Improving the Efficiency of CSMA using Reservations by Interruptions

Chuan Heng Foh and Moshe Zukerman
EEE Department
The University of Melbourne
Parkville, Vic. 3052, Australia
Email: {chuanhf,m.zukerman}@ee.mu.oz.au

***Abstract*-This paper proposes a new carrier sense multiple access (CSMA) protocol, called CSMA with Reservations by Interruptions (CSMA/RI). This new protocol uses a novel approach to reserve capacity for multiple users. The performance of the new protocol is studied by simulations under realistic (long range dependent) traffic conditions and compared with CSMA/CD as well as with its work conserving G/D/1 queue. It is demonstrated that CSMA/RI always offers better performance than CSMA/CD, and under certain realistic assumptions regarding packet size, the performance of CSMA/RI can be very close to that of its G/D/1 benchmark.**

I. INTRODUCTION

The challenge of developing an efficient yet reliable medium access protocol (MAC) has been an important research topic for over 30 years. Due to spatial distribution of the stations, and the bursty nature of the traffic, it has been considered impossible to achieve a reliable, simple, work conserving and a perfect scheduler MAC protocol.

Recently, we have experienced significant growth in research, developments and deployments of multi-media multi-service access networks. With the advent of technology came the need for new MAC protocols to meet Quality of Service (QoS) demands for different services.

In many of these new developments, old ideas are being recycled. For example: ideas based on the Aloha protocol [1] and tree algorithms [2] are now reused in MAC proposals for modern multi service wireless and wireline networks. Many of these new proposals employ a central controller which receives reservations from the member stations that wish to transmit and allocate bandwidth dynamically. Stations contending for access to the network must compete with each other to inform the controller of their need for a share of the available bandwidth. This leads to an access protocol which is based on two phases: (1) a collision based access protocol for the signals sent by the stations to the central controller, and (2) a collision free transmission which follow bandwidth allocation by the central controller to the stations in accordance with a certain scheduling algorithm.

As the access part of the network is the most costly, and since old ideas related to collision based MAC protocols are still and will be for the foreseeable future in extensive use, it is still important to review those old fundamental concepts and ideas and make an effort to improve on them perhaps by **breaking the most sacred rules.** In this paper we are making a significant improvement for carrier sense multiple access with collision detection (CSMA/CD) protocol by allowing stations to interrupt a successful transmission by another station, and this way to make a reservation. We call our protocol CSMA with Reservations by Interruptions, or CSMA/RI in short.

The CSMA/CD protocol is a collision-based protocol – an extension of the Aloha protocol. While in Aloha, stations transmit and, if they collide, they transmit later randomly, under CSMA/CD, stations listen to the channel and try to transmit when the channel is idle. We propose a further improvement by allowing a station, which has data to transmit, to interrupt a successful transmission, just for a short while, to notify all other stations that it would like to reserve capacity for transmission. The small interrupted portion of the ongoing transmission is immediately retransmitted by the source as it can also detect the interruption. The overhead is thus minimized and a fair order of transmissions can be maintained. If several stations interrupt a transmission (make reservations) at the same time, collisions may occur since any of these stations is not aware of others making a reservation. In this case, as in CSMA/CD, the stations may use any known conflict resolution algorithm to resolve their collisions. In this paper, we use the *Truncated Binary Exponential Backoff* (BEB) method used in the Ethernet protocol [3]. This method is chosen because of its wide acceptance in the "real" world. However, this choice is not obligatory, and as mentioned above, any other collision resolution algorithm could be used.

The remainder of the paper is organized as follows. In Section II, we describe the mechanism of CSMA/RI protocol. In Section III, we provide insight into the performance of CSMA/RI supported by extensive quantitative simulation results including a wide range of traffic models. The conclusion is finally drawn in Section IV.

II. THE CSMA/RI PROTOCOL

Consider a slotted system whereby time is divided into fixed length intervals. Each of these time intervals will be called a *slot*. The purpose of the slotted system assumption is to simplify the explanation and simulation of this new protocol. Nevertheless, CSMA/RI can be implemented as a non-slotted system.

Let T be the duration of a slot. All stations in the network are synchronized so that packet transmission is always commenced at the beginning of a slot. Henceforth, we shall use the concept of slot to refer also to the amount of data [bits] which can be transmitted within a time slot.

Let τ denote the maximum propagation delay between any two stations in the network. As in [4], in order to use common mechanisms for detecting collisions and aborting collided transmissions in a slotted channel, the minimum duration of a slot is $T=2\tau$. In addition, the channel is sensed idle by all stations τ units of time after the end of successful transmissions, and a successful transmission is detected τ units of time after it is started.

We introduce the concept of a *ready station* to refer to a station that has a packet or more waiting for transmission. Our proposed CSMA/RI protocol is an extension of the 1-persistent CSMA/CD protocol version described by the following rules:

(R1) If the channel is sensed idle, a ready station transmits its packet immediately. It is required to monitor the channel status in case of a collision.

(R2) If the channel is sensed busy, a ready station keeps monitoring the channel status. As soon as the channel becomes idle, the ready station transmits into the next slot with probability one.

(R3) Upon a detection of a successful transmission, each station reads the data from the ongoing packet transmission into its local buffer. Only the station to which the packet is addressed to may use the data, others should discard the data.

(R4) If a collision is detected, each ready station reschedules the retransmission individually to some later time based on a certain collision resolution algorithm. As mentioned in the introduction, in this paper we choose the BEB algorithm.

CSMA/RI enhances CSMA/CD by adding reservations by interruptions. The reservation is performed during an ongoing packet transmission. It is done by interrupting the ongoing packet transmission with a short pseudo-noise broadcast to all stations. The ongoing packet transmission is then resumed right after the interruption recovering the interrupted slot. Upon the completion of that ongoing packet transmission, only the stations that have performed reservation (by interruption), henceforth called *RI stations*, are allowed to access the channel. The rules respectively corresponding to the above CSMA/CD rules are described as follows:

(R1*) is the same as (R1).

(R2*) If the channel is sensed busy, in the case where a successful transmission is detected earlier, i.e. the channel is carrying a packet, (R2a*) applies to a ready station, otherwise the channel is busy due to a collision, and then (R2b*) applies.

(R2a*) A ready station may interrupt the packet transmission to make a reservation and to become an RI station only if the reservation has not been performed earlier in that packet transmission. Otherwise, it becomes a backlogged station. Upon the completion of the successful packet transmission, only an RI station is allowed to transmit into the next slot. A backlogged station remains silent and continues to monitor the channel.

(R2b*) In this case, if a ready station becomes ready during a collision, it transmits into the next slot as soon as the channel becomes idle as in (R2).

(R3*) Upon detection of a successful transmission, in addition to following Rule (R3), each station, either an RI which failed to obtain a channel or a backlogged station, waits for a randomly chosen waiting time that is not longer than the packet transmission time. During the waiting time, the station is required to monitor the channel to detect if other stations make reservations. If such reservation is made by other stations, the station aborts its reservation attempt and becomes a backlogged station. On the other hand, if no one else have made a reservation during the station's waiting time, then by the time its waiting time expires, the station performs the reservation by interruption.

(R4*) is the same as (R4).

(R5*) For a backlogged station, following the completion of a successful transmission, if the channel remains idle for at least a slot, it becomes a ready station and transmits its packet into the idle channel immediately as in (R1*).

We choose the 1-persistent version because for CSMA/RI it is more efficient than the p-persistent or the non-persistent version. Due to the reservations made, the likelihood of collision is significantly reduced and therefore the more aggressive 1-persistent version will be more efficient. As discussed earlier, several stations may interrupt a successful transmission at the same time slot. Nevertheless, the likelihood of such *interruption collision* is small, thus the resulting *packet collisions* are infrequent.

By (R3*), the reservation operation is carried out right after detecting a successful packet transmission. Then, each station waits for a random number of slots to initiate the reservation procedure. This random number has a discrete uniform distribution based on the packet size. For example, if the packet size is equal to 11 slots, the waiting time in number of slots will be equal to any of the following: 2, 3, ..., 11, each with probability 1/10. Recall that the first slot in a packet transmission cannot be used due to the detection time requirement. If two or more stations pick the same waiting time, this leads to an interruption collision which in turn leads to a packet collision. In this case, the collision will be resolved as in CSMA/CD (e.g. by BEB algorithm).

The reservation procedure is performed by interrupting the ongoing transmission with pseudo-noise for a duration of τ. This pseudo-noise is broadcast to all stations in the network. Upon detecting the pseudo-noise, the sender ceases the packet transmission, all other stations abort their reservation procedures and mark themselves as backlogged stations. Because the pseudo-noise only lasts for τ, it will vanish within the same slot which ends the reservation procedure. The sender then continues the packet transmission of the same packet from the point where it was interrupted by the

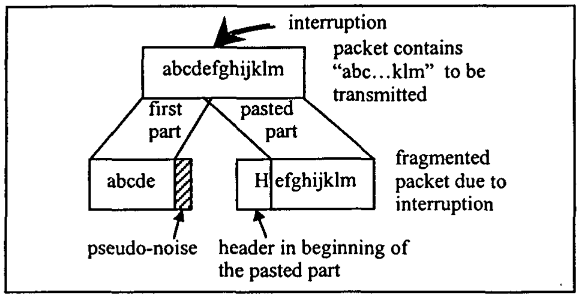

Fig. 1. The fragmented packet presented to the receiver

pseudo-noise. Since we assume that the channel is slotted, the fragmented packets can be recovered if the receiver is equipped with memory to buffer the receiving packet slot by slot. In practice, the recovery can be achieved by adding necessary information in the beginning of the pasted part of the packet and some data repeated, so that the receiver can paste the fragmented packet back together. (See Fig.1.)

After the completion of the packet transmission, only the RI stations can participate in the next packet transmission. Backlogged stations remain silent until the next successful transmission is detected. This way, CSMA/RI divides the ready stations into two groups during packet transmissions, the RI stations and the backlogged stations. This division reduces the number of stations participating in the contention.

Notice that the duration of a packet transmission is an important factor which affects the performance of CSMA/RI. Let n be the packet size in slots. Since reservations are based on randomly choosing a number among the numbers: 2, 3, ..., n, each with probability $1/(n-1)$, the larger the n, the lower the interruption collision probability, and hence the better the performance. This will be demonstrated in Section III.

In a noisy channel environment, it is not critical to distinguish between pseudo-noise and real noise. If noise occurred during an ongoing packet transmission, in CSMA/CD, the sender will abort the transmission and schedule the retransmission of the entire packet which is inefficient. By comparison, in CSMA/RI, the sender will correct the packet and save the retransmission overhead [5]. This is an additional benefit of CSMA/RI. Since the noise is treated as pseudo-noise that represents a reservation, RI stations may not exist in this case. However, the absence of RI stations will not lead to deadlock because an idle period leads to normal CSMA/CD collision resolutions as described in (R5*).

The packet size may be variable. In this case, the packet size should be specified in the header of the packet so that ready stations know how to choose their random waiting time for the interruptions.

It is possible that a station makes a reservation to become an RI station and shuts down due to an internal failure. In this case, (R5*) will apply to all backlogged stations after the channel is sensed idle for a minimum of one slot, so that they are allowed to transmit into an idle channel.

III. PERFORMANCE

In this section, we evaluate the performance of CSMA/RI by simulations. For all simulations in this section, a noise-free channel is assumed. We also assume that it is possible to achieve perfect time synchronization among the stations so that the overhead required in the pasted part of an interrupted packet depicted in Fig. 1 is excluded.

Furthermore, we assume that all stations are arranged in a star topology so that the propagation delay between any two stations is always τ. The duration of a slot is assumed to be 2τ.

A. Why CSMA/RI Performs Better than CSMA/CD ?

In Fig. 2, we demonstrate the improvement of CSMA/RI over CSMA/CD. The results in Fig. 2 are based on the assumptions that the number of stations is infinite, the arrival process is a Poisson process, and each arrival carries a packet of constant size, b. All these assumptions are consistent with those of [4] except that we assume that both CSMA/CD and CSMA/RI implement BEB instead of an adaptive retransmission algorithm used in [4].

Two cases of packet sizes: (1) 5 slots, and (2) 50 slots of the delay performance of CSMA/CD and CSMA/RI are presented in Fig. 2. The normalized mean packet delay is defined as the ratio between the mean packet delay and the packet transmission time. As demonstrated in Fig. 2, CSMA/RI achieves higher throughput than CSMA/CD in both cases.

In Fig. 3, we focus on the mean *Channel Assignment Delay* [4] (MCAD) of CSMA/CD and CSMA/RI. It is the number of wasted slots due to packet collisions before a successful packet transmission is obtained. As demonstrated in Fig. 3, CSMA/RI has significantly lower MCAD value than CSMA/CD. In particular, for the case of 50 slot packets, CSMA/RI has negligible MCAD for throughput level of up to 80%. **This lower MCAD value is the main reason that CSMA/RI performs better than CSMA/CD.**

B. The Disaster Scenario

In this traffic model [6], we assume a power up situation whereby M stations transmit simultaneously during a slot at a particular instant. A burst of size M therefore collides and has to be resolved. With no additional arrivals, we evaluate the delay performance for a constant packet size of 50 and 10 slots in CSMA/CD, CSMA/RI and G/D/1. The work conserving G/D/1 is used as a benchmark to represent the best delay performance possible to be achieved by a MAC protocol. In this case, it is important to know how close CSMA/CD and CSMA/RI are to this benchmark.

We demonstrate in Fig. 4 that for $b=50$, the normalized mean delay in CSMA/CD increases at a much higher rate than that of CSMA/RI as the burst size M increases, while

CSMA/RI performs very close to the work conserving G/D/1 benchmark.

The case of $b=10$ is also plotted in Fig. 4. We see that both CSMA/CD and CSMA/RI have higher normalized mean delay due to the shorter packet size, however CSMA/RI still performs better than CSMA/CD.

C. *A Stability Study of CSMA/RI*

In line with the channel stability study of Aloha protocols in [7], we present the system throughput of CSMA/CD and CSMA/RI as a function of the number of backlogged stations in Fig. 5. A Poisson arrival process and a constant packet size of 50 slots are assumed here.

Let M be the total number of stations. A station is either idle or backlogged. If a station is idle, it generates a new packet with probability σ within a timeslot. If the packet is successfully transmitted, the station returns to the idle state, otherwise it stays backlogged until the packet is transmitted. Let the number of backlogged stations be n, and S be defined as $S = (M-n)\sigma$, the curve of S as a function of n is known as the *Channel Load Line* [7]. Fig. 5 also includes the two Load Lines S_1 and S_2 related to the case $M=100$.

The Load Line S_1 represents a busy environment where stations are saturated. In this case, $\sigma=1$. Whereas in the case of the Load Line S_2, all stations equally share the total bandwidth. In this case, $\sigma = 1/M$ or 0.01.

We first observe that both Load Lines S_1 and S_2 intersect with the throughput curve of CSMA/RI at around 0.9, which is its maximum throughput. This high throughput level together with the long distance between the intersections signifies efficiency and stability. This result suggests that if the network consists of 100 stations, CSMA/RI can achieve high throughput even under very heavy load conditions (e.g. S_1). The wide operation range of CSMA/RI between the intersections means that it can achieve its maximum throughput for a wide range of traffic conditions. On the other hand, CSMA/CD has a narrow peak. In both S_1 and S_2 load conditions, CSMA/CD operates at relatively low throughput, especially for S_1.

D. *The LRD Traffic Model*

It was well verified in [8] that Poisson traffic assumption is inadequate to model Ethernet traffic. [9] shows that the M/Pareto can be used as a realistic traffic model for packet data traffic. M/Pareto traffic is a process composed of a number of overlapping bursts. Bursts arrive according to a Poisson process and have a Pareto distributed duration. It is characterized by four parameters which allow fitting of mean, variance, Hurst parameter, H, and level of aggregation. For simplicity, we fix the mean of the Pareto distributed burst size to one, so that the process can be characterized by its mean, variance and Hurst parameter.

We first assume infinite number of stations and constant packet size. Define the *burstiness* of LRD traffic process as the ratio of the standard deviation to the mean. In Fig. 6(a) to Fig. 6(c), we provide the delay performance for the case $b=50$, $H=0.9$ for different burstiness levels. The choice of $H=0.9$ is based on the study of [8]. To obtain a fair comparison, CSMA/CD and CSMA/RI were fed by the same LRD traffic generated by the M/Pareto model. We also compare its performance with its work conserving G/D/1 benchmark. As demonstrated in Fig. 6(a) to Fig. 6(c), CSMA/RI performs very close to this benchmark.

Notice that these almost-perfect performance results have been achieved for the case of 50 slot packets. We know that the performance of CSMA/RI degrades if the packet size is reduced. This is demonstrated in Fig. 6(d), where the delay performance of CSMA/RI backs off from its G/D/1 benchmark, and approaches that of CSMA/CD when $b=10$.

IV. CONCLUSION

We have presented a new version of CSMA called CSMA/RI. Under CSMA/RI, ready stations interrupt ongoing packet transmissions to reserve capacity. We have demonstrated by simulations, which were based on realistic traffic models and scenarios, that CSMA/RI performs significantly better than its CSMA/CD counterpart. For realistically long packets, it performs close to its G/D/1 benchmark.

V. REFERENCES

[1] N. Abramson, "The Aloha system - Another alternative for computer communications," *Proc. Fall Joint Comput. Conf.* AFIPS Conf., vol.37, pp. 281-285, 1970.

[2] R. Rom, M. Sidi, "Multiple Access Protocols," *Springer-Verlag*, New York 1990.

[3] R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," *Communications of the ACM*, vol. 19, no. 7, July 1976.

[4] S. S. Lam, "A Carrier Sense Multiple Access Protocol for Local Networks," *Computer Networks*, vol. 4, pp. 21-32, 1980.

[5] N. Maxemchuk, private communication.

[6] M. Ivanovich and M. Zukerman, "Worst Case Signalling Traffic for a Multi-service Signalling Protocol," *Proc. of IEEE ICC '98*, pp. 1431-1435, June 1998.

[7] S. S. Lam, L. Kleinrock, "Packet Switching in a Multaccess Broadcast Channel: Dynamic Control Procedures," *IEEE Trans. Commun.*, vol. COM-23, pp. 891-904, Sept. 1975.

[8] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, no. 1, pp. 1-15, Feb. 1994.

[9] R. G. Addie, T. D. Neame and M. Zukerman, "Modelling Superposition of Many Sources Generating Self Similar Traffic," *Proc. of ICC '99*, June 1999.

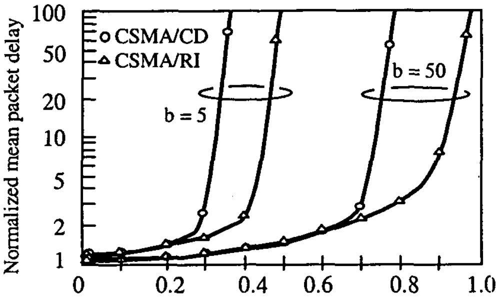

Fig. 2. CSMA/CD and CSMA/RI: delay versus throughput

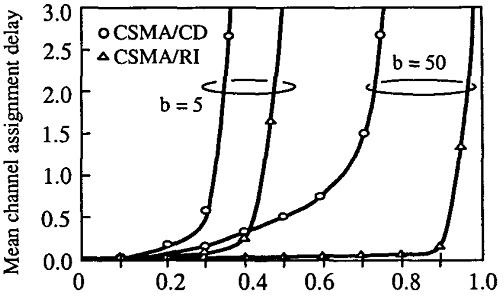

Fig. 3. CSMA/CD and CSMA/RI: MCAD versus throughput

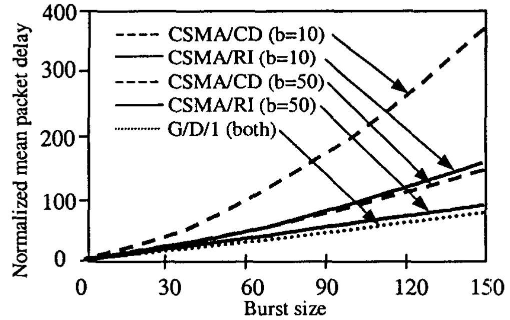

Fig. 4. Delay versus burst size for CSMA/CD, CSMA/RI and G/D/1

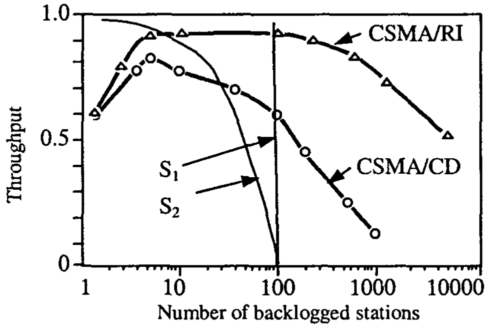

Fig. 5. Throughput versus number of backlogged stations for CSMA/CD and CSMA/RI

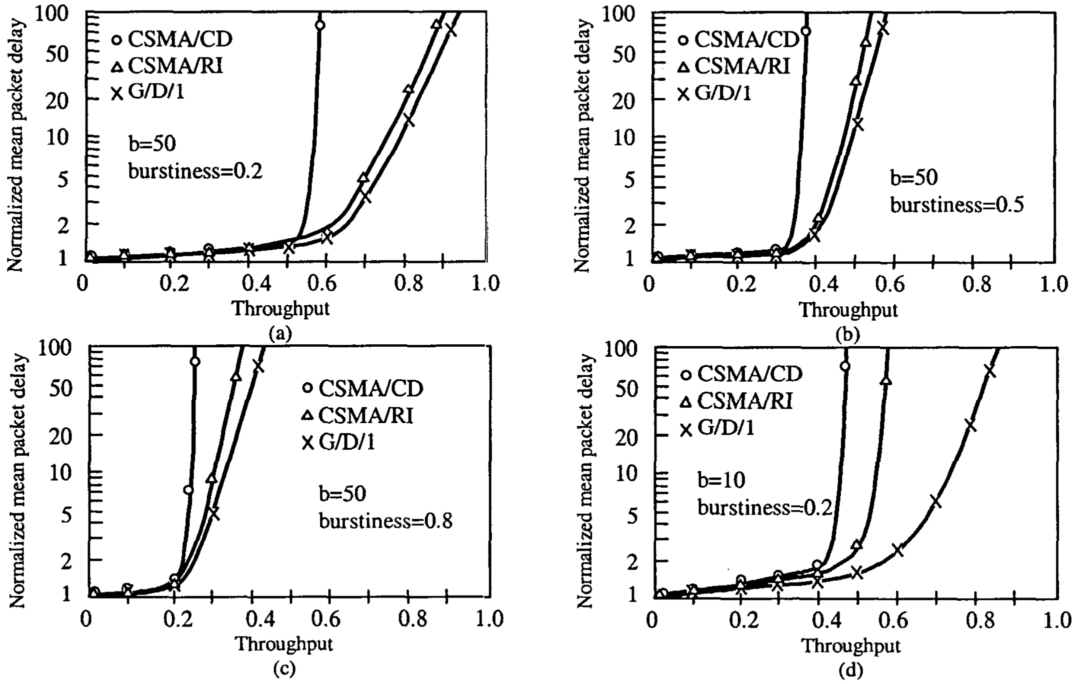

Fig. 6. CSMA/CD, CSMA/RI and G/D/1: delay versus throughput for different packet sizes and traffic burstiness