

RESEARCH

Open Access

BOB-RED queue management for IEEE 802.15.4 wireless sensor networks

Mu-Sheng Lin^{1*}, Jenq-Shiou Leu¹, Wen-Chi Yu¹, Min-Chieh Yu¹ and Jean-Lien C Wu²

Abstract

Multimedia services over resource constrained wireless sensor networks (WSNs) face a performance bottleneck issue from the gateway node to the sink node. Therefore, the queue management at the gateway node is crucial for diversified messages conveyed from the front nodes to the sink node. In this article, beacon order-based random early detection (BOB-RED) queue management is proposed. BOB-RED is a dynamic adaptation scheme based on adjusting beacon interval and superframe duration in the IEEE 802.15.4 MAC superframe accompanied with RED queue management scheme to increase the transmission efficiency of multimedia over WSNs. We focus on the performance improvement upon different traffic loads over WSNs. Evaluation metrics include end-to-end delay, packet delivery ratio, and energy consumption in IEEE 802.15.4 beacon enabled mode. Simulation results show that BOB-RED can effectively decrease end-to-end delay and energy consumption compared to the DropTail scheme.

Keywords: wireless sensor networks (WSNs), IEEE 802.15.4, superframe, beacon-enabled, beacon order (BO), superframe order (SO), queue management, DropTail, random early detection (RED)

1. Introduction

IEEE 802.15.4 standard [1] defines the protocol and interconnection of devices via radio communication in a wireless personal area network (WPAN). The standard uses CSMA/CA medium access mechanism and supports star as well as peer-to-peer topologies. It provides applications such as home entertainment and control, security alarms, industrial monitoring and control, personal mobile healthcare and tele-assist, etc. Two types of device called the full function device (FFD) and the reduced function device (RFD) are used in a LR-WPAN network. FFD is a fully functional device which can be a PAN coordinator, a coordinator, or just a device. RFD is a device with reduced functionality which can only function as an end device. It cannot communicate with any other device in addition to coordinator. We talk about the PAN-coordinator, which acts as a coordinator for the entire WPAN. It is authorized to provide synchronization services in an established network.

Little researches study transmission image or video over IEEE 802.15.4 networks [2-6]. The CMUcam project provides simple vision capabilities to small embedded systems in the form of an intelligent sensor. The CMUcam3 extends upon this idea by providing a flexible and easy to use open source development environment that complements a low cost hardware platform [7]. It can be used for environment surveillance, robotics, interactive toys, or object recognition and tracking. Traffic loads on multimedia services over resource constrained wireless sensor networks (WSNs) sometimes are huge and bursty. Transmission of image or video data requires careful handling to ensure that end-to-end delay is within acceptable range. So, queue management algorithm on a gateway node should allow temporary bursty traffic and prevent high delay. Up to now, there are many popular queuing management algorithms and packet scheduling mechanisms are proposed. As popular known, various scheduling mechanisms such as round-robin (RR), weighted round-robin (WRR), or weighted fair queuing (WFQ) are different from queue managements. Scheduling schemes focus on the sequence and timing of packet transmission, and queue management is obviously about managing queues in

* Correspondence: amoolsin@gmail.com

¹Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C

Full list of author information is available at the end of the article

forwarding devices such as router or relay node. Queue managements can separate into passive queue management (PQM) and active queue management (AQM). DropTail can be classified as a PQM algorithm since it is basically a simple first in first out (FIFO) mechanism where packets are dropped when queue length exceed buffer length. Random early detection (RED) is the most commonly used queue management algorithm in AQM class [8,9]. Patel et al. [10] proposed comparison between two queuing management system RED and DropTail for avoiding the congestion in high speed packet switched networks. Epiphaniou [11] focuses on three different mechanisms, namely, DropTail (FIFO), RED, and DiffServ, and their effects on realtime voice traffic. Flow RED is an extended version of RED. It behaves just like RED, but maintains per-flow states for all flows in the gateway node. Using this per-flow state, FRED preferentially drops the packets of flows that have queue sizes larger than the average per-flow queue size [12]. Deficit Round Robin is a variant of WFQ discipline. It allows WFQ to handle variable packet sizes in a fair manner. It guarantees nearly perfect fairness for flows that have at least one packet in the router buffer. Longest queue drop is used as a packet drop strategy [13]. Ali Ahammed and Banu analyze the performance of AQM algorithms including FRED, BLUE, SFB, and CHOKe. They aim at a thorough evaluation among these algorithms and illustrations of their characteristics by simulation [14]. Le et al. [15] evaluate the performance of their analytical on $M/M/1$ queuing model for IEEE 802.15.4 non-beacon-enabled mode at the 2.4 GHz in NS-2 network simulator. Mistic and Mistic [16] analyze the performance of a personal area network operating under the IEEE Standard 802.15.4 in the beacon-enabled mode, and derive the probability distribution of packet access delay and calculate the throughput. Buratti [17] proposed a mathematical model for the beacon-enabled mode of IEEE 802.15.4. Gao and He [18] proposed an individual beacon order adaptation (IBOA) algorithm for IEEE 802.15.4 networks, which can individually adapt the beacon interval (BI) and duty cycle of each node at the same time to the node's individual performance requirements. Jeon et al. [19] proposed a new duty-cycle adaptation algorithm for IEEE 802.15.4 beacon-enabled networks. They modify the reserved frame control field in the MAC layer header to deliver the end device's buffer occupancy and queuing delay.

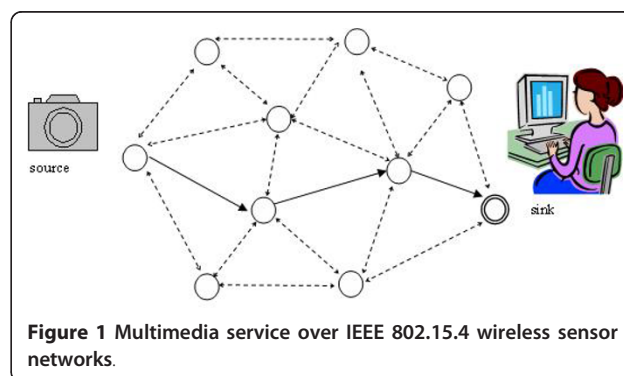
Despite many researches study about the applications of 802.15.4 on industrial and MAC protocol, but little attention in the past has been given to how different queuing management algorithms, such as DropTail and RED, perform in terms of different settings of beacon order (BO), superframe order (SO) parameters on the multimedia service over IEEE 802.15.4 WSNs. As per

our literature search, only a few studies have so far analyzed the impact of BO, SO value on IEEE 802.15.4 operation. But, none article invested RED queue management accompanied with BO, SO value for IEEE 802.15.4 WSNs. Moreover, most current gateway nodes use DropTail as a queue management scheme, which does not guarantee fairness and delay bound. There has been no motivation for realtime applications to use end-to-end congestion avoidance mechanisms for IEEE 802.15.4 WSNs. We study the performance evaluation of different queue management algorithms, such as DropTail, RED on gateway node in this article.

In this study, we focus on the performance improvement upon different traffic loads over WSNs. Figure 1 shows a typical multimedia application over WSNs that a front sensor node equipped with a camera sends an emergent message containing the surveillance image to the sink node once it detects an urgent or intrusive event. Different traffic types demand different packet delivery ratios and end-to-end delays. Urgent messages always have a first priority with a minimum end-to-end delay. On the other hand, keep-alive messages carrying small periodic data have a comparatively low priority which can tolerate a longer delay. Hence, a dynamic adaptation scheme based on adjusting BI and superframe duration (SD) in the IEEE 802.15.4 MAC superframe is proposed to increase the transmission efficiency of multimedia over WSNs.

This article presents beacon-order based RED (BOB-RED) queue management for congestion avoidance in IEEE 802.15.4 WSNs. The proposed scheme consists of a virtual threshold function, a dynamic adjusted per-flow drop probability, a dynamic modification of BO and SO strategy that decrease end-to-end delay, energy consumption, and increase throughput when there are different traffic type flows through the gateway node.

A comprehensive simulation for the proposed scheme using the NS-2 network simulator is also presented in this study. Evaluation metrics include end-to-end delay, packet delivery ratio, and energy



consumption using the beacon-enabled and non-beacon-enabled mode in IEEE 802.15.4 WSNs. Distinct network topologies like star, tree, and chain architectures comprising one coordinator and some stationary nodes are considered in the simulation. Simulation results show that a suitable queuing management scheme accompanied with an appropriate setting of BO, SO can effectively achieve a better performance. If the BO value is fixed, a smaller SO value would incur a higher end-to-end delay and a lower packet delivery ratio. If the BO value is equal to the SO value, a larger BO can achieve a higher packet delivery ratio and a lower average end-to-end delay. Besides, the BOB-RED queue management scheme can decrease end-to-end delay compared to the DropTail scheme.

The remainder of the article is structured as follows: in Section 2 we introduce IEEE 802.15.4 MAC superframe structure and the concepts of RED. In Section 3, we describe the BOB-RED algorithm in detail. In Section 4, we describe network configuration and assumptions. In Section 5, we present the results of our simulations. Finally, in Section 6, we present our main conclusions and suggest a number of areas for further study.

2. IEEE 802.15.4 MAC superframe structure and RED

2.1 IEEE 802.15.4 MAC superframe structure

IEEE 802.15.4 MAC protocol supports the beacon-enabled and non-beacon-enabled modes. In beacon-enabled mode, the access to the channel is managed through a superframe, starting with the beacon packet transmitted by the PAN coordinator. The superframe is subdivided into a contention access period (CAP), contention-free period (CFP), and an inactive part. Nodes in CAP use a slotted CSMA/CA to contention for channel and CAP containing a number of GTTs that can be allocated by the PAN coordinator to specific nodes. In the non-beacon-enabled mode, there are no regular beacons, but the coordinator may unicast beacons to a soliciting device. Communication among devices in the non-beacon-enabled mode uses unslotted CSMA/CA for decentralized access. This article considers CAP and inactive period only when the beacon-enabled mode is used. Figure 2 shows the IEEE 802.15.4 MAC superframe structure [20].

The structure of this superframe is described by the values of macBeaconOrder (BO) and macSuperframeOrder (SO). The MAC PIB attribute macBeaconOrder describes the interval at which the coordinator shall transmit its beacon frames. The superframe order is the variable which is used to determine the length of the SD which is divided into 16 time slots. Similarly, the BI is determined by the variable BO.

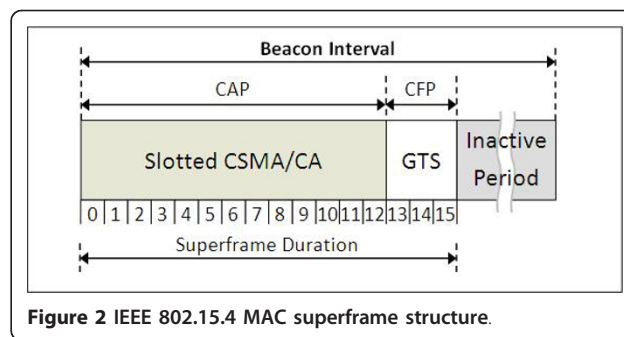


Figure 2 IEEE 802.15.4 MAC superframe structure.

Since the time of the SD cannot exceed the time of a BI, the condition for both parameters is $0 \leq SO \leq BO \leq 14$. When BO is greater than SO, it indicates that there is an inactive portion present in the superframe. Also for $SO = BO$, the BI is same as the SD indicating there is no inactive portion.

The values of macBeaconOrder, BO, and the BI are related as follows:

$$BI = aBaseSuperframeDuration \times 2^{BO}$$

$$SD = aBaseSuperframeDuration \times 2^{SO}$$

- BaseSuperframeDuration = 960 symbols = 15.36 ms, each time slot has a duration of $15.36/16 = 0.96$ ms.
- Non-beacon-enabled mode: $BO = SO = 15$. If $BO = 15$, the coordinator will not transmit beacon frames except when requested to do so, such as on receipt of a beacon request command. The value of macSuperframeOrder shall be ignored if $BO = 15$.

Three topologies are proposed in IEEE 802.15.4 protocol standard. In a star topology, data transmission can from a device to the coordinator or from the coordinator to the device (Figure 3) [1].

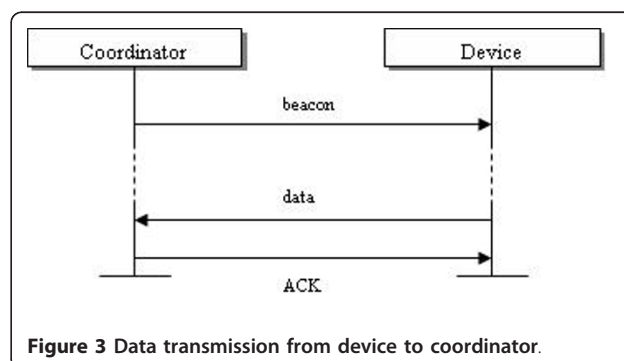


Figure 3 Data transmission from device to coordinator.

Following shows part of NS2 trace file in the beacon-enabled mode (BO = SO = 3). Node 0 is the coordinator and node 1 is an FFD that sends constant bit rate (CBR) traffic to node 0. We set the values of BO, SO equal to 3 so the value of BI is $15.36 \text{ ms} \times 2^3 = 122.88 \text{ ms}$. From trace file, we can observe node 0 sends BCN (beacon) every 122.88 ms (1.775232000 - 1.652352000).

```
s 1.652352000_0_MAC-0 BCN 12 [0 ffffffff 0 0]
s 1.775232000_0_MAC-0 BCN 12 [0 ffffffff 0 0]
s 1.789120000_1_MAC-0 CM1 17 [0 0 1 0]
r 1.789856033_0_MAC-0 CM1 17 [0 0 1 0]
s 1.790272000_0_MAC-0 ACK 5 [0 1 0 0]
r 1.790624033_1_MAC-0 ACK 5 [0 1 0 0]
s 1.898112000_0_MAC-0 BCN 20 [0 ffffffff 0 0]
```

In the tree topology, data transmission can from a device to the coordinator or from the coordinator to the device. Following shows node 1 sends an association request (CM1) to node 0.

```
s 1.789120000_1_MAC-0 CM1 17 [0 0 1 0]
r 1.789856033_0_MAC-0 CM1 17 [0 0 1 0]
s 1.790272000_0_MAC-0 ACK 5 [0 1 0 0]
r 1.790624033_1_MAC-0 ACK 5 [0 1 0 0]
.....
s 2.282144033_1_MAC-0 CM4 16 [0 0 1 0]
r 2.282848067_0_MAC-0 CM4 16 [0 0 1 0]
s 2.283072000_0_MAC-0 ACK 5 [0 1 0 0]
r 2.283424033_1_MAC-0 ACK 5 [0 1 0 0]
s 2.283712000_0_MAC-0 CM2 25 [0 1 0 0]
r 2.284704033_1_MAC-0 CM2 25 [0 1 0 0]
s 2.284896033_1_MAC-0 ACK 5 [0 0 1 0]
r 2.285248067_0_MAC-0 ACK 5 [0 0 1 0]
```

After that node 0 receives the request and sends back an ACK. Connection is established. Then node 1 sends a data request (CM4) and node 0 sends an ACK. Node 0 sends an association response (CM2) and node 1 sends back an ACK [1] (Figure 4).

2.2 Random early detection

RED, also known as random early discard or random early drop, is an AQM algorithm. The operation of RED queue management is shown in Figure 5.

When packets income, it calculates their current-occupied average queue length (Avr).

It estimates the average queue size as follows.

$$Avr = (1 - w_q) \times Avr + w_q \times q$$

where q is the instantaneous queue size, w_q is the time constant of the low pass filter.

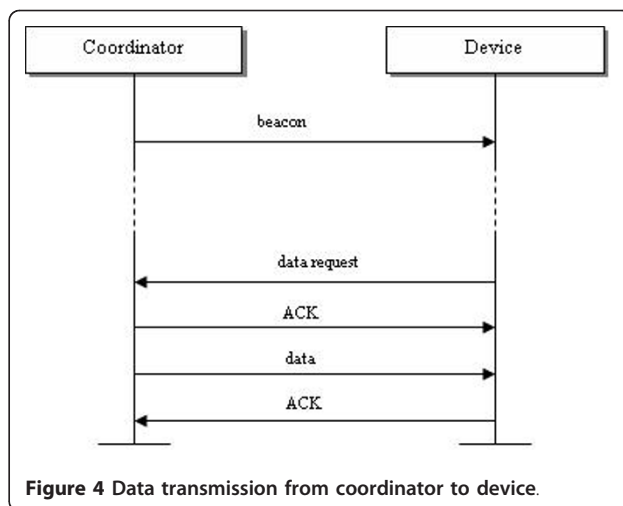


Figure 4 Data transmission from coordinator to device.

(2) If the Avr is smaller than MinThreshold (MinThres), the packet will be kept and sent to the queue waiting for the transmission.

(3) If the Avr is longer than MaxThreshold (MaxThres), all packets will be dropped.

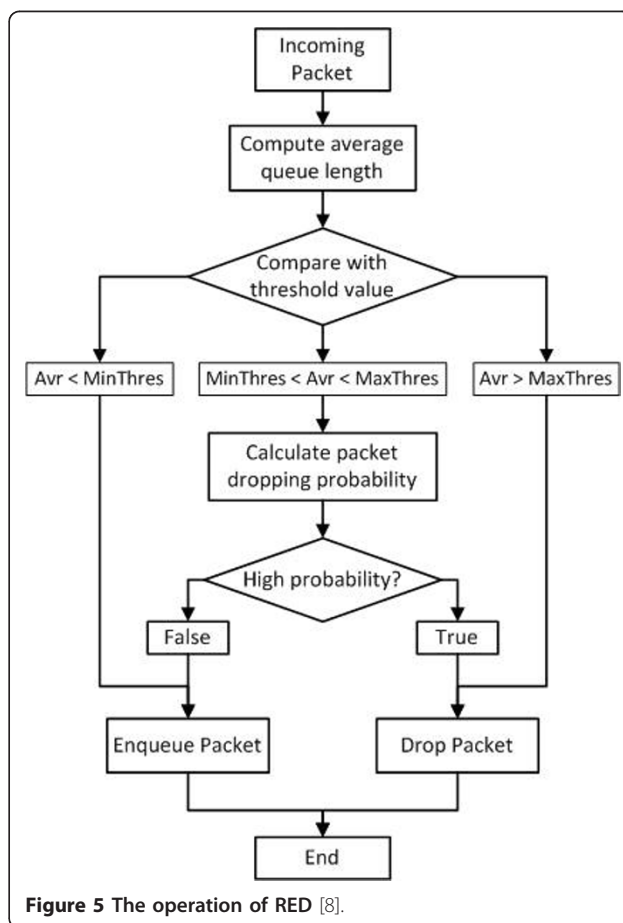


Figure 5 The operation of RED [8].

(4) If the Avr is between MinThres and MaxThres, the initial packet drop probability (P_b) of the packet will be a linear function of a number between 0 and Max_p (default value of max. packet drop probability).

$$P_b = Max_p \times (Avr - MinThres) / (MaxThres - MinThres)$$

The actual probability (P_a) is a function of the P_b and count of the number of packets enqueued since the last packet was dropped. $P_a = P_b / (1 - count \times P_b)$

Figure 6 shows the drop probability of RED. RED's performance is highly dependent on the settings of its control parameters. We choose several control parameters offered by NS2, which includes $qlen$, Max_p , Min_{th} , Max_{th} and q_w that users may according the requirements to adjust RED's performance. However, the impact of the individual parameter on the queue's performance is dependent on the others too. A set of parameters are listed as below.

1. $qlen$: queue length.
2. Max_{th} : maximum threshold for queue, $Max_{th} = qlen / 2$.
3. Min_{th} : minimum threshold for queue, $Min_{th} = Max_{th} / 3$.
4. Max_p : maximum value for P_b , $Max_p = 2 * packet_loss_rate$. NS-2 default value is 0.1.
5. q_w : queue weight, NS-2 default value is 0.002.

RED is the simplest queue management and comprehensively used in most routers. Despite RED algorithm is designed to accompany a transport-layer congestion control protocol such as TCP in IP Networks. We try to apply RED mechanism in IEEE 802.15.4 WSNs.

3. BOB-RED algorithm

3.1 Description of BOB-RED

Figure 7 shows the network topology with real-time traffic and non-real-time traffic in multihop IEEE

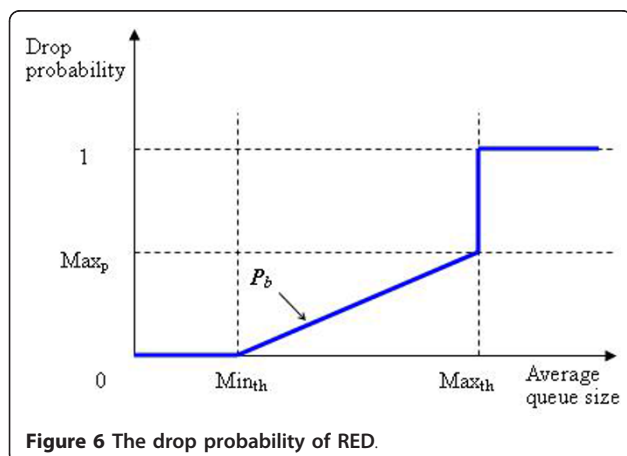


Figure 6 The drop probability of RED.

802.15.4 WSNs. Gateway node collects all the data from relay nodes and sends them to sink. Taking the IBOA [18] and DCA [19] for references, the BOB-RED adapts BO, SO of each gateway node individually to meet the needs of each gateway node working in a WSN. Unlike a unique BO, SO applied to all of the nodes, the BOB-RED assigns BO_i , SO_i for each neighbor node($i \in [1, N]$, N is the number of nodes working in the network) which nearby the coordinator.

BOB-RED algorithm can also be applied to large number of sensor nodes because it only runs on coordinator or gateway nodes. Multiple gateway nodes forward packets from end devices hop-by-hop to coordinator. Even though in a large-scale sensor networks, only a few relay nodes which around the coordinator or gateway nodes can transfer data directly to them. Owing to the gateway nodes must be FFDs in IEEE 802.15.4 WSNs, every gateway node can dynamic adapt the BO, SO values. If we apply BOB-RED algorithm to all gateway nodes, we must consider whether gateway nodes can communicate with the coordinator. Rapidly or continually adapt their BO, SO values may incur the link broken. In addition, large hop counts will increase large loss rate in our simulation. For simplify the complexity, only one gateway node is implemented in all our simulations.

The flowchart of BOB-RED algorithm is shown in Figures 8 and 9.

The operation of BOB-RED algorithm is very similar to RED queue management. Figure 10 shows the state transition diagram of BOB-RED. We describe the states of a gateway node as follows:

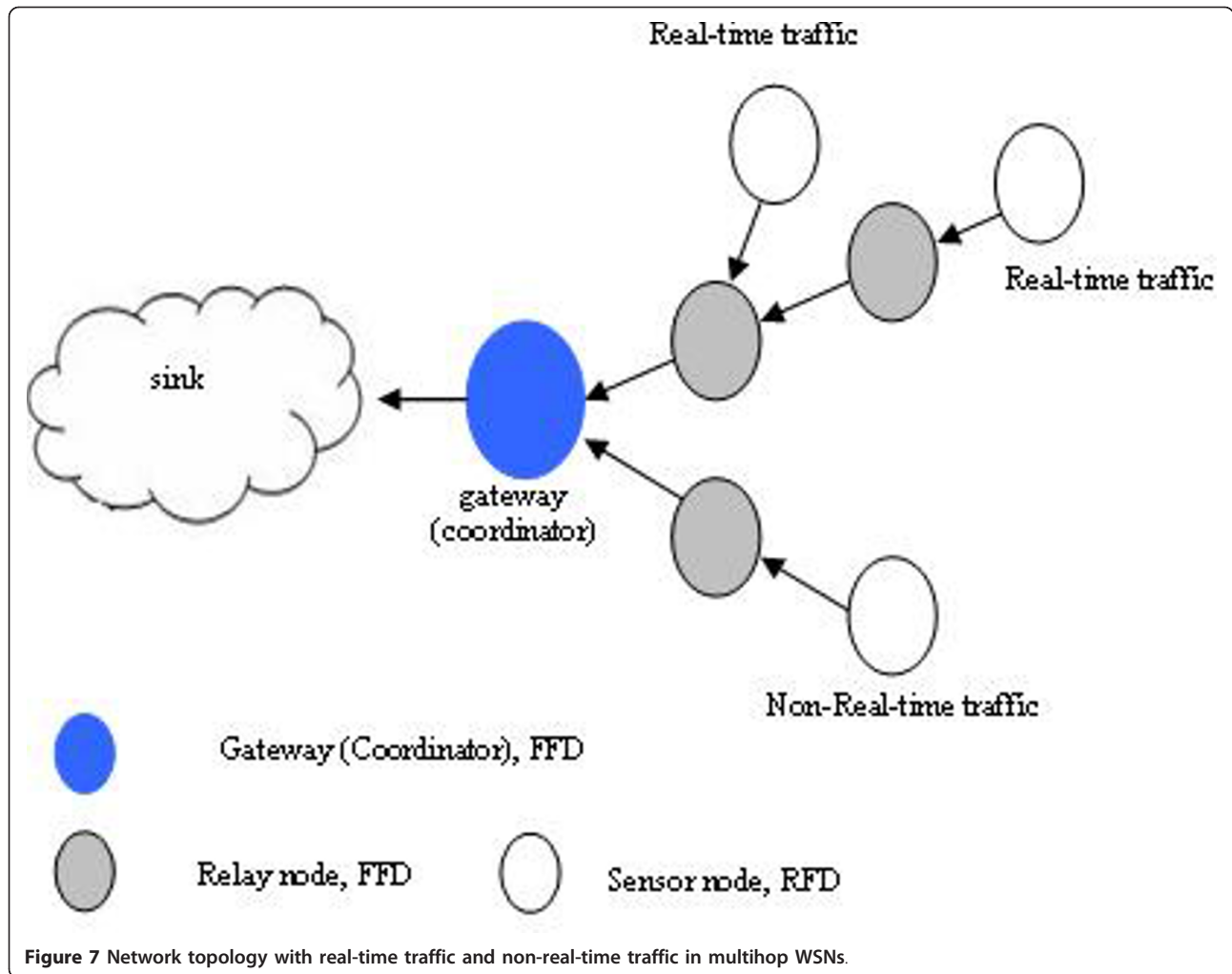
State 0: The coordinator first sets the initially min_{th} , max_{th} , max_p , q_w , BO and SO values. If we set BO and SO values as 3, then the coordinator begins to broadcast beacons. The gateway node calculates the average queue length avg_q .

State 1: If the avg_q is smaller than min_{th} , then BO and SO values decrease 1. If the avg_q is between min_{th} and k , the BOB-RED moves to State 2. If the BO and SO values are less than or equal to the BO_{min} , SO_{min} , the BOB-RED moves to State 3.

State 2: If the avg_q is between min_{th} and k , BO and SO values increase 1. If the avg_q is less than min_{th} , the BOB-RED moves to State 1. If the BO and SO values are greater than or equal to the BO_{max} , SO_{max} , the BOB-RED moves to State 4.

State 3: In this state gateway node operates using BO_{min} , SO_{min} values. If the avg_q is greater than min_{th} , the BOB-RED moves to State 2.

State 4: If the avg_q is greater max_{th} , all of packets will be dropped. If BO is still bigger than BO_{min} , the MAC decreases BO and SO values by 1, and adds the update information on BO and SO values to the



next beacon. If BO , SO values has reached BO_{max} , SO_{max} , the BOB-RED returns to State 1.

Figure 11 shows that the buffer size B is divided into four regions by thresholds min_th , k , max_th , where $min_th, k, max_th < B$. When packet arrives, the gateway node computes the average queue length. When the average queue length exceeds a preset threshold k , the gateway drops or marks each arriving packet with a certain probability, where the exact probability is a function of the average queue length. P_{ij} is the dropping probability, where i and j denote the number of real-time and non-real-time packets in the buffer, respectively.

The value of P_{ij} can be dynamically calculated based on the number of rear-time and non-real-time packets in the queue, i.e.,

$$P_{ij} = \begin{cases} 0 & \\ (i+j) - k + 1 & \\ \max_th - k + 1 & \end{cases} \quad (1)$$

where $P_{loss, rt}$ denotes the drop probability of real-time packet and $P_{loss, nrt}$ denotes the drop probability of non-real-time packet, respectively.

Drop probability of real-time packet lists as in Equation 2.

$$P_{loss, rt} = \begin{cases} 0 & avg_q < min_th \\ 0 & min_th \leq avg_q < k \\ \sum_{i=0}^B \frac{\lambda_{rt}}{\lambda_{rt} + \lambda_{nrt}} p_{i, B-i} & k \leq avg_q < max_th \\ 1 & max_th \leq avg_q \leq B \end{cases} \quad (2)$$

Drop probability of non-real-time packet lists as in Equation 3.

$$P_{loss, nrt} = \begin{cases} 0 & avg_q < min_th \\ \sum_{i=0}^B \sum_{j=0}^{B-i} \frac{(1 - \alpha_{i,j}) \lambda_{nrt}}{\lambda_{rt} + \lambda_{nrt}} p_{i,j} & min_th \leq avg_q < k \\ P_{ij} & k \leq avg_q < max_th \\ 1 & max_th \leq avg_q \leq B \end{cases} \quad (3)$$

Table 1 lists the dropping strategy with real-time and non-real-time traffics under different buffer occupancy.

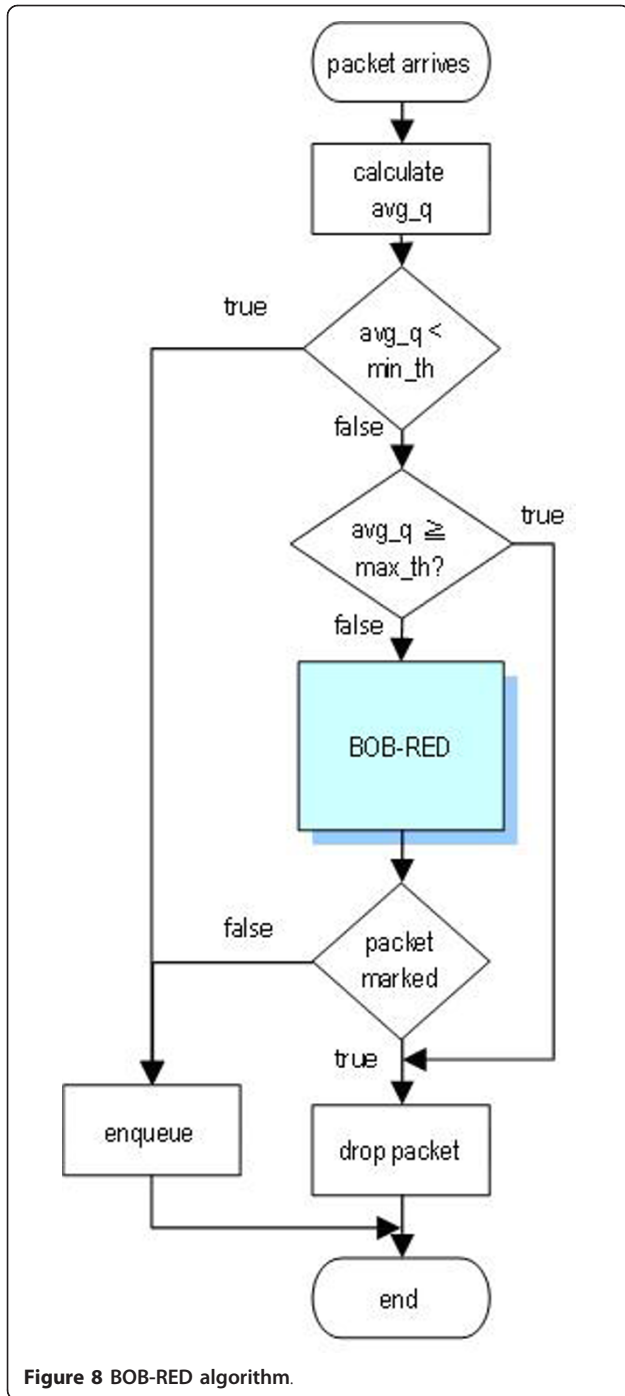


Figure 8 BOB-RED algorithm.

Following shows the pseudocode of BOB-RED algorithm.

for each packet arrival ($class_{rt}$)
 calculate the average queue size $avg_{q_{rt}}$
 if $min_th \leq avg_{q_{rt}} < k$
 accepted with probability 1
 else if $k \leq avg_{q_{rt}} < max_th$

calculate probability with probability

$$P_{loss, rt} = \sum_{i=0}^B \frac{\lambda_{rt}}{\lambda_{rt} + \lambda_{nrt}} p_{i, B-i}$$
 mark the arriving packet
 else if $max_th \leq avg_{q_{rt}}$
 drop the packet
 for each packet arrival ($class_{nrt}$)
 calculate the average queue size $avg_{q_{nrt}}$
 if $min_th \leq avg_{q_{nrt}} < k$
 calculate probability with probability

$$P_{loss, nrt} = \sum_{i=0}^B \sum_{j=0}^{B-i} \frac{(1 - \alpha_{i,j}) \lambda_{nrt}}{\lambda_{rt} + \lambda_{nrt}} p_{i,j}$$
 mark the arriving packet
 else if $k \leq avg_{q_{nrt}} < max_th$
 calculate probability with probability

$$P_{ij} = \frac{(i+j) - k + 1}{B - k + 1}$$
 mark the arriving packet
 else if $max_th \leq avg_{q_{nrt}}$
 drop the packet

3.2 Queuing model of BOB-RED

It is noted that, in Figure 11, we have both real-time and non-real-time traffic coexisting in the network. The queuing model is specifically designed in each gateway node.

We use Table 2 to summarize most of the notation we will use in the formulation.

The waiting time W_{rt} and W_{nrt} of real-time and non-real-time traffic at a gateway node i can be calculated according to Little's formula [21] as follows:

$$W_{rt} = \frac{\sum_{i=1}^B \sum_{j=0}^{B-i} i p_{i,j}}{\lambda_{rt}(1 - P_{loss, rt})} \quad (4)$$

$$W_{nrt} = \frac{\sum_{i=0}^B \sum_{j=1}^{B-i} j p_{i,j}}{\lambda_{nrt}(1 - P_{loss, nrt})} \quad (5)$$

where P_{ij} is the steady-state probability, where i and j denote the number of real-time and non-real-time packets in the buffer, respectively. The loss probabilities $P_{loss, rt}$ and $P_{loss, nrt}$ of real-time and non-real-time packets at a gateway node i are given as follows.

$$P_{loss, rt} = \sum_{i=0}^B \frac{\lambda_{rt}}{\lambda_{rt} + \lambda_{nrt}} p_{i, B-i} \quad (6)$$

$$P_{loss, nrt} = \sum_{i=0}^B \sum_{j=0}^{B-i} \frac{(1 - \alpha_{i,j}) \lambda_{nrt}}{\lambda_{rt} + \lambda_{nrt}} p_{i,j} \quad (7)$$

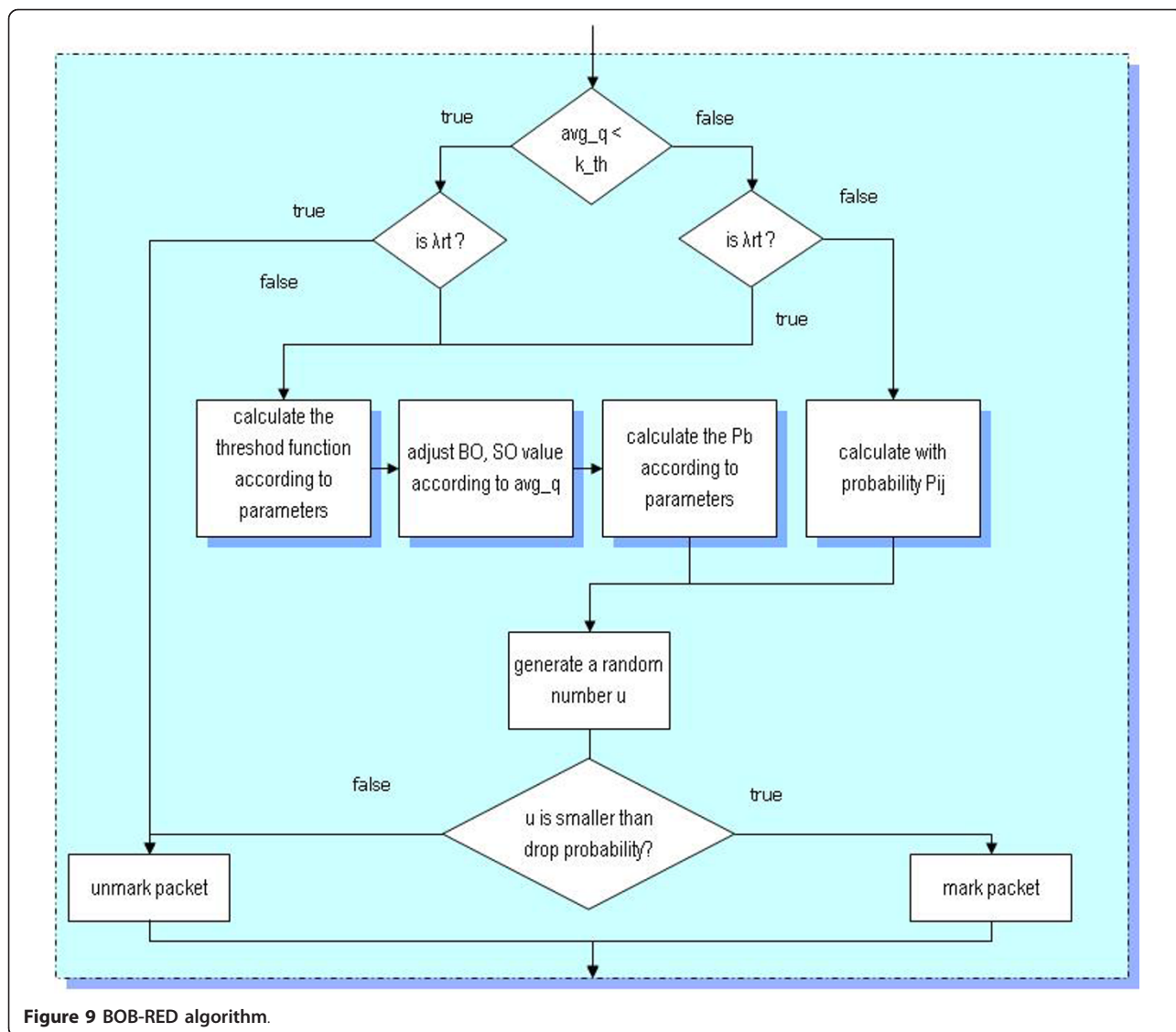


Figure 9 BOB-RED algorithm.

Since we ignore the propagation delay, the end-to-end queuing delay of real-time traffic for a particular path is

$$\begin{aligned}
 T_{end-end} &= \sum_{\forall i \in path} Q_{rt}^i = \sum_{\forall i \in path} \frac{1}{u_{rt}^i - \lambda_{rt}^i} \\
 &= \sum_{\forall i \in path} \frac{1}{u_{rt} - si\lambda_{rt} - \sum_{j=1}^{ri} sj\lambda_{rt}^i}
 \end{aligned} \tag{8}$$

Since the multimedia service is time-constrained. We expect the end-to-end delay along a path under a threshold value.

3.3 The relationship of BO, SO, traffic load against the performance metrics

We use virtual queues for the two different types of traffic, real-time traffic and non-real-time traffic, whose

packets are labeled as different flow number accordingly. On gateway node, there is an agent which determines the order of packets to be transmitted from the queues according to the BOB-RED algorithm. It also checks the type of the incoming packet and sends it to the appropriate queue according to average queue size avg_q , BO , SO , min_th , max_th , max_p parameters.

Table 3 concludes the relationship of BO, SO, traffic load against the performance metrics of delay, throughput and power consumption from many articles.

From many times of experiments, we find somewhat relationship between qualities of services with all the parameters.

We divide all the factors that affect end-to-end delay into five levels from vary parameters. For example, BO, SO values are from 0 to 15. If we set BO, SO value from 0 to 2, the level in spiderchart is 0.

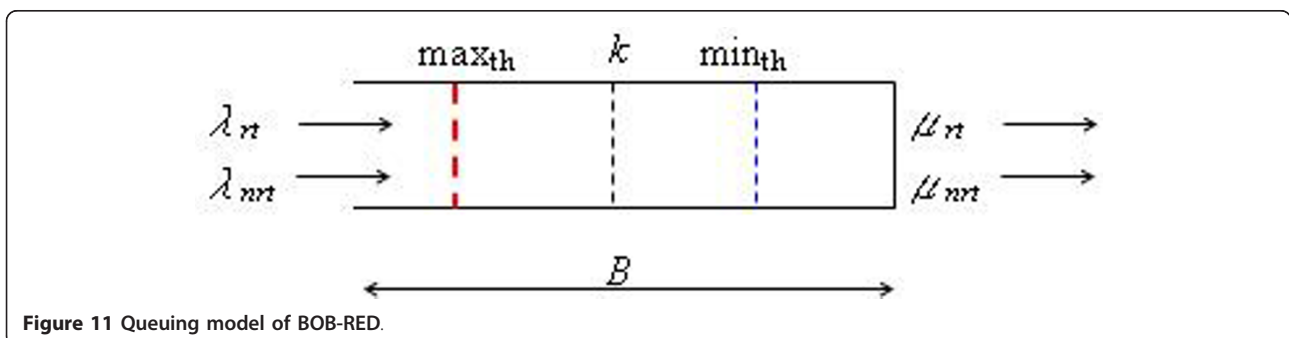
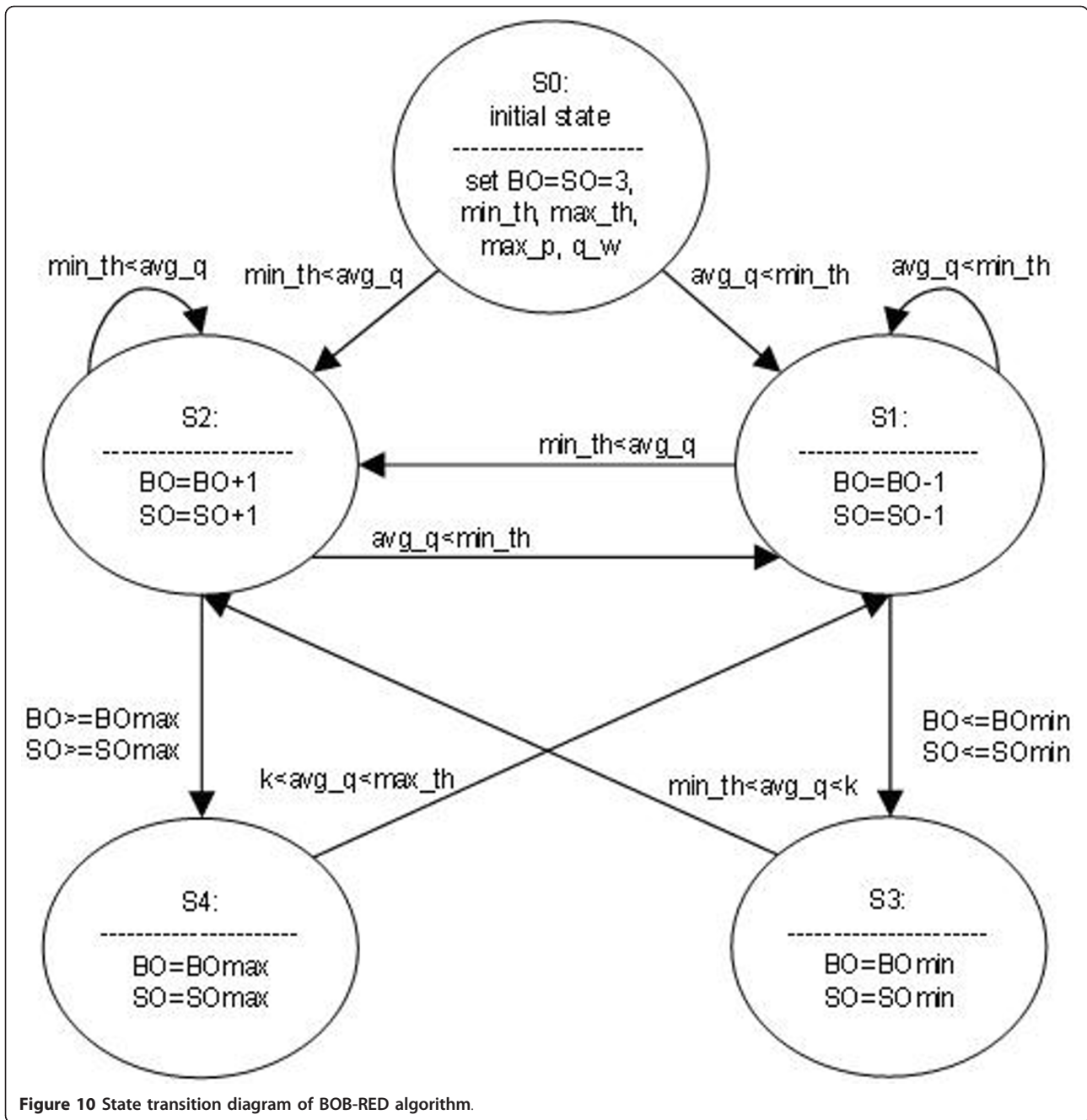


Table 1 BOB-RED dropping strategy

Buffer occupancy	Packet type	
	Real-time traffic	Non-real-time traffic
$avg_q < min_th$	Accepted with probability 1	Accepted with probability 1
$min_th \leq avg_q < k$	Accepted with probability 1	Rejected with drop probability $P_{loss, nrt}$
$k \leq avg_q < max_th$	Rejected with drop probability $P_{loss, rt}$	Rejected with probability $1 - P_{ij}$
$max_th \leq avg_q \leq B$	Rejected with probability 1	Rejected with probability 1

Figure 12 shows the spiderchart of achieving QoS constrained in end-to-end delay. From Figure 12, BO and SO obviously affect the end-to-end delay. Buffer size is not helpful to decrease end-to-end delay. Threshold k is helpful to increase throughputs.

4. Network configuration and assumptions

The solution performance evaluation is carried out under the NS2 simulator [22]. We use the ns2 module developed by Zheng and Lee for IEEE 802.15.4 (NS-2.28) in our simulation. In all simulation, we have considered the followings.

1. The physical layer consists of IEEE 802.15.4 compliant radio transmitter (tx) and receiver (rx) that operate in the ISM band at 2.4 GHz, with raw data rate 250 kbps. The modulation technique is Quadrature Phase Shift Keying (QPSK).
2. The MAC sub-layer implements the slotted/unslotted CSMA/CA.
3. The application layer includes three CBR traffic sources with different data rate and one sink in this article.

Table 2 Notation table

Notation	meaning
B	buffer size
min_th	minimum threshold for queue
max_th	maximum threshold for queue
K	a preset threshold value to separate different dropping strategy for real-time data and non-real-time data
λ_{rt}	real-time data generation rate
λ_{nrt}	non-real-time data generation rate
μ_{rt}	service rate for real-time data
μ_{nrt}	service rate for non-real-time data
Q_{rt}^i	queuing delay on a gateway node i for real-time traffic
s_i	the number of sensing neighbours of gateway node i on path
r_i	the number of relaying neighbours of gateway node i on path
$P_{loss, rt}$	loss probabilities of real-time packets
$P_{loss, nrt}$	loss probabilities of non-real-time packets
W_{rt}	waiting time of real-time traffic at a gateway node i
W_{nrt}	waiting time of non-real-time traffic at a gateway node i
$T_{end-end}$	end-to-end queuing delay for a particular path

4. Different scenarios may have different parameter values setting.
5. Burst traffic is generated from the camera that capturing event photo and sending it to sink immediately. We dynamically adapt BO, SO value to investigate the performance.

Performance metrics measured in this article are the following.

4.1 Average end-to-end delay

Average end-to-end delay is one of the most important metrics to emergent events. In WSNs, the end-to-end delay is the total time delay to deliver a packet from source to sink node. It is the sum of delays at all links within the end-to-end path. The delay at an intermediate node usually includes the following components: processing delay, queuing delay, transmission delay, propagation delay, and retransmission delay. We mainly consider the average end-to-end delay for all source traffic along a multihop path to sink node. By decreasing the packet retransmission, we can decrease the average end-to-end delay.

4.2 Packet delivery ratio

Packet loss may occur at any stage of a network transmission, mainly due to link failures, CSMA/CA channel access mechanism, RED problems. We use packet delivery ratio (PDR) to denote the performance.

$$PDR = \text{received packets/sent packets}$$

4.3 Energy consumption

The coordinator consumes the energy when it transmits beacon and ACK packet, receives data packet, and listens the channel. Where rxPower is power consumption in receiving a packet, txPower is power consumption in transmitting a packet, sleepPower is power consumption in sleep state, and idlePower is power consumption in idle state. To measure the energy consumption in our scenarios, we use the energy model in NS-2. For accelerating the power consumption in our simulation, we modify the default values of the NS2 default value [17]. We only measure the energy consumption on coordinator node in beacon-enabled/non-beacon-enabled modes.

Table 3 The relationship of BO, SO, traffic load against the performance metrics

BO = SO	BO (SO fixed)	SO (BO fixed)	Delay	Throughput	Power consumption
↑	↑	↑	↑	↑	↑
Small	Small	Small	Low	Low	Large
Large	Large	Large	High	High	Small
↓	↓	↓	↓	↓	↓
Traffic load (BO = SO)		Delay	Throughput	Power consumption	
↑		↑	↑	↑	
small		Low	High	Small	
Large		High	Low	Large	
↓		↓	↓	↓	

5. Performance evaluation

There are three scenarios in this study, namely, chain topology, tree topology, and star topology. The simulation parameters are summarized in Table 4. Simulation duration is set to 60 s of which the first 5 s allow the nodes to associate with the PAN coordinator, and the remaining time is used for sending application traffic.

5.1 Chain topology

Figure 13 illustrates the network topology with two FFD nodes (n1, n2) and one PAN coordinator (node 0). Each sensor node is set 20-m away from the other nodes. Three CBR traffic flows (Cbr1, Cbr2, and Cbr 3) enter node 1 and then be sent with the average data rate (2 packets/s) through the coordinator (node 0) to the sink

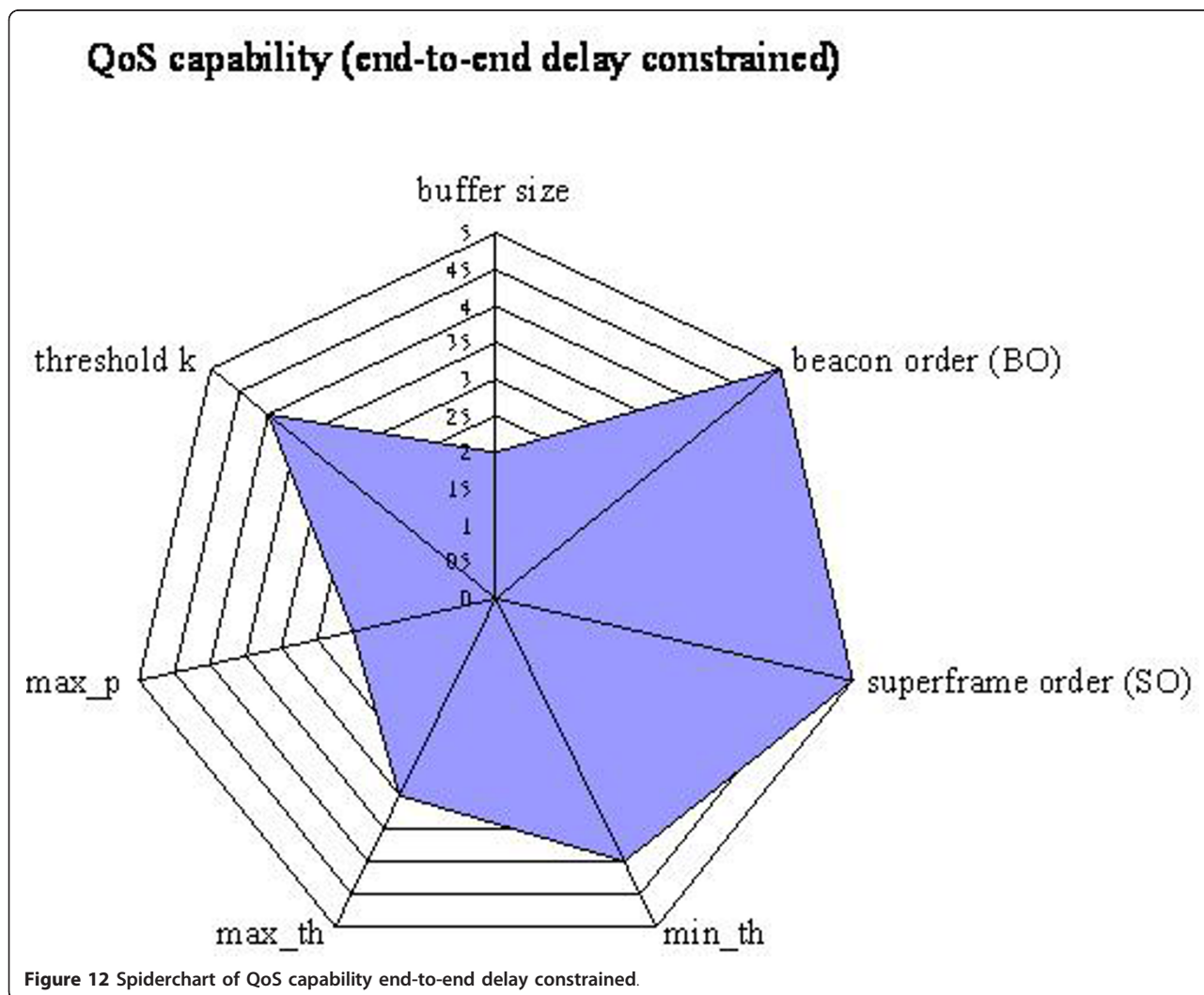


Figure 12 Spiderchart of QoS capability end-to-end delay constrained.

Table 4 Simulation parameters

NS-2 parameters		
Simulation tool	NS-2	
Radio range	25 m	
Data bit rate	250 Kbps	
Routing protocol	AODV	
Scale	50 m × 50 m	
Nodes	3, 4, 5 wireless static nodes	
Simulation time	60 s	
TX power	13.132 mW,	
RX power	13.528 mW,	
Idle power	712e-7 mW	
Sleep power	144e-8 mW	
Node's initial energy	10 J	
Mean packet size	50 bytes	
Traffic type	CBR	
RED parameters		
Minthresh	10	
Maxthresh	30	
k_th	20	
q_weight	0.002	
Gentle	True	
Mark_p	0.1	
Packet_loss_rate	0.05	

node (node 2). Each node's queue size is set to 50 packets.

5.2 Impact of different BO, SO for traffic loads (beacon-enabled mode)

For each configuration, we vary the inter-arrival time of the flows in source node to have different offered loads, assuming a constant packet size. With high offered loads, it causes higher packet loss because of the queue

Table 5 PDR and average delay (droptail, 2 pkts/s, SO = 0)

BO	PDR (%)	Avg. delay (s)
0	100	0.051798
2	100	0.273113
3	100	0.773140
4	1.67	12.413684
5	0	N/A

full. With low offered loads, it causes the increase in the average end-to-end delay.

Table 5 shows the PDR and average end-to-end delay (Avg. delay) using DropTail queue management scheme when SO value is fixed to 0. The PDR values are all zero when BO is greater than 5. Source node cannot transmit data successfully due to long BI and inactive period.

We run the simulation for different values of SO (and BO = SO) as shown in Table 6. When BO equals to SO, less SO value will result in average end-to-end delay increased. When BO value is small, the probability of collision is increased. The bottom of Table 6 shows when BO is fixed, higher SO values have lower average end-to-end delay because of inactive period decreased. Sensor nodes do not change into sleep mode when BO equals to SO.

5.3 Impact of traffic loads (non-beacon-enabled mode)

In this simulation, we send a CBR packet every 5 s for keep-alive traffic, the traffic with data rate 2 packets/s stands for normal traffic, and the traffic which sending 20 packets/s is defined as emergent multimedia traffic load. We set BO, SO value to 15 to evaluate the

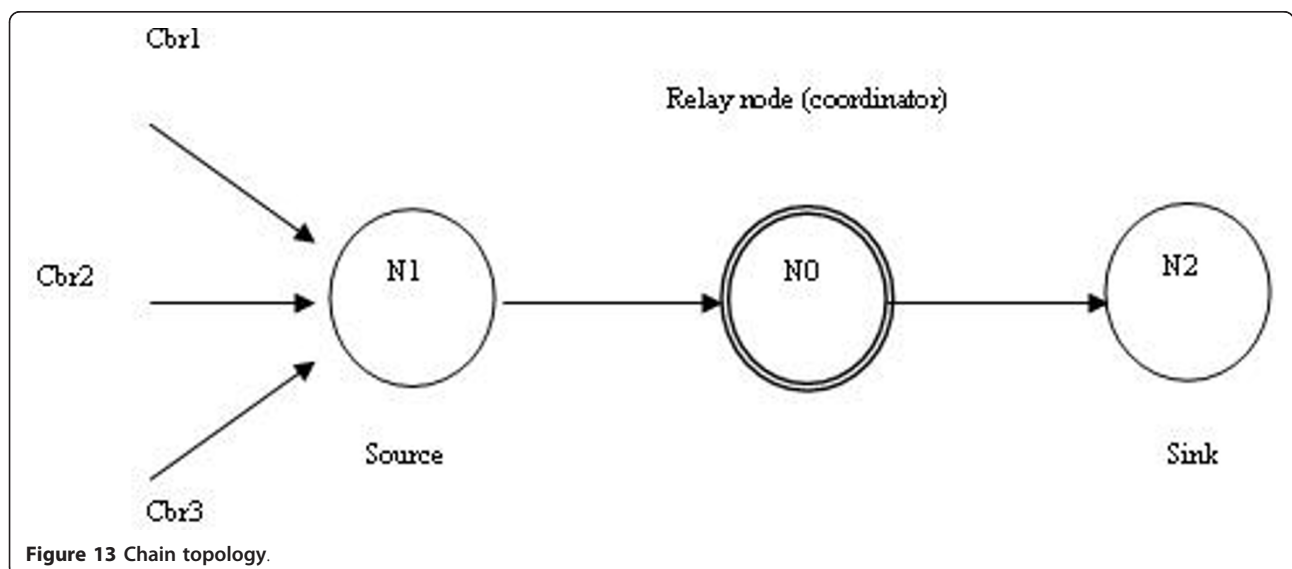


Figure 13 Chain topology.

Table 6 Impact of traffic loads (droptail)

BO	SO	50 pkts/s Avg. delay (s)	100 pkts/s Avg. delay (s)
0	0	1.424160	1.430937
1	1	0.764768	0.781810
2	2	0.612112	0.621294
3	3	0.539826	0.570217
4	4	0.526245	0.552005
5	5	0.536100	0.559807
6	6	0.535082	0.542379
7	7	0.551991	0.518774
3	0	8.071928	8.064701
3	1	5.832748	5.825045
3	2	1.691801	1.690489
3	3	0.539826	0.570217

performance in the non-beacon-enabled mode. Simulation time is 60 s and queue size is 50 packets.

Table 7 shows that all metrics (packet delivery ratio, end-to-end delay, and energy consumption) have not any differences between DropTail and BOB-RED algorithm. Due to unslotted CSMA/CA does not use beacon to coordinate devices. Node which acquires the channel can transmit data immediately, so the queue length always is empty. BOB-RED mechanism does not work in the non-beacon-enabled mode. The PDR value is only 39.27% with 20 packets/s traffic load because of the energy of node 0 runs out at 26.808665200 s.

5.4 Tree topology

In the second set of experiments, we study the effect of dynamic adaptation scheme. Figure 14 illustrates the network topology with four nodes defined using NS-2 (n0, n1, n2, n3). Each sensor node is set 20 m away from the other nodes. We assume node 1, node 2 does not affect each other. Simulation parameters are all the same as shown in Table 4.

We run the simulation for different values of BO, SO where BO equals to SO from 0 to 9 with burst traffic. DropTail is the queue management scheme. Node 2 sends CBR traffic to node 0 with 2 packets/s from 10 to 55 seconds. Following node 1 sends burst traffic with 100 packets/s from 25 to 45 s. All nodes are setting the same BO = SO = 3 value in default simulation. Dynamic adaptation scheme changes vary BO, SO values when burst traffic load comes.

Table 8 shows the comparison of default BO, SO value with dynamic adaptation scheme in the average end-to-end delay, PDR, and energy consumption

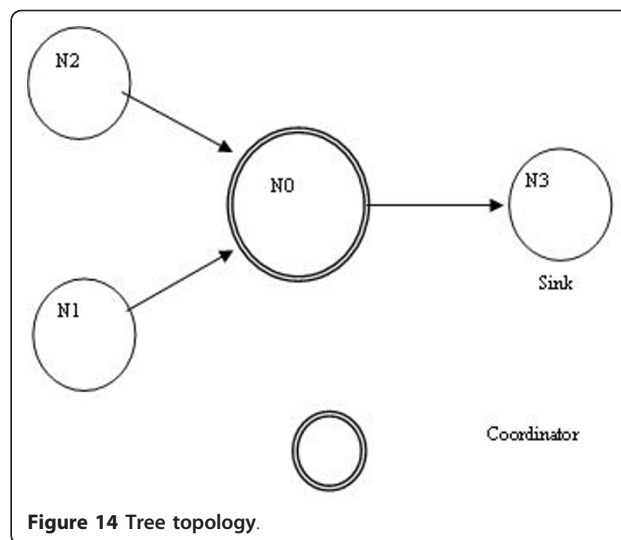


Table 8 Performance of dynamic adaptation scheme

BO = SO	Avg. delay (s)	PDR (%)	Residual energy (node 0 lifetime)
0	0.431532	25.68	Runs out at 35.987584067 s
1	0.256503	26.16	Runs out at 35.430432000 s
2	0.426992	24.63	Runs out at 59.960032000 s
3	0.313196	25.97	Runs out at 33.130784000 s
4	0.615563	25.11	Runs out at 43.386624067 s
5	0.467911	20.66	1.557760 J
6	0.299833	25.78	3.655319 J
7	0.841543	14.97	3.289051 J
8	4.010992	14.16	3.008140 J
9	0.287230	14.16	3.552419 J

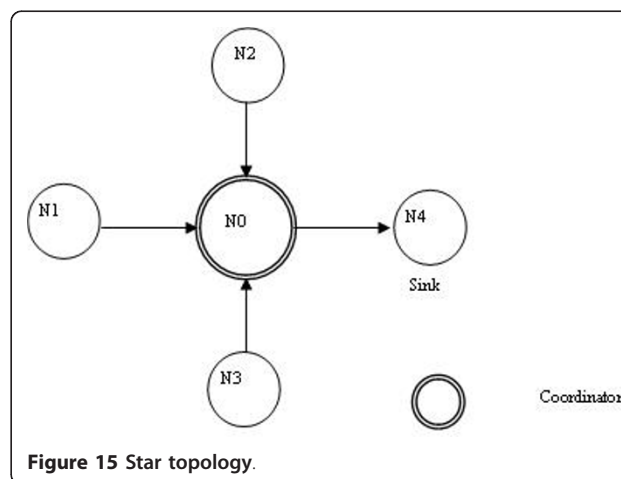


Table 7 Droptail vs. BOB-RED with different traffic loads

Traffic load (packets/s)	DropTail			BOB-RED		
	PDR (%)	Avg. delay (s)	Energy	PDR (%)	Avg. delay (s)	Energy
0.2	100	0.022207	9.712956	100	0.022207	9.712956
2	100	0.019916	7.442435	100	0.019916	7.442435
20	39.27	0.018763	0.005730	39.27	0.018763	0.005730

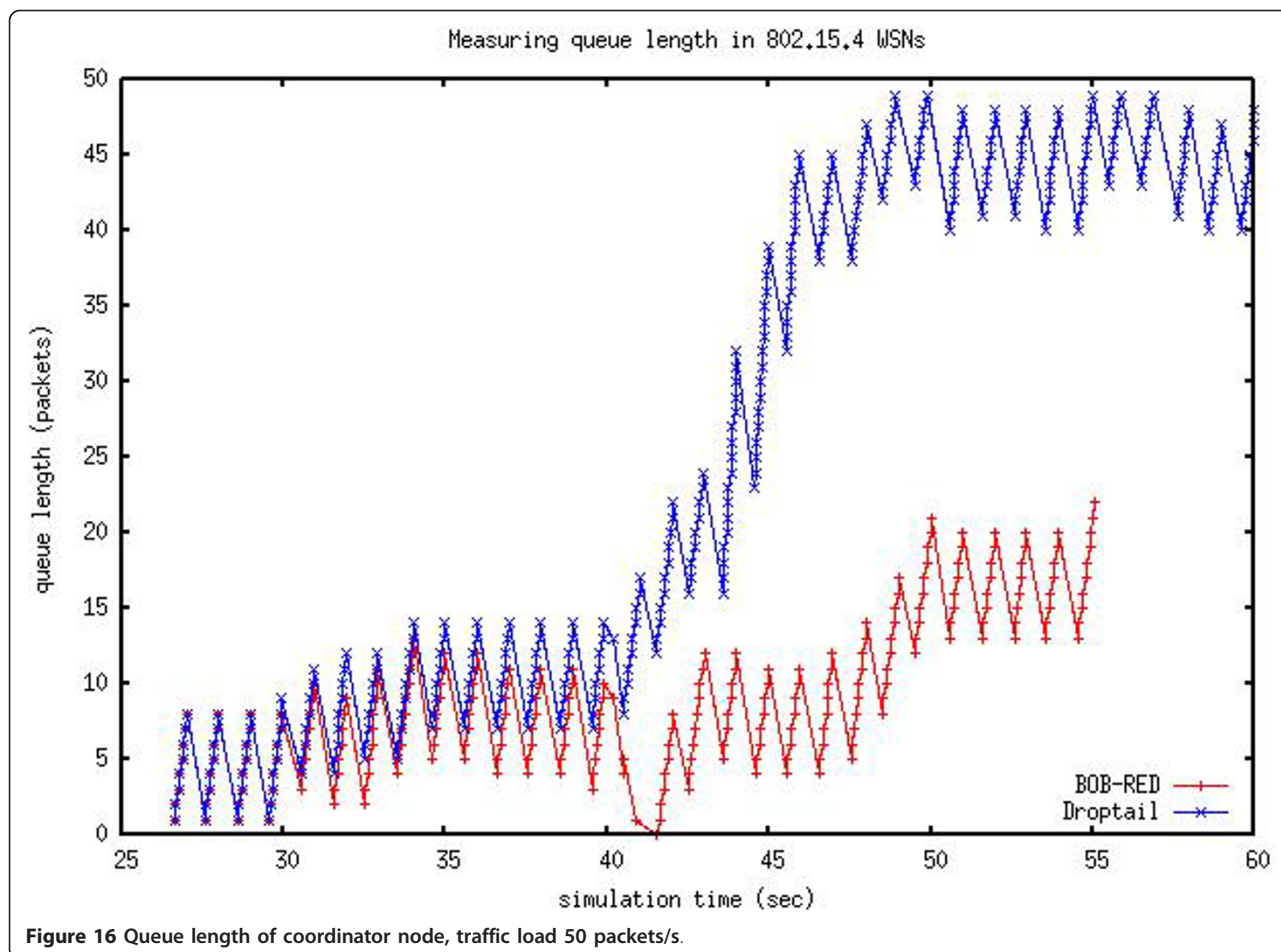


Figure 16 Queue length of coordinator node, traffic load 50 packets/s.

metrics. The average end-to-end delay is 0.313196 s and PDR is 0.2597 in default scheme. We observe that when setting $BO = SO = 6$, PDR is similar with default scheme but average end-to-end delay can effectively decrease in dynamic adaptation scheme. When the simulation ends, coordinator node 0 still remains 3.655319 J power in dynamic adaptation scheme. Conversely, residual energy of coordinator node 0 runs out at 33.130784000 s in default scheme. Simulation result shows suitably chooses BO, SO value can get longer lifetime and less average end to end delay than default setting of BO, SO values.

5.5 Star topology

For investigating the effect of our scheme in multihop environment, the modified star topology is used. We design three traffic flows along coordinator to sink node not only one hop to coordinator like standard. The topology of this scenario is shown in Figure 15. Four FFDs (nodes 1 to 4) are placed symmetrically around the PAN coordinator (node 0) with an equal distance of 20 m. The traffics generated by nodes 1, 2, and 3 send

CBR traffics with 50 packets/s along coordinator (node 0) to sink node (node 4). Node 1 sends CBR traffic from 10 to 55 s. Node 2 sends CBR traffic from 25 to 40 s. Node 3 sends CBR traffic from 40 to 55 s. The focus of the experiment is the head of the bottleneck link between n_0, n_4 and its buffer in particular. Node 0 is the intermediate gateway node with DropTail or BOB-RED queue implemented, with a buffer capacity of 50 packets and a queue fully monitored during the simulation.

5.6 Simulation study in queue management scheme

A similar investigation has been conducted by changing the actual queue mechanism to BOB-RED for all nodes.

Table 9 Droptail VS. BOB-RED with different traffic loads

Traffic load	DropTail		BOB-RED	
	PDR (%)	Avg. delay (s)	PDR (%)	Avg. delay (s)
1	40	0.666688	40	0.666688
50	6.85	7.357023	6.59	3.115317
100	3.43	7.276406	3.29	3.027992

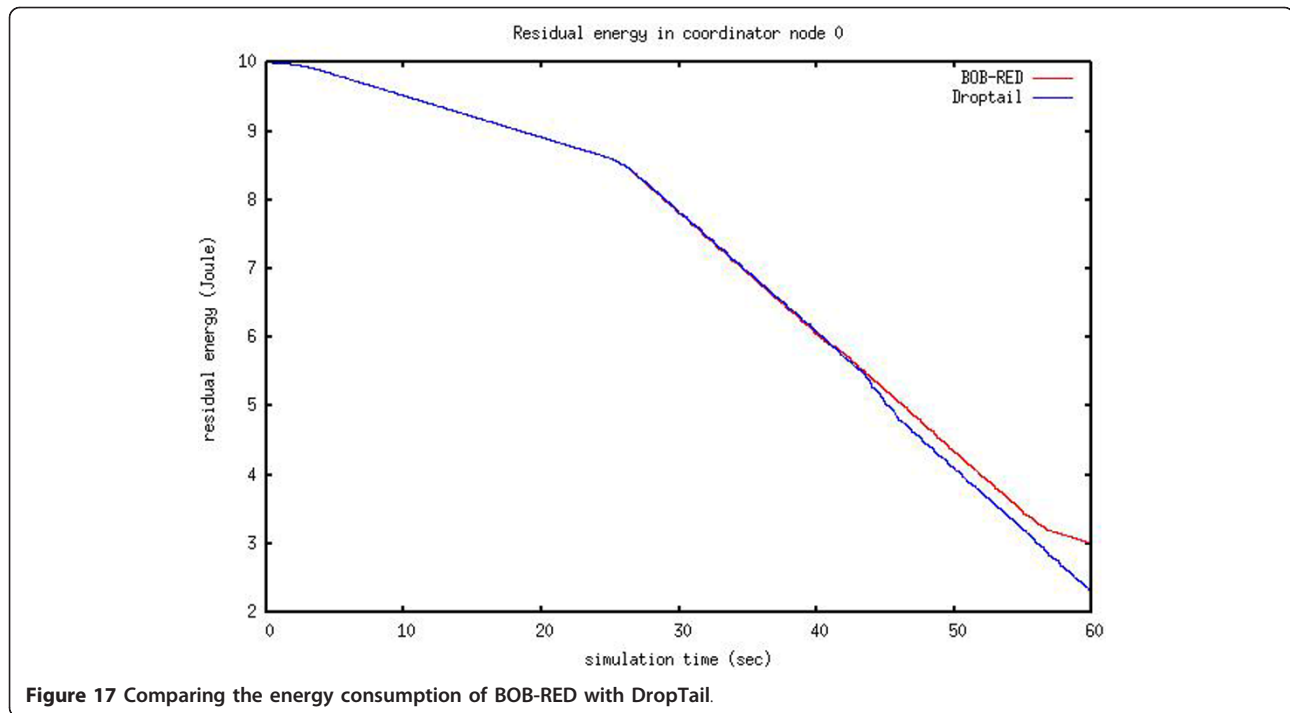


Figure 17 Comparing the energy consumption of BOB-RED with DropTail.

BOB-RED operation is based on the principle that as the probability of a packet being dropped increases, the possibility of this packet being enqueued decreases. Figure 16 shows the BOB-RED scheme has lower queue length than DropTail. We observe queue length at the gateway node 0 which is closest to the sink node and measuring the end-to-end delay from source node to sink node. By measuring the queue length of coordinator node (n_0), we can observe that queue occupy is not heavy. If the traffic load is getting heavier, the amount of packets stay in the queue will increase. BOB-RED can lessen the congestion by early detecting the queue status to drop packet to decrease the retransmission of arriving packets.

5.7 Simulation study in PDR, average end-to-end delay

Simulation results show that light traffic load with BOB-RED can get the same as DropTail in PDR and average end-to-end delay. When traffic load is getting heavier, despite the PDR is slightly lower than DropTail, but BOB-RED can get lower average end-to-end delay than DropTail. Table 9 shows the results.

5.8 Simulation study in energy consumption

Figure 17 shows the energy consumption with different queue management scheme.

The traffic load is 100 packets/s; the total simulation time is 60 s. When the traffic load is light (from 10 to 25 s), the energy consumption of DropTail is the same as BOB-RED. Conversely, both two mechanisms

consume energy rapidly when traffic load is getting heavier (from 25 to 40 s). Then, we can observe that residual energy of coordinator node 0 is higher than DropTail if using BOB-RED mechanism. This is because BOB-RED can lessen traffic congestion to decrease the energy consumption. As simulation result shows, using BOB-RED can get better energy consumption than DropTail when traffic load is getting heavier.

6. Conclusions

In this article, we evaluate several queue management algorithms with respect to their abilities of maintaining high resource utilization and low energy consumption in IEEE 802.15.4 beacon-enabled and non-beacon-enabled modes.

We compare the performance of BOB-RED and DropTail on simulation results, using DropTail as the evaluation baseline. The characteristics of different algorithms are also discussed and compared. We evaluate the impact of the following parameters on the performance of slotted CSMA/CA: (1) the beacon order and the superframe order, (2) queuing management scheme such as RED and DropTail, (3) different traffic loads with dynamic adaptation scheme. Simulation results show that RED queue management scheme accompanied with an appropriate setting of BO, SO can effectively achieve a better performance. Besides, the BOB-RED can decrease end-to-end delay and energy consumption compared to the DropTail scheme.

Despite the vary kinds of experiments have done in this article, how to correctly decide the BOB-RED parameters and BO, SO values according different kind of traffic loads is still an issue. In our future study, we will observe mobile sink and sensor nodes as possible. How the nodes mobility model and velocity to affect the end-to-end delay, packet delivery ratio, and better energy consumption in multimedia services over IEEE 802.15.4 wireless sensor networks is the direction of study.

Acknowledgements

The authors gratefully acknowledge the support by the National Science Council, Taiwan, under grants 100-2221-E-011-154-.

Author details

¹Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C ²Department of Computer and Communication Engineering, St. John's University, Taipei, Taiwan, R.O.C

Competing interests

The authors declare that they have no competing interests.

Received: 30 March 2011 Accepted: 21 September 2011

Published: 21 September 2011

References

1. PART 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Std. P802.15.4a/D5 (2006)
2. KY Wang, SS Lee, et al, Low-MAC FEC Controller for JPEG2000 Image Transmission over IEEE 802.15.4, in *WCSET 2009: World Congress on Science, Engineering and Technology*, 168–171 (February 2009)
3. KY Wang, SY Lee, et al, Robust JPEG2000 Image Transmission over IEEE 802.15.4, in *IEEE International Symposium on Electronic Design, Test and Application*, 253–257 (2008).
4. A Zainaldin, I Lambadaris, B Nandy, Video over Wireless Zigbee Networks-Multi-Channel Multi-Radio Approach, in *Wireless Communications and Mobile Computing Conference, 2008. IWCMC '08. International*, 882–887 (August 2008)
5. S Hengstler, H Aghajan, WISNAP, A Wireless Image Sensor Network Application Platform, in *TRIDENTCOM 2006* (March 2006)
6. G Pekheryev, Z Sahinoglu, P Orlik, G Bhatti, Image Transmission over IEEE 802.15.4 and ZigBee Networks, in *IEEE International Symposium on Circuits and Systems (ISCAS)*. **4**, 3539–3542 (May 2005)
7. CMUcam project, <http://www.cmuacam.org/>
8. Wikipedia, http://en.wikipedia.org/wiki/Random_early_detection
9. S Floyd, V Jacobson, Random early detection gateways for congestion avoidance. *IEEE/ACM Trans Netw.* **1**(4), 397–413 (1993). doi:10.1109/90.251892
10. S Patel, P Gupta, G Singh, Performance measure of Drop tail and RED algorithm, in *International Conference on Electronic Computer Technology (ICECT)*, 35–38 (2010)
11. G Epiphaniou, C Maple, P Sant, M Reeve, Affects of queuing mechanisms on RTP traffic: comparative analysis of jitter, end-to-end delay and packet loss, in *ARES '10 International Conference on Availability, Reliability, and Security*, 33–40 (2010)
12. D Lin, R Morris, Dynamics of random early detection, in *Proceedings of ACM SIGCOMM 97*, Cannes, France, pp. 127–137 (1997)
13. M Shreedhar, G Varghese, Efficient fair queuing using deficit round-robin. *IEEE/ACM Trans Netw.* **4**, 375–385 (1996). doi:10.1109/90.502236
14. GF Ali Ahammed, R Banu, Analyzing the performance of active queue management algorithms. *Int J Comput Netw Commun.* **2**(2), 1–19 (2010)
15. NT Le, SW Choi, YM Jang, Approximate queuing analysis for IEEE 802.15.4 sensor network, in *Second International Conference on Ubiquitous and Future Networks (ICUFN)*, 193–198 (2010)
16. Jelena Mišić, Vojislav Mišić, Access delay for nodes with finite buffers in IEEE 802.15.4 beacon enabled PAN with uplink transmissions. *Comput Commun J.* **28**(10), 1152–1166 (2005). doi:10.1016/j.comcom.2004.07.017
17. C Buratti, Performance analysis of IEEE 802.15.4 beacon-enabled mode. *IEEE Trans Veh Technol.* **59**, 2031–2045 (2010)
18. B Gao, C He, An individual beacon order adaptation algorithm for IEEE 802.15.4 networks, in *Communication Systems, 2008. ICCS 2008. 11th IEEE Singapore International Conference on*, 12–16 (November 2008)
19. J Jeon, JW Lee, JY Ha, WH Kwon, DCA: duty-cycle adaptation algorithm for IEEE 802.15.4 beacon-enabled networks, in *Vehicular Technology Conference, VTC2007-Spring*. IEEE 65th, pp. 110–113.
20. J Zheng, MJ Lee, Will IEEE 802.15.4 make ubiquitous networking a reality? A discussion on a potential low power, low bit rate standard. *IEEE Commun Mag.* **42**(6), 140–146 (2004)
21. L Kleinrock, *Queueing Systems*, vol. I. (John Wiley, 1976)
22. The Network Simulator - NS-2, <http://www.isi.edu/nsnam/ns/>

doi:10.1186/1687-1499-2011-107

Cite this article as: Lin et al.: BOB-RED queue management for IEEE 802.15.4 wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking* 2011 **2011**:107.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com