Fast Method for an Accurate and Efficient Nonlinear Signaling Analysis

Dan Jiao, *Fellow, IEEE*, and Jianfang (Olena) Zhu, *Member, IEEE*

***Abstract*—Conventional methods for signaling analysis are based on linear time invariant (LTI) assumptions. In this paper, we develop a method for efficient and accurate signaling analysis of nonlinear circuits. This method retains the nonlinear simulation as it is instead of linearizing the original nonlinear problem or adopting a simplified nonlinear model, thereby ensuring accuracy. Meanwhile, it does not suffer from the drawbacks of the brute-force nonlinear simulation in computational efficiency. Being a generic method for signaling analysis, the method is applicable to all kinds of nonlinearities as well as the LTI systems. Simulations of realistic industry nonlinear circuits have demonstrated the accuracy and efficiency of the proposed method.**

***Index Terms*—Fast simulations, nonlinear circuits, nonlinear simulations, nonlinear time invariant (LTI), signal integrity (SI), signaling analysis, time variant.**

I. Introduction

HIGH-SPEED digital design is facing numerous challenges in order to meet new design demands, such as higher data rates, lower power consumption, larger compactness, and higher level of system integration. To address these challenges, advanced signal integrity (SI) analysis of high-speed systems is crucial. Prevalent SI simulations [1] largely rely on linear time invariant (LTI) assumptions to calculate eye margins for low bit error rates (BERs) such as 10^{-12}. However, nonlinear and time variant effects in today's high-speed systems are becoming significant and thus critical to the system performance. For example, redrivers with strong nonlinear behaviors used in high-speed input/output (I/O) designs to drastically boost eye margins; significant nonlinearities and time variants due to control-tuning in equalizers, transmitters, receivers, etc. For such designs, the conventional LTI-based SI simulations could result in large errors. Without accurately and efficiently capturing the non-LTI phenomena, system design could collapse, leading to over-design issues, high cost, and high risk.

In the literature, there exist a number of methods that are aimed to solve the nonlinear time-variant simulation problem. Nonlinear models such as Volterra model [2], [3] and the neural network model [4] have been developed and utilized to linearize the nonlinear drivers. Behavioral simulation based tools such as CppSim [5] have been used to simulate nonlinear equalizers, transceivers, etc. Algorithms using double or multiple edge responses [6], [7] have been designed to take into account the impact of previous bits on the current bit. All of these methods have certain successes but also limitations. Research still needs to be done to satisfactorily model generic nonlinear time-variant phenomena. Without sufficient simulation capabilities of non-LTI elements in the design of platform, package, and silicon, designers have been forced to add cost in all of these areas to guardband for the uncertainty in the simulation predictions. It has eliminated many potentially valuable designs. It has also added large cost and manufacturing complexity to existing designs.

In this paper, we develop a method that is fundamentally different from all previous attempts. This method keeps the nonlinear simulation as it is instead of linearizing the original nonlinear problem or adopting a simplified nonlinear model, thus ensuring accuracy. Though nonlinear in nature, the proposed method does not suffer from the drawbacks of the traditional brute-force nonlinear simulations. Furthermore, it is generically applicable to both nonlinear time-variant systems and LTI systems. The method performs accurate and fast non-LTI signaling analysis with a capability of handling both intersymbol interference (ISI) and crosstalk. Moreover, it provides an effective metric to measure how severe the nonlinear time-variant effects are. In addition to high-speed digital design, this work can also benefit other areas where nonlinear simulation challenges are present.

This paper is a significant expansion of our conference paper [9]. It is expanded with a detailed presentation of the method, a fast algorithm for implementing the method, and also many new validation and application examples.

II. Proposed Method

A. General Idea

We start the proposed work by considering whether there exists a true solution to the problem being tackled. The answer is positive: to investigate the overall performance of a nonlinear channel/circuit for signaling, a rigorous but brute-force solution is to find all 2^m responses for *m*-bits random inputs, where *m* is the channel memory. The challenges of this solution approach are obvious. The 2^m responses cannot be synthesized using LTI principles such as shift invariant, proportionality, superposition,

Manuscript received December 15, 2016; revised February 4, 2017; accepted February 10, 2017. Date of publication March 20, 2017; date of current version April 20, 2017. This work is for the Special Issue and is an expanded version from the 2016 IEEE Symposia from Ottawa. This work was supported in part by Intel Corporation and in part by Semiconductor Research Corporation.

D. Jiao is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: djiao@purdue.edu).

J. Zhu is with Intel Corporation, Hillsboro, OR 97124 USA (e-mail: olena.j.zhu@intel.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEMC.2017.2671425

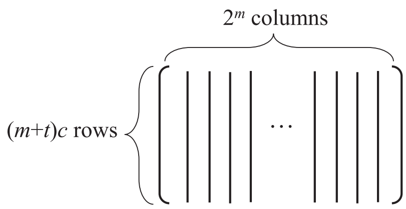


Fig. 1. Illustration of the response matrix.

convolution, etc. Linearization of a true nonlinear problem is known to be error prone. Performing a nonlinear simulation of $m \times 2^m$ bits is computationally prohibitive for greater than 10^{12} bits; but a BER of 10^{-12} and lower is required to reliably assess the performance of a high-speed channel. In this section, we present an effective method to address the aforementioned challenges.

Consider the response of a non-LTI system for an arbitrary m-bit input. Let the number of samples for each bit width be c. The response of the non-LTI system to each m-bit input can be represented by $(m + t)c$ discrete values, where t denotes the number of additional bits in the response. Notice that due to the ringing effects and other high-frequency effects, the response can be longer than the original width of the input. Let us put each discretized response as one column in a matrix, and call the matrix *response matrix*. The response matrix has $(m + t)c$ rows, and 2^m columns for storing all the 2^m responses, as illustrated in Fig. 1. The overlay of each column in the response matrix based on one bit's width produces the eye-diagram of the nonlinear system.

It is evident that the row dimension of the response matrix is always much smaller than its column dimension for a relatively large m, since 2^m grows much faster with m than $(m + t)c$ does. To give an example, consider $m = 100$, $t = 10$, $c = 20$; then $(m + t)c = 2200$ while $2^m = 1.27 \times 10^{30}$. As a result, the rank of the *response matrix* is no greater than row dimension $(m + t)c$. This means among 2^m columns, at most $(m + t)c$ columns are linearly independent. This further suggests that we do not have to simulate 2^m cases, since out of the 2^m cases, at most $(m + t)c$ responses are linearly independent, while all the other responses can be represented by the $(m + t)c$ responses. To understand this more easily, consider the N column vectors in an identity matrix. One can use these N vectors to represent an arbitrary vector of length N. The actual rank of the response matrix k is in fact even smaller than $(m + t)c$. This means we only need to perform $O(k)$ nonlinear simulations. From the resultant $O(k)$ responses, we can evaluate the overall channel performance just like the original 2^m responses do.

From the aforementioned analysis, it becomes clear that the original nonlinear signaling analysis problem can be transformed to an equivalent mathematical problem. This problem is to construct a rank-k representation of the response matrix $\mathbf{R}_{pq}$ $(p = (m + t)c,\ q = 2^m)$ as the following:

$$\mathbf{R}_{pq} \overset{\varepsilon}{=} \mathbf{A}_{p\times k}(\mathbf{B}^T)_{k\times q} \quad (k < \min(p, q)) \tag{1}$$

for a prescribed accuracy ε *without* generating all the columns of the response matrix, since each column represents one nonlinear simulation case. Notice that such a rank-k representation contains the *complete* information of the response matrix within the required error tolerance ε. Therefore, the worst-case response is contained in the $\mathbf{A}_{p\times k}(\mathbf{B}^T)_{k\times q}$ representation. Furthermore, one cannot randomly select $O(k)$ simulations to obtain (1) since such a brute-force approach cannot ensure accuracy.

For an LTI-system with an m-bit channel memory, the rank k of (1) is analytically known to be m, since each of the 2^m responses can be obtained from a linear superposition of m responses. To be specific, we can choose $\mathbf{A}$ as the matrix whose ith column is the system response to unit vector e_i, the ith bit of which is 1, with other bits being zero. As a result, matrix $\mathbf{A}$ is nothing but the response matrix for m bit sequences that make an identity matrix of order m. With $\mathbf{A}$ constructed in such a way, $\mathbf{B}$ is analytically known for all 2^m inputs, with each column of $\mathbf{B}^T$ nothing but the bit sequence corresponding to one of the 2^m bit sequences. Hence, one only needs to perform m simulations to find $\mathbf{A}$. In fact, one only needs to perform *one* simulation to find $\mathbf{A}$ using the shift invariant property of an LTI-system. Since $\mathbf{B}$ is analytically known, using (1), one can immediately obtain the complete response $\mathbf{R}$ by performing a computationally efficient sparse matrix–matrix multiplication, from which the worst-case response is also readily known. As can be seen, the proposed method, though motivated for solving non-LTI systems, is equally applicable to LTI systems. The proposed method is also different from prevailing methods for the LTI-based signaling analysis such as the peak distortion analysis.

For non-LTI systems with an m-bit input, the rank of the nonlinear response matrix is no less than m, and often much greater than m. From this perspective, the rank of the response matrix can be employed as a metric to measure the strength of the nonlinearity. In addition, $\mathbf{B}$ is not analytically known, and $\mathbf{A}$ cannot be constructed using the response to an identity matrix of order m either. Instead, they have to be found mathematically without using any LTI-based principle.

B. Fast Algorithm

To make the proposed method feasible in practice, we have to be able to efficiently find a low-rank representation of the response matrix. A naïve algorithm would be taking a singular value decomposition (SVD) of the response matrix directly. However, to do so, first, we need to know the entire response matrix; second, we have to perform the SVD on a dense matrix $\mathbf{R}$ fast, neither of which is feasible. A better algorithm is to use the adaptive cross approximation scheme [6]. This scheme only requires the knowledge of $O(k)$ rows and $O(k)$ columns of $\mathbf{R}$ to build a rank-k representation of $\mathbf{R}$. Although $O(k)$ columns of $\mathbf{R}$ can be generated without any difficulty, the $O(k)$ rows are not possible to be computed when m is large. This is because each row involves 2^m nonlinear simulation cases.

In this section, we develop a fast algorithm to find a low-rank representation of the response matrix. This algorithm only requires $O(k)$ nonlinear simulations to assess the performance of a nonlinear circuit for signaling instead of the original 2^m nonlinear simulations, where m is the bit number related to channel memory, and k is much less than 2^m, which is also adaptively determined based on a prescribed accuracy. The computational complexity of this algorithm is $kO(m)$, and hence allowing for the eye diagram to be generated efficiently for low BERs.

Our fast algorithm is developed based on the following important finding: the generation of a rank-k model of a matrix, in fact, only involves k columns of the matrix. In other words, even though a full-matrix-based $\mathbf{R}$ is available for use, at the end, only k columns are utilized to obtain a rank-k model. To understand this point, as a summation of k rank-1 representations, we can rewrite

$$\mathbf{R} = \mathbf{A}\mathbf{B}^T = a_1 b_1^T + a_2 b_2^T + \cdots + a_k b_k^T \tag{2}$$

where

$$\begin{aligned} \mathbf{A} &= [a_1\ a_2\ \cdots\ a_k] \\ \mathbf{B} &= [b_1\ b_2\ \cdots\ b_k]. \end{aligned} \tag{3}$$

The above-mentioned equation can be found from a successive rank-1 approximation until the accuracy is reached. Take the first rank-1 approximation $a_1 b_1^T$ as an example. The two vectors can be chosen as

$$\begin{aligned} a_1 &= \mathbf{R}(:, j_1) \\ {b_1}^T &= \frac{\mathbf{R}(i_1, :)}{\mathbf{R}(i_1, j_1)} \end{aligned} \tag{4}$$

where j_1 is the index of the column we arbitrarily choose to begin with, $\mathbf{R}(:, j_1)$ denotes the j_1th column of $\mathbf{R}$, i_1 is the row index corresponding to the largest entry in this column, $\mathbf{R}(i_1, :)$ denotes the i_1th row of $\mathbf{R}$, and $\mathbf{R}(i_1, j_1)$ is the entry of $\mathbf{R}$ at the i_1th row and j_1th column. The residual matrix $\mathbf{R}_1 = \mathbf{R} - a_1 b_1^T$ will hence have its i_1th row and j_1th column zeroed out, i.e.,

$$\mathbf{R}_1(:, j_1) = \mathbf{R}_1(i_1, :) = 0. \tag{5}$$

We then work on the residual matrix $\mathbf{R}_1$, and apply the same procedure of (4) to find a_2 and b_2^T. In other words, at any ith step where we obtain a_i and b_i^T, the $\mathbf{R}$ in (4) is replaced by the current residual matrix $\mathbf{R}_{i-1}$, which is nothing but

$$\mathbf{R}_{i-1} = \mathbf{R} - \sum_{\nu=1}^{i-1} a_\nu b_\nu^T. \tag{6}$$

From the aforementioned process, it can be seen clearly that, at every step, we find a column vector in the residual matrix and use this vector as an a vector. Therefore, the column vectors of $\mathbf{A}$, which make the rank-k model, are found in sequence as the following:

$$\begin{aligned} a_1 &= \mathbf{R}(:, j_1) \\ a_2 &= \mathbf{R}(:, j_2) - a_1 {b_1}^T(j_2) \\ a_3 &= \mathbf{R}(:, j_3) - a_1 {b_1}^T(j_3) - a_2 {b_2}^T(j_3) \\ &\cdots \\ a_k &= \mathbf{R}(:, j_k) - a_1 {b_1}^T(j_k) - a_2 {b_2}^T(j_k) - \cdots - a_{k-1} {b_{k-1}}^T(j_k) \end{aligned} \tag{7}$$

Algorithm 1: Obtaining (1) in a Small Number of Nonlinear Simulations based on Prescribed Accuracy.

1. Step 1:
   1) Start from any column j_1, perform a nonlinear simulation, obtain $\mathbf{R}(:, j_1)$. Let $a_1 = \mathbf{R}(:, j_1)$.
   2) In this column, find the largest entry. Use its row index as i_1. Then ${b_1}^T(j_1) = \mathbf{R}(i_1, j_1)/\mathbf{R}(i_1, j_1)$, which has only its j_1-entry filled at this time.
   3) Generate i_1th row of $\mathbf{R}_0$, $\mathbf{R}_0(i_1, :)$.
   4) In the row of $\mathbf{R}_0(i_1, :)/\mathbf{R}(i_1, j_1)$, find the largest entry, and use its column index as next column index j_2.
2. Step 2:
   While accuracy is not reached, do the following:
   1) $\nu = \nu + 1$
   2) Perform nonlinear simulation for j_ν-case, obtain a new column $\mathbf{R}(:, j_\nu)$.
   3) Extend ${b_m}^T (m = 1, 2, ..., \nu - 1)$ to have j_ν column entry
   4) Compute $a_\nu = \mathbf{R}(:, j_2) - \sum_{m=1}^{\nu-1} a_m {b_m}^T(j_2)$;
   5) In the a_ν column, find the largest entry. Use its row index as i_ν. Then ${b_V}^T(s) = \mathbf{R}(i_\nu, s) - \sum_{m=1}^{\nu-1} a_m(i_\nu) {b_m}^T(s)$, where $s = \{j_1, ..., j_\nu\}$.
   6) Generate i_vth row of $\mathbf{R}_0$, $\mathbf{R}_0(i_v, :)$.
   7) Find the largest entry in the row of $\mathbf{R}_0(i_\nu, :) - \sum_{m=1}^{\nu-1} a_m(i_\nu) {b_{m0}}^T$, this entry's column index is $j_{\nu+1}$.
   8) Check the accuracy of the current rank-ν model.

in which ${b_i}^T(j_m)$ ($i = 1, 2, \ldots, k-1; m = 1, 2, \ldots, k$) denotes the entry at the j_{m}th column in row vector ${b_i}^T$. Hence, the row vectors ${b_1}^T, {b_2}^T, \ldots, {b_k}^T$ never need to be generated at all columns, but at k columns only.

Based on the above-mentioned finding, we develop a fast algorithm to generate a rank-k model of $\mathbf{R}$. This algorithm only requires the knowledge of $O(k)$ columns of $\mathbf{R}$, and hence $O(k)$ nonlinear simulations. In addition, this algorithm is controlled by a prescribed accuracy. The details are given in Algorithm 1. Essentially, each row vector ${b_i}^T$ is filled progressively at the selected columns, instead of being generated as a whole. This can be seen from steps 1.2, 2.3, and 2.5. In step 2.7, ${b_{m0}}^T$ denotes the counterpart of ${b_m}^T$ corresponding to initialization matrix $\mathbf{R}_0$. The accuracy of Algorithm 1 is determined by using the following relative error (RelErr):

$$\text{RelErr} = \frac{\|a_v\|_2 \|b_v\|_2}{\sqrt{\sum_{v=1}^{k} \|a_v\|_2^2 \|b_v\|_2^2}} < \varepsilon \tag{8}$$

where the subscript 2 denotes a 2-norm. When the above-mentioned equation is satisfied, the residual matrix is negligible based on accuracy setting ε. As a result, we have found an accurate representation of the response matrix, and hence the procedure can be stopped. It is worth mentioning that the initialization matrix $\mathbf{R}_0$ does not need to be generated as a whole. We only need to generate them at the rows and columns

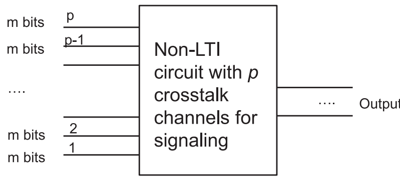


Fig. 2. Illustration of a non-LTI circuit with crosstalk.

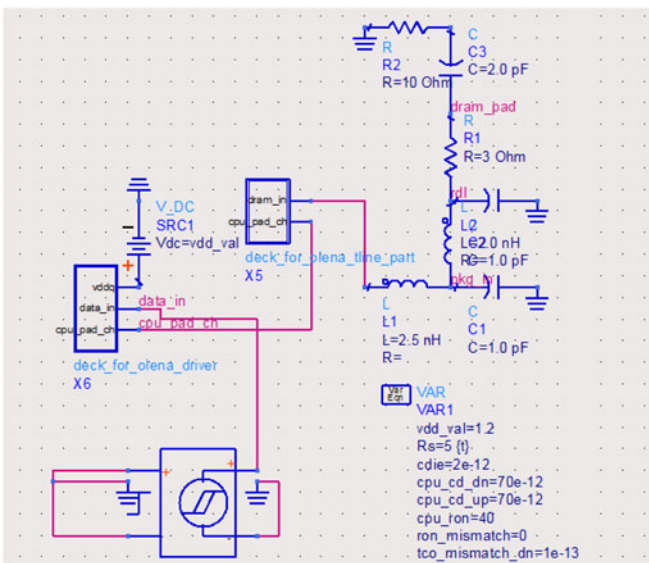


Fig. 3. Circuit schematic of a nonlinear DDR memory channel with a series of package and board interconnects. (Source: Intel)

adaptively selected by the proposed algorithm, when the algorithm proceeds step by step. This is reflected in Algorithm 1, as can be seen from steps 1.3 and 2.6.

C. Remark on Nonlinear Circuits With Crosstalk

For nonlinear circuits with p crosstalk channels shown in Fig. 2, the response matrix has $2^{m\times p}$ columns and $(m+t)c$ rows. A brute-force simulation would become very expensive even for a modest m. The efficient method described in the above-mentioned sections is equally applicable to the nonlinear signaling analysis with crosstalk.

III. Validation and Application

We have simulated various non-LTI circuits provided by a semiconductor industry company to validate the accuracy and efficiency of the proposed method for the fast non-LTI signaling analysis. In the eye-diagrams plotted in this section, the vertical axis denotes the voltage in volts, and the horizontal axis denotes the unit interval subdivided into ten equal subintervals if not specifically specified. The sampling rate is chosen to be ten samples per bit width, which is 0.1 ns per sample for the first and the third example, and 0.04 ns per sample for the second example. This is chosen based on the accuracy required to capture the frequencies present in the non-LTI system response. Notice that the low-rank nature of the response matrix is irrespective of the sampling rate one uses to sample the system response. In addition, the rank of the response matrix can never exceed its row dimension, which is $(m+t)c$.

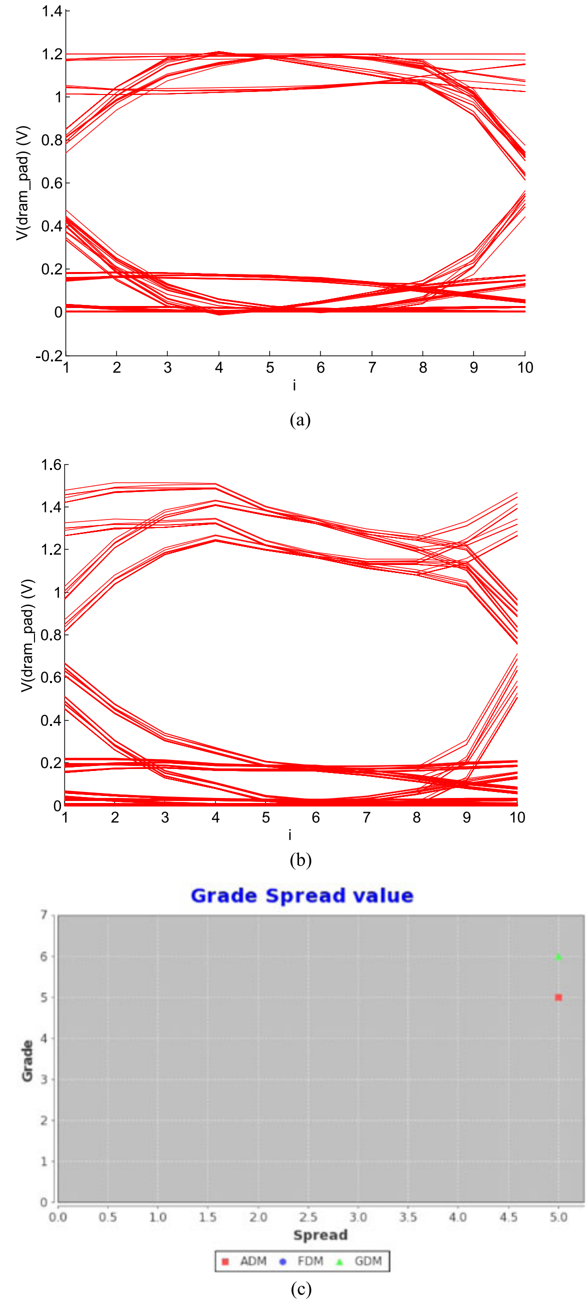


Fig. 4. Comparison between the non-LTI eye-diagram and the LTI-based one. (a) Eye-diagram from brute-force 2^m nonlinear simulations. (b) LTI-based eye-diagram. (c) Grade and spread value plot using [13] (FDM and GDM overlap.)

A. DDR Memory Channel With a Non-LTI Transmitter Equalizer

The first example is a double data rate (DDR) memory channel with a non-LTI transmitter equalizer. The circuit schematic plotted in advanced design system (ADS) is shown in Fig. 3.

For a proof of the concept, we first consider a case of 5-bit random input, thus $m=5$, and there are in total 32 possible responses. In Fig. 4(a) and (b), we plot the eye-diagram of

TABLE I
SORTED SINGULAR VALUES

29.1272939016438
12.6757724478313
9.20500792081679
8.55704210053796
8.51526984676052
0.757153995555508
0.551676890692430
0.460981125018731
0.422223389621352
0.166786999218057
0.136895127147346
0.104243159908216
0.0953420816066097
0.0374210869402222
0.0193856439123355
0.0167483232696945

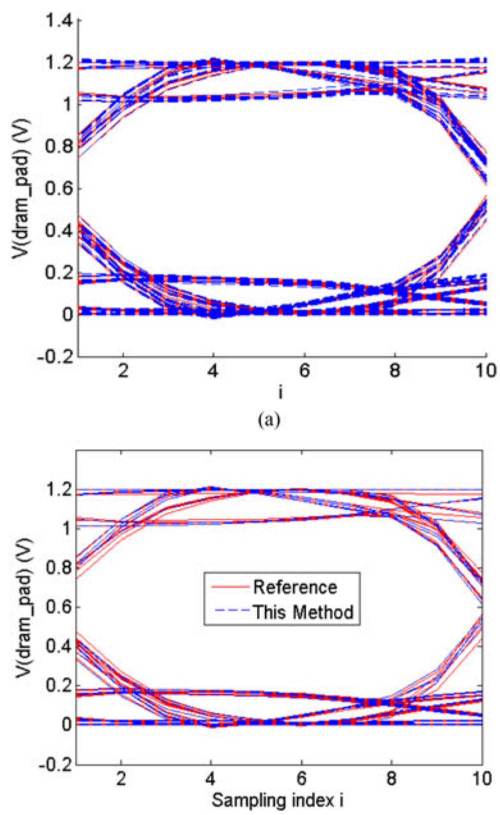


Fig. 5. Eye-diagram generated from a rank-9 representation (blue) in comparison with that from brute-force 2^m nonlinear simulations (red). (a) Rank-9 representation generated from a full response matrix. (b) Rank-9 representation generated from nine simulations.

the nonlinear channel obtained from 32 brute-force nonlinear simulations, and the eye-diagram of the nonlinear channel obtained from a single-bit response and LTI concepts. Obviously, the two eye-diagrams are very different, indicating that an LTI-based treatment of this non-LTI channel would be significantly wrong. This fact is further verified by using the feature selective validation technique [10]–[13] to compare the two data sets. The grade and spread values are plotted in Fig. 4(c), which can be seen as high as 5. Hence, in this example, the nonlinear effect is strong, which is also evident from the more than 100 mV error

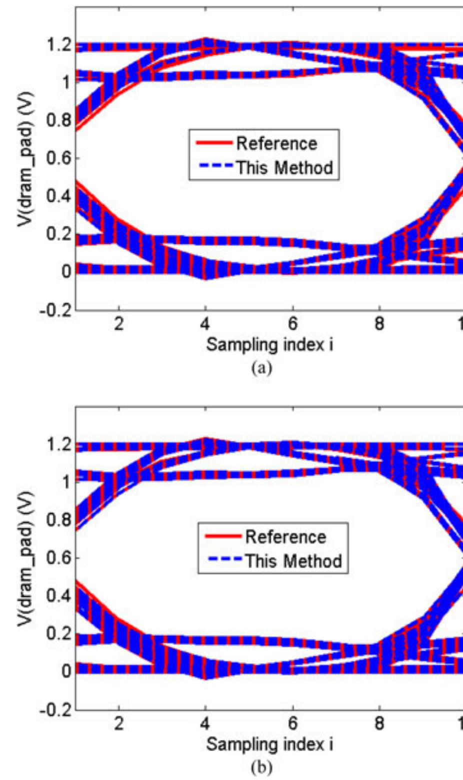


Fig. 6. Eye-diagrams generated from the proposed fast algorithm using two different accuracy settings in comparison with that obtained from brute-force 2^m nonlinear simulations (Reference). (a) Prescribed accuracy $\varepsilon = 1\%$. (b) Prescribed accuracy $\varepsilon = 0.1\%$.

in the received voltage. It should also be noted that such a strong nonlinearity is not common in many existing products.

To examine whether the response matrix is low rank or not, in Table I, we list the singular values of the response matrix sorted from the largest to the smallest. It is clear that the rank of the response matrix is 9 for 1% accuracy, since all the singular values beyond the ninth singular value are less than 1% of the maximum singular value, and thereby $||\mathbf{R} - \mathbf{R}_{\mathrm{rank}-9}||/||\mathbf{R}|| < 1\%$. As a result, we numerically verify that the response matrix is of low rank. We then recover the eye-diagram using the nine singular vectors only. The recovered eye-diagram is shown to agree very well with the original one as can be seen from Fig. 5(a).

If we have to first generate all 2^m responses, then obtain a low-rank representation of the response matrix, the proposed method is not going to work since the computational cost is the same as that of the brute-force method. The key merit of this method is that we only need to generate $O(k)$ simulations, where k is the rank, to obtain a rank-k representation of the response matrix. To demonstrate this point, in Fig. 5(b), we plot the eye-diagram generated from a rank-9 representation of the response matrix obtained from just nine simulations. It is clear that the eye-diagram agrees very well with that obtained from brute-force 32 simulations.

Next, we consider a case of 6-bit random input, thus $m = 6$, and there are in total 64 possible responses. In Fig. 6(a) and (b), we plot the eye-diagrams obtained from the proposed fast algorithm for two accuracy settings, $\varepsilon = 1\%$ and

TABLE II
LIST OF RANK, COLID, AND RELERR

rank = 1	colId = 1	relErr = 1.00000e + 00
rank = 2	colId = 2	relErr = 7.04472e–01
rank = 3	colId = 4	relErr = 5.72729e–01
rank = 4	colId = 5	relErr = 5.39081e–02
rank = 5	colId = 9	relErr = 4.96631e–01
rank = 6	colId = 3	relErr = 7.47490e–02
rank = 7	colId = 11	relErr = 1.39256e–01
rank = 8	colId = 19	relErr = 6.04356e–01
rank = 9	colId = 35	relErr = 5.11983e–01
rank = 10	colId = 6	relErr = 4.02074e–02
rank = 11	colId = 51	relErr = 3.29282e–01
rank = 12	colId = 34	relErr = 4.05464e–02
rank = 13	colId = 39	relErr = 5.71664e–01
rank = 14	colId = 18	relErr = 5.35226e–03

0.1%, respectively. Excellent agreement can be observed with the eye-diagram of the nonlinear channel generated from 64 brute-force nonlinear simulations.

In Table II, we list the rank, nonlinear case simulated (colId), and resultant accuracy (relErr) of the proposed method for achieving a prescribed accuracy of $\varepsilon = 1\%$. As can be seen, the final rank is 14 for this nonlinear channel. This rank is determined by the proposed algorithm based on the prescribed accuracy. One does not need to artificially assume it.

In this example, we have also studied the performance of the proposed algorithm with respect to initialization matrix $\mathbf{R}_0$. We compare two representative approaches. Approach one is to perform one nonlinear simulation of the channel for a randomly selected bit sequence, and then take an arbitrary resultant value corresponding to 0 to fill in all 0 locations in the response matrix, and take one resultant value corresponding to 1 to fill in all 1 positions in the response matrix. Approach two is to use the LTI-based response matrix as an initialization matrix. This initialization matrix can be efficiently obtained by simulating just one bit sequence corresponding to integer 1, and then using the shift invariant and superposition property to obtain other columns of the response matrix. In Fig. 7, we plot the eye-diagrams obtained from these two initialization approaches based on the same accuracy setting $\varepsilon = 1\%$. It is evident that both yield accurate results, which demonstrates that the proposed fast algorithm does not rely on a good initialization to produce accurate results. However, a better initialization helps enhance performance. As can be seen, the LTI-based initialization, since in this example it serves as a good initial guess of the true response matrix, is shown to have a better performance.

We then proceed to simulate a larger case for which ADS performed a nonlinear simulation of 5×10^4 random bits, whereas the proposed method only requires five simulations of five bits each. The CPU time cost by the proposed method is 0.9 s, whereas that cost by the brute-force ADS simulation is 1775 s.

B. USB3.1 Channel With a Redriver

Next, a USB3.1 channel with a redriver is simulated. Its topology is illustrated in Fig. 8(a). Redriver is an active component, which may contain receiver equalizers, transmitter de-emphasis, amplifiers, etc. It is being used in high speed I/O links such as

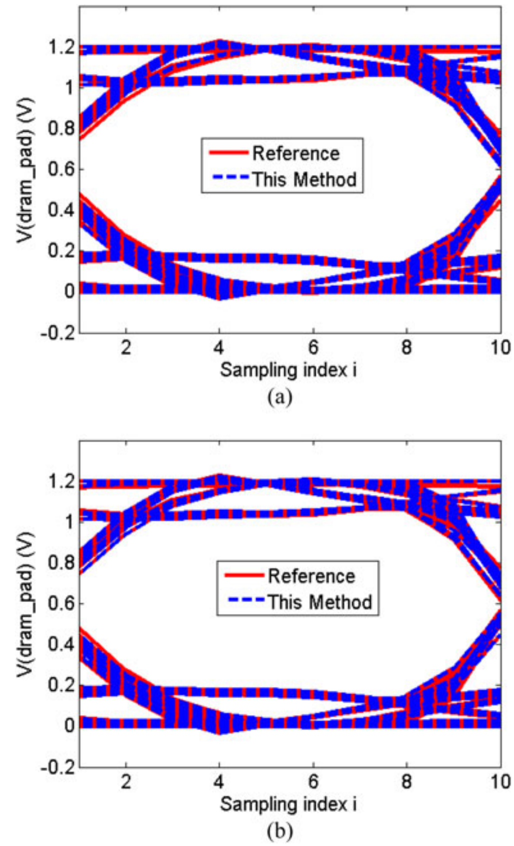


Fig. 7. Comparisons of the eye-diagrams generated using two different initialization approaches for the same accuracy setting of 1%. (a) Approach 1. (b) Approach 2.

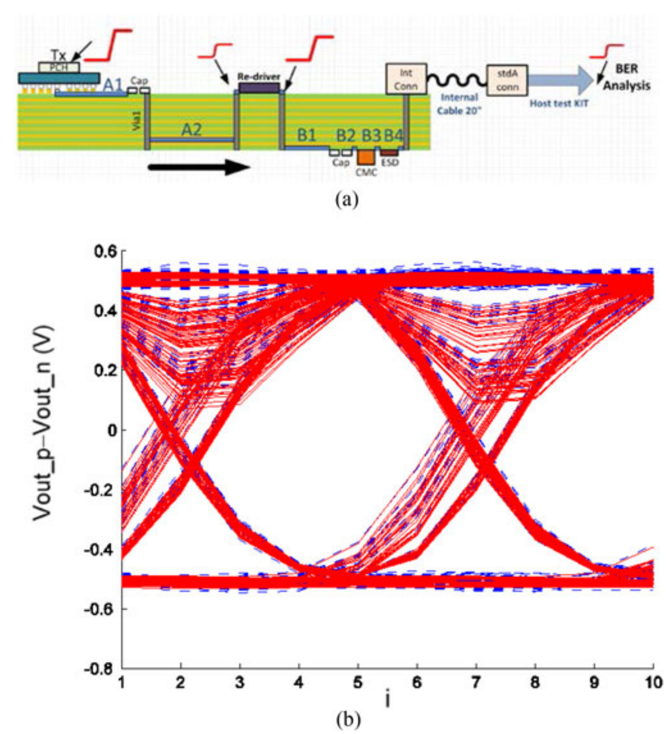


Fig. 8. Simulation of a redriver circuit. (a) Topology. (b) Eye-diagram of the differential voltage (Voutp-Voutn) from a brute-force nonlinear simulation (red) in comparison with that from the proposed method (blue).

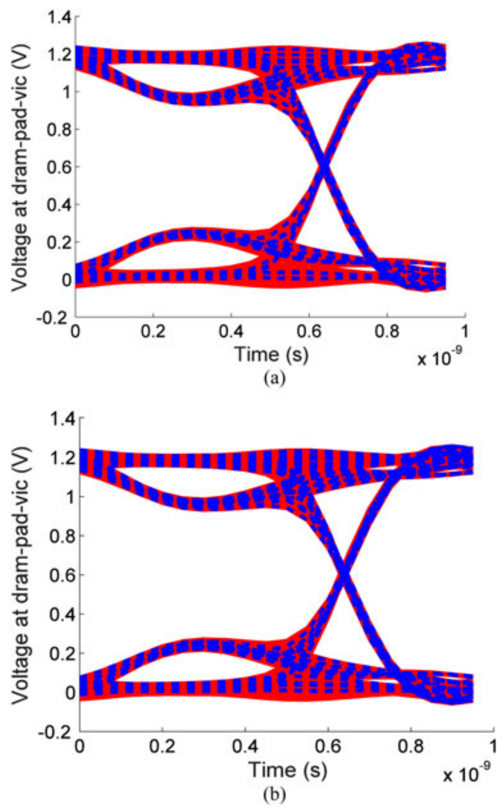


Fig. 9. Eye-diagrams generated from the proposed fast algorithm using two different accuracy settings (blue) in comparison with that obtained from brute-force $2^{m\times p}$ nonlinear simulations (red). (a) Prescribed accuracy $\varepsilon = 1\%$. (b) Prescribed accuracy $\varepsilon = 0.1\%$.

USB 3.1, SATA 6 Gbps, 10GBase-KR, and SAS 12 Gbps to boost the eye margins. Due to the strong non-LTI nature of the redriver, it is very challenging to accurately and efficiently evaluate SI performance of a channel with a redriver. A comparison of an LTI-based simulation with a non-LTI based analysis in this example reveals totally different results, indicating the strong nonlinearity of the circuit being analyzed. In Fig. 8(b), we plot the eye-diagram obtained from the proposed method for a 12-bit random input using a rank-26 representation in comparison with the eye-diagram generated by a brute-force simulation. Excellent agreement can be observed. Comparing the ∼26 nonlinear simulations we perform with the original 2^{12} simulations one needs to do, the efficiency one can gain from the proposed method is obvious. In addition, if we compare the rank required by the second example with that of the first example, it is evident that the nonlinearity of the redriver circuit is much stronger.

C. Nonlinear Signaling Analysis With Crosstalk

Next, a nonlinear circuit with crosstalk is simulated, where the interconnect network is modeled by a 10-line T-line model that has crosstalk with each other, each of which is driven by a nonlinear driver. We consider a case of 5-bit random input and with two drivers activated, thus $m = 5$, $p = 2$, and there are in total 1023 possible nontrivial responses. The pulse has a bit width of 1 ns, and a rising/falling time of 20 ps. The sampling rate is 0.1 ns. In Fig. 9(a) and (b), we plot the eye diagrams obtained from the proposed fast algorithm for two accuracy settings, $\varepsilon = 1\%$ and 0.1%, respectively. Excellent agreement can be observed with the eye diagram obtained from 1023 brute-force nonlinear simulations. For 1% accuracy, only 10 nonlinear simulations are required using the proposed method; for 0.1% accuracy, only 20 nonlinear simulations are required. Compared to the 1023 brute-force nonlinear simulations, the efficiency of the proposed method is evident.

TABLE III
LIST OF RANK, COLID (NONLINEAR SIMULATION CASE NUMBER IN BASE 10), AND RELERR FOR ACCURACY $\varepsilon = 1\%$

Rank	colId	RelErr
1	1	1.00000e + 00
2	701	8.81825e–01
3	858	5.78155e–01
4	346	2.02582e–02
5	428	4.53900e–01
6	668	3.17919e–01
7	418	2.17079e–01
8	70	3.24757e–01
9	71	2.31992e–01
10	582	1.68083e–03

TABLE IV
LIST OF RANK, COLID (NONLINEAR SIMULATION CASE NUMBER IN BASE 10), AND RELERR FOR ACCURACY $\varepsilon = 0.1\%$

Rank	ColId	RelErr
1	1	1.00000e + 00
2	701	8.81825e–01
3	858	5.78155e–01
4	346	2.02582e–02
5	428	4.53900e–01
6	668	3.17919e–01
7	418	2.17079e–01
8	70	3.24757e–01
9	71	2.31992e–01
10	582	1.68083e–03
11	347	1.06357e–02
12	859	1.41035e–03
13	796	4.45786e–02
14	692	8.15084e–02
15	957	4.92815e–03
16	700	7.08598e–03
17	559	5.62035e–02
18	879	1.24205e–02
19	875	1.39570e–02
20	669	4.40254e–04

In Tables III and IV, we list the rank, nonlinear colId, and relErr (8) of the proposed method for achieving a prescribed accuracy of $\varepsilon = 1\%$ and 0.1% respectively. As can be seen, the rank is determined by the proposed algorithm based on the prescribed accuracy. One does not need to artificially assume it. In addition, the final rank is 10 for $\varepsilon = 1\%$, and 20 for $\varepsilon = 0.1\%$, which is also the number of nonlinear simulations performed. It is much less than 1023 required by the brute-force approach.

IV. CONCLUSION

In this paper, a method is developed for fast and accurate signaling analysis of nonlinear circuits. This method is not restricted to one kind of nonlinearity. It is generic applicable to all kinds of nonlinearities. In this method, the nonlinear simulation is retained to be nonlinear instead of being linearized. However, it does not suffer from the computational cost of a brute-force nonlinear simulation because of its underlying fast algorithm. This fast algorithm only requires $O(k)$ nonlinear simulations instead of 2^m nonlinear simulations to assess the performance of a nonlinear circuit, where m is the bit number related to channel memory, and k is much less than 2^m, which is also adaptively determined based on a prescribed accuracy. The proposed method provides a convenient nonlinear indicator on how severe the nonlinear effect is in the signaling analysis of a non-LTI system, and it accurately analyzes a non-LTI system with fast CPU run time. Simulations of nonlinear circuits involving both ISI and crosstalk have validated the accuracy and efficiency of the proposed method and fast algorithm. In this work, random jitter has not been considered, which will be incorporated in our future work.

Acknowledgment

The authors would like to thank Dr. W.-k. Shih, Dr. A. Norman, Dr. Z. Zhang, Dr. B. Casper, Dr. X. Ye, Dr. J. Liao, Dr. H. Heck, Dr. S. Hall, Mr. A. Bhatt, Dr. H. Chen, Mr. R. Melster, Dr. N. Menezes, Dr. X. Dong, Dr. K. Xiao, Dr. C. Ye, Dr. E. Chiprout, and Dr. C. Gu for their help and support to this work.

References

[1] B. K. Casper, M. Haycock, and R. Mooney, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling scheme," in *Proc. VLSI Circuits Dig. Tech. Papers Symp.*, Jun. 2002, pp. 54–57.

[2] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. New York, NY, USA: Wiley, 1980.

[3] W. T. Beyene, "Peak distortion analysis of nonlinear links," in *Proc. 2013 IEEE 22nd Conf. Electr. Perform. Electron. Packag. Syst.*, 2013, pp. 169–172.

[4] O. Nelles, *Nonlinear System Idientification: From Classical Approaches to Neural Networks and Fuzzy Models*. Berlin, Germany: Springer-Verlag, 2001.

[5] [Online]. Available: http://www.cppsim.com/.

[6] J. Ren and K. S. Oh, "Multiple edge responses for fast and accurate system simulations," *IEEE Trans. Adv. Packag.*, vol. 31, no. 4, pp. 741–748, Nov. 2008.

[7] F. Lambrecht, C.-C. Huang, and M. Fox, "Technique for determining performance characteristics of electronic systems," *U.S. Patent 6 775 809*, Mar. 14, 2002.

[8] S. Börm, L. Grasedyck, and W. Hackbusch, "Hierarchical matrices," Lecture Note 21, Max Planck Inst. Math. Sci., Leipzig, Germany, 2003.

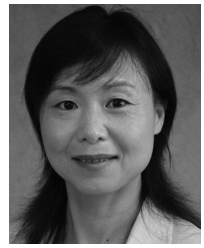
[9] D. Jiao and J. Zhu, "Fast algorithm for nonlinear signaling analysis," in *Proc. 2016 IEEE Int. Symp. Electromagn. Compat.*, 2016, pp. 196–199.

[10] *Standard for Validation of Computational Electromagnetics Computer Modeling and Simulation—Part 1, 2*, IEEE Standard P1597, 2008.

[11] A. P. Duffy, A. J. M. Martin, A. Orlandi, G. Antonini, T. M. Benson, and M. S. Woolfson, "Feature selective validation (FSV) for validation of computational electromagnetics (CEM). Part I—The FSV method," *IEEE Trans. Electromagn. Compat.*, vol. 48, no. 3, pp. 449–459, Aug. 2006.

[12] A. Orlandi, A. P. Duffy, B. Archambeault, G. Antonini, D. E. Coleby, and S. Connor, "Feature selective validation (FSV) for validation of computational electromagnetics (CEM). Part II—Assessment of FSV performance," *IEEE Trans. Electromagn. Compat.*, vol. 48, no. 3, pp. 460–467, Aug. 2006.

[13] FSV tool. [Online]. Available: http://orlandi.ing.univaq.it/pub/FSV_Tool_20.



Dan Jiao (S'00–M'02–SM'06–F'16) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2001.

She then worked in the Technology Computer-Aided Design (CAD) Division, Intel Corporation, until September 2005, as a Senior CAD Engineer, a Staff Engineer, and a Senior Staff Engineer. In September 2005, she joined Purdue University, West Lafayette, IN, USA, as an Assistant Professor with the School of Electrical and Computer Engineering, where she is currently a Professor. She has authored two book chapters and more than 260 papers in refereed journals and international conferences. Her current research interests include computational electromagnetics, high-frequency digital, analog, mixed-signal, and RF-integrated circuit (IC) design and analysis, high-performance VLSI CAD, modeling of microscale and nanoscale circuits, applied electromagnetics, fast and high-capacity numerical methods, fast time-domain analysis, scattering and antenna analysis, RF, microwave, and millimeter-wave circuits, wireless communication, and bio-electromagnetics.

Dr. Jiao has served as a Reviewer for many IEEE journals and conferences. She is an Associate Editor for the IEEE Transactions on Components, Packaging, and Manufacturing Technology. She was among the 85 engineers selected throughout the nation for the National Academy of Engineering's 2011 US Frontiers of Engineering Symposium. She received the 2013 S. A. Schelkunoff Prize Paper Award of the IEEE Antennas and Propagation Society, which recognizes the Best Paper published in the IEEE Transactions on Antennas and Propagation during the previous year, the 2010 Ruth and Joel Spira Outstanding Teaching Award, the 2008 National Science Foundation CAREER Award, the 2006 Jack and Cathie Kozik Faculty Start up Award (which recognizes an outstanding new faculty member of the School of Electrical and Computer Engineering, Purdue University), a 2006 Office of Naval Research Award under the Young Investigator Program, the 2004 Best Paper Award presented at the Intel Corporation's annual corporate-wide technology conference (Design and Test Technology Conference) for her work on generic broadband model of high-speed circuits, the 2003 Intel Corporation's Logic Technology Development (LTD) Divisional Achievement Award in recognition of her work on the industry-leading BroadSpice modeling/simulation capability for designing high-speed microprocessors, packages, and circuit boards, the Intel Corporation's Technology CAD Divisional Achievement Award for the development of innovative full-wave solvers for high frequency IC design, the 2002 Intel Corporation's Components Research the Intel Hero Award (Intel-wide she was the tenth recipient) for the timely and accurate 2-D and 3-D full-wave simulations, the Intel Corporation's LTD Team Quality Award for her outstanding contribution to the development of the measurement capability and simulation tools for high frequency on-chip crosstalk, and the 2000 Raj Mittra Outstanding Research Award presented by the University of Illinois at Urbana-Champaign.

Jianfang (Olena) Zhu (M'11) received the B.S. degree in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 2006, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 2011.

She is a System Architect in Client Computing Group at Intel Corporation, Hillsboro, OR, USA. She has authored one book chapter and more than 35 papers in refereed journals and international conferences. She has more than 16 patent filings at Intel.

Dr. Zhu received the 2015 IEEE International Conference on EMC/Signal and Power Integrity Best Paper Finalist Award, the 2010 IEEE International Symposium on Antennas and Propagation Best Student Paper Finalist Award, the 2010 IEEE Transactions on Advanced Packaging Best Paper Finalist Award, and multiple Intel Group/Division Recognition Awards.