DQLAN - A DQRAP Based LAN Protocol

Chien-Ting Wu
Computer Science Department
Illinois Institute of Technology
wuchien@iitvax.iit.edu

Graham Campbell
Computer Science Department
Illinois Institute of Technology
campbell@iitmax.iit.edu

Abstract*[1] *- DQRAP approaches the performance of an idealized M/D/1 queueing system. It provides the theoretical basis for developing practical protocols that can operate in diverse multiaccess channel environments such as LANs, MANs, WANs and satellite networks. In this paper we present DQLAN, a DQRAP based protocol that supports variable length frames in a high-speed LAN environment. The DQLAN protocol is described and simulation results for 10, 100 Mbps, and 1 Gbps versions are presented. The simulation results at 10 Mbps are compared with CSMA/CD.

1 Introduction

DQRAP (Distributed Queueing Random Access Protocol) is contention-based medium access control (MAC) protocol developed at the Illinois Institute of Technology. It is described by Xu [1] and Xu and Campbell [2]. DQRAP provides performance with respect to throughput and delay that is superior to all other contention-based MACs. The performance is such that O'Connell discusses DQRAP in a companion paper at this workshop as the basis for his suggestion that MAC protocols should be revisited as the means of supporting high performance computing [3]. O'Connell discusses DQRAP in the context of a slotted channel that in practice requires segmentation of all data. In this paper we present DQLAN, a version of DQRAP that supports variable length frames. We propose DQLAN as both a conventional LAN and as the highly efficient switching fabric that in a parallel processing environment could support processor to processor transfer of variable length blocks of data at rates of 100 Mbps over a distance of one km and 1 Gbps over 100 meters. Section 2 provides a brief description of DQRAP. Section 3 describes the DQLAN protocol. Section 4 compares, using simulation, the performance of DQLAN and CSMA/CD at 10 Mbps and simulation results for DQLAN at 100 Mbps and 1 Gbps. Section 5 presents conclusions.

2 DQRAP Basics

DQRAP utilizes a channel divided into fixed-length slots. Each slot consists of m CMS' (control minislots) and one single data slot. The CMS' are used to resolve contention and the DS is used to send data. In DQRAP, each node maintains two binary counters that respectively represent the transmisison queue TQ and the collision resolution queue RQ. TQ indicates how many packets are already queued in sequence and are waiting to be transmitted. RQ indicates how many groups of packets are trying to resolve a contention. The values of these two binary counters are common in all nodes.

The protocol consists of three rules: data transmission rules (DTR), request transmission rules (RTR), and queueing discipline rules (QDR). DTR controls when a node can transmit into the data slot. Nodes obey the RTR to send an initial request into the CMS' and subsequently to resolve contention. When nodes receive feedback from the CMS', they use QDR to update the TQ and RQ and to also record their position in the queue.

The key feature is the use of CMS' to reserve a data slot, and if a collision occurs in the reservation process to use the same CMS' to sort things out. The reservation and collision resolution processes operate in parallel with data transmission. This results in increased channel utilization and decreased packet delay as compared with all other contention protocols. The required bandwidth of the CMS' will typically be in the 3% - 10% range.

DQRAP moves seamlessly between immediate access at light offered loads to a pure reservation process at heavier offered loads. When the load is light, a very high percentage of the packets are sent immediately as with Aloha and CSMA. When the load is high, most packets are transmitted after making a reservation. DQRAP throughput reaches an instantaneous value of one at an offered load of one or higher. The analytical proof of this is shown by Zhang [4] but it can be explained intuitively. Collisions in data slots occur only when the immediate access feature is used, i.e., when the TQ (transmission queue) is zero. From queuing theory we know that as the offered load approaches one the queue grows to infinity, thus ensuring that the TQ is always non-zero. A non-zero TQ forces all data transmission to utilize the reservation process, there are no collisions in the data slot and thus a throughput of one is achieved. The performance of DQRAP approaches that of an idealized M/D/1 system. The average delay for DQRAP is 13.5 with load at 0.95. At the same load, the average delay for an idealized M/D/1 system is 11.0.

Standard DQRAP works well in an environment where "a" < 0.5 ("a" is the ratio of maximum propagation delay in the network to the transmission time of a standard frame). But, when compared to Ethernet or the other LAN protocols

---

[1] Invited Paper - HiNet '95, Santara Barbara, CA, April 25, 1995.

with variable length frames, DQRAP with its fixed slot size requires segmentation of data, adding the overhead of an adaptation layer as with ATM. An alternative to using an adaptation layer is to utilize variable length frames.

A DQRAP based LAN supporting variable length data slots has been named DQLAN and is described in the next section.

3 DQLAN Protocol

3.1 Channel Model. In DQLAN, the channel is divided into slots of variable length. Each slot consists of m control-minislots (CMS) and one data-slot (DS). The length of each CMS is fixed whereas the DS has a variable length. There is no minimum length requirement for the DS. The topology of a DQLAN is shown in Figure 1. Every station connects to the headend by a pair of transmission media; one is the inbound channel and the other is the outbound channel. The headend is a passive hub. Any input from one of the inbound channels will output to all the outbound channels. Thus, although the topology is physically a star, it acts like a bus: a transmission from any one station is received by all other stations, and if the signals sent by two or more stations arrive at the headend at the same time, a collision occurs.

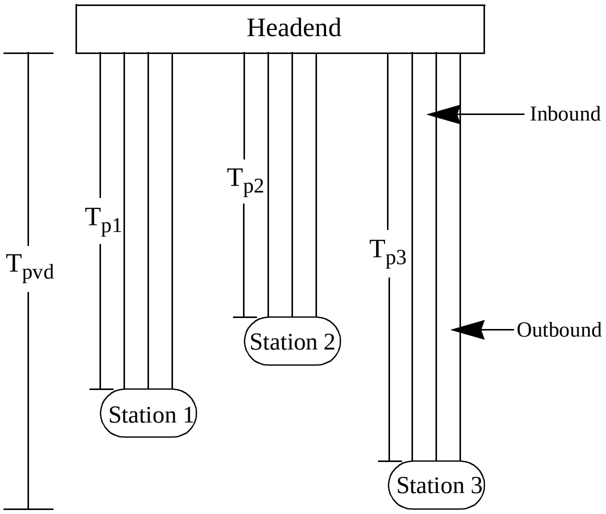

Figure 1 DQLAN Topology

3.2 Synchronization. To apply DQRAP to the channel model described in the last section, synchronization is the key factor. The requirement is that all the k^{th} CMS sent by different stations must arrive at the headend at the same time. The collided signal is then transmitted on all outbound channels and thus all the stations detect that a collision has occurred in the k^{th} CMS. Therefore, all stations transmitting into a specific minislot or data slot on the inbound channel must transmit at the correct time. The question is how they determine the starting transmission time for the first, second,..., and m^{th} CMS' and the data slot. If the transmission medium which connects the stations and the headend have the same length, there is no problem. In this case, after the stations have detected all CMS' and if the data slot is empty, then all the stations can react immediately. Otherwise, they calculate the data slot transmission time according to the length field in the data slot. The reaction time is based on the following two conditions:

1. If the calculated transmission time, T_{ds}, is less than $2T_p$ where $2T_p$ is the two-way propagation delay (from station through inbound channel to headend and then through outbound channel to the station), the station reacts immediately.
2. If the data slot transmission time is greater than $2T_p$, the station waits $T_{ds} - 2T_p$ to react.

But, in the real word the cable length for each station could be different. A ranging process is used to move all stations to a virtual distance beyond the furthest station. Every station uses a procedure to determine the propagation delay between itself and the headend. Assume the propagation delay for the virtual distance is T_{pvd} and the propagation delay between station i and headend is T_{pi}, then station i will delay the reaction for time $2(T_{pvd} - T_{pi})$.

Another problem is that if no station sends requests and data for a period of time, the clocks maintained by each station could drift apart. This will affect the synchronization of the following slots. This problem is addressed by utilizing an active monitor that periodically transmits a slot marker to provide timing information for all stations. Every station has the potential to be the active monitor: there is a conventional initialization procedure to select the initial active monitor and subsequently the active station becomes the active monitor.

Figure 2 shows the frame format for DQLAN. T_{sm} is the slot marker transmission time and T_{cms} is one CMS transmission time (including guard time). The frame length field indicates the length of the data slot. Its transmission time is $T_{lf.}$ T_{ds} is the data slot transmission time, including T_{lf}, and is based on the length field.

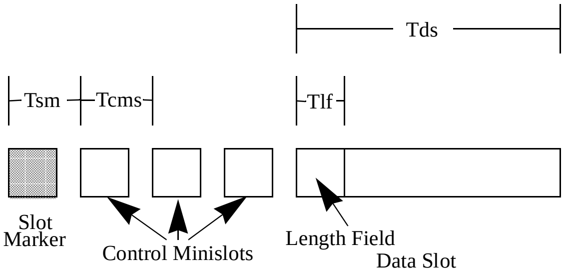

Figure 2 DQLAN Frame Format

When stations receive the length field, they calculate how long the transmitting station will continue. There is no minimum frame length requirement but if the data slot transmission time is less than $2T_{pvd}$, it will be treated as $2T_{pvd}$. There are three conditions under which the data slot transmission time will be less than $2T_{pvd}$:

1. The data slot is empty.
2. Only one station sends data in data slot and its transmission time is less than $2T_{pvd}$.
3. Two or more stations send data to the data slot. This only

happens when TQ and RQ are both equal to zero. Therefore, stations can use TQ, RQ and feedback from CMS' to determine whether there is a collision in the data slot. If it is determined that the data slot is collided, the transmitting stations will stop sending data immediately. Hence, the data slot transmission time is less than $2T_{pvd}$.

All the stations are synchronized on the slot marker. After the slot marker is detected, they detect the contents of the CMS' and determine the feedback. Using the CMS' feedback and the previous value of TQ, they determine the status of the current data slot. The status could be empty, single or collided. If the previous value of TQ is greater than zero, then the status of the data slot must be single. If the previous value of TQ is zero and there is one single status in the CMS' and the rest of the CMS' are empty, then the status of the data slot is still single. If the previous value of TQ is zero and all of the CMS' are empty, then the data slot status is empty. If the previous value of TQ is zero and if the status of one or more CMS' are collided or more than one CMS' are single, then the data slot must be collided. If the data slot is empty or collided stations treat the data slot transmission time as $2T_{pvd}$. Otherwise, they calculate the data slot transmission time based on the information in the length field. Note that stations with different cable lengthes receive the slot marker at different times and will also have different starting times for the next cycle.

When station i receives the last CMS at time t, it performs the following synchronization formula to calculate the next cycle starting time for itself:

1. According to TQ, RQ and CMS' feedback, determine the status of the data slot. If the status is empty or collided, set T_{ds} equal to $2T_{pvd}$, otherwise, wait T_{lf} to get the length information and use this information to calculate T_{ds}.
2. If $T_{ds} < 2T_{pvd}$, $T_{ds} = 2T_{pvd}$.
3. $t_{next} = t + T_{ds} + 2(T_{pvd} - T_{pi})$

If station i is the active monitor, the start of the slot marker will be sent at time t_{next}. If it has a request to send in the k^{th} CMS, the transmission will start at time $t_{next} + T_{sm} + (k - 1)T_{cms}$. The starting transmission time of the data slot is $t_{next} + T_{sm} + mT_{cms}$.

3.3 Protocol. The DQLAN protocol is similar to DQRAP except it uses a variable length data slot. We mentioned earlier that in DQRAP two distributed queues are maintained by each station: transmission queue TQ and collision resolution queue RQ. Actually, TQ and RQ are binary counters. In addition, each station maintains two indexes, TQ_{seq} and RQ_{seq}, to track their location in TQ and RQ. If TQ_{seq} or RQ_{seq} are equal to zero, this station is not involved in the transmission or resolution queue. Here, we assume that the stations have only one buffer, therefore one set of indices is adequate. A station can support multiple buffers by using multiple sets of indices.

In DQRAP, the protocol consists of three rules: (1) data transmission rule (DTR), (2) request transmission rule (RTR), and (3) queueing discipline rule (QDR). When stations detect feedback from the CMS', they use QDR to update the values of TQ and RQ and to update the TQ_{seq} and RQ_{seq} if necessary. Let F_k, k = 1,2,..m, denote feedback from the k^{th} CMS. The values of F_k can be E: empty, S: single, or C: collide. If a station transmits a request in a CMS, it enter into TQ or RQ in the order of the CMS number from 1 to m. The QDR, RTR, and DTR for DQLAN are as follows.

Queueing Discipline Rule (QDR)
```
{
  $TQ_{prev}$ = TQ;
  if (TQ > 0)
    TQ--;
  if ($TQ_{seq}$ > 0)
    $TQ_{seq}$--;
  if (RQ > 0)
    RQ--;
  if ($RQ_{seq}$ > 0)
    $RQ_{seq}$--;
  for (n_request = 0, k = 1; k <= m; k++)
    if ($F_k$ == S)
      n_request++;
    else if ($F_k$ == C)
      n_request += 2;
  for (k = 1; k <= m; k++) {
    if ($F_k$ == S) {
      if ($TQ_{prev}$ == 0 && n_request == 1) {
        /* This is the case where a station has immediate
          access.*/
        if (the station sends in DS)
          clear the buffer to empty
      }
      else {
        TQ++;
        if (the station sends request into the $k^{th}$ CMS)
          $TQ_{seq}$ = TQ;
      }
    }
    else if ($F_k$ == C) {
      RQ++;
      if (the station sends request into the $k^{th}$ CMS)
        $RQ_{seq}$ = RQ;
    }
  }
}
```

Request Transmission Rule (RTR)
```
{
  if ((RQ == 0 && the buffer is not empty && $TQ_{seq}$ == 0)
    || (RQ > 0 && $RQ_{seq}$ == 1)) {
    select a number k between 1 and m randomly
    wait (k - 1) * $T_{cms}$
    transmit request
  }
}
```

Data Transmission Rule (DTR)
```
{
  if (TQ == 0 && RQ == 0 && buffer is not empty)
```

```
    transmit data
  else if ($TQ > 0$ && $TQ_{seq} == 1$) {
    transmit data
    clear buffer to empty
  }
}
```

By using the above rules and the synchronization formula, the DQLAN protocol is described as follows:

```
DQLAN Protocol:
{
  $TQ = RQ = TQ_{seq} = RQ_{seq} = 0$;
  clear buffer to empty
  while(TRUE) {
    wait for slot marker
    receive the slot marker at time t
    detect feedback from CMS', $t = t + m * T_{cms}$
    calculate next cycle starting time $t_{next}$ based on the synchronization formula
    modify TQ, RQ, $TQ_{seq}$, $RQ_{seq}$ by using QDR
    wait for $t_{next}$
    $t = t_{next}$
    if (the station is the active monitor)
      send slot marker at time t
    $t = t + T_{sm}$
    use RTR to send request
    $t = t + m * T_{cms}$
    use DTR to send data
  }
}
```

Following is an example which shows the operation of DQLAN. In Figure 3 the channel is divided into slots of variable length. Each slot starts with one slot marker and is followed by two fixed length CMS' and one variable length data slot. The timing in the arrival axis and the transmission axis reflect the view from the headend. The first slot is empty. At time t1, the slot marker arrives and TQ and RQ are equal to zero. Both A and B send a request into the CMS and data into the data slot. Since there is a collision in the data slot, A and B cease transmitting after determining the collision. This causes the length of the data slot for this cycle to be $2T_{pvd}$, and $t2 = t1 + T_{sm} + 2T_{cms} + 2T_{pvd}$. As a result A and B enter into RQ, because their requests collided in the first CMS. In the third slot A and B send requests into the CMS only and they leave the data slot empty. This still makes the duration of the data slot equal to $2T_{pvd}$, $t3 = t2 + T_{sm} + 2T_{cms} + 2T_{pvd}$. In the third slot A and B send requests to the first and second CMS respectively, therefore A and B enter into the first and second entry of TQ respectively. TQ becomes 2. In the fourth slot, A sends data into the data slot, hence every station detects a valid length field, then calculates the next cycle starting time based on the length field. $t4 = t3 + T_{sm} + 2T_{cms} + T_{ds}$. TQ is decreased by 1 and B goes to the first entry of TQ. In the fourth slot, C sends a request to the CMS successfully and enters the second entry of TQ. TQ goes back to 2. In the fifth slot, D and E send requests to the CMS' and B sends data in the data slot. D and E enter RQ and every station calculates the next cycle starting time based on the length field sent by B. In the sixth slot, since the data transmission time for C is less than $2T_{pvd}$, using the synchronization formula, the duration for the data slot is still set to $2T_{pvd}$. A portion of the bandwidth in this data slot is unused.

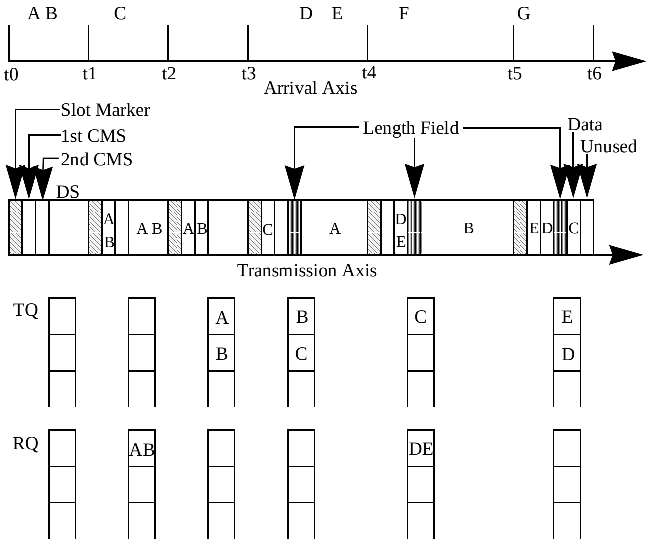

Figure 3 DQLAN Operation Example

4 Performance Evaluation

Simulation is used to analyze the utilization and average delay of DQLAN. Ethernet is the most commonly used LAN product in the world, thus CSMA/CD was chosen to provide a basis of comparison with DQLAN in the 10 Mbps environment. DQLAN and CSMA/CD were simulated using "C" for a variety of loads, different distributions of traffic, and mixes of traffic.

Since DQLAN relies on CMS' to resolve the contention, the overhead of CMS's must be considered in simulation. Methods to acquire ternary feedback in a CMS are addressed by [5]. The Combinatoric Ternary Feedback (CTFB) method is considered in DQLAN. The idea of CTFB is to assign each station a unique pattern such that the overlapping of two or more of the patterns can be distinguished from one single pattern. The patterns can be constructed using 0s and 1s. When stations send requests in CMS', each station sends its unique bit pattern into a CMS, the receiver determines the feedback status by counting the number of '1' bits appearing in each CMS. According to the Binomial theorem, if 12 bits are used to construct the bit pattern, and each station uses 6 '1' bits to identify itself, there are C(12,6) = 924 different patterns that can be assigned. This is adequate for a conventional LAN. Therefore, each CMS will contain 12 bits for a pattern and say 4 bits for guard band.

The other overhead to be considered is the slot marker. No information is carried in the slot marker but a desireable option is for the active monitor to transmit its values of TQ and RQ so that newly active stations can instantly join the network. The length of the slot marker depends on the medium and electrical/optical technology used. In the DQLAN simulation, a 16 bit length slot marker was utilized.

4.1 DQLAN and CSMA/CD Comparisons. An environment of 10 Mbps bit rate and 2.5 km cable length is used to compare the performance of DQLAN and CSMA/CD. DQLAN with 3 CMS' was simulated. Slotted 1-persistent CSMA/CD with a 512 bit slot time was simulated.

Figure 4 shows the utilization for both DQLAN and CSMA/CD plotting load versus frame length. For 128 byte frame length, the CSMA/CD efficiency is very poor. The maximum utilization achieved is 0.4 at an offered of 0.4. As the load increases beyond 0.4, the utilization decreases. For a 1024 byte frame length, the CSMA/CD efficiency is satisfactory. At a load equal to 0.9, it provides 84% utilization. For DQLAN, no matter what the load and frame length, utilization is always equal to the offered load up to the capacity of the channel. Figure 5 shows that the average delay of DQLAN increases slowly as the load or frame length increases. With CSMA/CD, when the load is light, the delay increase is similar to DQLAN. However, when the load is high, the delay increases dramatically. In the CSMA/CD simulation, the average delay did not count the frames dropped due to the number of collisions exceeding 16, so the average delay is not infinite.

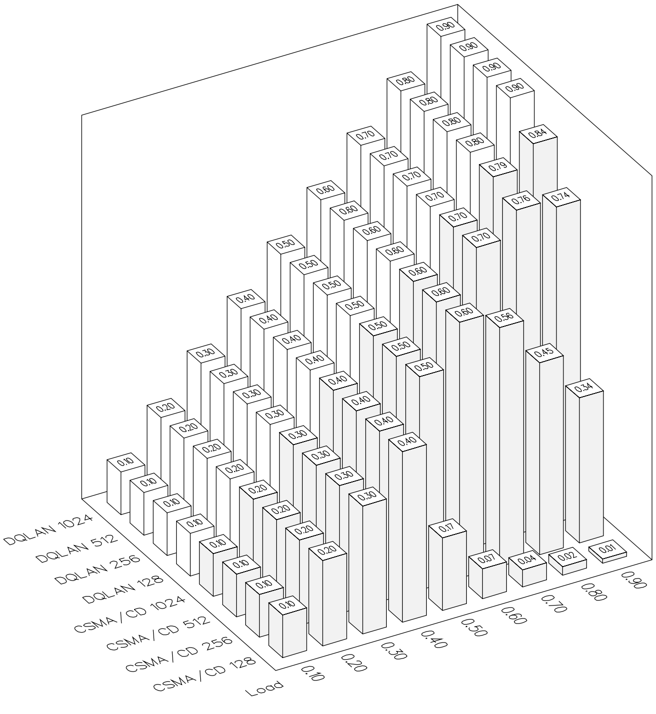

Figure 4 DQLAN & CSMA/CD Utilization
Load versus Fixed Frame Size

Figure 6 and Figure 7 show the utilization and average delay respectively for load versus variable frame length with uniform distribution. These two pictures shows that DQLAN provides better performance than CSMA/CD when the load is light, and much better performance than CSMA/CD when the load is high.

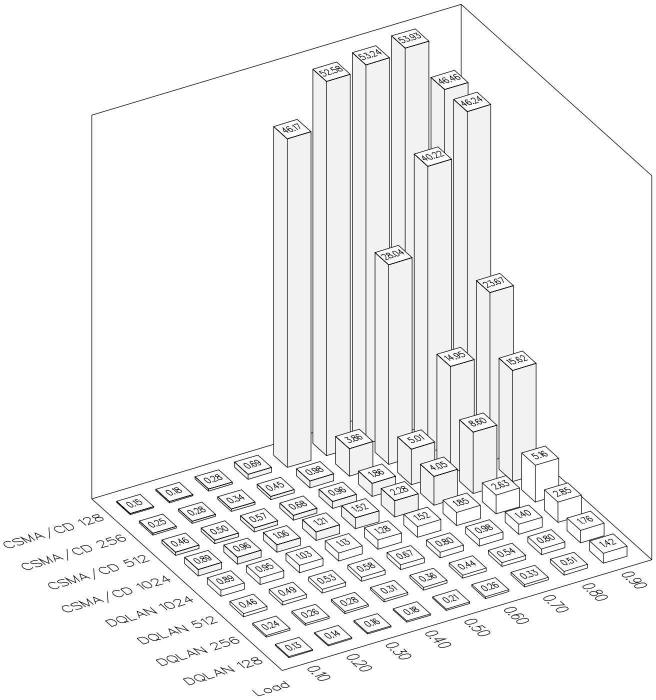

Figure 5 DQLAN & CSMA/CD Average Delay
Load versus Fixed Frame Size

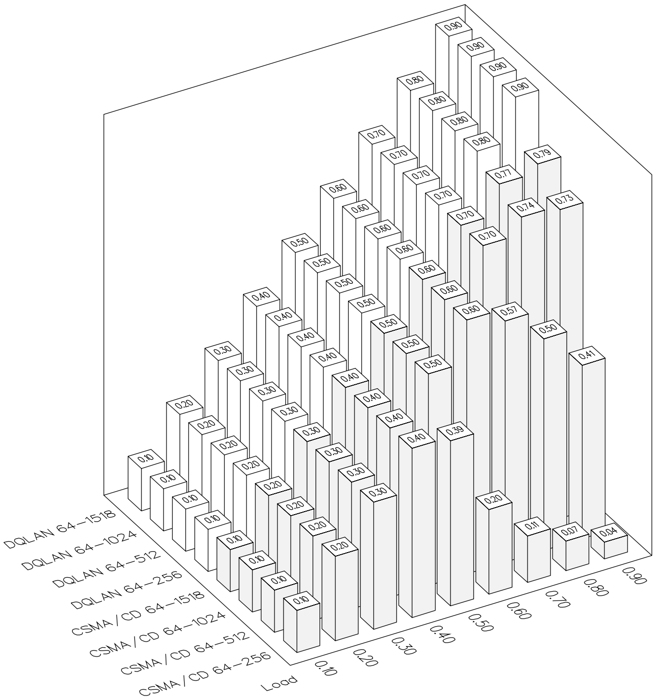

Figure 6 DQLAN & CSMA/CD Utilization
Load versus Variable Frame Size

The previous comparisons are based on one single traffic type. In today's client-server computing, in most cases a client sends a small request message to a server, and the

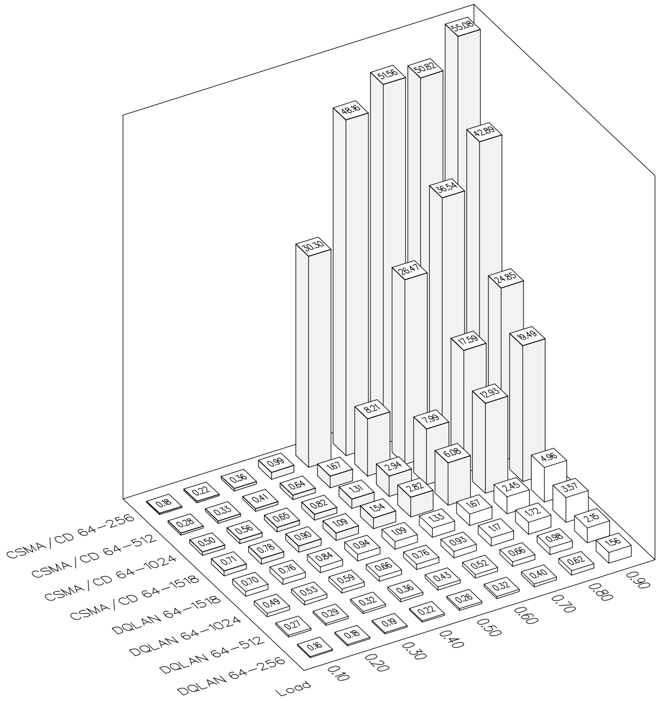

Figure 7 DQLAN & CSMA/CD Average Delay
Load versus Variable Frame Size

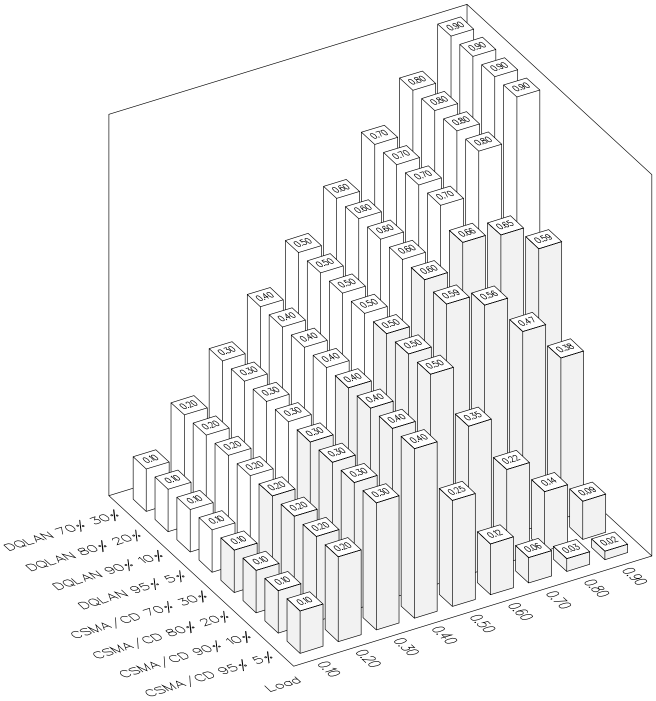

Figure 8 DQLAN & CSMA/CD Utilization
Load versus Combinations of Traffic

server returns a large reply message. Therefore, traffic can be classified as consisting of short and long frames. We assume that the short frame lengths are between 64 bytes and 128 bytes and the long frame lengths are between 512 bytes and 1518 bytes (longest frame in IEEE 802.3). Figure 8 and Figure 9 show the utilization and average delay respectively for load versus different combinations of traffic. The 95% and 5% combination means that the percentage of short frames is 95%, and the percentage of long frames is 5%. For this example, the average long frame size is about 5 times longer than the short frame size, therefore, long frame traffic generates about 1/4 of the total load. The results show that CSMA/CD can not exceed 66% utilization with 70% and 30% traffic combinations and it is worse under other traffic combinations. However, DQLAN provides solid performance under any combination of traffic, the utilization is still equal to the load given and the average delay does not jump to a high value when the load is high. Based on this comparison, we see that DQLAN outperforms CSMA/CD under all traffic patterns.

4.2 100 Mbps DQLAN Performance. The outstanding performance of DQLAN has been demonstrated in a 10 Mbps system. How about the performance of the 100 Mbps DQLAN? Will it provide good utilization up to any given load? Will the access time reduce to 1/10 of the 10 Mbps DQLAN?

To simulate a 100 Mbps DQLAN, we must consider the cable length to accommodate the change in bit rate and thus the value of "a". The IEEE 802.3 standard for 100 Mbps sets a maximum length of 250 meters, so the maximum cable length was set to 250 meters in the simulation. The slot marker was set to 16 bits, and each CMS was set to 16 bits. Table 1 shows the simulation results for the mixed traffic with 80% short and 20% long frames. The first column shows the load generated, the second column shows the utili-

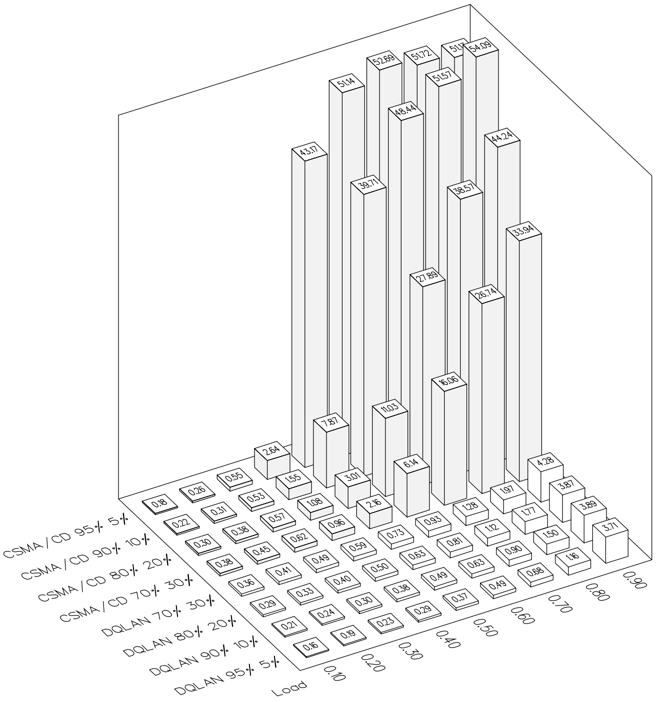

Figure 9 DQLAN & CSMA/CD Average Delay
Load versus Combinations of Traffic

zation of the 100 Mbps DQLAN. We see that the utilization is exactly the same as the load given. The third and fourth columns show the average delay in microseconds for 100 Mbps and 10 Mbps respectively. These numbers indicate that 100 Mbps DQLAN access time is close to 1/10 of 10 Mbps DQLAN access time.

Table 1 100 Mbps DQLAN Utilization and Delay

Load	100 Mbps Utilization	100 Mbps Delay	10 Mbps Delay
0.10	0.10	29	288
0.20	0.20	34	332
0.30	0.30	40	402
0.40	0.40	49	497
0.50	0.50	62	631
0.60	0.60	81	809
0.70	0.70	114	1121
0.80	0.80	188	1768
0.90	0.90	438	3870

Figure 10 shows the average delay for load versus combinations of traffic. The average delay still increases as a function of load and average frame size. This means that the major factor in the longer delay is due to the longer service time, not a longer collision resolution time. The longest average delay of 463 microseconds occurs with the 70% and 30% combination, since its average frame size is larger than the 80% and 20%, 90% and 10%, and 95% and 5% combinations.

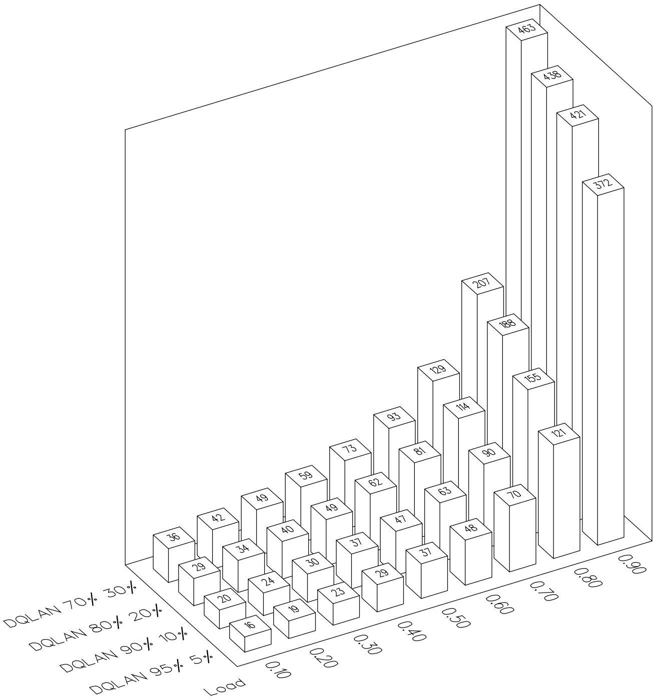

Figure 10 100 Mbps DQLAN Average Delay
Load versus Combinations of Traffic

4.3 1 Gbps DQLAN. The demand from multimedia applications will push the conventional LAN speed to 1 Gbps in the near future. So, when we discuss a high speed LAN, the 1 Gbps environment must be considered. Now, let us consider whether DQLAN is applicable for 1 Gbps environment. In the simulation, the virtual distance is shortened to 100 meters. Table 2 shows the simulation results, time unit for average delay is the microsecond. The first column shows the maximum available channel capacity when the slot marker and CMS' overhead are excluded. The second column shows the load. In Table 2, we show that if frames' T_{ds} are less than $2*T_{pvd}$, then $2*T_{pvd} - T_{ds}$ of the bandwidth will be lost. The load plus the lost bandwidth is the virtual load shown in the third column. The maximum utilization that can be achieved is Available Capacity - (Virtual Load - Load). For example, when the load is 0.9 the available capacity is 0.962 and the virtual load is 1.026, the maximum utilization is 0.962 - (1.026 - 0.9) = 0.836. Under this circumstance, DQLAN still achieves 0.83 utilization. We know from queueing theory that when the arrival rate is equal to or greater than the capacity of the server, the queue length becomes unbounded and the average delay approaches infinity. This is why the average delay is so high when the load is 0.9.

There are three possible solutions to improve the utilization. First, shorten the cable length to keep the virtual load close to actual load. Second, modify the DQLAN protocol to make the TQ more intelligent. Third, use interleaved DQRAP when "a" is greater than 0.5. Methods 2 and 3 are described in [6].

Table 2 Gpbs DQLAN Performance

Available Capacity	Load	Virtual Load	Utilization	Average Delay
0.996	0.10	0.114	0.10	2.56
0.991	0.20	0.227	0.20	3.20
0.987	0.30	0.342	0.30	4.04
0.983	0.40	0.456	0.40	5.16
0.979	0.50	0.568	0.50	6.81
0.974	0.60	0.684	0.60	9.59
0.970	0.70	0.796	0.70	15.57
0.966	0.80	0.914	0.80	46.65
0.962	0.90	1.026	0.83	9285.12

5 Conclusions

DQLAN performance is superior to all other contention based protocols when offered traffic follows a Poisson arrival pattern. DQLAN even has the edge over deterministic MACs since it has the immediate access feature as well as providing full utilization at high offered loads. DQLAN supports a true priority scheme, one that causes all higher priority traffic to transmit before lower priority traffic [7]. If a fixed size slot is utilized then preemption is supported, i.e., a high priority transmission can interrupt a lower priority transmission. Synchronous operation is supported with fixed slots, i.e., in a 100 Mbps system two DS3 channels could be dynamically established in three slot times. The piece de resistance is the economics: the DQLAN switching fabric is passive, the protocol is implemented as a four-state machine based on the TQ and RQ in each of the stations. Data and timing information is copied from one bus, messages are written on the other bus without regard to what might be there, i.e., a simple "or" operation.

In the 1 Gbps system described in Section 4.3 a virtual distance of 100 meters was assumed. We expect that a 10 Gbps system at 10 meters is feasible.

We suggest that DQLAN be seriously considered for both conventional LAN operations and as the switching fabric supporting high performance parallel computing.

References

[1] W. Xu, "Distributed Queueing Random Access Protocol for a Broadcast Channel", Ph.D. Thesis, Illinois Institute of Technology, 1990.

[2] W. Xu and G. Campbell, "A Distributed Queueing Random Access Protocol for A Broadcast Channel", SIGCOMM, Computer Communication Review, pp. 270-278, Vol. 23, No. 4, 1993.

[3] W. T. O’Connell, "Can Shared Access Networks Adequately Support High-Performance Computing? Re-visiting MAC Protocols", Invited Paper, HiNet’95, Santa Barbara, CA April 1995.

[4] X. Zhang and G. Campbell, "Performance Analysis of Distributed Queueing Random Access Protocol - DQRAP", Unpublished Paper, Illinois Institute of Technology, 1993.

[5] G. Campbell, "Acquiring Ternary Feedback in Control Minislots", Unpublished Paper, Computer Science Dept., Illinois Institute of Technology, 1993.

[6] C. T. Wu, "DQRAP Based Local Area Networks", Ph.D. Thesis, Illinois Institute of Technology, 1994.

[7] H. J. Lin and G. Campbell, "PDQRAP - Prioritized Distributed Queueing Random Access Protocol", 19th Conference on Local Computer Networks, pp. 82-91, Oct 1994.