An electrical-level superposed-edge approach to statistical serial link simulation

An Electrical-Level Superposed-Edge Approach to Statistical Serial Link Simulation

Michael Tsuk
Ansoft LLC
67 S. Bedford St., Suite 202E
Burlington, MA 01803
mtsuk@ansoft.com

Daniel Dvorscak
Ansoft LLC
67 S. Bedford St., Suite 202E
Burlington, MA 01803
ddvorscak@ansoft.com

Chin Siong Ong
Ansoft LLC
67 S. Bedford St., Suite 202E
Burlington, MA 01803
csong@ansoft.com

Jacob White
MIT
Cambridge, MA 02139
white@mit.edu

ABSTRACT

Brute-force simulation approaches to estimating serial-link bit-error rates (BERs) become computationally intractable for the case when BERs are low and the interconnect electrical response is slow enough to generate intersymbol interference that spans dozens of bit periods. Electrical-level statistical simulation approaches based on superposing pulse responses were developed to address this problem, but such pulse-based methods have difficulty analyzing jitter and rise/fall asymmetry. In this paper we present a superposing-edge approach for statistical simulation, as edge-based methods handle rise/fall asymmetry and jitter in straightforward way. We also resolve a key problem in using edge-based approaches, that edges are always correlated, by deriving an efficient inductive approach for propagating the edge correlations. Examples are presented demonstrating the edge-based method's accuracy and effectiveness in analyzing combinations of uniform, Gaussian, and periodic distributed random jitter.

Categories and Subject Descriptors

4.5 [**Interconnect and Power Networks**]: Reliability Analysis

Keywords

Interconnect, Eye Diagrams, Circuit Simulation, Bit-Error Rate, Serial Link

1. INTRODUCTION

Over the past decade there has been a revolution in the field of signal integrity engineering due to the pervasive use of high-speed serial links as the means of moving large amounts of data inside and between computing systems. This revolution has been driven by the declining cost of transmitter and receiver circuitry relative to the cost of the interconnect media. Basically, inexpensive yet sophisticated transmit and receive circuitry can compensate for poor "wiring" performance [2, 3].

The slow response of limited-bandwidth interconnect implies that for transmit and receive circuitry to achieve gigabit-per-second data transfer rates (corresponding to bit periods on the order of a nanosecond), such circuitry must manage intersymbol interference (ISI) that can span dozens of bit periods. In addition, transmitters and receivers must also be designed to be insensitive to random fluctuations in bit start times, also known as transmit jitter [6, 10]. Finally, target bit-error rates (BERs) are extremely low, on the order 10^{-12} (one error in 10^{12} bits). These three issues combine to make brute-force simulation an impractical tool for estimating BERs, and led to the development of statistical methods [4, 5, 7].

The most widely used statistical alternative to brute-force simulation is based on superposing time-shifted pulse responses [4], an approach that has proven to be quite effective. The advantage of pulse-based approaches is that they can take advantage of the fact transmitted bits are uncorrelated, unless the bits are encoded, but the disadvantages of pulse-based approaches is the difficulty modeling edge-based phenomenon such as transmit jitter and asymmetries in rise and fall transitions.

In this paper we describe a new edge-based statistical approach for estimating BERs, one that easily handles rise/fall asymmetry and jitter. Our key result is the derivation of an efficient inductive algorithm for dealing the fact that edges are innately correlated, even when bits are uncorrelated. In next section we present background on the serial analysis problem and describe pulse-based methods, and in Section 3 we present a fairly formal probabilistic derivation of a novel inductive approach for handling edge correlation. In Section 4 we show how to include transmit jitter in the edge-based method, and in Section 5 we present examples that validate the approach and demonstrate its abilities to analyze problems with substantial ISI and a variety of jitter

ICCAD'09, November 2–5, 2009, San Jose, California, USA.

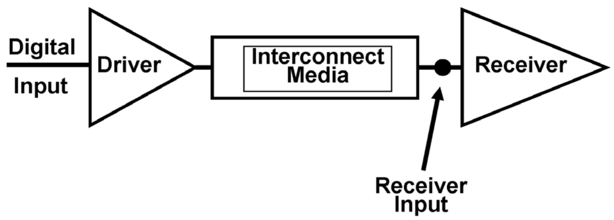

Figure 1: A transmission system diagram.

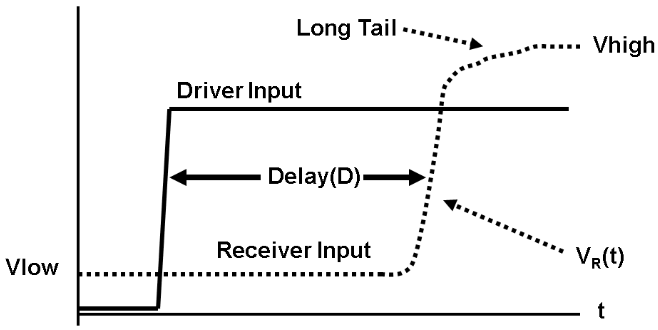

Figure 2: A drawing of a typical rising edge response with features noted.

distributions (Gaussian, uniform and periodic).

2. BACKGROUND AND DEFINITIONS

A high-speed serial link is divided into three parts: the transmitter, the interconnect, and the receiver, as diagrammed in Figure 1. The interconnect is almost always linear and time-invariant (LTI), and is usually represented by electromagnetic scattering parameters or as a collection of circuit elements. The transmitter and receiver are typically represented as detailed transistor-level subcircuits, though higher-level behavioral representations are also used. In algorithms described below, the transition behavior of the transmitter along the filtering effect of the interconnect and receiver input are all lumped together, and used to generate the rising $v_R(t)$ and falling $v_F(t)$ edge responses as seen at the input to the receiver (see Figure 1). In the descriptions below, we will make use of many features of edge responses, (an example of a rising edge response is given in Figure 2.) Note from Figure 2 that we define the delay, D, as the time between when the driver digital input transitions and the receiver input is fifty-percent of the way to its final value. We will also define v_{high} and v_{low}, the steady-state high and low values at the receiver input, also noted in Figure 2.

One of the most important features of these edge responses in typical high-speed serial links is that they have relatively long tails (again see Figure 2), typically the consequence of skin-effect or dielectric loss in the interconnect. These long tails are the cause of intersymbol interference, or ISI, a fundamental source of bit errors. Another important source of errors in these channels is jitter. Transmit jitter is the variation in the transition times of the digital input. Transmit jitter comes in many forms, all of which can be handled by statistical techniques. The form that will be considered in this paper is random jitter, where each transition is displaced in time from its nominal location by an amount given by a random variable with zero mean and a probability density function (such as Gaussian, periodic, uniform, or some combination).

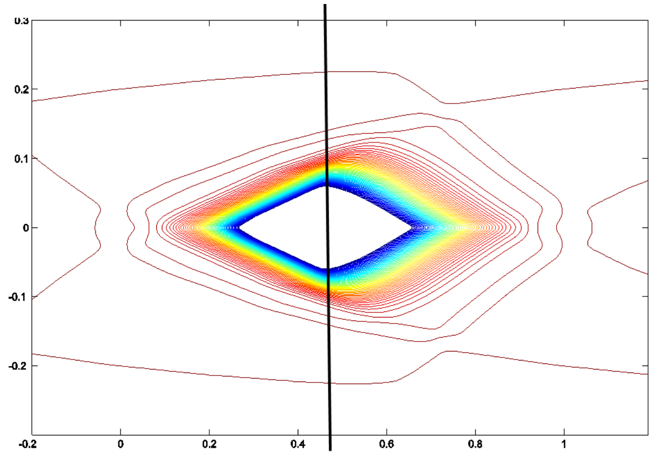

Figure 3: A sample eye diagram with dark line at likely sample time τ.

2.1 Eye diagrams

An eye diagram is a method for examining the fitness of a serial link. Originally, the eye diagram was a measurement tool: the receiver input waveform for a link was plotted on an oscilloscope in persistence mode, triggering every bit period. In order for the receiver to be able to distinguish low bits from high ones, there needed to be an open area in the diagram where no waveforms crossed. With the usual types of low-pass channels, this open area resembled a human eye, hence the name.

One can generate a similar diagram using transient simulation: the transient waveform for a long sequence of bits is simply shifted and overlaid. However, the need to analyze extremely low bit-error-rates for modern high-speed serial links makes transient simulation impractical; bit sequences of lengths in the trillions are needed to generate an eye diagram with sufficient probability resolution to accurately assess low BERs. Instead, statistical approaches are used to generate a probability density function (PDF) for the voltage distribution at each time τ. The voltage at time τ is a continuous random variable, denoted V_τ, with PDF given by $f_{V_\tau}(v)$ (following the notation in [1]). In Figure 3 we show an eye diagram with a line at time τ, and in Figure 4 we show a typical probability density function with its typical bimodal behavior.

Bit-error rates can be computed by integrating the areas under the PDFs that correspond to incorrect decisions of received bits. In the simplest version, those are the areas under the PDF below a certain threshold if the bit is supposed to be high, and above that threshold if the bit is supposed to be low.

2.2 Pulse-based Statistical Approach

One widely used, and extremely efficient, statistical approach is to represent $v(\tau)$ as a superposition of pulse responses. To more precisely define a pulse response, consider a typical serial link. If the digital input to the serial link

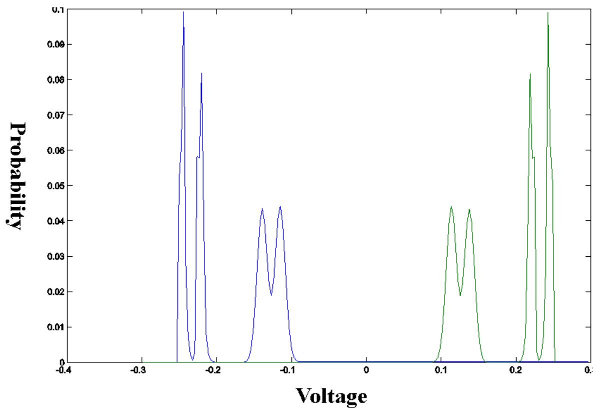

Figure 4: Probability density function along line in eye diagram, showing bimodal behavior.

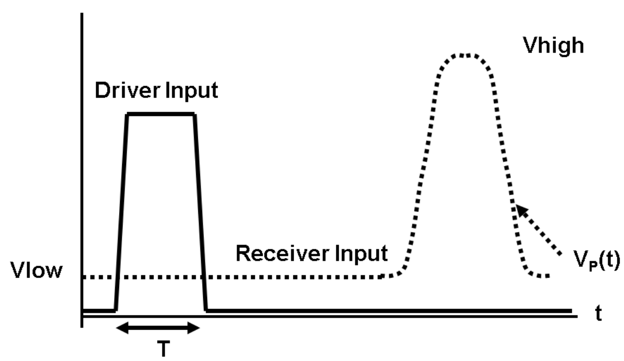

Figure 5: A drawing of a typical pulse response.

driver corresponds to sending a single one bit surrounded by zero bits, then the voltage at the receiver input will eventually rise from its steady-state low value, denoted v_{low}, may stabilize briefly at at its steady-state high value, denoted v_{high}, and then eventually fall back down to v_{low}. An example pulse response is shown in Figure 5, where note that T, shown in the figure, is the period of time that the digital input is high when transmitting a single "one" bit. We also refer to T as the *bit period*.

Using the pulse superposition approach, the formula for the receiver voltage at time τ is given by

$$v(\tau) \approx \sum_{k=0}^{N} p_k v_p(D + kT + \tau) \tag{1}$$

where $v_p(t)$ is the pulse response, D is the interconnect delay (time from driver input transition to receiver input fifty percent rise), and p_k is equal to the value of the k^{th} bit (either zero or one).

If the transmitted bits have not been encoded, then the individual bits are uncorrelated (the case of bit correlation due to encoding was considered in [8]). Assuming that bits are uncorrelated and that pulse responses can be superposed leads to a particularly simple formula for the PDF needed for bit-error rate estimation,

$$f_{V_\tau}(v) = *\prod_{k=0}^{N} \frac{1}{2}\left(\delta(v) + \delta\left(v - v_p(\tau_k)\right)\right) \tag{2}$$

where $f_{V_\tau}(v)$ is the PDF of V_τ, τ_k is defined as

$$\tau_k \equiv D + kT + \tau \tag{3}$$

and is used to simplify (2) as well as subsequent equations, N is the number of bit periods needed to capture all the intersymbol interference, and therefore $v_p(\tau_L) \approx 0$ for $L > N$. Finally, in a slight abuse of common notation, we use $*\prod_{k=0}^{N}$ to indicate that N convolutions of $N+1$ terms are performed to compute $f_{V_\tau}(v)$.

Each convolution in (2) involves one term with two impulses, and therefore the cost of directly computing the N convolutions grows as 2^N. This high computational cost is easily avoided by first discretizing the entire range of possible receiver input voltages on to a grid of M uniformly spaced voltage sample points, effectively converting the PDF, $f_{V_\tau}(v)$, to a probability mass function, $p_{V_\tau}(v_j), \quad j \in \{0, ..., M-1\}$. Given the discretization of the receiver voltages, the N convolutions in (2) become discrete convolutions, and can be computed in M^2N operations. This cost can be further reduced to $NM \log M$ by using the Fast Fourier Transform (FFT).

The pulse-based approach has proved to be enormously effective [4], but there are two significant difficulties with the pulse-based statistical approach. First, if the rise and fall waveforms are not perfect mirror images of each other, superposition of neighboring pulses fails to generate a steady high or low signal. Second, it is difficult to model transmit jitter. Simply jittering the pulses creates unphysical correlation between the jitter of neighboring edges, and in the case of the steady high or steady low signal mentioned above, jittering the pulses can create unphysical glitches in the resulting waveform.

3. EDGE-BASED STATISTICAL APPROACH

If the transmit circuitry settles in much less than a bit period, and if the interconnect and the loading effects of the receiver are nearly linear, then the voltage at the receiver input can be accurately approximated by superposing what we will refer to as edge responses. As noted above, superposing responses can be an extremely effective approach to computing the PDFs of voltages needed to estimate bit-error rates, and superposing pulse responses is appealing because these responses are uncorrelated in the unencoded case. We instead consider superposing edge responses because of the modeling limitations of superposing pulse responses, and must address the problem that edge responses are correlated regardless of whether or not the transmitted bits have been encoded.

3.1 Edge Responses

To more precisely define an edge response, consider again a typical serial link. If the digital input to the serial link driver makes a single transition from zero to one, then the voltage at the receiver input will eventually rise from v_{low} to v_{high}. We refer to the receiver input waveform as the rising edge response, and denote it as $v_R(t)$. Equivalently, if the driver input makes a single transition from one to zero, then the receiver input waveform will eventually fall from v_{high} to v_{low}, and this falling edge response is denoted by $v_F(t)$. If edge superposition is used to compute the receiver input voltage due to a complicated pattern of digital bits at the input, then

$$v(\tau) \approx \sum_{k=0}^{N} \alpha_k v_R(\tau_k) + \beta_k v_F(\tau_k) \tag{4}$$

where $v(\tau)$ is the receiver input voltage at time τ, k is a bit index, τ_k is as defined in (3), N is the maximum number of bit periods over which there is intersymbol interference, and α_k and β_k take on the values of either zero or one, depending on the bit pattern. Here $k = 0$ corresponds to the current bit being received, and higher values of k correspond to bits that were transmitted further in the past. The coefficient $\alpha_k = 1$ when the k^{th} bit is one and the $k+1^{th}$ bit is zero, and $\alpha_k = 0$ otherwise. Analogously, $\beta_k = 1$ when the k^{th} bit is zero and the $k+1^{th}$ bit is one, and $\beta_k = 0$ otherwise.

When evaluating the voltage at the receiver input for a given driver input bit sequence, using (4) can have accuracy advantages over using (1). Unlike pulses, however, edges are correlated even when the associated bits are uncorrelated. For example, there must be a falling edge between any pair of rising edges. This correlation complicates using edge superposition to compute the PDF for receiver input, $f_{V_\tau}(v)$. It is the key insight in this paper, derived below, that there is an efficient inductive approach for computing $f_{V_\tau}(v)$ from edge responses. In the next subsection we will consider the jitter-free case, and then generalize our result in a subsequent subsection.

3.2 Inductive Approach

In order to develop an inductive approach that can be used to compute $f_{V_\tau}(v)$ using edge responses, it is helpful to define an intermediate random variable and a conditional PDF involving that intermediate variable. Let

$$V_\tau^k \tag{5}$$

denote the random variable associated with the receiver input voltage at time τ, under the constraint that there are no edges between the current received bit and the k^{th} previous bit. Alternatively, V_τ^k can be thought of as the random variable whose values are receiver input voltages generated by a restricted set of bit patterns, ones where bits $0, 1, ..., k$ all have the same value. Note that random variable V_τ^0 could also be referred to as V_τ, as there are no bit constraints.

Let the conditional PDF

$$f_{V_\tau^k|B_k}(v|b) \tag{6}$$

denote the conditional PDF for the random variable V_τ^k given that the k^{th} bit has value b, where b is either one or zero. Note that $f_{V_\tau^0|B_0}(v|b)$ is the PDF for V_τ given that the received bit has value b. Since b is equally likely to be a *one* or a *zero*, at least in the unencoded case,

$$f_{V_\tau}(v) = \frac{1}{2} f_{V_\tau^0|B_0}(v|0) + \frac{1}{2} f_{V_\tau^0|B_0}(v|1). \tag{7}$$

If N is properly selected, N will be large enough so that the current bit and the N^{th} previous bit have no intersymbol interference, then the rise and fall edge responses must have stabilized over N bit periods. That is, the rise and fall edge responses must have the property that $v_R(t) = v_{high}$ and $v_F(t) = v_{low}$ for all $t > (D + NT)$. That v_R and v_F have known values for large time leads to an explicit formula for conditional PDF of V_τ^N. In particular,

$$\begin{aligned} f_{V_\tau^N|B_N}(v|0) &= \delta(v - v_{low}) \\ f_{V_\tau^N|B_N}(v|1) &= \delta(v - v_{high}) \end{aligned} \tag{8}$$

where again δ is the usual impulse function. The formulas in (8) have a simple interpretation. If only zeros (ones) have been transmitted for the last N bits, then $v(\tau)$ is almost certainly equal to v_{low} (v_{high}).

From (7) and (8), $f_{V_\tau^N|B_N}(v|b)$ is known and $f_{V_\tau^0|B_0}(v|b)$ is the density function needed for bit-error rate estimation. It is perhaps painfully obvious that what is needed is an iteration formula for computing $f_{V_\tau^{k-1}|B_{k-1}}(v|b)$ from $f_{V_\tau^k|B_k}(v|b)$.

3.3 Iteration Formula

It is possible to relate $f_{V_\tau^k|B_k}(v|b)$ to $f_{V_\tau^{k-1}|B_{k-1}}(v|b)$ using a few conditional probability identities and exploiting the fact that $v(\tau)$ is the sum of suitably shifted edge responses. Before beginning the derivation, recall that B_k is the discrete random variable associated with the k^{th} bit, and takes on either the value zero or the value one. We consider the two cases for B_k separately, as although such a presentation is less dense, it seems clearer.

To start, note that standard conditional probability identities [1] yield

$$\begin{aligned} &f_{V_\tau^{k-1}|B_{k-1}}(v|1) = \\ &f_{V_\tau^{k-1}|B_{k-1},B_k}(v|1,0)P(B_k = 0|B_{k-1} = 1) \\ &+f_{V_\tau^{k-1}|B_{k-1},B_k}(v|1,1)P(B_k = 1|B_{k-1} = 1) \end{aligned} \tag{9}$$

and

$$\begin{aligned} &f_{V_\tau^{k-1}|B_{k-1}}(v|0) = \\ &f_{V_\tau^{k-1}|B_{k-1},B_k}(v|0,0)P(B_k = 0|B_{k-1} = 0) \\ &+f_{V_\tau^{k-1}|B_{k-1},B_k}(v|0,1)P(B_k = 1|B_{k-1} = 0). \end{aligned} \tag{10}$$

That the bits are uncorrelated and equally likely to be a zero or a one yields a simplification for a pair of terms in the above pair of equations. In particular,

$$\begin{aligned} &f_{V_\tau^{k-1}|B_{k-1},B_k}(v|1,1)P(B_k = 1|B_{k-1} = 1) = \\ &\qquad \frac{1}{2} f_{V_\tau^k|B_k}(v|1) \\ &f_{V_\tau^{k-1}|B_{k-1},B_k}(v|0,0)P(B_k = 0|B_{k-1} = 0) = \\ &\qquad \frac{1}{2} f_{V_\tau^k|B_k}(v|0), \end{aligned} \tag{11}$$

where we exploited the fact that if $B_k = B_{k-1}$, there is no edge between the k^{th} and $k-1^{th}$ bit.

The as yet unsimplified terms in (9) and (10) are associated with conditions that guarantee there will be an edge of a particular type between the k^{th} and $k-1^{th}$ bits, but the impact of that additional edge is easy to compute thanks to the superposed-edge assumption for $v(\tau)$; one need only evaluate a properly time-shifted edge response and add the resulting value to $v(\tau)$. Therefore, the corresponding update to the PDF of the associated random variable is to convolve the density function with a shifted impulse function. This simple impact on the PDF can be used to relate

$f_{V_\tau^{k-1}|B_{k-1}}(v)$ to $f_{V_\tau^k|B_k}(v)$ for the as yet unsimplified pair of terms in (9) and (10),

$$
\begin{aligned}
f_{V_\tau^{k-1}|B_{k-1},B_k}(v|1,0)P(B_k=0|B_{k-1}=1) &= \\
\frac{1}{2}\left(f_{V_\tau^k|B_k}(v|0)*\delta(v-v_R(\tau_{k-1}))\right)& \\
f_{V_\tau^{k-1}|B_{k-1},B_k}(v|0,1)P(B_k=1|B_{k-1}=0) &= \\
\frac{1}{2}\left(f_{V_\tau^k|B_k}(v|1)*\delta(v-v_F(\tau_{k-1}))\right)& \qquad (12)
\end{aligned}
$$

where $*$ denotes convolution.

Combining (11) and (12) yields a formula that can be used to compute $f_{V_\tau^{k-1}|B_{k-1}}(v|b)$ from $f_{V_\tau^k|B_k}(v|b)$. More explicitly,

$$
f_{V_\tau^{k-1}|B_{k-1}}(v|1) = \frac{1}{2}\left(f_{V_\tau^k|B_k}(v|1)+f_{V_\tau^k|B_k}(v|0)*\delta(v-v_R(\tau_{k-1}))\right) \qquad (13)
$$

and

$$
f_{V_\tau^{k-1}|B_{k-1}}(v|0) = \frac{1}{2}\left(f_{V_\tau^k|B_k}(v|0)+f_{V_\tau^k|B_k}(v|1)*\delta(v-v_F(\tau_{k-1}))\right). \qquad (14)
$$

3.4 Inductive Algorithm

A more abstract representation of (14) and (13) helps to simplify subsequent use and also clarifies some of the dependencies. In particular, applying (14) and (13) can be denoted as

$$
f_{V_\tau^{k-1}|B_{k-1}}(v|b) = \Gamma\left(f_{V_\tau^k|B_k}(v|b),\delta(v-v_R(\tau_k)),\delta(v-v_F(\tau_k))\right) \qquad (15)
$$

where Γ is an operator that maps three PDFs to a single PDF. Note that Γ does not depend on explicitly k, rather the input densities to Γ contain the k dependence. Also, recall that $\delta(v-v_R(\tau_k))$ and $\delta(v-v_F(\tau_k))$ are probability density functions associated with the deterministic values of the edge responses. In the next section, when jitter is considered, these two probability density functions will no longer be impulses.

One use of the Γ notation is to simplify the description of the inductive approach for computing the probability density function needed for computing bit-error rates, $f_{V_\tau|B}(v|b)$. Since $f_{V_\tau|B}(v|b) = f_{V_\tau^0|B_0}(v|b)$, and since the probability density function, $f_{V_\tau^N|B_N}(v|b)$ is known (see equation (8)), $f_{V_\tau|B}(v|b)$ can be computed by starting with $f_{V_\tau^N|B_N}(v|b)$ and repeatedly applying Γ. That is, Γ can used to compute $f_{V_\tau^{N-1}|B_{N-1}}(v|b)$ from $f_{V_\tau^N|B_N}(v|b)$, and then reapplied $N-1$ more times to generate $f_{V_\tau^0|B_0}(v|b) = f_{V_\tau|B}(v|b)$.

As is clear from (14) and (13), each application of Γ involves computing two convolutions and adding two PDFs. These computations can be made efficient using the same voltage discretization strategy described for the pulse-based methods. In particular, since the argument range for each of the probability density functions is the range of voltage values at the receiver input, this range of voltages can be discretized on to a grid of M uniformly spaced voltage sample points. Then, instead of manipulating PDFs, projected versions (now probability mass functions) are manipulated instead. That is, all the PDFs are approximated as tables of M voltage-probability pairs. If the voltage samples are uniformly spaced, then the convolutions in (14) and (13) become discrete convolutions and are easily accelerated using the FFT, just as in the pulse superposition case.

4. JITTER

As noted above, superposed edge responses can sometimes be a more accurate model of serial link behavior than superposing pulse responses, but statistical analysis using edge responses is more complicated due to the innate edge correlation. Although this improved modeling accuracy can be helpful in the case of rise and fall response asymmetries, the ability to properly model transmit jitter is the more compelling reason to address the complications of using edge-response based statistical techniques.

4.1 Models

Designers have identified a number of sources of transmit jitter [3], and most can be modeled as a distribution of random shifts in transition times for the digital input to serial link driver. Such distributions generate shifts in the rise and fall responses, $v_R(t)$ and $v_F(t)$. These time shifts can be incorporated in the superposition formula for $v(\tau)$ given in (4), as

$$
v(\tau) \approx \sum_{k=0}^{N} \alpha_k v_R(\tau_k + t_{J_k}) + \beta_k v_F(\tau_k + t_{J_k}) \qquad (16)
$$

where t_{J_k} is the value of the random variable T_{J_k} that has a PDF that best models the jitter. The simplest choice for the jitter PDF is a zero mean Gaussian, but more realistic models and test strategies include periodic and uniformly distributed jitter [6].

Although not a practical strategy, it is perhaps instructive to consider that (16) can be used to generate a Monte-Carlo approach for estimating the PDF of $V(\tau)$, $f_{V_\tau}(v)$, in the presence of jitter. Each Monte-Carlo evaluation would require selecting one of 2^N possible bit patterns and then selecting a set of N time shifts from the PDF the T_{J_k}'s. Clearly, the dimension of the space that a Monte-Carlo method would have to explore would be extraordinarily high.

4.2 Edge-based Statistical Approach

To see how to add jitter to the above inductive algorithm for computing f_{V_τ} using edge responses, consider the simplification formula (12). In that formula, conditional probability density functions are convolved with the impulsive density functions associated with sampling the deterministic rise and fall responses. Specifically, in the no-jitter case,

$$
\begin{aligned}
f_{V_R^{\tau_k}}(v) &= \delta(v-v_R(\tau_k)) \\
f_{V_F^{\tau_k}}(v) &= \delta(v-v_F(\tau_k)), \qquad (17)
\end{aligned}
$$

where $f_{V_R^{\tau_k}}(v)$ and $f_{V_F^{\tau_k}}(v)$ denote the PDFs for the rise and fall response functions sampled at $\tau_k \equiv D + kT + \tau$.

Adding jitter to the edge responses implies that the sampled responses are no longer deterministic, and therefore no longer have impulsive PDFs. Instead, as shown in Figure 6, the PDF for time shifts that model jitter generate a derived probability density function for the edge response voltages. That is, if the probability density function for the jitter is given by $f_{T_{j_k k}}(t)$, then

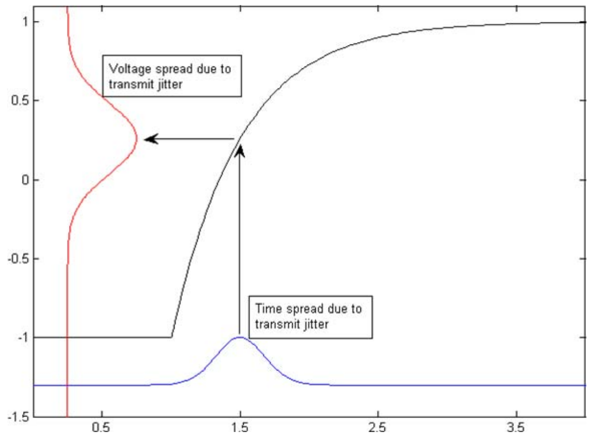

Figure 6: Relating the probability density function for jitter in time to the derived probability density function in voltage.

$$f_{V_R^{\tau_k}}(v) = \frac{d}{dv}\int_{t|v_R(\tau_k+t)\le v} f_{T_{J_k}k}(t)dt$$
$$f_{V_F^{\tau_k}}(v) = \frac{d}{dv}\int_{t|v_F(\tau_k+t)\le v} f_{T_{J_k}k}(t)dt. \quad (18)$$

As a simple, though impractical, example, consider that if the jitter time-shifts are modeled as a zero-mean Gaussians with very small standard deviations, then the PDFs for the rise and fall edge response voltages will be Gaussians with means shifted by $v_R(\tau_k)$ and $v_F(\tau_k)$ respectively, and standard deviations scaled by $dv_R(\tau_k)/dt$ and $dv_F(\tau_k)/dt$ respectively. In practice, as noted above, PDFs are represented as tables of voltage-probability pairs, and such tables of voltage-probability pairs are easily computed for the PDF derived from the PDF of the jitter regardless of its form. Gaussian or periodic distributions, or any combination, can easily be considered, as will be shown in the examples that follow.

5. ALGORITHM SUMMARY

A brief summary of the steps used to analysis a serial link using the edge-based strategy is:

1. Extract circuit-level model of the interconnect.

2. Combine circuit-level model of interconnect with transistor-level models of driver and receiver circuits.

3. Compute edge responses using a circuit simulator.

4. Extract v_{high}, v_{low}, and D from edge responses.

5. Determine the $f_{V_\tau}(v)$ for samples values of τ between $D - 0.2T$ and $D + 0.2T$ using the edge-based inductive algorithm.

6. Generate eye diagrams, bathtub curves, or bit-error rate plots.

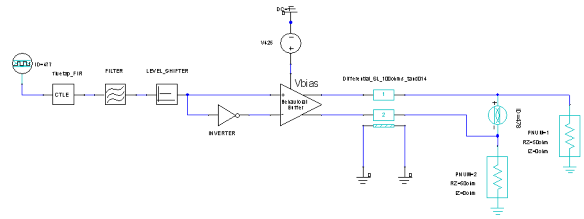

Figure 7: A differentially driven serial link connector

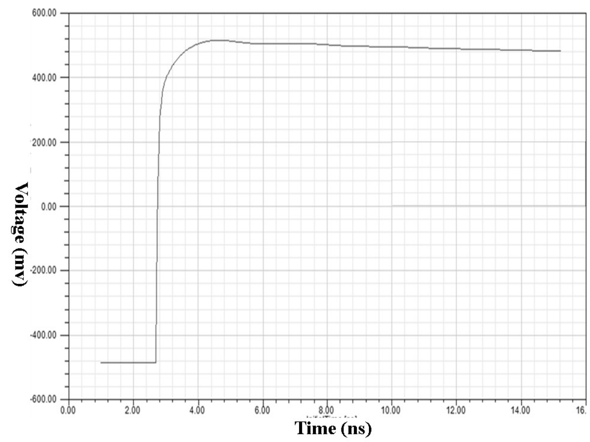

Figure 8: Rising edge (step) response of the differentially driven serial link.

6. EXAMPLE RESULTS

In this section we consider a differentially driven serial link, shown in Figure 7, to both verify and demonstrate the effectiveness of the edge-based statistical approach. The serial link driver is a proprietary design that uses a 125 picosecond bit period, and the transmission line in the serial link is a 25 centimeter long pair of wires surrounded by a grounded shield. The nominal grounded and coupling capacitances of the wires are 120 and 80 picofarads per meter respectively, the nominal series self and coupling inductances are 0.35 and 0.025 microhenries per meter respectively, the series resistance varies from 40 to 320 ohms per meter with increasing frequency, and the resistance coupling to ground varies from 50 ohms to 5 ohms with increasing frequency. The rising edge response for the serial link is shown in Figure 8.

The accuracy of statistical approach depends directly on how well superposing time-shifted edge responses approximates the actual transient response of a serial link. To test the accuracy of the superposed-edge response approximation, a sequence of ten random bits was applied to the input of the link in Figure 7, and the link transient response was computed using standard circuit simulation. The link transient response was also approximated using superposed edge responses, and the two transient results compared in the plot in Figure 9. As can be seen from the plot, the superposed-edge response approximation is such a close match to the circuit simulation results that the two are almost indistinguishable.

Once a delay time and a threshold voltage have been de-

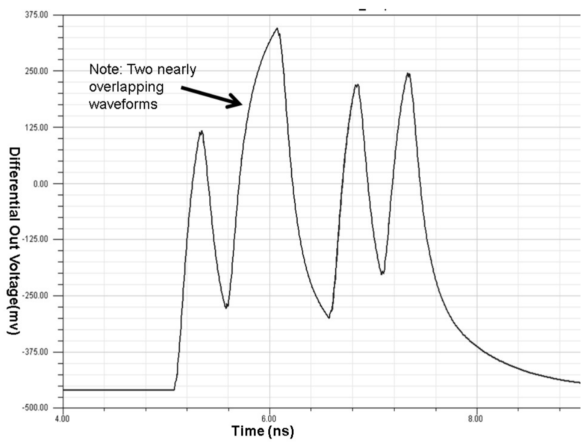

Figure 9: A comparison of circuit and edge-superposition transient simulation for a ten-bit input pattern. Notice that the two results are so close that they are nearly indistinguishable on the plot.

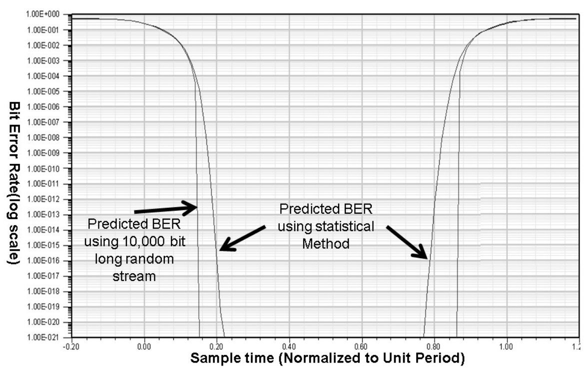

Figure 10: A comparison of bit-error rate versus sample time using the statistical approach and using 10^4 bits with edge-superposition (see text). Transition time jitter was included, with the jitter distribution given by a combination of a Gaussian with a one picosecond standard deviation and periodic jitter with maximum deviation of five picoseconds.

termined by examining edge responses and an eye diagram (Figure 3), then the conditional PDF of the voltage distribution, $f_{V_\tau^0|B_0}(v|b)$, can be used to determine bit error rates versus sample time τ. The plot of bit error rate versus sample time is often referred to as a “bathtub” curve, and is commonly used to characterize a serial link. In the following examples, we examine the bathtub curve to demonstrate the validity and accuracy of the statistical approach.

To demonstrate that the statistical method is correct, we compared the serial link bathtub curve generated by the statistical approach to results generated by superposing time-shifted edge responses from long pseudo-random bit sequences with jittered transition times. For the bathtub curve results shown in Figures 10, and 11, the distribution of the transition time jitter is a combination of a Gaussian with a one picosecond standard deviation and a periodic distribution with maximum deviation of five picoseconds [11]. In Figure 10, the statistical approach for generating the bathtub curve is compared to generating the bathtub curve using

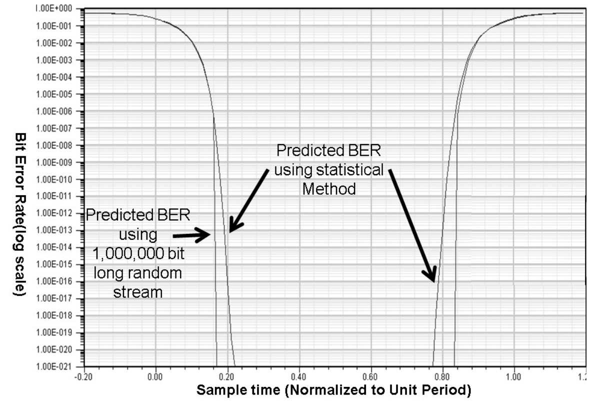

Figure 11: A comparison of bit-error rate versus sample time using the statistical approach and using 10^6 bits with edge-superposition (see text). Transition time jitter is the same as in Figure 10.

the results from superposing the time-shifted and jittered edge responses associated with a pseudo-random 10^4 long bit sequence. The experiment is repeated with a pseudo-random 10^6 long bit sequence and the results are given in Figure 11. Note that as the length of the bit sequence increases, the bathtub curve generated using pseudo-random bit sequences is slowly converging to the statistical results.

In a second example, the distribution of the transition time jitter is altered by increasing the standard deviation of the Gaussian part of the distribution to five picoseconds. The bathtub curves were recomputed using the statistical method and a 10^4-long pseudo-random bit sequence. From the bathtub curve plots in Figures 12, it can be seen that the higher Gaussian standard deviation results in bit errors even when the sample time is near its optimal point. Notice also that results generated by a 10^4 long pseudo-random bit sequence does not capture the center region of the bathtub curve. For the above two examples the statistical approach completed in approximately 30 seconds using a typical workstation, and generated estimates for bit error rates down to well below 10^{-12}. Superposing the one million jittered edge responses required approximately 300 seconds, but generated a bathtub curve that was only accurate for bit error rates above 10^{-7}. In order to accurately estimate bit error rates as low as 10^{-12} using superposed edge responses from pseudo-random bit sequences, days of computer time would be needed. Note that circuit simulation of the serial link with pseudo-random bit sequence inputs was not attempted, as days of computer time would be required even to estimate bit error rates down to 10^{-4}.

As a final example, we demonstrate the impact of using four-tap feed-forward equalization on the bit error rate (Figure 13) for the case where transition time jitter is the same as in the example in Figure 11. By comparing Figure 13 to Figure 11, it is easily seen that the feed-forward equalization has widened the very low bit error rate region (bottom of the bathtub in the curve).

7. CONCLUSIONS

Designing the sophisticated circuitry used in modern high-speed serial links requires analysis tools that are able to accurately estimate very low bit error rates, as low as 10^{-12}.

Recently developed statistical methods based on pulse responses can estimate such low bit error rates, but pulse-based methods have difficulty modeling edge-related effects such as transmitter jitter or asymmetric rise and fall behavior. In this paper we derived an inductive algorithm that makes it computationally feasible to perform statistical simulation using a superposed-edge response approach, even though edge responses are innately correlated. The advantage of the edge-response based statistical approach is that jitter and rise and fall asymmetries can be handled in a straightforward way, and examples are used to demonstrate the validity and effectiveness of the edge-based approach.

Although beyond the scope of this paper, it is possible to extend the edge-based method to handle such features as decision-feedback equalization, 8b/10b and other types of encoding [8], as well as crosstalk from neighboring channels.

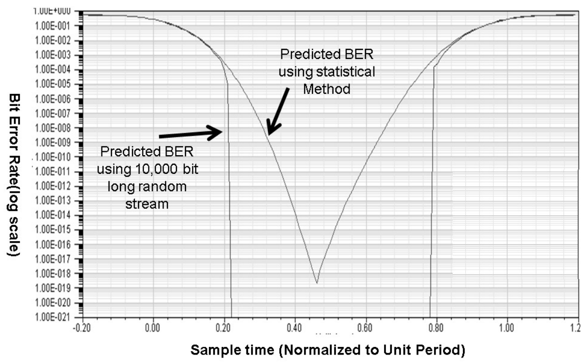

Figure 12: A comparison of bit-error rate versus sample time using the statistical approach and using 10^4 bits with edge-superposition (see text). Transition time jitter is the same as in Figure 10, except the Gaussian standard deviation was increased to *five* picoseconds.

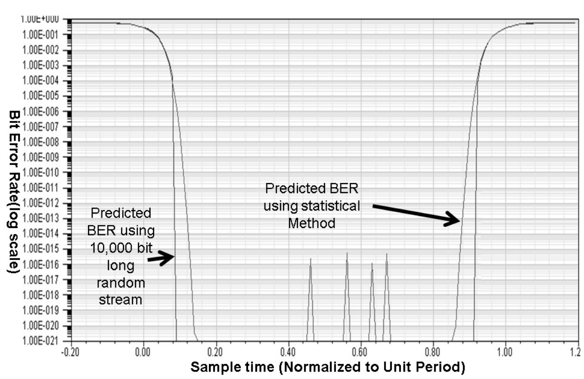

Figure 13: A comparison of bit-error rate versus sample time using four-tap feed-forward equalization. Compare to figure 11 and note wider low bit error rate region.

8. REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, Introduction to Probability, Athena Scientific, Belmont, Massachusetts, 2002.

[2] D. Oh, F. Lambrecht, J. Ren, S. Chang, B. Chia, C. Madden, and C. Yuan, “Prediction of System Performance Based on Component Jitter and Noise Budgets,” IEEE Topical Meeting on Electrical Performance of Electronic Packaging, October 2007.

[3] M. P. Li, “Design and Test for Multiple Gbps Communication Devices and Systems,” Intl. Engineering Consortium, 2005.

[4] A. Sanders, M. Resso and J. D’Ambrosia, “Channel Compliance Testing Utilizing Novel Statistical Eye Methodology,” DesignCon 2004.

[5] K. K. Kim, J. Huang, Y.-B. Kim, and F. Lombardi, “Analysis and Simulation of Jitter Sequences for Testing Serial Data Channels,” *IEEE Trans. on Industrial Informatics*, Vol. 4, No. 2, May 2008.

[6] M. Shimanouchi, “An Approach to Consistent Jitter Modeling for Various Jitter Aspect and Measurement Methods”, *ITC Proceedings*, pp. 848-857, 2001

[7] Y. Chang and D. Oh, “Fast ISI Characterization of Passive Channels Using Extreme Value Distribution,” *IEEE Topical Meeting on Electrical Performance of Electronic Packaging*, October 2007.

[8] N. Blitvic and V. Stojanovic, “Statistical Simulator for Block Coded Channels with Long Residual Interference,” *IEEE International Conference on Communications*, 2007.

[9] M. P. Li, “Design and Test for Multiple Gbps Communication Devices and Systems”, *Intl. Engineering Consortium*, 2005.

[10] A. F. Sanders, “Fundamental Stochastic Jitter Processes Associated with Clock and Data Recovery: A Tutorial” Analog Circuit Design: High-speed Clock and Data Recovery, High-performance Amplifiers, Power Management, M. Steyaert, A. H. M. van Roermund and H. Casier, Eds, Springer, Netherlands, 2008.

[11] N. Ou, T. Farahmand, A. Kuo, S. Tabatabaei, and A. Ivanov, “Jitter Models for the Design and Test of Gbps-Speed Serial Interconnects,” IEEE Design and Test of Computers, vol. 21, Jul–Aug 2004.