(\text{hp})$
(22) $R_{\text{link}(i,j)}(\text{nt}) \leftarrow R_{\text{link}(i,j)}$
(23) $R_{\text{path}(i,\text{Dst})}(\text{nt}) \leftarrow R_{\text{path}(i,\text{Dst})}$
(24) $\text{DL}_{\text{node}(i)}(\text{nt}) \leftarrow \text{DL}_{\text{node}(i)}$
(25) $\text{DL}_{\text{path}(i,\text{Dst})}(\text{nt}) \leftarrow \text{DL}_{\text{path}(i,\text{Dst})}$
(26) **end if**
(27) (add a new record for the neighbor node j's information in the neighbor table)
(28) $\text{ID}_{\text{Dst}}(\text{nt}) \leftarrow \text{ID}_{(\text{Dst})}(\text{hp})$
(29) $\text{ID}_j(\text{nt}) \leftarrow \text{ID}_j(\text{hp})$
(30) $L_j(\text{nt}) \leftarrow L_j(\text{hp})$
(31) $D_{(j,\text{Dst})}(\text{nt}) = 0$
(32) $D_{(i,j)}(\text{nt}) \leftarrow D_{(i,j)}$
(33) $C_j(\text{nt}) \leftarrow C_j$
(34) $T_j(\text{nt}) \leftarrow T_j(\text{hp})$
(35) $R_{\text{link}(i,j)}(\text{nt}) \leftarrow R_{\text{link}(i,j)}$
(36) $R_{\text{path}(i,\text{Dst})}(\text{nt}) \leftarrow R_{\text{path}(i,\text{Dst})}$
(37) $\text{DL}_{\text{node}(i)}(\text{nt}) \leftarrow \text{DL}_{\text{node}(i)}$
(38) $\text{DL}_{\text{path}(i,\text{Dst})}(\text{nt}) \leftarrow \text{DL}_{\text{path}(i,\text{Dst})}$

Algorithm 1: Neighbor table constructor algorithm for ZEQoS.

The average probability of successful transmission $\overline{X}_i$ after every 4 seconds is calculated by using (1). Consider

$$\overline{X}_i = \frac{N_{\text{Acks}}}{N_{\text{Trans}}}, \tag{1}$$

where N_{Acks} = number of acknowledgements and N_{Trans} = number of transmissions.

The link reliability between node i and neighbor node j ($R_{\text{link}(i,j)}$) is calculated by using the exponentially weighted moving average (EWMA) equation (2). Consider

$$R_{\text{link}(i,j)} = (1 - \rho_r) R_{\text{link}(i,j)} + \rho_r \times \overline{X}_i, \tag{2}$$

where ρ_r is the average weighting factor that satisfies $0 < \rho_r \leq 1$. Algorithm 1 uses $\rho_r = 0.4$.

The path reliability between node i and destination node Dst ($R_{\text{path}(i,\text{Dst})}$) is calculated by using (3). Consider

$$R_{\text{path}(i,\text{Dst})} = R_{\text{link}(i,j)} \times R_{\text{path}(j,\text{Dst})}. \tag{3}$$

ID_{Dst}	L_{Dst}	NH_E	NH_{R1}	NH_{R2}	NH_{R3}	NH_D	$R_{option1(i,Dst)}$	$R_{option2(i,Dst)}$	$R_{option3(i,Dst)}$	$DL_{path(i,Dst)}$

Figure 5: Routing table structure.

The values of $R_{link(i,j)}$ and $R_{path(j,Dst)}$ are used from (2) and Hello packet (hp), respectively. The calculation of finding $R_{path(i,Dst)}$ is given in Algorithm 1 (lines 1–4).

The delay due to the queues of MAC and network layers and channel capture ($DL_{queue+channel}$) is calculated by using the exponentially weighted moving average (EWMA) formula. Consider

$$\begin{aligned} DL_{queue+channel} &= (1-\rho_d) * \left(DL_{MAC\_queue}(dm) + DL_{channel}(dm) + DL_{Net\_queue}\right) \\ &+ \rho_d * \left(DL_{MAC\_queue}(dm) + DL_{channel}(dm) + DL_{Net_{queue}}\right), \end{aligned} \quad (4)$$

where the values of MAC queue delay and channel capture time are received from delay module (dm), whereas the values of network queue delays are calculated on network layer. The initial value of $DL_{queue+channel}$ is the delay of the first packet sent by the node. The selection of ρ_d value is the personal choice and experience, but it should satisfy $0 < \rho_d \leq 1$. The recommended values are $0.2 \leq \rho_d \leq 0.3$. Algorithm 1 uses $\rho_d = 0.2$.

The value of node delay ($DL_{node(i)}$) is calculated with the addition of the packet delays due to transmission, queuing, processing, and capturing of the channel. Consider

$$DL_{node(i)} = DL_{trans(i)}(dm) + DL_{queue+channel} + DL_{proc}. \quad (5)$$

The path delay between node i and destination node Dst ($DL_{path(i,Dst)}$) is calculated by using (6). Consider

$$DL_{path(i,Dst)} = DL_{node(i)} + DL_{path(j,Dst)}(hp), \quad (6)$$

where initial value of $DL_{path(j,Dst)}$ is zero when $j = Dst$.

The values of $DL_{node(i)}$ are calculated in (5) and $DL_{path(n,Dst)}$ is received from Hello packet (hp).

The calculation of finding $DL_{path(i,Dst)}$ is shown in Algorithm 1 from lines 5–9.

Algorithm 1 (lines 11-12) calculates the communication cost (C_j) and distance from node i to the neighbor node $j(D_{(i,j)})$ by using (7). Consider

$$\begin{aligned} D_{(i,j)} &= \sqrt{\left(X_i - X_j\right)^2 + \left(Y_i - Y_j\right)^2}, \\ C_j &= \frac{\left(T_j * D_{(i,j)}^2\right)}{E_j}, \end{aligned} \quad (7)$$

where X_i, Y_i stand for the X, Y coordinates of node i and X_{DST}, Y_{DST} represent the X, Y coordinates of the destination. It is also assumed that the locations of NSC and MDCs are known. The RSSI localization technique given in [25] is used to calculate the values of X_i and Y_i of the node i. The values of T_j, $D_{(i,j)}$, and E_j are received from Hello packet (hp). The shorter distance ($D_{(i,j)}$), lower device type (T_j), and higher residual energy (E_j) will generate a lower communication cost (C_j). The node j with lowest value of C_j is the best choice for next hop.

Lines 13–26 of Algorithm 1 show that a new record for the destination is added in neighbor table if the distance from the neighbor node j to the destination ($D_{(j,Dst)}$) is less than the distance from the node i to the destination; that is, $D_{(j,Dst)}(hp) < D_{(i,Dst)}$.

A new record with the information of the neighbor node j is also added with the new calculated values as shown in Algorithm 1 from lines 27–38.

The neighbor table constructor algorithm repeats the same process of updating the neighbor table after receiving every new Hello packet.

5.3. Routing Services Module. The routing services module contains four submodules: QoS classifier, routing table constructor algorithm, routing table, and path selector algorithm. The QoS classifier submodule is responsible for categorizing the data packets into delay-sensitive packets (DSPs), reliability-sensitive packets (RSPs), and ordinary packets (OPs). The routing table constructor algorithm is used to construct and update the routing table. The routing table submodule stores the required information of the next hop(s) for the data packets. The routing table structure for node i is shown in Figure 5. The path selector algorithm chooses the best path(s) for each category (DSP, RSP, or OP) of traffic, based on the QoS requirement.

5.3.1. Routing Table Constructor Algorithm. The neighbor table entries are used to construct the routing table. Neighbor table contains multiple records for each destination. The routing table constructor algorithm determines the best next hops OPs, RSPs, and DSPs. It filters the neighbor table and only chooses an entry with the best values for the routing table. As shown in Algorithm 2, a new record is added in the routing table for each destination Dst $\in$ {MDC, NSC, BAN}. Lines 2–8, 9–27, and 28–34 are used to determine the values related to the OPs, RSPs, and DSPs, respectively.

The next hop for OPs (NH_E) will be the destination ID (ID_{Dst}) if the neighbor node is also the destination node (line 2). Otherwise a neighbor node j with the lowest communication cost (C_j) will be selected as next hop (NH_E).

For RSPs, the routing table constructor algorithm of ZEQoS finds three possible paths to ensure the minimum required reliability. For each destination, the three paths with highest reliabilities ($R_{path1(i,Dst)}$, $R_{path2(i,Dst)}$, and $R_{path3(i,Dst)}$) are chosen and their corresponding next hops (NH_{R1}, NH_{R2},

INPUT: Neighbor table, i's neighbor table records $\mathrm{NH}_{(i,\mathrm{Dst})}$, $\forall$ Dst $\in$ {MDC, NSC, BAN}

(1) **for** each destination Dst $\in$ {NSC, MDC, BAN} **do**
(2) **if** ($\mathrm{ID}_j(\mathrm{nt}) == \mathrm{ID}_{\mathrm{Dst}}(\mathrm{nt})$) **then**
(3) $\mathrm{NH}_E \leftarrow \mathrm{ID}_{\mathrm{Dst}}(\mathrm{nt})$
(4) **else**
(5) **if** ($C_j == \min_{k \in \mathrm{NH}_{(i,\mathrm{Dst})}} C_k$) **then**
(6) $\mathrm{NH}_E \leftarrow \mathrm{ID}_j(\mathrm{nt})$
(7) **end if**
(8) **end if**
(9) NH_R = {All neighbor nodes $j \in \mathrm{NH}_{(i,\mathrm{Dst})}$}
(10) **if** ($\mathrm{NH}_R == \mathrm{NULL}$) **then**
(11) Put NULL in $\mathrm{NH}_{R1}, \mathrm{NH}_{R2}, \mathrm{NH}_{R3}, R_{\mathrm{option1}(i,\mathrm{Dst})}, R_{\mathrm{option2}(i,\mathrm{Dst})}, R_{\mathrm{option3}(i,\mathrm{Dst})}$
(12) **else**
(13) Sort NH_R in descending order of $R_{\mathrm{path}(i,\mathrm{Dst})}$
(14) NH_{R1} = first neighbor node $j \in \mathrm{NH}_R$
(15) $R_{\mathrm{option1}(i,\mathrm{Dst})} = R_{\mathrm{path}(i,\mathrm{Dst})}$
(16) $P_{\mathrm{error}} = 1 - R_{\mathrm{option1}(i,\mathrm{Dst})}$
(17) **if** ($|\mathrm{NH}_R| > 1$)
(18) NH_{R2} = second neighbor node $j \in \mathrm{NH}_R$
(19) $P_{\mathrm{error}} = P_{\mathrm{error}} * (1 - R_{\mathrm{path}(i,\mathrm{Dst})})$
(20) $R_{\mathrm{option2}(i,\mathrm{Dst})} = 1 - P_{\mathrm{error}}$
(21) **end if**
(22) **if** ($|\mathrm{NH}_R| > 2$)
(23) NH_{R3} = third neighbor node $j \in \mathrm{NH}_R$
(24) $P_{\mathrm{error}} = P_{\mathrm{error}} * (1 - R_{\mathrm{path}(i,\mathrm{Dst})})$
(25) $R_{\mathrm{option3}(i,\mathrm{Dst})} = 1 - P_{\mathrm{error}}$
(26) **end if**
(27) **end if**
(28) NH = {All neighbor nodes $j \in \mathrm{NH}_{(i,\mathrm{Dst})}$}
(29) **if** ($|\mathrm{NH}| == 1$) **then**
(30) $\mathrm{NH}_D \leftarrow \mathrm{NH}$
(31) **else if** ($|\mathrm{NH}| > 1$) **then**
(32) Sort NH in ascending order of $\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$
(33) NH_D = first neighbor node $j \in \mathrm{NH}$
(34) **end if**
(35) (add a new record for the Dst's information in the routing table)
(36) $\mathrm{ID}_{\mathrm{Dst}} \leftarrow \mathrm{ID}_{\mathrm{Dst}}(\mathrm{nt})$
(37) $L_{\mathrm{Dst}} \leftarrow L_{\mathrm{Dst}}(\mathrm{nt})$
(38) $\mathrm{NH}_E \leftarrow \mathrm{NH}_E$
(39) $\mathrm{NH}_{R1} \leftarrow \mathrm{NH}_{R1}$
(40) $\mathrm{NH}_{R2} \leftarrow \mathrm{NH}_{R2}$
(41) $\mathrm{NH}_{R3} \leftarrow \mathrm{NH}_{R3}$
(42) $\mathrm{NH}_D \leftarrow \mathrm{NH}_D$
(43) $\mathrm{NH}_{\mathrm{option1}(i,\mathrm{Dst})} \leftarrow \mathrm{NH}_{\mathrm{option1}(i,\mathrm{Dst})}$
(44) $\mathrm{NH}_{\mathrm{option2}(i,\mathrm{Dst})} \leftarrow \mathrm{NH}_{\mathrm{option2}(i,\mathrm{Dst})}$
(45) $\mathrm{NH}_{\mathrm{option3}(i,\mathrm{Dst})} \leftarrow \mathrm{NH}_{\mathrm{option3}(i,\mathrm{Dst})}$
(46) $\mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})} \leftarrow \mathrm{DL}_{\mathrm{path}(i,\mathrm{Dst})}$
(47) **end for**

Algorithm 2: Routing table constructor algorithm for ZEQoS.

and NH_{R3}) are stored in the routing table. The routing table constructor calculates and stores the three options for RSP. Line 9 of Algorithm 2 shows that the node i identifies the next hop candidates by searching the records which have the same $\mathrm{ID}_{\mathrm{Dst}}$ in neighbor table and stores them in the variable NH_R. If NH_R is empty, it means there is no next hop stored in NH_R. The node stores NULL to NH_{R1}, NH_{R2}, NH_{R3}, $R_{\mathrm{option1}(i,\mathrm{Dst})}$, $R_{\mathrm{option2}(i,\mathrm{Dst})}$, and $R_{\mathrm{option3}(i,\mathrm{Dst})}$. If NH_R is not empty, the next hop nodes' information is stored in the routing table one after another in descending order of their path reliabilities $R_{\mathrm{path}(i,\mathrm{Dst})}$. The first neighbor node j with the highest reliability in the routing table is stored as NH_{R1} (line 14). If there are two entries in NH_R then the aggregate reliability of first and second paths ($R_{\mathrm{option2}(i,\mathrm{Dst})}$) is calculated (lines 17–21). In case of more than two entries in NH_R, the aggregate reliability of first, second, and third

```
INPUT: Routing table, i's routing table records NH(i,Dst), ∀ Dst ∈ {MDC, NSC, BAN}
(1)  for each data packet do
(2)   if data packet is delay-sensitive packet (DSP)
(3)     if (DL path(i,Dst) <= DL req) then
(4)       send to NH D
(5)     else
(6)        drop the packet immediately
(7)     end if
(8)     else if data packet is reliability-sensitive packet (RSP)
(9)       if (R option1(i,Dst) > R req)
(10)        send to NH R1
(11)      else if (R option2(i,Dst) > R req)
(12)        send to NH R1 and NH R2
(13)        else if (R option3(i,Dst) > R req)
(14)          send to NH R1, NH R2 and NH R3
(15)          else
(16)           drop the packet immediately
(17)      end if
(18)    else if data packet is Ordinary Packet (OP)
(19)        send to NH E
(20)         else
(21)        drop the packet immediately
(22)  end if
(23) end for
```

Algorithm 3: Path selector algorithm for ZEQoS.

paths ($R_{\text{option3}(i,\text{Dst})}$) is calculated (lines 22–26). In the routing table, the three paths with highest reliabilities ($R_{\text{path1}(i,\text{Dst})}$, $R_{\text{path2}(i,\text{Dst})}$, and $R_{\text{path3}(i,\text{Dst})}$) are chosen and their corresponding next hops (NH_{R1}, NH_{R2}, and NH_{R3}) are stored for each destination in the routing table. The routing table constructor calculates and stores the three options for RSP. The detailed calculations of $R_{\text{option1}(i,\text{Dst})}$, $R_{\text{option2}(i,\text{Dst})}$, and $R_{\text{option3}(i,\text{Dst})}$ are discussed in earlier work [6].

For DSP data, the path delay $\text{DL}_{\text{path}(i,\text{Dst})}$ has been calculated by using the neighbor table constructor algorithm (line 9 of Algorithm 1) and stored in neighbor table for each next hop candidate. The node stores the neighbor node's IDs in the variable NH (line 28). If NH has only one entry, this means there is only one path available. The node stores this entry to NH_D (line 30). Otherwise the node sorts the NH entries in ascending order with respect to the path delay (i.e., $\text{DL}_{\text{path}(i,\text{Dst})}$) values and then stores the first entry which has the lowest path delay in NH_D (lines 32-33). The next hop candidate NH_D is then stored with its path delay value ($\text{DL}_{\text{path}(i,\text{Dst})}$) in the routing table. Algorithm 2 (lines 27–38) shows that a new record for the destination Dst is added with the calculated values.

The routing table constructor algorithm repeats the same process of updating the routing table after receiving every new Hello packet.

5.3.2. Path Selector Algorithm. The data packets from both upper layers and packet classifiers are received by QoS classifier. The QoS classifier classifies the packets into DSP, RSP, and OP data. For each data packet, the path selector algorithm checks the QoS requirement and chooses the most appropriate next hop(s). Lines 2–7, 8–17, and 18–21 of Algorithm 3 are used for the selection of appropriate next hops of DSPs, RSPs, and OPs, respectively. The path selector algorithm compares the delay requirement (DL_{req}) with the path delay ($\text{DL}_{\text{path}(i,\text{Dst})}$) of NH_D which is stored in the routing table. If the path delay ($\text{DL}_{\text{path}(i,\text{Dst})}$) is lower than required delay (DL_{req}), the packet is sent to NH_D (lines 3-4). Otherwise, the packet is dropped (line 6).

For RSPs, the path selector algorithm checks if the reliability of a single path exceeds R_{req}; then a single path is used to send these packets through NH_{R1} (lines 9-10). In case the required reliability is greater than the reliability of any single path, then, the path selector selects two paths (by using NH_{R1} and NH_{R2}) whose aggregate reliability is more than the requested R_{req} (lines 11-12). If not, three paths are used as long as their aggregate reliability is greater than the R_{req} (lines 13-14) or else the packet is dropped. The method of finding the aggregate reliabilities is given in Algorithm 2 and discussed in Section 5.3.1. For OPs, the path selector algorithm returns the next hop NH_E (lines 18-19). Any unknown packet should be dropped without assigning any next hop (line 21).

5.4. QoS-Aware Queuing Module. The data packets are sent to the QoS-aware queuing module (QQM) after the selection of appropriate next hop(s) by routing services module. QQM receives the data packets and separates these packets in three classes (DSPs, RSPs, and OPs). An individual queue is used for each class of packets. QQM functions are the same as discussed in [17]. The priority of the DSPs queue is higher than that of the RSPs and OPs queues. The RSPs queue has lower priority than DSPs queue. The priority of OPs queue is

Table 3: Parameters information.

Deployment	Area	16 m by 21 m
	Deployment type	Movable source node BAN_2 (shown in Figure 6)
	Number of nodes	49 nodes (24 BANs, 24 MDCs, and 1 NSC)
	Initial nodes locations	As shown in Figure 6
	Initial node energy	18720 J (=2 AA batteries)
	Buffer size	32 packets
	Link layer trans. rate	250 Kbps
	Transmit power	−25 dBm
Task	Application type	Event-driven
	Max. packet size	32 Bytes
	Traffic type	CBR (constant bit rate)
MAC	IEEE 802.15.4	Default values
Simulation	Time	2003 seconds (3 seconds is setup time)

the lowest. By default, the DSPs queue with highest priority sends the packets first. The packets from lower priority RSP queue will be sent only when the DSPs queue is empty. The OPs need to wait until the DSPs and RSPs queues are empty. However, for fair treatment of OPs data, a timeout is used by all the queues. A queue sends the packets to the MAC layer within the period specified by the timeout for that queue. QQM changes the control from higher priority queue to lower priority queue after the queue timeout occurs.

6. Performance Evaluation

OMNeT++ based simulator Castalia [26] is used to test the performance of the proposed ZEQoS routing protocol. The simulation results prove that the ZEQoS approach based on end-to-end path delays and reliabilities in addition to the available energy and geographic information of the node are more effective for all data types (i.e., OPs, DSPs, and RSPs) when compared with DMQoS and noRouting protocols. The simulations are done by considering a real 24-bed hospital scenario outlined in Section 6.1. The details about the scenario, parameters information, and performance results for the simulations are provided below.

6.1. 49 Nodes in Hospital Environment. A real 24-patient bed hospital with a movable source node is considered for the testing of ZEQoS routing protocol, as shown in Figure 6. The approximate measurements used for this hospital environment are similar to the hematology-oncology unit of the children's hospital named IWK Health Centre Halifax, NS, Canada. The approximate area covered by this unit is 16 m by 21 m. The distance between two beds is 3 meters which is a recommended transmission range for BAN communication in hospital environment. Each BAN transmits the data to its respective MDC. All the BANs and MDCs are sending or receiving Hello protocols to/from other nodes and the NSC. The total numbers of nodes used in this scenario are 49 which include 24 BANs, 24 MDCs, and 1 NSC. The NSC is placed on the left side of the deployment area. The patient rooms are in four rows. The room numbers 1–7, 8–12, 13–17, and 10–24 are in rows 1, 2, 3, and 4, respectively. Room number 18 and the nursing station are just in front of all these rows.

The MDCs and BANs are movable but normally an MDC placed in a room moves only within that room. BANs can move freely anywhere. It is assumed that the MDC of one room has a connection with the MDC of the next room. The patient node BAN_2 is considered as a movable BAN coordinator (BANC). As a fast walking patient, the speed of movable BANC is set to 1 meter per second. The node BAN_2 moves vertically as shown by the green arrows in Figure 6. The source node BAN_2 displays its data to MDC_2.

6.2. Parameters Used for Simulations. The transmit power used in simulations is −25 dBm. The transmission range of −25 dBm is about 3 meters which is the recommended value for BAN communication [2] in hospital environment. The network parameters used in our simulations are shown in Table 3.

6.3. Performance Results. The source nodes send a total of 95 K data packets in the 49-node hospital environment. The above mentioned parameters are calculated after the transmission of every 9.5 K packet of all types sent by the source nodes. All types of data packets including OPs, DSPs, and RSPs are sent from source nodes. To achieve a 97% confidence interval for the illustrative results, three runs are simulated in every experiment which may introduce a maximum error of 3×10^{-3}, based on the error calculation done by Castalia simulator [27]. The below two cases are considered for the same scenario shown in Figure 6.

Case 1. A fixed number of DSPs and RSPs but a variable number of OPs are sent from the source nodes. The number of DSPs is 1.2 K when 9.5 K packets are sent by source nodes. After that 7 K DSPs are consistently included in the offered traffic loads by source nodes. The 7 K RSPs are included consistently in all offered traffic loads. The OPs are continuously increased from 1 K to 81 K as with the increase of offered traffic load from 9.5 K to 95 K, respectively. The types of data packets included in the offered traffic load are shown in Figure 7(a).

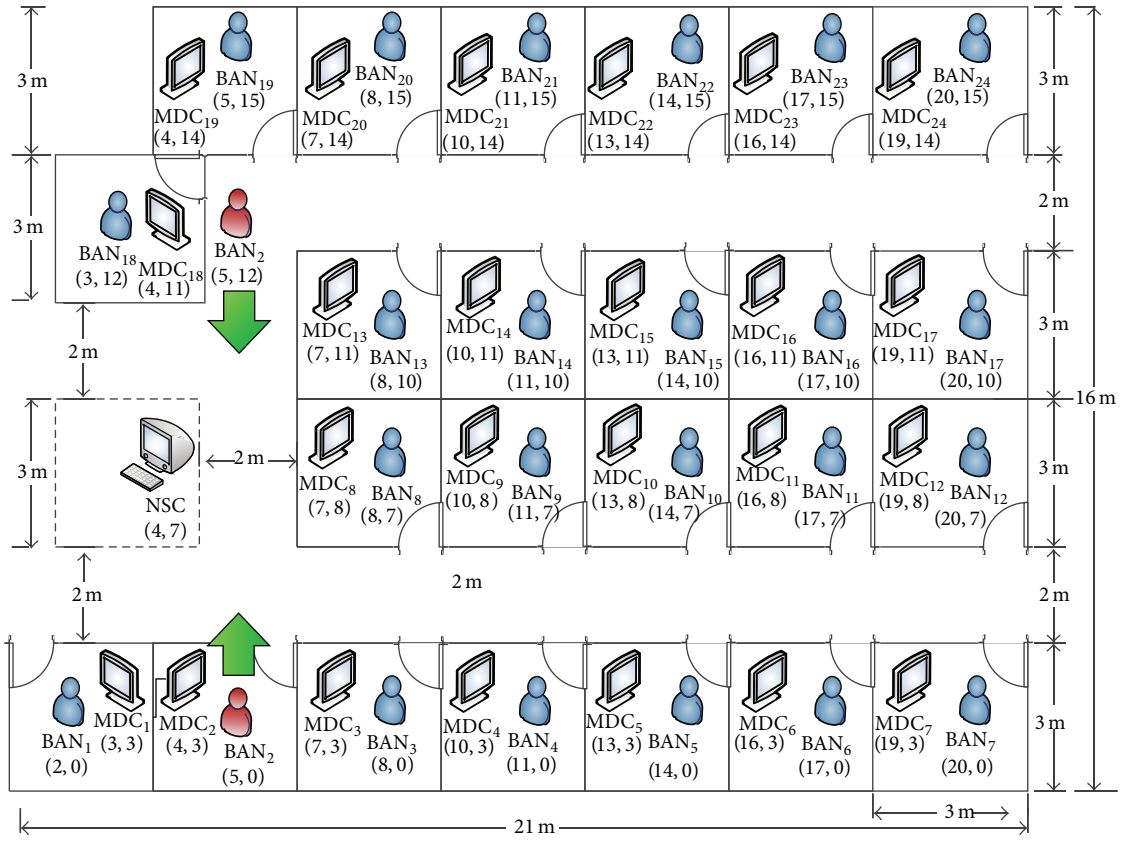


FIGURE 6: Node deployment for 24 patient beds in hospital environment.

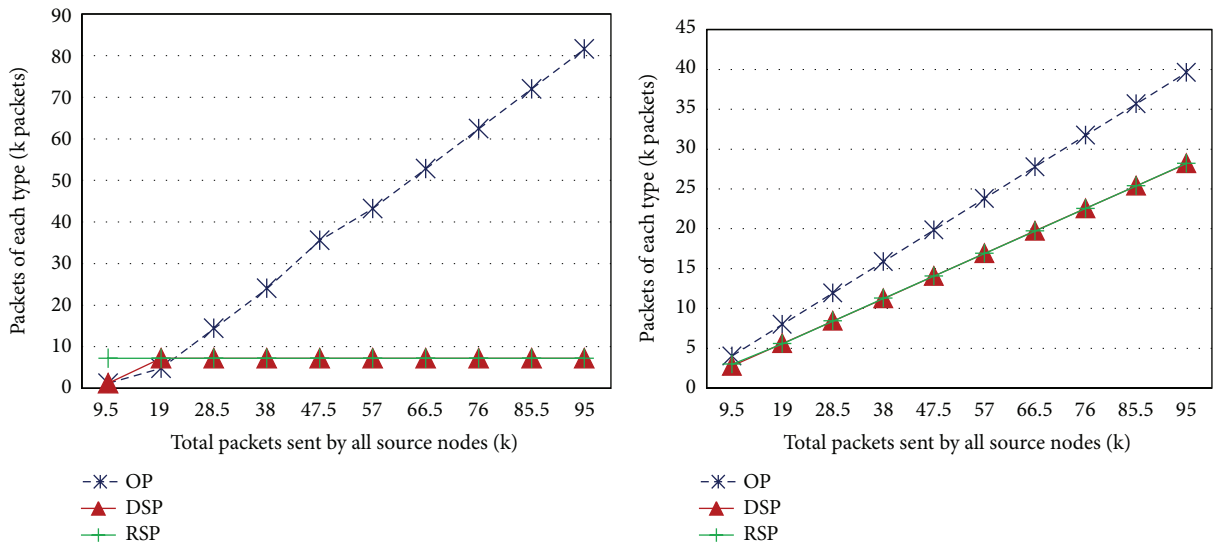
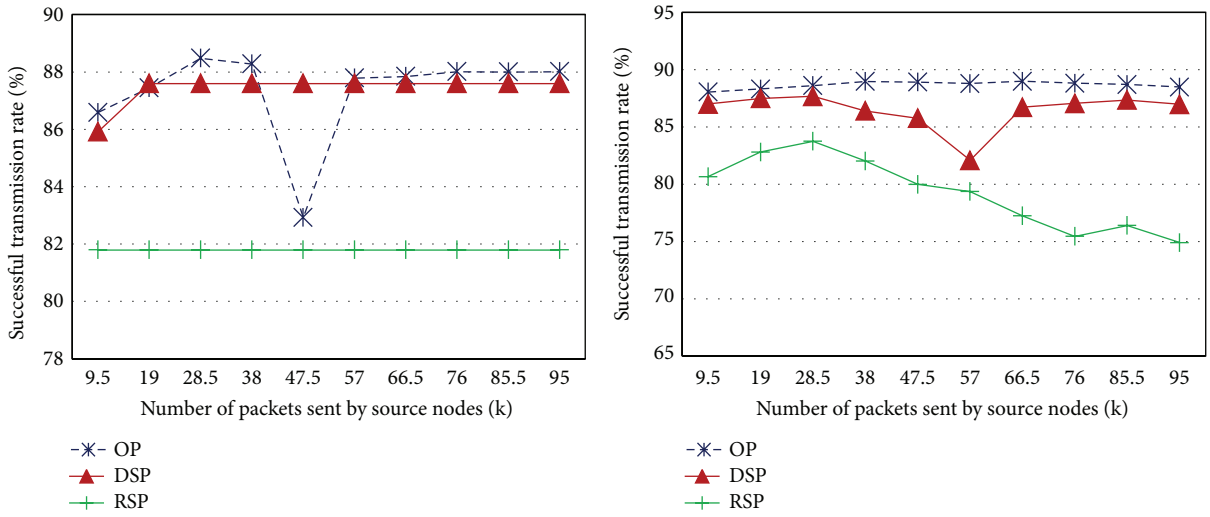


FIGURE 7: Offered traffic by source nodes.

Case 2. A variable number of OPs, DSPs, and RSPs are sent with the ratio of 40%, 30%, and 30%, respectively. The OPs constitute from 4 K to 39.5 K packets as the offered traffic load is increased from 9.5 K to 95 K. Similarly, DSPs and RSPs packets constitute from 2.8 K to 28 K packets of each type, when the total offered traffic load by source nodes is increased from 9.5 K to 95 K packets. Figure 7(b) shows the types of packets included in the offered traffic load for Case 2.

The throughput, packets forwarded by intermediate nodes, network traffic, packets dropped at the network layer,



(a) Case 1 (b) Case 2

Figure 8: Throughput versus offered traffic.

packets dropped on MAC layer, and energy consumption are measured. The performance results of each parameter are discussed below.

6.3.1. Throughput. The throughput is measured by calculating the number of packets received successfully at the destination nodes. The successful transmission rate or throughput is measured after the transmission of every 9.5 K packet sent by the source. For Case 1, Figure 8(a) shows that ZEQoS provides a consistent reliability which is in excess of 82%, 85%, and 81% for OPs, DSPs, and RSPs, respectively. For Case 2, as shown in Figure 8(b), the successful transmission rate of OPs, DSPs, and RSPs is in excess of 88%, 86%, and 75%, respectively.

The results from Figure 8 show that the mechanism of ZEQoS handles all the data types (i.e., OPs, DSPs, and RSPs) successfully with higher throughput. ZEQoS overcomes the issues of traffic congestion by using the end-to-end path delays and reliabilities for DSPs and RSPs, respectively. Also the transmission of RSPs over redundant paths ensures the higher reliability of RSPs packets.

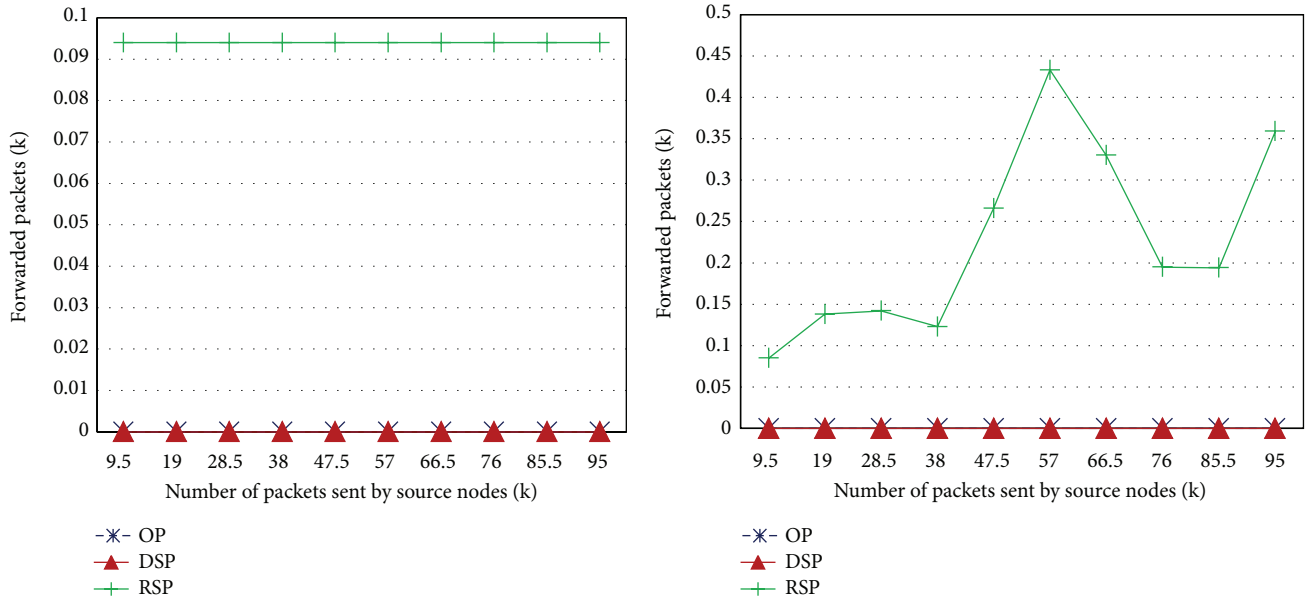
The path selection mechanism of ZEQoS considers the geographic location, energy availability, end-to-end path delays, and end-to-end path reliabilities for all nodes in the network which helps improve the overall throughput for all the data types.

6.3.2. Packets Forwarded by Intermediate Nodes. The approach used in ZEQoS for the selection of the most appropriate next hop is very effective. In the proposed ZEQoS scheme, a BAN coordinator does not send data to other BAN coordinators unless it is necessary. The BAN coordinator in the proposed ZEQoS sends data to another BAN coordinator only if it is necessary. The BAN coordinators send the data packets directly to the destinations. In noRouting, the delay-sensitive data packets are forwarded to random next hop devices instead of algorithm's next hop based on end-to-end path delay routes. Figure 9 shows the number of OPs, DSPs, and RSPs forwarded by the intermediate nodes.

It is seen from Figure 9 that no OPs or DSPs data packets are forwarded by any intermediate nodes. In Case 1, the number of RSPs forwarded by intermediate nodes is only 94 which is negligible when compared to the overall network traffic. In Case 2, from Figure 9(b) it is shown that the intermediate nodes forwarded 85 to 433 RSPs when offered traffic is increased from 9.5 K to 95 K. The control of Hello packets broadcast also helps reduce the packets forwarded by intermediate nodes.

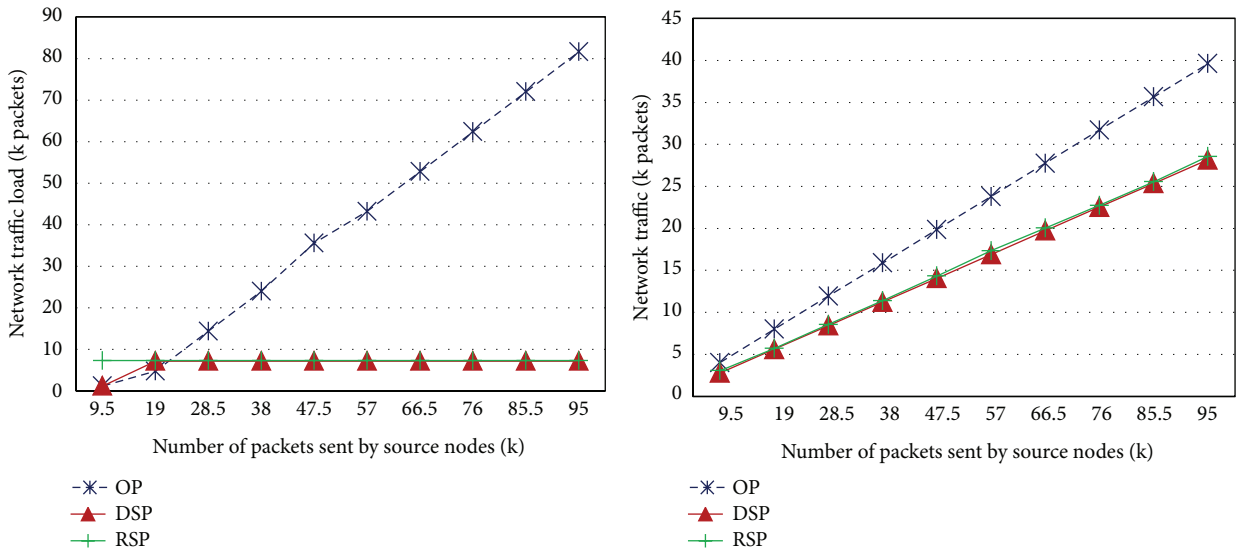
6.3.3. Overall Network Traffic. The lower number of forwarded packets as discussed in the previous section helps reduce the overall network traffic. The Hello packets are not added in this network traffic. In Case 1, Figure 10(a) shows that the overall network traffic due to OPs, DSPs, and RSPs are almost 7 K, 7 K, and 1 K to 81 K, respectively. The numbers of Hello packets are 179 K to 2198 K when 9.5 K to 95 K offered traffic load is applied from the source nodes, respectively. In Case 2, the overall network traffic due to OPs, DSPs, and RSPs is almost 4 K to 39 K, 2.5 K to 28 K, and 3 K to 28.5 K, respectively, as shown in Figure 10(b). In addition to data packets, 182 K to 2171.5 K Hello packets are also part of overall network traffic when 9.5 K to 95.5 K packets are sent by source nodes, respectively.

6.3.4. Packets Dropped at the Network Layer. In previous protocols like DMQoS [17], the source nodes calculate the hop-by-hop delay and reliability of the next hop nodes for the DSPs and RSPs, respectively, and send the data to the best next hop which has lowest delay for DSPs and highest reliability for RSPs. The next hop then calculates the delays or reliabilities of its upstream nodes. The packets are dropped in case of not meeting the requested delay or reliability by all neighboring upstream nodes. ZEQoS resolves this problem by using the



(a) Case 1 (b) Case 2

Figure 9: Packets forwarded by intermediate nodes.

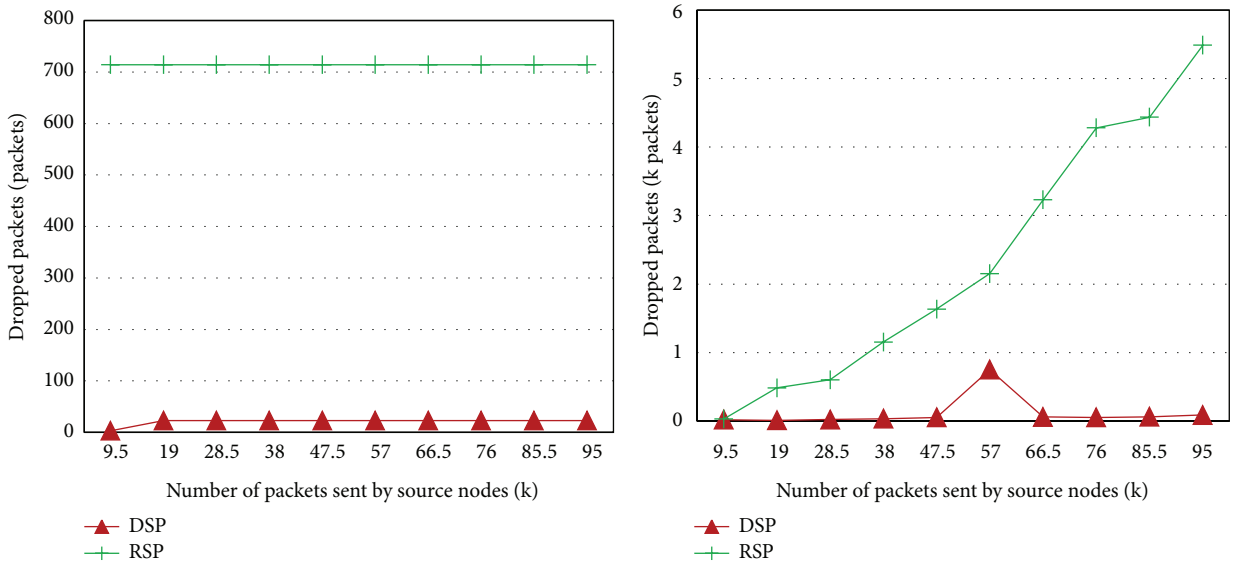


(a) Case 1 (b) Case 2

Figure 10: Overall network traffic versus offered load.

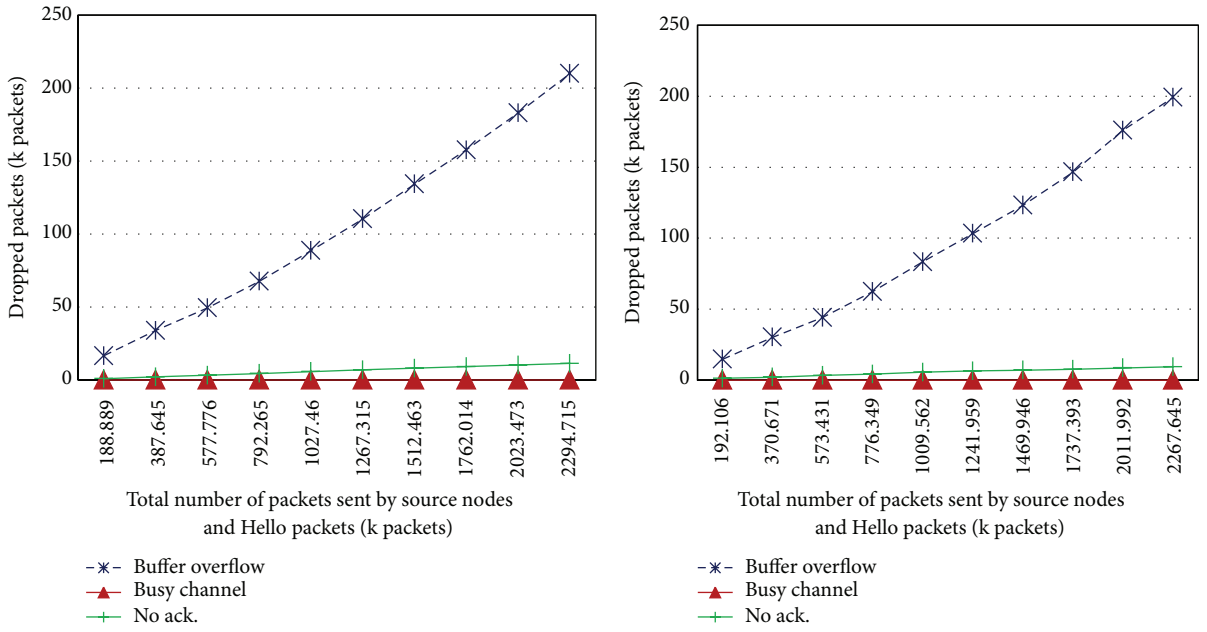
end-to-end path delays and reliabilities for DSPs and RSPs, respectively. Also the use of three redundant paths for RSPs in ZEQoS ensures better transmission rate. In Case 1, ZEQoS drops 23 DSPs and 714 RSPs data packets for all the traffic loads as shown in Figure 11(a). In Case 2, Figure 11(b) shows that the DSPs and RSPs dropped at the network layer due to not meeting the requested reliability and delay requirements are an average of 0.2% and 4.4%, respectively.

6.3.5. Packets Dropped by the MAC Layer. The total number of packets dropped by the MAC layer due to buffer overflow, busy channel, and no acknowledgements is measured. Figure 12 shows the packets dropped by MAC layer for Cases 1 and 2, respectively. The total offered traffic including Hello packets is 188 K to 2294 K and 192 K to 2267 K for Cases 1 and 2, respectively. No data packets are dropped due to busy channel in both cases. Also the packets dropped due to no acknowledgments increases from 1 K to 11 K in both cases. It is seen from Figure 12 that packets dropped due to the MAC buffer overflow are very high. In Case 1, the packets dropped due to the buffer overflow are 16 K to 209 K, whereas, in Case 2, the average packets dropped due to buffer overflow is 8.4%.



(a) Case 1 (b) Case 2

Figure 11: Packets dropped at the network layer due to lower delay or reliability requirements.



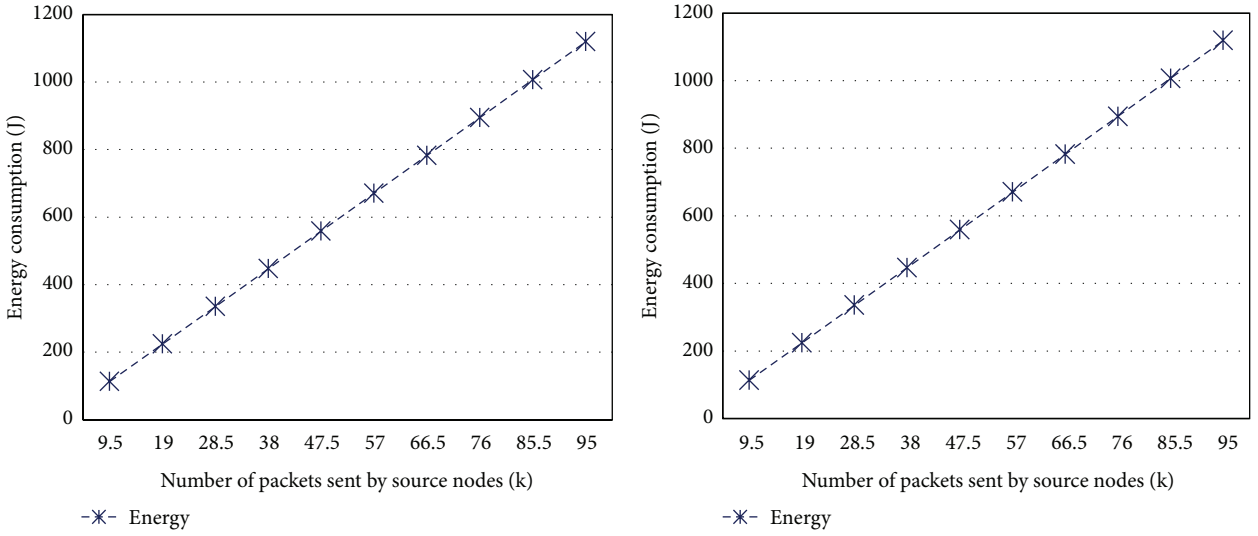
(a) Case 1 (b) Case 2

Figure 12: Packets dropped by the MAC layers.

6.3.6. Overall Energy Consumption. The overall energy consumption in both cases for ZEQoS is discussed in this section. It shows that ZEQoS provides a consistent and more reliable delivery of all three types of data packets (OPs, DSPs, and RSPs) as previously discussed in Section 6.3.1. The energy consumptions of both cases are similar as shown in Figure 13. The figure shows that ZEQoS consumes 112 to 118 Joules of energy when the offered load is 9.5 K to 95 K data packets as sent by source nodes. The drawback of ZEQoS is to consume much higher energy as compared to the energy consumption of the protocols (EPR, QPRD, and QPRR) which are not handling all three data types OPs, DSPs, and RSPs at a time.

7. Performance Comparison with DMQoS and NoRouting

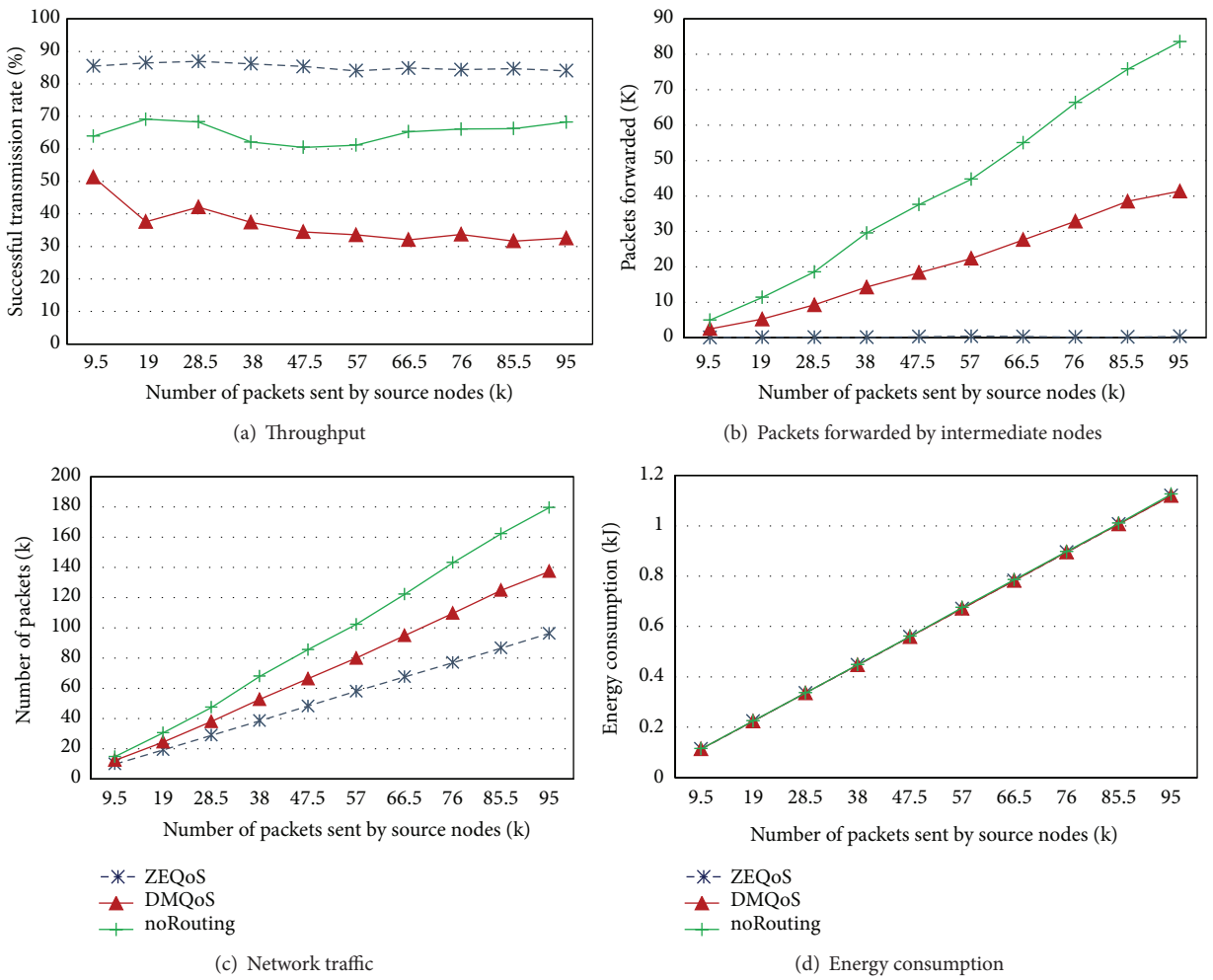
In this section, the performance of ZEQoS is compared with the DMQoS routing protocol [17] and noRouting. Norouting mechanism is used in the noRouting is case. The packets are



(a) Case 1

(b) Case 2

Figure 13: Overall energy consumption.



(a) Throughput

(b) Packets forwarded by intermediate nodes

(c) Network traffic

(d) Energy consumption

Figure 14: Performance comparison for different parameters.

forwarded to random next hop devices instead of following any algorithm's next hop. The comparison with noRouting is used to verify whether forwarding the packets to a random next hop device results in a better successful transmission rate than the ZEQoS routing which is based on energy and QoS-aware algorithm. The experimental results, shown in Figure 14, prove that the approach used by ZEQoS is more effective. The node deployment used for this test is similar to Case 2 of Section 6.3. The network parameters used in our simulations are shown in Table 2. The offered traffic load generated from nodes is 40%, 30%, and 30% of OPs, DSPs, and RSPs, respectively. The total 95 K packets are sent from the source nodes and the results are noted after every 9.5 K packet.

Figure 14(a) shows that ZEQoS provides a consistent 84% throughput; however, it is seen that the reliabilities of DMQoS and noRouting are on average 36% and 65%, respectively. Figure 14(b) shows that the packets forwarded by the intermediate nodes in ZEQoS are almost negligible; whereas, DMQoS and noRouting forward 21 K and 42 K packets, respectively. The hop-by-hop mechanism used in DMQoS causes the increased forwarded packets. The network traffic is increased when more packets are forwarded by intermediate nodes as shown in Figure 14(c). The increased network traffic causes the traffic congestion and more packets are dropped on MAC and network layers as explained in Sections 6.3.4 and 6.3.5. It is seen from Figure 14(d) that the energy consumption for all three protocols is the same for all the traffic loads.

8. Conclusion

A new modular energy and QoS-aware routing protocol (ZEQoS) for hospital BAN communication is proposed in this paper. The modules of new protocol are divided into two main types: MAC layer modules and network layer modules. MAC layer modules include the MAC receiver, the reliability module, the delay module, and the MAC transmitter. The packet classifier, the Hello protocol module, the routing services module, and the QoS-aware queuing module are included in network layer modules.

The proposed ZEQoS routing protocol provides a mechanism with the help of neighbor table constructor algorithm, routing table constructor algorithm, and path selector algorithm to calculate the communication costs, end-to-end path delays, and end-to-end path reliabilities of all possible paths from a source to destination and then decides on the best possible path(s) with the consideration of QoS requirement of the OPs, RSPs, and DSPs.

OMNeT++ based simulator Castalia 3.2 [26] was used to test the performance of the proposed protocol. The simulations were performed by considering a real hospital scenario when a source node was movable. All three types of data packets OPs, RSPs, and DSPs were sent from the source nodes. Both fixed and variable numbers of OPs, DSPs, and RSPs were considered. The simulation results showed that the ZEQoS had in excess of 81% and 75% throughput for all classes of packets in fixed and variable cases, respectively, when offered traffic load of 9.5 K to 95 K packets was used. The simulation results showed that the ZEQoS had superior performance in excess of 84% throughput when compared with DMQoS and noRouting provides 36% and 65%, respectively.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

References

[1] P. J. Riu and K. R. Foster, "Heating of tissue by near-field exposure to a dipole: a model analysis," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 8, pp. 911–917, 1999.

[2] "IEEE 802.15 WPAN Task Group 6 (TG6) Body Area Networks," IEEE 802.15, November 2007, http://www.ieee802.org/15/pub/TG6.html.

[3] S. Ullah, H. Higgins, B. Braem et al., "A comprehensive survey of wireless body area networks on PHY, MAC, and network layers solutions," *Journal of Medical Systems*, vol. 36, no. 3, pp. 1065–1094, 2012.

[4] Z. A. Khan, *Advanced Zonal Rectangular LEACH (AZR-LEACH): an energy efficient routing protocol for wireless sensor networks [Ph.D. thesis]*, Dalhousie University, Halifax, Canada, 2012.

[5] Z. Khan, S. Sivakumar, W. Phillips, and B. Robertson, "QPRD: QoS-aware peering routing protocol for delay sensitive data in hospital body area network communication," in *Proceedings of the 7th International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA '12)*, pp. 178–185, Victoria, Canada, November 2012.

[6] Z. Khan, S. Sivakumar, W. Phillips, and B. Robertson, "A QoS-aware routing protocol for reliability sensitive data in hospital body area networks," *Procedia Computer Science*, vol. 19, pp. 171–179, 2013.

[7] T. Lu and J. Zhu, "Genetic algorithm for energy-efficient QoS multicast routing," *IEEE Communications Letters*, vol. 17, no. 1, pp. 31–34, 2013.

[8] D. Djenouri and I. Balasingham, "Traffic-differentiation-based modular QoS localized routing for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 6, pp. 797–809, 2011.

[9] X. Liang and I. Balasingham, "A QoS-aware routing service framework for biomedical sensor networks," in *Proceedings of the 4th IEEE International Symposium on Wireless Communication Systems (ISWCS '07)*, pp. 342–345, Trondheim, Norway, October 2007.

[10] K. Zeng, K. Ren, W. Lou, and P. J. Moran, "Energy aware efficient geographic routing in lossy wireless sensor networks with environmental energy supply," *Wireless Networks*, vol. 15, no. 1, pp. 39–51, 2009.

[11] X. Liang, I. Balasingham, and S.-S. Byun, "A reinforcement learning based routing protocol with QoS support for biomedical sensor networks," in *Proceedings of the 1st International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL '08)*, Aalborg, Denmark, October 2008.

[12] M. A. Razzaque, M. M. Alam, M. Mamun-Or-Rashid, and C. S. Hong, "Multi-constrained QoS geographic routing for heterogeneous traffic in sensor networks," *IEICE Transactions on Communications*, vol. E91-B, no. 8, pp. 2589–2601, 2008.

[13] D. Djenouri and I. Balasingham, "New QoS and geographical routing in wireless biomedical sensor networks," in *Proceedings of the 6th International Conference on Broadband Communications, Networks and Systems (BROADNETS '09)*, Madrid, Spain, September 2009.

[14] S. Wu and K. S. Candan, "Power-aware single- and multipath geographic routing in sensor networks," *Ad Hoc Networks*, vol. 5, no. 7, pp. 974–997, 2007.

[15] E. Felemban, C.-G. 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. 738–753, 2006.

[16] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, "A spatiotemporal communication protocol for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 16, no. 10, pp. 995–1006, 2005.

[17] A. Razzaque, C. S. Hong, and S. Lee, "Data-centric multiobjective QoS-aware routing protocol for body sensor networks," *Sensors*, vol. 11, no. 1, pp. 917–937, 2011.

[18] M. Chen, T. Kwon, and Y. Choi, "Energy-efficient differentiated directed diffusion (EDDD) in wireless sensor networks," *Computer Communications*, vol. 29, no. 2, pp. 231–245, 2006.

[19] M. Chen, V. C. M. Leung, S. Mao, and Y. Yuan, "Directional geographical routing for real-time video communications in wireless sensor networks," *Computer Communications*, vol. 30, no. 17, pp. 3368–3383, 2007.

[20] X. Huang and Y. Fang, "Multiconstrained QoS multipath routing in wireless sensor networks," *Wireless Networks*, vol. 14, no. 4, pp. 465–478, 2008.

[21] M. Chen, T. Kwon, S. Mao, Y. Yuan, and V. Leung, "Reliable and energy-efficient routing protocol in dense wireless sensor networks," *International Journal on Sensor Networks*, vol. 4, no. 1-2, pp. 104–117, 2008.

[22] M. Chen, V. C. M. Leung, S. Mao, Y. Xiao, and I. Chlamtac, "Hybrid geographic routing for flexible energydelay tradeoff," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 4976–4988, 2009.

[23] Z. Khan, S. Sivakumar, W. Phillips, and N. Aslam, "A new patient monitoring framework and Energy-aware Peering Routing Protocol (EPR) for Body Area Network communication," *Journal of Ambient Intelligence and Humanized Computing*, 2013.

[24] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V. C. M. Leung, "Body area networks: a survey," *Mobile Networks and Applications*, vol. 16, no. 2, pp. 171–193, 2011.

[25] J. Xu, W. Liu, F. Lang, Y. Zhang, and C. Wang, "Distance measurement model based on RSSI in WSN," *Wireless Sensor Network*, no. 2, pp. 606–611, 2010.

[26] NICTA, "Castalia," National ICT Australia, March 2011, https://castalia.forge.nicta.com.au/index.php/en/.

[27] NICTA, "Castalia, Wireless Sensor Network Simulator," May 2014, http://castalia.research.nicta.com.au/index.php/en/.