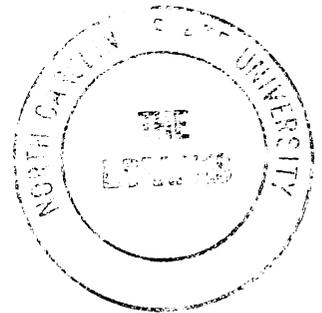


Importance Sampling Methodologies for Simulation of Communication Systems with Time-Varying Channels and Adaptive Equalizers

Wael A. Al-Qaq

Michael Devetsikiotis

J. Keith Townsend



Center for Communications and Signal Processing,
Department of Electrical and Computer Engineering
North Carolina State University

TR-93/8

April 1993

TK5101

A1

T72

93/8

1993

To appear (with revisions) in the *IEEE Journal on Selected Areas in Communications*

Importance Sampling Methodologies for Simulation of Communication Systems with Time-Varying Channels and Adaptive Equalizers

Wael A. Al-Qaq, *Student Member IEEE*

Michael Devetsikiotis, *Student Member IEEE*

J. Keith Townsend, *Member IEEE*

Center for Communications and Signal Processing,
Department of Electrical & Computer Engineering,
North Carolina State University, Raleigh, NC 27695-7914

Abstract

Formula-based analysis of communication links with time-varying channels and adaptive equalizers is not always tractable — Monte Carlo simulation must be used to obtain bit error rate (BER) estimates of the system. Conventional Monte Carlo simulation will not provide estimates for low BER's due to excessive run time. Although importance sampling (IS) techniques offer the potential for large speed-up factors for BER estimation using MC simulation, IS techniques have not been used for simulating communication links with adaptive equalizers.

In this paper we present, for the first time, two IS methodologies for MC simulation of communication links characterized by time-varying channels and adaptive equalizers. One methodology is denoted here as the “twin system” (TS) method. A key feature of the TS method is that biased noise samples are input to the adaptive equalizer, but the equalizer is only allowed to adapt to these samples for a time interval equal to the memory of the system.

In addition to the TS technique, we also present a statistically biased, but simpler technique for using IS with adaptive equalizers which is based on the independence assumption between the equalizer input and the equalizer taps (the “IA” method).

Experimental results are presented that show run time speed-up factors of two to seven orders of magnitude for a static linear channel with memory, and of two to almost five orders of magnitude for a slowly varying, random linear channel with memory, for both the IA and TS methods.

1. This research was sponsored in part by an IBM Fellowship.
2. Portions of this paper have been presented at the IEEE GLOBECOM'92, Orlando, Florida, December 1992.

1 Introduction

Performance analysis of digital communication links characterized by time-varying channels is a difficult problem. The recent flurry of interest in wireless mobile communications has also increased the importance of this problem. In these links, adaptive equalizers are used to track the time-varying channel and ensure satisfactory performance [1, 2]. Formula-based analysis of these links is not always possible — Monte Carlo simulation must be used to obtain bit error rate (BER) estimates of the system. Depending on the complexity of these systems, Monte Carlo simulation can provide accurate BER estimates of a time-varying system with an adaptive equalizer down to approximately 10^{-4} . Although this may be sufficient for many voice-band applications, in the future these networks will require lower error rates, thus exceeding the capabilities of conventional Monte Carlo simulation.

Importance sampling (IS) techniques offer the potential for large speed-up factors for BER estimation using MC simulation [3, 4, 5, 6]. However, IS techniques have not been used to speed-up MC simulation of communication links with adaptive equalizers. Appropriate IS parameter values must be used, since poor choices for these parameters could even *reduce* simulation efficiency, resulting in slow-down instead of speed-up. In the past, we have presented methods for locating near-optimal IS parameter values for the simulation of *static* communication channels [7, 8].

In addition to formulation problems, the use of IS in MC simulation of adaptive systems during the training period (i.e., the initial period when the actual data transmitted is fed back to the adaptive equalizer to help it correctly track channel variations) is faced with the difficulty that the adaptive system will attempt to undo the effect of the IS biasing scheme. A necessary requirement of any good IS biasing scheme is that the scheme causes a significant increase in the “raw” (i.e., unweighted) BER of the system. Since we are primarily concerned with the performance of adaptive equalizers in the period subsequent to the training period, and since adaptive equalizers adjust to force the BER toward zero during the training period, our IS technique is applied to the adaptive system in the period that follows the training period. During this period, the output decisions are fed back to update the equalizer taps and track channel variations.

A fundamental complication in finding an appropriate IS methodology is the fact that adaptive systems possess a cumulative memory that extends back to time zero. The deleterious effects of large memory on IS are well known. Also, the adaptive equalizer taps, and thus the optimal IS parameter values, are a function of time.

In this paper we present, for the first time, two IS methodologies for MC simulation of communication links characterized by time-varying channels and adaptive equalizers. One is denoted here as the “twin system” (TS) method. A key feature of the TS method is that biased noise samples are input to the adaptive equalizer, but the equalizer is only allowed to adapt to these samples for a time interval equal to the memory of an equivalent system model. The memory for this equivalent system model remains constant. At the end of this interval, a decision is collected. The adaptive equalizer taps are then reset to values which would have resulted had there been no IS. A new optimal IS biasing value for the next vector of samples is calculated, and the process (a “biasing cycle”) is repeated.

Embodied in the TS technique is a novel simulation model consisting of the “original system” and a “twin system”. The original system keeps track of the equalizer taps variation with time with unbiased noise samples. Biased noise samples are applied to the twin system. Also, MC statistics (using IS) are measured at the output of the *twin* system. Equalizer tap values are transferred from the original system to the twin system at the beginning of each biasing cycle.

The chief contribution of this paper is to present the theoretical foundation for the TS and IA estimators and demonstrate the potential of these techniques with examples. Keep in mind that it is not possible to calculate the exact BER using closed-form analysis even for a static linear channel with an adaptive equalizer.

To formulate the TS method, it was necessary to develop an equivalent system model which combines memory elements on both the analog and digital sides of the decision operation into one equivalent system. The equivalent system model was derived for the LMS algorithm, but the TS and IA methodologies presented in this paper are applicable to all adaptive algorithms with differences merely in the details of developing an equivalent system model.

In addition to the TS technique, we also present a method for using IS with adaptive equalizers which is based on the independence assumption between the equalizer input and the equalizer taps. This “IA” method has clear implementation advantages over the twin system method, but yields a statistically biased estimator, with the bias increasing with the amount of correlation between the input data and taps. The assumption of independence between the input and equalizer taps is widely used in literature on adaptive systems (see [9] and references within). Furthermore, and most importantly, our experimental results also support this assumption and demonstrate that the IS estimator based on the IA method offers significant speed-up factors, while its relative statistical bias with respect to the “correct” TS estimator is acceptably small. Thus the IA estimator provides an alternative approach, and could be especially useful for cross-checking and validation.

The details of the equivalent system are presented in Section 2. The twin system and independence assumption methods are formulated in Section 3. Experimental results presented in Section 4 show run time speed-up factors of two to seven orders of magnitude for a static linear channel with memory, and of two to almost five orders of magnitude for a slowly varying, random linear channel with memory for both the IA and TS methodologies. The overhead of the TS technique is modest, reducing the net speed-up by roughly a factor of 2 for a 6-tap equalizer, and a factor of approximately 3.5 for a 10-tap equalizer. The proof of a result useful for selecting biasing parameter values is given in the Appendix.

2 Formulation

2.1 Adaptive Systems Model

Fig. 1 shows the block diagram of a digital communication system in which a linear adaptive equalizer is used to compensate for the distortion caused by the transmission medium (channel). The binary sequence of information symbols a_k is fed to the transmit filter $p(t)$

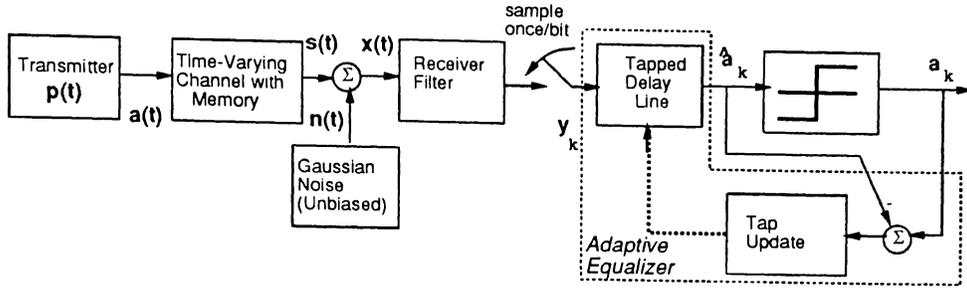


Figure 1: Block diagram of a digital communication system which includes a time-varying channel with memory and an adaptive equalizer.

whose output is given by

$$a(t) = \sum_{k=0}^{\infty} a_k p(t - kT_s)$$

where $\frac{1}{T_s}$ is the bit rate in the system. The channel $z(\cdot, t)$ distorts the transmitted signal $a(t)$ and causes intersymbol interference (ISI). The signal $s(t)$ is given by

$$s(t) = z(a(t), t)$$

The signal coming out of the channel is corrupted by additive noise, which is usually wide band Gaussian noise. The resulting signal at the input to the receiver is given by

$$x(t) = s(t) + n(t)$$

The output of the (linear) receiver is sampled once every bit interval (i.e. every T_s seconds) to obtain

$$y(k) = x(t) * q(t)|_{t=kT_s}$$

In addition to the distorted transmitted symbol, the output signal $y(k)$ contains intersymbol interference due to channel distortion, as well as additive noise.

Let the equalizer be an FIR filter (tapped delay line) with M adjustable taps, $h_i(k)$, $i = 0, \dots, M - 1$, at every time k . The equalizer output may be expressed as

$$\hat{a}_k = \sum_{m=0}^{M-1} h_m(k) y(k - m)$$

There are different algorithms for updating the equalizer taps with time k . These algorithms vary in performance, complexity, and applications. The mean square error (MSE) at instant k is given by

$$MSE(k) = E\{e^2(k)\} = E\{(a_k - \hat{a}_k)^2\} = E\{(a_k - \sum_{m=0}^{M-1} h_m(k) y(k - m))^2\}$$

Typically, the purpose of the adaptive equalizer is to minimize the MSE with respect to the equalizer taps [1, 2]. In this paper, we base our discussion on the least mean square (LMS)

algorithm, but the general MC and IS formulations of sections 2.3 and 2.4 with our IS methodologies of section 3 are applicable to all adaptive algorithms with differences merely in the details of finding the equivalent system model.

According to the LMS algorithm, the search for the minimum MSE is done using the gradient descent technique. For real valued data signals, the corresponding update equation for the equalizer taps is given by

$$\mathbf{h}(k) = \mathbf{h}(k-1) + \mu e(k-1)\mathbf{y}(k-1)$$

where

$$\mathbf{h}(k) = [h_0(k), h_1(k), \dots, h_{M-1}(k)]$$

and

$$\mathbf{y}(k) = [y(k), y(k-1), \dots, y(k-M+1)]$$

Furthermore,

$$e(k) = a_k - \hat{a}_k$$

where μ is the step size, with $0 < \mu < 2/[MP_{yy}(0)]$ [1]. $P_{yy}(0)$ is the power in the signal $y(k)$ that is easily estimated from the received signal.

2.2 Equivalent System Model

In order to develop a model of our adaptive system suitable for use with IS, we need to express the final output \hat{a}_k as a function of the random inputs $x(l)$, $l = 0, \dots, k$, to the receiver filter. This function must have components in both the analog and the digital parts of the receiver, posing certain modeling difficulties. We overcome these difficulties as follows: Let M_1 be the channel memory in bits. Let M_2 be the receiver filter memory in bits. Denote the receiver filter taps by

$$\mathbf{q} = [q_1, q_2, \dots, q_{M_2L}]$$

where L is the number of samples per bit. Let \mathbf{S} , \mathbf{N} , and \mathbf{X} denote the $1 \times \hat{M}$ vectors of bit-length vectors $\mathbf{s}(i)$, $\mathbf{n}(i)$, and $\mathbf{x}(i)$, respectively, defined as follows:

$$\mathbf{S}(k) = [s(k) | s(k-L) | \dots | s(k - (\hat{M} - 1)L)]$$

$$\mathbf{s}(i) = [s(i), \dots, s(i-L+1)], \quad i = k, k-L, \dots, k - (\hat{M} - 1)L$$

$$\mathbf{N}(k) = [n(k) | n(k-L) | \dots | n(k - (\hat{M} - 1)L)]$$

$$\mathbf{n}(i) = [n(i), \dots, n(i-L+1)], \quad i = k, k-L, \dots, k - (\hat{M} - 1)L$$

Then

$$\mathbf{X}(k) = \mathbf{S}(k) + \mathbf{N}(k) = [x(k) | x(k-L) | \dots | x(k - (\hat{M} - 1)L)]$$

with

$$\mathbf{x}(i) = [x(i), \dots, x(i-L+1)], \quad i = k, k-L, \dots, k - (\hat{M} - 1)L$$

The important modeling result that follows is that

$$\hat{a}_k = \mathbf{X}(k)\mathbf{h}_{eq}^T(k)$$

where $\hat{M} = M + M_2 - 1$ bits (corresponding to $\hat{M} \times L$ samples),

$$\mathbf{h}_{eq}(k) = \mathbf{h}(k)\mathbf{A}^T$$

and \mathbf{A} is the $\hat{M}L \times M$ matrix

$$\begin{bmatrix} \mathbf{q}_{M_2L \times 1}^T & \mathbf{0}_{L \times 1} & \cdots & \mathbf{0}_{L \times 1} \\ \mathbf{0}_{L \times 1} & \mathbf{q}_{M_2L \times 1}^T & \cdots & \mathbf{0}_{L \times 1} \\ \cdot & \mathbf{0}_{L \times 1} & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \mathbf{0}_{L \times 1} \\ \mathbf{0}_{L \times 1} & \mathbf{0}_{L \times 1} & \cdots & \mathbf{q}_{M_2L \times 1}^T \end{bmatrix}$$

Observe that, since $\mathbf{y}(k) = \mathbf{X}(k)\mathbf{A}$, we have $\mathbf{y}(k-1) = \mathbf{X}(k-L)\mathbf{A}$ and

$$\mathbf{h}_{eq}(k) = \mathbf{h}_{eq}(k-1) + \mu e(k-1)\mathbf{X}(k-L)\mathbf{A}\mathbf{A}^T \quad (1)$$

Thus, $\mathbf{h}_{eq}(k)$ is a function of $\mathbf{X}(k-L)$ and $\mathbf{h}_{eq}(k-1)$. But $\mathbf{h}_{eq}(k-1)$ is a function of $\mathbf{X}(k-2L)$ and $\mathbf{h}_{eq}(k-2)$ and so on, therefore we can conclude that

$$\mathbf{h}_{eq}(k) = \mathbf{g}(\mathbf{x}(0), \dots, \mathbf{x}(k-L)) = \mathbf{g}'(\mathbf{X}_C(k-L), \mathbf{X}(k-L)) \quad (2)$$

where \mathbf{g} and \mathbf{g}' are functionals dependent on \mathbf{A} and μ and

$$\mathbf{X}_C(k) = [\mathbf{x}(k - \hat{M}L) | \dots | \mathbf{x}(0)]$$

is the vector “complementary” to $\mathbf{X}(k)$. Note that (2) is general in describing the cumulative memory behavior of most adaptive algorithms, where the adaptive equalizer taps at any instant k are a function of the input samples extending all the way back to $k = 0$.

2.3 Monte Carlo Formulation

For the time-varying communication system in question, the probability of a detection error at any instant k is given by

$$\begin{aligned} P_e(k) &= E\{I(\mathbf{X}(k)\mathbf{h}_{eq}^T(k))\} \\ &= \int I(\mathbf{a}\mathbf{b}^T) f_{\mathbf{X}(k), \mathbf{h}_{eq}(k)}(\mathbf{a}, \mathbf{b}) d\mathbf{a} d\mathbf{b} \\ &= \int_{\Omega_{\mathbf{X},k}} f_{\mathbf{x}(0), \dots, \mathbf{x}(k)}(\xi_0, \dots, \xi_k) d\xi_0 \dots d\xi_k \end{aligned} \quad (3)$$

where $f_{\mathbf{X}(k), \mathbf{h}_{eq}(k)}$ is the joint probability density function (pdf) of $\mathbf{X}(k)$ and $\mathbf{h}_{eq}(k)$, $f_{\mathbf{x}(0), \dots, \mathbf{x}(k)}$ is the joint pdf of the $\mathbf{x}(i)$'s, $i = 0, \dots, k$,

$$\Omega_{\mathbf{X},k} = \{(\mathbf{x}(0), \dots, \mathbf{x}(k)) : I(\mathbf{X}(k)\mathbf{h}_{eq}^T(k)) = 1\}$$

and $I(\cdot)$ is an *indicator* function equal to 1 when the detected bit is in error, and 0 otherwise.

The bit error rate (BER) given by (3) is defined as the *instantaneous* BER. A Monte Carlo (MC) estimator of (3) is given by the sample mean expression

$$\hat{P}_e(k) = \frac{1}{N} \sum_{i=1}^N I(\mathbf{X}(k, i) \mathbf{h}_{eq}^T(k, i)) \quad (4)$$

where $\mathbf{X}(k, i)$ and $\mathbf{h}_{eq}^T(k, i)$ are i.i.d. repetitions of $\mathbf{X}(k)$ and $\mathbf{h}_{eq}^T(k)$, respectively. Note that $\mathbf{X}(k, i)$ and $\mathbf{h}_{eq}^T(k, i)$ are not independent, and therefore cannot be generated independently. The averaging in the above estimator is taken with respect to the ensemble of the discrete-time random process $I(\mathbf{X}(k) \mathbf{h}_{eq}^T(k))$. For a low BER (which requires a large N in (4)), generating i.i.d. repetitions of $\mathbf{h}_{eq}^T(k)$ is a formidable task. As evident from eq. (2), each repetition requires the equalizer to start adapting all over again starting at $k = 0$ up to time $k - L$, an interval which can be arbitrarily long. To resolve this problem, we make the assumption that the discrete-time random process $I(\mathbf{X}(k) \mathbf{h}_{eq}^T(k))$ is ergodic for $k_0 < k \leq K$, where k_0 is chosen to be greater than the training period, and K is some large number. This assumption is reasonable to make in the period after the equalizer taps have stabilized (but continue to adapt), and tap fluctuations are mainly due to time variations in the channel and the additive noise. Using the assumption of ergodicity, we can replace ensemble averages by time averages, thus we can replace the estimator of (4) by the following estimator

$$\hat{P}_e = \frac{1}{K - k_0} \sum_{k=k_0+1}^K I(\mathbf{X}(k) \mathbf{h}_{eq}^T(k)) \quad (5)$$

2.4 Importance Sampling Formulation

In order to apply IS observe that

$$P_e(k) = \int_{\Omega_{\mathbf{X},k}^*} f_{\mathbf{x}(0),\dots,\mathbf{x}(k)}^*(\xi_0, \dots, \xi_k) w_{\mathbf{x}(0),\dots,\mathbf{x}(k)}(\xi_0, \dots, \xi_k) d\xi_0 \dots d\xi_k \quad (6)$$

where $w_{\mathbf{x}(0),\dots,\mathbf{x}(k)}(\xi_0, \dots, \xi_k) = f_{\mathbf{x}(0),\dots,\mathbf{x}(k)}(\xi_0, \dots, \xi_k) / f_{\mathbf{x}(0),\dots,\mathbf{x}(k)}^*(\xi_0, \dots, \xi_k)$ and

$$\Omega_{\mathbf{X},k}^* = \{(\mathbf{x}(0), \dots, \mathbf{x}(k)) : I(\mathbf{X}^*(k) \mathbf{h}_{eq}^{*T}(k)) = 1\}$$

$\mathbf{X}^*(k)$ denotes the biased vector $[\mathbf{x}^*(k) | \mathbf{x}^*(k - L) | \dots | \mathbf{x}^*(k - (\hat{M} - 1)L)]$ and the asterisk of $\mathbf{h}_{eq}^*(k)$ implies that the equalizer taps *have been adapting* to the *biased* random numbers $\mathbf{x}^*(0), \dots, \mathbf{x}^*(k)$.

If the sequence of input data $\mathbf{x}^*(i), i = 0, 1, \dots, k$, are independent, then (6) can be written as

$$P_e(k) = \int_{\Omega_{\mathbf{X},k}^*} \prod_{i=0}^{i=k} f_{\mathbf{x}(i)}^*(\xi_i) \prod_{i=0}^{i=k} w_{\mathbf{x}(i)}(\xi_i) d\xi_0 \dots d\xi_k \quad (7)$$

with $w_{\mathbf{x}(i)}(\xi_i) = f_{\mathbf{x}(i)}(\xi_i) / f_{\mathbf{x}(i)}^*(\xi_i)$. Thus, an *unbiased*, IS-based, MC estimator of (3) is given by

$$\hat{P}_e^*(k) = \frac{1}{N^*} \sum_{i=1}^{N^*} I(\mathbf{X}^*(k, i) \mathbf{h}_{eq}^{*T}(k, i)) \prod_{j=0}^{j=k} w_{\mathbf{x}(j,i)}(\mathbf{x}^*(j, i)) \quad (8)$$

Similarly (assuming ergodicity), the IS-based, MC estimator becomes

$$\hat{P}_e^* = \frac{1}{K^* - k_0} \sum_{k=k_0+1}^{K^*} I(\mathbf{X}^*(k) \mathbf{h}_{eq}^{*T}(k)) \prod_{j=0}^{j=k} w_{x(j)}(\mathbf{x}^*(j)) \quad (9)$$

It can be seen from (8) that an unbiased estimate of $P_e(k)$ requires the collection of weights from the time the system starts adapting (i.e $k = 0$). the same requirement also applies to the estimate of (9). In this situation (regardless of the biasing scheme), the main difficulty encountered is related to the *infinite cumulative* memory involved in (3) and (6), with the obvious detrimental effects on IS performance. In general, it is not clear how one should bias the noise source starting at $k = 0$, and how that would affect the variance (which is impossible to find in closed form) of the estimates given by (8) and (9).

3 IS Methodologies

3.1 Twin System IS

To overcome the difficulties introduced by the increasing cumulative memory involved in (3) and (6), we propose a block biasing approach, namely the *twin system* (TS) method: (6) can be rewritten as

$$P_e(k) = \int_{\Omega_{\mathbf{X},k}^*} f_{\mathbf{X}_C(k),\mathbf{X}(k)}^*(\mathbf{a}, \mathbf{b}) w_{\mathbf{X}_C(k),\mathbf{X}(k)}(\mathbf{a}, \mathbf{b}) d\mathbf{a} d\mathbf{b} \quad (10)$$

Therefore, when only $\mathbf{X}(k)$ is biased, denoted by $\mathbf{X}^*(k)$, where

$$w_{\mathbf{X}_C(k),\mathbf{X}(k)}(\mathbf{a}, \mathbf{b}) = f_{\mathbf{X}_C(k),\mathbf{X}(k)}(\mathbf{a}, \mathbf{b}) / f_{\mathbf{X}_C(k),\mathbf{X}(k)}^*(\mathbf{a}, \mathbf{b})$$

$\mathbf{h}_{eq}^*(k)$ is the value assumed by the *resulting biased random vector of taps*:

$$\mathbf{h}_{eq}^*(k) = \mathbf{g}(\mathbf{x}(0), \dots, \mathbf{x}(k - \hat{M}L), \mathbf{x}^*(k - (\hat{M} - 1)L), \dots, \mathbf{x}^*(k - L))$$

At each instant k the random vector $\mathbf{X}(k)$ is biased while the random vector $\mathbf{X}_C(k)$ is left unmodified. Then, a statistically unbiased IS estimate of $P_e(k)$ based on N^* decisions is given by

$$\hat{P}_{e,TS}^*(k) = \frac{1}{N^*} \sum_{i=1}^{N^*} I(\mathbf{X}^*(k, i) \mathbf{h}_{eq}^{*T}(k, i)) w_{\mathbf{X}_C(k),\mathbf{X}(k)}(\mathbf{X}_C(k, i), \mathbf{X}^*(k, i)) \quad (11)$$

Similarly, under the ergodicity assumption, we have the following IS estimator

$$\hat{P}_{e,TS}^* = \frac{1}{K^* - k_0} \sum_{k=k_0+1}^{K^*} I(\mathbf{X}^*(k\hat{M}L) \mathbf{h}_{eq}^{*T}(k\hat{M})) w_{\mathbf{X}_C(k\hat{M}L),\mathbf{X}(k\hat{M}L)}(\mathbf{X}_C(k\hat{M}L), \mathbf{X}^*(k\hat{M}L)) \quad (12)$$

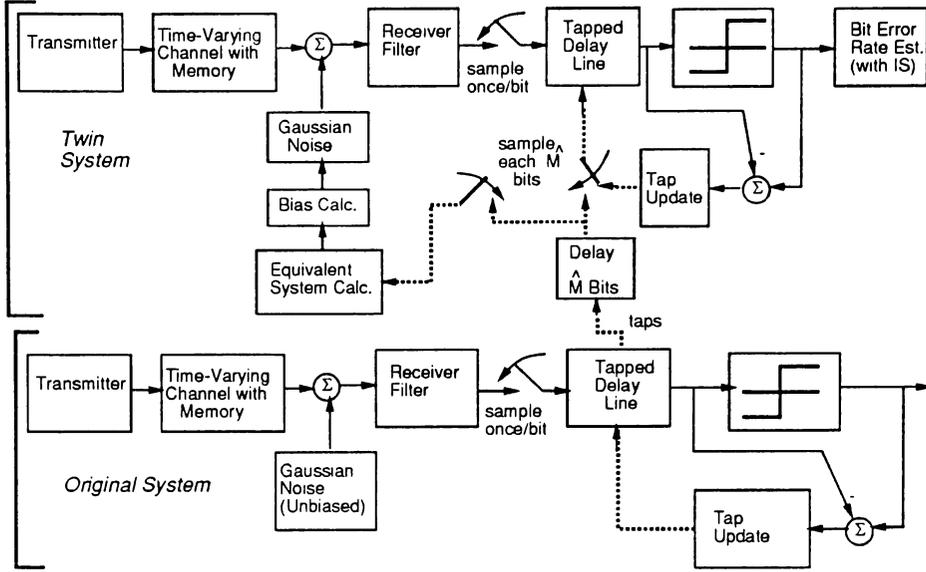


Figure 2: Block diagram of the twin system simulation approach. Equalizer taps from the original, unbiased system are used by the twin system to reset the equalizer at the beginning of each biasing cycle.

Furthermore, if the random vectors $\mathbf{X}_C(k)$ and $\mathbf{X}^*(k)$ are statistically independent, the weight function becomes

$$w_{\mathbf{X}_C(k), \mathbf{X}(k)}(\mathbf{X}_C(k, i), \mathbf{X}^*(k, i)) = w_{\mathbf{X}(k)}(\mathbf{X}^*(k, i)) = f_{\mathbf{X}(k)}(\mathbf{X}^*(k, i)) / f_{\mathbf{X}^*(k)}(\mathbf{X}^*(k, i))$$

since $\mathbf{X}_C(k)$ is left unbiased. The terms in the summation of (12) are *not independent* since the sequence of random vectors $\{\mathbf{h}_{eq}^T(k\hat{M})\}, k_0 < k \leq K^*$ is not independent. This will contribute to an increase in the MC estimator variance. In an attempt to improve the variance of the estimator, we let decisions be spaced \hat{M} bits (corresponding to $\hat{M}L$ samples) apart to guarantee that the sequence of random vectors $\{\mathbf{X}(k\hat{M}L)\}$ is i.i.d. [5].

We call the above estimator of (12) the *twin system* (TS) estimator, since it can be implemented as follows: Maintain two “parallel” versions of the system model. The *original* system is allowed to evolve without IS biasing. In the so-called “twin system”, biased noise samples are input to the adaptive equalizer, but the equalizer is only allowed to adapt to these samples for a time interval equal to the memory \hat{M} of the equivalent system \mathbf{h}_{eq} . At the end of this interval a decision is collected. The adaptive equalizer taps of the twin system are then reset to values which would have resulted had there been no IS (i.e., the taps of the original, unbiased system). A new optimal IS biasing value for the next vector of samples is calculated, and the process (a “biasing cycle”) is repeated. Fig. 2 shows the block diagram of our simulation approach.

It is important to note that, for the *instantaneous* TS estimator to be statistically unbiased (i.e., $E\{\hat{P}_{e,TS}^*(k)\} = P_e(k)$), the biased signal $\mathbf{X}^*(k)$ must be allowed to go through the process of adaptation, as suggested by the form of \mathbf{h}_{eq}^* .

During the short time interval from $k - \hat{M}$ to k , the adaptive equalizer does not react

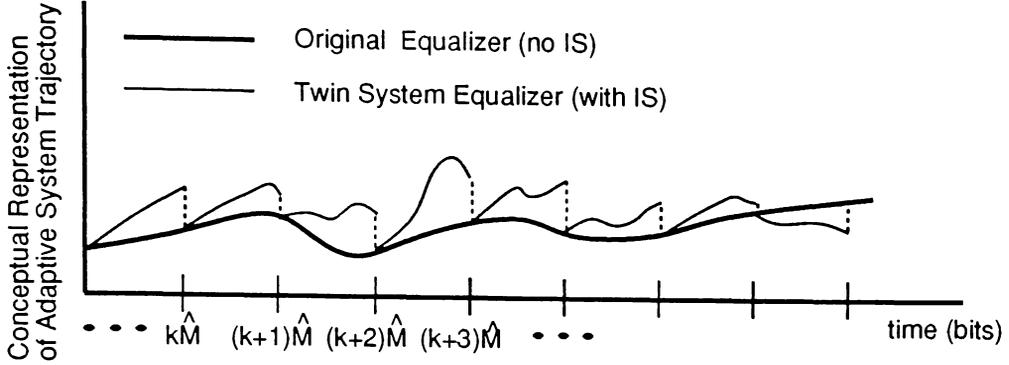


Figure 3: Conceptual representation of the adaptive system trajectory for the original, unbiased system (bold), and the equalizer in the twin system. The twin system equalizer adapts to the biased noise samples during a biasing cycle. At the beginning of a new biasing cycle, the equalizer in the twin system is then reset using tap values which would have resulted from unbiased noise samples.

properly (as our results will also show) in order to adapt to the biased signal $\mathbf{X}^*(k)$ allowing us to introduce more artificial errors into the system. Therefore, the TS estimator can offer significant speed-up factors over the conventional MC estimator, as is also indicated in our empirical results. Fig. 3 illustrates the TS approach by showing a conceptual system trajectory. Each \hat{M} bits is denoted here as a “biasing cycle”. The bold curve in Fig. 3 represents the adaptive equalizer trajectory in the original, unbiased system. This is only a *conceptual* plot since the equalizer trajectory is multidimensional. The plot is useful, however, in understanding the basic idea behind the method. The thin curve denotes the trajectory of the twin system. In this system, biased noise samples are fed into the adaptive equalizer. An estimate of the BER is taken once each biasing cycle. The equalizer in the twin system is then “reset” to the trajectory which would have resulted had there been unbiased noise samples input (taken from the original system).

The variance of the *instantaneous* TS estimator is

$$\sigma_{\hat{P}_{e,TS}(k)}^2 = \int \sigma_{\hat{P}_{e,TS|\mathbf{X}_C(k)}(k)}^2 f_{\mathbf{X}_C(k)}(\mathbf{a}) d\mathbf{a} \quad (13)$$

where the conditional variance $\sigma_{\hat{P}_{e,TS|\mathbf{X}_C(k)}(k)}^2$ is given by

$$\sigma_{\hat{P}_{e,TS|\mathbf{X}_C(k)}(k)}^2 = \frac{1}{N^*} \int_{\{\Omega_{\mathbf{X},k}^*|\mathbf{X}_C(k)\}} f_{\mathbf{X}(k)}(\mathbf{b}) \left(w_{\mathbf{X}(k)}(\mathbf{b}) - P_{e|\mathbf{X}_C(k)}(k) \right) d\mathbf{b} \quad (14)$$

$P_{e|\mathbf{X}_C(k)}(k)$ is the probability of a detection error at time k *conditioned* on $\mathbf{X}_C(k)$, and

$$\begin{aligned} \{\Omega_{\mathbf{X},k}^*|\mathbf{X}_C(k)\} &= \{\mathbf{X}(k) : I(\mathbf{X}^*(k)\mathbf{h}_{eq}^{*T}(k)) = 1 | \mathbf{X}_C(k)\} \\ &= \{\mathbf{X}(k) : I(\mathbf{X}^*(k)\mathbf{h}_{eq}^{*T}(k)) = 1, \mathbf{h}_{eq}^*(k) = \mathbf{g}'(\mathbf{X}_C(k-L), \mathbf{X}^*(k-L))\} \end{aligned}$$

The variance of the TS estimator of (12) is given by (with N^* in (14) set to unity)

$$\sigma_{\hat{P}_{e,TS}^*}^2 = \frac{1}{(K^* - k_0)^2} \sum_{k=k_0+1}^{K^*} \sigma_{\hat{P}_{e,TS}^*(k\hat{M})}^2 + \frac{1}{(K^* - k_0)^2} \sum_{\substack{k=k_0+1 \\ k \neq l}}^{K^*} \sum_{l=k_0+1}^{K^*} \sigma_{\hat{P}_{e,TS}^*(k\hat{M})\hat{P}_{e,TS}^*(l\hat{M})} \quad (15)$$

where $\sigma_{\hat{P}_{e,TS}^*(k\hat{M})\hat{P}_{e,TS}^*(l\hat{M})}$ is the covariance of the estimates $\hat{P}_{e,TS}^*(k\hat{M})$ and $\hat{P}_{e,TS}^*(l\hat{M})$, $k \neq l$, which are found using (11) with $N^* = 1$. Our goal is to choose $f_{\mathbf{X}(k\hat{M}L)}^*$ for $k_0 < k \leq K^*$ such that the first sum of (15) is minimized. The effect this will have on the second double summation is not clear, but our experimental results show large run time speed-up factors for a static linear channel, and a slowly varying, random linear channel. The choice of $f_{\mathbf{X}(k)}^*$ determines the magnitude of the estimator variance and is, therefore, crucial for the resulting speed-up factor over conventional MC.

3.2 Alternative IS Based on the Independence Assumption

Before we focus on finding an $f_{\mathbf{X}(k)}^*$ that will minimize or nearly-minimize the estimator variance, we present an *alternative* approach to the TS method. This alternative is based on a widely used assumption in the adaptive filtering literature, namely that the random data vector $\mathbf{X}(k)$ and the random vector $\mathbf{h}_{eq}(k)$ are *statistically independent* [9]. Under this assumption, eq. (3) can be rewritten as

$$P_e(k) \approx P_{e,IA}(k) = \int I(\mathbf{a}\mathbf{b}^T) f_{\mathbf{X}(k)}^*(\mathbf{a}) w_{\mathbf{X}(k)}(\mathbf{a}) f_{\mathbf{X}_C(k)}(\mathbf{b}) d\mathbf{a} d\mathbf{b} \quad (16)$$

where

$$w_{\mathbf{X}(k)}(\mathbf{a}) = f_{\mathbf{X}(k)}(\mathbf{a}) / f_{\mathbf{X}(k)}^*(\mathbf{a})$$

This implies the following IS-based block simulation approach: Allow the trajectory of the original adaptive system to evolve *without biasing*. For each block of \hat{M} bits (i.e., $\hat{M} \times L$ samples) make an IS-based decision by observing a *biased* output $\mathbf{X}^*(k)\mathbf{h}_{eq}^T(k)$, where $\mathbf{X}^*(k)$ is drawn *independently* from the $\mathbf{X}(k)$ that has already been used for the simulation and, therefore, independently from $\mathbf{h}_{eq}(k)$. The corresponding *instantaneous* IS estimator (called here the *instantaneous* IA estimator) is given by

$$\hat{P}_{e,IA}^*(k) = \frac{1}{N^*} \sum_{i=1}^{N^*} I(\mathbf{X}^*(k,i)\mathbf{h}_{eq}^T(k,i)) w_{\mathbf{X}(k)}(\mathbf{X}^*(k,i)) \quad (17)$$

Similarly, the IA estimator equivalent to the TS estimator of (12) is given by

$$\hat{P}_{e,IA}^* = \frac{1}{K^* - k_0} \sum_{k=k_0+1}^{K^*} I(\mathbf{X}^*(k\hat{M}L)\mathbf{h}_{eq}^T(k\hat{M})) w_{\mathbf{X}(k\hat{M}L)}(\mathbf{X}^*(k\hat{M}L)) \quad (18)$$

The potential advantage of this approach with respect to the TS method presented in Section 3.1 is that the IA method is much simpler to implement. Another advantage of this

approach is that, as we will discuss in the sequel, an *optimal* biasing setting is readily available. In general, however, the IA estimator is a statistically biased estimator of $P_e(k)$. The amount of statistical bias is dependent on how well the independence assumption describes the specific system model at hand. Our empirical results indicate that, most of the time the IA estimator yields results that are acceptably close to the TS estimator. In practice, it is left to the experimenter to choose which one to use, based on the trade-off between statistical bias and simplicity.

The preceding development of the IA method is based on the assumption that $\mathbf{X}(k)$ and $\mathbf{h}_{eq}(k)$ be statistically independent. The IA method, as it is actually implemented, only requires the weaker condition that the *noise* component $\mathbf{N}(k)$ of $\mathbf{X}(k)$ are statistically independent of $\mathbf{h}_{eq}(k)$.

In the application section, we will compare this alternative IA estimator with the TS estimator and present empirical data on its statistical accuracy and relative statistical bias.

3.3 Optimal IS Biasing for TS and IA Estimators

As was stressed earlier, the choice of $f_{\mathbf{X}(k)}^*$ determines the magnitude of the estimator variance and the resulting speed-up factor over conventional MC. We restrict our attention to the *translation* biasing scheme, which has demonstrated the best improvement in the past for similar (although not time-varying or adaptive) models [6, 10, 7, 8]. In the translation scheme, we define the biased random vector as $\mathbf{X}^*(k) = \mathbf{X}(k) + \mathbf{c}(k)$. We call $\mathbf{c}(k)$ the translation vector at the k th instant. The biased pdf is then simply $f_{\mathbf{X}(k)}^*(\mathbf{X}) = f_{\mathbf{X}(k)}(\mathbf{X} - \mathbf{c}(k))$.

Our goal is to choose a translation vector $\mathbf{c}(k)$ at each decision instant k such that the conditional variance given by (14), and therefore the estimator variance in (13) (and the first summation in (15)) are minimized. For non-adaptive systems, and for the case corresponding to i.i.d. Gaussian inputs to a static, linear receiver filter, with impulse response \mathbf{h} , it was implied in [6] and proved in [10] that the optimal translation is in the direction of the impulse response \mathbf{h} , i.e., it is best to set $\mathbf{c}(k) = C(k)\mathbf{h}$. Thus, the optimal translation vector at instant k is given by $\mathbf{c}_{opt}(k) = C_{opt}(k)\mathbf{h}_{eq}(k)$ where $C_{opt}(k)$ is found by differentiating the estimator variance expression with respect to $C(k)$ and setting the result to zero. The optimal value C_{opt} under these conditions, was also found in [6].

In our case, the “input” vector $\mathbf{X}(k)$ is still Gaussian and uncorrelated, but in general, $\mathbf{X}(k)$ will have a *covariance matrix*, $\Sigma_X(k)$. In order to apply IS to this $\mathbf{X}(k)$ we need to extend the results in [6, 10] appropriately. We do this in the Appendix, where we show that the optimal translation vector \mathbf{c}_{opt} is then

$$\mathbf{c}_{opt}(k) = C_{opt}(k)\Sigma_X(k)\mathbf{h}(k) \quad (19)$$

and furthermore, we obtain the optimal translation amount $C_{opt}(k)$.

Another important point for the application of the above result to a *time-varying* (but still non-adaptive) system is that *the optimal translation vector needs to be re-evaluated for every block of data*, because the channel impulse response, the mean vector, and the covariance matrix are all potentially changing with time.

Finally, for our adaptive system model, the above $\mathbf{c}_{opt}(k)$ will no longer be optimal unless the equivalent system response $\mathbf{h}_{eq}^*(k)$ is statistically *independent* of the input vector $\mathbf{X}^*(k)$ at time k . The implications for our techniques are the following: All TS estimator is statistically unbiased, but the above $\mathbf{c}_{opt}(k)$ is optimal only to the extent that the independence assumption is true. For cases where that assumption is not true, no strictly optimal IS translation setting is available right now. On the other hand, the IA estimator, although statistically biased, immediately admits this $\mathbf{c}_{opt}(k)$ as optimal with no complications.

Our empirical observations indicate that either estimator can be used with very significant speed-up factors over conventional MC.

4 Applications and Experimental Results

4.1 Set-Up

In order to verify the effectiveness of applying the IS scheme to adaptive systems, several simulation experiments were performed. The technique was applied to a system with a 6-tap LMS adaptive equalizer. The step size of the algorithm was found for each experiment by estimating the power of the signal at the input of the equalizer.

Two channels were considered, a static linear channel, and a randomly time-varying channel. The static channel had a normalized impulse response $v(l) = (\alpha)^l / \sum_{l=0}^{M_1 L - 1} v^2(l)$, $l = 0, \dots, M_1 L - 1$ with $\alpha < 1$ a user-selected parameter.

The time-varying channel consisted of a uniformly distributed multiplicative factor followed by a static linear channel similar to the above:

$$s(l) = v(l) * a(l)r(l)$$

where $r(l)$ was uniformly distributed, centered at 1.0, with a total spread of 0.6, and $g(\cdot)$ the same as above. Our artificial random “fading” was made *slow* by holding the same value of $r(l)$ for the entire duration of a bit. Fig. 4 shows the impulse response of the static linear channel used and plots of the equivalent impulse response $g(k, t)$ of the time-varying channel at three different time instants, $t = 18, 36$, and 72 . In both types of channels, a memory of two bits ($M_1 = 2$) was considered.

Baseband binary signaling was used in the examples, with levels of $-A$ and $+A$ corresponding to 0 and 1 data values respectively, with a signal level $A = 5$. The Gaussian noise was zero mean with variance $\sigma^2 = 0.5$. The receiver filter was a one bit ($M_2 = 1$) matched filter. As stated earlier, the equalizer memory length was $M = 6$. Throughout our experiments we used $L = 9$ samples per bit. The signal-to-noise ratio (SNR) for the example systems was $SNR \approx 27$ dB. In each experiment, the adaptive equalizer was initialized to zero, then trained for a short period of time (100 ~ 200 bits).

In the twin system experiments, $\hat{P}_{e,TS}^*$ were obtained according to (12) for each channel configuration. Decisions were spaced \hat{M} bits (where \hat{M} is the memory of the equivalent system \mathbf{h}_{eq}). The noise vector $\mathbf{N}(k)$ was biased along the direction of the equivalent impulse response $\mathbf{h}_{eq}(k)$, since for additive white gaussian noise, the covariance matrix $\Sigma_X(k)$ is

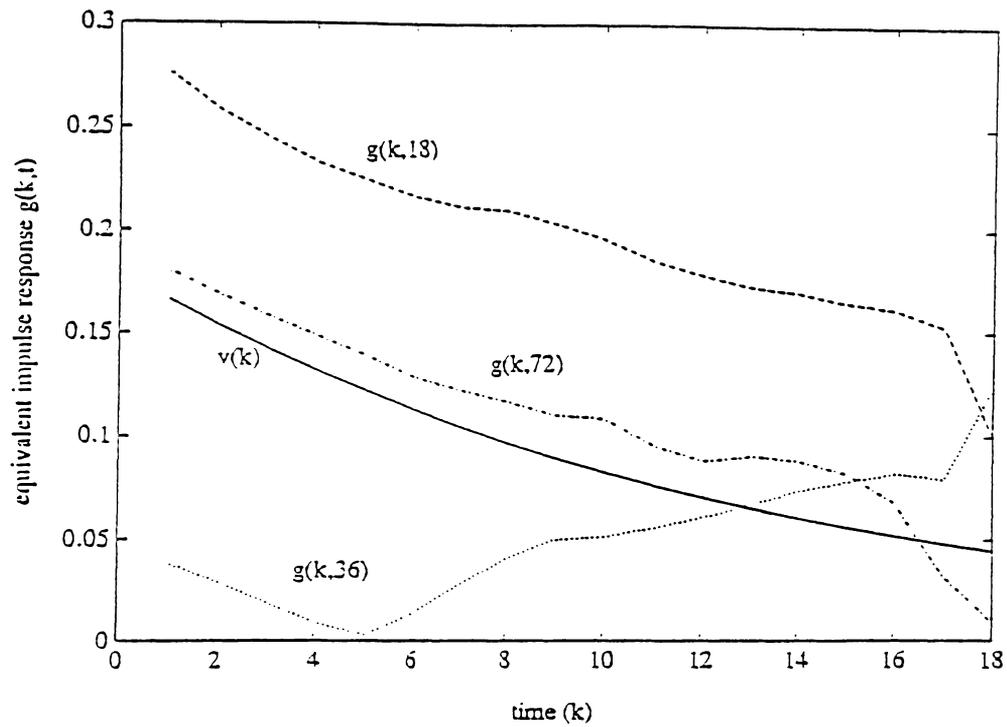


Figure 4: Plot of the time-varying impulse response of the channel for four different values of time. The original impulse response is given by $v(k)$ (solid).

System	n_d	$\hat{P}_{e,TS}^*$	$\hat{\sigma}_{\hat{P}_{e,TS}^*}^2$	95% Confidence	Speed-up
Static $\alpha = 0.94$	833	1.6×10^{-5}	4.5×10^{-11}	$(1.4 \times 10^{-5},$ $1.7 \times 10^{-5})$	3.8×10^2
Static $\alpha = 0.928$	1300	2.7×10^{-8}	1.2×10^{-16}	$(2.4 \times 10^{-8},$ $3.0 \times 10^{-8})$	1.5×10^5
Static $\alpha = 0.92$	1500	9.7×10^{-11}	1.7×10^{-21}	$(8.6 \times 10^{-11},$ $1.1 \times 10^{-10})$	3.4×10^7
Time-Varying $\alpha = 0.925$	833	1.0×10^{-5}	8.9×10^{-11}	$(7.4 \times 10^{-6},$ $1.3 \times 10^{-5})$	10^2
Time-Varying $\alpha = 0.9$	1500	2.7×10^{-9}	3.7×10^{-17}	$(1.0 \times 10^{-9},$ $4.4 \times 10^{-9})$	1.82×10^4

Table 1: Simulation data and speed-up factors for static and time-varying systems with an adaptive equalizer. Importance sampling based on the twin system approach was used in the simulations.

simply diagonal. Estimates of $\hat{P}_{e,IA}^*$ using the independence assumption were obtained during the same experiments. These estimates, as mentioned earlier, did not require the twin equalizer to adapt. Random vectors $\mathbf{N}^*(k)$ for this estimator were drawn independently of the $\mathbf{N}(k)$, from the biased (translated) pdf.

4.2 Results

Tables 1 and 2 summarize our experimental observations. In calculating confidence coefficients and speed-up factors we replaced the “true” BER (which was unknown) by the TS estimate. Between 833 and 1500 decisions per run (n_d), with 50 runs were made for each system in order to obtain the sample variance, $\hat{\sigma}^2$.

Notice the exceptional speed-up factors over conventional MC, which become unique when recalling that this is the first application of IS to the simulation of adaptive systems. It is also remarkable that these speed-up factors were obtained with a suboptimal biasing procedure Recall, however, that the TS estimators are theoretically statistically unbiased. Thus, the suboptimality of the IS parameter settings used here only affected the speedup-factor. The relative statistical bias of the IA estimator is minimal to modest and suggests that this estimator deserves more study.

The measured overhead of the twin system technique (twin system implementation, optimal translation calculations etc.) was modest, reducing the net speed-up by roughly a factor of 2 ($\approx 40/72$) for a 6-tap equalizer, and a factor of approximately 3.5 ($\approx 41/145$) for a 10-tap equalizer.

System	n_d	$\hat{P}_{e,IA}^*$	$\hat{\sigma}^2_{\hat{P}_{e,IA}^*}$	Rel. Bias, IA	95% Confidence	Speedup
Static $\alpha = 0.94$	833	5.6×10^{-6}	8.9×10^{-12}	0.65	$(4.8 \times 10^{-6}, 6.4 \times 10^{-6})$	6×10^2
Static $\alpha = 0.928$	1300	7.3×10^{-9}	3.1×10^{-17}	0.73	$(5.8 \times 10^{-9}, 8.8 \times 10^{-9})$	1.4×10^5
Static $\alpha = 0.92$	1500	1.3×10^{-11}	1.1×10^{-22}	0.87	$(1.0 \times 10^{-11}, 1.6 \times 10^{-11})$	6.0×10^7
Time-Varying $\alpha = 0.925$	833	3.0×10^{-6}	1.5×10^{-11}	0.7	$(1.9 \times 10^{-6}, 4.1 \times 10^{-6})$	1.4×10^2
Time-Varying $\alpha = 0.9$	1500	6.8×10^{-11}	5.0×10^{-20}	0.97	$(6.0 \times 10^{-12}, 1.3 \times 10^{-10})$	8.0×10^4

Table 2: Simulation data and speed-up factors for static and time-varying systems with an adaptive equalizer. Importance sampling based on the independence assumption approach was used in the simulations.

5 Conclusions

We have presented, for the first time, two IS methodologies for MC simulation of communication links characterized by time-varying channels and adaptive equalizers. In the TS method, biased noise samples are input to the adaptive equalizer, but the equalizer is only allowed to adapt to these samples for a time interval equal to the memory of the system. The TS method results in a statistically unbiased estimator, and circumvents the problems associated with system memory which increases with time in the simulation.

In addition to the TS technique, we also present a statistically biased, but simpler technique for using IS with adaptive equalizers which is based on the independence assumption between the equalizer input and the equalizer taps (the “IA” technique).

Experimental results show run time speed-up factors of two to seven orders of magnitude for a static linear channel with memory, and of two to almost five orders of magnitude for a slowly varying, random linear channel with memory for both the IA and TS methods.

Acknowledgement

The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

- [1] S. S. Haykin. *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1986.
- [2] S. U. H. Qureshi. Adaptive Equalization. *Proc. IEEE*, 73:1349–1387, Sep. 1985.

- [3] P. Balaban. Statistical Evaluation of the Error Rate of the Fiberguide Repeater Using Importance Sampling. *Bell Sys. Tech. J.*, 55(6):745–766, Jul.-Aug. 1976.
- [4] K. S. Shanmugan and P. Balaban. A Modified Monte-Carlo Simulation Technique for the Evaluation of Error Rate in Digital Communication Systems. *IEEE Trans. Commun.*, COM-28(11):1916–1924, Nov. 1980.
- [5] M. C. Jeruchim. Techniques for Estimating the Bit Error Rate in the Simulation of Digital Communication Systems. *IEEE J. Select. Areas Commun.*, SAC-2(1):153–170, Jan. 1984.
- [6] D. Lu and K. Yao. Improved Importance Sampling Technique for Efficient Simulation of Digital Communication Systems. *IEEE J. Select. Areas Commun.*, 6(1), Jan. 1988.
- [7] M. Devetsikiotis and J. K. Townsend. A Useful and General Technique for Improving the Efficiency of Monte Carlo Simulation of Digital Communication Systems. In *Proc. of IEEE GLOBECOM '90*, San Diego, CA, 1990.
- [8] M. Devetsikiotis and J. K. Townsend. An Algorithmic Approach to the Optimization of Importance Sampling Parameters in Digital Communication System Simulation. To appear in the *IEEE Trans. Commun.*
- [9] N. J. Bershad and L. Z. Qu. On the Probability Density Function of the LMS Adaptive Filter Weights. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 37:43–56, Jan. 1989.
- [10] R. J. Wolfe, M. C. Jeruchim, and P. M. Hahn. On Optimum and Suboptimum Biasing Procedures for Importance Sampling in Communication Simulation. *IEEE Trans. Commun.*, COM-38(5):639–647, May 1990.

Appendix

We present here an extension to the findings in [6] and [10]. Both the *optimal translation direction* and the *optimal translation amount* are found for the case of *correlated, non-identically distributed* Gaussian inputs to a linear system.

Specifically, suppose that the M -dimensional vector \mathbf{x} is Gaussian distributed, with mean \mathbf{m} and covariance matrix $\Sigma_{\mathbf{x}}$ (note that in this Appendix vectors should be taken as *column* vectors). Let $\hat{P}_e^*(\mathbf{h}, \mathbf{m}, \Sigma_{\mathbf{x}}, \mathbf{c})$ be the IS estimator

$$\hat{P}_e^*(\mathbf{h}, \mathbf{m}, \Sigma_{\mathbf{x}}, \mathbf{c}) = \frac{1}{N} \sum_{i=1}^N I_{T_o}(\mathbf{h}^T \mathbf{x}_i^*) w_{\mathbf{x}}(\mathbf{x}_i^*)$$

of the probability

$$P_e(\mathbf{h}, \mathbf{m}, \Sigma_{\mathbf{x}}) = P\{\mathbf{h}^T \mathbf{x} > T_o \mid \mathbf{h}, \mathbf{m}, \Sigma_{\mathbf{x}}\}$$

conditioned on \mathbf{h} , \mathbf{m} and $\Sigma_{\mathbf{x}}$, where $I_{T_o}(y) = 1$ when $y > T_o$, and $I_{T_o}(y) = 0$ otherwise, and the \mathbf{x}_i^* 's are drawn from the Gaussian joint density of the \mathbf{x} 's translated by \mathbf{c} .

Our goal is to find a translation vector \mathbf{c} , such that the estimator variance is *minimized*. Under the given assumptions, the estimator variance is

$$\begin{aligned}\sigma_{IS}^2(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) &= E(\hat{P}_e^{*2}(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c})) - P_e^2(\mathbf{h}, \mathbf{m}, \Sigma_X) \\ &= \int_{\mathbf{h}^T \mathbf{x} > T_O} \left(\frac{1}{\sqrt{2\pi}} \right)^M |\Sigma_X^{-1}|^{1/2} \frac{A^2(\mathbf{x})}{B(\mathbf{x}, \mathbf{c})} d\mathbf{x} - P_e^2(\mathbf{h}, \mathbf{m}, \Sigma_X)\end{aligned}$$

where

$$A(\mathbf{x}) = \exp \left\{ -\frac{\mathbf{x}^T \Sigma_X^{-1} \mathbf{x}}{2} \right\}$$

and

$$B(\mathbf{x}, \mathbf{c}) = \exp \left\{ -\frac{(\mathbf{x} - \mathbf{c})^T \Sigma_X^{-1} (\mathbf{x} - \mathbf{c})}{2} \right\}$$

Minimizing $\sigma_{IS}^2(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c})$ is equivalent to minimizing $V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) \equiv E(\hat{P}_e^{*2}(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}))$, with respect to \mathbf{c} . It follows after some algebraic manipulation that

$$V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) = \exp \left\{ \mathbf{c}^T \Sigma_X^{-1} \mathbf{c} \right\} Q \left(\frac{T_O - \mathbf{h}^T \mathbf{m} + \mathbf{h}^T \mathbf{c}}{(\mathbf{h}^T \Sigma_X \mathbf{h})^{1/2}} \right)$$

Let $C_O \equiv \mathbf{h}^T \mathbf{c}$ be the effective translation at the *output*, and $\sigma^2 \equiv \mathbf{h}^T \Sigma_X \mathbf{h}$ be the output signal variance.

Optimal direction: Let \mathbf{T} be a non-singular linear transformation and

$$\mathbf{x}' = \mathbf{T}\mathbf{x}$$

$$\mathbf{m}' = \mathbf{T}\mathbf{m}$$

$$\Sigma_X' = \mathbf{T}\Sigma_X\mathbf{T}^T$$

Then $V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) = V(\mathbf{h}', \mathbf{m}', \Sigma_X', \mathbf{c}')$ when $\mathbf{h}' = (\mathbf{T}^{-1})^T \mathbf{h}$ and $\mathbf{c}' = \mathbf{T}\mathbf{c}$. Let \mathbf{I}_M denote the identity matrix of dimension $M \times M$. Choose \mathbf{T} such that $\Sigma_X' = \sigma_1^2 \mathbf{I}_M$, for some σ_1^2 , i.e., \mathbf{T} diagonalizes and normalizes Σ_X . Such a transformation always exists because Σ_X is *symmetric* and *positive definite*.

From [10] we know that, when $\Sigma_X' = \sigma_1^2 \mathbf{I}_M$, $V(\mathbf{h}', \mathbf{m}', \Sigma_X', \mathbf{c}')$ is minimized along the direction \mathbf{h}' , i.e., $\mathbf{c}' = C'_{opt} \mathbf{h}'$. It follows that for all \mathbf{c} ,

$$\begin{aligned}V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) &= V(\mathbf{h}', \mathbf{m}', \Sigma_X', \mathbf{c}') \\ &\geq V(\mathbf{h}', \mathbf{m}', \Sigma_X', \mathbf{c}'_{opt}) \\ &= V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{T}^{-1} \mathbf{c}'_{opt}) \\ &= V(\mathbf{h}, \mathbf{m}, \Sigma_X, C'_{opt} \mathbf{T}^{-1} \mathbf{h}')\end{aligned}$$

Therefore $V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c})$ is minimized for $\mathbf{c}_{opt} = C'_{opt} \mathbf{T}^{-1} \mathbf{h}' = C'_{opt} \mathbf{T}^{-1} (\mathbf{T}^{-1})^T \mathbf{h} = C'_{opt} / \sigma_1^2 \Sigma_X \mathbf{h}$, since $\mathbf{T}\Sigma_X\mathbf{T}^T = \sigma_1^2 \mathbf{I}_M$ implies $\Sigma_X / \sigma_1^2 = \mathbf{T}^{-1} (\mathbf{T}^{-1})^T$. We conclude that $V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c})$ is minimized for $\mathbf{c}_{opt} = C_{opt} \Sigma_X \mathbf{h}$, for some C_{opt} .

Optimal translation amount: When $\mathbf{c} = C\Sigma_X\mathbf{h}$, it follows that $\mathbf{c}^T\Sigma_X^{-1}\mathbf{c} = C_O/\sigma^2$, since

$$\mathbf{c}^T\Sigma_X^{-1}\mathbf{c} = C^2\mathbf{h}^T\Sigma_X^T\Sigma_X^{-1}\Sigma_X\mathbf{h} = C^2\mathbf{h}^T\Sigma_X^T\mathbf{h} = C^2\sigma^2$$

and

$$C_O^2 = (\mathbf{h}^T\mathbf{c})^2 = C^2(\mathbf{h}^T\Sigma_X\mathbf{h})^2 = C^2\sigma^4$$

Therefore, when $\mathbf{c} = C\Sigma_X\mathbf{h}$, the estimator mean square value can be written as a function of a *scalar* quantity C_O :

$$V(\mathbf{h}, \mathbf{m}, \Sigma_X, \mathbf{c}) = V'(\mathbf{h}, \mathbf{m}, \Sigma_X, C_O) = \exp(C_O) Q\left(\frac{T_O - \mathbf{h}^T\mathbf{m} + C_O}{\sigma}\right)$$

which is the same mean square term minimized in [6]. Using similar manipulations (and the common $Q(\cdot)$ approximation when its argument is greater than 4), one finds

$$C_{O,opt}^2 = (T_O - \mathbf{h}^T\mathbf{m})^2 + \sigma^2$$

and

$$C_{opt}^2 = \frac{C_{O,opt}^2}{\sigma^4} = \frac{(T_O - \mathbf{h}^T\mathbf{m})^2 + \sigma^2}{\sigma^4}$$