A Smart Algorithm for QoS Support in Ad hoc Networks

Seyed Hossein Hosseini Nazhad Ghazani[1], R.M.Alguliev[2]

[1]Ph.D Candidate, Institute of Information Technology of ANAS, Baku, Azerbaijan
[2]Doctor of Technical Sciences, Professor, Corresponding member of ANAS, Baku, Azerbaijan
[1]S.HosseiniNejad@gmail.com   [2]Secretary@iit.ab.az

Abstract: Real-time traffic support is one of the interesting Quality of Service (QoS) issues in Ad hoc networks. The wireless ad hoc networks are severely affected by bandwidth, so, supporting QoS in these networks face problems. In this paper, we have proposed a fully distributed MAC algorithm to support the QoS in ad hoc networks. This algorithm provides delay fairness for real-time flows and services these flows considering current QoS of them. This algorithm could be used in emergency and hospital environments. By simulation, we have shown that our mechanism achieves delay fairness, and functions adequately to support real-time traffic in practical environments where real-time traffic and non-real-time traffic coexist in an identical wireless Ad hoc networks.
[Seyed Hossein Hosseini Nazhad Ghazani, R.M.Alguliev. **A Smart Algorithm for QoS Support in Ad hoc Networks**. Life Sci J 2012;9(3):321-325] (ISSN:1097-8135). http://www.lifesciencesite.com. 44

Keywords: Ad hoc, Quality of Service, Differential Service, Back-off mechanism.

1. Introduction

An ad-hoc network is a local area network (LAN) that is built spontaneously as devices connect. Instead of relying on a base station to coordinate the flow of messages to each node in the network, the individual network nodes forward packets to and from each other. Considering the comfortably establishing of ad hoc networks, the use of this type of network is increasing day to day. So, real-time applications will be popular in these networks. Therefore, supporting real-time traffic in wireless LANs (Local Area Network) is an interesting QoS issue. To support real-time traffic in wireless LANs, several algorithms have been studied [1, 2, 3]. All of them use Point Coordination Function (PCF) of IEEE802.11 [4] to support real-time traffic. These mechanisms can provide bounded delay service, but, considering that they could not be used in distributed mode, so, they could not be used in ad hoc mode. On the other hand, for supporting real-time traffic in ad hoc mode, a distributed control mechanism is required. To give the finite transmission opportunities to flows having various features, distributed control mechanisms provide fairness for flows. Take into account that real-time applications are sensitive to delay rather than throughput, we believe that fairness provided for real-time flows should be delay fairness.

For supporting real-time traffic in ad hoc networks, many distributed control mechanisms have been proposed [5, 6, 7]. Although these mechanisms provide higher-priority service for real-time traffic by differentiating real-time traffic from non-real-time traffic they can provide neither delay fairness. In this paper, a fully distributed MAC mechanism is proposed for supporting the QoS of the flows in ad hoc networks that provides delay. To achieve delay fairness, we introduce a back-off algorithm that controls back-off time based on waiting-time, the time a frame has experienced since it was entered to the link interface. This algorithm is preferred to others because it classifies the flows in the network without overhead and any control packet.

The rest of this paper is organized as following. Section 2 describes the DCF method of IEEE 802.11, Section 3 concerns with the QoS frameworks, algorithms and works done to classify the flows and support QoS of flows. In Section 4, by simulation, we confirm that our mechanism functions adequately for supporting real-time traffic in wireless Ad hoc networks, and Section 5 is associated with conclusion.

2. IEEE802.11 Standard-DCF

IEEE802.11 standard describes MAC layer and physical layer specifications for IEEE802.11 wireless LAN [4]. Two access control methods are defined in the MAC layer specifications. One is DCF (Distributed Coordination Function), and the other is PCF (Point Coordination Function). In the following we describe the DCF method (PCF method usable only in infrastructure mode of IEEE 802.11).

Distributed Coordination Function (DCF) is the fundamental MAC technique of the IEEE 802.11 based WLAN standard. DCF employs a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) with Binary Exponential Back-off (BEB) algorithm. In DCF, if a station has frames to be transmitted, a station creates back-off time of the frame, which is used to resolve a contention. Back-off time is calculated using a back-off algorithm. In BEB, back-off time is calculated using the following expressions:

$$CW = (CW_{min} + 1)2^{RC} - 1 \qquad (1)$$

$$CW = \min(CW, CW_{max}) \qquad (2)$$

$$B = \lfloor CW * rand() \rfloor * Slot\_Time \qquad (3)$$

Where CW is contention window, RC is retransmission count of a frame, which RC=0 when the frame is to be transmitted at the first time and RC=n when the frame is to be retransmitted at the n-th time, CW_{min} is the minimum value of CW, which is equal to CW when RC=0, CW_{max} is the maximum value of CW, min(a,b) is the function that returns the smaller number of a and b, rand() is a function that returns a value chose n randomly from the interval from 0 to 1, B is back-off time. If the station, for a time interval greater than a DIFS, determines the medium in idle state, decrements the Back-off time. When the back-off time reaches to 0, the station transmits a frame. After transmitting the frame, if the node receives the ACK for the transmitted frame, the station resets RC and repeats the above procedures for the next frame. If the node does not receive the ACK, it repeats the above procedures for the same frame with incremented RC.

3. QoS Framework
3.1. Find_Fix_Routers for real-time traffics
When a node wants to send a real-time flow, it must, first of all, call for Find_Fix_Router process in order to find a valid path. By a valid path, it is meant a path which is composed of fixed nodes and/or lea mobile nodes and provides for the QoS of the desirable flow.

Find_Fix_Router process based on the modified AODV routing protocol. The modified protocol reflect the selection of stationary routes for real-time traffics. When a source node initiates route discovery for real-time traffic with strict quality requirements, only the fixed routers respond to the control packets by either forwarding the RREQ, or unicasting a RREP. The mobile nodes do not respond to these packets, unless they are the destination.

Find_Fix_Router also enables effective admission control when the network utilization is saturated. This requires accurate estimation of channel utilization and prediction of flow quality, i.e., throughput or transmission delay. The proposed QoS approach is based on model-based resource estimation mechanism, called *MBRP*[17]. By modeling the node back-off behavior of the MAC protocol and analyzing the channel utilization, MBRP provides both per-flow and aggregated system wide throughput and delay [16].
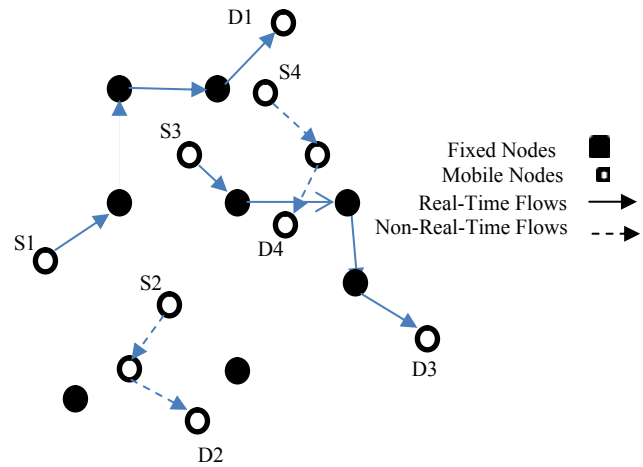

Figure 1: Different paths for different flows

3.2. Distributed MAC for Real-time Flows
In Ad hoc networks, the priority scheduling algorithms are based on IEEE 802.11 [8]. Currently, there are some algorithms that differentiate between flows through assigning various minimum contention window sizes (CW_{min}), Arbitrary Inter Frame Spacing (AIFS), or back-off ratios, but, these differentiations are static. To achieve service differentiation, as well as to adapt to the current usage of network, we combine the collision rate and current QoS of flows with the exponential back-off mechanism in IEEE802.11. To do it, classifies flows to Real-Time flows and Best-effort flows. Upon receiving the first packet from the flow related to one of those classes, each node, locally builds a queue for that flow. Then it inserts this packet and subsequent packets related to that flow in this queue. It is noted that contrary to real-time flows where a separate queue is built in every node for each flow, only a queue is built for all best-effort flows in every node. Figure 2 shows the queues built in each node to manage various flows. We will use these queues in managing flows and calculating contention window of them.

3.3. Mechanisms for Supporting QoS of Flows
A distributed MAC mechanism supporting real-time traffic is desired to coexistence with a MAC mechanism for non-real-time traffic. Because real-time traffic and non-real-time traffic coexists in a practical wireless LAN, a MAC mechanism of wireless LANs is required to support both real-time traffic and non-real-time traffic. Now, DCF of IEEE802.11 is the most widely used wireless LAN protocol and is the standard for non-real-time traffic. Therefore, a MAC mechanism for real-time traffic is required to coexist with a MAC mechanism for non-

real-time traffic such as DCF.

To meet this purpose, we propose a new MAC mechanism. DCF cannot support delay fairness, because BEB (Binary Exponential Back-off), the back-off algorithm have been used in DCF, decides back-off time of a frame regardless of how long the frame is kept waiting to be transmitted. Then, to achieve delay fairness, we introduce "waiting-time" defined as the time a frame has experienced since it was entered to the link, and we give smaller back-off time to a frame having larger waiting-time.

The Real-time flows, such as conversational audio/video conferencing, require that packets arrive at the destination within a certain delay bound ("Request-Time). The Best-effort flows, such as file transfer, can adapt to changes in bandwidth and delay. Due to the different requirements of flows, each types of flow have its own contention window adaptation rule, as follows:

A) Best Effort Flows:

Best effort flows are tolerant to changes in service levels and do not have any hard requirements about bandwidth or packet delay. The purpose of updating the back-off size of best effort flows is to prevent best effort flows from congesting the network and degrading the service level of real-time flows and this is done by controlling the network congestion.

$$CW^{(n+1)} = CW^{(n)} \times (1 + \gamma(f - F^{(n)})) \qquad (4)$$

In header of real-time packets we have added two fields entitled "Create-Time" and "Waiting-Time". The Create-Time field has been shown how many time elapsed as the birth time of a packet. Also, the waiting-time field has been shown the elapsed time of a packet in a node. In proposed algorithm, we have tried service to packets that have the largest waiting-time. On the other hand, if the value of "Create-Time" will be greater than the value of "Request-Time", then this packet will be rejected, because the time to live of this packet is finished. The "Request-Time" shows the acceptable value for service of real-time packets. A Real-Time flow adopts the back-off algorithm by the following expressions:

$$CW = (CW_{min} + 1)2^{RC} - 1 \qquad (5)$$

$$CW = \min(CW, CW_{max}) \qquad (6)$$

$$B = \lfloor CW * rand() \rfloor \qquad (7)$$

$$B = \lfloor B * K / t \rfloor \qquad (8)$$

$$B = \max(B, B_{min}) \qquad (9)$$

$$B = \min(B, B_{max}) \qquad (10)$$

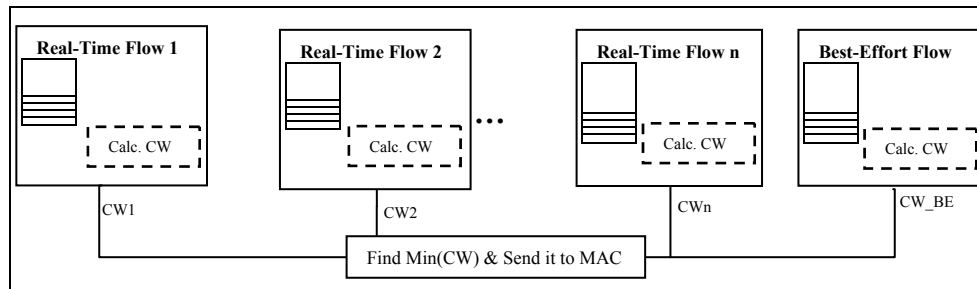Where t is waiting-time, K is a constant value of which is the same in any station.

Figure 2: Queues in each node (Network Layer)

When the average idle channel time F is smaller than the threshold value f, the network is considered congested and the contention window size of the best effort traffic is increased to avoid decreasing the service level of real-time traffic. On the other hand, if the network is lightly loaded so that the idle channel time is larger than f, the contention window size of best effort traffic is decreased so that the idle bandwidth can be utilized.

B) Real-Time Flows:

This back-off calculation algorithm is composed of 2 steps. In the first step, the BEB procedure decides B according to the expressions (5), (6) and (7). (5), (6) and (7) are equivalent to (1), (2) and (3), respectively. We utilize the contention resolution functionality of BEB in this step. In the second step, B is adjusted according to the expressions (8), (9) and (10). This step decides B to be inversely proportional to t, whereby a frame of larger waiting time has higher transmission priority.

Expressions (9) and (10) limit B in the range from B_{min} to B_{max}.

The pseudo-code of Receive and Send functions are as follows:

Receive Sensitive Packet (P as **packet**)
{
   If (Create_Time(p)>ReT(p)) **then**
     Reject (P)
  Static Waiting_Time=0 ;
  Waiting_Time++;
}

Send Sensitive Packet (P as **packet)**
{
  Create_Time(p) = Create_Time (p)+Waiting_Time(p);
}

The pseudo-code of back-off computation is as follows:

  Back-off_Time()
  {
    If (TypeOf(Flow)='Real-Time') **then**

$$CW = (CW_{min} + 1)2^{RC} - 1$$

$$CW = min(CW, CW_{max})$$

$$Back - off = \lfloor CW * rand() \rfloor$$

$$Back - off = \lfloor Back - off * K / t \rfloor$$

$$Back - off = max(Back - off, Back - off_{min})$$

$$Back - off = min(Back - off, Back - off_{max})$$

    Else If (TypeOf(Flow)='non-Real-Time') **then**

$$CW^{(n+1)} = CW^{(n)} \times (1 + \gamma(f - F^{(n)}))$$

$$Back–off=Rand[0,(2^r +R_{col})*CW]*Slot\_Time$$

    }

Where R_{col} denotes the collision rate between a station's two successful frame transmissions and r is a positive number.

By the Back-off-Time(), all flows dynamically manage their contention parameters to meet their own QoS needs. A real-time flow that did not get its required QoS in the past due to competition from other flows try to lower back-off and so, give media as soon as possible. A best effort flow, on the other hand, increases its contention window size when the network is considered busy and hence releases the channel to the real-time flows (Eq.(4)). More importantly with attention to flow's current status, traffics with same class will have different back-off

value when collisions occur. Specifically, after a collision occurs, low priority traffic will back-off for longer, and subsequently high priority traffic will have a better chance of accessing the channel. Contrary to [9], [10], in our proposed algorithm, no piggy-backed schedule information and neighborhood scheduling tables are needed. Therefore, there is no control message overhead imposed by our proposed algorithm.

In next section, the simulation of the proposed algorithm is proved.

4. Simulation

In this section we evaluate the proposed MAC mechanism by simulation. The simulation is done in ns-2 and network size is 1000m*1000m. In the simulation environment n senders transmit frames destined for an identical receiver at constant bit rate. The configurations are shown in Table 1.

In this paper, we use fairness index [11] to evaluate accuracy of delay fairness and differentiation. Fairness index is defined by the:

$$fairness\,index = \frac{(\sum_{i=1}^{n} d_i / w_i)^2}{n \sum_{i=1}^{n} (d_i / w_i)^2}$$

Where n is the number of senders, d_i is the delay of sender i, and w_i is the weight of sender i. Fairness index is greater than 0, and less than or equal to 1. It approaches to 1 as $d_1 / w_1, d_2 / w_2, ..., d_n / w_n$ are becoming the same.

Table I: Configuration

Sender	CW_{min}	CW_{max}	K	B_{min}
1-n	31	1023	0.0005	1
B_{max}	**W**	**IFS**	Queue length	Frame Size
1023	1	DIFS	16 KBytes	1024 byte

We evaluate the proposed MAC mainly by comparing the simulation results of the proposed MAC with those of DCF. In the configuration (Table I), n senders all implement the proposed MAC mechanism and have the parameters in the table. For the purpose of the comparison of the proposed MAC and DCF, we have all the senders implement DCF too. In the case, they have the parameters in the table except K, B_{min}, B_{max}.

Figure 3 shows simulation results in the case where the n sender all implement the proposed MAC or in the case where the n senders all implement DCF.

The lines of "min" present delays of the sender having the smallest delay of the n senders, the lines of "max" present delays of the sender having the largest delay of the n senders, and the lines of "ave" present the average values of the delays of the n senders. Both in the proposed mechanism and in DCF, if $n \leq 6$, all the senders experience little delay because the medium load is light. On the other hand, if $n \geq 7$, the average delay of the n senders increases as n grows larger. Additionally, there are two features when $n \geq 7$. One is that the average delay in the proposed mechanism is smaller than that in DCF. The other is that the difference between the maximum delay and the minimum delay in the proposed mechanism is much smaller than that in DCF. In other words, the proposed MAC is a better mechanism to achieve low delay and delay fairness than DCF. This advantage is derived from our original procedure (8), where larger waiting-time leads to smaller back-off time.
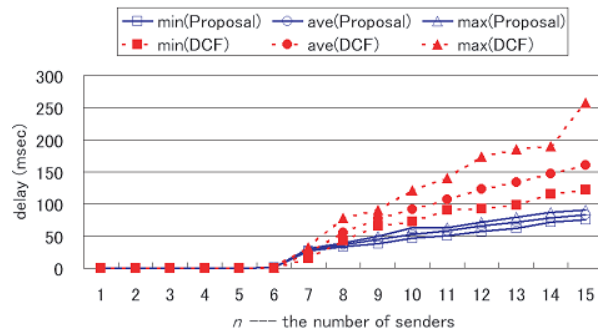

Figure 3: Delay of Flows

5. Discussions

In this paper we introduce a new distributed MAC protocol that supports QoS of flows in ad hoc networks, which this protocol is simple, fully distributed and use no control packets. An important benefit of this protocol is that it does not need resource reservation and therefore, it does not have the problems related to the use of in-bound and out-bound signals to reserve and free the resources, and the network bandwidth is not occupied by reserving and freeing the resources.

In the future, we will investigate the effect of different values t, f, r on the throughput and delay related to different classes.

Corresponding Author:
Dr. Seyed Hossein Hosseini Nazhad Ghazani

1/26/2012

Azerbaijan National Academy of Science,
Institute of Information Technology
Baku, Azerbaijan (http://www.ict.az/en/index.php)
E-mail: S.HosseiniNejad@gmail.com

References

1. Maazen A., Weihua Z., Ping W., Link layer solutions for supporting real-time traffic over CDMA wireless mesh networks, Published in: Journal Wireless Networks, Volume 10 Issue 1, Kluwer Academic Publishers Hingham, MA, USA, January 2004.
2. Ang-ping Sh. , Chi-Hsun L. , Shih-lin W. , Yu-Chee T. , A Priority MAC Protocol to Support Real-time Traffic in Ad Hoc Networks, ACM Wireless Networks, 2004.
3. Malathi Veeraraghavan, Nabeel Cocker, and Tim Moors. Support of voice services in IEEE 802.11 wireless LANs. In Proceedings of the IEEE INFOCOM 2001 Conference, pp. 488–497, Anchorage, Alaska, USA, April 2001.
4. Gollagi S.G., Ankaliki S.G.and Zinage H.R. MAC Layer's Misbehavior Handling in Wireless Network, Int'l Journal of Computational Intelligence Techniques, ISSN: 0976–0466, Vol.1, Issue 2, pp-18-21, 2010.
5. Maode M., Zheng X., Supporting Real-Time Service in Packet-Switched Wireless Networks, International Journal of Business Data Communications and Networking (IJBDCN), Volume 2, Issue 1. 2006.
6. François Baccelli, Nicholas Bambos, Nicolas Gast, Distributed delay-power control algorithms for bandwidth sharing in wireless networks. IEEE/ACM Transactions on Networking (TON), Volume 19 Issue 5, IEEE Press Piscataway, NJ, USA, October 2011.
7. Eryilmaz, A. et al. "Distributed Cross-Layer Algorithms for the Optimal Control of Multihop Wireless Networks." Networking, IEEE/ACM Transactions on 18.2: 638-651. IEEE 2011.
8. Heng Xu, Comparative Analysis of Scheduling Algorithms in Ad Hoc Mobile Networking, Parallel and Distributed Computing, Applications and Technologies, Fudan University, China, 2005.
9. V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly. "Distributed Multi-Hop Scheduling and Medium Access with Delay and Throughput Constraints". In Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking (MobiCOM'01), Rome, Italy, July 2001.
10. Haiyun Luo, Songwu Lu, Vaduvur Bharghavan, Jerry Cheng, and Gary Zhong. A Packet Scheduling Approach to QoS support in Multihop Wireless Networks. ACM Journal of Mobile Networks and Applications(MONET), Special Issue on QoS in Heterogeneous Wireless Networks, 2002.
11. Sangheon P., Hojin L., Call Setup Latency Analysis in SIP-Based Voice over WLANs, Communications Letters, Volume: 12, Issue: 2 Page(s): 103-105 IEEE, February 2008.