Improved Collision Resolution Algorithms for Multiple Access Channels with Limited Number of Users *

Chiung-Shien Wu † and Po-Ning Chen ‡

† Computer and Communication Research Labs.
ITRI, Hsinchu, Taiwan 310, R.O.C.

‡ Department of Communication Engineering,
National Chiao Tung University, Hsinchu, Taiwan

October 8, 1997

Subject Area: *Multiple access chennels, Data communications*

Correspondence Address:

Dr. Po-Ning Chen

Department of Communication Engineering

National Chiao-Tung University

1001, Ta-Hsueh Road, Hsin Chu Taiwan 30050, R.O.C.

E-mail:poning@cc.nctu.edu.tw

TEL: +886-3-5712121 ext. 54532

FAX: +886-3-5710116

* This work is currently supported by the *R.O.C. Ministry of Economic Affairs* under the project No. 3P12200 conducted by ITRI, and the *R.O.C. National Science Council* under the project No. NSC87-2213-E-009-139.

ABSTRACT

This paper presents a collision resolution scheme for multiple random access based on tree algorithms. Given the number of transmitters is small, simulations show that the proposed scheme improves ternary tree algorithms, which prove optimal under infinite *s-tations. It is concluded that exponentially increasing back-off windows with average around* 3 *may yield a better performance.*

Keywords: Tree algorithm, Contention resolution.

1 Introduction

The contention resolution technique has been considered as an essential part for distributed control of multiple access to shared channels in a communication system. Plural transmitters attached to the shared channel are time-synchronized and start their transmission of packets only at the beginning of a slot. When more than one transmitters send their packets to the same slot, a collision occurs and this status indicating COLLISION will feedback to all the transmitters for starting later retransmissions based on a random waiting time. If, fortunately, only a transmitter has sent its packet on a slot, then a status indicating NO-COLLISION will also feedback to inform a successful transmission.

Channel access algorithms using the above mechanism are called Random-Access Algorithms (RAA) and the slotted- ALOHA is the most original work [1]. However, the maximum throughput is only 0.36. Another algorithm called, Collision Resolution Algorithm (CRA), proposed by Capetanakis [2] and Tsybakov/Mikhailov [3] has adopted the tree resolution concept, and the maximum throughput is improved up to 0.407 [4]. A further improvement up to 0.4878 made by Mosely [5] in which ternary feedback and bias splitting was used. Usually, a random-access algorithm uses binary feedback scheme that indicates COLLISION or NO-COLLISION status as mentioned previously. Ternary feedback means that one of three feedback messages will be returned, namely IDLE, COLLISION, and NO-COLLISION, to enhance the performance. All the above works were analyzed by assuming infinite number of transmitters and the upper bound of the maximum throughput has been shown to be no greater than 0.587 [6].

A great improvement on RAA can also be achieved by using carrier sensing. This type of improvement is even more significant than the tree algorithms and thus many realistic networks such as Ethernet has adopted this method [7]. However, the carrier sensing function is sometimes hard or impractical to be implemented in communication medium such as wireless links, broadband coaxial cable, ..., etc. And it is usually the case that the channel is shared by a small number of users instead of infinite number of users. Therefore in this paper, we re-examine the tree algorithm under the condition that no carrier sensing function will be used and the number of users is finite. We also make a simple requirement that the channel status is binary feedback instead of ternary feedback.

2 Tree Algorithm

According to [4], there are two types of tree algorithms, namely the Free-Access Algorithm and the Blocked-Access Algorithm. Before describing the tree algorithm, we explain the following algorithm types.

1. **Free-Access Algorithm.** Newly arrived packets are transmitted immediately at the beginning of the next slot following their arrival.

2. **Blocked-Access Algorithm.** Newly arrived packets are transmitted in the first slot after all previous conflicts are resolved, i.e., new packets are blocked at their transmitters until all the other transmitters with collided packets have successfully retransmitted these packets.

The tree algorithm will work with one of above algorithms and produce different variation of algorithms and performance. The basic tree algorithm is described as follows.

3. **Basic n-ary Tree Algorithm.** After a collision, each transmitter that has sent a packet would flip an *"n-sided coin"* with value 1, 2, 3, …, n. This will split the contending transmitters into n subsets. For those transmitters in the subset with value i ($1 \leq i \leq n$) , we assign each of these transmitters an index i. Then, in the very next slot, transmitters with index 1 are permitted to send. If the outcome is NO-COLLISION (only one transmitter is with index 1), then each of the other transmitters with index greater than 1 will decrease the index by 1. If the outcome is COLLISION, then each of the other transmitters with index greater than 1 will increase the index by $n-1$ and the colliding transmitters with index 1 will once again flip the *"n-sided coin"* to resolve the contention.

The basic n-ary tree algorithm can be combined with the Free-Access Algorithm or the Blocked-Access Algorithm to form different protocol variations. Thus, the combination of 1. and 3. will produce the *Free-Access Tree algorithm (FAT)*. The combination of 2. and 3. will produce the *Blocked-Access Tree algorithm (BAT)*. Note that the best throughput is 0.403 for FAT and 0.368 for BAT when $n = 3$ and the number of transmitters is infinite.

3 The Hybrid-n Scheme

Given that a plurality of users are transmitting packets to a shared communication medium, the tree algorithm will divide the transmitting users into n groups upon detecting a collision condition and, in the later retransmission, collisions will only occur in between the users that fall into the same group. In FAT and BAT algorithms, the number of groups that the system is split into is fixed at each detected collision, for example, $n = 2$, or $n = 3$. However, we note that the number of collided transmitters in each slot may be dynamically changing. It is possibly helpful to let the value of n change at different point of collisions. For example, we let $n = 3$ in the first collision and $n = 2$ in the next collision. We refer to this scheme as the *Hybrid-n* in the sense that the value of n is not fixed. Here a *Transmission Splitting Vector*, $TSV = \{n_1, n_2, n_3, \ldots, n_k\}$, is used for the purpose of recording the changing sequence of the different value of n. For simplicity, we also refer to the original FAT and BAT algorithms with fixed value of n as the *Fixed-n scheme*. The parameter k in the Hybrid-n scheme is called the TSV length and it must be a positive integer. When $k = 1$, the scheme works as the Fixed-n scheme does. When $k > 1$, all the collided transmitters will be split into n_1 subgroups at the first collision, n_2 subgroups at the second collision, and n_i subgroups at the i-th collision. For the $(k+1)$-th collision, the system will rotate back to the first value of n, i.e., n_1.

4 Simulations Results

We observe the performance of the proposed scheme by simulation. We assume that the system has finite number of transmitters (users) and the packet arrival rate on each transmitter is assumed as the Poisson arrival process. A preliminary simulation result was summarized in Table I, where the delay performance is measured by the average number of time-slots that a transmitter should wait before a packet is successfully sent. In the first simulation, we let $TSV = \{3, 3, 3, 2\}$ for 10 transmitters. As shown in Table I, the performance for BAT is improved a little by using the Hybrid-n scheme. However, the FAT scheme is not improved by Hybrid-n. In the second simulation, we let $TSV = \{2, 4, 8, 2, 2, 2, 2, 2\}$ for 10 transmitters. From Table I, the performance for BAT is improved significantly by the Hybrid-n scheme. And the FAT scheme is also improved by Hybrid-n, although just a little. In the third simulation, we let $TSV = \{2, 4, 8, 16, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2\}$ for 10 transmitters. Also from Table I, the performance for both BAT and FAT is improved non-trivially by the Hybrid-n scheme. We also observe that the performance of the Hybrid-n scheme varies with the number of transmitters. In Figures 1 and 2, we set the number of transmitters to 50 and 5 respectively. The result shows that under a small number of transmitters, the Hybrid-n scheme will work better than the Fixed-n scheme. In Figure 3, the average throughput for BAT using Hybrid-n under different number of transmitters is shown. The result also conforms to that observed in the previous table and figures.

Table I: Simulation result of different contention resolution algorithms.

Methods		Max. Throughput	Delay 1 load=0.38	Delay 2 load=0.42	Delay 3 load=0.46
Fixed-n n=3	BAT	0.397	30.989	6536.618	9719.615
	FAT	0.446	9.482	96.070	2307.926
TSV1	BAT	0.422	30.387	1182.707	6116.560
	FAT	0.439	18.608	203.026	3069.802
TSV2	BAT	0.523	18.937	42.548	107.289
	FAT	0.491	17.052	107.596	1465.079
TSV3	BAT	0.524	33.495	50.221	76.811
	FAT	0.519	19.906	64.498	310.396

* *TSV1={3, 3, 3, 2}, TSV2={2, 4, 8, 2, 2, 2, 2, 2}, TSV3={2, 4, 8, 16, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2}*

The performance of Hybrid-n scheme is now summarized as follows.

1. Using the Hybrid-n scheme, the improvement on the blocked-access algorithm is better than that on the free-access scheme.

2. For both free- and blocked- access algorithms using hybrid-n, the performance is better when the number of users is not big, i.e., 5 - 20 users approximately.

3. The performance of using exponential-increased n followed by a sequence of $n = 2$ is better than others that use random-selected n.

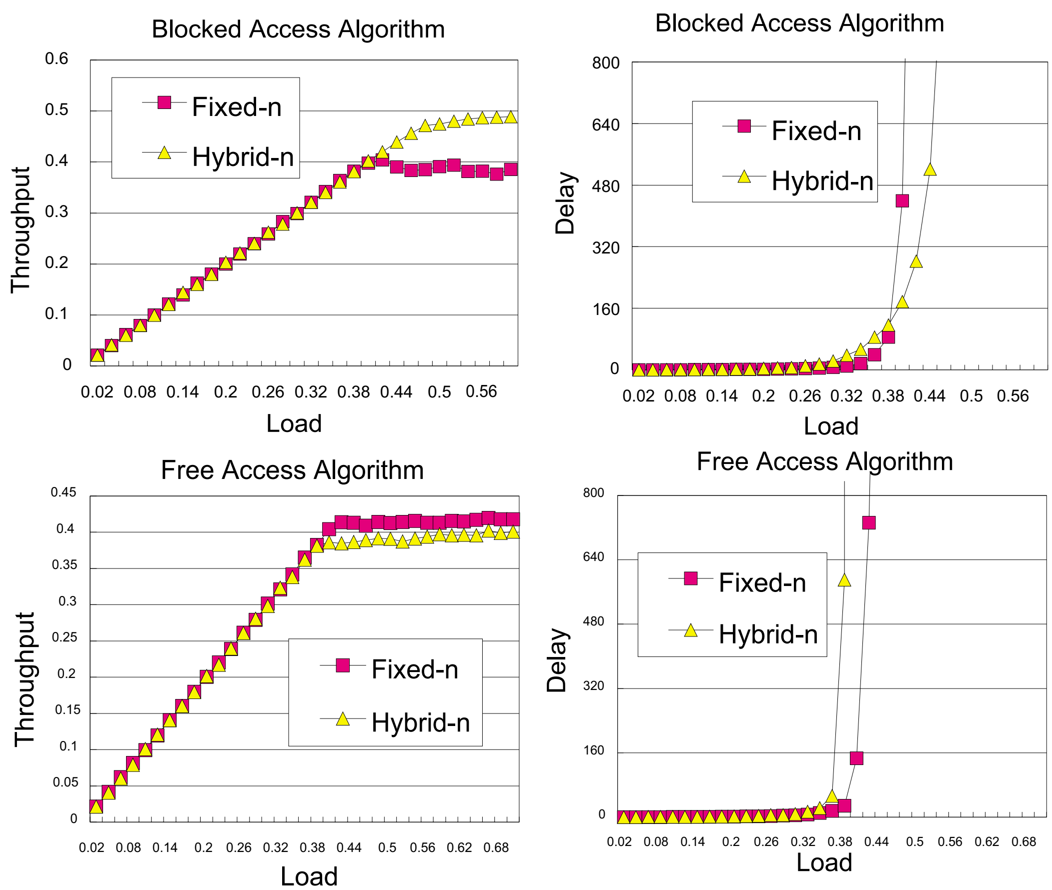

Figure 1: Simulation result for TSV={2, 4, 8, 16, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2}, 50 transmitters.

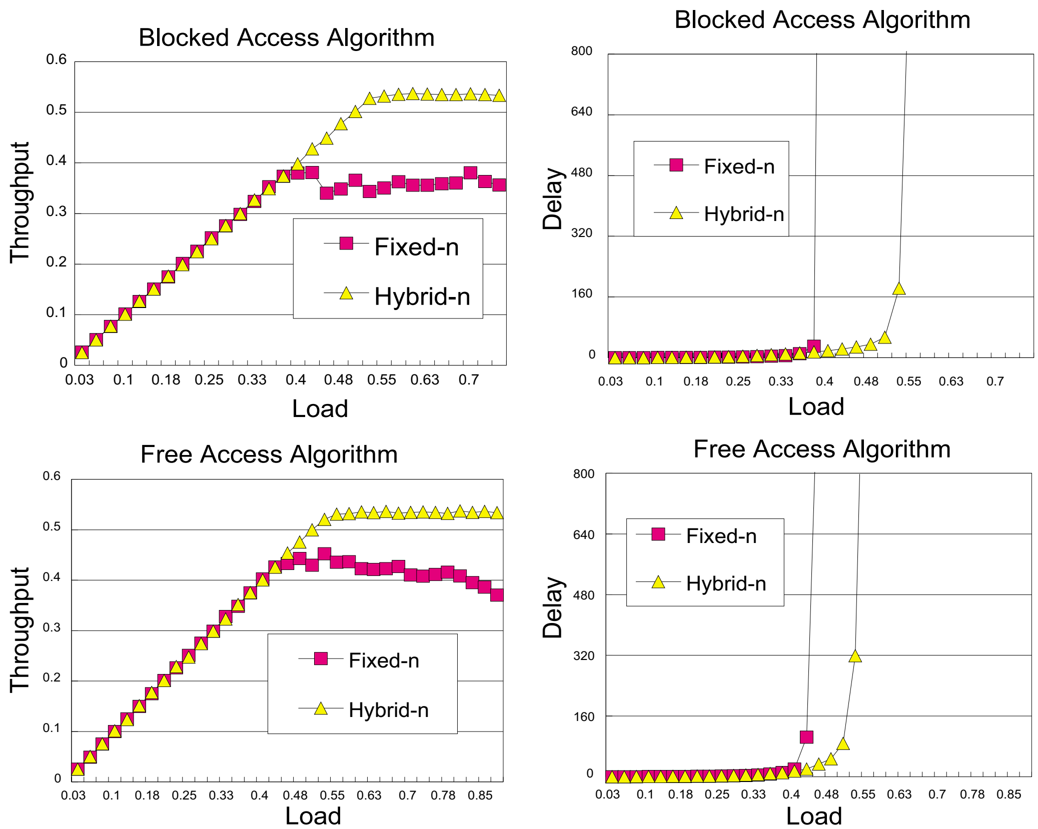

Figure 2: Simulation result for TSV={2, 4, 8, 16, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2}, 5 transmitters.

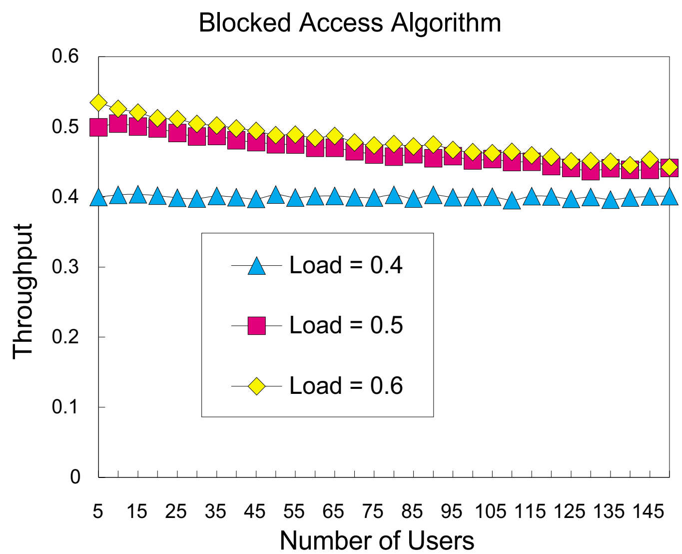

Figure 3: Result for different number of transmitters (BAT).

From the above results, we note that we have adopted some heuristic way in defining the TSV sequence to achieve a good performance. First, we know that the value of n or each $n_i \in$ TSV is strongly related to the time interval required to resolve a collision. Therefore, to have a good delay performance, we need a small value of n in average. Since $n = 3$ is optimal for Fixed-n scheme under infinite number of users, we let

$$\overline{n} = \frac{\sum_{i=1}^{k} n_i}{k} \approx 3 \tag{1}$$

For example, the average value of n is 3 when $TSV = \{2, 4, 8, 2, 2, 2, 2, 2\}$.

The second heuristic is that the sequence of n_i should be exponentially increasing, e.g., 2, 4, 8, 16, ..., etc. This is because that an exponentially increasing function on the value of n will possibly resolve the contention quickly. However, the drawback is the longer access delay. In Figure 4, we let $TSV = \{2, 4, 8, 16, 32, 64, 2, 2, 2, 2, 2, 2, 2, 2\}$, and thus the average value of each n is 10.14. The result shows that the throughput is very good, but the delay is significantly higher than that shown in Figure 3, where the average value of each n_i is 3. Therefore, we need to combine the above- mentioned heuristics with the previous heuristic to get an overall acceptable performance.

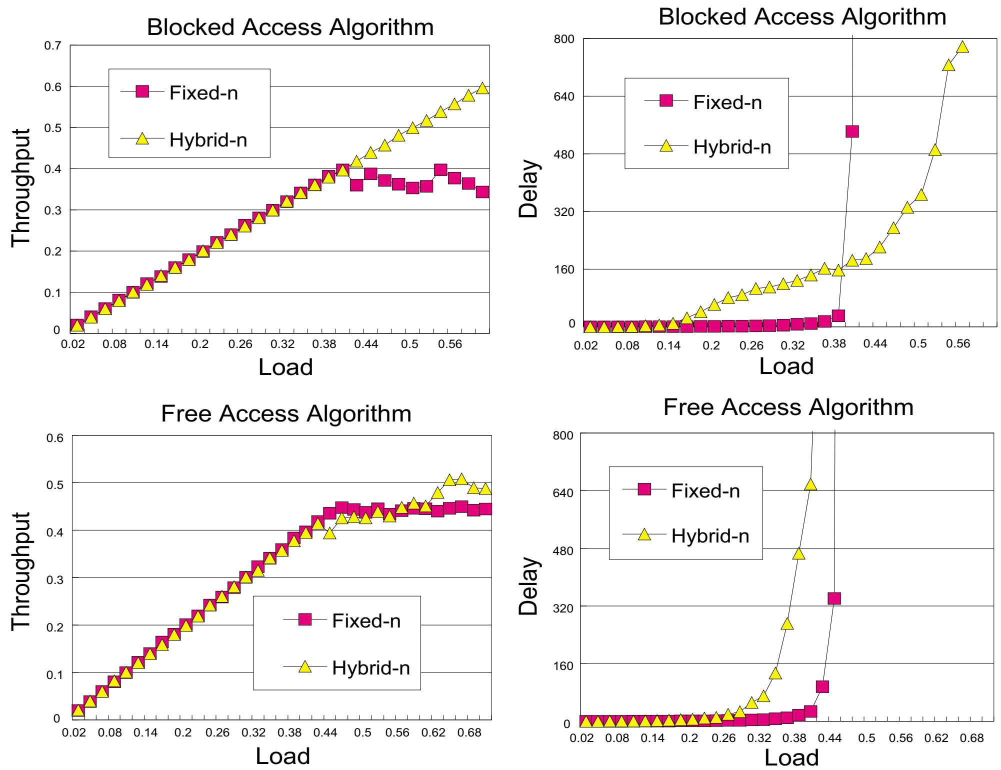

Figure 4: Simulation result for TSV={2, 4, 8, 16, 32, 64, 2, 2, 2, 2, 2, 2, 2, 2}, 10 transmitters.

An interesting characteristic on the Hybrid-n scheme is observed here. The performance improvement on the blocked-access algorithm is better than that on the free-access algorithm. We try to explain the phenomenon as follows. The Hybrid-n scheme was proposed to improve the tree algorithm by a sequence of dynamic n in the hope that the collision will be resolved quickly. Now given that the number of colliding users is fixed within a time interval, the dynamically changing sequence of n will help if, at each collision, the number that these users are split into is optimally designed. However, if the number of colliding users will be increasing during the time of collision resolution, then the effect of Hybrid-n will be somehow released by the new arrival of packet requests.

5 Concluding Remarks

In a wireless environment, efficient contention resolution is very important to the MAC protocols when the communication channel is to be accessed by multiple wireless users. Protocols based on packet reservation are commonly used when the pure contention-based access is not good enough, e.g., the PRMA (Packet Reservation Multiple Access) [8], or PRMA++ [9]. For example, since maximum throughput of the well-known slotted ALOHA is 0.36, the packet reservation scheme will allow a user to send 3 packets in average if this user successfully sends its request to a central controller via the contention-based access. And thus, the throughput can be achieved up to almost 1 because $0.36 \times 3 > 1$. However, the delay will be longer and a unfairness may occur between the wireless users since each user is allowed to send a burst of data. If the efficiency of the contention-based access can be higher, then the maximum burst length that each user must reserve can be decreased.

In this paper, we proposed a Hybrid-n scheme that achieves a throughput of 0.5 when the number of users is small. According to the packet reservation protocol, the maximum burst length is now only 2 in order to have a throughput of 1 (since $0.5 \times 2 = 1$). A lower delay and a more fair access can be obtained. We now conclude this paper by giving a summary on the proposed scheme, as follows.

1. The Hybrid-n scheme adopts a dynamic change on the sequence of n.

2. The average value of each n is approximately 3 for a better performance.

3. An exponentially increasing function on the beginning elements of the Hybrid-n sequence is better.

4. The Hybrid-n scheme works well with the blocked-access tree algorithm.

References

[1] N. Abramson, "The ALOHA system V Another Alternative for Computer Communications", in Proc. *AFIPS Conf. Fall Joint Comp. Conf.,* 1970, vol .37, pp. 281-285.

[2] J. Capetanakis, "The Multiple Access Broadcast Channel: Protocol and Capacity Considerations", *Ph.D. dissertation, Massachusetts Institute of Technology.,* Cambridge, 1977.

[3] B.S. Tsybakov, and V.A. Mikhailov, "Free Synchronous Packet Access in a Broadcast Channel with Feedback", pp. 259-280, *in Plenum Press translation of article in Problemy Peredachi Informatsii,* vol. 14, no. 4, pp. 32-59, Oct.-Dec. 1978.

[4] P. Mathys, and P. Flajolet, "Q-ary Collision Resolution Algorithms in Random- Access Systems with Free or Blocked Channel Access", *IEEE Transactions on Information Theory,* vol. IT-31, no. 2, pp. 218-243., March 1985.

[5] J. Mosely, and P.A. Humblet, "A Class of Efficient Collision Resolution Algorithms for Multiple Access Channels", *IEEE Transactions on Communications,* vol. COM-33, no. 2, pp. 145-151, Feb. 1985.

[6] V.A. Mikhailove, and B.S. Tsybakov, "Upper Bound for the Capacity of a Random Multiple Access System", *in Problemy Peredachi Informatsii (USSR),* 17:90-95, 1981.

[7] D. Bertsekas, and R. Gallager, "Data Networks", *second edition, Prentice Hall, Inc.,* 1992.

[8] D.J. Goodman, "Packet Reservation Multiple Access for Local Wireless Communications", *IEEE Transactions on Communications,* vol. 37, no. 8, pp. 885-890, Aug. 1989.

[9] J. Dunlop, et al., "Performance of a Statistically Multiplexed Access Mechanism for a TDMA Radio Interface", *IEEE Personal Communications,* pp. 56-64, June 1995.