A QoS Approach to Hybrid TDMA with Heuristic Traffic Shaping for Time Critical Application Environments

G. D. Pallas, Student Member IEEE, G. I. Papadimitriou Senior Member, IEEE, A. S. Pomportsis, Member, IEEE
Aristotle University of Thessaloniki, Computer Science Dept.
Email: gpall@ccf.auth.gr

Abstract

QoS/HyTDMA-HTS is a new collision-free MAC sublayer protocol which provides the well known HyTDMA-HTS protocol with QoS extensions which enable it to offer quality services in demanding environments with critical applications for which time sensitivity is a key issue. QoS/HyTDMA-HTS can operate in bursty traffic networks, successfully regulating QoS parameters of hosts running real-time services even under high traffic loads, while at the same time, the traffic shaping techniques which it employs, prevent other hosts from sending large amounts of traffic and monopolizing the available bandwidth.

1. Introduction

With the introduction and widespread use of VoIP [1] and other time critical technologies such as teleconferencing, much effort has been given into implementing quality of service into the IP layer, and now as the (essentially broadcast) WLAN technology advances, the need for incorporating QoS features in the MAC sublayer [2] emerges [3].

In a time-critical environment the hosts running the core applications (e.g. call managers, voice gateways) are by default not considered equal to other client stations on the LAN from the point of access benefits they should enjoy, thus ruling out the choice of anything from the static allocation MAC protocol family [4] (e.g. TDMA).

Other proposals for the MAC sublayer include using learning automata and a probabilistic modeling [5] of the network's traffic in order to address the key issue of allocating bandwidth effectively within bursty traffic conditions, as in VPL/LTDMA [6]. VPL/LTDMA solved the issue of idle timeslots due to stations with low traffic, problem which caused increased packet delays to high traffic stations, and gave priority to stations having a high packet flow rate, thus keeping their packet delays to a minimum –but with no guarantee.

An advance to VPL/LTDMA was HyTDMA-HTS which achieved much lower packet transmission delay times and a better channel utilization both in high and medium traffic networks [7], due to its ability to enable stations to transmit packets of arbitrary size, thus eliminating the need for fragmentation and the following overhead costs, and in addition, the traffic shaping techniques it employed, efficiently regulated the bandwidth usage so that no host could monopolize it.

In this paper, a new MAC sublayer protocol is introduced which, while using techniques of HyTDMA-HTS and retaining all their beneficiary results, uses an entirely different prioritization scheme in order to incorporate QoS characteristics, so that it can be used in local networks where time-critical applications are hosted and thus, network access parameters (e.g. packet delay) should be guarantied.

This paper is organized as follows:

Section 2 introduces the reader to QoS/HyTDMA-HTS, and is followed by section 3 which gives an insight to the way the protocol implements its features. Section 4 describes the traffic shaping mechanism of QoS/HyTDMA-HTS as well as its QoS modules which guarantee quality of service to the network. Section 5 tests the protocol under high bursty traffic loads. Sections 6 and 7 follow, in which QoS/HyTDMA-HTS is tested under non-bursty traffic and real traffic collected from a LAN, respectively. In Section 8 we discuss on the results of the simulations and the traffic models that were selected. In the last section, we draw some conclusions.

Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005)
1530-1346/05 $20.00 © 2005 IEEE

2. About the QoS/HyTDMA-HTS protocol

QoS/HyTDMA-HTS is a new dynamic bandwidth allocation protocol, able to deliver QoS features on local networks which would benefit from a stable network providing guaranteed access to the shared medium, for example server farms, or networks hosting time-sensitive applications.

Our new protocol features two additional modules to those ones implemented in HyTDMA-HTS, namely the QoS adjuster and the QoS monitor [8]. The QoS monitor observes QoS parameters (e.g. packet delay) and sends feedback to the QoS adjuster which then configures the station selection mechanism accordingly, in order to achieve the goals set by the application layer (e.g. VoIP application).

According to QoS/HyTDMA-HTS, each node on the network is assigned a priority integer number in the range [1,10], where 10 is the maximum priority and 1 is the lowest, and a desired packet delay (DPD). Maximum priority is assigned to stations which have much traffic, or stations which should have low and, most importantly, constant packet delays in order not to experience jitter which is disturbing to real-time applications.

During the protocol's function, the QoS monitor module of QoS/HyTDMA-HTS feeds the current QoS parameters to the adjuster module which regulates access to the shared medium so that each station's packet delay stays as close to it's set desired packet delay (DPD), as much as the current network traffic permits and as much as its priority gives it an advantage over the other stations.

That means that the protocol guarantees that a station with maximum priority will always have a packet delay within a short range of its DPD, whereas packet delays of stations with lower priority will divert more from their DPD in case of very high loads on the network. In short, QoS/HyTDMA-HTS makes sure that the more important a station is, the less will it divert from its set QoS parameters.

3. The QoS/HyTDMA-HTS implementation

QoS/HyTDMA-HTS is a full knowledge, thus collision-free protocol: At every instant, all stations on the network know which of them have traffic in their queues waiting for transmission and which not. This is achieved by encapsulating in the transmitted data frames some meta-information which consists of how many packets remain in the station's queue and its size. The overhead of the said meta-information is minimal given the average-sized Ethernet frame [7], [9], especially considering the overall gain in performance.

There are two pools from which the stations are selected for transmission: the idle-station pool and the non-idle, where the hosts are classified depending on the size of their queue that they reported using the meta-information. The protocol then selects hosts for transmission from the two pools with an adjustable round-robin scheme so that it can fine tune to any given network.

One the strong features of QoS/HyTDMA-HTS is its traffic shaping routines which adjust the bandwidth utilization so that no station suffers from 'starvation' in case some other stations are flooding the shared medium with great amounts of traffic. In the later case, the protocol imposes a 'penalty' to those stations, denying them transmission, until their transmission rate converges to a common average.

In addition to the above, the QoS adjuster module processes the QoS monitor metrics and regulates the penalties in the cases of stations with a high priority. Actually, as the priority of a station rises, the penalties imposed to it diminish, and in the case of a maximum priority station, absolutely no penalties are imposed. And before the round-robin selection scheme is applied, QoS adjuster checks whether the desired packet delays (DPDs) of the high priority stations are satisfied. When not, the round-robin scheme is bypassed and a direct station selection occurs.

All of the above function in a way that under heavy load conditions, QoS/HyTDMA-HTS behaves like TDMA, thus optimally [2], whereas at low traffic conditions, the protocol selects only the non-idle stations, thus saving transmission timeslots, keeping the packet delays to a minimum and maximizing channel utilization, while at the same time guaranteeing that critical hosts on the network get as much bandwidth as they need, a fact that will be made clear by the simulation results in the following paragraphs.

4. The traffic shaping mechanism and the QoS modules of QoS/HyTDMA-HTS

As we analyzed before, the QoS monitoring module performs checks to see whether the set QoS goals for each station are satisfied. This is accomplished as follows: First, the i^{th} station's packet delay divergence from the set DPD is calculated according to the formula:

$$\mathrm{DIV}_i = \frac{tpd_i}{tpb_i * DPD_i} - 1$$

where tpd_i is the total packet delay of station i and tpb_i is the number of packets that station i has broadcasted. In case the station has better access to the

network than the one demanded (i.e. average packet delay < DPD_i), then $DIV_i < 0$, so there is no need for bandwidth adjustment and the protocol proceeds as normal to the round robin selection scheme. In case the station falls short of the DPD goal (i.e. $DIV_i > 0$), the parameters are passed to the QoS adjuster module.

The later module first checks to see how significant for the network is the excess of the station's set DPD. Its significance is calculated by two factors: a) the priority of the station, and b) how much the station's packet delay has diverted from the desired DPD. We must here note that only stations with a priority of 5 and higher are checked for DPD excesses.

To arithmetically express this significance, we used the following logarithmic formula in order to scale the divergence (which theoretically can be arbitrarily large) and quantify the importance of the DPD excess for station i:

$$\text{precedence factor}_i = \log(DIV_i + 1) * \text{priority}_i$$

Next, in order for the station with the maximum precedence factor (MPF) to get selected to transmit, it has to exceed the accepted range around its DPD as given in the next table:

Table 1. QoS divergence limits for QoS/HyTDMA-HTS

Priority	5	6	7	8	9	10
MPF %	60	50	40	30	20	10
MPF	0.6	0.5	0.4	0.3	0.2	0.1

As we can see, the higher the priority, the more strict becomes the divergence limit, after which the station gets immediately selected in order to regulate its packet delay in the accepted range around its DPD.

This was the QoS function of the protocol, which coupled with the traffic shaping techniques which we will describe now, achieves better results than other similar protocols such as VPL/LTDMA. Traffic shaping works in QoS/HyTDMA-HTS by heuristically calculating the 'penalty' (i.e. transmission denial) by which a station should be fined, in case it transmits much more traffic than the other stations and thus monopolizes the bandwidth. The penalty calculation is performed using the following formulas:

First, a priority-weighted average packet size is calculated:

$$PW\_APS = \frac{\sum_{i=1}^{n} b_i * PC_i}{\sum_{i=1}^{n} Pck_i * PC_i}, \text{ where } PC_i = -P_i / 9 + 10/9$$

PW_APS is the priority-weighted average packet size, PC_i is the priority coefficient (traffic of high priority stations are taken less into consideration), b_i is the size in bits of the station's queue, Pck_i is the number of packets in the queue and P_i is the i^{th} station's priority. Next, the actual penalty is calculated (i.e. the number of rounds for which station i will be denied transmission):

$$\text{Penalty}_i = \frac{b_i * PC_i}{Pck_i * PW\_APS} - 1, \text{ where } PC_i \text{ same as above}$$

According to the above formula, the penalty increases to the extent that station i's transmission rate exceeds the average transmission rate, but the quantity of the penalty is inversely proportional to the station's priority. That totally excludes from the traffic shaping mechanism the very important stations (i.e. with priority 10). This strategy satisfies, as verified by the simulation results, the QoS goals that have been set, and in addition manages effectively the available bandwidth by forcing low-priority stations to conform their transmission rate to the common average, where at the same time allocating enough bandwidth to important stations for their time sensitive applications.

5. QoS/HyTDMA-HTS performance under bursty traffic

In order to thoroughly test the QoS mechanism of our protocol, we conducted a number of simulations under various types and loads of network traffic. In the first simulation, we used a bursty traffic model to show the effect that the traffic shaping and QoS modules have on a typical high-load computer LAN where there are two main servers with much more traffic than the others. In order to compare, fig. 1 was produced without station priorities, whereas fig. 2 with priorities 10 and 6 for stations A and B respectively, so that the QoS modules would take effect on the two station's performance.

Fig. 1. shows that the packet delay for the two server stations grows extremely rapid under heavy load, and this is because the traffic shaping techniques due to lack of bandwidth, penalize the two stations so that all the others do not starve (this is proved by the fact that the average delay is rising much slower than the delay of stations A and B).

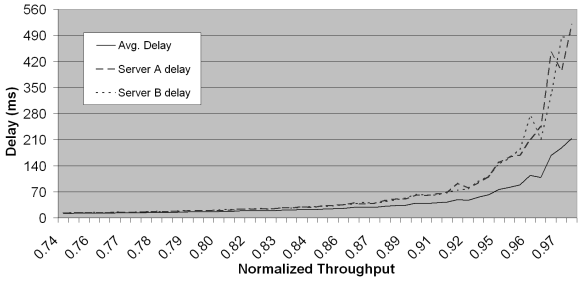

Fig. 1. LAN's servers do not use priorities (i.e. no QoS mechanism used)

As we mentioned before, the next figure (fig. 2) was produced using priority 6 for station B and 10 (max) for station A. The DPD (desired packet delay) for both stations A and B was set to 50 msec.

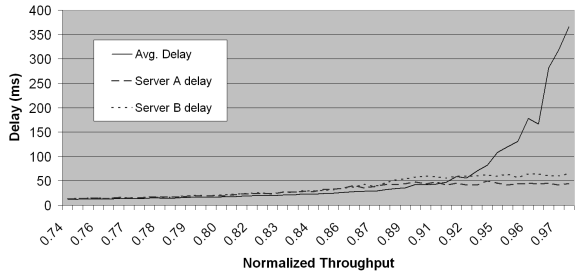

Fig. 2. QoS/HyTDMA-HTS with QoS mechanism in action

It can be clearly seen that now that the two high-traffic stations have high priorities, their QoS demands are fulfilled according to Table 1: as seen from the graph, station B has a packet delay which just exceeds the station's DPD (50 msec), whereas station A stays well below the set DPD.

Of course, as said before, successfully achieving to keep the mean packet delay of the two high traffic stations nearly constant under so high traffic conditions, inevitably implies a cost in network terms: the average packet delay of all the low-priority stations increases by a non-negligible factor (in this case 20-25%), which is out weighted by the resulting gain, especially since the stations affected are not of utmost importance.

6. QoS/HyTDMA-HTS performance under non-bursty traffic

While QoS/HyTDMA-HTS is a protocol mainly designed to efficiently operate in a computer LAN, the algorithms it employs enable it to also operate in environments where the traffic generated does not have bursty characteristics. So, for the sake of completeness, we simulated a network with two main server stations which produce much traffic, and tested how well QoS/HyTDMA-HTS manages to keep the packet delays of the two important stations under control (i.e. within the specified accepted range), even under a very high load.

Again like before, the two station's DPD is set to 50 msec and their priorities are 6 and 10 (for stations B and A, respectively). To make the comparison clear, fig. 3 has the simulation results without priorities and fig. 4 with the above priorities enabled.

From the simulation results in fig. 3, we see that stations A and B, not having any special treatment due to their low priority, are heavily penalized in order for their traffic transmission rate to be controlled, which results in their packet delays to greatly rise compared to the much lower average packet delay of all other stations.

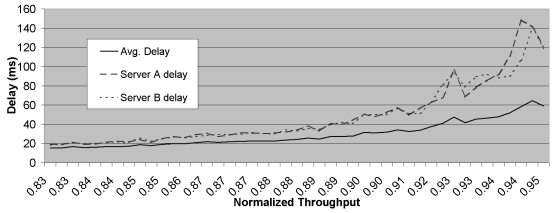

Fig 3. QoS/HyTDMA-HTS, no QoS enabled, under non-bursty traffic

But, as soon as the two station's priorities are set to a high value, we observe (fig. 4) that while the average packet delay rises as the load gets high, the two server stations keep their packet delays almost constant, within the ranges set by their priorities and Table 1.

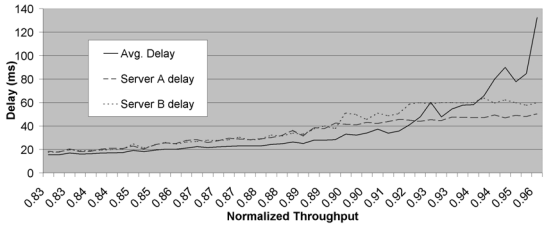

Fig 4. QoS/HyTDMA-HTS, with QoS enabled, under non-bursty traffic

In this case, the 58 and 33 percent gain in performance for stations A and B respectively, has the effect of rising about 33 percent the average delay of all low-priority stations.

7. QoS/HyTDMA-HTS performance simulation results using real traffic traces

In this paragraph, the simulation results for QoS/HyTDMA-HTS vs. HyTDMA-HTS and VPL/LTDMA using real traffic traces are presented. The traffic was gathered from a LAN of our university which consists of about a hundred hosts.

Among these hosts there are two main servers which run services for the students like mail, ftp, web, telnet, ssh, etc. These servers produce the main amount of traffic on the LAN, whereas the rest of the client stations contribute occasional bursts of traffic to the LAN.

In the simulations run, the average packet delay for the two servers was calculated, along with the average packet delay for the rest of the stations, using the three protocols mentioned above. VPL/LTDMA was chosen because it was the only protocol which could directly compete with QoS/HyTDMA-HTS since it has an

embedded learning-automata [5], [6] mechanism in order to give priority to stations with much traffic to transmit, although it is not based on administratively set priorities like QoS/HyTDMA-HTS.

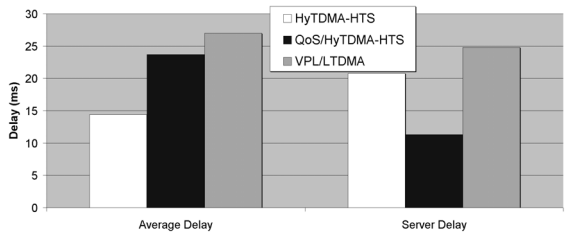

Fig. 5. QoS/HyTDMA-HTS vs. HyTDMA-HTS and VPL/LTDMA using real traffic

What we can here see, is what was theoretically expected. HyTDMA-HTS using its traffic shaping mechanisms, achieved to keep the average packet delay very low by penalizing the stations which produced much traffic (i.e. the two servers) and that explains the corresponding high server delay column.

VPL/LTDMA had the higher average packet delay and this is because its station selection scheme had it difficult to conform to the highly bursty character of the traffic and some bandwidth was lost due to idle timeslots. On the other hand, the server delay for VPL/LTDMA was somewhat better than the average delay because the learning automata of its selection scheme spotted out the traffic producing stations, giving them priority over the others.

Lastly, QoS/HyTDMA-HTS achieves to keep the server delay at very low levels (far lower than the other two protocols) with the minor (in our case study) side-effect of a relatively high average packet delay for the low priority (unimportant) stations.

In related work [7], it has been proved that HyTDMA-HTS performs better than known MAC sub-layer protocols like VPL/LTDMA [6], TDMA [4] and HyTDMA. TDMA was not included in our simulations since it is a static allocation protocol which cannot perform in an environment where the traffic is not almost uniformly sourcing from all the stations, but instead, a couple of stations produce the main volume of traffic on the LAN [2].

8. Discussion on the simulation results and the traffic models used

First of all, the bursty traffic model which was used for the first set of simulations presented in section 5, was a widely used model [10], [11] for representing a typical high load computer network. The only difference was that two stations were selected about 20 to 30 percent more often than the others, and in addition, their packet sizes were 20 to 30 percent larger than the packets of the other stations. These percentages were selected such based on real measurements which were taken from various LANs in the university by differentiating the packets sent by the servers of the LAN from the rest stations. This way, we calculated that a server in a typical LAN transmits about 20-30 percent more often that the client stations, and the packets are also about 20-30 percent larger due to hosts accessing the server's ftp service, remote file-system service, or some other service which involves the transmission of large quantities of data.

The results using this model were quite impressive (figs. 1 and 2). When the QoS modules were inactive because no priorities had been set, the two server's packet delays followed the upward trend of the average packet delay, as the network load got higher (fig. 1). As soon as we set the priorities and the desired packet delays (DPD) for the two servers, the protocol achieved to keep the delays very low (and always within the wanted range of Table 1) as compared to the average delay of the low priority stations which continued to rise as the load rose (fig. 2).

The second model used was a non-bursty one, like the models used in related work [6], [7]. It was nevertheless modified in order not to produce uniformly distributed traffic among the stations, but instead traffic sourcing by a large percent from a couple of stations (i.e. the servers of the LAN running the time-sensitive applications like VoIP).

Again as before, the results showed that our protocol successfully retained a very low packet delay for the two servers, although due to the high load, the average packet delay would continue rising (fig. 4). The desired packet delay for station A which was the station with the maximum priority (10) was not exceeded at all, whereas for the station B with priority 6, as seen from the graph, DPD was exceeded about 20 percent under the highest load, which still is far away from the Table 1 limit (50 percent).

About the last simulation model with the real traffic traces, we need to say that however accurate are the results when using traffic models, still, using real traffic generated on a real LAN gives even more realistic results. Using a method described in [6] to collect the traffic traces from the LAN, we found out what was to be expected: About 5 percent of the stations on the LAN produced about 70 percent of the traffic.

Figure 6 shows together the traffic distributions for the three traffic models which were used:

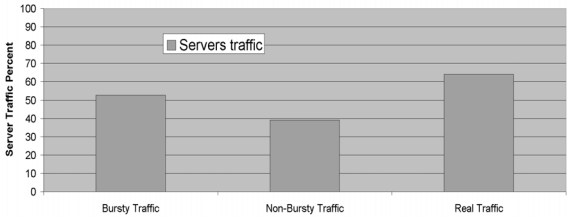

Fig 6. Traffic distribution between servers and clients

The columns represent the percentage of the servers traffic within the whole traffic of the local network. As we see, in most cases the traffic produced by the main servers consists the main volume of traffic transmitted on a network.

In such an unequal traffic distribution environment, it is very probable that low-traffic stations will starve for bandwidth and high-traffic stations will monopolize the common channel usage, unless the MAC protocol employs some sort of mechanism for distributing bandwidth in an equal manner.

So, the simulation results on the real traffic traces exactly revealed the double function of QoS/HyTDMA-HTS: First, with its traffic shaping mechanism achieves not to let the high-traffic stations take over the common channel by applying penalties (this is seen when QoS/HyTDMA-HTS is run without station priorities, so that the QoS modules are disabled). And second, when stations are distinguished to high-priority and low-priority ones, then the protocol adjusts the bandwidth usage according to each station's priority, desired packet delay and traffic.

9. Conclusion and discussion

Our protocol is proved to maintain a very stable and constant packet delay for the important stations through its prioritization scheme, thus eliminating the jitter phenomenon which plagues real-time services. It also gives the ability to stations to self adjust their desired QoS access characteristics [12].

Future enhancements to the protocol could include optimizing it for use with wireless networks, incorporating the ability for monitoring and adjusting more QoS metrics (e.g. delay jitter, loss ratio and cost [13]), or finding more effective ways for implementing the traffic shaping and the QoS adjustment mechanisms.

Closing, we should remark that the widespread use of real-time services (with VoIP above all) in conjunction with the rising use of shared medium protocols like wireless, demand protocols which can efficiently distribute bandwidth and at the same time deliver QoS, features for which QoS/HyTDMA-HTS seems to be heading in the right direction.

REFERENCES

[1] J.H. James, "Implementing VoIP: A Voice Transmission Performance Progress Report", IEEE Communications Magazine, July 2004, vol. 42, no. 7, pp. 36

[2] Andrew S. Tanenbaum, Computer Networks, Prentice Hall, 3rd edition, chapter 4.

[3] HSIAO-KUANG WU, "Dynamic QoS Allocation for Multimedia Ad Hoc Wireless Networks", IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) Project

[4] I. Rubin, "Access control disciplines for multi-access communications channels: Reservation and TDMA schemes," IEEE Trans. Inform. Theory, vol. 25, pp. 516-526, 1979.

[5] G.I. Papadimitriou, A.S. Pomportsis, Learning-automata-based TDMA protocols for broadcast communication systems with bursty traffic, IEEE Communications Letters, Vol. 4, No. 3, March 2000

[6] G.I. Papadimitriou and G. D. Pallas, A Self-Adaptive Protocol for Broadcast LANs with Variable Packet Length, IEEE Communications Letters, vol.8, no.1, pp.72-74, January 2004

[7] G. D. Pallas, G. I. Papadimitriou, Hybrid TDMA with Heuristic Traffic Shaping: An efficient bandwidth allocation approach for heavy loaded LANs, publication pending

[8] D. Grakanin, "Quality of Service for Networked Virtual Environments", IEEE Communications, April 2004, vol. 42, no. 4, pp. 42-48

[9] IEEE Standard 802.3. Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications. IEEE, New York, NY, October 2000.

[10] S. L. Danielsen, C. Joergensen, B. Mikkelsen, and K. E. Stubkjaer, "Analysis of a WDM packet switch with improved performance under bursty traffic conditions due to tunable wavelength converters," J. Lightwave Technol., vol. 16, pp. 729–735, May 1998.

[11] M. W. McKinnon, G. N. Rouskas, and H. G. Perros, "Performance analysis of a photonic single-hop ATM switch architecture with tunable transmitters and fixed frequency receivers," Perform. Eval., vol. 33, no. 5, pp. 113–136, June 1998.

[12] C. Politis, "Cooperative Networks for the Future Wireless World", IEEE Communications Magazine, Sept. 2004, vol. 42, no. 9, pp. 70-79

[13] Abdullah N. Alghannam , Michael E. Woodward , J. E. Mellor, Security As A QoS Routing Issue, PG Net 2001 Symposium