Evaluation of a Distributed Broadcast Scheduling Protocol for Multihop Radio Networks

Xiaopeng Ma, Errol L. Lloyd
Department of Computer and Information Sciences
University of Delaware,
Newark, DE 19716

ABSTRACT

In [6] we introduced a fully distributed algorithm (FDAS) for broadcast scheduling in multihop packet radio networks. In this paper we analyze the performance of that algorithm relative to non-distributed broadcast scheduling algorithms and to a random access method. Both analytical and simulation results are presented. These analyses establish that FDAS has superior throughput and medium access delay performance at both the node and system levels.

I. Introduction

Although broadcast scheduling has been widely studied in packet radio networks (PRNs) (e.g. [2], [1], [7], [3], [8]) the first fully distributed algorithm (*FDAS*) was developed only recently [6]. In FDAS:

- No global information is required, either in central or individual sites. Rather, a station schedules itself after collecting information from it's one-hop and two-hop neighbors.
- The method is fully distributed. Thus, multiple stations can simultaneously run the scheduler portion of the protocol, and stations in non local portions of the network can transmit normally even while stations are joining or leaving the network.

In this paper we analyze the performance of FDAS relative to non-distributed broadcast scheduling algorithms and to an ALOHA ([9], [4]) based channel access method. Both analytical and simulation results are presented. The analyses establish that FDAS has superior throughput and medium access delay performance at both the node and system levels. This fact makes FDAS especially appealing for military situations where communication can be guaranteed even under very heavy loads and where the network configuration is dynamically changing.

II. Channel access - Background

The network that we study is a broadcast multihop PRN. In such a network, it is required that the transmissions of a station be received by all of that station's one-hop neighbors. If a collision occurs, then the station must retransmit until the transmission is received by each of the station's one-hop neighbors. At any given time, a station may transmit or receive, but not both. If a station simultaneously receives packets from multiple sources, then no packet can be recovered and a retransmission is necessary. If a station receives a transmission without collision, then it is assumed to be error free. As is standard, we assume: that all stations are equipped with omnidirectional antennas; that there is global time synchronization among all stations; and, that all transmissions occur at the beginning of a time slot.

A. M-ALOHA: A random access method

For channel access, the most widely used approach is *random access*. Here, stations transmit of their own volition, without mutual coordination, and if a collision occurs, then stations retransmit. One such method is *M-ALOHA*, a slotted ALOHA where:

- packets are all of a fixed size;
- time is partitioned into slots of duration equal to the transmission time of a packet plus the longest propagation time between one-hop neighbors;
- when the queue is non-empty, the packet at the front of the queue is transmitted with probability p;
- the transmitting station knows whether or not the transmission was successful before the next time slot

Prepared through participation in the Advanced Telecommunications-Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAL01-96-2-0002.

begins[1]. If there was a collision, the packet remains in the queue until it is successfully transmitted.

B. Broadcast scheduling and FDAS

An alternative to random access is the method of scheduling access to the channel so that collisions are guaranteed to not occur. As such, a *broadcast scheduling algorithm* produces an infinite *schedule* so that each station in the network is periodically assigned a slot for transmission and so that all transmissions are received collision free. The potential drawback of broadcast scheduling is that stations are restricted to transmitting in particular *slots* rather than *at will* as in the random access methods.

In FDAS, each station in the network is assigned a *transmission slot* and a *transmission cycle*[6]. Here, a station transmits for the first time in its transmission slot, and then every transmission cycle number of slots thereafter. The requirement for this assignment is that for any station, when it's time for the station to transmit, none of its one-hop and two-hop neighbors transmits. Obviously, for a station (system), a minimum (average) transmission cycle is desired. In FDAS each station first assigns itself a transmission slot using information only about it's one and two away neighbors and then the station sets its transmission cycle to be the least power of two that is at least as big as the largest transmission cycle of it's one and two away neighbors.

C. Performance Measures

In this paper, we study throughput, medium access delay and throughput-delay tradeoffs. Following standard practice, we define the *throughput* of a station to be the expected number of packets that can be successfully transmitted by the station per time slot, and the *medium access delay* of a station is the expected number of time slots between a packet's arrival at the station and it's successful transmission.

III. An analytic evaluation

In [5] we took a node-oriented approach to determine the throughput and the medium access delay for broadcast scheduling and M-ALOHA. For broadcast scheduling, those results established:

$$S_B = \lambda \tag{1}$$

$$D_B = T[\lambda T + \frac{(\lambda T)^2}{2(1-\lambda T)}] + \frac{T}{2} \tag{2}$$

where S_B is the throughput, D_B is the medium access delay, T is the transmission cycle of the station X being analyzed, and λ is the average arrival rate of packets to that station's queue.

In the case of M-ALOHOA, the focus for a station X is on its *interference set*. That is, the station X itself, along with all of its one-hop and two-hop neighbors. In the analysis we assumed that all of the stations in an interference set have similar statistical characteristics. Thus, the stations within a given interference set not only transmit with the same probability, but they also have the same rate of successful transmissions. In [5] we showed that the throughput S_M and medium access delay D_M of M-ALOHA are:

$$S_M = \left(\frac{\beta}{1+\beta}\right)^{\beta} \cdot \frac{1}{1+\beta} \tag{3}$$

$$D_M = \frac{\alpha}{\beta\lambda(1-\lambda)} \tag{4}$$

where λ denotes the arrival rate at X and β denotes the number of one-hop and two-hop neighbors.

A. A Comparison on Capacities

In previous analysis, the throughput and medium access delay for broadcast scheduling are based on the packet arrival rate (λ) and the transmission cycle (T), and for M-ALOHA they are based on the packet arrival rate (λ) and on the size (β) of the unit subset. As such, the ratio of the capacities for broadcast scheduling C_B and M-ALOHA C_M follow (after maximizing and simpifing) from equations (1) and (3) as:

---

[1] We do not concern ourselves with the practicality of this implementation, which if anything, favors M-ALOHA over broadcast scheduling.

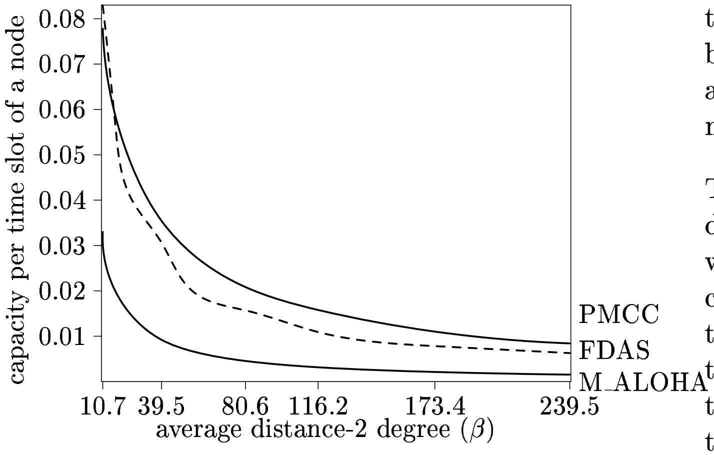


Fig. 1. A comparison of node capacities

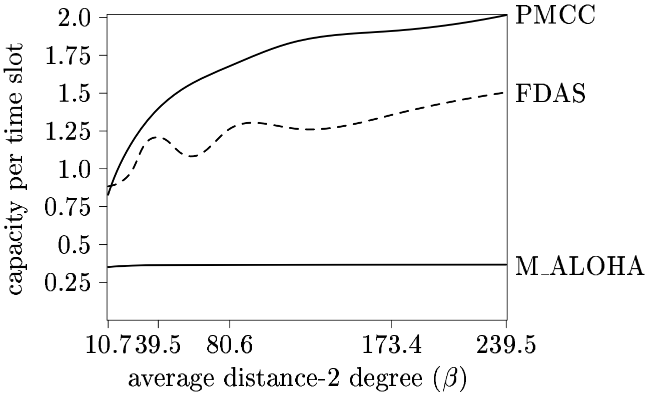


Fig. 2. A comparison of unit subset capacities

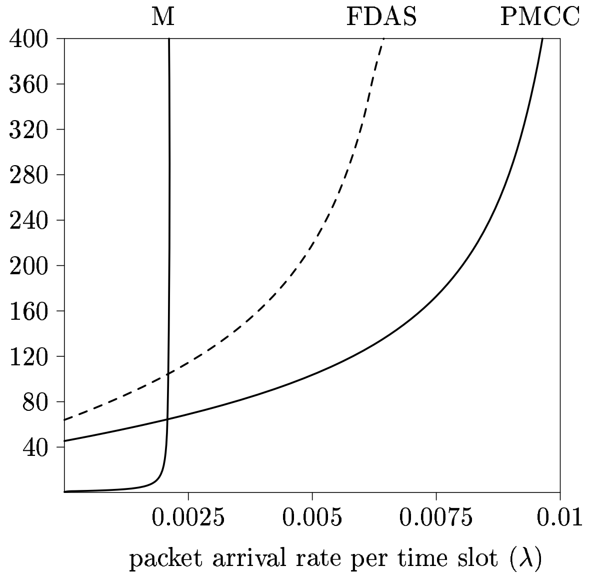
$$C_B/C_M = \frac{1+\beta}{T}(1+\frac{1}{\beta})^\beta \tag{5}$$

Equation (5) implies that for a fixed β, a smaller T results in a larger ratio, and hence under the condition $T < (1+\beta)(1+\frac{1}{\beta})^\beta$, the capacity of broadcast scheduling exceeds that of M-ALOHA.

Now the question is, does the above inequality generally hold for FDAS? Figures 1 and 2 illustrate the capacities for *FDAS*, along with *max_cont_color* (denoted there by *PMCC* and M-ALOHA). The figures are plotted by applying the relevant parameters T and β obtained from the simulation of *FDAS* and *PMCC* respectively. In calculating the capacities for M-ALOHA here, we utilized the average distance-2 degree of the nodes in the tested networks of roughly the same density (as measured by the average distance-2 degree of the nodes in a network) as the parameter β. In calculating the capacities for the broadcast scheduling protocols, we utilized the average transmission cycle of the nodes in the involved networks as the parameter T.

The x-coordinates in Figure 1 denote the average distance-2 degree of the nodes in the involved networks. The y-coordinates in Figure 1 denote the capacity per time slot of a node. That is, a protocol with a value y for its y-coordinate indicates that y packets can be expected to be successfully transmitted by a node in a time slot under that protocol. It can be seen that both of the broadcast scheduling protocols has a much higher capacity than could be reached by M-ALOHA. The capacity of the *FDAS* fluctuates for networks of different densities, but is still consistently much higher than that of M-ALOHA, and a consistenly a bit weaker than that of *PMCC*. Note that as the average distance-2 degree grows from 10.7 to 239.5, FDAS has a capacity that ranges from 2.51 to 4.10 times greater than M-ALOHA. Overall the capacity of PMCC averages about 15% greater than FDAS.

From Figure 1 and the capacity ratios, we conclude that *FDAS* (as does *PMCC*) has more than a 2-fold advantage over M-ALOHA in terms of node capacity. Second, as the number of distance-2 neighbors grows, so does the throughput advantage of *FDAS* over M-ALOHA: there is a nearly 5-fold advantage when the average number of distance-2 neighbors is about 240. This is because as the number of distance-2 neighbors of a station becomes larger, so does the potential for collisions among transmissions by those stations in M-ALOHA. In *FDAS*, though the schedule likely grows a bit longer with additional stations, there are more opportunities for stations to share time slots in the schedule. To see this point, Figure 2 plots the capacities that can be achieved by a unit subset in the protocols. From that figure, it can be seen that the capacity that can be achieved by a unit subset in M-ALOHA is virtually constant across network densities at roughly 0.36 packets per time slot. In contrast, both *FDAS* and *PMCC*, the capacity of a unit subset grows (almost) steadily as the distance-2 neighbors grows. Note also that *FDAS* consistently underperforms *PMCC* by a modest amount. This is not surprising given that *PMCC* is the strongest of the centralized methods for broadcast scheduling.



(b) $\beta = 173.4$

Fig. 3. A comparison of medium access delay

B. A Comparison of Medium Access Delay

In comparing the relative medium access delay of *FDAS* to that of M-ALOHA and *PMCC*, consider Figure 3. In that figure, there are two sub-figures, one for each of two values of β—the average number of distance-2 neighbors of the nodes in the network. The values for β range from medium to relatively large. In these sub-figures, the x-coordinates indicate the packet arrival rate to a station. The y-coordinates indicate the expected medium access delays of the station.

From Figure 3, the following can be observed.

- For a given β and a given T, there exists a value γ such that, when the packet arrival rate λ is less than γ, then the medium access delay of M-ALOHA is superior to that of *FDAS*. The value of γ is relatively small - corresponding to a packet arrival rate under 0.0025 per time slot.
- On average, the least medium access delay for *FDAS* is half of the transmission cycle. So, if the transmission cycle is large, then the least medium access delay is also fairly large. In contrast, the least medium access delay for M-ALHOA can approach 1, *if* the combination of λ and β is sufficiently small.
- Although M-ALOHA can sometimes achieve a very short access delay, because of the randomness in the success of transmission, it also has the potential that a station is blocked for an unpredictably long period even if the average packet arrival rate is well under its capacity. Clearly this cannot occur in *FDAS*.
- For a given medium access delay, the maximum packet arrival rate for *FDAS* is about 70% that of *PMCC* and is significantly more than that of M-ALOHA.

IV. Simulation Results

While the analysis in the proceeding section provides convincing evidence that FDAS has a strong performance both in regard to centralized scheduling methods and random access methods, that analysis focussed on the individual node level. In this section we evaluate the relative performances of FDAS against PMCC and M-ALOHA at the *system level* in terms of *system throughput*[2] and *system delay*[3]. Specifically, the protocols PMCC, FDAS and M-ALOHA are evaluated utilizing the comprehensive discrete event simulation tool OPNET.

A. OPNET Model

To compare the relative performance of broadcast scheduling to that of M-ALOHA at the system level, we simulated the protocols PMCC, FDAS, and M-ALOHA utilizing OPNET. Under OPNET, a network consists of a collection of nodes or stations that communicate with each other. A node is usually consists of several co-related (software and/or hardware) devices[4]. Each of the devices in OPNET modeling is driven by a *process model* that controls the underlying functionality of the device. Thus, we developed (written in C and the EMA (External Model Access) package[5] of OPNET for fast deployment of a PRN):

- a node model to specify the constituting devices of a node and the packet flows between the devices
- a process model that simulates broadcast scheduling protocols PMCC and FDAS

---

[2] the expected number of successful packets that can be transmitted by all of the stations in a network per time slot

[3] the expected number of time slots experienced by a packet from the arrival of the packet to any of the stations in a network until the successful transmission of the packet

[4] A device is referred to as a *module* in OPNET documents.

[5] For a description of the EMA package, please refer to the External Interface volume of the OPNET document.

- a process model that simulates M-ALOHA
- a program for fast deployment of a multi-hop packet radio network

Below is a description of the models.

A.1 The Node Model

We utilized a uniform node model for both type of networks: broadcast scheduling and M-ALOHA. In our specification, a node consists of these devices:

- a packet generator to generate packets according to an exponential inter-arrival (Poisson arrival) distribution and a specified packet arrival rate.
- a processor to implement a simulated medium access control protocol PMCC or FDAS or M-ALOHA.
- a radio transmitter to transmit packets.
- a radio receiver to receive packets from neighbors.
- a sink to dispose of completed packets

Among the devices that we utilized in our node model, the packet generator, the radio transmitter, the radio receiver and the sink are standard process models that are provided by the OPNET package. For the processor, we developed two process models *P_scheduling* and *P_maloha* respectively for broadcast scheduling and for M-ALOHA (see below).

The devices in our node model are connected by packet streams that interact in the expected fashion and are not described here.

A.2 The Process Models

The Process Model for Broadcast Scheduling. The process model *P_scheduling* (Figure 4) contains 7 states. Most relevant is state *idle*, where three events or interrupts can initiate a transition:

- When a packet is generated by the packet generator of the node (indicated by the event *pk_from_G*), then the process enters state *enqueue* to place the packet at the tail of the queue Q (maintained by the node for the packets to be transmitted).
- When a time slot for the node to transmit occurs (indicated by the event *my_slot*), then the process enters state *slot*. In state *slot*, there are two cases. In the case where the Q is empty, the process returns to the *idle* state. Otherwise, the process enters state *transmit*. In state *transmit*, the packet *pk* at the head of Q is sent to the radio transmitter for transmission, and the statistics about packet delay and throughput are modified. Then, the process goes through state *dequeue* to remove the packet *pk* from Q and sends *pk* to sink for disposition. After that, the process returns back to *idle* state.
- When the end of simulation interrupt is received, the process enters state *end*.

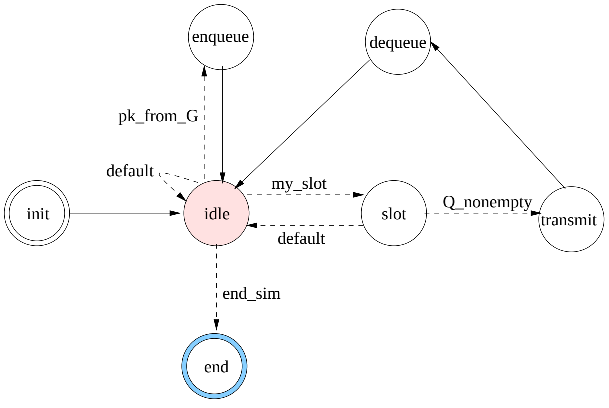


Fig. 4. The process model for broadcast scheduling

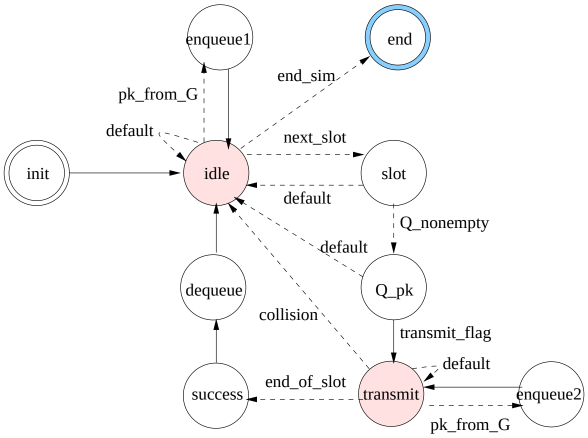


Fig. 5. The process model for M-ALOHA

The Process Model for M-ALOHA. The process model *P_maloha* (Figure 5) consists of 10 states. Most relevant is the state *idle* where three events or interrupts can initiate a transition to other states:

- When a new packet is generated by the packet generator of the node (indicated by the event *pk_from_G*), then the process enters state *enqueue1* to place the packet at the tail of the queue Q.

- When a new time slot begins the process enters state *slot*. If Q is non-empty, the process enters state *Q_pk* (see below). Otherwise, if Q is empty, the process returns back to *idle* state.
- When end of simulation interrupt is received by the process, then the process enters state *end*.

Upon entering state *Q_pk*, the process decides whether or not to transmit the packet *pk* at the head of Q. If it is decided that the node transmits packet *pk* in the current time slot, the process enters *transmit* state. Otherwise, the process goes back to *idle* state.

When the process is in state *transmit*, the process sends the packet *pk* to the transmitter for transmission. After that, the process stays in the state *transmit* and acts accordingly as follows.

- If a collision has been detected, then the process goes back to state *idle*. We will discuss how to detect a collision shortly.
- If a new packet is generated by the packet generator of the node, then the process enters state *enqueue2* to place the new packet at the tail of the queue Q and then returns back to state *transmit* unconditionally.
- At the conclusion of the current time slot (which indicates that the transmission of packet *pk* succeeded) then the process enters state *success*. In state *success*, the statistics about packet delay and throughput are modified. Then the process goes through state *dequeue* to remove the packet *pk* from Q and sends *pk* to sink for disposition. After that, the process returns back to *idle* state.

In our analytical analysis, under the assumption that all of the stations in the unit subset have similar statistical characteristics, a maximum successful rate of transmission can be achieved if node B transmits with probability $1/(1+\alpha)$ in a time slot when B has a non-empty queue. Here, α is the number of distance-2 neighbors of B that have a non-empty queue at the beginning of the time slot. However, in reality, the values of α for different stations in the unit subset of B may differ by a significant degree and may vary from time to time. This means that it is impractical to statically determine a transmission probability. Thus, a critical question in the process model *P_maloha* is: if the queue Q is non-empty at the beginning of a time slot, how should it decide whether or not to transmit the packet at the head of the queue Q? In our simulation, we utilized the *binary exponential back-off algorithm* [9].

	Nodes	Mx β	Av β	T_{PMCC}	Av T_{FDAS}
N_1	200	27	14.09	13	14.16
N_2	400	49	30.29	24	30.63

TABLE I
SUMMARY OF N_1 AND N_2

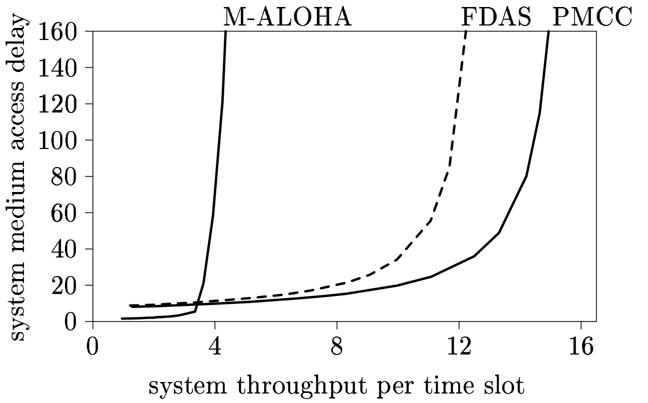
B. Simulation Comparison of System Performance

We simulated two networks, each generated on an area of 100 by 100 grid units. The transmission radius R was 10 grid units. The first network N_1 consisted of 200 nodes. The second network N_2 consisted of 400 nodes. The networks were generated by randomly placing the indicated number of nodes in the network area. Table I provides a summary of the resulted networks N_1 and N_2. In Table I, the third column indicates the D2 degree of the network; the fourth column indicates the average D2 degree of the nodes in the network; the fifth column indicates the uniform transmission cycle of the protocol PMCC and the last column indicates the average transmission cycle of the protocol FDAS.

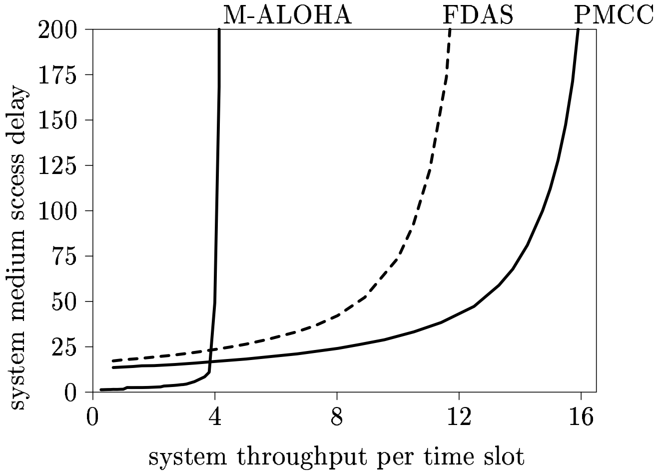
In each trial of the simulation, all of the stations were provided with the same packet arrival rate. Each trial either has a duration that is 7200 time slots or has such a duration that on average each station transmits at least 120 times.

The throughput-delay trade-offs of the protocols obtained from the OPNET simulations are presented in Figure 6. From these results, we conclude:

- The maximum system throughput of *FDAS* is intermediate to that of M-ALOHA and *PMCC*: FDAS has more than a 2.5-fold advantage over M-ALOHA in network N_1, and a more than 3-fold advantage in network N_2. By comparing the results for N_1 and N_2, we also observe that as the (average) number of distance-2 neighbors grows, so does the maximum throughput advantage of *FDAS* over M-ALOHA. By comparison, *FDAS* loses about 25% in throughput



(a) for a network of 200 nodes



(b) for a network of 400 nodes

Fig. 6. A Comparison: FDAS to PMCC and M-ALOHA

to *PMCC* for any given medium access delay.

- The minimum system delay for M-ALOHA can approach as small as about one time slot when the network is extremely lightly loaded. On the other hand, the minimum system delay for *FDAS* is at least half of the (average) transmission cycle even when the network is extremely lightly loaded.
- There is a "threshold value" τ such that, when the packet arrival rate is less than τ, then the system delay of M-ALOHA is superior to that of *FDAS*. Otherwise, the reverse is true. Results for both N_1 and N_2 show that τ hovers just under 4. That is, when the packet arrival rate to the network is above 4, then the delay for *FDAS* is superior to that of M-ALOHA. Indeed, for higher arrival rates than 4 packets per time slot, M-ALOHA is unable to handle the load. In comparing *FDAS* with *PMCC* there is no such threshold, as *FDAS* is always weaker than *PMCC*.
- Although space prevents a thorough discussion, we note that the analystic results described earlier and the simulation results described are very highly correlated.

REFERENCES

[1] Arr-Mien Chou and Victor O.K.Li. Slot allocation strategies for TDMA protocols in multi-hop packet radio networks. *IEEE INFOCOM*, 2:710–716, 1992.

[2] Anthony Ephremides and Thuan V. Truong. Scheduling broadcasts in multi-hop radio networks. *IEEE Transactions on Communications*, 38(4):456–460, April 1990.

[3] Mark L. Huson and Arunabha Sen. Broadcast scheduling algorithms for radio networks. *IEEE MILCOM*, pages 647–651, 1995.

[4] Hisashi Kobayashi, Yoshikuni Onozato, and Dieu Huynh. An approximation method for design and analysis of an ALOHA system. *IEEE Transactions on Communications*, COM-25(1):148–157, January 1977.

[5] X. Ma and E. Lloyd. Throughput/delay of broadcast scheduling in multihop packet radio networks. *Proceedings of the Fourth Annual Federated Laboratory Symposium on Advanced Telecommunications/ Information Distribution Research Program (ATIRP)*, pages 239–243, 2000.

[6] Xiaopeng Ma and Errol L. Lloyd. A distributed protocol for adaptive broadcast scheduling in packet radio networks. *Workshop record of the 2nd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIAL M for Mobility) 10 pages*, October 1998.

[7] Ram Ramanathan and Errol L. Lloyd. Scheduling algorithms for multi-hop radio networks. *IEEE/ACM Transactions. on Networking*, 1:166–177, 1993.

[8] Arunabha Sen and Mark L. Huson. A new model for scheduling packet radio networks. *IEEE INFOCOM*, pages 1116–1124, 1996.

[9] Andrew S. Tanenbaum. *Computer Networks.* Prentice-Hall, Inc., Upper Saddle River, New Jersey 07458, 1996.