10th Annual
Conference of Computer Society of Iran
February 15-17, 2005

Asymptotic Behavior of Learning Automata Operating in State Dependent Nonstationary Environments

Hamid Beigy
Computer Engineering Department, Sharif University of Technology, Tehran, Iran
Institute for Studies in Theoretical Physics and Mathematics (IPM), School of Computer Science, Tehran, Iran
beigy@ce.sharif.edu

M. R. Meybodi
Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran
meybodi@ce.aut.ac.ir

Abstract. In this paper, we introduce a new state dependent nonstationary environment and study the asymptotic behavior of SL_{R_I} learning algorithm operating under the proposed environment. It is shown that the SL_{R-I} automaton operating in the proposed nonstationary environment, equalizes the expected penalty strengths of actions. This model was motivated by applications of learning automata in call admission in cellular networks.

1 Introduction

A learning automaton interacts with a random environment and attempts to determine the best action that the environment offers. The learning in learning automata is achieved by interacting with the environment and processing its response (reinforcement signal) to the actions that are chosen. The learning process of an automaton can be described as follows: the automaton is offered a set of actions by the environment with which interacts and it is constrained to chose one of these actions. When an action is chosen, the automaton is rewarded or penalized with certain probability. A learning automaton is one that learns the optimal action, which has the minimum penalty probability.

Learning automata can be classified into two main families: *fixed structure learning automata* and *variable structure learning automata* [1]. Variable structure learning automata are represented by a triple $< \underline{\beta}, \underline{\alpha}, T >$, where $\underline{\beta}$ is a set of inputs, $\underline{\alpha} = \{\alpha_1, \ldots, \alpha_2\}$ is a set of actions, and T is learning algorithm. The learning algorithm is a recurrence relation and is used to modify action probabilities ($\underline{p}$) of the automaton. Various learning algorithms have been reported in the literature. Let α_i be the action chosen at time n as a sample realization from probability distribution $\underline{p}(k)$. The SL_{R-I} learning algorithm updates its action probabilities according to the following equations.

$$p_j(n+1) = \begin{cases} p_j(n) - a[1-\beta(n)]p_j(n) & j \neq i \\ \\ p_j(n) + a[1-\beta(n)]\sum_{k \neq i} p_k(n) & j = i \end{cases} \quad (1)$$

where a the learning parameter.

A random environment is defined as a 3-tuple $< \underline{\alpha}, \underline{\beta}, \underline{S} >$, where $\underline{\alpha}$ and $\underline{\beta}$ are the set of actions and the set of reinforcement signals, respectively. The set $\underline{S} = \{s_1, \ldots, s_r\}$ is the set of penalty strengths (penalty probabilities), where $s_i(n) = \text{Prob}\,[\beta(n)|\alpha(n) = \alpha_i]$. The random environments can be classified in various ways depending on the nature of the vectors $\underline{S}$ and vector $\underline{\beta}$. According to the nature of the set $\underline{S}$, the random environment could be classified into stationary and nonstationary environments. If the set of penalty strengths $\underline{S}$ is not varied with the time, the environment is called *stationary*; otherwise it is called *nonstationary*.

Based on the nature of $\underline{\beta}$, environments could be classified in three classes: P-, Q-, and S-models. The output of a P-model environment has two elements of *success* or *failure*. In Q-model environments, set

$\underline{\beta}$ has a finite number of values in the interval $[0, 1]$ while in S-model environments, set $\underline{\beta}$ has infinite number of values in the interval $[0, 1]$.

Most of researches on learning deals with the problem of designing a learning automaton operating in stationary environments. However, in many applications such ac computer nctworks [2–6], it is quite reasonable for us to assume that the penalty probabilities are stationary and the penalty probabilities are time varying. Therefore, the performance of learning automata should be judged in such context. Expediency of a learning automaton with fixed strategy may decrease in nonstationary environments and it may become non-expedient. The learning process must have sufficient flexibility to track the better action. The success of the learning procedure depends on the changes in the environment and information about environment that is collected by learning automata.

In this paper, we propose a new state dependent nonstationary environment and study the asymptotic behavior of SL_{R_I} learning algorithm operating under the proposed environment. It is shown that the SL_{R-I} automaton operating in the proposed nonstationary environment, equalizes the expected penalty strengths of actions. This model was motivated by applications of learning automata in call admission in cellular networks [7, 8].

The rest of this paper is organized as follows. Section 2 reviews the nonstationary environments. Section 3 gives the proposed nonstationary environment and section 4 concludes the paper.

2 Nonstationary Environments

The nonstationary environments can be divided into three groups: *periodic environments*, *Markovian switching environments*, and *state dependent nonstationary environments*, which will be explained briefly in the next three subsections. These models are proposed when a finite action-set learning automaton operates under these environments .

2.1 Periodic Environments

A periodic nonstationary environment consists of d stationary environments $E_1, E_2, \cdots, E_d$. The environment is in one of $E_1, E_2, \cdots, E_d$ environments in periodic manner with unknown period T [9, 10]. No analytical results have been investigated for the periodic environments.

2.2 Markovian Switching Environments

In Markovian switching environments (MSE), it is assumed that the nonstationary environment is composed of d stationary environments $E_1, E_2, \cdots, E_d$, which are states of a Markov chain. This chain is ergodic and described by a $d \times d$ state transition matrix, T. The nonstationary environment is switched from environment E_i to E_j with probability T_{ij}. The Markovian switching environments can be classified into the following six models.

Model A This model has state transition probability $T_{ij} = \delta$ for all $i \neq j$, where $\delta(\delta < 1/d)$ represents the average frequency of switching [9]. A small value of δ implies a slowly varying environment. The performance of some fixed and variable structure learning automata operating in Markovian switching environment has been analyzed by Tsetline [9] and Varshavski and Vorontsova [1], respectively. For fixed structure learning automata, it is shown that optimal memory decreases when the switching rate increases [9].

Model B This model is called *source oriented model* and has been proposed by Oommen and Masum, where the probability of switching from E_i to E_j depends on the source state E_i [11]. If switching probabilities are equal to δ, this model reduces to model A.

Model C In this model, which is called *destination oriented model*, the probability of switching from E_i to E_j depends on the destination state E_j [11].

Model D In this model it is assumed that the switching matrix itself is time varying [11]. This model is called *time varying switching environment*. At instant n, the probability of switching from E_i to E_j is specified by $T_{ij}(n)$. The analysis of learning automaton operating in this model of environments with general forms of T seems to be intractable. The analysis of a special case in which $T_{ij}(n)$ decreases as n increases is given in [11].

Model E In this model, it is assumed that the environment switches at instant n according to a transition probability which depends on the input received by the automaton at that instant, $\beta(n)$ [12]. If we assume that $\beta(n) \in \{0, 1\}$, there are two $d \times d$ transition matrices $T^{(0)}$ and $T^{(1)}$ corresponding to the $\beta(n) = 0$ and $\beta(n) = 1$, respectively.

Model F This model is special case of model E in which environment may switch only if reward is received by the automaton or equivalently $T^{(1)} = I_{d\times d}$ [13]. This model arises in the course of using stochastic automata in random search algorithms.

2.3 State Dependent Nonstationary Environments

In state dependent nonstationary environments, actions of automaton affect the characteristic of the response from environment. This model was motivated by applications of learning automata in telephone and data network traffic routing [2–4, 14]. The state dependent nonstationary environments could be classified into the following three models.

Model A In this model, penalty strength $s_i(n)$ increases if action α_i is selected and decreases if action α_j (for $i \neq j$) is selected [2].

Model B A mathematically more tractable model assumes the penalty strengths are function of action probability vector [14, 3]. This allows the environment and automaton to be described by a Markov process whose equilibrium behavior can be analyzed.

Model C This model describes the transient behavior of state dependent model B [4]. In this model, the penalty strengths are functions of previous penalty strengths and current action probability vector of automaton.

3 The Proposed Nonstationary Environment

In this section, we first give a mathematical model for the nonstationary environment under which the SL_{R-I} learning automaton operates and then study the behavior of the automaton operating in the modeled environment. This model was motivated by applications of learning automata in call admission in cellular networks [7, 8]. For the sake of simplicity, we use an automaton, with two actions α_1 and α_2 with probability vector $\underline{p}(n) = [p_1(n), p_2(n)]^t$ at any stage n.

The nonstationary environment at stage n can be completely described by input set $\underline{\alpha} = \{\alpha_1, \alpha_2\}$, a continuous output set $\underline{\beta} = \{[0, 1]\}$, and penalty strengths $s_1(n)$ and $s_2(n)$, where smaller value of β means more favorable output. The penalty strengths $s_1(n)$ and $s_2(n)$ when the automaton chooses action α_i (for $i = 1, 2$) is changed according to

$$s_j(n+1) = \begin{cases} s_j(n) + \theta'_{ij}(n) & \text{with probability of } q_i(n) \\ s_j(n) + \phi'_{ij}(n) & \text{with probability of } 1 - q_i(n), \end{cases} \tag{2}$$

where $\theta'_{ij}(n)$ and $\phi'_{ij}(n)$ is the amount of changes in $s_j(n)$ (for $j = 1, 2$) when the automaton chooses action α_i (for $i = 1, 2$) at stage n. In general, $\theta'_{ij}(n)$, $\phi'_{ij}(n)$, and $q_i(n)$ (for $i, j = 1, 2$) can be functions of $p_1(n)$, $s_1(n)$, and $s_2(n)$, but to keep the model both simple and tractable we assume that $\theta'_{ij}(n)$ and $\phi'_{ij}(n)$ are constants θ'_{ij} and ϕ'_{ij}, and $q_i(n)$ is a constant q_i. Since $s_1(n)$ and $s_2(n)$in (2) are in the interval $[0, 1]$, we need to have the following assumption.

Assumption 1 *$\theta'_{ij}(n)$ and $\phi'_{ij}(n)$ are θ'_{ij} and ϕ'_{ij}, respectively unless they lie outside of the interval $[0, 1]$. That is*

$$\theta'_{ij}(n) = \begin{cases} \theta'_{ij} & \text{if } 0 \leq s_j(n) + \theta'_{ij} \leq 1 \\ s_j(n) & \text{if } s_j(n) + \theta'_{ij} < 0 \\ 1 - s_j(n) & \text{if } s_j(n) + \theta'_{ij} > 1. \end{cases} \tag{3}$$

Similarly,

$$\phi'_{ij}(n) = \begin{cases} \phi'_{ij} & \text{if } 0 \leq s_j(n) + \phi'_{ij} \leq 1 \\ s_j(n) & \text{if } s_j(n) + \phi'_{ij} < 0 \\ 1 - s_j(n) & \text{if } s_j(n) + \phi'_{ij} > 1. \end{cases} \tag{4}$$

By the above assumptions, the changes in probability strengths, $\theta'_{ij}(n)$ or $\phi'_{ij}(n)$ (for $i,j=1,2$), are constant except when the penalty strengths given by equation (2) lie outside of the interval $[0,1]$.

In the following two lemmas, we compute $s_1(n)$ and $s_2(n)$. Let $\tilde{p}_1(n)=\{p_1(0),p_1(1),\ldots,p_1(n-1)\}$. It is assumed that $s_1(n)$ and $s_2(n)$ are in the interval $[0,1]$, i.e. $\theta_{ij}(n)$ and $\phi_{ij}(n)$ (for $i,j=1,2$) are constant. The resulting analysis is approximate in the sense that it is not valid when $s_1(n)$ and $s_2(n)$ are close to zero or one.

Lemma 1. *Let $\theta_{ij}(n)=q_i(n)\theta'_{ij}$ and $\phi_{ij}(n)=(1-q_i(n))\phi'_{ij}(n)$. The expected value of penalty strength $s_2(n)$ and $s_1(n)$ are equal to*

$$\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right]=s_2(0)+\left[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})\right]\sum_{i=0}^{n-1}p_1(i)+n\left[\theta_{22}+\phi_{22}\right] \tag{5}$$

$$\mathrm{E}\left[s_1(n)\,|\tilde{p}_1(n)\right]=s_1(0)+\left[(\theta_{11}+\phi_{11})-(\theta_{21}+\phi_{21})\right]\sum_{i=0}^{n-1}p_1(i)+n\left[\theta_{21}+\phi_{21}\right]. \tag{6}$$

Proof: Computing the expectations of $s_2(n)$ on the sequence of action probabilities

$$\tilde{p}_1(n)=\{p_1(0),p_1(1),\ldots,p_1(n-1)\},$$

we obtain

$$\begin{aligned}\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right]&=\mathrm{E}\left[s_2(n-1)\,|\tilde{p}_1(n)\right]+p_1(n-1)q_1(n-1)\theta'_{12}(n-1)+(1-p_1(n-1))q_2(n-1)\theta'_{22}(n-1)\\&\quad+p_1(n-1)(1-q_1(n-1))\phi'_{12}(n-1)+(1-p_1(n-1))(1-q_2(n-1))\phi'_{22}(n-1).\end{aligned} \tag{7}$$

Suppose that $q_i(n)=q_i$ (for $n\geq 0$) and using assumption 1, the above equation becomes

$$\begin{aligned}\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right]&=\mathrm{E}\left[s_2(n-1)\,|\tilde{p}_1(n)\right]+p_1(n-1)\theta_{12}+(1-p_1(n-1))\theta_{22}\\&\quad-p_1(n-1)\phi_{12}+(1-p_1(n-1))\phi_{22}\end{aligned} \tag{8}$$

$$\begin{aligned}&=p_1(n-1)\left[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})\right]+\left[\theta_{22}+\phi_{22}\right]+\mathrm{E}\left[s_2(n-1)\,|\tilde{p}_1(n)\right]\\&=p_1(n-1)\left[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})\right]\\&\quad+p_1(n-2)\left[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})\right]+2\left[\theta_{22}+\phi_{22}\right]+\mathrm{E}\left[s_2(n-2)\,|\tilde{p}_1(n)\right]\\&=s_2(0)+\left[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})\right]\sum_{i=0}^{n-1}p_1(i)+n\left[\theta_{22}+\phi_{22}\right].\end{aligned} \tag{9}$$

The expected value of penalty strength $s_1(n)$ is obtained in similar way. ■

Now, we are ready to study the behavior of the SL_{R-I} learning algorithm in the given nonstationary environment. In the automaton–environment connection, $(p_1(n),s_1(n),s_2(n))$ describe a discrete time continuous state Markov process with $(0,1,0)$ and $(1,0,1)$ as the absorbing states. Computing the expectations on the sequence of action probabilities $\tilde{p}_1(n)=\{p_1(0),p_1(1),\ldots,p_1(n-1)\}$, we obtain

$$\begin{aligned}\Delta p_1(n)&=\mathrm{E}\left[p_1(n+1)-p_1(n)\,|\tilde{p}_1(n)\right]\\&=a[1-p_1(n)]p_1(n)\mathrm{E}\left[1-s_1(n)\,|\tilde{p}_1(n)\right]-a[1-p_1(n)]p_1(n)\mathrm{E}\left[1-s_2(n)\,|\tilde{p}_1(n)\right],\\&=a[1-p_1(n)]p_1(n)\mathrm{E}\left[s_2(n)-s_1(n)\,|\tilde{p}_1(n)\right].\end{aligned} \tag{10}$$

We are interested in studying the equilibrium points of equation (10). The equilibrium points of equation (10) are those points that satisfy the equation $\Delta p_1(n)=0$, where the expected changes in the probability is zero. The equilibrium points of equation (10) are

$$\begin{aligned}&p_1(n)\to 0,\\&p_1(n)\to 1,\\&\mathrm{E}\left[s_2(n)-s_1(n)\,|\tilde{p}_1(n)\right]\to 0.\end{aligned} \tag{11}$$

We assume that equation (10) converges to one of its equilibrium points. Two equilibrium points $p_1(n)=0$ and $p_1(n)=1$ correspond to the absorbing states as in a conventional SL_{R-I} learning algorithm operating in a stationary environment and $p_1(n)$ converges to a constant, i.e. zero or one. The third equilibrium point, $\mathrm{E}\left[s_2(n)-s_1(n)\,|\tilde{p}_1(n)\right]\to 0$, represents an entirely different kind of behavior and convergence of $p_1(n)$ depends on the evolution of the penalty strengths, which are studied in the following lemma.

Lemma 2. *When $p_1(n)$ converges in the sense that* $\mathrm{E}\left[s_2(n) - s_1(n)\,|\tilde{p}_1(n)\right] \to 0$, *then*

$$\frac{1}{n}\sum_{i=0}^{n-1} p_1(i) \to \frac{[(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})]}{[(\theta_{11}+\phi_{11})+(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})-(\theta_{12}+\phi_{12})]}. \tag{12}$$

Proof: Substituting (5) and (6) in $\mathrm{E}\left[s_2(n) - s_1(n)\,|\tilde{p}_1(n)\right] \to 0$, we obtain

$$\mathrm{E}\left[s_2(n) - s_1(n)\,|\tilde{p}_1(n)\right] = [s_2(0) - s_1(0)] + n\left[(\theta_{22}+\phi_{22}) - (\theta_{21}+\phi_{21})\right] + \left[(\theta_{12}+\phi_{12}) - (\theta_{22}+\phi_{22}) - (\theta_{11}+\phi_{11}) + (\theta_{21}+\phi_{21})\right]\sum_{i=0}^{n-1} p_1(i). \tag{13}$$

Since $\mathrm{E}\left[s_2(n) - s_1(n)\,|\tilde{p}_1(n)\right] \to 0$, the above equation implies that

$$\frac{1}{n}\sum_{i=0}^{n-1} p_1(i) \to \frac{[(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})]}{[(\theta_{11}+\phi_{11})+(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})-(\theta_{12}+\phi_{12})]}. \tag{14}$$

■

Substituting equation (12) in (5) and simplifying we get

$$\begin{aligned}\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right] &= s_2(0) + n\frac{[(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})]\,[(\theta_{12}+\phi_{12})-(\theta_{22}+\phi_{22})]}{(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})+(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})} + n\left[\theta_{22}+\phi_{22}\right] \\ &= s_2(0) + n\frac{(\theta_{22}+\phi_{22})(\theta_{11}+\phi_{11})-(\theta_{21}+\phi_{21})(\theta_{12}+\phi_{12})}{(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})+(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})}.\end{aligned} \tag{15}$$

Equation (15) must be used with caution. Its validity holds only when $\theta_{ij}(n)$ and $\phi_{ij}(n)$ are constants and consequently when $s_2(n)$ is not in the vicinity of zero or one. However, (15) gives the valuable information that $\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right]$ decreases towards zero when the second and third terms are negative and increases towards one when the second and third terms are positive. A similar analysis for $s_1(n)$ yields

$$\begin{aligned}\mathrm{E}\left[s_1(n)\,|\tilde{p}_1(n)\right] &= s_1(0) + n\frac{[(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})]\,[(\theta_{11}+\phi_{11})-(\theta_{21}+\phi_{21})]}{(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})+(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})} + n\left[\theta_{21}+\phi_{21}\right] \\ &= s_1(0) + n\frac{(\theta_{11}+\phi_{11})(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})(\theta_{12}+\phi_{12})}{(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})+(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})}.\end{aligned} \tag{16}$$

and hence $\mathrm{E}\left[s_1(n)\,|\tilde{p}_1(n)\right]$ behaves in the same way as $\mathrm{E}\left[s_2(n)\,|\tilde{p}_1(n)\right]$.

Theorem 1. *The SL_{R-I} automaton operating in nonstationary environments given by equation (2) equalizes the expected penalty strengths of two actions.*

Proof: It is seen from (12) that the limit to which $\frac{1}{n}\sum_{i=0}^{n-1} p_1(i)$ converges depends only on the changes in the penalty strengths. In (12), $(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})$ corresponds to the difference between changes in s_2 and s_1 when action α_2 is performed and $(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})$ corresponds to the difference between changes in s_1 and s_2 when action α_1 is performed. From (12), it is clear that the ratio in which the two actions α_1 and α_2 are chosen in the long run is inversely proportional to these changes, i.e. $(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})$ and $(\theta_{11}+\phi_{11})-(\theta_{12}+\phi_{12})$. It must be noted that $p_1(n)$ does not converge to a constant value since this would imply an absorbing state other than zero or one. The convergence is in the sense indicated in (13) and is that of a sample average. The probabilities of $p_1(n)$ and $p_2(n)$ of the two actions as well as the penalty strengths $s_1(n)$ and $s_2(n)$ vary with time in such manner that

$$\mathrm{E}\left[s_2(n) - s_1(n)\,|\tilde{p}_1(n)\right] \to 0, \tag{17}$$

or the expected penalty strengths of two actions tend to be equalized. This is achieved by α_1 being chosen a fraction $\frac{(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})}{(\theta_{11}+\phi_{11})+(\theta_{22}+\phi_{22})-(\theta_{21}+\phi_{21})-(\theta_{12}+\phi_{12})}$ of the total number of times in the long run. The equalization of the expected values of penalty strengths is one significant feature of this model. ■

4 Conclusions

In this paper, we presented a new state dependent nonstationary environment and study the asymptotic behavior of SL_{R_I} learning algorithm operating under the proposed environment. It is shown that the SL_{R-I} automaton operating in the proposed nonstationary environment, equalizes the expected penalty strengths of actions. This model was motivated by applications of learning automata in call admission in cellular networks.

References

1. K. S. Narendra and K. S. Thathachar, *Learning Automata: An Introduction.* New York: Printice-Hall, 1989.
2. K. S. Narendra and M. A. L. Thathachar, "On the Behavior of Learning Automata in a Changing Environment with Routing Applications," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-10, pp. 262–269, 1980.
3. P. R. Srikantakumar and K. S. Narendra, "A Learning Model for Routing in Telephone Networks," *SIAM Journal of Control and Optimization*, vol. 20, pp. 34–57, Jan. 1982.
4. O. V. Nedzelnitsky and K. S. Narendra, "Nonstationary Models of Learning Automata Routing in Data Communication Networks," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-17, pp. 1004–1015, Nov. 1987.
5. H. Beigy and M. R. Meybodi, "Adaptive Uniform Fractional Guard Channel Algorithm: The Steady State Analysis," in *Proceedings of the ninth International Symposium on Wireless Systems and Networks (ISWSN'03), Dhahran, Saudi Arabia*, Mar. 2003.
6. H. Beigy and M. R. Meybodi, "Adaptive Uniform Fractional Channel Algorithms," *Iranian Journal of Electrical and Computer Engineering, Accepted for Publication*, 2004.
7. H. Beigy and M. R. Meybodi, "Dynamic Guard Channel Scheme Using Learning Automata," in *Proceedings of Sixth World Multiconference on Systemmics, Cybernetics and Informatics, Orlando, USA*, July 2002.
8. H. Beigy and M. R. Meybodi, *A Learning Automata Based Dynamic Guard Channel Scheme*, vol. 2510 of *Springer-Verlag Lecture Notes in Computer Science*, pp. 643–650. Springer-Verlag, Oct. 2002.
9. M. L. Tsetline, *Automata Theory and Modeling of Biological Systems.* New York: Academic Press, 1974.
10. K. S. Narendra and R. Viswanthan, "A Two–Level System of Stochastic Automata for Periodic Random Environments," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-2, Apr. 1972.
11. B. J. Oommen and H. Masum, "Swithching Models for Nonstationary Random Environments," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-25, pp. 1034–1339, Sept. 1995.
12. J. Had-El and Y. Rubinstein, "Optimal Performance of Stochastic Automata in Switched Random Environment," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-7, Sept. 1977.
13. L. A. Rasrrign and K. K. Ripa, "Synthesis of Optimum Algorithms for Discretely Distributed Random Search for Markov Entities," *Automatic Control*, vol. 5, no. 6, pp. 12–19, 1971.
14. P. Srikantakumar, *Learning Models and Adaptive Routing in Telephone and Data Communication Networks.* PhD thesis, Departement of Electrical Engineering, University of Yale, USA, Aug. 1980.