On the Use of Learning Automata in the Control of Broadcast Networks: A Methodology

Georgios I. Papadimitriou, *Senior Member, IEEE*, Mohammad S. Obaidat, *Senior Member, IEEE*, and
Andreas S. Pomportsis

***Abstract*—Due to its fixed assignment nature, the well-known time division multiple access (TDMA) protocol suffers from poor performance when the offered traffic is bursty. In this paper, an adaptive TDMA protocol, which is capable of operating efficiently under bursty traffic conditions, is introduced. According to the proposed protocol, the station which is granted permission to transmit at each time slot is selected by means of learning automata (LA). The choice probability of the selected station is updated by taking into account the network feedback information. The system which consists of the LA and the network is analyzed and it is proven that the choice probability of each station asymptotically tends to be proportional to the probability that this station is not idle. Although there is no centralized control of the stations and the traffic characteristics are unknown and time-variable, each station tends to take a fraction of the bandwidth proportional to its needs. Furthermore, extensive simulation results are presented, which indicate that the proposed protocol achieves a significantly higher performance than other well-known TDMA protocols when operating under bursty traffic conditions.**

***Index Terms*—Bursty traffic, learning automata (LA), learning medium access control (LMAC), population-based incremental learning (PBIL), time division multiple access (TDMA).**

I. INTRODUCTION

THE KEY issue in broadcast networks is how to determine who gets to use the channel. A broad range of demand assignment, random access, and fixed assignment protocols have been proposed as solutions to this problem.

Demand assignment protocols [1], [2] are based on a signaling procedure which allows certain network entities to be informed about the transmission and networking needs and demands of the network stations. Random access protocols [1], [2] are characterized by the fact that stations contend for access to the communications channel, in accordance with an algorithm that can lead to colliding transmissions. All of the collided packets are scheduled for retransmission. Fixed assignment protocols [1]–[11] assign a fixed portion of the available bandwidth to each station. In this way, collisions are avoided. Due to the absence of collisions, protocols of this family achieve a high performance when the traffic of each station is stable and *a priori* known. However, when the traffic is bursty, fixed assignment protocols are not capable of being adapted to the sharp changes of the stations' traffic [1], [3]. Therefore, their performance is dramatically degraded.

In this paper, an adaptive time division multiple access (TDMA) protocol which is capable of operating efficiently under bursty traffic conditions is introduced. According to the proposed protocol, the station which is granted permission to transmit is determined by means of learning automata (LA) [12]–[28] that implement a variation of the population-based incremental learning (PBIL) algorithm [29]–[31]. LAs are adaptive decision making devices that operate in unknown stochastic environments and progressively improve their performance via a learning process. The reader can consult [14] in order to study the various families of LAs.

At each time slot, the LAs take into account the network feedback information in order to update the choice probability of the selected station. The probability updating scheme is designed in such a way, that the choice probability of each station asymptotically tends to be proportional to the probability that this station is not idle. In this way, the number of idle slots is minimized and the network performance is significantly improved. When the traffic conditions of a station change, this leads to a change of the choice probability of this station. Therefore, the protocol is capable of being adapted to the sharp load changes of a bursty traffic environment.

The proposed learning medium access control (LMAC) protocol is applicable to a broad range of broadcast network architectures, including bus, star, and wireless local area networks (LANs). This paper focuses on the theoretical aspects of LMAC rather than on its application to specific network architectures.

The paper is organized as follows. The problem formulation is given in Section II. The proposed LMAC protocol is presented in Section III, while an analysis of the asymptotic behavior of the proposed scheme is presented in Section IV. In Section V, extensive simulation results are presented which indicate the superiority of the LMAC protocol over other well-known TDMA protocols. Finally, concluding remarks are given in Section VI.

II. PROBLEM FORMULATION

Let $U = \{u_1, \ldots, u_N\}$ be the set of stations, where N is the number of stations. All the stations are connected to a broadcast transmission medium (e.g., a copper wire or an optical coupler). Data are transmitted in the form of packets. All packets are of equal length. The time axis is slotted, with the slot duration being equal to the packet transmission time. Packet transmissions are synchronized with the time slots. Thus, no packet transmission is allowed to start in the middle of a time slot.

Manuscript received November 15, 2001; revised March 1, 2002.

G. I. Papadimitriou and A. S. Pomportsis are with the Department of Informatics, Aristotle University, 54006 Thessaloniki, Greece (e-mail: gp@csd.auth.gr; apombo@csd.auth.gr).

M. S. Obaidat is with the Department of Computer Science, Monmouth University, West Long Branch, NJ 07764 USA (e-mail: obaidat@monmouth.edu).

Publisher Item Identifier S 1083-4419(02)06469-5.

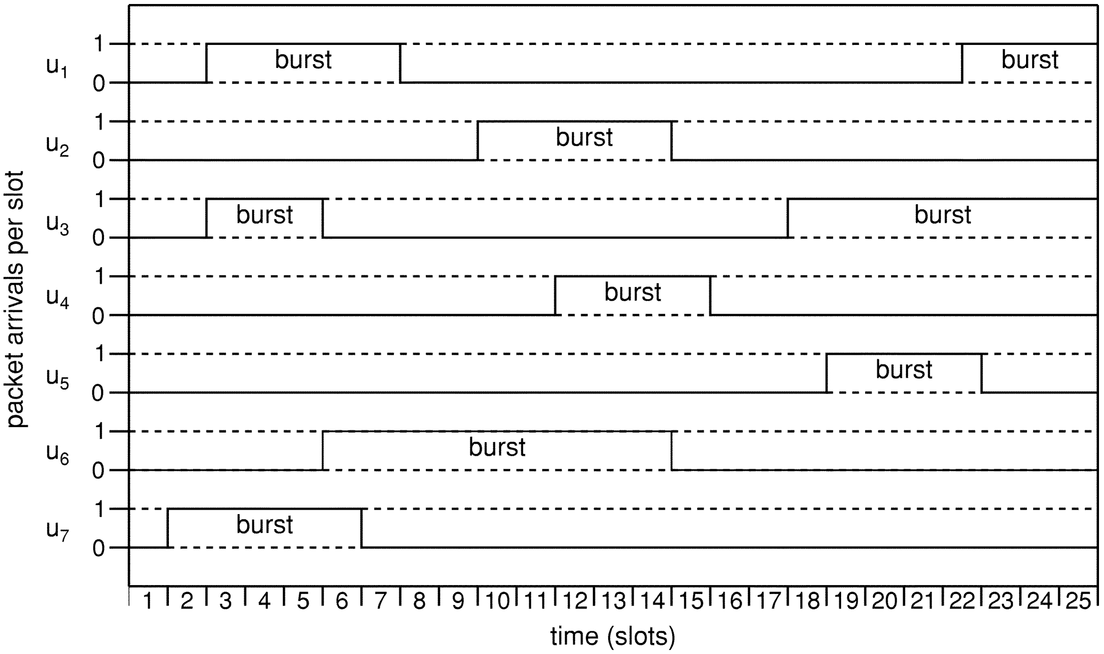


Fig. 1. Snapshot of bursty traffic over a period of 25 slots. Packets are arriving at the stations in bursts. During a burst, the station has exactly one packet arrival at each time slot. During the idle periods, there are no packet arrivals.

Each station is provided with a waiting queue where the packets are temporarily stored while waiting to be transmitted. For all stations, the queue capacity is assumed to be equal to Q packets. When a packet arrives at a station u_i while the waiting queue of this station is full, the packet is discarded. Otherwise, if the queue is not full, it is stored in the queue. It remains there until it is transmitted.

The traffic, which is offered to the stations, is assumed to be bursty [32]. Packets arrive at the stations in long bursts (see Fig. 1). Thus, each station can be in one of two states: active or idle. When a station is active, it has one packet arrival at each time slot. On the other hand, when a station is idle, it has no packet arrivals. The mean burst duration is B slots. In order to avoid collisions, only one station is allowed to transmit at each time slot.

In a bursty traffic environment, the bandwidth demands of stations are asymmetric and time-variable. The main challenge in such an environment, is to share the available bandwidth among the stations according to their needs. The key problem is to determine which station is granted permission to transmit at each time slot. In order to reduce the number of idle slots, stations which have packets to transmit must be granted permission to transmit more frequently than other stations. This is not an easy task, because each station operates independently from the others and has no knowledge of their states.

A second problem is to guarantee that—although there is no centralized coordination between the stations—all the stations arrive at the same conclusion on which station is granted permission to transmit at each time slot.

The proposed LMAC protocol copes with the above problems by using a variation of the PBIL algorithm.

III. THE LMAC PROTOCOL

The LMAC protocol is based on a variation of the PBIL algorithm [29]–[31]. The PBIL algorithm is a combination of competitive learning [12]–[28] and genetic algorithms (GAs)[33], [34]. The PBIL algorithm attempts to explicitly maintain statistics about the search space to decide where to sample next. The object of the algorithm is to create a real valued probability vector, which when sampled reveals high-quality solution vectors with high probability. A probability vector which is denoted by $P = \{P[1], \ldots, P[K]\}$ is maintained, with $P[j] (j = 1, \ldots, K)$ being the probability of obtaining a "1" in the jth position. A number of sample vectors are generated according to the probability vector P. Then, the sample vectors are evaluated and the probability vector is updated toward the best sample vector. The algorithmic description of the PBIL algorithm is presented below [30].

```
Procedure PBIL;
begin
  (* Initialize the probability vector *)
  for $j$:=1 to $K$ do $P[j]$:=0.5;
  repeat
    (* Generate samples according to probabilities $P$*)
    for $i$:=1 to $S$ do
    for $j$:=1 to $K$ do
      if RND[j] < $P[j]$ then $sample\_vector[i, j]$:=1
                else $sample\_vector[i, j]$:=0;
    (* Update Probability Vector *)
    for $i$:=1 to $S$ do
  $evaluation[i]$:=$evaluate(sample\_vector[i])$;
    $best\_vector$:=$find\_best\_vector(sample\_vector, evaluation)$;
    for $j$:=1 to $K$ do $P[j]$:=$P[j]*(1-L)+best\_vector[j]*L$;
  forever;
end;
```

where

S	number of sample vectors generated before update of the probability vector;
L	learning rate (how fast to exploit the search performed);
K	number of bits in a generated sample vector;
$RND[j]$ for $j=1, \ldots, K$	random numbers which are chosen from the (0,1) interval according to the uniform probability distribution.

The proposed LMAC protocol uses a variation of the PBIL algorithm. The number of bits in a vector is taken to be equal to the number of network stations, thus, $K = N$. A sample vector schedule is generated according to the probability vector P. Thus, $S = 1$.

The generated sample represents a transmission schedule. The presence of a "1" in the jth position of the $schedule$ ($schedule[j] = 1$) implies that station u_j is granted permission to transmit during this transmission schedule. On the other hand, the presence of a "0" in the jth position of $schedule$ ($schedule[j] = 0$) implies that station u_j is not granted permission to transmit during this schedule. After all stations u_j with $schedule[j] = 1$ are granted permission to transmit, the probability vector is updated according to the network feedback information. Then, a new sample vector is selected according to the new probability vector, and so on.

The presented algorithm differs from the PBIL algorithm in that the probability updating scheme is not based on a total evaluation of each schedule, but on a separate evaluation for each element of the schedule. If $schedule[j] = 1$ and u_j has no packets to transmit (idle slot), then $P[j]$ decreases. On the other hand, if $schedule[j] = 1$ and u_j has a packet to transmit (successful transmission), then $P[j]$ increases. Let $u(t)$ be the station which is granted permission to transmit at time slot t and $slot(t)$ be the channel status during this time slot. The following probability updating scheme is used [where: $L, a \in (0, 1)$ and $P[j] \in (a, 1)$]:

$$\begin{aligned} &P[j]=P[j]+L(1-P[j]) \text{ if } u(t)=u_j \text{ and } slot(t) <> IDLE \\ &P[j]=P[j]-L(P[j]-a) \text{ if } u(t)=u_j \text{ and } slot(t)=IDLE. \end{aligned} \tag{1}$$

At each station u_j, the above learning algorithm is implemented by means of a learning automaton. Since the offered traffic is bursty, when the selected station u_j has a packet to transmit, it is probable that this station will have packets to transmit in the near future. Therefore, its choice probability $P[j]$ is increased according to (1). On the other hand, when the selected station u_j is idle, it is probable that this station will remain idle in the near future. Therefore, its choice probability $P[j]$ is decreased according to (1).

When the choice probability of a station converges to 0, then this station is not granted permission to transmit for a long period. During this period, it is probable that the station transits from idle to busy state. However, since the station is not granted permission to transmit, the protocol is not capable of "sensing" the transition. The role of parameter a, is to prevent the choice probabilities of the stations from taking values in the neighborhood of 0, in order to increase the adaptivity of the protocol.

All of the stations use the same probability updating scheme and due to the broadcast nature of the network, the network feedback information is common for all the stations. Consequently, all the stations always contain the same choice probabilities. Furthermore, since the same random number generator and the same seed is used by all the stations, it follows that all the stations select the same station which is granted permission to transmit [10]. Therefore, although there is no centralized coordination between the stations, the protocol is collision-free.

The flowchart of the proposed LMAC protocol is given in Fig. 2. Its algorithmic description is presented below.

```
Procedure LMAC;
begin
  $t := t + 1$;
  ($*$ Initialize the probability vector $P$ $*$)
  for $j$:=1 to $N$ do $P[j]$:=0.5;
  repeat
    ($*$ Use vector $P$ to generate a transmission schedule $*$)
    for $j$:=1 to $N$ do
      if $RND[j] < P[j]$ then $schedule[j]$:=1 else $schedule[j]$:=0;
    ($*$ Update the probability vector $P$ $*$);
    for $j$:=1 to $N$ do if $schedule[j] = 1$ then
      begin
        $t$:=$t + 1$;
        $u(t)$:=$u_j$ ($*$ $u_j$ is granted permission to transmit at slot $t$ $*$)
        if $slot(t)$ <> IDLE then $P[j]$:=$P[j]+L*(1-P[j])$
        else $P[j]$:=$P[j] - L*(P[j] - a)$;
      end;
  forever;
end;
```

Let us consider an example in order to clearly demonstrate the operation of the proposed protocol. Assume, for example, that $N = 7$ and $P = \{0.7, 0.2, 0.5, 0.1, 0.4, 0.6, 0.2\}$. The transmission schedule is constructed by selecting seven random numbers $RND = \{RND[1], \ldots, RND[7]\}$ from the (0,1) interval according to the uniform probability distribution and comparing them with the corresponding probabilities P. Assume, for example, that $RND = \{0.1, 0.3, 0.9, 0.2, 0.3, 0.2, 0.5\}$. If $RND[j] < P[j]$ (for $j = 1, \ldots, 7$) then station u_j is included in the current transmission schedule. Otherwise, it is not included in the schedule. In our example, the resulting transmission schedule consists of stations u_1, u_5, and u_6. These stations are sequentially granted permission to transmit

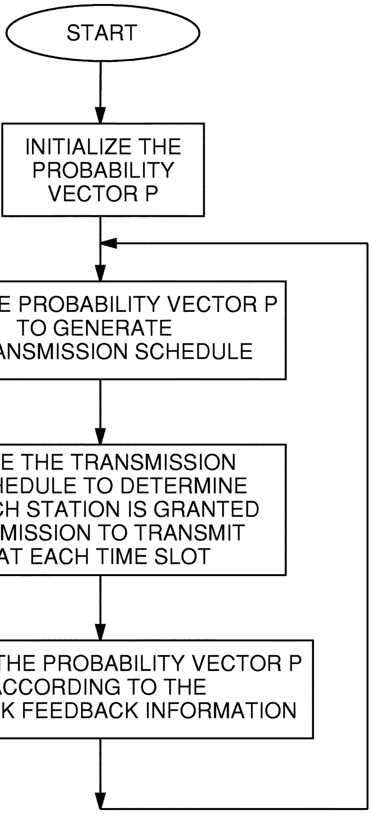


Fig. 2. Flowchart that outlines the operation of the LMAC protocol.

at the next three slots. Then, the probability vector P is updated according the network feedback information (successful transmission or idle slot) during these slots by using the probability updating scheme (1). Then, a new transmission schedule is generated and same procedure is repeatedly executed.

IV. ANALYSIS

It can be proven that under the LMAC protocol each station tends to take a fraction of the available bandwidth, proportional to the probability that this station is not idle. Therefore, the portion of the bandwidth, which is assigned to each station tends to be proportional to the station's needs. In this way, the number of idle slots is minimized and the network performance is improved. The above statement is formally expressed by the following theorem.

Theorem 1: If the probability updating scheme (1) is used and d_i is the probability that station u_i is not idle (for $i = 1, \ldots, N$), then for any station u_i

$$\lim_{t\to\infty,\, L\to 0,\, a\to 0} P[i](t) = d_i.$$

Proof: To prove Theorem 1, we shall use the following theorem, which is presented in [28] and [35].

Theorem 2: Let $x(t)_{t\geq 0}$ be a stationary Markov process dependent on a constant parameter $\theta \in [0, 1]$. Each $x(t) \in I$, where I is a subset (any subset) of the real line. Let $\delta x(t) = x(t+1) - x(t)$. The following are assumed to hold:

1) I is compact.
2) $E[\delta x(t) | x(t) = y] = \theta\, \omega(y) + O(\theta^2)$.
3) $E[|\delta x(t)|^2 \mid x(t) = y] = \theta^2\, b(y) + o(\theta^2)$.
4) $E[|\delta x(t)|^3 \mid x(t) = y] = O(\theta^3)$ where

$$\sup_{y\in I} \frac{O(\theta^k)}{\theta^k} < \infty \text{ for } k = 2, 3 \text{ and } \sup_{y\in I} \frac{o(\theta^2)}{\theta^2} \to 0 \text{ as } \theta \to 0.$$

5) $\omega(y)$ has a Lipschitz derivative in I.
6) $b(y)$ is Lipschitz in I.

If Assumptions 1–6 hold, $\omega(y)$ has a unique root y^* in I and $d\omega/dy|_{y=y^*} < 0$, then

$\text{var}[x(t)|x(0) = x] = O(\theta)$ uniformly for all $x \in I$ and $t \geq 0$.

For any $x \in I$ the differential equation $(dy(\tau))/d\tau = \omega(y(t))$ has a unique solution $y(\tau) = y(\tau, x)$ with $y(0) = x$ and $E[x(t)|x(0) = x] = y(t\theta) + O(\theta)$ uniformly for all $x \in I$ and $t \geq 0$.

$(x(t) - y(t\theta))/\sqrt{\theta}$ has a normal distribution with zero mean and finite variance as $\theta \to 0$ and $t\theta \to \infty$.

Note: $\omega(y)$ and $b(y)$ can be any functions that satisfy the above conditions. A presentation of the proof of Theorem 2 is beyond the scope of the paper.

Proof of Theorem 2: The proof of the above theorem is due to M. F. Norman and can be found in [35].

To apply the above theorem to the proof of Theorem 1, we have to identify

$$\begin{aligned} x(t) &= P[i](t) \\ \theta &= L \\ I &= (a, 1). \end{aligned}$$

We have

$$\begin{aligned} &E\left[\delta P[i](t) | P[i](t) = P[i]\right] \\ &\quad = \frac{P[i]}{\sum_{k=1}^{N} P[k]} \left(d_i L \left(1 - P[i]\right) - \left(1 - d_i\right) L \left(P[i] - a\right)\right) \\ &\quad = L \frac{P[i]}{\sum_{k=1}^{N} P[k]} \left(-P[i] + d_i + a\left(1 - d_i\right)\right) \\ &\quad = L\omega\left(P[i]\right) \end{aligned} \tag{2}$$

$$\begin{aligned} &E\left[|\delta P[i](t)|^2 | P[i](t) = P[i]\right] \\ &\quad = L^2 \frac{P[i]}{\sum_{k=1}^{N} P[k]} \left(d_i \left(1 - P[i]\right)^2 + \left(1 - d_i\right) \left(P[i] - a\right)^2\right) \\ &\quad = L^2 b(P[i]) \end{aligned} \tag{3}$$

$$\begin{aligned} &E\left[|\delta P[i](t)|^3 | P[i](t) = P[i]\right] \\ &\quad = L^3 \frac{P[i]}{\sum_{k=1}^{N} P[k]} \left(d_i \left(1 - P[i]\right)^3 + \left(1 - d_i\right) \left(P[i] - a\right)^3\right) \\ &\quad = O(L^3). \end{aligned} \tag{4}$$

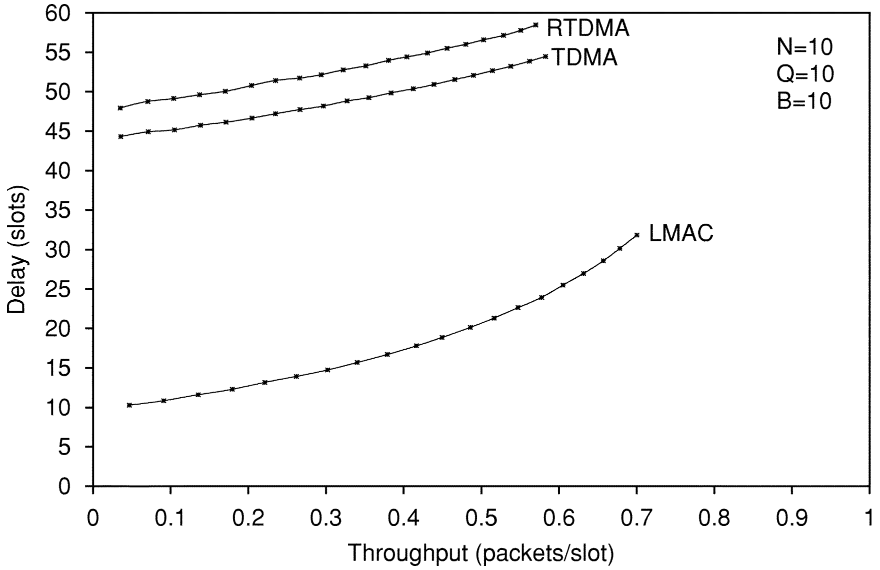


Fig. 3. Delay versus throughput characteristics of LMAC, TDMA, and RTDMA when applied to network N_1.

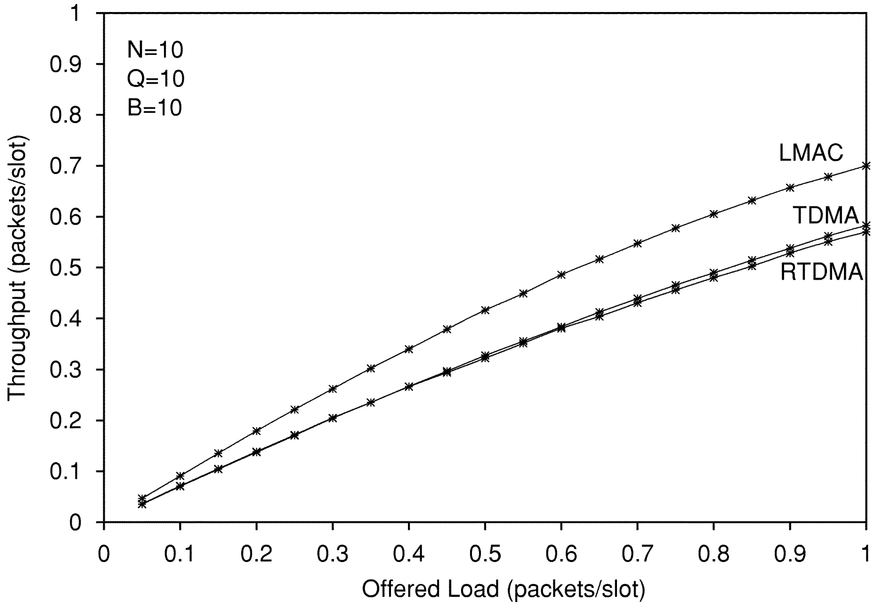


Fig. 4. Throughput versus load characteristics of LMAC, TDMA, and RTDMA when applied to network N_1.

The functions $\omega(P[i])$ and $b(P[i])$ are defined as follows:

$$\omega(P[i]) = \frac{P[i]}{\sum\limits_{k=1}^{N} P[k]}\,(-P[i] + d_i + a\,(1-d_i)) \tag{5}$$

$$b(P[i]) = \frac{P[i]}{\sum\limits_{k=1}^{N} P[k]}\left(d_i\,(1-P[i])^2 + (1-d_i)(P[i]-a)^2\right). \tag{6}$$

It is immediately seen that Assumptions 1–6 are satisfied. It can also be proven that $b(P[i])$ and $\omega'(P[i])$ are Lipschitz in $(a, 1)$ by showing that their first derivatives [$b'(P[i])$ and $\omega''(P[i])$, respectively] are bounded [36] for $P[i] \in (a, 1)$.

It remains to show that $\omega(P[i])$ has a unique root $P[i]^r$ near the point $P[i]^* = d_i$ and that $d\omega(P[i])/dP[i]|_{P[i]=P[i]^r} < 0$. It is immediately seen that $\omega(P[i])$ has a unique root at the point $P[i]^r = d_i + a(1-d_i)$. Since a can be arbitrarily small, it follows that $P[i]^r$ is in the neighborhood of the point $P[i]^* = d_i$. The derivative of $\omega(P[i])$ at this point is

$$\left.\frac{d\omega(P[i])}{dP[i]}\right|_{P[i]=P[i]^r} = \left.\frac{d\left(\frac{P[i]}{\sum\limits_{k=1}^{N} P[k]}\,(-P[i] + d_i + a\,(1-d_i))\right)}{dP[i]}\right| P[i] = P[i]^r$$

$$= -\frac{1}{1 + \frac{\sum\limits_{k=1, k\neq i}^{N} P[k]}{P[i]^r}} < 0. \tag{7}$$

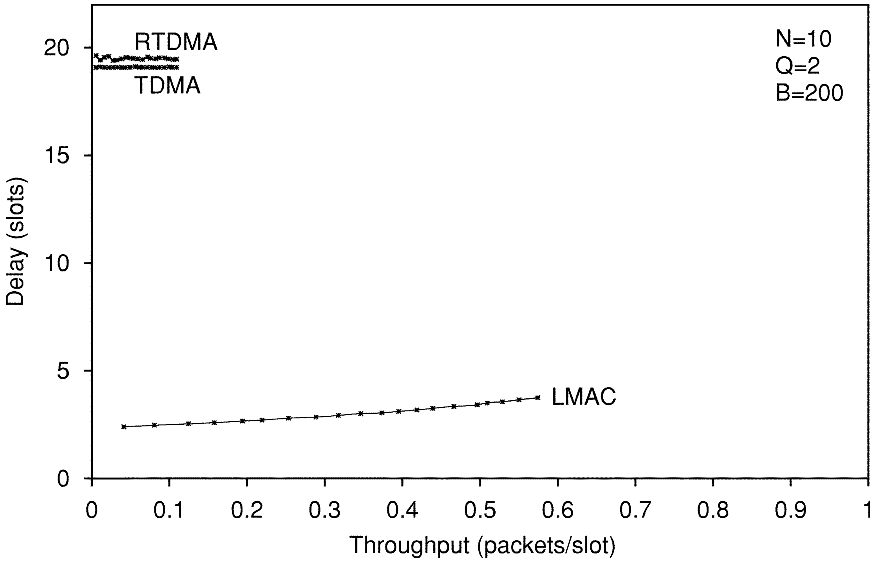


Fig. 5. Delay versus throughput characteristics of LMAC, TDMA, and RTDMA when applied to network N_2.

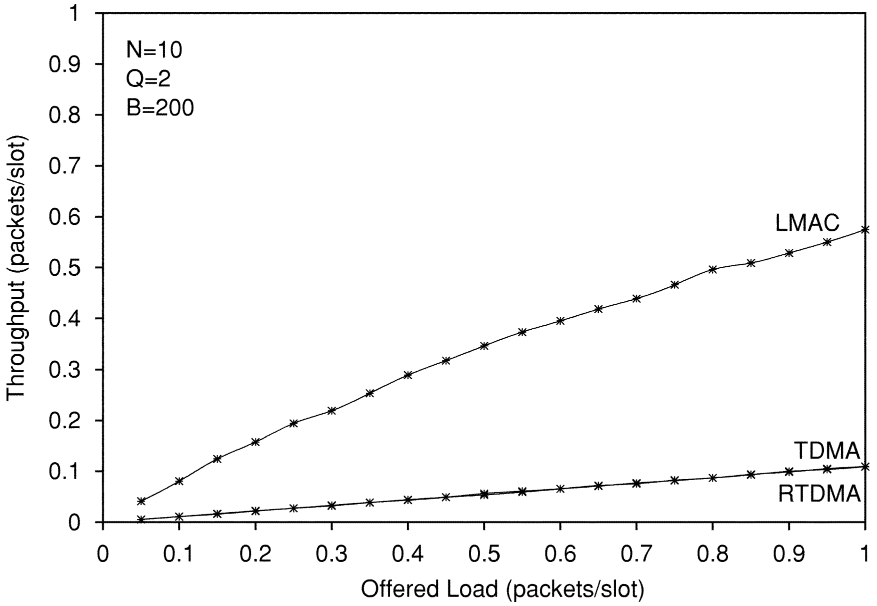


Fig. 6. Throughput versus load characteristics of LMAC, TDMA, and RTDMA when applied to network N_2.

It has been shown that $\omega(P[i])$ has a unique root $P[i]^r$ in the neighborhood of the point $P[i]^* = d_i$ and that the derivative of $\omega(P[i])$ at this point is negative.

If we set $P[i](\tau) = P[i]^r$, the differential equation $(dP[i](\tau))/d\tau = \omega(P[i](\tau))$ is satisfied $(0 = 0)$. Thus, $P[i](\tau) = P[i]^r$ is a solution of the above differential equation. From Theorem 2, it is derived that this solution is unique, thus all the solutions starting in $(a, 1)$ of the differential equation $(dP[i](\tau))/d\tau = \omega(P[i](\tau))$ converge to the point $P[i](\tau) = P[i]^r \simeq P[i]^* = d_i$. According to Theorem 2, we have

$$\lim_{t\to\infty, a\to 0} E\left[P[i](t)\right] = P[i]^* + O(L)$$

and

$$var[P[i](t)] = O(L) \qquad \text{for all } t.$$

Consequently

$$\lim_{t\to\infty, L\to 0, a\to 0} P[i](t) = d_i \qquad \text{q.e.d.} \tag{8}$$

The exact values of a and L depend on the environment where the automata operate. When the environment is slowly switching or when the environmental responses have a high variance, a and L must be very close to zero in order to guarantee a high accuracy. On the other hand, in a rapidly switching environment or when the variance of the environmental responses is low, higher values of a and L can be used, in order to increase the adaptivity of the protocol. Thus, when the burst length is high or the queue length is low, then small values of a and L must be selected. On the other hand, when the burst length is low or when the queue length is high, then a and L can be much higher.

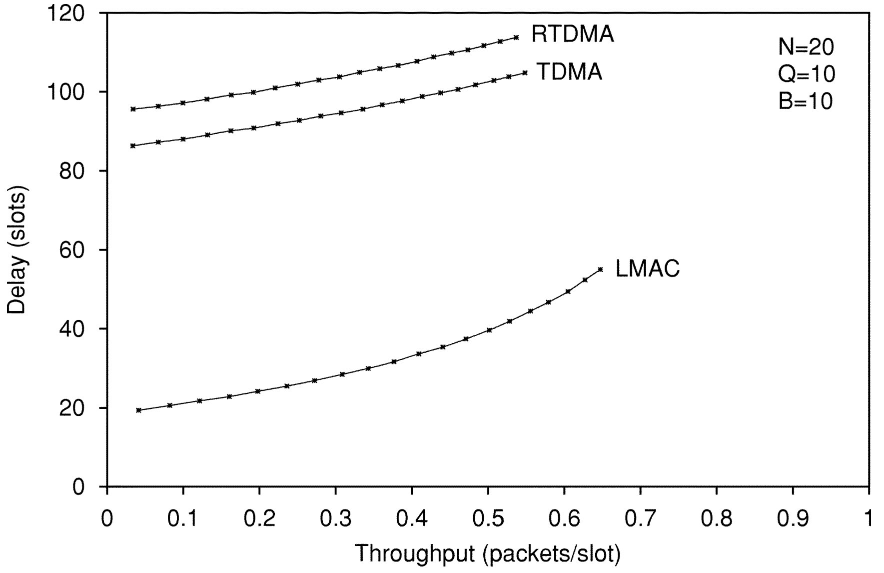


Fig. 7. Delay versus throughput characteristics of LMAC, TDMA, and RTDMA when applied to network N_3.

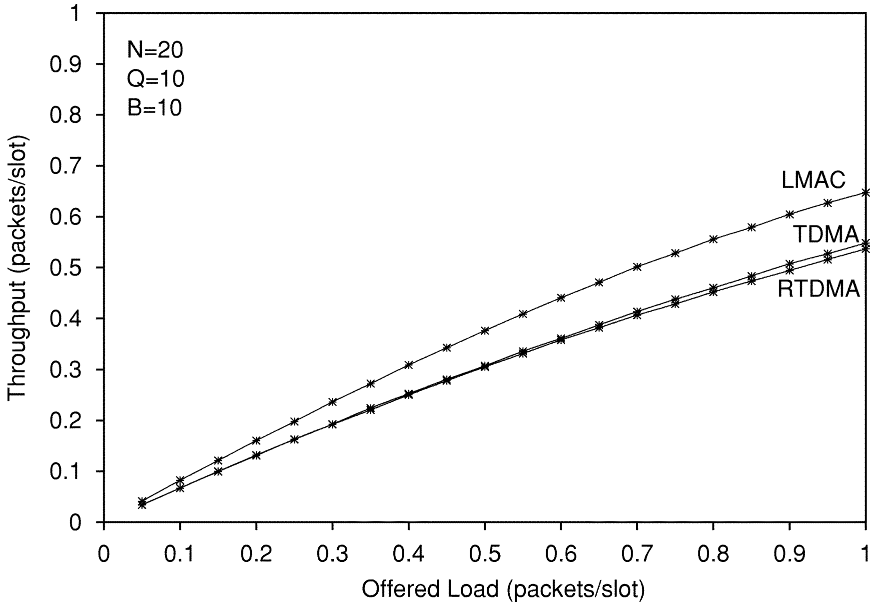


Fig. 8. Throughput versus load characteristics of LMAC, TDMA, and RTDMA when applied to network N_3.

V. SIMULATION RESULTS

In this section, the proposed LMAC protocol is compared to two representatives of TDMA protocols, namely, TDMA [2]–[9] and the random time division multiple access (RTDMA) [10], [11].

The protocols which are under study are compared by simulation using four different networks $(N_1, N_2, N_3, \text{and } N_4)$ and under bursty traffic conditions. The bursty traffic was modeled in the following way (which is identical to the one presented in [32]): Each node can be in one of two states S_0 and S_1. When a node is in state S_0 then it has no packet arrivals. When a node is in state S_1 then at each time slot it has one packet arrival. Given a station is in state S_0 at time slot t, the probability that this station will transit to state S_1 at the next time slot is P_{01}. The transition probability from state S_1 to state S_0 is P_{10}. Clearly, the mean number of time slots that the station spends in state S_1 is given by $1/P_{10} = B$, where B is the mean burst length. The periods that each station stays in states S_0 and S_1 are assumed to be geometrically distributed with means of $1/P_{01}$ and B, respectively. The probability for a station to be in state S_1 is given by $P_{01}/(P_{01} + P_{10})$. All the stations are assumed to have the same load. The total load which is offered to the network is R packets/slot. Therefore, each station has a load of R/N packets/slot and consequently, P_{01} can be calculated as follows:

$$\frac{R}{N} = \frac{P_{01}}{P_{01} + P_{10}} \Leftrightarrow P_{01} = \frac{\frac{R}{N}P_{10}}{1 - \frac{R}{N}} = \frac{R}{B(N-R)}.$$

The number of stations N, the queue size Q, and the mean burst length B, were taken to be as follows:

a) Network N_1: $N = 10$, $Q = 10$, $B = 10$;
b) Network N_2: $N = 10$, $Q = 2$, $B = 200$;
c) Network N_3: $N = 20$, $Q = 10$, $B = 10$;
d) Network N_4: $N = 4$, $Q = 2$, $B = 1000$.

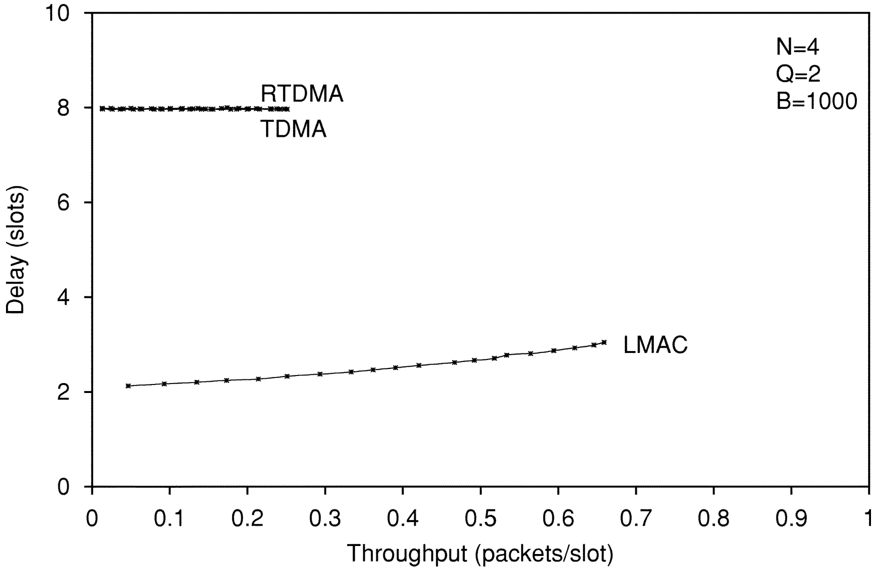


Fig. 9. Delay versus throughput characteristics of LMAC, TDMA, and RTDMA when applied to network N_4.

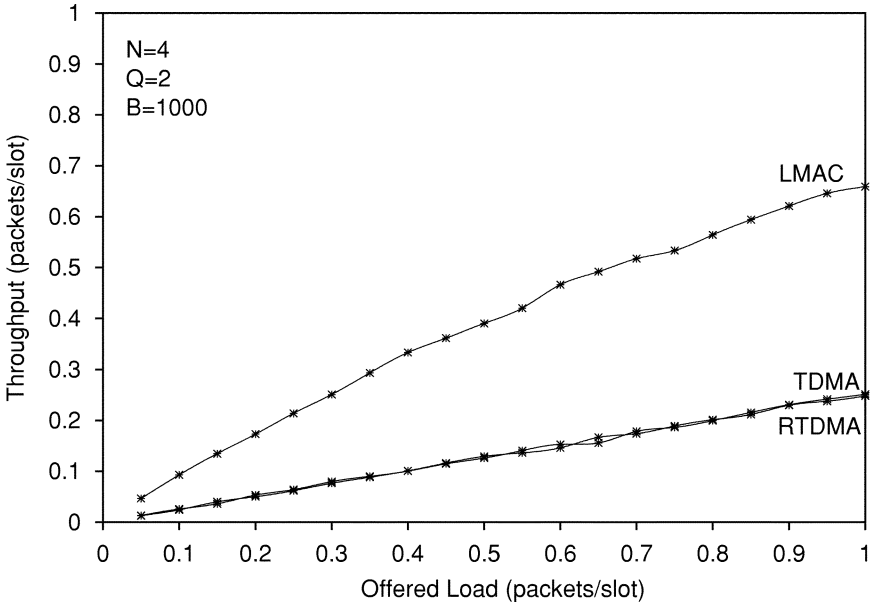


Fig. 10. Throughput versus load characteristics of LMAC, TDMA, and RTDMA when applied to network N_4.

The above values of the number of stations and the queue size are realistic for LANs. The same values have also been used in [37].

We have used the following two broadly used performance metrics in order to compare the three protocols:

1) the delay versus throughput characteristic;
2) the throughput versus offered load characteristic.

The delay versus throughput characteristics of the compared protocols when they are applied to networks N_1, N_2, N_3, and N_4 are depicted in Figs. 3, 5, 7, and 9, respectively. The throughput versus offered load characteristics of the compared protocols when they are applied to networks N_1, N_2, N_3, and N_4 are depicted in Figs. 4, 6, 8, and 10, respectively.

From the above graphs, it becomes clear that LMAC achieves a significantly higher delay-throughput and throughput-load performance than the TDMA and RTDMA protocols, when operating under bursty traffic conditions. The performance improvement which is achieved by the use of LMAC is higher when the offered traffic is more bursty. The proposed LMAC protocol achieves a mean packet delay which is form 80% to 90% lower than the one achieved by TDMA or RTDMA. Furthermore, LMAC achieves a throughput improvement from 20% to 500% in comparison to TDMA or RTDMA.

VI. CONCLUSION

This paper has presented a new medium access control protocol for broadcast networks. According to the proposed LMAC protocol, the station, which is granted permission to transmit at each time slot, is selected by means of LAs, which are capable of being adapted to the changes of the stations' traffic. Therefore, the new protocol is capable of achieving a low delay and a high throughput in the dynamic bursty traffic environment.

The main characteristics of the LMAC protocol are reiterated in the following paragraphs.

It achieves a high performance, even when the offered traffic is bursty. LMAC achieves a throughput improvement from 20% to 500% and a delay decrease from 80% to 90% in comparison to TDMA or RTDMA.

The protocol is self-adaptive and each station tends to take a fraction of the available bandwidth proportional to its needs. Furthermore, when a station goes down for a long period, its choice probability converges to a and consequently, the available bandwidth is shared between the other stations.

No centralized control of the stations is required, since the protocol is fully distributed.

It is fault-tolerant, since its operation is not affected by a possible node failure.

No significant increase in the implementation cost is introduced. The only additional cost, in relation to TDMA or RTDMA, is the cost of the processors that implement the LAs.

The use of LAs offers a new, highly promising approach to the design of self-adaptive multiaccess protocols for communication networks. We are currently working in this direction.

REFERENCES

[1] A. S. Tanenbaum, *Computer Networks*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[2] I. Rubin and J. Baker, "Media access control for high-speed local area and metropolitan area networks," in *Proc. IEEE*, vol. 78, Jan. 1990, pp. 168–203.

[3] W. Stallings, *Data and Computer Communications*, 6th ed. Englewood Cliffs, NJ: Prentice-Hall, 2000.

[4] I. Rubin and Z. Zang, "Message delay analysis of TDMA schemes using slot assignments," *IEEE Trans. Commun.*, vol. 40, pp. 730–737, Apr. 1992.

[5] W. W. Chu and A. G. Konheim, "On the analysis and modeling of a class of computer communication systems," *IEEE Trans. Commun.*, vol. COM-20, pp. 645–660, 1972.

[6] I. Rubin, "Message delays in FDMA and TDMA communication channels," *IEEE Trans. Commun.*, vol. C-27, pp. 769–777, 1979.

[7] ——, "Access control disciplines for multi-access communications channels: Reservation and TDMA schemes," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 516–536, Aug. 1979.

[8] S. S. Lam, "Delay analysis of a time division multiple access (TDMA) channel," *IEEE Trans. Commun.*, vol. COM-25, pp. 1489–1494, 1977.

[9] I. Rubin and L. F. DeMoraes, "Message delay distributions for a TDMA scheme under nonpreemptive priority discipline," *IEEE Trans. Commun.*, vol. COM-32, pp. 583–588, May 1984.

[10] A. Ganz and Z. Koren, "Performance and design evaluation of WDM stars," *J. Lightwave Technol.*, vol. 11, pp. 358–366, Feb. 1993.

[11] A. Ganz, "End-to-end protocols for WDM star networks," in *Protocols for High-Speed Networks*, H. Rudin and R. Williamson, Eds. Amsterdam, The Netherlands: North-Holland, 1989, pp. 219–235.

[12] K. Najim and A. S. Poznyak, *Learning Automata: Theory and Applications*. New York: Pergamon, 1994.

[13] K. S. Narendra and M. A. L. Thathachar, *Learning Automata: An Introduction*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[14] M. A. L. Thathatchar and P. S. Sastry, "Varieties of learning automata: An overview," *IEEE Trans. Syst., Man, Cybern. B*, vol. 32, pp. 711–722, Dec. 2002.

[15] B. J. Oommen and M. Agache, "Continuous and discretized pursuit learning schemes: Various algorithms and their comparison," *IEEE Trans. Syst., Man, Cybern. B*, vol. 31, pp. 277–287, June 2001.

[16] B. J. Oommen and J. K. Lanctot, "Discretized pursuit learning automata," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 931–938, Aug. 1990.

[17] B. J. Oommen and J. P. R. Christensen, "Epsilon optimal discretized linear reward-penalty learning automata," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, pp. 451–458, June 1988.

[18] G. I. Papadimitriou, "Hierarchical discretized pursuit nonlinear learning automata with rapid convergence and high accuracy," *IEEE Trans. Knowledge Data Eng.*, vol. 6, pp. 654–659, Aug. 1994.

[19] ——, "A new approach to the design of reinforcement schemes for learning automata: Stochastic estimator learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 6, pp. 649–654, Aug. 1994.

[20] A. A. Economides, "Multiple response learning automata," *IEEE Trans. Syst., Man, Cybern. B*, vol. 26, pp. 153–156, Feb. 1996.

[21] A. A. Economides, P. A. Ioannou, and J. A. Silvester, "Decentralized adaptive routing for virtual circuit networks using stochastic learning automata," in *Proc. IEEE INFOCOM*, 1988, pp. 613–622.

[22] S. Lakshmivarahan and M. A. L. Thathachar, "Absolutely expedient learning algorithms for stochastic automata," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 281–286, May 1973.

[23] M. A. L. Thathachar and P. S. Sastry, "A class of rapidly converging algorithms for learning automata," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 168–175, Feb. 1985.

[24] K. Najim and A. S. Poznyak, *Learning Automata: Theory and Applications*. New York: Pergamon, 1994.

[25] G. I. Papadimitriou and D. G. Maritsas, "Self-adaptive random access protocols for WDM passive star networks," *Proc. Inst. Elect. Eng., Comput. Dig. Tech.*, vol. 142, pp. 306–312, July 1995.

[26] ——, "Learning automata-based receiver conflict avoidance algorithms for WDM broadcast-and-select star networks," *IEEE/ACM Trans. Networking*, vol. 4, pp. 407–412, June 1996.

[27] G. I. Papadimitriou and A. S. Pomportsis, "Learning-automata-based TDMA protocols for broadcast communication systems," *IEEE Commun. Lett.*, vol. 4, pp. 107–109, Mar. 2000.

[28] K. S. Narendra and M. A. L. Thathachar, "On the behavior of a learning automaton in a changing environment with application to telephone traffic routing," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-10, pp. 262–269, May 1980.

[29] S. Baluja, "Population-based incremental learning: A method for integrating genetic search-based function optimization and competitive learning," Sch. Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-CS-94-163, 1994.

[30] ——, "Genetic algorithms and explicit search statistics," in *Advances in Neural Information Processing Systems 9*, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds. Cambridge, MA: MIT Press, 1997, pp. 319–325.

[31] C. Gonzalez, J. A. Lozano, and P. Larranaga, "The convergence behavior of PBIL algorithm: A preliminar approach," Univ. Basque Country, Spain, Tech. Rep. EHU-KZAA-IK-3/99.

[32] S. L. Danielsen, C. Joergensen, B. Mikkelsen, and K. E. Stubkjaer, "Analysis of a WDM packet switch with improved performance under bursty traffic conditions due to tuneable wavelength converters," *J. Lightwave Technol.*, vol. 16, pp. 729–735, May 1998.

[33] H. Dawid, *Adaptive Learning by Genetic Algorithms: Analytical Results and Applications to Economic Models*. New York: Springer-Verlag, 1999.

[34] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press, 1998.

[35] M. F. Norman, *Markovian Learning Processes*: SIAM Review, 1974, vol. 16, pp. 143–162.

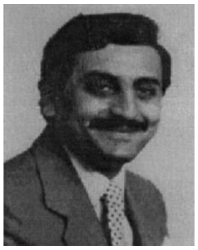
[36] S. M. Nikolsky, *A Course of Mathematical Analysis*. Moscow, USSR: MIR, 1977.

[37] B. Mukherjee and J. S. Meditch, "The p_i-persistent protocol for unidirectional broadcast bus networks," *IEEE Trans. Commun.*, vol. 36, pp. 1277–1286, Dec. 1988.

Georgios I. Papadimitriou (M'89–SM'02) received the Diploma and Ph.D. degrees in computer engineering from the University of Patras, Patras, Greece, in 1989 and 1994, respectively.

From 1989 to 1994, he was a Teaching Assistant at the Department of Computer Engineering of the University of Patras and a Research Scientist at the Computer Technology Institute, Patras. From 1994 to 1996, he was a Postdoctorate Research Associate at the Computer Technology Institute. From 1997 to 2001, he was a Lecturer at the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece. Since 2001, he has been an Assistant Professor at the Department of Informatics, Aristotle University of Thessaloniki. His research interests include design and analysis of broadband networks and learning automata.

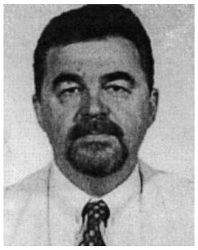
Prof. Papadimitriou is a Member of the Editorial Board of the *International Journal of Communication Systems*. He is also an Associate Editor of the journal *Simulation: Transactions of the Society for Modeling and Simulation International*. He is the coauthor of the books *Wireless Networks*, Wiley, 2003 and *Multiwavelength Optical LANs*, Wiley, 2003. He is also co-editor of the book *Applied System Simulation: Methodologies and Applications* (Norwell, MA: Kluwer) to appear in March 2003. He is the author of more than 70 refereed journal and conference papers.



Mohammad S. Obaidat (S'85–M'86–SM'91) was born in Irbid, Jordan. He received the B.S. degree in electrical engineering from Aleppo University, Aleppo, Syria, and the M.S. and Ph.D. degrees in computer engineering with a minor in computer science from The Ohio State University, Columbus.

He is an internationally well-known Academic/Researcher/ Scientist. He is currently a tenured Full Professor of Computer Science at Monmouth University, West Long Branch, NJ. Among his previous positions are Chair of the Department of Computer Science and Director of the Graduate Program at Monmouth University and a Faculty Member at the City University of New York. He has received extensive research funding and has published over 240 refereed technical articles in refereed scholarly journals and proceedings of refereed international conferences. He is the author of the book *Performance Evaluation of Computer and Telecommunications Systems* (Wiley), to appear in 2003. He is the coauthor of the books *Wireless Networks*, Wiley, 2003 and *Multiwavelength Optical LANs*, Wiley, 2003. He is also the co-editor of the book *Applied System Simulation: Methodologies and Applications* (Norwell, MA: Kluwer), to appear in March 2003. His research interests include performance evaluation of computer systems, algorithms and networks, telecommunications and networking systems, wireless communications and networks, high-performance and parallel computing/computers, applied neural networks and pattern recognition, computer security, and speech processing. He has served as a Consultant for several corporations worldwide.

Dr. Obaidat is the Chief Editor of the *International Journal of Communication Systems* published by Wiley. He is also a Technical Editor of *Simulation: Transactions of the Society for Modeling and Simulations (SCS) International*. He is an Associate Editor/Editorial Board Member of seven other refereed scholarly journals, including three IEEE Transactions. He has guest edited four special issues of the journal *Simulation* journal. He also guest edited a special issue on Neural Network Applications of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS. He also guest edited special issues for the *Computer Communications Journal* and the *Journal of Computers Electrical Engineering*. Currently, he is guest editing special issues for the *Transactions of the Society for Computer Simulation*, *Elsevier Computer Communications Journal*, and the the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS PART B. He has served as the Steering Committee Chair and the Program Chair of the 1995 IEEE International Conference on Electronics, Circuits, and Systems (ICECS'95), and as the Program Chair of the 1996 IEEE International Phoenix Conference on Computers and Communications (IPCCC"96). He served as the Vice General Chair of the 1997 IEEE International Performance, Computing, and Communications Conference and the General and Program Chair of the 1997 Summer Computer Simulation (SCSC) Conference. He also served as the General Chair of SCSC'98 and SCSC'99. He also served as Vice Program Chair for the 1997 IEEE International Conference on Computer Communications and Networks (IC^{3}N'97). He served as the General Co-Chair of the 1998 IEEE International Performance, Computing, and Communications Conference. He is the founder of the SCS International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS) and has served as the General Chair of SPECTS'98, SPECTS'99, SPECTS'01, and SPECTS'02. He will be the General Chair of SPECTS'2003 and steering committee chair of SCSC2003. He has served as the Steering Committee Chair of SCSC2002, SCSC2001, and SCSC2002. He served as the Vice Program Chair of the 2001 International Conference on Parallel Processing (ICPP'01). He has served on the program, advisory, organizing, and steering committees of numerous international conferences. He is the Chair of the International Advisory Committee of ICECS2003 Conference. He has received a recognition certificate from the IEEE. Between 1994 and 1997, he has served as Distinguished Speaker/Visitor of the IEEE Computer Society. Since 1995, he has been serving as an ACM Distinguished Lecturer. Between 1996 and 1999, he served as an IEEE/ACM Program Evaluator of the Computing Sciences Accreditation Board/Commission, CSAB/CSAC. He is the founder and first Chairman of SCS Technical Chapter (Committee) on PECTS (Performance Evaluation of Computer and Telecommunication Systems). He has been a member of the Board of Directors of the Society for Modeling and Simulation International, SCS, since 1994. He is now the Vice President of Conferences of the Society for Modeling and Simulation International. He has been invited to lecture and give keynote speeches worldwide. Prof. Obaidat has been elected to be a Fellow of the Society for Modeling and Simulation International. He is the Scientific Advisor of the 2002 UNDP/World Bank Workshop on Fostering Digital Inclusion-The Role of ICT in Development that is part of the MDF-4. Recently, he has been awarded a Nokia Research Fellowship.



Andreas S. Pomportsis received the B.S. degree in physics in 1978 and the M.S. degree in electronics and communications in 1981, both from the University of Thessaloniki, Thessaloniki, Greece, the Dipl. degree in electrical engineering from the Technical University of Thessaloniki in 1985, and the Ph.D. degree in computer science from the University of Thessaloniki In 1987.

Currently, he is a Professor at the Department of Informatics, Aristotle University of Thessaloniki. His research interests include computer networks, learning automata, computer architecture, parallel and distributed computer systems, and multimedia systems.