

A Stochastic Technique for On-line Prediction and Tracking of Wireless Packet Networks

R. Chandramouli

Multimedia Systems, Networking, and Communications (MSyNC) Lab
Department of Electrical and Computer Engineering
Stevens Institute of Technology

Abstract

A stochastic learning technique to predict and track the states of a two state wireless packet network that changes according to a Markov process is proposed. The proposed technique does not use pilot symbols or training data for state estimation. Instead, prediction and tracking the network state information is performed in an on-line fashion using stochastic iterative control suitable for real-time/pseudo real-time applications. The proposed method is computationally simple making it a good candidate for low power wireless applications. The computational power consumption due to the predictor/tracker can be traded-off easily for accuracy and speed. Experimental results are promising for the speed and accuracy with which the time-varying network state is predicted and tracked. Theoretical convergence of the iterative control algorithm are also discussed.

1. Introduction

Wireless networks are characterized by limited bandwidth and randomly time-varying links. Therefore, it will be beneficial to adapt the parameters of the transmitter and the receiver according to the channel conditions in order to maximize performance factors such as bandwidth and power efficiency. For example, network-state adaptive power control can prolong the battery-life of wireless devices by adjusting the transmitter power to match the state of the wireless network. When the network state is *good*, transmit power can be decreased and when it is *bad*, it can be increased in order to maintain a constant link quality measured in terms of performance measures such as bit error rate and throughput. Another example where network state information can be exploited beneficially is network-depending scheduling for low power. If the communicating devices know the true state of the wireless network ahead

of time then transmission and reception can be scheduled in such a way that information loss due to network errors caused by factors such as deep fading can be minimized.

Pilot-symbol aided channel estimation schemes have been investigated previously in the literature (e.g., see [1], [2] and [3], and references therein). In these types of schemes, pilot symbols are periodically transmitted along with the data symbols. The receiver then estimates the statistics of the channel by decoding the received pilot symbols. While this technique is quite popular, it has some inherent disadvantages. Pilot symbols can consume sufficient amount of bandwidth depending on the accuracy with which the channel is to be estimated. It has been observed that pilot symboling can cause up to a 14% loss in capacity [1]. In some cases even a 50% loss in throughput is observed [7]. This drawback exists in mobile standards such as the GSM. Also, insertion of pilot symbols causes delays. Such delays may not be acceptable for some applications such as real-time multimedia traffic.

An alternative to pilot-symbol aided channel estimation is the blind estimation schemes. Here, limited or no pilot (or training) symbols are used; however, the channel is estimated using only the received observations through complex signal processing operations. These schemes usually tend to be computationally expensive [5],[8] hence making them unattractive for low power wireless networking. There has been an enormous amount of interesting work done in this area and the research still goes on. We refer to [7] for an excellent overview on this subject. A balance between estimation accuracy versus power consumption will be useful in low power wireless networks.

We consider a finite state Markov (FSM) model for the wireless packet network in this paper. FSM models have been used successfully to model wireless networks [4], [9]. In these models, each state of the finite-state Markov model corresponds to a certain quality of the wireless link. For example, in [4], a two-state Markov model is used where the states correspond to *good* (G) and *bad* (B) states. When the link is in the G state, the bit error probability is within

acceptable levels (usually low) and in the B state the link has an unacceptable bit error rate. Generalizations of this model to multiple states can be found in [9] where each state corresponds to a certain channel signal to noise ratio (from low to high). In the FSM model, the channel is assumed to have transitions from one state to another following the Markov property, *i.e.*, the channel state transition process is a Markov chain. Burst error channels can be very well characterized with the FSM model.

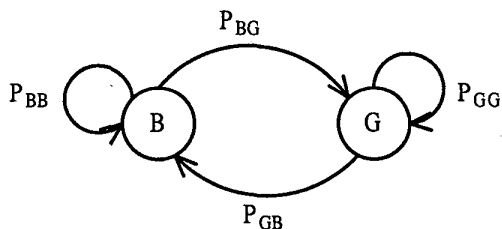


Figure 1. Two state Markov model for a wireless link.

Our goal is to predict the wireless network state and keep track of it without using pilot-symbols or complex signal processing. Towards this objective, we use a stochastic iterative learning control technique. This is based on the use of a stochastic learning automaton [6]. The state of the wireless network is learnt on the fly by the transmitter via simple adaptive stochastic learning based on feedback from the receiver regarding the *quality* of the wireless link. No pilot symbols are used. We note that like any other stochastic learning technique, during the learning phase the performance of the wireless transmission could suffer due to mismatches in the actual network state and the predicted one. However, for applications such as wireless video transmission, such errors are tolerable due to the inherent redundancy in a video stream. Another advantage of the proposed network state estimator is its computational simplicity which is highly desirable for low power wireless applications. Without loss of generality we present results only for the two state Gilbert-Elliot channel model [4] (shown in Figure 1).

The paper is organized as follows. A brief description of the two state Markov model for the wireless network is given in Section 2. The stochastic iterative learning control framework is discussed in Section 3. Section 4 deals with the proposed network state prediction and tracking technique. Experimental results are given in Section 5 and conclusions in Section 6.

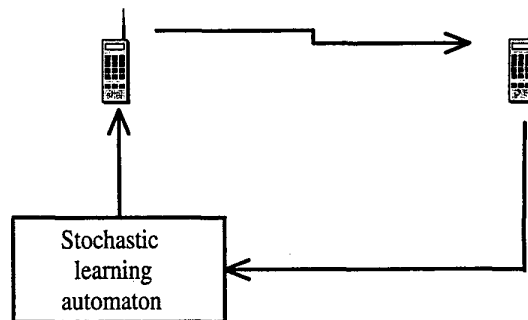


Figure 2. Schematic of the proposed network state prediction and tracking framework.

2. Two State Markov Model

We consider the problem of predicting and tracking the states of a two state wireless network with state transitions from a Markov chain as shown in Figure 1. The state transition probability matrix is given by

$$\begin{bmatrix} P_{BB} & P_{BG} \\ P_{GB} & P_{GG} \end{bmatrix} \quad (1)$$

where $P_{BB} = P(\text{current state is B} | \text{previous state is B})$ etc. These transition probabilities depend on the rate of transmission, fading conditions etc. [9]. The burstiness of the channel is controlled by the transition probability matrix. Each state in the model corresponds to a binary symmetric channel. If these states are predicted ahead of time and tracked then the communication parameters can be adapted accordingly for improving the system performance.

3. Stochastic Learning Automaton

In this section we briefly discuss the concept behind the variable structure stochastic learning automaton (VSLA) [6]. Abstractly, a learning automaton can be considered to be an object that can choose from a finite number of actions. For every action that it chooses, the random environment in which it operates evaluates that action. A corresponding feedback is sent to the automaton based on which the next action is chosen. As this process progresses the automaton learns to choose the *optimal* action for that unknown environment asymptotically. The stochastic iterative algorithm used by the automaton to select its successive actions based on the environment's response defines the stochastic learning algorithm. An important property of the learning au-

tomaton is its ability to improve its performance with time while operating in an unknown environment.

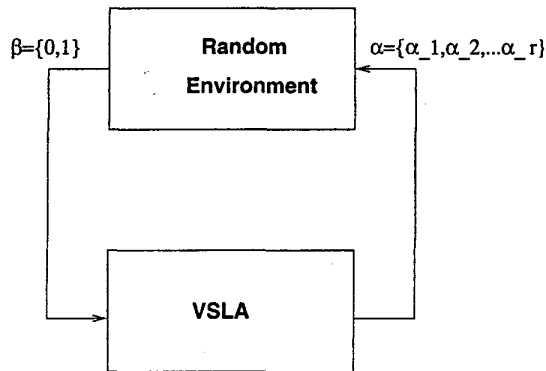


Figure 3. Variable structure stochastic learning automaton.

A VSLA is described by a 5-tuple $\{\alpha, \beta, p, T, C\}$, where $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ is the set of r actions from which the automaton can choose any action at time n denoted by $\alpha(n)$, $\beta = \{0, 1\}$ is the set of binary response to the VSLA from the environment for a chosen action; $\beta = 0$ is a reward and $\beta = 1$ corresponds to a penalty and $p = \{p_1, p_2, \dots, p_r\}$ is the set of probabilities with which each action is chosen. These are functions of time. T is the stochastic learning algorithm according to which the elements of the set p are updated at each time n , i.e., $p(n+1) = T[\alpha(n), \beta(n), p(n)]$, where the i th element of the set $p(n)$ is $p_i(n) = Pr(\alpha(n) = \alpha_i)$, $i = 1, 2, \dots, r$, $\sum_{i=1}^r p_i(n) = 1$, $\forall n$ and $p_i(1) = \frac{1}{r}$, $\forall i$. $C = \{c_1, c_2, \dots, c_r\}$ is the set of penalty probabilities conditioned on the chosen action, where $c_i = Pr(\beta = 1 | \alpha(n) = \alpha_i)$, $i = 1, 2, \dots, r$. The values of c_i are unknown and it is assumed that C has a unique minimum element. The working of the learning automaton can be described as follows. Initially at $n = 1$ one of the actions is chosen by the automaton at random with a given probability. This action is then applied to the system and the response from the environment is observed. If the response is favorable ($\beta = 0$) then the probability of choosing that action for the next period of time is updated according to the updating rule T . Then, another action is chosen and the response is observed. This process is repeated until a stopping criterion is reached. When the learning process stops the algorithm has learnt to choose the optimal action and the characteristics of the unknown random environment.

4. Network State Prediction and Tracking

The proposed framework for stochastic prediction and tracking of the wireless network state is shown in Figure 2. The transmitter sends a packet initially without knowing the actual network state. An initial estimate for the probability of the wireless network being in one of the two states is assumed (usually $1/2$). The receiver computes the link quality based on the number of errors in a received packet or a sequence of packets. This is possible using the error detection capability of an error control code in the packet or CRC bits. If the number of errors in the packet are acceptable, then a reward is sent to the transmitter via the feedback channel, if not, a penalty is transmitted. Based on this feedback the transmitter updates its current probability estimates for the networks states and the transmission parameters such as power and coding are chosen (controlled) appropriately. This process continues until the probability estimates converge and the channel state is predicted with a high confidence. When the network state changes the reward and penalty feedback also change accordingly to reflect the current status. The network state probabilities are updated by the learning automaton stochastically and the new network state is learnt enabling channel tracking. We assume that the feedback to the stochastic learning automaton is error-free. Effects of erroneous feedback and feedback delays are part of the ongoing research. Also, in our implementation we have assumed a feedback after every packet. This can be relaxed and a feedback can be sent to the learning automaton after every few packets.

We propose to use a linear iterative learning algorithm for the variable structure stochastic learning automaton (VSLA). Note that the VSLA now operates in a non-stationary environment because the network states changes according to a Markov chain. But, when the network is in a specific state, it is stationary in the sense that the bit error rate is a constant. At any time n the automaton in Figure 2 can take one of two actions, namely,

$$\alpha(n) = \begin{cases} \alpha_1 : \text{do not transmit packet} \\ \alpha_2 : \text{transmit packet, } n = 1, 2, 3, \dots \end{cases} \quad (2)$$

This can be achieved by the transmitter via power control. When the transmitted power is zero, it corresponds to the action of not transmitting. Let $\beta_i(n)$, $i = 1, 2$, $n = 1, 2, 3, \dots$ correspond to the feedback from the receiver for action $\alpha_i(n)$. $\beta(n) = 0$ is a reward and $\beta(n) = 1$ is a penalty. Then the proposed stochastic learning algorithm to predict and track the non-stationary environment is de-

scribed by a linear reward-penalty learning algorithm [6],

$$\begin{aligned}
 &\text{if } \alpha(n) = \alpha_1 \text{ and } \beta_1(n) = 0 \\
 &p_1(n+1) = p_1(n) + a(1 - p_1(n) - A); \\
 &p_2(n+1) = p_2(n) - a(p_2(n) - A), \\
 \\
 &\text{if } \alpha(n) = \alpha_1 \text{ and } \beta_1(n) = 1 \\
 &p_1(n+1) = p_1(n) - b(1 - p_1(n) - B); \\
 &p_2(n+1) = p_2(n) + b(p_2(n) - B), \\
 \\
 &\text{if } \alpha(n) = \alpha_2 \text{ and } \beta_1(n) = 0 \\
 &p_1(n+1) = p_1(n) - a(p_1(n) - B), \\
 &p_2(n+1) = p_2(n) + a(1 - p_2(n) - B); \\
 \\
 &\text{if } \alpha(n) = \alpha_2 \text{ and } \beta_1(n) = 1 \\
 &p_1(n+1) = p_1(n) + b(p_1(n) - A) \\
 &p_2(n+1) = p_2(n) - b(1 - p_2(n) - A)
 \end{aligned} \tag{3}$$

where a and b are parameters that control the speed versus accuracy of the stochastic learning algorithm and A and B control how close the action probability estimates get to 0 and 1. They also determine the amount of exploration versus exploitation during learning. The following three remarks answer questions about the convergence and optimality of the stochastic learning control algorithm. Proofs are not given due to space limitations.

Remark 1: The discrete-time homogeneous Markov process $\{p(n) = (p_1(n), p_2(n))\}$ with state space given by the unit simplex, $S_2 = \{p : p_1 + p_2 = 1, p_1, p_2 \geq 0\}$ does not have any absorbing states. This enables the tracking of a time-varying environment.

Remark 2: $\{p(n)\} \rightarrow p^*$ in distribution.

Remark 3: The iterative stochastic control defined by Eq. (3) is asymptotically optimal.

5. Experimental Results

For experimental evaluation of the proposed technique we considered slowly time-varying bursty wireless channels. We now present a specific example from our experiments. A wireless network with the following state transition probability matrix as given in Eq. (1) was simulated,

$$\begin{bmatrix} 0.98 & 0.02 \\ 0.02 & 0.98 \end{bmatrix} \tag{4}$$

The bit error probability in state B was 0.5 and in G it was equal to 10^{-3} . The simulated true states of the network are shown in Figure 4. We observe the bursty nature of the network. The following choices for the parameters were made in the simulation, $a = b = 0.3$, $A = 0.01$

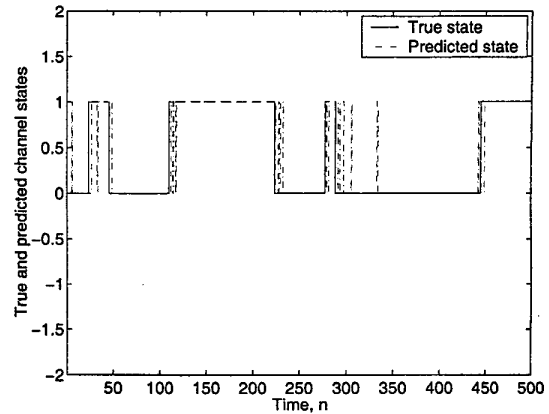


Figure 4. True and predicted network states. "0" stands for state B and "1" denotes state G.

and $B = 0$. The predicted states using the proposed algorithms as defined by Eq. (3) are shown in Figure 4. From the figure we see that the learning algorithm performs quite well and is able to track the sudden changes in the network state. Of course, during the initial learning period, there are some mis-predictions. This is seen in the figure as mismatches between the true and predicted network states. But, applications such as wireless video can tolerate these mis-predictions. Note that the learning period is sufficiently small when compared to the duration of a state allowing its use in many practical wireless multimedia networking applications over bursty channels.

Figures 5 and 6 show the evolution of the state probability estimates with time. We again see that these estimates are consistent with the evolution of the network states as seen in Figure 4. When the channel state is G, the probability of transmission increases and when the state changes to B, transmission probability decreases accordingly.

6. Conclusion

A stochastic learning technique for on-line prediction and tracking of a two state Markov model for wireless packet networks is proposed. The proposed method does not incur the overhead due to pilot symbols and is computationally simple; thus, requiring very little computational power. Speed versus prediction/tracking accuracy can be controlled by the application by changing the parameters of the algorithm. The method has been shown to be asymptotically optimal. This method is particularly suited for

network-adaptive wireless multimedia applications where low power and high throughput are crucial.

Acknowledgements

Part of this work was supported by NSF grant no. 0082064.

References

- [1] J. Cavers. An analysis of pilot-symbol assisted modulation for rayleigh fading channels. *IEEE Trans. on Vehicular Technology*, 40:686–693, Nov. 1991.
- [2] S. Crozier, D. Falconer, and S. Mahmoud. Reduced complexity short-block data detection techniques for fading time-dispersive channels. *IEEE Trans. on Vehicular Technology*, 41:255–265, Aug. 1992.
- [3] P. S. C. Georghiades. Complementary sequences for isi channel estimation. *IEEE Trans. on Information Theory*, 47(3):1145–1152, March 2001.
- [4] E. Gilbert. Capacity of a burst-noise channel. *Bell Systems Tech. J.*, 39(9):1253–1265, Sept. 1960.
- [5] H. Liu and K. Li. Low complexity receivers for cdma communications over frequency-selective channels. *Proc. Thirtieth Asilomar Conference on Signals, Systems, and Computers*, 1:632–636, 1997.
- [6] K. Narendra and M. Thathachar. *Learning Automata: An Introduction*. Prentice Hall, 1989.
- [7] L. Tong and S. Perreau. Multichannel blind identification: From subspace to maximum likelihood methods. *Proc. of the IEEE*, 86(10):1951–1968, Oct. 1998.
- [8] M. Tourmier, A. Ferreol, and P. Larzabal. Low complexity blind space-time identification of propagation parameters. *Proc. ICASSP*, 5:2873–2876, 1999.
- [9] H. Wang and N. Moayeri. A finite-state markov channel—a useful model for radio communication channels. *IEEE Trans. on Veh. Tech.*, 44:163–171, Feb. 1995.

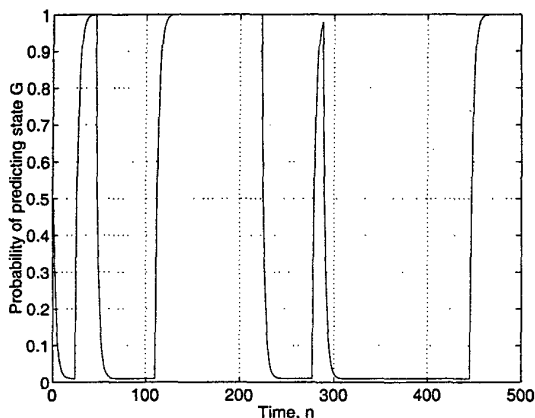


Figure 5. Evolution of the probability of predicting state G from Eq. (3).

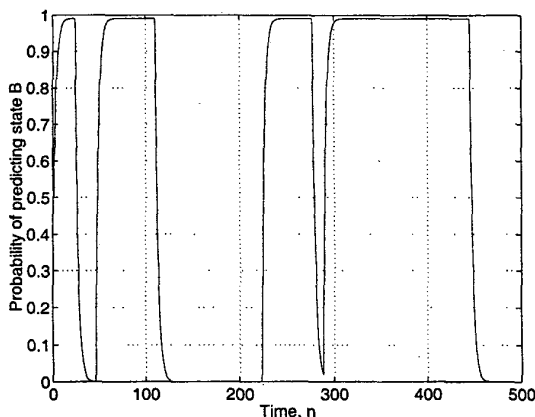


Figure 6. Evolution of the probability of predicting state B from Eq. (3).