On a MAC protocol based on distributed cycles

Włodek Dobosiewicz* Paweł Gburzyński†

To appear in the Journal of High-Speed Networks

Abstract

This paper introduces a novel transmission scheduling scheme applicable to high-speed ring networks. The essence of the proposed scheme is a special marker, called the *terminator*, circulating in the ring. The maximum normalised throughput of our protocol does not degrade with the increasing propagation length of the ring and the stations endure small access delays under light load. Our protocol also offers a full-scale isochronous service without any need for fixed reservations, negotiations, or other preparatory stages.

Keywords: Ring networks, Medium-access protocols, Network performance, Capacity–1 protocols.

1 Introduction

This paper describes a new approach to medium access in ring networks. The resulting topology/protocol combination is best suited for a fast MAN (metropolitan-area network) or a gigabit CAN (campus-area network). On the other hand, the new protocol is only slightly better than FDDI [11, 12] for medium-speed CANs.

Following tradition, we will use the *normalised propagation delay* to measure time, distance, and packet length [17]. We also abstract from coding issues, which may otherwise obscure the discussion, as FDDI uses a 4B5B code. We use information bits, so that an FDDI symbol is made of 4 bits, not 5. The FDDI network together with its terminology will be used in this paper as a reference.

---

[*] Supported in part by NSERC Grant No. OGP9110. Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2H1. email: dobo@cs.ualberta.ca.

[†] Supported in part by NSERC Grant No. OGP9183. Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2H1. email: pawel@cs.ualberta.ca.

Let L denote the maximum propagation delay between a pair of stations and l_p denote the average packet length (also in bits). The ratio of L/l_p is commonly denoted by a. A large value of L (and thus a) can result from a very long network (in *metres*) or/and a very high transmission rate. There is a demand for protocols that are capable of using a fixed fraction of the channel bandwidth regardless of the value of a. We call such protocols **capacity–1 protocols**. A number of *capacity–1 protocols* have been proposed in the literature [9, 16, 19, 21, 7, 1].

The topology used by our network is the same as in FDDI, i.e., the **dual counter-rotating ring**. In normal circumstances, only one of the two rings is used for communication. Consequently, we will restrict our presentation to the operation of a single ring.

A typical protocol for a ring network consists of two sets of rules: the transmission rules and the cleaning (packet removal) rules. One way of implementing both parts of the ring protocol is token passing. Most token-based protocols, FDDI in particular, give the right to transmit to the token-holding station only. This way only one station can transmit a packet at a time. Moreover, the time spent by the token on passing between stations is wasted. As the relative contribution of this time to the total operation time of the network becomes larger for long networks and/or higher transmission rates, token-based protocols don't have the capacity–1 property.

2 The distributed cycle protocol

The distributed-cycle protocol, called DCP for short, presented in this section bears some resemblance to FDDI [11, 10, 2]. As in FDDI, there is a single special packet circulating in the network. Its role is best compared to the role of special markers or flags used in some bus protocols (e.g., in Fasnet [20] or Expressnet [23]) to initiate the so-called transmission rounds. As the most visible responsibility of the station being visited by the marker is to erase from the ring all the traffic arriving from upstream, the marker used in DCP is called the *terminator*.

The terminator is inserted into the network upon startup and is passed around indefinitely. A station S_i receiving the terminator **must** hold it for an amount of time equal to the value of THT_i, a constant associated with the station.[1] While holding the terminator, the station effectively removes from the ring all packets arriving from upstream.

Assume that the network operates in a slotted manner. Each slot carries two binary flags: the standard *full/empty* flag which tells whether the slot carries a payload,[2] and the *terminator* mark, which a station can use to pass the terminator to the next station down the ring. The ring is filled with slots upon network initialisation. Initially, all these slots are marked as empty (their *full* flags are cleared) and their *terminator* flags are all set to ones (the terminator flag equal one means that the terminator is **not passed** within the slot). The terminator is initially assigned to one station which will hold it for the prescribed *terminator holding interval* and then pass it downstream.

Each station monitors the terminator flags of all incoming slots to learn when it receives the terminator. The station does it by unconditionally setting the terminator bit in each relayed slot to one, simultaneously reading the previous value of the bit. If the previous value turns out to be zero, the station knows that it has acquired the terminator and temporarily removed it from the network.

The time that elapses between two consecutive arrivals of the terminator at the same station is constant and the same for all stations. We will denote this time by TTRT which stands for *Total Terminator Rotation Time.* The value of TTRT is given by the formula:

$$\text{TTRT} = L + \sum_{i=1}^{n} \text{THT}_i$$

L is the propagation length of the ring and it includes repeater delays caused by all stations.

TTRT is expressed discretely in slots and is therefore accurate. By counting the slots passing by, every station can know in which slot it is going to receive the terminator. Consequently, the explicit terminator flag carried by the slot header is superfluous. Nonetheless, the presence of this flag makes the protocol more reliable in the face of possible station failures and other abnormal conditions.

Each station is equipped with a counter called the *Terminator Rotation Timer* (TRT) which

---

[1] Priorities may be implemented by assigning different values of THT to different stations.

[2] Using the terminology of DQDB [9] we will call this payload a *segment.*

counts the number of slots passed since the moment the station last released the terminator. This counter is set to zero when the station starts relaying the slot with the terminator.

A station holding the terminator logically disconnects the ring. This is accomplished by unconditionally clearing (setting to zero) the full flags in all incoming slots. The slot in which the station has received the terminator is the first slot to be cleared; therefore, the terminator flag should precede the full flag in the slot header.[3]

Assume that the stations are assigned sequential numbers from 0 to $n-1$, reflecting the stations' order on the ring. Assume also (for simplicity) that THT is the same for all stations. Suppose that a station S_i has a segment addressed to station S_j. S_i is allowed to start transmitting the segment in the first empty incoming slot that satisfies the condition: $\text{TRT}_i \geq (d-1) \times \text{THT}$, where d is the number of hops separating the sender from the destination ($d = (j - i + n) \bmod n$). Technically, once the TRT counter has reached the value required by the transmission rule, the station unconditionally sets to one the full flag in every incoming slot, simultaneously examining the previous contents of this flag. If the previous contents were zero, the station knows that it has reserved the slot; then it fills the slot's payload area with the backlogged segment.

Note that while a station is holding the terminator, it will perceive all incoming slots as empty. Moreover, the value in its TRT counter is large enough for a transmission to the most distant destination. Therefore, such a station enjoys a temporary privilege in accessing the ring. By no means, however, the right to transmit is limited to this station only.

Figure 1 illustrates the transmission rule in terms of a space-time diagram for a network consisting of six stations. For simplicity, the terminator holding time is equal to one slot per station. The vertical axis represents time (which flows "downward") and the horizontal axis corresponds to the space covered by the ring medium. The thick vertical segments and slanted arrows mark the slots during which a station is holding the terminator. Note that a station transmitting is such a slot (along the arrow) is guaranteed that its segment will make a complete circle through the ring (its path is clear for at least one ring-length). On the other hand, the

---

[3] This way the protocol can operate without buffering any portion of an incoming slot.

safe transmission distance for a station that does not hold the terminator is proportional to the amount of time elapsing since the moment the station released the terminator. For example, the maximum transmission distance for station S_0 transmitting in slot Δ_1 (i.e., in the slot in which the terminator is released) is one hop. This distance is determined by the location of the thick vertical bar representing the moving disconnection point of the ring. Similarly, in slot Δ_3, station S_0 can transmit for three hops and in slot Δ_5, it can transmit for five hops.

The condition for success: $\textsc{trt} > (d-1) \times \textsc{tht}$ does not have to be computed dynamically each time its outcome is needed. The values on the right-hand side can be precomputed at the network initialisation stage and stored in lookup tables. If different stations use different values of THT, the multiplication should be replaced by a summation.

In addition to the cleaning responsibilities of a terminator-holding station, each station is expected to strip the segments that it transmitted. Assume that all stations use the same value of THT and station S_i transmits a segment at time t_s (measured at the beginning of the slot in which the segment has been transmitted). If the terminator was released by the station at a time later than $t_s - n \times \textsc{tht}$, the segment either will not return to the station at all (it will be stripped by a terminator-holding station preceding S_i) or it will return to S_i at the moment when the station is holding the terminator. Otherwise, the segment will return to the station and will have to be stripped explicitly. In the first case, the station has no further interest in the segment. In the second case, S_i remembers that L slots later, when the slot completes its circle through the network and arrives back at the station, its full flag will have to be cleared unconditionally. As a station may transmit a number of segments in sequence before the first element of this sequence arrives back at the station, it has to maintain a FIFO queue of slot numbers to strip. A station stripping its own segment from a slot can reuse this slot immediately to transmit another segment, if the transmission rule (the current position of the terminator) allows it to do so.

The transmission condition can be simplified by eliminating d and assuming that each packet is addressed to the worst-case destination—the station located immediately upstream of the transmitter, or even the transmitter itself. This will also simplify the stripping rules as each packet will be guaranteed to make a full circle through the ring. Note that this assumption must be made for a broadcast packet. Then, the condition for a successful transmission becomes: $\text{TRT} \geq \text{TTRT} - \text{THT} - \text{L}$.

It is possible to implement an unslotted version of the protocol with THT and TRT being actual timers, not just slot counters. The terminator is represented by a special packet, e.g., similar to the token in FDDI, which is passed around by the stations. A station receiving the terminator packet aborts its retransmission and holds it for a prescribed amount of time. While a station is holding the terminator, it disconnects the ring and strips all the traffic arriving from upstream. Packet stripping and frame reuse become now a bit trickier than in the slotted version, but they still are implementable, assuming a reasonable tolerance of clocks at the stations.[4]

DCP handles synchronous traffic in a flexible manner. If a low jitter is the primary criterion for synchronous traffic, it is a natural choice to transmit synchronous packets (segments) during the terminator visit only, giving them a priority over asynchronous packets. As the terminator visits every station at highly regular intervals, this approach offers the lowest jitter possible in a ring network.[5] A station holding the terminator would transmit asynchronous traffic only if there were no synchronous packets left; additionally, asynchronous packets would be transmitted without the terminator. Each station S_i has a guaranteed bandwidth for synchronous traffic that is equal to THT_i/TTRT. Unlike in FDDI and DQDB, bandwidth guaranteed for synchronous traffic, but unused, may be used freely for transmitting asynchronous packets.

A higher *maximum* bandwidth for synchronous traffic may be obtained by allowing the transmission of synchronous packets without the terminator—at the possible expense of a non-zero packet loss rate. It can be shown that if the sum of terminator holding times at all stations is a divisor of L, each station, besides receiving the terminator, gets a "window of opportunity" to transmit synchronous packets at very regular intervals.

---

[4] An attempt to reuse a stripped packet frame may be preempted by an upstream transmission. Therefore, stations have to recognise collisions—in a similar way as in Expressnet [23].

[5] This jitter is practically zero. Its only source is the variability in the repeater delays.

3 Performance

We used simulation to compare the performance of DCP, FDDI, METARING [5] (unslotted and slotted variants), and DQDB. The networks and their protocols were modeled in SMURPH [13, 14]. Each point in a performance curve was produced as an average of four independent experiments. The number of messages transmitted during a single experiment varied as a function of the network size, the offered load, and the token rotation time; its range was between 200000 and 2000000 messages.

Three lengths of a ring were considered: 10^5 bits, 10^6 bits, and 10^7 bits. Assuming a 200 km ring, the three propagation lengths represent transmission rates of $100 Mb/s$, $1 Gb/s$, and $10 Gb/s$. In the case of DQDB, the specified length is the length of the dual bus. The number of stations was the same for all networks and equal 32.

Two versions of DCP, slotted and unslotted, were implemented, each of the two versions occurring in two variants. In variant 1 (denoted by *w1*) a transmitting station did not know the number of hops separating it from the packet recipient. In variant 2 (denoted by *w2*) a transmitting station knew the location of the packet recipient and it could use the more subtle transmission condition. The unslotted variants of our protocol are compared with FDDI and the slotted variants are compared with DQDB; both variants are also compared with METARING.

The traffic was uniform with the probability of every pair (*sender*, *receiver*) being the same. In the unslotted version of DCP the packet format of FDDI was assumed, although the packet length was fixed—to facilitate frame reuse. The same packet format (with variable payload length) was used in the unslotted variant of METARING. Two message length distributions were considered. In one set of experiments, the message length was fixed at 8192 bits. This was also the (fixed) packet length (the payload) for DCP. Two values of the terminator holding time were investigated: 8424 bits (which allowed each station to transmit a single packet) and 33762 bits (four packets per station). In the proposed protocol, THT was the same for all stations. In FDDI, the above value of THT was translated into the total token holding time by multiplying it by 32 and adding to the propagation length of the ring—to produce the value of TTRT. In the second set of experiments, the message length was exponentially distributed with the mean of

4096. The packet length for DCP (the payload) was 1056 in that case. The terminator holding time (token holding time for FDDI) was 5128 bits per station.

In the slotted implementation of DCP, the slot format of DQDB was assumed. Thus, each slot consisted of a 384-bit payload and 40-bit header. The terminator holding time (this issue does not concern DQDB) was one slot per station. The message length was fixed and equal to the payload size.

In METARING a parameter k is used; its role can be compared to the role of THT in FDDI and in DCP. We assumed that the token holding time that allows each station in FDDI to transmit one packet per token cycle corresponds to $k = 1$ in METARING. This assumption may be considered unfair to METARING, since this protocol is capable of achieving much higher throughput when larger values of k are used, as partially illustrated in figure 4; the additional throughput is offset by an increased risk of starvation [8]. A modification to METARING, which aims at removing the starvation potential is presented in [3].

Figures 2, 3, 4 show the message access delay versus throughput[6] for the unslotted variants of our protocol and compare it with FDDI (denoted by f) and METARING (denoted by m). The throughput of METARING has been divided by two (the network uses two rings whereas both DCP and FDDI are single-ring networks). The token/terminator holding time was 8424 bits per station and the message length was fixed at 8192 bits.

Figure 4 shows the performance of FDDI for two token holding times. Although the maximum throughput gets larger with increasing TTRT, the message delays stay unchanged. METARING exhibits a similar behaviour: as the figure shows, its throughput can be pushed by increasing the value of k, but then the network becomes starvation-prone [5]. For short networks, METARING performs very well, even if k is small. For example, the maximum throughput of METARING ($k = 1$) for the 100Kb ring was 5.22 (the curve is not shown in Figure 2); it was equal to 7.35 for the 10Mb ring and $k = 500$.

---

[6] Our definition of message access delay covers the amount of time elapsing from the moment a message becomes ready for transmission (appears on top of the transmission queue) until the last bit of the message is successfully transmitted. Our definition of throughput excludes packet preambles, headers, and trailers, i.e., the throughput is *effective*.

The same pattern is observed for variable message length. Figure 5 shows the message access delay versus throughput for the 10Mb rings loaded with variable-length messages. The mean message length was 4096 bits and the token/terminator holding time per station was 5128 bits.

Figure 6 compares the performance of the slotted variants of DCP and METARING with DQDB (denoted by d). To make the comparison fair, the throughput of DQDB (and METARING) has been divided by half. Note that DQDB (in its bus-shaped-as-a-ring topology) uses twice as much fibre as our protocol and has two sets of transmitters/receivers per station. The maximum throughput achieved by the slotted variants of DCP exceeds (marginally) that of DQDB. The transmission condition in DCP is fulfilled more often for longer rings. In the 10Mb network, our protocol gives practically zero access time for light load which makes it indistinguishable from DQDB. For the same reason, the difference between the two variants of our protocol tends to disappear when the ring becomes long. Note that slotted METARING has serious problems for this network size. The transmission unit (slot) is much shorter than in the case of the unslotted variant and the maximum throughput achieved by METARING is low, even though k is very big.

Similarly to DQDB, the maximum throughput achieved by DCP doesn't deteriorate with the increasing propagation length of the channel, but, in contrast to DQDB, our protocol is **absolutely fair**. Although it is possible to improve the fairness of DQDB in a number of ways (e.g., see [15, 6, 22, 4, 18]), these methods are usually quite involved and based on a feedback mechanism, whose performance depends on the propagation length of the network.

Real-life applications seldom involve uniform traffic patterns. Thus, it is important to investigate the performance of MAC-level protocols for other patterns, such as **biased** or **bursty** traffic.

The behaviour of our protocol for these two traffic patterns was examined by simulation. The following network context was used in the reported experiments:

- The ring length was 10^5 bits.
- There were 32 stations attached to the network.
- All the stations generated *uniform* traffic at the same rate. Additionally, three selected

stations generated non-uniform traffic of a given type.

- The **slotted** version of the protocol was used.

The two traffic patterns were modelled in the following way:

Bursty traffic: stations 0, 12, and 26 generated traffic bursts totaling 5×10^4 bits (combined) every 10^5 bits of time. These bursts were made of short, single-packet messages whose receivers were chosen at random.

Biased traffic: stations 0, 12, and 26 continuously generated additional uniformly distributed traffic at a combined rate of 1 bit per 2 bits of time.

In both cases, the access delay/throughput results were exactly the same as in the case of uniform traffic. Thus, the interesting aspect of the protocol behaviour is the variance of mean access delays among the stations (depending on their distance from the non-uniform stations).

Figures 7 and 8 present the mean access delays (in slots) at individual stations as a function of uniform traffic load (the non-uniform traffic load being fixed at 50%). The numbers in the legend give the uniform load level, e.g., $w40\%$ represents DCP for uniform traffic generated at a rate of 0.4 bit per bit of time.

For comparison, the behaviour of FDDI for the same traffic is also shown. To make the results comparable, the packet size and form was made identical to the slot format used in WTP. In figures 7 and 8, the THT used for FDDI equals 100 slots (a TTRT of around 15 *ms* if the slot format of DQDB is adopted). The highest load of uniform traffic was reduced to 0.42, since this was the saturation point for the given parameters.

4 Summary

We presented a MAC–level protocol for a ring network. The protocol is based on passing a token-like special packet whose role is to provide a mobile cleaning agent—the so called terminator. With our approach, stations are allowed to transmit spontaneously and preempt (or be preempted by) other stations based on a dynamically changed structure of priorities imposed by the rotating terminator. Our protocol has a better performance than other ring protocols of similar complexity (FDDI, METARING), if the network is sufficiently long and/or fast.

A slotted variant of the proposed protocol has some features similar to DQDB. Similarly as DQDB, it is a *capacity–1* protocol, but, unlike DQDB, it is absolutely fair.

References

[1] C. Baransel, W. Dobosiewicz, and P. Gburzynski. CBRMA++/SR: On the design of a MAN/WAN MAC protocol for high–speed networks. *IEEE Journal on Selected Areas in Communications*, 11(8):1268–1277, Oct. 1993.

[2] W. Bux. Token-ring local-area networks and their performance. In *Proceedings IEEE*, volume 77, pages 238–256, Feb. 1989.

[3] J. Chen, I. Cidon, and Y. Ofek. A local fairness algorithm for gigabit LANs/MANs with spatial reuse. *IEEE Journal on Selected Areas in Communications*, 11(8):1183–1192, Oct. 1993.

[4] S. Cheung. Controlled request DQDB: Achieving fairness and maximum throughput in the DQDB network. In *Proceedings of IEEE INFOCOM'92*, pages 180–189, Florence, Italy, May 1992.

[5] I. Cidon and Y. Ofek. A full-duplex ring with fairness and spatial reuse. In *Proceedings of IEEE INFOCOM'90*, pages 969–981, 1990.

[6] M. Conti, E. Gregori, and L. Lenzini. A methodological approach to an extensive analysis of DQDB performance. *IEEE Journal on Selected Areas in Communications*, 9(1):76–87, Jan. 1991.

[7] W. Dobosiewicz and P. Gburzyński. A new topology for MANs: the pretzel ring. In *Proceedings of IEEE INFOCOM'92*, pages 2408–2414, Florence, Italy, May 1992.

[8] W. Dobosiewicz and P. Gburzyński. On token protocols for high–speed multiple–ring networks. In *Proceedings of the International Conference on Network Protocols*, pages 300–307, San Francisco, USA, Oct. 1993.

[9] Distributed Queue Dual Bus Subnetwork of a Metropolitan Area Network. IEEE Std. 802.6–1990, July 1991.

[10] D. Dykeman and W. Bux. Analysis and tuning of the FDDI media access protocol. *IEEE Journal on Selected Areas in Communications*, 6:997–1010, July 1988.

[11] Fiber Distributed Data Interface (FDDI) – Token Ring Media Access Control (MAC). American National Standard for Information Systems, Doc. No. X3, 139–1987, Nov. 1987.

[12] Fiber distributed data interface, System Level Description. Digital Equipment Corporation, June 1990.

[13] P. Gburzyński and P. Rudnicki. Object-oriented simulation is SMURPH: A case study of DQDB protocol. In *Proceedings of 1991 Western Multi Conference on Object-Oriented Simulation*, pages 12–21, Anaheim, California, Feb. 1991.

[14] P. Gburzyński and P. Rudnicki. *The SMURPH Protocol Modelling Environment.* University of Alberta, Department of Computing Science, Edmonton, Alberta, Canada T6G 2H1, 1991.

[15] E. Hahne, A. Choudhury, and N. Maxemchuk. Improving the fairness of DQDB networks. In *Proceedings of IEEE INFOCOM'90*, pages 175–184, San Francisco, CA, June 1990.

[16] P. Jacquet and P. Muhlender. Machnet: A Simple Access Protocol for High Speed or Long Haul Communications. In *SIGCOMM'90 Symposium*, pages 90–100, Sept. 1989.

[17] L. Kleinrock. *Queuing Systems.* John Wiley & Sons, 1975.

[18] L. Kumar and A. Bovopoulos. An access protection solution for heavy load unfairness in DQDB. In *Proceedings of IEEE INFOCOM'92*, pages 190–199, Florence, Italy, May 1992.

[19] J. Limb. A Simple Multiple Access Protocol for Metropolitan Area Networks. In *SIGCOMM'90 Symposium*, pages 69–78, Sept. 1990.

[20] J. Limb and C. Flores. Description of Fasnet, A unidirectional local area communications network. *Bell Systems Technical Journal*, Sept. 1982.

[21] H. Mueller, M. Nassehi, and J. Wong. DQMA and CRMA: New access schemes for Gbit/s LANs and MANs. In *Proceedings of IEEE INFOCOM'90*, San Francisco, CA, June 1990.

[22] B. Mukherjee and S. Banerjee. Alternative strategies for improving the fairness in and an analytical model of DQDB networks. In *Proceedings of IEEE INFOCOM'91*, pages 879–888, Miami, FL, Apr. 1991.

[23] F. Tobagi, F. Borgonovo, and L. Fratta. Express-net: A high-performance integrated-services local area network. *IEEE Journal on Selected Areas in Communication*, 1(5):898–913, Nov. 1983.

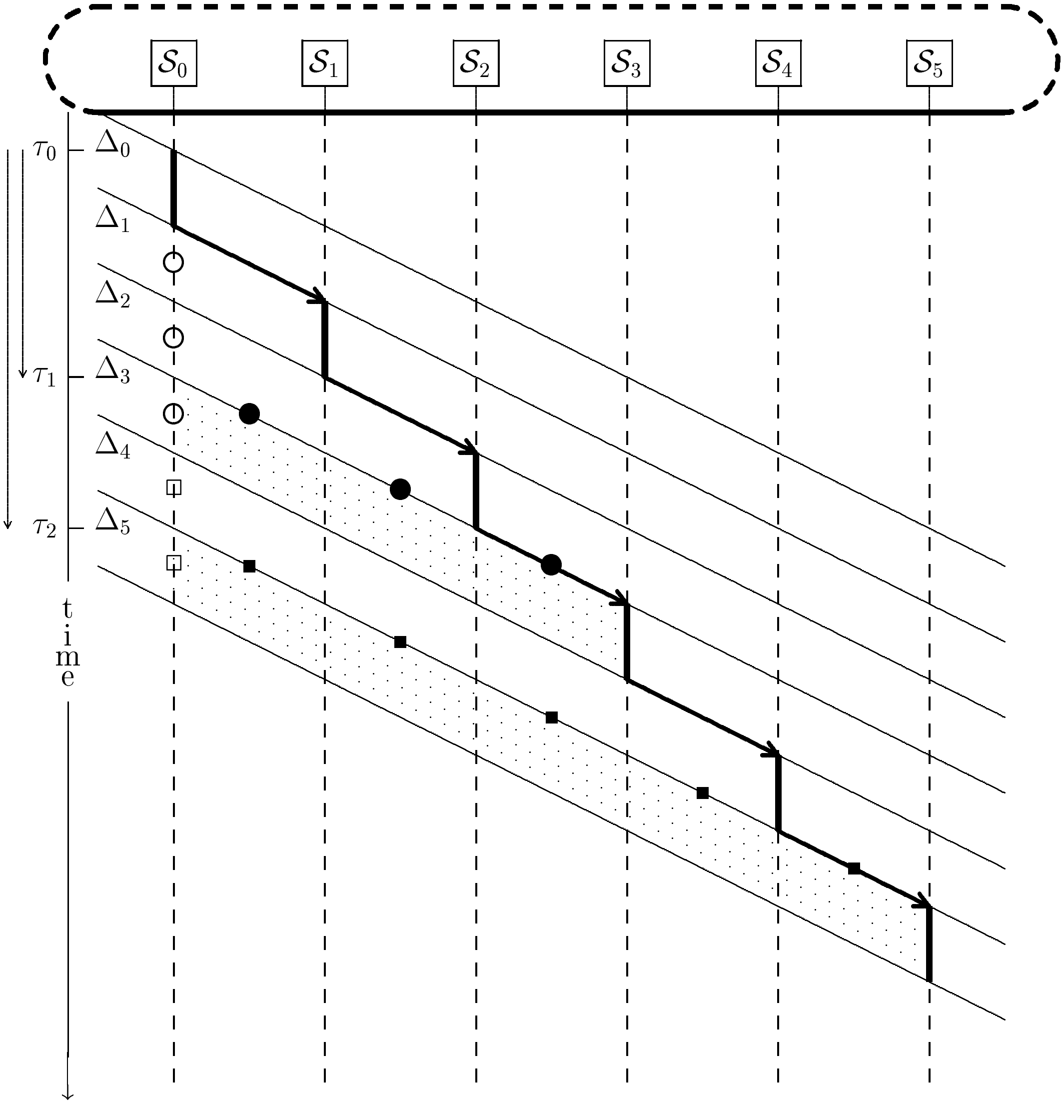

Figure 1: Space-time diagram of DCP for six stations.

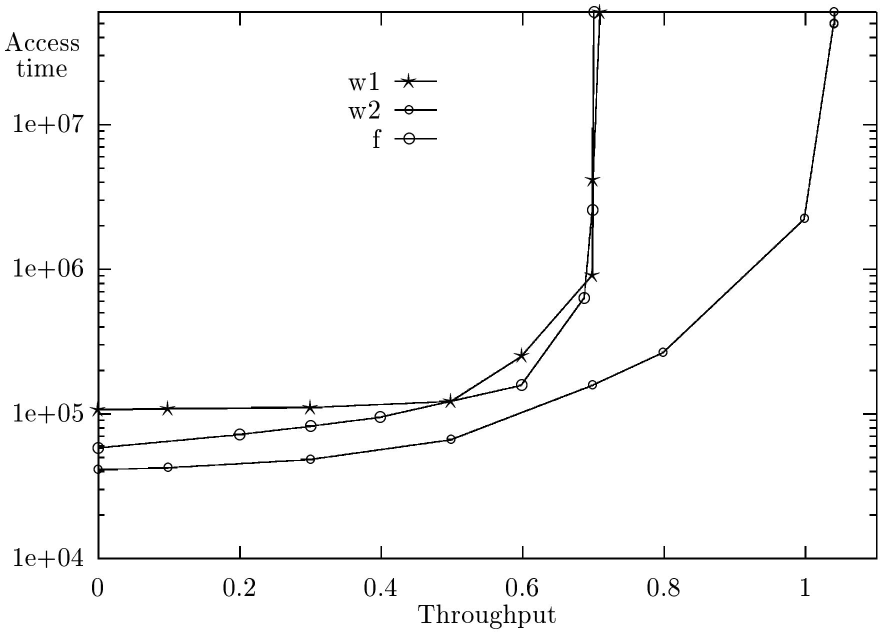

Figure 2: Ring length = 100Kb, THT = 8Kb, fixed message length = 8Kb, packet length = 8Kb.

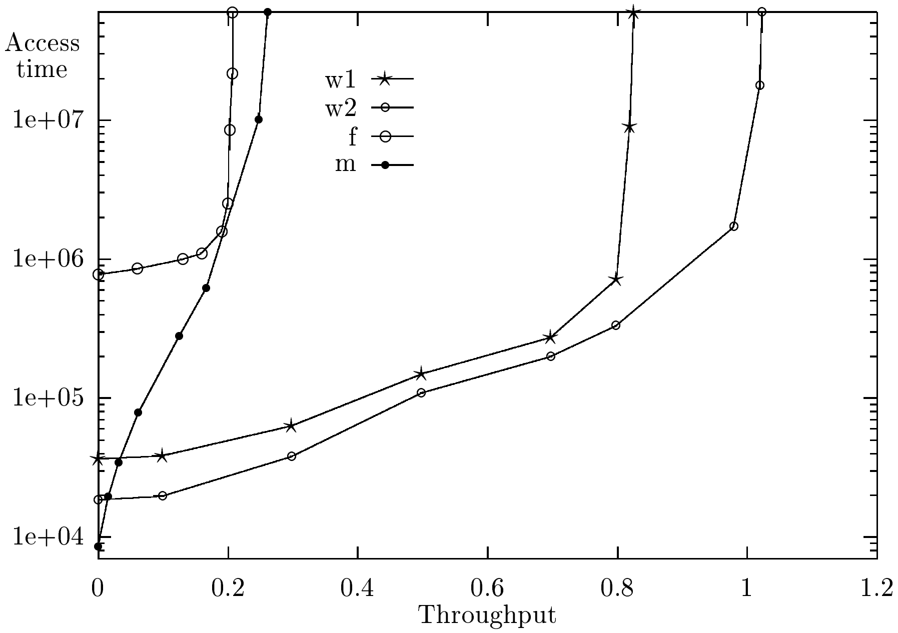

Figure 3: Ring length = 1Mb, THT = 8Kb, $k = 1$ (METARING), fixed message length = 8Kb, packet length = 8Kb.

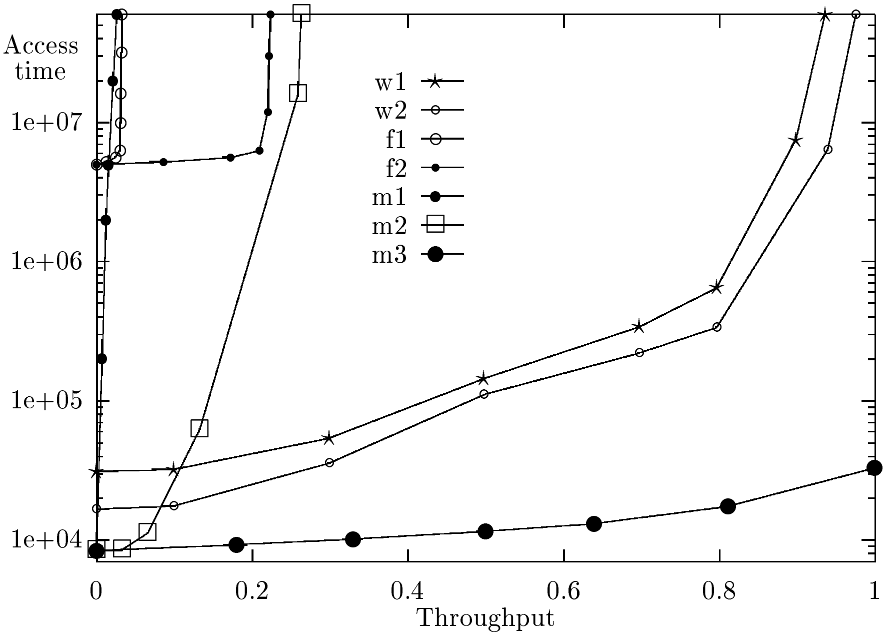

Figure 4: Ring length = 10Mb, THT = 8Kb (WTP), 8Kb (FDDI f1), 80Kb (FDDI f2), $k = 1$ (METARING m1), 10 (METARING m2), 500 (METARING m3), fixed message length = 8Kb, packet length = 8Kb.

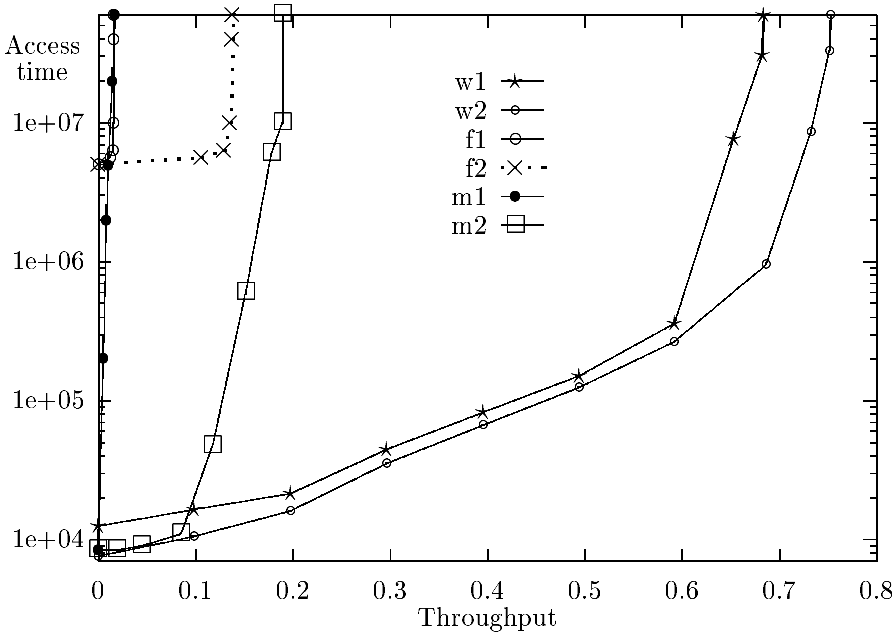

Figure 5: Ring length = 10Mb, THT = 4Kb (WTP), 4Kb (FDDI f1), 40Kb (FDDI f2), $k = 4$ (METARING m1), 40 (METARING m2), variable message length, mean = 4Kb, packet length (WTP) = 1Kb.

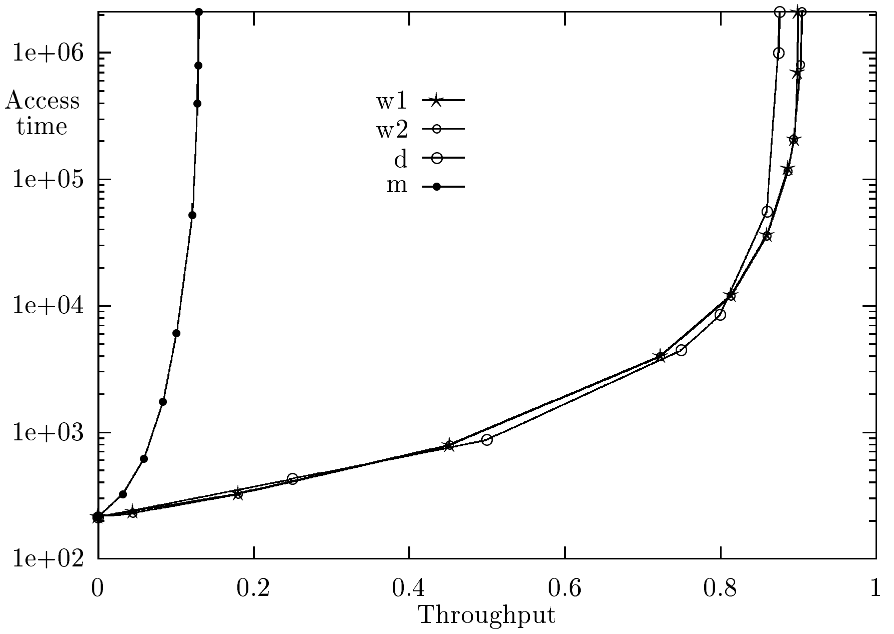

Figure 6: Ring length = 10Mb, THT = 1 slot, $k = 100$ (METARING).

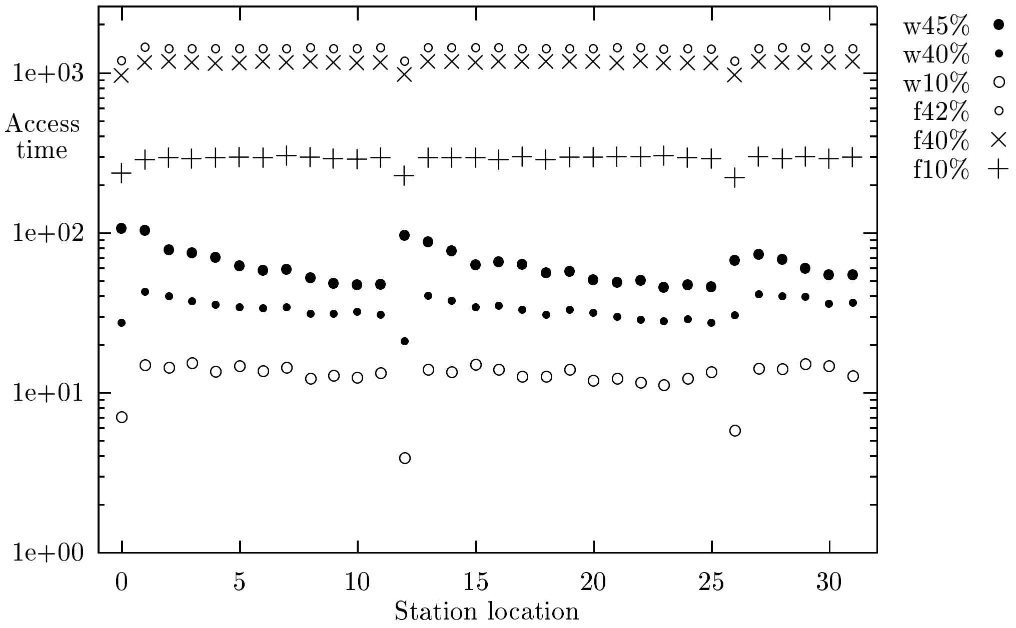

Figure 7: Ring length = 100K, THT = 1 slot, bursty traffic against uniform load.

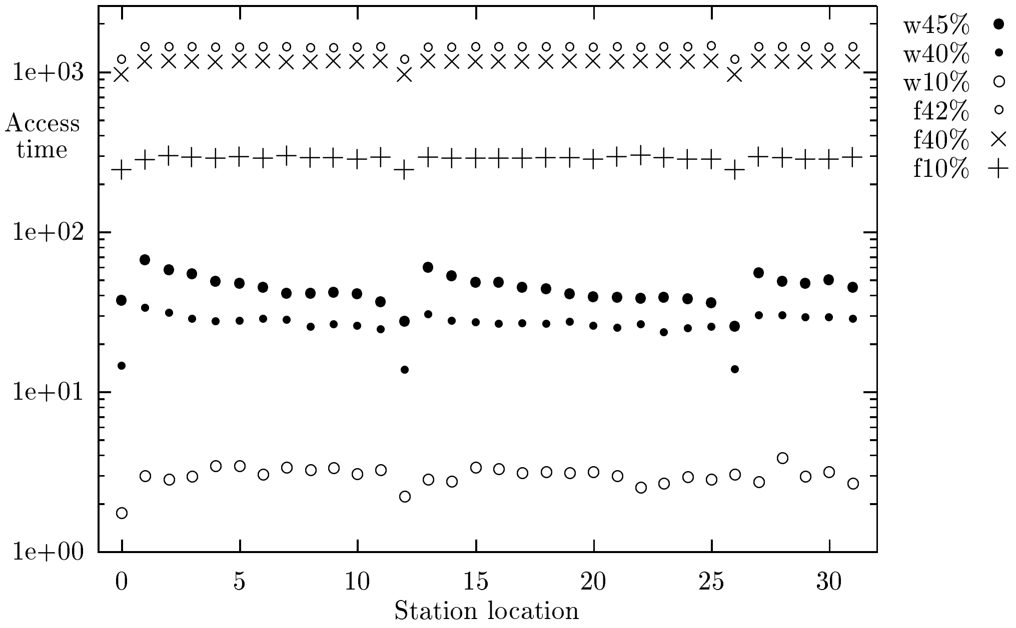

Figure 8: Ring length = 100K, THT = 1 slot, biased traffic against uniform load.