A Self-Adaptive Protocol for Broadcast LANs with Variable Packet Length

G.I. Papadimitriou, Senior Member IEEE, G. D. Pallas, Student Member IEEE, and A.S. Pomportsis, Member IEEE

The authors are with the Department of Informatics, Aristotle University, 54006 Thessaloniki, Greece.

E-mail: gp@csd.auth.gr

Abstract

In this letter, a new medium-access sublayer self-adaptive protocol for broadcast networks is introduced. This new protocol is capable of operating efficiently under heavily bursty traffic conditions and can handle stations which transmit packets of arbitrary length. TDMA, RTDMA and other fixed-assignment protocols have various drawbacks, that learning protocols such as LTDMA are known to resolve. But as a result of LTDMA's fixed-sized timeslots, reduced channel usage, wasted timeslots, as well as network equipment overloading can occur. The VPL/LTDMA protocol which is here introduced is tested using real LAN traffic traces and is proved to be successfully resolving the above issues. In addition it achieves a high throughput-delay performance when operating under realistic traffic conditions.

1. Introduction

Broadcast networks present us with the key issue of determining who gets to use the common channel. Learning automata protocols such as LTDMA [1], address this issue by assigning to stations different choice probabilities, so that each station has the proper bandwidth to best suit its needs.

But LTDMA, although taking the right decisions about who gets control of the channel, has a number of significant weaknesses: a) The timeslots are fixed-sized, resulting in stations which have to shape their traffic in specific sized datagrams, thus wasting a fraction of bandwidth transmitting dummy padding bits. Not only that, but breaking data into fixed-sized packets also introduces a huge overhead for end stations as well as network active components such as routers. And b) When LTDMA happens to select for transmission an idle station, a whole timeslot is wasted.

This letter presents a new protocol which is proved to overcome the above drawbacks. VPL/LTDMA (Variable Packet Length LTDMA), outperforms all other TDMA protocols, permits stations to transmit arbitrarily big data packets, fairly (according to their needs) distributes the available bandwidth among them, and uses a station-to-transmit scheme, based on learning automata [2]-[5].

The letter is organized as follows: In Section 2, the basic concept of the new protocol is introduced. The inner workings of the choice-probability assigning algorithm is presented in Section 3, followed by Section 4, presenting the necessity of introducing a new protocol parameter, the "idle timeslot". Section 5 deals again with the idle timeslot parameters from the aspect of minimizing its size. In Section 6, we present the method of obtaining the traffic traces for use with the simulations of VPL/LTDMA and in Section 7 we summarize the results of the simulations conducted. Finally, the main characteristics of VPL/LTDMA are given in section 8.

2. The VPL/LTDMA protocol concept

VPL/LTDMA is based on the concept that a station currently transmitting packets, will also continue to transmit packets in the near future. The contrary also applies, i.e. a currently idle station will probably stay idle for a while. These interesting properties are a direct consequence of the fact that traffic generated by computers is highly bursty as Tanenbaum suggests [13] (see also Section 6).

It is possible to take advantage of this fact, by implementing a station-to-transmit selection scheme, which assigns different choice probabilities to each station, so that stations which send traffic across the network are assigned a high choice probability, whereas idle stations are assigned a low choice probability.

So, each time VPL/LTDMA selects a station to transmit but the station does not have packets in its queue, its choice probability is decreased. This means that during the next timeslots, the station will have reduced probability to be selected resulting in fewer wasted timeslots and the saved bandwidth will be allocated to stations which need it.

On the contrary, when a station is selected for transmission and, indeed, transmits a packet, then its

1530-1346/03 $17.00 © 2003 IEEE

choice probability is increased, as it is highly probable that more packets will follow. It is obvious that such a station selection scheme results in a better channel usage and smaller transmission delays per packet.

3. The VPL/LTDMA learning algorithm

This section describes the actual algorithm according to which the choice probabilities for each station are updated during the function of the new protocol.

According to VPL/LTDMA, each station is provided with a learning automaton containing the basic choice probability P_i of each station u_i, for $i = 1 \ldots N$, where N is the total number of stations. During the initialization process, all stations are assigned the same choice probability, equal to 1/N.

The station selection scheme operates in four discrete steps as follows: a) at each timeslot, using a common pseudorandom number generator initialized with a common seed, all stations produce the same "random" number in the range (0, 1).

b) Next, all stations normalize their probability vectors according to the formula:

$$\Pi_i = P_i / \sum_{k=1}^{N} P_k$$

c) The normalized choice probabilities in vector Π_i are added starting with the probability of station 1, until the sum reaches or exceeds the random number chosen in the first step. When this happens, the selected station is the station whose choice probability was last added. This method ensures that all stations are selected according to the traffic patterns they have produced in the near past.

Next, comes the most important part of the new protocol: The updating of the station's choice probabilities. After a station has been selected for transmission, all stations update this station's choice probability in their automata, according to the network feedback information.

Depending on whether the selected station actually transmitted a packet or not, the following choice probability updating scheme is used: (supposing that u_i was the selected station)

$P_i \leftarrow P_i + L(1 - P_i)$, if u_i transmitted a packet

$P_i \leftarrow P_i - L(P_i - a)$, if u_i did not transmit a packet,
where $L, a \in (0,1)$

In case the selected station had a packet to transmit, it will probably have packets to transmit in the near future. Therefore, its choice probability is increased. If the station did not transmit a packet, it is probable that it will stay silent for a while, therefore its choice probability is decreased.

Two parameters that need further explanation are the protocol constants "L" and "a". It is obvious from the structure of the above scheme that, the bigger L is, the quicker is the choice probability convergence. Since we are dealing with highly bursty traffic, it is quite reasonable to set L to a relatively high value, so that when for example a transmitting station turns idle, its choice probability quickly decreases. Likewise, when this station turns again active, the probability should quickly increase to catch on to the traffic that the station is going to produce and has meanwhile produced.

Parameter "a" plays the crucial role of not letting the choice probability of a station which remained idle for a long time, to converge to zero. If the choice probability of a station converged to zero, then, when it would generate traffic, the automata wouldn't be able to sense the transition.

Let us take a look at the equation which decreases the choice probability of an idle station: $P_{i,t+1} \leftarrow P_{i,t} - L(P_{i,t} - a)$. It is a recursive equation, and when solved, it yields:

$P_{i,t} = a + (k - a)(1 - L)^t$, where "k" is the initial choice probability ($k = P_{i,1}$), and "t" is the t^{th} timeslot.

Now we can see what happens to the choice probability of a station u_i, if it is selected for a number of times but has nothing to transmit:

$\lim_{t \to \infty} P_{i,t} = \lim_{t \to \infty} [a + (k - a)(1 - L)^t] = a$,
since $0 < L < 1$, and thus $0 < 1 - L < 1$.

What we have accomplished, is that a station which has long been idle, when it will turn active again, it will be selected to transmit, although with a very low probability. Then, its choice probability will be increased, and very quickly, the station will send all the traffic is has meanwhile generated.

About the exact values of the parameters 'L' and 'a', the simulations have shown that L should be in the range (0.8, 0.95) for best adaptability and thus performance. About 'a', the value should be in the range (0.01, 0.05), although the exact value in not very important considering the very abrupt nature of the traffic being simulated.

One should observe about VPL/LTDMA, that the protocol is collision free, although it requires absolutely no central coordination. This is achieved since we have a broadcast network and thus, the feedback information is common for all stations. That way it is possible that all stations always have the same choice probabilities in their automata. And since there is a common pseudorandom number generator, seeded with a common initial value, all stations always select the same station for transmission.

4. The "idle timeslot" parameter

Deciding to design a new protocol which would not use fixed-sized timeslots, we were confronted with the problem of how to make stations aware of the transmition completion of the selected station. LTDMA for example does not have such a problem since the timeslot is known to all stations. But VPL/LTDMA, on the other hand, uses arbitrary sized timeslots depending on the needs of each station, and thus each transmission is completed in a variable time period, which is not a priori known to all stations.

In our case, the real problem is the finite speed of the data propagation on the common channel. Since a bit transmitted by a station does not reach all other stations in an instant, it is obvious that all the stations will understand whether the selected station had a packet to transmit or not, not before a certain period of time has passed.

All stations update their automata as soon as they receive the first bit from the station that was selected to transmit. But taking into consideration what we mentioned above, how could the protocol identify a station which has nothing to transmit?

In order to keep the desired property of decentralization in VPL/LTDMA, the only plausible way to make stations aware that a selected station had nothing to transmit is to introduce a new protocol parameter, the "idle timeslot". If a station is selected, although it doesn't have any packets to transmit, then it remains silent for a fixed time period (more details in the next section), so that the other stations can update their automata accordingly: decrease the station's choice probability.

5. The size of the "idle timeslot"

It is of no doubt that the introduction of the idle timeslot results in an increased overhead for our protocol. Consequently, it is vital to minimize the size of the idle timeslot as much as we can without disturbing the normal protocol function.

If we suppose that our broadcast network is based on copper wiring (which is most often the case), we know that data can be transmitted at an average speed of about 200.000 Km/sec. That means that a single bit would require about 5 μsec to traverse a LAN of 1 Km. This should be the minimum size of the idle timeslot or else the stations would loose synchronization of their automata. Fig. 1 will clarify this point.

Figure 1. Bit transmission latency

As we said before, a single bit transmitted from station A, needs 5 μsec to reach station C and 4 μsec to reach station B. Let us suppose that stations A and C are the most remote stations on the network and that all stations have selected station A for transmission. It is obvious that if the size of the idle timeslot was to be less than 5 μsec, the very first bits from station A would traverse the LAN, would reach station B and station B would update its automaton increasing choice probability for station A. But the bits would reach station C after the idle timeslot period had expired and thus, station C would have decreased the choice probability for station A. This inconsistency between station B's and station C's automata would later result in collisions in the LAN because for each timeslot not all stations would select the same station for transmission. So, it is clear that the idle timeslot period would have to be at least the time needed for a bit to traverse the LAN.

6. Collection of real traffic traces for the simulations

Usually, when a new protocol is proposed, it is simulated assuming a certain traffic model. For example, supposing that we have bursty traffic, we create a model where each station is either in an idle state or in an active state with different state transition probabilities. And a station in an active state is generating a packet at each timeslot with a specific probability.

The question is why should we use such an abstraction of reality to simulate a new protocol, when there is the opportunity to use real traffic traces, collected from a LAN. A simulation using LAN traffic offers a more realistic view upon the new protocol's characteristics, and helps us not to fall into the perspective trap: It is obvious that a certain protocol will yield desirable results when simulated with a model producing exactly the data that the protocol was designed for. This is why many lab-tested products fail in real life.

For our simulations, we used the traffic produced by our department's LAN, which consists of about a hundred computers always connected to the network. Among these computers are a few mail, web, ftp and database servers which always keep the traffic to a minimum level. We collected our traffic samples during the day, when all computer labs are open to the students and consequently due to p2p programs and other downloads, the traffic is high.

A linux server on the department's LAN was used to collect the traces using the well-known "tcpdump" [12] unix tool. Tcpdump sets the server's Ethernet card to promiscuous mode, so that the card captures from the broadcast network all packets, including those which do not have the server as destination IP.

We can see below a diagram showing exactly what Tanenbaum suggests: The traffic on the LAN is highly bursty [13]. There are peaks reaching about 1300 packets/sec arrival rate, and moments when packet arrivals are minimal.

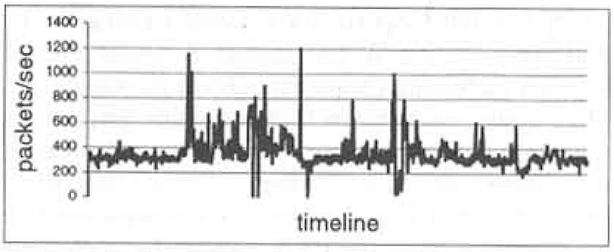

Figure 2. The bursty character of a LAN's traffic

The evidently bursty character of the traffic makes it very difficult for non-adaptive protocols like TDMA to achieve low delay times, since certain stations which have many packets to transmit in their queues will have to wait for the protocol to give timeslots to other stations, which possibly, have nothing to transmit. Here's the corresponding figure of TDMA's responsiveness to figure's 2 traffic:

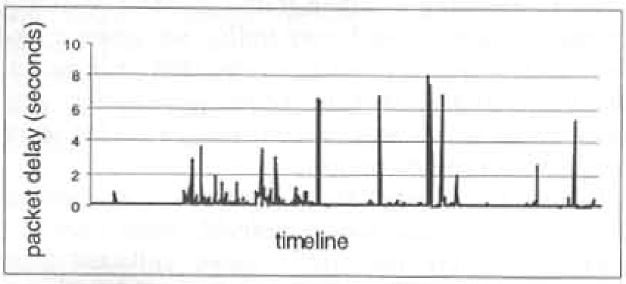

Figure 3. Packet delay for TDMA

With VPL/LTDMA this does not happen. In case a station produces a traffic burst, the protocol self-adapts and gives high selection priority to the station for as long as the burst continues. After the station turns idle again, the protocol assigns to the station a low selection priority. We can see that on the next figure.

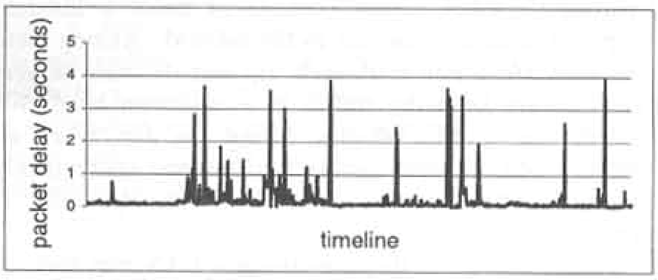

Figure 4. Packet delay for VPL/LTDMA

We observe that when traffic bursts occur, the delays inevitably grow bigger, but VPL/LTDMA never delayed the packets more than 5 seconds, whereas TDMA's round robin mechanism had delays as large as 10 seconds.

7. Simulation results

This section summarizes the results from a series of simulations of the new protocol. VPL/LTDMA was compared to protocols TDMA [6]-[10] and RTDMA [11], two representative TDMA protocols. Because TDMA and RTDMA use fixed-sized timeslots, they were converted so as to be able to handle packets of arbitrary size.

The converted TDMA uses a round-robin transmit scheme, i.e. all stations transmit the one after the other, and each transmission is followed by an idle timeslot, so that the next station knows when to start its own transmition. The converted RTDMA is basically similar, with the only difference being that the station which transmits every time is selected randomly using a common pseudorandom number generator seeded with a common initial value between the stations. Of course, the random selection of stations regardless of their activity or inactivity has a deep impact in the protocol's performance, which is shown on the figures which follow.

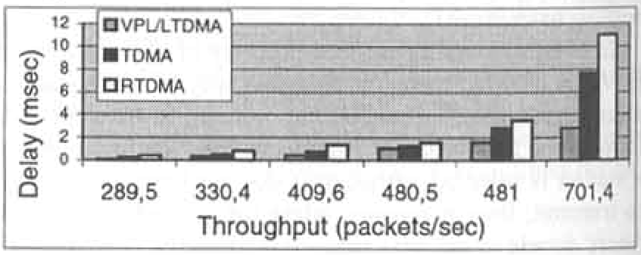

Figure 5. Delay versus throughput (5 μsec idle timeslot)

Figure 5 shows us the simulation results for the three protocols, using a 5 μsec idle timeslot. Six independent simulations were conducted, each one with a different network load. VPL/LTDMA has a clearly better performance, since the delay is only a fraction of the delay of the other protocols. Therefore, it becomes evident that VPL/LTDMA manages the available bandwidth very efficiently, and the delay is little affected even when the load more than doubles. Note: There are two simulations with almost the same throughput (480,5 and 481 packets/sec) but different delay results, because during the trace collection, a DoS attack with very large ICMP packets against a host on our LAN occurred.

Next figure shows the results for an 8 μsec idle timeslot.

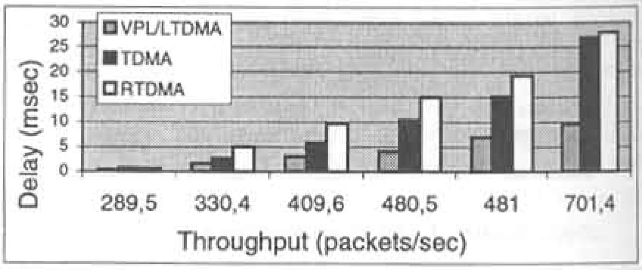

Figure 6. Delay versus throughput (8 μsec idle timeslot)

Figure 6 shows what we have expected: There is an obvious degradation in the performance of all protocols due to the larger idle timeslot period. But in spite of this general performance degradation, VPL/LTDMA always has an advantage over the other two protocols and it is worth observing that VPL/LTDMA's delay is increasing with a much slower rate as the traffic increases than the other two protocols.

8. Conclusion

A new protocol has been presented, able to operate in most of today's networks, supporting arbitrary length packets, high throughput-delay performance and a learning algorithm in order to manage and allocate effiently the available bandwidth. The new protocol is fully decentralized and requires no central coordination of the stations. The simulations which were conducted using realistic traffic from a LAN proved that VPL/LTDMA performs far better than other TDMA protocols like RTDMA, even under heavy and bursty traffic conditions. The most important characteristics of the new protocol are summarized below:

1. Support for arbitrary packet length which results in better channel usage.
2. No packet fragmentation takes place, resulting in less overhead for stations as well as network active components.
3. The bandwidth that is being wasted in case the protocol selects for transmission an idle station is insignificant.
4. A high performance is achieved, even when the offered traffic is bursty.
5. The average per packet delay is low, and very little affected by large throughput increasing.
6. A fraction of the available bandwidth is assigned to each station, proportional to the station's changing needs.
7. VPL/LTDMA is fully distributed: No centralized control of the stations is required.
8. The "idle timeslot" parameter can be adjusted to the network's special characteristics, in order to achieve better performance.

References

[1] G. I. Papadimitriou and A. S. Pomportsis, "Learning-Automata-Based TDMA Protocols for Broadcast Communication Systems with Bursty Traffic", IEEE *Communications Letters*, vol. 4, no. 3, March 2000

[2] P.Nicopolitidis, G.I.Papadimitriou and A.S.Pomportsis, "Using Learning Automata for Adaptive Push-Based Data Broadcasting in Asymmetric Wireless Environments", IEEE Transactions on Vehicular Technology, to appear.

[3] G.I.Papadimitriou and A.S.Pomportsis, "Self-Adaptive TDMA Protocols for WDM Star Networks: A Learning-Automata-Based Approach", IEEE Photonics Technology Letters, vol.11, no.10, pp. 1322-1324, October 1999.

[4] G.I.Papadimitriou and D.G.Maritsas, "Learning Automata-Based Receiver Conflict Avoidance Algorithms for WDM Broadcast-and-Select Star Networks", IEEE/ACM Transactions on Networking, vol.4, no.3, pp.407-412, June 1996.

[5] K. Najim and A. S. Poznyak, *Learning Automata: Theory and Applications*. New York, NY: Pergamon, 1994.

[6] J.M. Capone and I. Stavrakakis, "Delivering QoS requirements to traffic with diverse tolerances in a TDMA environment", IEEE/ACM *Trans. Networking*, vol 7, pp 75-87, Feb. 1999.

[7] W. Stallings, *Local and Metropolitan Area Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[8] I. Rubin and J. Baker, "Media access control for high speed local area and metropolitan networks", Proc. IEEE, vol. 78, pp. 168-203, Jan. 1990.

[9] G. Anastasi, D. Grillo, and L. Lenzini, "An access protocol for speech/data/video integration in TDMA-based advanced mobile systems", IEEE J. *Select. Areas Commun.*, vol. 15, pp. 1498-1509, Oct. 1997

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

[11] A. Ganz and Z. Koren, "Performance and design evaluation of WDM stars", J. Lightwave Technol., vol 11, pp. 358-366, Feb 1993.

[12] Linux.org documentation team, "Tcpdump documentation and applications", http://www.linux.org

[13] A. S. Tanenbaum, *Computer Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1996.