CRMA 고속 네트워크를 위한 슬롯 재사용 알고리즘

김성수 이성호[O] 양영규
한국전자통신연구원
{kss62937, sholee, ykyang}@etri.re.kr

Slot Reuse Algorithm for CRMA High Speed Networks

Sung-Soo Kim Seong-Ho Lee[O] Young-Kyu Yang
Computer & Software Technology Laboratory
ETRI(Electronics and Telecommunications Research Insititute)

요 약

Cyclic-Reservation Multiple-Access(CRMA) is an access scheme for high-speed local and metropolitan area networks based on folded-bus or dual-bus configurations. CRMA provides high throughput and fairness independent of the network speed or distance.

This paper describes a simulation-based quantitative analysis of the performance gains obtained by introducing slot reuse in CRMA. Generally, a longer cycle length means a longer access delay and a lower throughput. Therefore, it is desirable to develop a scheme such that the cycle length is the shortest. In this paper, we will study the problem of reducing the total number of empty slots generated within every cycle. However, it has been shown that the problem is NP-complete under the constraint that all empty slots used by a station in a cycle are required to be consecutive. We present the algorithm that improves previous novel approach by using previous node information.

We compare our slots reuse scheme with several slot reuse algorithms such as region scheme (FMR), address schemes, novel approach in terms of the following two important performance criteria: average cycle length and average slot utilization ratio. As compared with the one proposed in novel algorithm, the new scheme makes the cycle length much shorter. Besides, the resulting slot utilization and the access delay are better than those of the other two schemes.

1. INTRODUCTION

In the future, high-speed local and metropolitan area networks will provide capacities beyond one Gigabit per second and a geographical coverage of hundreds of kilometers. To support such networks, several high-speed and cost effective access schemes have been proposed[7][5][10][9].

The dual bus configuration is a popular configuration for LAN and MAN and several access schemes have been suggested for this topology, e.g., DQDB, Simple, Fastnet and CRMA. Cyclic-Reservation Multiple Access(CRMA), based on slotted unidirectional dual bus structure, is an access scheme for high speed local and metropolitan area networks[8][6][7][1].

The CRMA protocol coordinates the access to the bus by cycles and stations must reserve slots for a cycle in which they intend to transmit. In order to reserve slots for transmission on Bus *A*, station *M*, the head station for Bus *B*, generates special *Reserve* commands on Bus *B*. A *Reserve* command is generated on Bus *B* for each Bus *A* cycle. Every command contains a cycle number and a counter that collects reservations of the stations for this cycle. A station increments the counter by the number of slots it wants to reserve and station *1*, the head station for Bus *A*, receives the *Reserve* command form Bus *B* and generates a corresponding cycle on Bus *A* by issuing a *Start* command. A *Start* command is issued only for cycles for which reservations were made and every *Start* command is followed by as many free slots as were reserved for the corresponding cycle. Figure 1. shows this mechanism.

In the original CRMA reservation scheme, there exists an undesirable feature : once a slot is used, it propagates to the end of the bus even though the packet might already have been read by its destination station[7]. Consequently, the length of a cycle is always equal to the total number of empty slots required by all the stations in the cycle. On the other hand, the phenomena of traffic locality usually happens in the present networks, including metropolitan area networks. By exploiting the property of traffic locality, several methods based on the concepts of *slot reuse* have been proposed to improve the performance of the original CRMA reservation scheme. In these slot reuse methods, a packet that has passed its destination is taken out of the transmission line such that the bandwidth is released and can be further reused. In other words, downstream stations may have a chance to reuse those slots which have already been used by upstream stations and whose packets have already been received by their destinations. Obviously, in such a situation where traffic localities exist and slot reuse methods are adopted, it may be unnecessary for Headend *A* to generate so may empty slots in each cycle. Thus, a shortest cycle length is obtained. In general, a shorter cycle length means a better slot utilization as well as a shorter access delay. In this paper, we will study the slot reuse problem, or equivalently, the problem of reducing the total number of empty slots generated within every cycle.

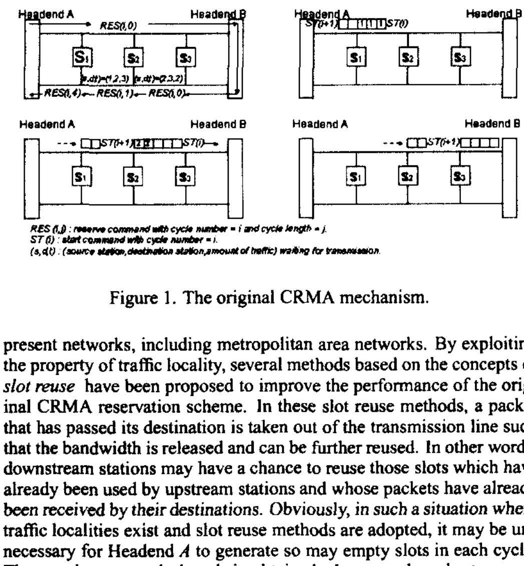

Figure 1. The original CRMA mechanism.

2. RELATED WORKS

The idea behind the region scheme is to divide the bus into regions and to associate every reservation to a pair consisting of the source region and the destination region. The above two regions will be referred to respectively as the source and destination regions of the reservation.

Huang *et al.* introduced slot reuse method, called the *cycle compression algorithm* [2][3]. They proved that the problem of finding a shortest cycle for a reservation pattern of a CRMA high-speed network is *NP-complete* . They also proposed an $O(n^2)$ approximation algorithm to find an approximate solution of CCP(cycle compression problem); where n is the number of stations in the network. FMR(Finding Multiple-Level Region) algorithm which they proposed is a one of the region schemes.

In address schemes, the exact destination address of every reservation is supplied at reservation stage, and every station releases slots that are destined to itself.

Sharon *et al.* investigated several simple schemes such as region schemes and address schemes for slot reuse in CRMA[10]. They showed that a better slot reuse is achieved when stations receive an equal throughput in every cycle compared to a proportional throughput. In case of local traffic, they introduced simple scheduling schemes that use explicit addressing information in order to compute a transmission scheduling.

The *First Fit* algorithm is based on the well known result that when all reservations are of equal length, the First Fit algorithm produced an optimal scheduling in polynomial time. In case of equal length reservations, the First Fit scheme is as follows: This scheme is an extension of First Fit to the case where the reservations are no necessarily of the same length. Reservations are scheduled by First Fit as if they were of equal length. However, for every slot in the table we also keep the maximal number of slots that is needed by any reservation that was scheduled into the slot. This number is called the *expand number* of the slot.

The second scheme schedules the reservations similarity to First Fit, i.e. in the scheduling process it considers all the reservations to be of equal length and it schedules the reservations in the same order as First Fit. However, a reservation is not scheduled into the first possible slot as in First Fit, but rather according to the following criterion: all the reservations of length between $2^i + 1$ to 2^{i+1} for $i \geq 1$ are scheduled into the same group of slots.

Last address scheme schedules reservation by the following method. When scheduling a reservation, the scheduler first looks for all the slots into which the reservation can be scheduled, as follows:

- expand number $\geq$ message size is scheduled into the slot with the *minimal expand number*
- expand number $<$ message size is scheduled into the slot with the *maximal expand number*
- otherwise is scheduled into a new slot (expand number = message size)

Marsan *et al.* investigated the slot reuse approach, and studied how it can improve the performances of the most promising protocols for metropolitan area networks such as CRMA(a IBM proposal) and DQDB(the IEEE 802.6 standard)[4]. They proposed a novel approach to slot reuse in CRMA which performed well and met the constraint of keeping contiguous the slots belonging to the same message.

3. EXTENDED NOVEL ALGORITHM

We now describe a extended novel algorithm to slot reuse in CRMA, based on the dual bus configuration where stations make reservations on one bus and transmit on the other bus. The propagation of information about reservations requires some waste in transmission bandwidth, which must be overcompensated by the gain obtained by slot reuse for the algorithm to be beneficial. A *cluster* is the information about available slots and destination for the slot reuse. Each cluster has the information for identifying the slot cluster available for possible reuse, such as, the number of slots in the cluster, the position of the first slot of the cluster in the transmission cycle and the index of the station up to which the slots in the cluster can be reused. However, Full information on the cycle profile is considered still too costly to be propagated, so that we propose to limit the information associated with the reserve command to one of the slot clusters that make the profile and to its position inside the cycle.

According to the Marsan's proposal(so called *novel proposal*)[4], each station receives from the downstream station a reserve command carrying the following information: the cycle number, the cycle length, the position of the first slot of the propagated cluster, the size of the cluster, the first downstream station accessing slots in the cluster, previous node information(destination, message size).

The difference between the *novel proposal* and our proposal is to have the information related to previous node information. Each station must decide which cluster information must be propagated to the upstream station with the reserve command, according to its transmission needs and to the information received from the downstream station.

Major steps of our proposed algorithm for the slot reuse given as follows:

1. determine the possibility of the slot reuse at each station.
2. create the cluster which is transmitted to next station.

The algorithm which determines the slot reuse possibility describes in **Algorithm 1**. The criterion of the *novel proposal* in order to decide which cluster information must be propagated to the upstream, is to maximize the area of the propagated cluster. However, the station which would propagate the cluster cannot know which solution is optimal, since the transmission requirements of upstream stations should be known for an optimal choice to be possible. So, the first criterion of our proposal is to check the previous message size and message length against the reusable slots and index of station in cluster. We select clusters which can be scheduled into slots for previous station's reserved command. And then, we choose the maximum-area cluster among these clusters. Figure 2. shows this situations. The algorithm to select the cluster is formally given in **Algorithm 2**.

Algorithm 1 Slot Reuse Determination Algorithm

if (msg_dest[i] $\leq$ clst[i] and msg_size[i] $\leq$ clst_size[i]) **then**
  slot reuse;
  set cycle_size[i-1] = cycle_size[i];
else
  no slot reuse;
  set cycle_size[i-1] = msg_size[i];
end if

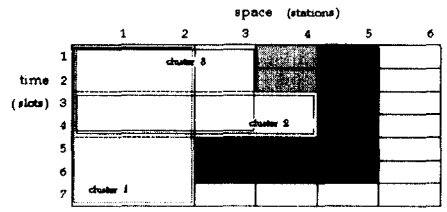

Figure 2. Time/Space diagram for the reservation of station 3[4].

Algorithm 2 cluster selection algorithm

Step 1. *initialization*
  set p_msg_size = msg_queue[i-1].msize;
  set p_msg_dest = msg_queue[i-1].mdest;.
Step 2. *check cluster which can be scheduled*
  if ($clst_i$.Rslot $\geq$ p_msg_size)
    and ($clst_i$.Tostation $\geq$ p_msg_dest)
  then add the $clst_i$ to cluster selection list.
Step 3. *select the maximum-area cluster in selection list.*
  set sel_clst = max($clst_0$,...,$clst_i$);
Step 4. *propagate the selected cluster.*

4. SIMULATION RESULTS

All simulation models, whose results are presented in this paper, were coded in C++ and run on PentimumIII-850 processor under the Windows 2000. There are three distribution types of parameters for traffic simulation such as message inter-arrival time, message length distribution, message destination distribution. We define a message inter-arrival time distribution as follows:

$$\frac{1}{\lambda_1} \ln \frac{1}{U(0,1)} \tag{1}$$

In each system size and traffic pattern we assume four different message length distributions. In first case we simply assume that the lengths are derived from the uniform distribution in range 1 to 100. In the second case we assume that the lengths are exponentially distributed with an average of 100 slots. In the third case, we assume that the lengths are geometric with standard deviation of 6 slots. We can describe the following distribution equation.

$$\lceil \log_{p_1} U(0,1) \rceil \tag{2}$$

In the last case, we assume that the lengths are a batch traffic: 80% of the reservations are one slot long and 20% are 16 slots long. We can describe the following batch distribution policy.

$$l = \begin{cases} m & \text{if } U(0,1) < \epsilon \\ n & \text{otherwise} \end{cases} \tag{3}$$

Destination addresses for burst force are chosen according to two options: uniform traffic and traffic with locality. According to the first option, every message is directed with the same probability to any station in the network except for the transmitter.

$$p = \begin{cases} \frac{1}{NumOfStations - 1} & \text{if } j < i \\ \text{discard it} & \text{otherwise} \end{cases} \tag{4}$$

According to the second option, messages are most likely directed to nearby stations; the probability of transmitting a message from station i to station j, with $i \neq j$, is

$$P_{ij}(p) = \frac{p(1-p)^{|i-j|-1}}{2-(1-p)^{n-1}-(1-p)^{i-1}} \tag{5}$$

where n is the total number of traffic sources and p is a parameter in the range (0,1) selecting the amount of locality in the traffic pattern. This equation represents a *normal geometric distribution*, i the source station, j the destination station and p determines the level of locality of the traffic. We selected p to be 0.1, i.e, a traffic with high degree of locality. We generate message samples at each station using above eight message patterns according to message length and destination distributions. We use this traffic sample to compare slot reuse performance among several algorithms.

It can be observed from Table 1 to 4 that in most cases, the first address scheme is the best while FMR is the worst. Slot utilization ratio is computed as following:

$$r = \frac{A_c}{A_{sr}} \tag{6}$$

, where A_{sr} denotes average used cycle length and A_c = total cycle length / the number of stations. Our extended novel algorithm has better slot reuse performance than original novel algorithm and FMR scheme.

Message Length	A1	A2	A3	Ours	Novel	FMR
uniform[1..100]	1.83	1.9	1.9	1.53	1.51	1.08
exp(0.01)	1.58	1.62	1.63	1.45	1.48	1.09
geom, s.d.=6	2.1	2.1	2.07	1.55	1.53	1.08
batch	2.08	2.2	2.2	1.55	1.53	1.09

Table 1. The performance comparison of the slot reuse algorithms for the case of uniform traffic (50 stations)

(A1, A2, A3 : address scheme 1, 2, 3, FMR : finding multi-level region, Novel : original novel algorithm, Ours : extended novel algorithm)

Message Length	A1	A2	A3	Ours	Novel	FMR
uniform[1..100]	1.89	2.06	2.05	1.52	1.52	1.04
exp(0.01)	1.63	1.8	1.79	1.5	1.5	1.05
geom, s.d.=6	2.17	2.23	2.17	1.54	1.53	1.05
batch	2.17	1.53	1.53	1.54	1.53	1.05

Table 2. The performance comparison of the slot reuse algorithms for the case of uniform traffic (100 stations)

Message Length	A1	A2	A3	Ours	Novel	FMR
uniform[1..100]	5.3	4.36	4.3	3.2	3.06	1.47
exp(0.01)	3.43	2.81	2.76	2.81	2.68	1.6
geom, s.d.=6	7.05	5.88	5.84	3.44	3.2	1.48
batch	7.1	6.29	6.29	3.24	3.24	1.47

Table 3. The performance comparison of the slot reuse algorithms for the case of geometric traffic (50 stations)

Message Length	A1	A2	A3	Ours	Novel	FMR
uniform[1..100]	6.41	5.65	5.57	3.16	3.14	1.19
exp(0.01)	3.96	3.64	3.55	2.93	2.93	1.21
geom, s.d.=6	8.41	7.33	6.81	3.29	3.21	1.2
batch	8.64	8.09	8.09	3.28	3.21	1.19

Table 4. The performance comparison of the slot reuse algorithms for the case of geometric traffic (100 stations)

5. CONCLUSION

In this paper we showed a comparison results among CRMA slot reuse algorithms such as address schemes, FMR algorithm, novel algorithm. The implementation of our extended novel algorithm improves better slot utilization than original novel algorithm by using previous station information. We have presented a method of improving the slot utilization rather than original novel algorithm by simply using previous station information. In the experimental results, we compare several slot reuse algorithm. As a result, first address scheme, one of address schemes, is best for achieving a high slot reuse ratio. Also, our proposed extended novel algorithm have better slot utilization than original novel algorithm and FMR scheme.

REFERENCES

[1] M. W. Garrett and S. Q. Li. A study of slot reuse in dual bus multiple access networks. In *Proc. INOFOCOM'90*, 1990.

[2] N. F. Huang, H. I. Liu, and G. K. Ma. A cycle compression algorithm for crma high-speed networks. In *IEEE ICC '93*, pages 1363–1367, 1993.

[3] N. F. Huang, H. I. Liu, and G. K. Ma. A new reservation scheme for crma high-speed networks. In *4th IFIF Conf. High Performance Networking*, pages A2–1–A2–15, 1993.

[4] M. A. Marsan, C. Casetti, S. M. Grasso, and F. Neri. Slot reuse in mac protocols for man's. In *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 11, NO. 8*, pages 1290–1301, Oct. 1993.

[5] M. A. Marsan and F. Neri. A comparative study of mac protocols for integrated mans. In *Int. J. Comput. Simulat.*, pages 61–95, 1993.

[6] M. M. Nassehi. Crma: an access scheme for high-speed lans and mans. In *Proceedings of SUPERCOMM/ICC*, pages 1698–1702, April 1990.

[7] M. M. Nassehi. Cyclic-reservation multiple-access scheme configuration. In *Proceedings EFOC/LAN*, pages 246–251, June 1990.

[8] M. A. Rodrigues. Erasur node: Performance improvements for the ieee 802.6 man. In *Proc. INFOCOM'90*, 1990.

[9] O. Sharon and A. Segall. A simple scheme for slot reuse without latency in dual bus. In *IEEE/ACM Trans. Networking*, volume 1, Feb. 1993.

[10] O. Sharon and A. Segall. Schemes for slot reuse in crma. In *IEEE/ACM Transactions On Networking, VOL. 2, NO. 3*, pages 269–278, JUNE. 1994.