

A Cross-Layer Channel Access and Routing Protocol for Medical-Grade QoS Support in Wireless Sensor Networks

Young-Duk Kim · Kook-Rae Cho · Hui-Sup Cho · Dongkyun Kim

Published online: 7 November 2013
© Springer Science+Business Media New York 2013

Abstract One of principal design issues of a Wireless Sensor Network (WSN) for medical information systems is to classify received packets based on their priorities and guarantees so that they can be transmitted reliably, thus satisfying QoS requirements. In addition, when the target WSN requires multi-hop communications and the traffic load increases significantly, it is challenging to support both load balancing and suitable QoS at the same time. In this paper, we propose a new reliable protocol termed Cross-layer Channel Access and Routing (CCAR), which simultaneously supports both MAC and routing operations for medical-grade QoS provisions. CCAR initially determines the routing path with the lowest traffic load and low latency using newly defined channel quality factors. Concurrently, the source node allocates the predefined QoS Access Category to each packet and reserves the channel along the route. In addition, CCAR introduces an effective route maintenance scheme to avoid link failures in bottlenecked intermediate nodes, which prevents unnecessary packet drops and route rediscovery evocations. Finally, through both simulation studies and real test-bed experiments, we evaluate the performance of CCAR by comparing it with other conventional protocols, demonstrating that the proposed protocol can more efficiently support medical-grade QoS packets, especially when the network is heavily loaded.

Keywords Wireless Sensor Networks · Medical-grade QoS · Wireless routing protocol · Wireless MAC protocol

Y.-D. Kim · K.-R. Cho · H.-S. Cho
Daegu Gyeongbuk Institute of Science and Technology (DGIST), Deagu, Republic of Korea
e-mail: ydkim@dgist.ac.kr

D. Kim (✉)
School of Computer Science and Engineering, Kyungpook National University,
Deagu, Republic of Korea
e-mail: dongkyun@knu.ac.kr

1 Introduction

The Wireless Sensor Network (WSN) [36] has become one of the most popular forms of technology for monitoring widely scattered sensors for various purposes. These networks work in many areas such as in medical, military, agriculture and industrial systems. Among these applications, the development of a medical data monitoring system has been the subject of much research interest in that such a system can provide medical services anytime and anywhere from a remote site. However, when a WSN for monitoring medical data is deployed, a number of open problems remain. The most common issues are discussed.

First, in general, a medical monitoring system requires real-time data processing and transmission, as several types of monitoring data are critical to the patient's life and because such data should be delivered to the doctor as soon as possible. However, most existing legacy protocols (e.g., IEEE 802.15.4 [10], ZigBee [3]) for WSN treat all packets in the same way. Thus, Quality of Service (QoS) enabled protocols are necessary to handle the emergent medical-grade applications.

Second, for QoS support, the MAC protocol should be able to recognize medical-grade packets and transmit them over wireless media according to their QoS levels. Usually, medical-grade packets are handled with more reliable and fast channel-access mechanisms than other types of data packets. Although the IEEE 802.15.4 protocol provides a Guaranteed Time Slot (GTS) period for contention-free accesses, it requires strict time synchronization with the coordination node, which is not only a burden on the sink node for managing the beacon frame but also complicates the extension of the network topology.

Third, besides MAC layer operations, the design of an efficient routing protocol is also a significant issue for supporting QoS packets in multi-hop-based WSNs [18]. Specifically, wireless sensor nodes, which are characterized by limited bandwidth, low buffer capacity, inconstant transmission delay and unbalanced traffic loads, pose additional challenges for establishing an optimal QoS-aware routing route. Moreover, due to the increased requirements for real-time data traffic, network congestion can easily occur. In such situations, the sensor nodes experience significantly long queuing delays, severe packet collisions and link failures, all of which prevent WSNs from supporting QoS traffic. Furthermore, because the conventional routing protocols such as DSR [16] and AODV [24] only try to find the shortest route based on the hop distance without considering traffic concentrations, performance degradation is inevitable. Thus, the routing protocol for such environments should consider not only differentiated packet management schemes but also a dynamic load balancing approach while establishing a route.

Finally, after the target medical data is gathered from various health-related sensors, the medical monitoring system with a WSN should reliably record and store all of the measured information in order to establish a personal healthcare database which is provided to the doctor for further investigation and analysis. For example, the design of message-conversion schemes such as Health Level 7 (HL7) [12] and IEEE/ISO 11073 [31] is one of most challenging issues for a WSN-based medical monitoring system. In addition to the aforementioned open problems, several technical challenges also exist. Examples include protocol design in cross-layer optimization, topology management, and methods that reduce energy use.

In this paper, in an effort to resolve these problems, we first propose a new channel quality metric which considers the transmission delay and buffer capacity of each node with a predefined Access Category (AC) to support medical-grade QoS traffic. We then propose an efficient cross-layer protocol termed Cross-layer Channel Access and Routing (CCAR) which not only provides an optimal routing path with the lowest traffic load and low transmission delay but also a differentiated channel access mechanism according to the

channel quality metric and packet priority levels. In addition, by adopting the HL7 message exchange standard, we show that the proposed protocol is fully compatible with other medical monitoring systems.

The rest of this paper is organized as follows. In Sect. 2, we review several related works for QoS provisioning in WSNs. Section 3 describes the proposed protocol in detail and Sect. 4 presents a performance evaluation of the proposed protocol. Finally, Sect. 5 concludes this paper with a brief summary.

2 Related Works

2.1 MAC Protocols for QoS Provisions

Given that legacy protocols such as SMAC [35], BMAC [25], TMAC [7] and IEEE 802.15.4 do not deal with QoS frames, there are several research results that support and improve QoS provisioning for WSNs at the MAC layer. The Optimized Priority Assignment Mechanism (OPAM) [26] is a type of MAC layer approach for medical-grade QoS support that dynamically schedules different types of packets by monitoring individual queue delays without invoking upper layer congestion controls. To do this, OPAM introduces a packet classifier to assign data levels, and the resulting packet priorities are continuously monitored. In other work [28], the authors proposed another QoS MAC protocol which satisfies application-specific requirements by providing a dynamic adjustment of the Contention Window (CW). In addition, it tries to save energy through the use of a differentiated duty cycle and an active time mechanism. However, this protocol does not suggest specific access categories or priority levels for medical-grade QoS support; moreover, it also neglects to consider the traffic load when determining node priority levels. PRIMA [1] also suggested a priority queue-based MAC protocol as well as energy saving mechanisms by combining the CSMA and TDMA modes. In addition, PRIMA minimizes unnecessary idle listening periods by allowing a node with no data to transmit to sleep promptly, thus also reducing the energy consumption in WSNs. Hybrid Medium Access Control (HMAC) [33] is another combined protocol for CSMA and TDMA. HMAC proposes a channel-reservation scheme for delay-sensitive traffic by assigning it as high-priority traffic for access to the channel and by providing rapid transmissions using a short frame structure. However, these studies [1, 33] strictly require precise time synchronization between nodes and the clustered network topology. Furthermore, although these previous works try to offer QoS support using various network parameters, they do not simultaneously consider a routing algorithm for congestion avoidance and load balancing, which consequently does not guarantee performance enhancements in multi-hop-based WSNs.

2.2 Routing Protocols for Load Balancing and QoS Provisions

Besides conventional on-demand protocols such as AODV and DSR, there are several state-of-the-art routing protocols for QoS support and load balancing in multi-hop wireless networks. First, in order to avoid congested routes, Dynamic Load-Aware Routing (DLAR) [19] is a simple and efficient load balancing protocol which finds routes with minimal traffic loads by defining the traffic loads using the number of buffered packets in the nodes. However, this protocol only addresses buffered packets and neglects to consider other channel link parameters and packet differentiation schemes. Congestion-Aware Routing (CARM) [4] is another load-balancing routing protocol which additionally considers the data rate, the MAC overhead and the link delay to improve channel utilization and throughput. However,

it focuses on the Mobile Ad-hoc Networks (MANET) environment and does not address any QoS-related operations. AQOR [34] is one of state-of-the-art protocols with QoS support and it introduced a computation method of available bandwidth and end-to-end delay for per flow QoS requirement. Similarly, other existing routing protocols [14, 20, 32] deal with QoS services by considering various routing metrics such as data rate, battery consumption, delay jitter and etc. However, they are mainly compatible with Wireless Local Area Network (WLAN) based MAC protocols such as IEEE 802.11 DCF and IEEE 802.11e, which is not acceptable for resource limited MAC protocol over WSNs. Furthermore, they do not simultaneously consider MAC layer channel operations and medical QoS requirements. Energy Efficient QoS based Multipath Routing (EQSR) [2] is proposed to provide QoS aware multipath routing by using residual energy, buffer size and SNR to predict optimal paths. Although EQSR achieves load balancing through distributing the traffic across multiple path, it potentially has a channel interference problem caused by adjacent nodes. Moreover, it also has the burden of state maintenance for periodically updating the routing information. Localized Multi-Objective Routing (LOCALMOR) [9] is a localized QoS routing protocol for WSNs that provides packet-level differentiated management while considering latency, reliability and residual energy in nodes. MMSPEED [11] is another QoS provisioning protocol which guarantees end-to-end delay and reliability requirements based on location awareness. However, both LOCALMOR and MMSPEED are only suitable for nodes with location-assisted devices such as a Global Positioning System (GPS), which may be an additional overhead factor for tiny sensor platforms. Other researchers [30] proposed architecture for a cross-layer QoS framework over wireless multimedia sensor networks which maximizes the number of video stream sources using cross-layer signaling for the application, routing and MAC layers. However, this proposal is also conceptually based on the IEEE 802.11 standard and therefore cannot therefore be applied directly to IEEE 802.15.4-based wireless sensor nodes.

2.3 Approaches for Wireless Medical Systems

Meanwhile, numerous studies have been conducted pertaining to the implementation and development of wireless medical monitoring systems and solutions [5, 6, 13, 17, 23]. Kang et al. [17] proposed a medical-grade wireless architecture for electrocardiogram (EEG) data transmission by providing collision-free MAC operations and error control functions. Chen et al. [5] developed a ZigBee-based patient-monitoring system which gives reliable transmissions with the anycast routing protocol. This system also uses a fall-detection algorithm for patient care, with events transmitted to the closest router node. Nathalie et al. [8] proposed a distributed diagnosis scheme based on the medical alert which monitors abnormal parameters of each sensor mote. In other work [6], a PDA-based wireless medical information system was introduced which supports HL7 messages between the sender and receiver and establishes a central web server for maintaining patient database systems. Morak et al. [23] developed a mobile-phone-based medical monitoring system which uses a combination of Bluetooth and Near-Field Communication (NFC) for interconnecting between mobile phones and medical sensor nodes. Hu et al. [13] considered the congestion and packet loss problems in wireless healthcare systems. They proposed a wavelet-based signal compression scheme and a receiver-based loss prediction method for congestion reduction and loss compensation, respectively.

Overall, previous works tended to tackle design and implementation issues of actual medical monitoring systems. However, they still experience performance problems when data traffic exists in a multi-hop environment, and most do not simultaneously consider MAC and routing functions for QoS requirements. Thus, in this paper, we consider both

MAC and routing issues for medical-grade QoS and provide reliable communications even in congested situations.

3 Cross-Layer Channel Access and Routing Protocol

The main contribution of the proposed protocol, CCAR, is to satisfy medical-grade QoS requirements by allowing each sensor node to access a wireless channel dynamically and route the packets according to their priorities and network conditions. To do this, CCAR initially defines a new channel estimation metric called the QoS Factor (QF) in each node. $QF(i)$ at node i is calculated as follows:

$$QF(i) = \frac{D_{cur(t)}^i}{D_{avg(\theta,t)}^i} + \frac{B_{avg(\theta,t)}^i}{B_{cur(t)}^i} \tag{1}$$

Here, $D_{cur(t)}^i$ is the required packet delivery delay between the sender and receiver, which are mutually neighbors with a one-hop distance at current time t . $D_{avg(\theta,t)}^i$ is the average packet delivery delay between neighboring nodes during the predefined period θ , that is $[t-\theta, t]$. In addition, $D_{cur(t)}^i$ is expressly calculated by following expression:

$$D_{cur(t)}^i = D_{header}^i + D_{queue}^i + D_{backoff}^i + D_{trans}^i + D_{pro}^i \tag{2}$$

In this equation, D_{header}^i denotes the delays in node i for the packet header processing time (e.g., adding and removing the header), D_{queue}^i is the packet queuing delay, $D_{backoff}^i$ is the backoff delay for channel contention with neighbors, D_{trans}^i is the transmission delay affected by the actual data rate, and D_{pro}^i is the propagation delay over the physical medium. Among these measurements, because both the packet header processing time and the propagation time are generally invariable values, the required delivery delay is primarily dependant on other factors, such as the backoff time and the transmission time. Specifically, it is important to note that the backoff time includes the channel contention delay, which is mostly affected by neighboring nodes and their traffic loads. Thus, this backoff delay factor is considered to be a very important parameter for handling delay-sensitive QoS packets.

The other factors, $B_{cur(t)}^i$ and $B_{avg(\theta,t)}^i$ are the available buffer length of the interface at node i at current time t and the average available buffer length of node i during the period between $[t - \theta, t]$, respectively. The measurement of $B_{cur(t)}^i$ is calculated by Eq. (3).

$$B_{cur(t)}^i = B_{Max}^i - B_{occ(t)}^i \tag{3}$$

where B_{Max}^i is the maximum buffer length of node i and $B_{occ(t)}^i$ is the number of actually occupied packets in the interface buffer of node i at current time t . That is, the available buffer length denotes the actual remaining buffer length in the queue; when node i receives more packets than this level, numerous packets are dropped due to overflows, which is one of the substantial reasons for network congestion. Because congestion due to overflows leads to long end-to-end packet delays, buffer monitoring for each node is required to distribute the traffic load over the nodes with high buffer availability.

3.1 Routing Algorithm for QoS Support

When a source node using CCAR wishes to transmit data packets, it initially calculates $QF(s)$ and then starts to perform the route-discovery procedure based on its routing table

containing this factor. For route discovery, the node exchanges RREQ (Route Request) and RREP (Route Reply) control packets, which is an operation compatible with ordinary on-demanding protocols such as AODV and DSR. And the node also uses RERR (Route Error) packet during route maintenance procedure. The basic definitions of RREQ, RREP and RERR are as follow:

- RREQ—A route request packet for the route discovery procedure in case there is no route information available in the routing table. This packet is broadcasted to neighbor nodes until it finds a route to the destination.
- RREP—A route reply packet for RREQ in case a node has route information to destination. This packet is routed back to the origination of RREQ.
- RERR—A route error packet which is used to notify other nodes that a route failure has occurred. Finally, a node tries to find an alternative route for route maintenance purpose.

However, note that the proposed approach is different from conventional on-demanding routing protocols due to the fact that CCAR chooses the route with the lowest QF values from among all candidate routes instead of the shortest simple path with the lowest number of hops. This is expressed in the following equation:

$$Min_R \left[\sum_{i \in R} QF(i) \right] \tag{4}$$

Here, R is the set of the nodes composing the candidate route. The first process of route discovery is to search for previous route information in the existing routing table. If there are no matched fields in the table, the source node generates RREQ and broadcasts it to the neighbor nodes, after which each intermediate node only rebroadcasts newly received RREQ until the destination node receives it. In order to calculate QF values along the route, every intermediate node adds its own measured QF to the received instance from the previous node and the node then inserts this accumulated value $\sum_{i \in R} QF(i)$ into RREQ for rebroadcasting. In addition, a source node also inserts an Access Category (AC) value into RREQ to indicate explicitly what type of QoS-sensitive packets the source node wishes to transmit. This AC information enables the intermediate nodes to adjust their MAC parameters dynamically during the channel access time. The detailed parameter-tuning scheme at the MAC layer is described in section B.

In addition to proposal of accumulated $QF(i)$ and AC, CCAR newly defines a T_{EX} value which denotes the expiration period for ensuring prioritized channel access at the MAC layer. That is, after route establishment is completed, each node on the active route can reserve prioritized accesses to the wireless channel according to the predefined AC value during T_{EX} . In addition, T_{EX} is updated whenever a node receives another Route Reply (RREP) from the destination node or new data packets from the source node. If T_{EX} expires without transmitting any data packets, the node transmits packets via the best effort principle, which is the default channel access scheme. As a result, the final additional header information of RREQ includes three fields : { Accumulated $QF(i)$, AC and T_{EX} }, which is shown in Fig. 1.

When RREQs are broadcast, the destination node can receive multiple RREQs from different routes and chooses the route with the lowest cost according to equation (4). The

Source ID (with sequence #)	Dest. ID (with sequence #)	Route ID	$QF(i)$	AC	T_{EX}
--------------------------------	-------------------------------	----------	---------	----	----------

Fig. 1 RREQ packet structure

destination node then creates RREP and transmits it to the source node along the selected route, which is done via unicast. During the RREP traversal process, the intermediate nodes in the route and nodes overhearing RREP update their routing table with fresh information. Meanwhile, if the destination receives multiple RREQs with identical QF values, it simply selects a route with the minimum hop distance via the shortest path criteria.

During the route-discovery procedure, CCAR does not allow intermediate nodes to transmit their RREP packets using their own route caches because all RREQs have to be delivered to the destination node to investigate the QF values of the entire route. If the intermediate nodes transmit their RREPs, the route obtained from the route cache may be stale, especially when the topology is frequently changed. Moreover, the data rate of the intermediate node can also change due to the Signal-to-Interference-plus-Noise Ratio (SNIR). Consequently, by prohibiting intermediate nodes from sending their RREPs, CCAR can obtain fresh route information.

In addition to the route-discovery procedure, CCAR has a route maintenance mechanism that monitors active data flows. Although conventional protocols merely monitor link failures between the source and destination, CCAR additionally monitors $B_{avg(\theta,t)}^i$ values at each node whenever a data packet passes through it along the route, which enables CCAR to identify the degree of traffic concentration on nodes and prevent network congestion. To do this, CCAR newly defines B_{TH}^i , which denotes possible buffer threshold values at node i . If $B_{avg(\theta,t)}^i$ is smaller than B_{TH}^i , the current node is considered to be potentially congested and CCAR searches for an alternative node to create another route as a destination in its routing table, after which it resumes the packet relaying process. If an alternative cannot be found, CCAR generates a RERR packet and transmits it to the source node. Any node receiving (or overhearing) this RERR removes the corresponding route information in its routing table. The source node then restarts the route-discovery procedure if there is no alternative route information between all nodes on the legacy route. Meanwhile, the link-monitoring function for every node can result in a substantial energy loss. Thus, CCAR calculates $B_{avg(\theta,t)}^i$ only when there are packets for transmitting and receiving.

3.2 Channel Access for QoS Support

After the optimal route with the fewest QF values is determined with CCAR at the network layer, the source node starts to transmit data packets. Although this method provides faster and more reliable transmissions than other routes, it does not guarantee prioritized channel acquisition among other neighbors due to the fixed transmission probability at the MAC layer, which is a common phenomenon in CSMA/CA-based networks. To mitigate this problem, CCAR newly defines AC to reflect prioritized service levels, which is originally issued by the source node via RREQ and RREP. This implies that AC is initially assigned to each data packet which passes through the intermediate nodes of the established route, after which it reserves a wireless channel with differentiated levels of access probability at the MAC layer. Figure 2 shows four separate queues with respected to the different ACs in the node. In order to determine different ACs at the MAC layer, CCAR modifies the reserved bits of the frame control field in MAC header and inserts a QoS field as shown in Fig. 3. That is, when the application of the sensor node generates a data packet, it maps the packet to predefined ACs (e.g. HL7 application : AC3, Temperature/Humidity sensor: AC2) and then, configures the frame control fields to provide differentiated packet treatment.

Although there are several network parameters related to channel contention, such as CW, the Number of Backoffs (NB), and Clear Channel Assessment (CCA), CW tuning is one of the most common mechanisms for QoS provision. For example, IEEE 802.11e [15] was

Fig. 2 Access categories (ACs) for QoS support in CCAR

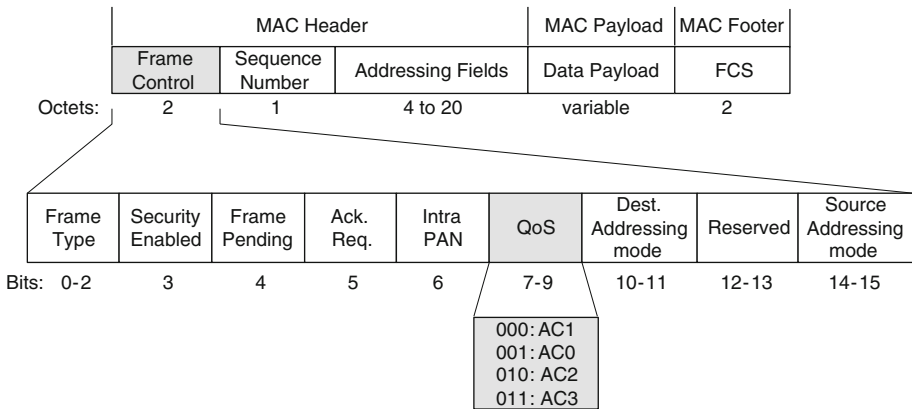
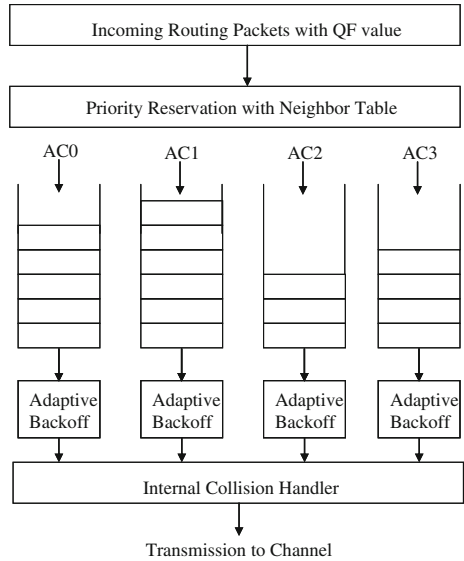


Fig. 3 QoS determination with AC bits in MAC header

introduced to support QoS in WLANs. It defines four separate queues based on different ACs in each node. Each queue has its own Arbitration Inter Frame Space (AIFS), CW_{min} , CW_{max} and Transmission Opportunity (TXOP) values for differentiated packet handling methods. In the case of IEEE 802.15.4, the backoff time of each node is determined by a random function $(2^{BE} - 1)$ and the default Minimum BE (MinBE) and Maximum BE (MaxBE) are 3 and 5, respectively. Thus, the backoff range is only between 7 and 31, which is not suitable to support dynamic QoS because it is not only a narrow range but also introduces fixed access probability. Thus, firstly, in order to provide a more dynamic backoff range for QoS support over IEEE 802.15.4, CCAR statically increases the total backoff range according to ACs: [3, 36] for AC3, [3, 24] for AC2 and AC1, [3, 12] for AC0. Secondly, after configuring initial backoff ranges, it adopts a CW-tuning mechanism by introducing a new dynamic backoff algorithm according to each AC. The detailed proposed backoff algorithm is described in Table 1. When a data packet arrives through the upper layer, it is entered to its proper internal

Table 1 Adaptive backoff for AC

Type	AC	Adaptive Backoff
Medical data	3	MinBE = 1 MaxBE = 5 If $(BE - QF > MinBE)$ then $Adaptive\ Backoff = 2^{BE-QF} - 1,$ Else $Adaptive\ Backoff = 2^{MinBE} - 1,$
Surveillance data	2	MinBE = 3 MaxBE = 5 $Adaptive\ Backoff = Default\ Backoff - QF $
Best effort data	1	Default Backoff
Yield data	0	MinBE = 3 MaxBE = 6 If $(BE+QF < MaxBE)$ then $Adaptive\ Backoff = 2^{BE+QF} - 1,$ Else $Adaptive\ Backoff = 2^{MaxBE} - 1,$

queue by the AC mapping process. The packet is then transmitted after calculating a new backoff delay.

AC3 denotes a medical data type, and it supports the fast channel access by subtracting QF from BE, leading to an exponential reduction of the backoff delay. In addition, MinBE is set to 1 to extend the reduction range of the backoff. Because a node with a higher QF implies that it experiences a higher transmission delay and lower buffer capacity, it can acquire the privilege to transmit data packets more promptly than other nodes by adopting a low backoff delay. Incidentally, this prioritized transmission opportunity enables the congested node to resolve its congestion status more quickly. AC2 supports surveillance data such as security and environmental monitoring data, which is less significant than medical-grade data but nonetheless requires faster transmission than other types of best-effort data. For example, a temperature sensor for fire detection and a smog sensor for pollution detection are two such surveillance purposes. In AC2, the new backoff delay is calculated by subtracting QF from the default backoff only if the result does not have a negative value. Because AC2 linearly reduces the default value, it is considered as a more moderated transmission than AC3. The general best-effort packets are assigned to AC1, which is equivalent to the default backoff delay of IEEE 802.15.4. Finally, AC0 denotes a yield data type which has a high backoff delay compared to a data packet with more significant QoS, such as AC3. Because severe channel contention due to the increase in the number of prioritized packets also leads to an increase in the number of packet collisions as well as a long transmission delay, nodes with low-priority data are requested to have fewer transmission opportunities by CCAR. The first yield operation is to set MinBE and MaxBE to 3 and 6, respectively, to increase the backoff delay. The process is completed by adding QF to the existing BE in an inverse operation to that of AC3.

In order to investigate the data priorities of neighboring nodes, each node maintains a neighbor AC table which consists of the following set: {Neighbor ID, Neighbor’s AC, and T_{EX} }. The table is updated with recent information when receiving or overhearing data

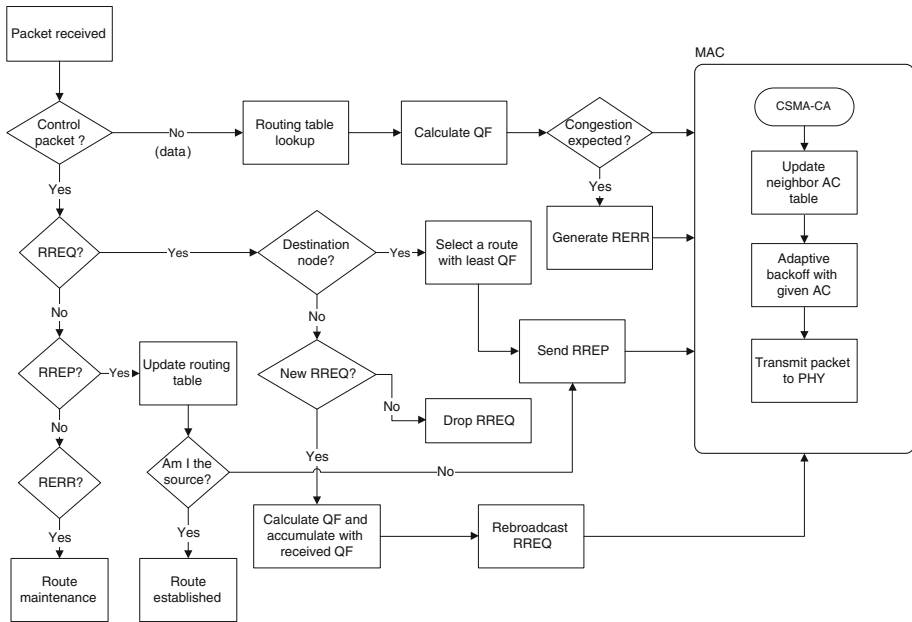


Fig. 4 Overall flow chart of the operations in CCAR

packets or RREPs containing selected route information. Each entry in the table is only maintained for the period of T_{EX} to remove stale information. Although trivial data packets can be the AC0 type, CCAR suggests that a node with AC1 immediately changes its data into AC0 if it detects any neighbors with AC3 packets in the table. Otherwise, if the node detects no neighbors with AC3 or if the timer is expired, it normally participates in channel contention after changing its data type to AC1. Overall flow charts of the operations and MAC operations for CCAR are shown in Figs. 4 and 5, respectively.

4 Performance Evaluation

To evaluate the performance of CCAR, we conducted simulations via NS-2 [22] and compared the results with AODV and DLAR, which are a conventional on-demand routing protocol and a load-balancing routing protocol, respectively. For the MAC protocol, the conventional IEEE 802.15.4 protocol is adopted. The simulation lasted 900 s. By default, the simulations were configured with 100 sensor nodes and a sink node randomly distributed over an $80\text{ m} \times 80\text{ m}$ rectangular topology. All nodes are capable of routing, and we assume that all nodes, except the sink node, move freely at the given speeds between 2–4 km/h which reflect the average walking speeds of patients in hospitals. The radio propagation range and the interference range for each node are set to 9 and 18 m, respectively. The nodes in the interference range can overhear the transmission. The data packet size is 50 bytes and the total number of data connections between the source and destination is set to 30 (AC3 connections: 5, AC2 connections: 10, AC1 connections: 15). We assume that all source nodes generate a User Datagram Protocol (UDP) with Constant Bit Rate (CBR) traffic because Transmission Control Protocol (TCP) may invoke its own congestion control, making it difficult to observe

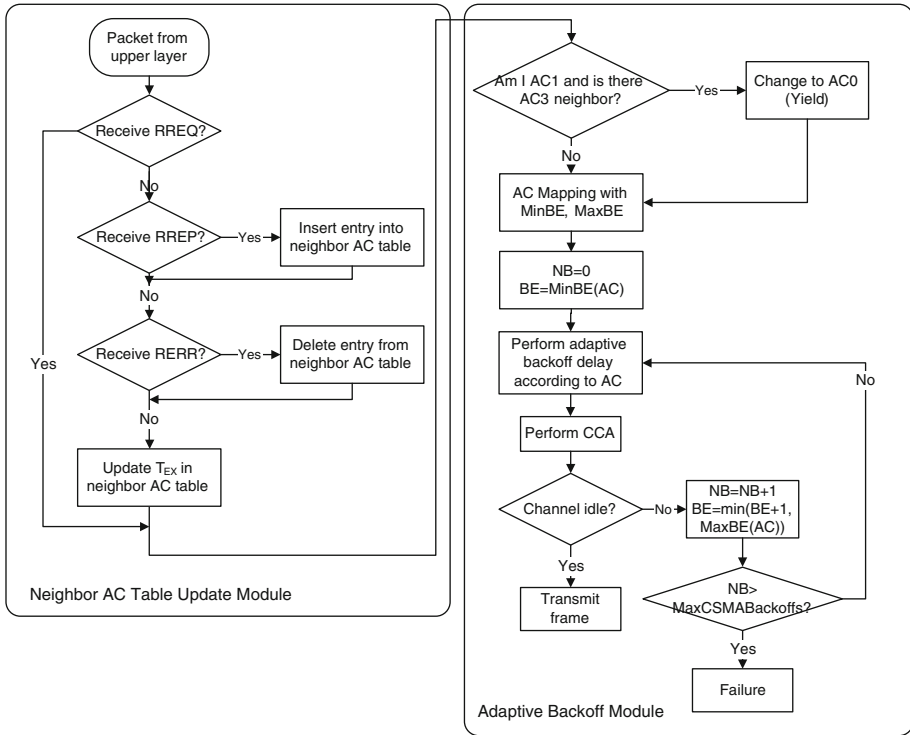


Fig. 5 Flow chart of the MAC operations in CCAR

network congestion and load balancing during experiments. The network traffic is increased with a packet arrival time interval ranging from 0.1 to 0.8 s. In each data flow, the source node is randomly selected without duplication while the destination is fixed as the predetermined sink node. We configure the node scheduling with a 100% duty cycle by setting the Beacon Order (BO) equal to 3 and the Superframe Order (SO) equal to 3, both of which are suitable for representing continuous data transmissions. The duty cycle is calculated by Eq. (5).

$$duty_cycle = \frac{Superframe_Duration}{Beacon_Interval} = \frac{1}{2^{BO-SO}} \tag{5}$$

The maximum buffer size of the interface of each node is set to 50, and three different B_{TH}^i values of 15, 30, and 45 with 3 different θ values of 5, 10, and 15 s are used to observe their effects. Both the active route timeout which denotes how long a route is kept in the routing table and T_{EX} in the AC neighbor table are set to 10s according to earlier research results [27]. However, if the target network requires other dynamic parameters such as node mobility and various link error ratios, these timeout values also need to be adjusted dynamically, which is beyond the scope of this paper.

For an evaluation and performance comparison with conventional protocols, we employed the following major performance metrics:

- End-to-end packet delivery ratio—the average number of data packets actually received by the sink node over the number of data packets originated by sources.

Fig. 6 Number of dropped packets

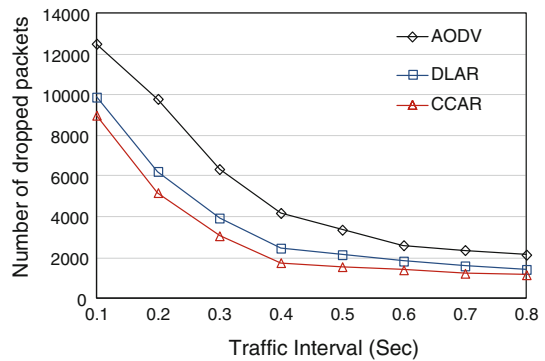
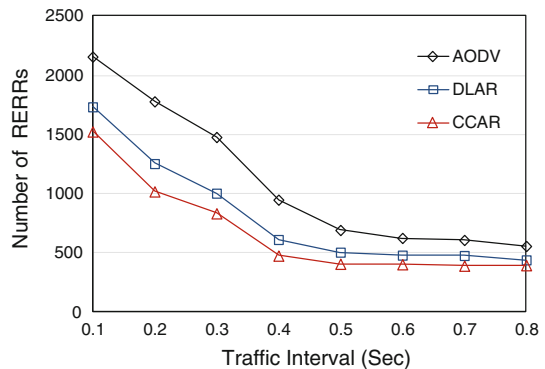


Fig. 7 Number of RERRs



- End-to-end delay—the average time that elapses from the time a data packet is originated by a source to when it is successfully received by the sink node.
- Normalized control packet overhead—the total number of data and control packets transmitted by a sensor node in the network divided by the total number of data packets received by the sink node.

In addition, we measured the number of dropped packets and the number of route failures to illustrate the efficiency of transmissions and route maintenance, respectively.

Figures 6 and 7 illustrate the number of dropped packets at the interfaces of the nodes and the number of transmitted RERR packets, respectively. These results indicate the degree of transmission and connection failures mainly caused by buffer overflows and packet collisions. As the network traffic increases (e.g., due to a short packet arrival interval), the number of transmission errors also increases. However, CCAR shows fewer errors than the conventional AODV and DLAR schemes, particularly during heavy traffic conditions because the proposed load-balancing scheme successfully avoids congested routes and has a more robust route-maintenance mechanism. Although DLAR also outperforms AODV, it results in more packet errors than CCAR because DLAR only consider the traffic load along the route and does not use optimized cross-layer operations. In addition, AODV uses its routing table during the route-discovery procedure, which substantially increases the number of transmission errors committed when stored information is stale. However, CCAR acquires fresher route information by preventing intermediate nodes from transmitting RREP packets.

Figure 8 describes the packet delivery ratios of the AODV, DLAR and CCAR schemes as a function of the traffic interval. The delivery ratio of CCAR is better than those of AODV

Fig. 8 Delivery ratio

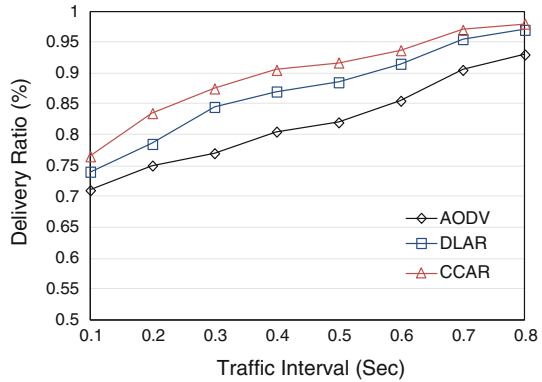
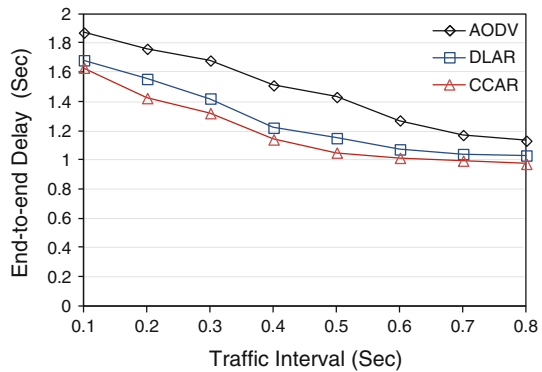


Fig. 9 End-to-end delay



and DLAR in every interval because there are fewer packet losses as a result, as shown in Figs. 6 and 7. Moreover, given that CCAR does not experience a rapid decrease even during heavy traffic conditions, it is clear that CCAR is more stable and reliable than the others. However, when the traffic is extremely heavy (e.g., a traffic interval of 0.1) or light (e.g., a traffic interval of 0.8), the performance gap is saturated owing to the occurrence of network congestion overall and clear channel transmission, respectively.

The packet end-to-end delay is shown in Fig. 9 as a function of the traffic load. Although all protocols show similar patterns with a delay increase in congested situations, both AODV and DLAR have higher delivery latency than CCAR, as they experience more transmission failures due to buffer overflows and collisions. These failures result in additional RREQ flooding to search for alternative routes, which significantly increases the packet end-to-end delay as well as the probability of packet collisions. Furthermore, the effective route maintenance scheme of CCAR for monitoring substantially congested nodes also improves the end-to-end delay situation by mitigating RERR packets in intermediate nodes. Figure 10 shows a comparison of the normalized routing overhead according to the traffic interval. As shown in the Fig., CCAR outperforms AODV and DLAR when the traffic interval is between 0.1 and 0.6 due to the fact that the congestion-avoidance mechanism of CCAR mitigates unnecessary evocations of the route rediscovery process, which requires additional control packet generation steps such as RREQ and RREP. However, when a low traffic load exists on the network (e.g., an interval between 0.7 and 0.8), AODV outperforms the other protocols because it uses its route cache information to conduct rapid searches at

Fig. 10 Control packet overhead

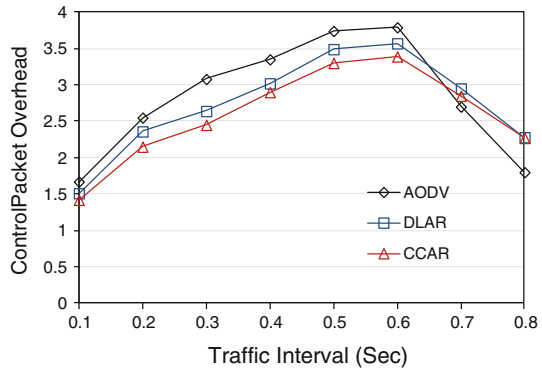
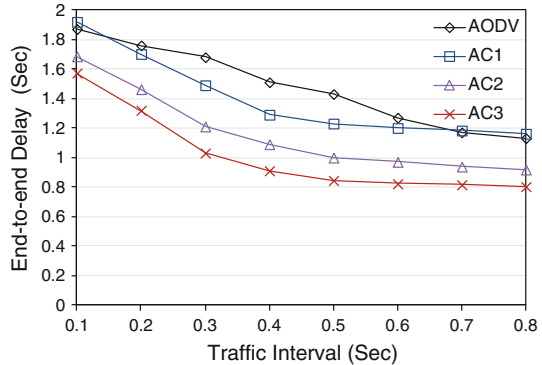


Fig. 11 End-to-end delay for ACs



intermediate nodes as well as source nodes during the route-discovery process. Although this route cache mechanism suppresses RREQ forwarding to destination nodes, it may contain stale cache information, which eventually increases the number of route failures in a heavily loaded network. Meanwhile, the overhead gradually decreases as the traffic load increases (e.g., at an interval between 0.1 and 0.6), as packets are transmitted through a previously established route, which is one of general features of on-demand routing protocols. When the interval is between 0.7 and 0.8, the overhead is decreased again because the non-busy channel experiences fewer route failures, which finally mitigates route-rediscovery evocations with RREQ flooding.

For a performance comparison between the internal ACs of CCAR, we measured the end-to-end delay. This result is shown in Fig. 11 as a function of the traffic interval. The result shows that the end-to-end delay of AC3 is faster than those of AC2 and AC1 because the flow with AC3 accesses the channel with a shorter backoff delay. In contrast, AC1 experiences a higher delay than the others because it acquires the channel during the MAC competition time. Furthermore, for this reason, AC1 experiences a longer end-to-end delay than AODV when the intervals are 0.1, 0.7 and 0.8, while both AC3 and AC2 outperform AODV regardless of the traffic interval. However, AC1 also outperforms AODV when the interval is between 0.2 and 0.6 via load-balancing routing. Above all, due to the concession by AC1, medical-grade data denoted as AC3 will realize significantly improved end-to-end delays for emergency applications.

Finally, Tables 2 and 3 show a performance comparison of the delivery ratio, end-to-end delay and the number of dropped packets according to various predefined parameters

Table 2 Various parameters of CCAR with a 0.2 traffic interval

Parameter		Traffic interval of 0.2 s		
B_{TH}^i	θ	Delivery ratio	End-to-end delay	Packet dropped
45	15	0.84	1.49	5485
30	10	0.83	1.42	5196
15	5	0.83	1.43	5238

Table 3 Various parameters of CCAR with a 0.7 traffic interval

Parameter		Traffic interval of 0.7 s		
B_{TH}^i	θ	Delivery ratio	End-to-end delay	Packet dropped
45	15	0.96	1.03	1288
30	10	0.97	0.99	1160
15	5	0.97	1.01	1239

(B_{TH}^i and θ) of CCAR to identify their effects and contributions. Table 2 represents a heavily loaded condition with a traffic interval 0.2 s, while Table 3 shows a light traffic condition with a traffic interval 0.7 s. Although it is not easy to find the optimal parameters due to the dynamic network conditions, each parameter combination affects the performance of CCAR somewhat. As shown in both tables, the performance is slightly improved when B_{TH}^i is set to 30 and when θ is set to 15, which indicates that when B_{TH}^i corresponds to 60% of the total buffer size, CCAR can approximate the optimal level of performance.

5 Application Design with the HL7 Interface

As our target application for the CCAR protocol is to support a medical information exchange system based on WSNs, we designed a simple but effective HL7-based personal healthcare monitoring system which consists of a number of client agents and a server agent. The client agents gather health-related data via pre-developed healthcare sensors (e.g. blood pressure monitor, pulse oximeter, EKG monitor and etc.) which are directly connected to client agents. The healthcare sensors for experiments are shown in Fig. 12. After gathering health data, the user can check whole information from the application GUI and then transmits it to a remotely located server agent. The topology architecture is illustrated in Fig. 13 and sensor nodes are logically classified into two groups to gathering HL7 data and to gather general data. The medical data group is developed for encoding and transmitting HL7 messages such as patient admissions (e.g., the User ID, name, address, and phone number), medical data (e.g., height/weight, body temperature, blood pressure, heart rate, and SPO2 level). This HL7 data corresponds to the AC3 type of CCAR for privileged transmissions. The general data group is used for non-medical data transmissions and corresponds to the AC1 and AC2 data types of CCAR. After all data packets are transmitted to the HL7 server, which is considered as the sink node, the server stores the received data in a health database for monitoring and for further examinations by potential users from remote sites.

For an actual implementation assessment, we developed sensor devices using the CC2420 model by Texas Instruments (TI) Chipcon supplied the RF transceiver and the TI MSP4030 (16-bit) model served as the main processor. The wireless radio transmission range was 30 ~ 40 meters and the RF power control ranged from 0 to -20dBm. In the test-bed, we

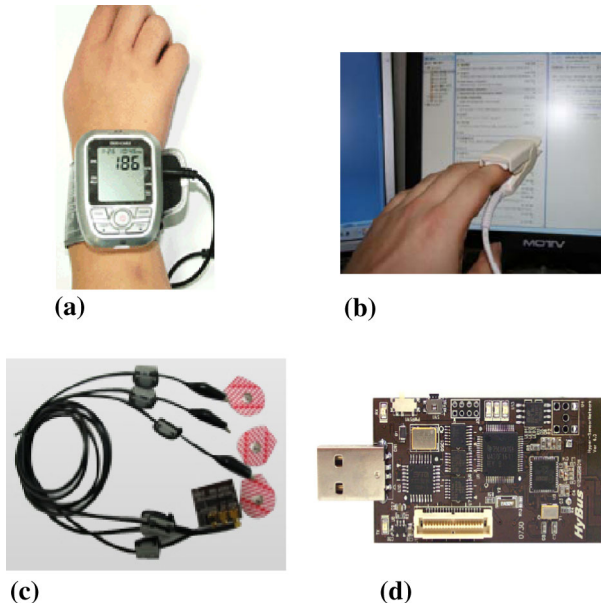


Fig. 12 Healthcare sensors for experiments **a** Blood pressure monitor **b** Pulse oximeter **c** EKG monitor **d** Sensor mote

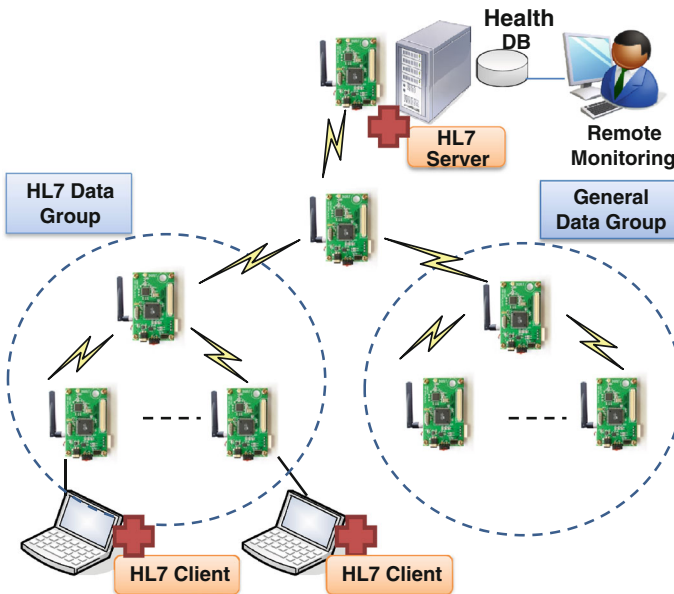


Fig. 13 Conceptual topology of the application system

used five sensor nodes for the HL7 client (AC1) and 25 nodes for the general data group (10 nodes for AC2 and 15 nodes for AC1). All sensor nodes were randomly distributed with a maximum hop distance of 3–4 from the sink node. For the operating software, we adopted TinyOS

Table 4 Measurement of the packet loss ratio

Node ID (HL7 group)	Loss ratio (%)	Node Id (General group)	Loss ratio (%)
A	3.91	V	4.85
B	4.12	W	5.11
C	3.66	X	5.03
D	2.98	Y	4.74
E	3.75	Z	4.94

Table 5 Measurement of the packet delivery latency

Node ID (HL7 group)	Delivery latency (s)	Node Id (General group)	Delivery latency (s)
A	0.98	V	1.26
B	1.01	W	1.31
C	0.99	X	1.21
D	0.97	Y	1.27
E	1.12	Z	1.32

[21]. The length of each data packet was 40 bytes and the duty cycle was fixed at 100 %, which indicates that all sensor nodes transmit packets during transmission interval in order to process privileged medical data. In addition to a sleepless transmission scheme, if the network is characterized by specific bursty traffic instances of route reestablishment, channel interference and collisions, is simulates a highly loaded network. Although a performance evaluation was conducted in a previous section, we have performed another experiment using a Sensor Network Analyzer (SNA) by Daintree Networks [29] to measuring the packet loss ratio and the delivery latency during the communication between the end node and the sink node.

Table 4 shows the measured results of the packet loss ratio, which is the ratio of the number of failed packets at the destination to the total number of transmitted packets by source nodes. In Table 4, the loss ratios of all HL7 nodes show better results than general nodes by the fact that the ratios of HL7 group are less than 5 %, which is considered more robust for managing healthcare data. When the HL7 server successfully received messages without any packet loss, it decodes the messages via a HL7 parser. Table 5 shows a performance comparison of the delivery latency rates between HL7 clients and the general nodes. As shown in the results, nodes belonging to the HL7 group demonstrate better performance than the other nodes, as CCAR provides HL7 clients with privileged transmission opportunities. Hence, this experiment reveals that the proposed CCAR is highly suitable for supporting medical-grade data when the application data is classified into predefined ACs.

In summary, the performance of CCAR is evaluated by both test-bed implementation and simulations, where the results show that CCAR provides good performance in terms of packet delivery ratio, end-to-end delay and normalized control overhead especially when the network requires medical grade packet differentiation and dynamic load balancing.

6 Conclusion

WSN technology with support for medical-grade data is considered to be one of the core components for developing various medical monitoring systems which require QoS support

for privileged medical data transmissions. In this paper, we proposed an efficient protocol termed CCAR, which not only provides medical-grade QoS support but also achieves load balancing while communicating in multi-hop-based WSNs. To do this, CCAR newly defines a QF parameter and AC categories to estimate the link quality and for packet discrimination, respectively. CCAR then attaches the calculated parameters onto RREQ packets to find the optimal route with the lowest traffic load. During the communication session, the MAC module of CCAR dynamically adjusts the channel contention according to the AC of each packet. Through a simulation and an experiment involving an actual test-bed, we show that CCAR outperforms conventional protocols in terms of the packet delivery ratio, end-to-end delay, and the control packet overhead, thus demonstrating that the proposed method is suitable for use in medical information systems.

In future works, we plan to investigate more detailed QoS requirements of medical information systems in order to design a more robust protocol than CCAR. In addition, we will study an energy-efficient protocol for resource-limited medical WSNs.

Acknowledgments This work was supported by Daegu Gyeongbuk Institute of Science and Technology (DGIST) and was funded by Ministry of Science, ICT and Future Planning of Korea.

References

1. Ben-Othman, J., Mokdad, L., Yahya, B. (2011). An energy efficient priority-based QoS MAC protocol for wireless sensor networks. In *Proceedings of IEEE ICC*.
2. Ben-Othman, J., & Yahya, B. (2010). Energy efficient and QoS based routing protocol for wireless sensor networks. *Journal of Parallel and Distributed Computing*, 7(8), 849–857.
3. Chen, S.-K., et al. (2012). A reliable transmission protocol for ZigBee-Based wireless patient monitoring. *IEEE Transaction on Information Technology in Biomedicine*, 16(1), 6–16.
4. Chen, X., Haley, M.-J., & Jayalath, A. (2007). Congestion aware routing protocol for mobile ad hoc networks. In *Proceedings of IEEE VTC 2007*.
5. Chen, S.-K., Kao, T., Chan, C.-T., Huang, C.-N., Chiang, C.-Y., Lai, C.-Y., et al. (2012). A reliable transmission protocol for ZigBee-based wireless patient monitoring. *IEEE Transactions on Information Technology in Biomedicine*, 16(1), 6–16.
6. Choi, J., Yoo, S., Park, H., & Chun, J. (2006). MobileMed: A PDA-based mobile clinical information system. *IEEE Transactions on Information Technology in Biomedicine*, 10(3), 627–635.
7. Dam, T.-V., & Langendoen, K. (2003). An adaptive energy-efficient MAC protocol for wireless sensor networks. In *Proceedings of the international conference on embedded networked sensor systems*.
8. Dessart, N., Fouchal, H., & Hunel, P. (2010). Distributed diagnosis over wireless sensors networks. *Concurrency and Computation: Practice and Experience*, 22(10), 1240–1251.
9. Djenouri, D., & Balasingham, I. (2011). Traffic-differentiation-based modular QoS localized routing for wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(6), 797–809.
10. Draft IEEE Std. 802.15.4, Part 15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area, Networks (WPANs). September 2006.
11. Felemban, E., Lee, C.-G., & Ekici, E. (2006). MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 5(6), 738–754.
12. Health Level Seven Standard, URL:<http://www.hl7.org/>.
13. Hu, F., Xiao, Y., & Hao, Q. (2009). Congestion-aware, loss-resilient bio-monitoring sensor networking for mobile health applications. *IEEE Journal of Selected Areas in Communications*, 27(4), 450–465.
14. Hughes, B., Cahill, V. (Dec. 2003). Achieving real-time guarantees in mobile ad hoc wireless networks. In *24th IEEE Real-time systems symposium*.
15. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements, 2005.
16. Johnson, D., Hu, Y., Maltz, D. (February 2007). The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4", IETF RFC 4728.
17. Kang, K.-T., Park, K.-J., Song, J.-J., & Yoon, C.-H. (2011). A medical-grade wireless architecture for remote electrocardiography. *IEEE Transactions on Information Technology in Biomedicine*, 15(2), 260–267.

18. Kemal, A., & Mohamed, Y. (2005). A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(5), 325–349.
19. Lee, S.-J., & Gerla, M. (2001). Dynamic load-aware routing in ad hoc networks. In *Proceedings of IEEE ICC*.
20. Lee, Z.-K., Lee, G.-C., Oh, H.-R., & Song, H. (2012). QoS aware routing and power control algorithm for multimedia service over multi-hop mobile ad hoc network. *Wireless Communications and Mobile Computing*, 12(7), 567–579.
21. Levis, P., Madden, S., Polastre, J., Szewczyk, R., K. et al. (2005). TinyOS: An operating system for sensor networks ambient intelligence. *Ambient Intelligence*, pp. 115–148.
22. McCanne, S., & Floyd, S. NS network simulator. URL:<http://www.isi.edu/nsnam/ns>.
23. Morak, J., Kumpusch, H., Hayn, D., Modre-Osprian, R., & Schreier, G. (2012). Design and evaluation of a telemonitoring concept based on NFC-enabled mobile phones and sensor devices. *IEEE Transactions on Information Technology in Biomedicine*, 16(1), 17–23.
24. Perkins, C. E., Royer, E., & Das, S. (July 2003). Ad-hoc on-demand distance vector routing. IETF RFC 3561.
25. Polastre, J., Hill, J., Culler, D. (Nov. 2004). Versatile low power media access for wireless sensor networks. In *Proceedings of the ACM SenSys'04*.
26. Raza, I., Chaudhry, S. R., Hussain, S. A., Abid, S. A., & Raza, H. (2012). Optimised priority assignment mechanism for biomedical wireless sensor networks. *IET Wireless Sensor Systems*, 2(2), 92–102.
27. Richard, C., Perkins, C., & Westphal, C. (2005). Defining an optimal active route timeout for the AODV routing protocol. In *Proceedings of the IEEE SECON*.
28. Saxena, N., Roy, A., & Shin, J. (2008). QoS-based energy-aware MAC protocol for wireless multimedia sensor networks. In *Proceedings of the IEEE VTC*.
29. Sensor Network Analyzer (SNA), URL:<http://www.daintree.net/products/sna.php>.
30. Shah, G., Liang, W., & Akan, O. (April 2012). Cross-layer framework for QoS support in wireless multimedia sensor networks. *IEEE Transactions on Multimedia*. doi:10.1109/TMM.2012.2196510.
31. The Institute of Electrical and Electronics Engineers, Inc. (2008). *Health informatics-personal health device communication part 20601: Application of a profile-optimized exchange protocol, IEEE Std 11073–20601-2008*, pp. 1–198.
32. Wang, M., Kuo, G.-S. (Sept 2005). An application-aware QoS routing scheme with improved stability for multimedia applications in mobile ad hoc networks. In *Proceedings of the IEEE VTC 2005*.
33. Wang, H., Zhang, X., Naït-Abdesselam, F., & Khokhar, A. (2010). Cross-layer optimized MAC to support multi-hop QoS routing for wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 59(5), 2556–2563.
34. Xue, Q., & Ganz, A. (2003). Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks. *Journal of Parallel and Distributed Computing*, 63(2), 154–165.
35. Ye, W., Heidemann, J., & Estrin, D. (2004). Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(3), 493–506.
36. Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks*, 52(12), 2292–2330.

Author Biographies



Young-Duk Kim received the B.S. degree in Computer Education from SungKyunKwan University, Korea, in 2004 and the M.S degree in Computer and Communication Engineering from POSTECH, Korea, in 2006. He is currently a Ph.D candidate at Kyungpook National University, Korea. Since 2006, he has been with IT Convergence Laboratory, Daegu Gyeongbuk Institute of Science and Technology (DGIST), where currently he is a Senior Research Engineer. His research interests include wireless sensor networks, mobile ad hoc networks, u-healthcare systems and intelligent embedded systems.



Kook-Rae Cho received a B.S. degree in Computer Science from Bukyoung University, Korea in 2004, and a M.S degree in Computers and Communications Engineering from Pohang University of Science and Technology (POSTECH), Korea in 2006. Since 2006, he has been with the Division of IT Convergence, Daegu Gyeongbuk Institute of Science and Technology (DGIST), where currently he is a researcher. His research interests include u-healthcare systems, mobile security and intelligent embedded systems.



Hui-Sup Cho received the B.S. and M.S degrees in Electronics from Kyungpook National University, Korea in 1999 and 2005, respectively. He was a research engineer at LG Electronics, Korea (2001–2005). Since 2005, he has been with the Division of IT Convergence, Daegu Gyeongbuk Institute of Science and Technology (DGIST), where currently he is a senior researcher. His research interests include u-healthcare systems, biosensor systems and intelligent embedded systems.



Dongkyun Kim received B.S. degree in the Department of Computer Engineering, Kyungpook National University, Daegu, Korea, in 1994. He also obtained the M.S. and Ph.D. degrees in the School of Computer Science and Engineering, Seoul National University, Seoul, Korea, in 1996 and 2001, respectively. He was a visiting researcher at the Georgia Institute of Technology, Atlanta, in 1999. He also performed a post-doctorate program in the Computer Engineering Department, University of California at Santa Cruz, in 2002. He served a vice chair of IEEE TCPC Ad Hoc Mobile Wireless Networks Subcommittee (2000–2002) and a TPC chairman of the Symposium on Ad Hoc Wireless Networks (IEEE SAWN 2002 in conjunction with IEEE GLOBECOM 2002) and has been a TPC member of several IEEE conferences. He has also been a TPC member of IASTED conferences. Currently, he is a professor in the School of Computer Science and Engineering, Kyungpook National University, Korea. His research interests include mobile ad hoc networks, wireless sensor networks, wireless LAN and next-generation all IP networks.