An Adaptive Congestion Alleviating Protocol for Healthcare Applications in Wireless Body Sensor Networks: Learning Automata Approach

Nazbanoo Farzaneh Bahalgardi[i]; Mohammad Hossein Yaghmaee[ii *] and Donald Adjeroh[iii]

ABSTRACT

Wireless Body Sensor Networks (WBSNs) involve a convergence of biosensors, wireless communication and networks technologies. WBSN enables real-time healthcare services to users. Wireless sensors can be used to monitor patients' physical conditions and transfer real time vital signs to the emergency center or individual doctors. Wireless networks are subject to more packet loss and congestion. To alleviate congestion, the source transmission rate and node arrival rate should be controlled. In this paper, we propose Learning based Congestion Control Protocol (LCCP) for wireless body sensor networks. LCCP joins active queue management and rate adjustment mechanism to alleviate congestion. The proposed system is able to discriminate different physiological signals and assign them different priorities. Thus, it would be possible to provide better quality of service for transmitting highly important vital signs. The simulation results confirm that the proposed protocol improves system throughput and reduces delay and packet dropping. We also evaluate the performance of the AQM mechanism with no rate adjustment mechanism to show the advantage of using both AQM and rate adjustment mechanism together.

KEYWORDS

Active Queue Management, Congestion Control, Learning Automata, Transport protocol, Wireless Body Sensor Network

1. INTRODUCTION

The use of Wireless Sensor Networks (WSN) in healthcare applications is growing fast in the recent years. WBSN is a collection of small, low power sensing devices wirelessly connected to a resource-rich aggregation device [1]. The wireless body sensor network plays an important role for healthcare monitoring applications. For these applications, it is essential to be able to reliably collect physiological readings from humans via body sensor networks. Such networks could benefit from Quality of Service (QoS) mechanisms that support prioritized data streams, especially when the channel is impaired by interference or fading [2]. Due to the nature of wireless sensor networks, congestion occurrence is an unavoidable problem. It not only wastes the scarce energy due to a large number of retransmissions and packet drops, but also hampers the event detection reliability. In WSN it is important to know how to detect congestion and how to control it. The congestion results in a long delay in data delivery and wasting of energy due to lost or dropped packets. Healthcare applications require QoS in terms of both packet loss rate and delay. The vital signs should be delivered to the emergency center with lower packet loss and delay. As a result alleviating congestion is indispensable in WBSNs.

This paper addresses the problem of congestion control in wireless body sensor networks. In this paper a joint active queue management and rate adjustment mechanism based on learning automata is presented. The rest of this paper is organized as follows. In section 2, we present a review of the related researches in transport protocols in WSN and WBSN. A brief overview of learning automata is presented in section 3. In section 4, the proposed protocol is fully derived. Section 5 evaluates the proposed mechanism both theoretically and through simulations. We conclude the paper in section 6.

i Nazbanoo Farzaneh. Computer Engineering Department, Ferdowsi University of Mashhad, Azadi Square, Mashhad, IRAN. (email: Farzaneh@stu-mail.um.ac.ir)

ii * Corresponding Author, Mohammad Hossein Yaghmaee Moghaddam is with the Department of Computer Engineering and Center of Excellence on Soft Computing and Intelligent Information Processing, Ferdowsi University of Mashhad, Mashhad, Iran (e-mail: yaghmaee@ieee.org).

iii Donald Adjeroh, Full Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV 26506 (email: don@csee.wvu.edu)

2. RELATED WORK

The study of transport protocol in WSNs has been the subject of extensive research. In [3] a new priority based rate and congestion control protocol for wireless multimedia sensor networks is presented. It consists of two major units, namely Congestion Control Unit (CCU) and Service Differentiation Unit (SDU). Each sensor node has two different priority indexes: *traffic class priority* and *Geographical priority*. The total priority of a node is defined as the product of these two priorities. The CDU determines the congestion intensity by calculating the difference between the input and the output rate. The rate adjustment unit calculates the new rate based on the congestion index and source traffic priority. The SDU supports different QoS for different traffic classes. The RCRT [4] protocol uses the length of retransmission list as the indicator of congestion. The presence of too many packets in the retransmission list will be interpreted as high density congestion. The RCRT protocol uses its rate allocation component to assign rates to each flow in keeping with a rate allocation policy. RCRT boasts a NACK based end-to-end loss recovery scheme. The sink detects packet losses and repairs them by requesting end-to-end retransmission of the packets from source nodes. In [5] a novel congestion control protocol for vital signs monitoring in wireless biomedical sensor networks is proposed. To minimize congestion in each intermediate sensor node, a separate queue is allocated to each child node to store its input packets. To discriminate between different traffic classes in each intermediate node they use a multi-threshold mechanism. Based on the current congestion degree and the priority of its child nodes, the parent node dynamically computes and allocates the transmission rate for each of its children. When the central computer which maintains the physiological data for each patient detects any anomaly in the received data, it sends a special message to the particular patient's sensor node and increases the patient's priority. LACAS [6] is a Learning Automata-Based Congestion Avoidance Scheme for healthcare wireless sensor networks. In LACAS there is an automaton in every intermediate node which regulates the node's incoming rate for controlling congestion locally in that node. For the input to the automaton at time, *t*=0, the automaton has five action which are based on the rate with which an intermediate sensor node receives the packets from the source node. The learning parameter is drop packets. The most optimal action, at any time instant, among the set actions in a node, is decided by the number of packets dropped. To be precise, the rate of flow of data into a node for which there is the least number of packets dropped is considered to be the most optimal action. In [7], a mobile environment, in which intermediate nodes and destination nodes (doctors) are mobile, is considered and a modification of LACAS for mobile environment is presented. A dynamic quality of service (QoS) approach for U-healthcare in Wireless multimedia sensor and actor networks is presented in [8]. The authors consider multiple QoS constraints to optimize the network utilization. Multiple classes of health information are considered. Each class has a bandwidth level. In order to adjust the transmission rate, when available bandwidth is less than the required bandwidth, a node decides which packet classes should be dropped. HU and et al in [9] proposed an accurate feature extraction method to compress the healthcare signals to reduced congestion. Compression data can also reduce data rate. For this purpose, a method based on multi-scale wavelet analysis is presented. LACCP is proposed in [10]. LACCP is a congestion control protocol based on learning automaton in WBSNs. LACCP can adjust intermediate node arrival rate and source sending rate using learning automata.

3. LEARNING AUTOMATA

This section presents a brief overview of learning automata [11]. A learning automaton is a mechanism that can be applied to learn the characteristics of a system's environment. An environment is represented by a triple $E \equiv \{\alpha,\beta,c\}$, where $\alpha \equiv \{\alpha_1,\alpha_2,...,\alpha_r\}$ is defined as all actions of the automaton and r is the total number of actions. $\beta \equiv \{\beta_1,\beta_2,...,\beta_m\}$ denotes the response received by the automaton and $c \equiv \{c_1,c_2,...,c_r\}$ represents the set of penalty probabilities (each c_i corresponds to an action α_i in set, α) .the goal here is to find an optimal action among a set of actions so that the average penalty received by the environment is minimized. Automaton uses a vector $P(n)=\{P_1(n),P_2(n),...P_r(n)\}$ which represents the probability distribution for choosing one of the actions at cycle n. In each cycle n, an action α_i is selected with probability p_i and the environment provides a penalty or reward c_i, which is used by the automaton to update the probabilities in $P(n)$. The general scheme for updating action probabilities is:

$$
\begin{aligned}
P_i(n+1) &= P_i(n) + (1-\beta(n))\sum_{j,i} g_j(P(n)) \\
&\quad - \beta(n)\sum_{j,i} h_j(P(n)) \;,\quad if\,\alpha(n)=\alpha_i \\
P_i(n+1) &= P_i(n) - (1-\beta(n))g_i(P(n)) \;+ \\
&\quad + \beta(n)h_i(P(n)) \;,\quad if\,\alpha(n)\neq\alpha_i
\end{aligned} \tag{1}
$$

where $\beta(n)$ is normalized in [0,1]. The lower the value of $\beta(n)$ the more favorable the response. g_i and h_i (i=1,2,...,r) are continuous, nonnegative functions and associated with reward and penalty functions for action

α_i, respectively. Depending on the functions g_i and h_i, several linear and non-linear reinforcement (updating) schemes can be obtained. Linear schemes are the simplest and commonly used. In general definition of linear reinforcement schemes, reward and penalty functions can be obtained as:

$$g_k(P(n)) = a.P_k(n) \quad , \quad h_k(P(n)) = \frac{b}{r-1} - b.P_k(n) \qquad 0 < a, b < 1 \tag{2}$$

The parameter "a" is associated with the reward response, and the parameter "b" with the penalty response. [11].

4. PROPOSED MODEL

One of the causes of congestion is the lack of correspondence between the source packet sending rate and the network available capacity. Network congestion causes longer queues in intermediate nodes leading to greater end to end delays, which results into lower network throughput.

In WBSN sensors are attached to different patients. Each sensor is used to monitor a vital sign. Evidently, in such life-critical applications involving a large number of patients, congestion is extremely undesirable.

LACAS[6] is the automata base congestion control protocol in healthcare WSN. All the intermediate nodes have automata stationed in them, which are tasked to monitor and control the rate of flow of data through them. The base of LACAS is on equating the packet arrival rate and the packet service rate. LACAS tries to make both of these rates equal, preventing any kind of queuing at the nodes to a large extent. Although LACAS is capable of adaptively learning and "intelligently" choosing "better" data rates, it has few following drawbacks:

- LACAS limited the number of rates (actions) associated with an automaton to 5. These 5 rates are defined randomly and not changed during simulation. As a result the network may have poor performance due to the selecting non-proper rates (actions). Non-proper rate allocation may result in inefficient channel utilization.
- LACAS does not require the source nodes to be fed back by the intermediate nodes to slow down. Although this action can reduce the number of forwarded messages, it could not improve the performance of the network. If the congestion condition continues, intermediate node's queue lengths increase suddenly, this leads to an increase in the number of drop packets. Therefore, other mechanisms such as redirect, path change, source rate decrease and etc are required to reduce congestion.
- Although LACAS has been presented for healthcare applications, it does not consider different types of vital signs. In LACAS all traffics are the same.

In this section, we explain the proposed model in detail. Figure 1 shows an overall view of the system under study. The proposed congestion control and service differentiation protocols are placed in all sensor nodes in the system which is designed for remote monitoring of patients. The proposed model consists of two different parts: 1) a learning and priority based AQM protocol and 2) a learning automata based rate adjustment mechanism. For simplicity, we name the proposed model as LCCP (Learning based Congestion Control Protocol). Since there are different traffic flows in the WBSN network, and each of the network traffic has its own requirements, the network should behave differently in response to different types of the network traffic. In the current study we consider 2 different classes, namely *Critical* and *Normal*. A c*ritical* class is dedicated to loss and delay intolerant traffics and a *normal* class belongs to other traffics. A learning based AQM protocol is used in the intermediate nodes in order to prevent sudden saturation of queue. This will in turn result in controlling the delay and congestion. The learning automata based transport protocol is located in the sink adjusts the source rate so that the network throughput is also maximized while the congestion is prevented. In the following subsections, we describe these two parts in details.

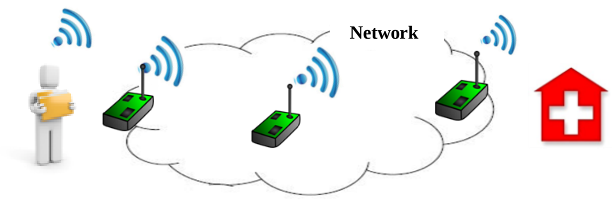


Figure 1: System overview

A. AQM mechanism in intermediate nodes

Since the queues in intermediate nodes have a limited capacity, if the rate of packet arrival is more than that of the packet exit, the queue will be filled up. This, in turn, causes increased packet loss and delays. Sudden saturation of queues can be avoided by controlling the rate of packets entering intermediate nodes. In a healthcare application, different patients would have different medical records in the system. If a patient is known to have a special need, it would be possible to assign more priority to the data transmitted from such a patient. In the proposed AQM protocol a packet is entered to the node's queue, based on its traffic class (Critical or Normal). Each node uses learning automata in order to adjust the rate of packet arrival. Figure 2 shows learning automata structure of intermediate nodes.

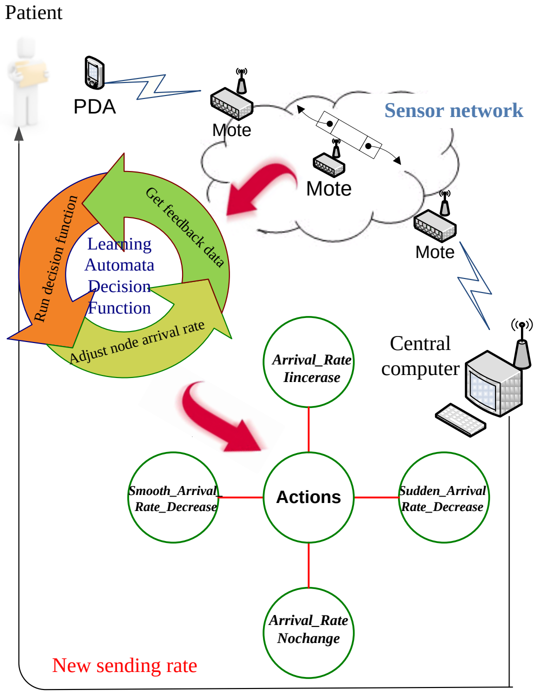


Figure 2: Structure of intermediate nodes

In each intermediate node, there is a variable automaton which is shown by {A, B, P, T} where *A* is a set four actions as follows:

A = {*Arrival_RateIincerase*, *No_change* , *Smooth_Arrival_Rate_Decrease*, , *Sudden_Arrival_Rate_Decrease* }. *B* includes the set of inputs and *P* is the probability vector of the four automata actions. *P(n+1) = T[a(n),B(n),P(n)]* is the learning automata where *n* is the stage number. Each automaton receives feedback from the environment (the network) after selecting every action and rates it either positive or negative. Based on the action, the transmission rate could be increased or decreased. A summary of the four automata actions are given below:

- **No change:** The network has reached stability and there is no need to change the arrival rate.
- **Arrival_Rate_Increase:** Since the network feedback indicates small queue size and channel load, the node can increase arrival rate in order to improve throughput.
- **Smooth_Arrival_Rate_Decrease:** Congestion is likely to occur or low density congestion has occurred. Therefore, packet arrival rate is decreased smoothly in order to avoid queue saturation.
- **Sudden_Arrival_Rate_Decrease:** Since congestion has occurred packet arrival rate is decreased quickly in order to avoid packet loss and queue saturation.

The intermediate nodes' learning automata adjusts the transmission rate based on two following learning parameters:

1) Number of packets in the queue
2) Number of lost packets

These two parameters provide a good assessment of the automata performance. Queue length and packet loss rate can be used as congestion detection parameters. A high arrival rate may cause a smaller free queue space and a larger packet loss rate. Let $\Delta L(t)$ and $L(t)$ show the loss variation and loss rate in time *t*. The learning automata placed in each node calculates $\Delta L(t)$ and $\Delta Q(t)$ as:

$$\Delta L(t) = L(t) - L(t-1),$$
$$\Delta Q(t) = Q(t) - Q(t-1) \qquad (3)$$

where ΔL and $L(t)$ are loss variation and loss amount in time *t*, respectively. $L(t-1)$ is the loss rate at time *t-1*. ΔQ and $Q(t)$ are queue length variation and queue length in time *t* respectively. $Q(t-1)$ is the queue length at time *t-1*. Network status can be determined based on the values of ΔL and ΔQ. Different values of ΔL will be interpreted as follows:

- $\Delta L = 0$ The amount of packet loss has not changed.
- $\Delta L < 0$: The amount of network loss has decreased. The value of ΔL shows the amount of decrease.
- $\Delta L > 0$: The amount of network loss has increased. The value of ΔL shows the amount of increase.

Different values of ΔQ will be interpreted as follows:

- $\Delta Q = 0$: Queue length has not changed in the network.
- $\Delta Q < 0$: Queue length has decreased in the network. The value of ΔQ shows the amount of decrease.
- $\Delta Q > 0$: Queue length has increased in the network. The value of ΔQ shows the amount of increase.

After choosing an action, the automata rewards or penalizes it based on the network feedback and according to rules given in the Table 1:

TABLE 1
THE AUTOMATON REWARD AND PENALIZE RULES PLACED IN THE INTERMEDIATE NODES

Reward Rules	**Rule 1: If** $(\Delta Q(t) << 0 \ \&\& \ \Delta L(t) <= 0)$ **Then the automaton is rewarded**
	Rule 2: if $(\Delta Q(t) >> 0 \ \&\& \ \Delta L(t) < 0)$ **Then the automaton is rewarded**
Punishment Rules	**Rule 3: if** $(\Delta Q(t) >> 0 \ \&\& \ \Delta L(t) > 0)$ **Then the automaton is penalized**
	Rule 4: if $(\Delta Q(t) << 0 \ \&\& \ \Delta L(t) > 0 \ \&\& \ arrival\_rate << initial\_rate)$ **Then the automaton is penalized**

The automaton is rewarded and penalized according to (4) and (5), respectively.

$$P_i(n+1) = P_i(n) + a(1 - P_i(n))$$
$$P_j(n+1) = (1-a)P_j(n) \quad \forall j \; j \neq i \tag{4}$$

$$P_i(n+1) = (1-b)P_i(n)$$
$$P_j(n+1) = (b/r-1) + (1-a)P_j(n) \quad \forall j \; j \neq i \tag{5}$$

In the above equations, " a " is the reward and "b" is the punishment parameter.

Shorter queue lengths in intermediate nodes lead to a shorter packet waiting time for receiving service. Although decreasing packet arrival rate decreases queue length, it also increases the number of lost packets (packets that will be lost because of low arrival rate) and decreases throughput. Therefore, the automaton is to strike a balance between queue length and network throughput. The number of lost packets will be increased if despite the changing of channel load, packet arrival rate remains constant. Therefore, an increase in packet loss indicates a lack of correspondence between arrival rate and the actual packet reception rate (channel load). On the other hand, longer queue length is also an indicator of a high volume of packets arriving at the node which is a sign of congestion. Therefore, although the arrival rate should be decreased as queue length nears saturation, with the decrease of network throughput and queue length the rate of arrival increases again. In other words, if the queue length decreases but the number of lost packets increases, this is an indicator of lowering network throughput and the automata will increase the node arrival rate. An increase in the number of lost packets causes the automata in the sink to decrease source sending rate in order to decrease congestion and increase network throughput. Therefore, the automata placed in intermediate nodes and the sink will decrease congestion concurrently.

Despite to LACAS the values of " a " and "b" in (2) and (3) are not constant and defined based on the congestion level. Thus, various congestion levels have different effects on the automata. Although at the beginning of operation all probabilities P_i are equal, as time passes the reward and punishment mechanism explained above will change these probabilities. The node arrival rate (r_t) is updated according to the automata actions. Each node has two different arrival rates, namely r_c and r_n for the *Critical* and *Normal* traffic, respectively. Thus the following relation for the node arrival rate r_t is always true:

$$r_t = r_c + r_n \tag{6}$$

Thus if the number of received critical packets increases, the number of normal arrival rate is then decreased. The *Critical* and *Normal* rates are calculated as follows:

$$a_t = w_c a_c + w_n a_n \; ; \quad w_c \geq w_n$$
$$p_c = \frac{w_c a_c}{a_t} \quad , \quad p_n = \frac{w_n a_n}{a_t} \tag{7}$$
$$r_c = p_c r_t \quad , \quad r_n = p_n r_t$$

where a_t , a_c and a_n are the total received packets, total received *Critical* packets and total received *Normal* packets per time, respectively. w_c and w_n are the priority weights of *Critical* and *Normal* packets, respectively.

A weighted fair queue (WFQ) scheduler is used to schedule the incoming packets. To provide a better quality of service for high priority traffic classes, the assigned weights used in the WFQ scheduler follows this rule: *Critical Class has higher priority.*

B. Rate adjustment mechanism placed at the sink

In order to control and prevent congestion we need to adjust the source rate based on the congestion level in the network. To do so, a method is presented based on the learning automaton in the sink that learns about the network congestion status and assigns the proper rate of the source(patient) based on their current status(Critical or Normal). If this rate is too low, the network throughput decreases drastically. On the other hand, when the source rate is too high it causes congestion and reduces network performance. Therefore, the suitable source rate would be determined based on the network status.

In this subsection a new congestion control and rate adjustment protocol based on automaton is presented. In the proposed protocol, variable learning automaton is used. This automaton has six actions as follows:

A = {*CINI*, *CINO*, *CIND*, *CONI*, *CONO*} where *C* and *N* respectively retuned for *Critical* and *Normal* sources and *I*, *O* and *D* shows rate Increase, rate nO change and rate Decrease respectively. Table 2 gives a summary of the six automaton actions.

TABLE 2
DEFINITION OF AUTOMATON ACTIONS IN THE SINK

Actions		**Critical Sources(C)**	
		Increase rate(I)	Rate no-change(O)
Normal Sources(N)	Increase rate(I)	CINI	CONI
	Rate no-change(O)	CINO	CONO
	Decrease rate(D)	CIND	COND

Rate no-change(O): The network has reached stability and there is no need to change the source rate.
Increase rate(I): Network congestion has been reduced; therefore the source rate will be increased.
Decrease rate(D): Due to congestion and increase in delay and packet loss, the source rate should be decreased.

The rules of determining the desirability of the selected action are based on the 3 parameters as follows:

1)Number of lost packets between two successful *deliveries*

2)Throughput

3)Traffic class.

To calculate the packet loss rate and delay variation, the learning automaton in the sink node uses the following equations:

$$\Delta Ls^{c}(t) = Ls^{c}(t) - Ls^{c}(t-1),$$
$$\Delta Ls^{n}(t) = Ls^{n}(t) - Ls^{n}(t-1),$$
$$\Delta T^{c}(t) = T^{c}(t) - T^{c}(t-1) \quad (8)$$
$$\Delta T^{n}(t) = T^{n}(t) - T^{n}(t-1)$$

where $\Delta Ls^{c}(t)$ and $\Delta Ls^{n}(t)$ show the total loss variation of *Critical* and *Normal* traffics at time t ,respectively. $Ls^{c}(t)$ and $Ls^{n}(t)$ denote the packet loss rate of *Critical* and *Normal* traffics at time *t*, respectively.

$\Delta T^{c}(t)$ and $\Delta T^{n}(t)$ are throughput variation of *Critical* and *Normal* packets at time t respectively. $T^{c}(t)$ and $T^{n}(t)$ denote the throughput of *Critical* and *Normal* traffics at time *t*, respectively.

After choosing an action, the automata rewards or penalizes it based on the network feedback as given in Table 3:

In Table 3, "a" is the reward and "b" is the punishment parameter. "a" and "b" are not constant and determined based on the congestion level. Equation (9) shows the relation between different values of the reward and punishment parameters.

$$0 < a_3 < a_2 < a_1 < 1$$
$$0 < b_2 < b_1 < 1 \quad (9)$$

The automaton is rewarded and penalized according to (4) and (5), respectively.

The larger the value of "*number of lost packets*" the higher the congestion will be. In addition, the lower throughput of arrived packets at the sink indicates that the number of packets in the intermediate nodes' queue has been increased due to the network congestion. Thus, the queuing delay and packet loss rate are increased. Therefore, throughput is an efficient parameter in determining network congestion.

TABLE 3

RULES OF AUTOMATON REWARD AND PENALIZE RULES PLACED IN THE SINK

REWARD PARAMETER	REWARD RULES
a_1	**Rule 1: if** $(\Delta Ls^c(t)=0 \ \&\& \ \Delta Ls^n(t)=0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)\ge 0)$
a_1	**Rule 2: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)\ge 0)$
a_1	**Rule 3: if** $(\Delta Ls^c(t)>0 \ \&\& \ \Delta Ls^n(t)>0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)<0)$
a_2	**Rule 4: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)\ge 0)$
a_3	**Rule 5: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)\ge 0)$
a_3	**Rule 6: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)<0)$
a_3	**Rule 7: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)<0)$
a_3	**Rule 8: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)\ge 0)$
a_2	**Rule 9: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)<0)$
a_3	**Rule 10: if** $(\Delta Ls^c(t)<0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)<0)$
Punishment parameter	**Punishment Rules**
b_1	**Rule 11: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)<0)$
b_1	**Rule 12: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)\ge 0)$
b_2	**Rule 13: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)<0)$
b_2	**Rule 14: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)<0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)\ge 0)$
b_1	**Rule 15: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)<0 \ \&\& \ T^n(t)>0)$
b_1	**Rule 16: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)<0)$
b_1	**Rule 17: if** $(\Delta Ls^c(t)\ge 0 \ \&\& \ \Delta Ls^n(t)\ge 0) \&\& (\Delta T^c(t)\ge 0 \ \&\& \ T^n(t)\ge 0)$

By selecting an action, we expect to have a reduction in the number of lost packets and delay. If this does not realize, the automaton will penalize itself. Therefore, using the information obtained from the sink, the source finds out the optimal rate. This will lead to a maximum throughput and congestion avoidance. The critical patients get more bandwidth than the others. The pseudo-code of the LCCP protocol is given in Figure 3.

Procedure LCCP

- **Given:**
  - ✓ *Set of intermediate nodes' actions :* $\alpha^{N} \equiv \{\alpha_1^{N}, \alpha_2^{N}, ..., \alpha_r^{N}\}$
  - ✓ *Set of sinks' actions :* $\alpha^{s} \equiv \{\alpha_1^{s}, \alpha_2^{s}, ..., \alpha_m^{s}\}$
  - ✓ *Set of intermediate nodes' probabilities:* $P^{N}(n) \equiv \{P_1^{N}(n), P_2^{N}(n), ..., P_r^{N}(n)\}$
  - ✓ *Set of sinks' probabilities:* $P^{s}(n) \equiv \{P_1^{s}(n), P_2^{s}(n), ..., P_m^{s}(n)\}$
  - ✓ *random environment: network*

- ✓ *number of intermediate nodes' actions: r*
- ✓ *number of sinks' actions: m*
- ✓ *penalty value :* $\{b_1,...b_x\}$
- ✓ *reward value :* $\{a_1,...a_x\}$
- ✓ *number of lost packets between two successful deliveries at time t: Ls(t)*
- ✓ *queue length at time t: Q(t)*
- ✓ *packet loss in sink at time t: L(t)*
- ✓ *end to end delay at time t: D(t)*

- **Algorithms :**

1. Initialize the probability of selecting an action from the set of actions as follows:

1.1 In intermediate nodes:

$$P_i = \frac{1}{r} \quad i = 1...r$$

1.2 In sink node:

$$P_i = \frac{1}{m} \quad i = 1...m$$

Repeat

2. Pick up action $\alpha(n) = \alpha_i(n)$ according to $P(n)$
3. Calculate the network changes as follows:

3.1 In intermediate nodes:

$$\Delta L(t) = L(t) - L(t-1),$$
$$\Delta Q(t) = Q(t) - Q(t-1)$$

3.2 In sink node:

$$\Delta Ls(t) = Ls(t) - Ls(t-1),$$
$$\Delta D(t) = D(t) - D(t-1)$$

4. Compute the environment response (β) according to calculated values in step 3.
5. Update the probabilities according to environment response as follows :

if the response (β) is favorite:

$$P_i(n+1) = P_i(n) + a\left[1 - P_i(n)\right]$$
$$P_j(n+1) = (1-a)P_j(n) \quad \forall j \;\; j \neq i$$

else

$$P_i(n+1) = (1-b)P_i(n)$$
$$P_j(n+1) = (b/r - 1) + (1-a)P_j(n) \quad \forall j \;\; j \neq i$$

6.

6.1 In intermediate nodes:

Update the <u>node arrival rate</u> according to selected action by the automaton

6.1 In sink node:

Update the <u>source sending rate</u> according to selected action

end loop

Figure 3: Pseudo-code of LCCP protocol

LCCP has briefly the following characteristics:

- LCCP is a learning automaton based congestion control protocol that intelligently reduces congestion. Unlike the LACAS protocol[6], the number of rates in the proposed protocol is not limited. Current rate can be increased or decreased according to the congestion level and patient (source) condition. Therefore, the optimum rate can be achieved.
- The reward and penalty values are variable. These values are determined based on the congestion level. So, the automata can be learned with a better quality and even less consuming time.
- LCCP can select the appropriate source rate in order to achieve a higher throughput and less packet loss.
- LCCP tries to choose a suitable packet arrival rate in the intermediate nodes by prevent the queuing delay so that the end to end delay is to be reduced.
- In LCCP, unlike the LACAS protocol, patients have different priority based on their physiological conditions. Thus, the proposed protocol is able to provide more network bandwidths for transmission of data packets related to the vital signs from patients in *Critical* need.

5. Simulation Results

To evaluate the performance of the proposed LCCP protocol, we simulated a wireless biomedical sensor network including 3 different patients. We used OPNET simulator [12]. To simulate a real environment, the intermediate nodes' power consumption parameters values are choose the same as 802.15.4-compliant RF transceiver CC2430 [13]. The Proposed protocol implementation was done by using the 802.15.4 protocol of the MAC layer. In addition to the proposed protocol, the well known LACAS protocol[6] was also implemented. The simulations were run using LAF protocol [14] as routing protocol and CSMA protocol as MAC layer protocol. The nodes were randomly distributed in the environment. We considered the following two different modes.

- **Combined mode:** In this mode, two learning based AQM mechanisms and rate adjacent mechanism are used together to reduce congestion.
- **Split mode:** to show the advantage of the combining AQM and rate adjacent mechanism, we consider no adjustment mechanism and evaluate the performance of the proposed protocol using only the AQM mechanism.

We also considered the following two different scenarios. In the first scenario, we assume all end sensor nodes (the patients) are in *Normal* condition. In the second scenario, at the beginning of the simulation all the patients are in *Normal* condition. Patient1 changes its status to *Critical* condition between time t=40 sec. and t=120 sec. Since the *Critical* status means that the patient is in a serious condition, patient1 speeds up its sending rate too. As a result in second scenario there are some important packets that are injected to the network with higher rates and should be delivered to the sink with lower packet dropping ratios and delays compared to first scenario.

In order to assess the performance of the proposed AQM algorithm, the following parameters were used in the simulation.

- **Packet loss ratio** = total number of lost packets / total number of generated packets.
- **Energy loss ratio**= total number of lost packets / total number of received packets by the sink.
- **Delivery ratio**= total number of received packets by the sink/ total number of generated packets.

- **Throughput** = total number of received packets by the sink / time
- **Energy usage** = (initial energy – remaining energy) / initial energy
- **Source delivery ratio** = number of patient *i* packets received by sink

A. Impact Of joint AQM mechanism with rate adjustment mechanism (combined mode)

Congestion control can be applied inside nodes through the Active Queue Management mechanism. AQM can potentially reduce packet loss rate in the network. AQM together with adjusting traffic rate at source nodes can provide a better congestion control and a better QoS in terms of both packet loss rate and delay can be achieved.

TABLE 4
IMPACT OF THE SERVICE DIFFERENTIATION ON THE PERFORMANCE OF THE NETWORK IN COMBINED MODE

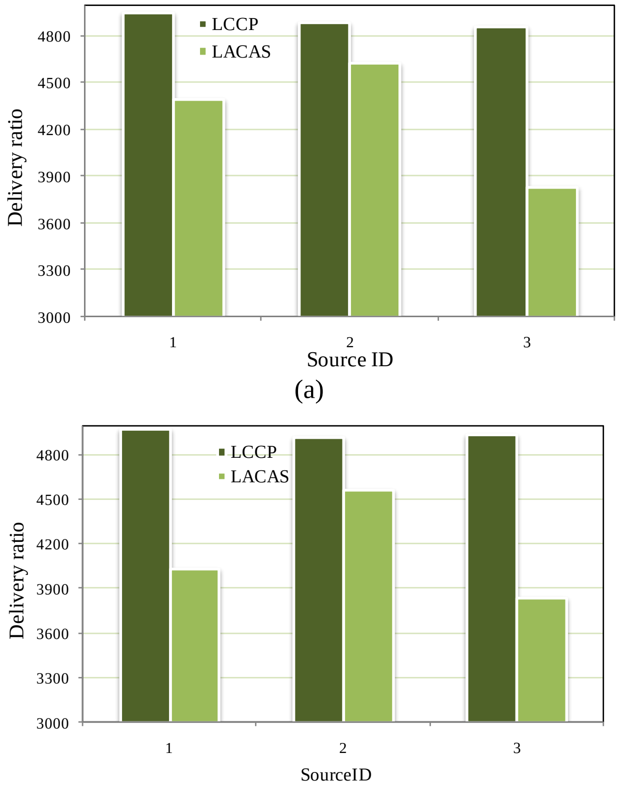
Scenario 1 (all patients are in *Normal* condition)

	LCCP			LACAS
	Total	**Critical Class**	**Normal Class**	**Total**
Average delay	1.9	-	1.9	9
Packet loss rate	0.02	-	0.02	0.14
Network energy loss ratio	0.024	-	0.024	0.15
Network delivery ratio	0.98	-	0.98	0.89

Scenario 2

	LCCP			LACAS
	Total	**Critical Class**	**Normal Class**	**Total**
Average delay	2.7	1.9	5	12
Packet loss rate	0.03	0.01	0.03	0.27
Network energy loss ratio	0.03	0.02	0.035	0.28
Network delivery ratio	0.97	0.99	0.97	0.83

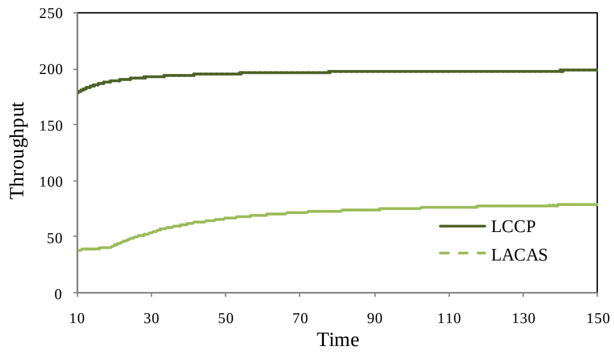
Table 4 shows the packet loss rate in the network. In sever congestion, the increase in channel load and queue length leads to a higher probability of loss in the intermediate nodes. Therefore, the number of accepted packets in the node is decreased. As a result, the delivery ratio is decreased. As seen in Table4, the number of *Critical* lost packets is less than that of the other class. A comparison of the loss in LCCP and LACAS protocols will show that the LCCP protocol has a much lower level of loss than the LACAS protocol. This is because the joint AQM mechanism and learning automaton-based mechanism in the sink which efficiently adjusts the node arrival rate and source transmission rate and avoids congestion and loss based on the feedback received from the network. This indicates the adaptability of the proposed AQM mechanism in the intermediate nodes with the congestion control protocol in the sink. Although the LACAS protocol adjusts the node arrival rate, the results shows the LCCP protocol is more successful in choosing the optimal rate. Table 4 also shows the network performance of both protocol for scenario1 and scenario2. It can be seen that the performance of LCCP protocol is better than that of the LACAS protocol. Obviously, shorter queue lengths will cause shorter packet queuing delay. *Critical* packets are delay intolerant so these packets have a higher priority in entering and exiting the node, so they reach their destinations faster. Using a proper AQM mechanism which estimates the arrival rate based on queue length and channel load along with an automaton-based mechanism that determines source transmission rate will keep the queue length constant at a desirable size and avoid its sudden expansion.



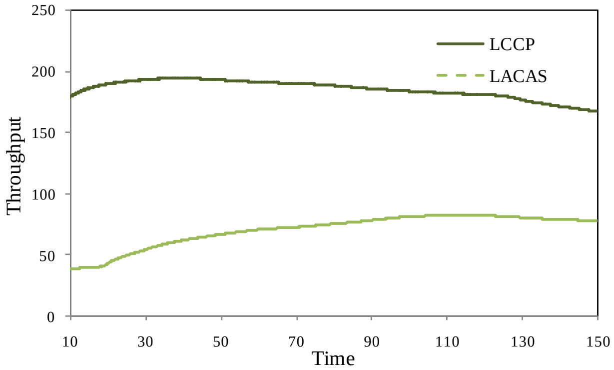
(b)

Figure 4: Source delivery in Combined mode a)Scenario 1 b)Scenario 2

Figure 4 shows the source node delivery ratio in both scenario1 and scenario2. As it is shown in this figure, the number of lost packets in LCCP is less than that of the LACAS protocol which is a result of using joint AQM and rate adjustment protocols. Consequently, in LCCP, the number of received packets increases. In the scenario2 mode, a great number of packets enter sensor nodes between time t=40 sec. and t=120 sec. Hence, the queues are filled quickly and the probability of loss is increased. In this mode, the number of lost packets in the intermediate nodes becomes higher in comparison with the scenario2 mode. Thus, due to increase in packet loss ratio, the network experiments decrease in throughput with increased offered load.



(a)

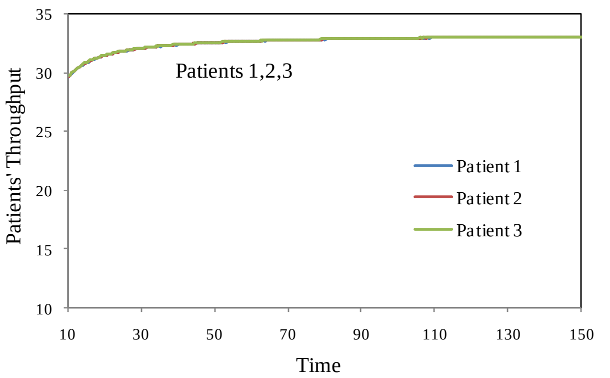


(b)

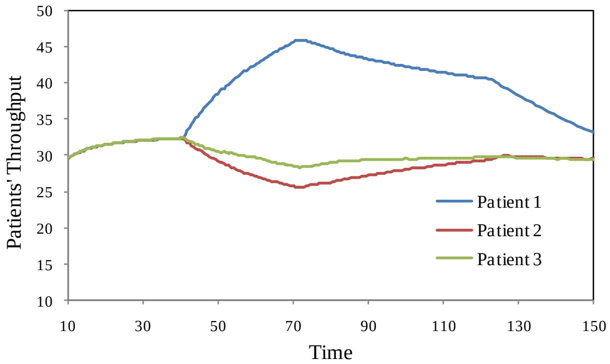
Figure 5: Total throughput in Combined mode a)Scenario 1 b)Scenario 2

Figure 5 plots the network throughput of both LACAS and LCCP protocols. Network throughput is the average rate of successful packet delivery. Source reduces its rate due to packet loss avoidance. Reducing transmission rate should not lead to a network throughput reduction.

We can observe from figure 6 that the proposed protocol can assign network bandwidth to each traffic class based on its priority. As shown in figure 6(a), for the proposed protocol, when all patients are in the *Normal* condition the network throughput is shared equally between the patients. In scenario 2 (figure 6(b)) when patient1 went to the *Critical* condition (during time interval [40s, 120s]), the system assigned more bandwidth to patient1. Therefore, Patient1's rate is increased. During this time interval there is a decrease in bandwidth assignment to the other patients. Unlike the proposed protocol, the LACAS protocol is not able to detect this change in patient condition, and hence could not adjust its bandwidth allocation to the patient in need.

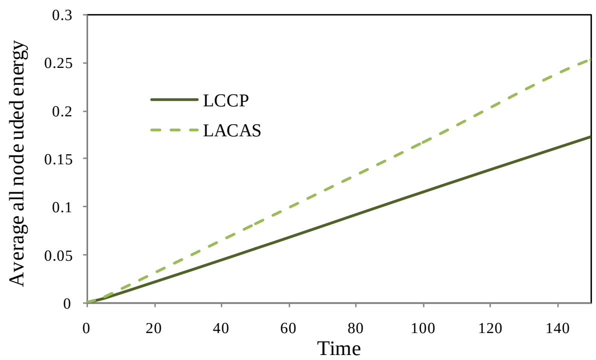


(a)

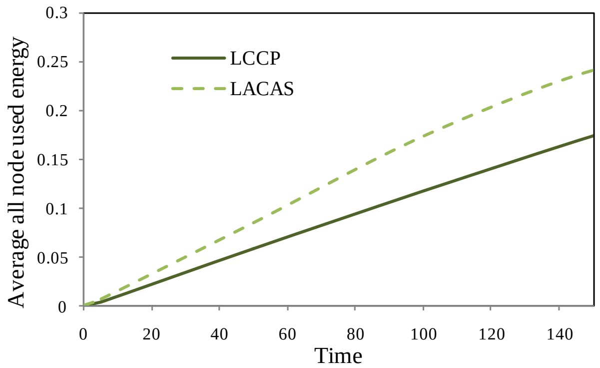


(b)

Figure 6: LCCP Patients' throughput in Combined mode a) Scenario 1 b) Scenario 2



(a)



(b)

Figure 7: Average nodes' used energy in combined mode a)Scenario 1 b)Scenario 2

Figure. 7 illustrates the average energy consumption of nodes. Collision is a major source of energy waste in a MAC protocol for wireless sensor networks. When a transmitted packet is corrupted it has to be discarded, and be retransmitted. The retransmission increase energy consumption. The goal of the proposed protocol is to adjust the sending rate and nodes arrival rate at each node properly, in order to improve the lifetime of WBSNs and alleviation congestion In the LCCP protocol, some messages are sent to the source from sink. Though it seems that in the proposed protocol, a node may consume more energy but since LCCP can control congestion better than LACAS, it consumes a lower energy due to a lower collision probability than LACAS.

B. Impact of AQM mechanism with no rate adjustment mechanism (split mode)

In the first section, we evaluate the impact of joint AQM and rate adjustment mechanism on the performance of the network. Thus in the following experiments we consider no source rate adjustment mechanism and compare it with the LACAS protocol to show the advantage of the proposed learning based AQM mechanism with the LACAS protocol. Moreover, the advantage of using rate adjustment mechanism together with AQM management is shown. In the following results only scenario 2 is considered.

TABLE 5
IMPACT OF THE SERVICE DIFFERENTIATION ON THE PERFORMANCE OF THE NETWORK IN SPLIT MODE

Scenario 2				
	LCCP			LACAS
	Total	**Critical Class**	**Normal Class**	**Total**
Average delay	5.5	3.2	10	12
Packet loss rate	0.04	0.022	0.05	0.27
Network energy loss ratio	0.04	0.023	0.051	0.17
Network delivery ratio	0.96	0.98	0.95	0.83

Table 5 shows the performance of the network in scenario2 with no rate adjustment mechanism. Comparing the results of tables 4 and table 5, the advantage of using source rate adjustment is obvious. Queuing delay is a function of the number of packets in the queue. In a high congestion, intermediate node's queue lengths increase suddenly so that the number of drop packets and delay are increased. Obviously, larger queue lengths will cause larger packet queuing delay. Therefore, other mechanisms such as source rate decrease are required to reduce congestion. The proposed automata based AQM mechanism has a better performance compared to the LACAS protocol. This is because of using proper learning parameters and appreciate action list.

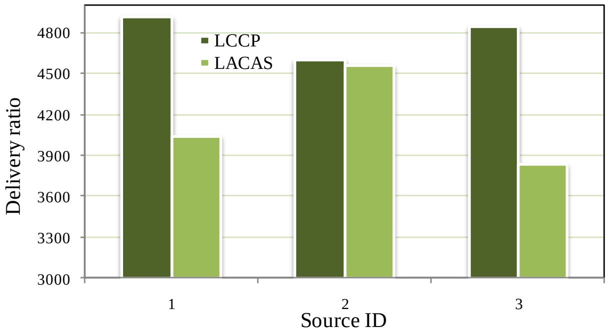


Figure 8: Source delivery in Scenario 2 in Split mode

Comparing figure 8 with figure 4(b), a high packet loss rate has been experienced in the Split mode. Although patient1 has a lower packet loss ratio due to its *Critical* status, the overall delivery ratio in the "Split mode" is higher than that of the "Combined mode". The rate adjustment mechanism controlling the load applied to the network so that congestion (*packet losses*) has a little probability to occur.

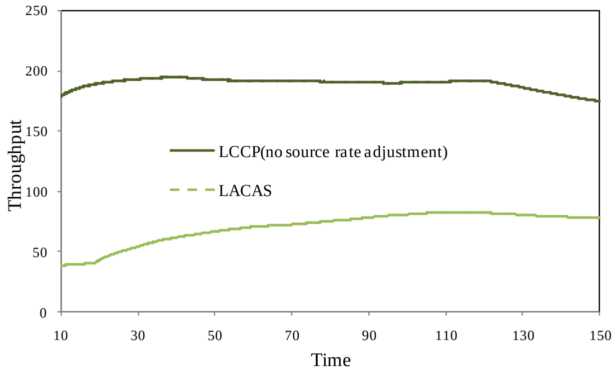


Figure 9: Total throughput in Scenario 2 in Split mode

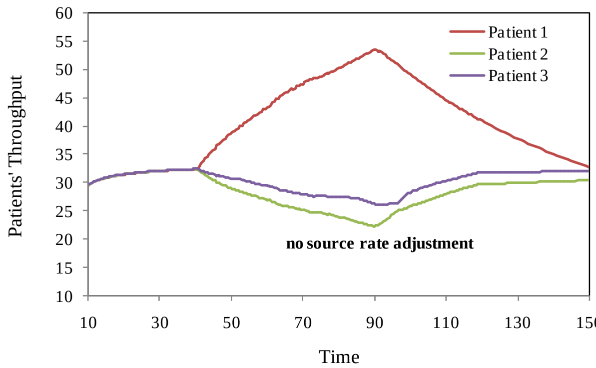


Figure 10: LCCP Patients' throughput in Scenario 2 in Split mode

Packet loss is one of the reasons why data *throughput* is reduced. Rate adjustment mechanism increases the transmission rate when the network becomes underutilized and decreases the transmission rate when the network becomes over utilized. Comparing figure 9 with figure 5(b), and also comparing figure 10 with figure 6(b), although "Split mode" has a little higher throughput than that of the joint mechanism, it experience a higher packet loss too. The objective is to achieve a certain throughput without incurring a high packet loss.

6. Conclusion

In this paper, we presented a congestion control protocol based on learning automaton for multi traffic healthcare applications in WBSNs. In healthcare monitoring systems, some of philological signals are more important than the others, and thus the need to be sent as quickly as possible to the central monitoring system. The proposed congestion control protocol namely, LCCP can adjust intermediate node arrival rate and source rate using learning automata. The proposed protocol is aimed at satisfying all the requirements of different types of traffic. To do so, two different traffic classes namely *Critical* and *Normal* were considered. In order to control congestion, a mechanism based on the learning automaton has been placed in the sink. At each intermediate node that gathers the patient's physiological data the sensed data are grouped into different classes. Using weighted scheduling mechanisms, higher priority classes are given better quality of service and more bandwidth than the lower priority classes. The simulation results indicate that the proposed protocol, by adjusting source rate, avoids loss caused by congestion. Furthermore, it has been shown that LCCP achieves higher performance and lower packet loss than LACAS.

7. References

[1] H. L. Ren, M. Q. H. Meng, and X. J. Chen, "Physiological Information acquisition through wireless biomedical sensor networks", in Proc. 2005 IEEE International Conference on Information Acquisition.

[2] G.Zhou, J. Liu, C. Wan, M. Yarvis, and J. Stankovic, "BodyQoS: Adaptive and Radio-Agnostic QoS for Body Sensor Networks", in Proc. 2008 Infocom.

[3] M.H. Yaghmaee, and D. Adjeroh, "Priority-based rate control for service differentiation and congestion control in wireless multimedia sensor networks". Computer network Journal, pp. 1798-1811,2009.

[4] J. Paek, and R. Govindan, "RCRT: Rate-Controlled Reliable Transport for Wireless Sensor Networks", in Proc. 2007 ACM Conference on Embedded Networked Sensor Systems (Sensys).

[5] M.H. Yaghmaee, and D. Adjeroh, "A Novel Congestion Control Protocol for Vital Signs Monitoring in Wireless Biomedical Sensor Networks," in Proc. 2010 WCNC, pp. 1-6.

[6] S. Misra, V. Tiwari, and M.S. Obaidat, "LACAS: Learning Automata-Based Congestion Avoidance Scheme for Healthcare Wireless Sensor Networks," IEEE Journal on Selected Areas in Communications, vol. 27, no. 4, pp. 466-479, 2009.

[7] R. Gunasundari, R. Arthi, and S. Priya, "An Efficient Congestion Avoidance Scheme for Mobile Healthcare Wireless Sensor Networks," Int. J. Advanced Networking and Applications, vol. 2, no,3, pp. 693-698, 2010.

[8] S. Kim, "Study on the Dynamic QoS Provisioning Scheme for U-Healthcare over WMSAN," Journal of Convergence Information Technology, vol. 6, no. 2, pp. 1-8, 2011.

[9] F. Hu, Y. Xiao, and Q. Hao, "Congestion-aware, loss-resilient biomonitoring sensor networking for mobile health applications." IEEE J. Selected. Areas Communication, vol. 27, no. 4, pp. 450–465, 2009.

[10] N. Farzaneh, M.H. Yaghmaee, "Joint Active Queue Management and Congestion Control Protocol for Healthcare Applications in Wireless Body Sensor Networks", In Procc. 9th International Conference on Smart Homes and Health Telematics(ICOST), doi: 10.1007/978-3-642-21535-3_12, 2011.

[11] K.S. Narendra, and M.A. Thathachar, "Learning Automaton: An Introduction," Prentice Hall, 1989.

[12] http://www.opnet.com/

[13] CC2430 Preliminary Data Sheet (rev. 2.1) SWRS036F, Jun. 2007, Chipcon Products from Texas Instruments

[14] H. Sabbineni, K. Chakrabarty, "Location-aided floooding: An energy efffcient data dissemination protocol for wireless sensor networks," IEEE Trans. on Computers, vol. 51, no. 1, pp. 36-46, 2005.