

Method for Accurate and Efficient Eye Diagram Prediction of Nonlinear High-Speed Links

Dan Jiao , *Fellow, IEEE*, Yuhang Dou, *Member, IEEE*, Jin Yan, Jianfang Zhu, and Adam Norman

Abstract—Signaling analysis of nonlinear high-speed circuits is challenging because it cannot rely on linear time-invariant principles, whereas an exhaustive nonlinear simulation is computationally prohibitive for low bit error rates. To find the worst-case eye, an exhaustive approach requires nonlinear simulations of 2^m bit patterns for a channel of m -bit memory. In this article, we represent the nonlinear responses to the 2^m inputs by a rank- k matrix, where k denotes the number of distinct waveforms in the responses for a prescribed accuracy, which is much smaller than 2^m . We further develop a fast full cross approximation algorithm to find the rank- k model with a low complexity independent of 2^m . Simulations of large-scale real-world nonlinear circuits with an over 100-bit channel memory demonstrate the accuracy, efficiency, and capacity of the proposed work.

Index Terms—Fast low-rank approximation, fast simulations, full cross approximation (FCA), nonlinear circuits, nonlinear simulations, nonlinear time-invariant (non-LTI), signal integrity, signaling analysis, time variant.

I. INTRODUCTION

NONLINEARITY in today's systems is critical, but the capability of existing methods is not sufficient to solve large-scale signaling problems with a large channel memory and a large number of crosstalk links. Prevalent SI simulations [1]–[8] largely rely on linear time-invariant (LTI) assumptions to calculate eye margins for low bit error rates. However, nonlinear and time-variant effects have become increasingly important in high-speed system design for higher data rates, lower power consumption, and small form factor. Without sufficient simulation capabilities of analyzing the impact of nonlinear elements in platform, package, and silicon, designers have to add guard band for the uncertainty in the simulation predictions. This may eliminate many potentially good designs. The problem is difficult to solve because an exhaustive simulation is prohibitively expensive, whereas no LTI-based principles can be utilized.

There have been a number of attempts to tackle the problem [9]–[11]. In [9], a Volterra functional series is used to

decompose a nonlinear link into multiple linear networks, and then the peak distortion analysis (PDA) is applied to the linear networks to construct a PDA of the nonlinear link. In [10], multiple edge responses are employed based on the concept of linear superposition. These methods linearize the original nonlinear problem or adopt a simplified nonlinear model. In [12], a new method is developed, which keeps the original nonlinear problem as it is. Meanwhile, the method does not suffer from the drawback of a brute-force nonlinear simulation. The underlying idea is to put the nonlinear responses in a response matrix, and find a low-rank representation of the matrix. However, the meaning of the rank is not revealed in [12]. The computational cost of the method is also high in both time and memory when a channel memory is large or a large number of crosstalk links need to be considered. This is because the algorithm requires generating a selected number of rows of the response matrix. When the channel memory m is large, even generating one row is not feasible because it has $2^m - 1$ entries.

In this article, we develop a fast method to perform the nonlinear signaling analysis. Its computational complexity scales as $O(k^2r)$ independent of the 2^m , where k is the number of distinct waveforms in the eye diagram for a prescribed accuracy, which is much smaller than 2^m ; and r is the number of samples per bit width. We also provide a theoretical analysis on the convergence and accuracy of the proposed fast algorithm. The algorithm is an accelerated full cross approximation (FCA) algorithm. The FCA [14] is known to have guaranteed accuracy in finding a low-rank approximation of a low-rank matrix using the matrix's selected rows and columns. However, the computational cost of an FCA is prohibitive, scaling as $O(k \times r \times 2^m)$ since the response matrix has a row dimension of r , and a column dimension of 2^m . We are able to develop a fast algorithm to reduce the huge cost of FCA to $O(k^2r)$, independent of the 2^m while preserving the accuracy of FCA. In addition, we develop a Bayesian optimization (BO) based method to validate the proposed work for large m . The proposed work has been applied to real-world large-scale nonlinear circuits. Its accuracy, efficiency, and capacity are demonstrated.

II. PROBLEM STATEMENT

For a channel whose memory is m bits, we can send a stream of a large number of random bits to the channel to examine the channel performance. The response at the current bit, $\text{bit}(0)$, can be affected by previous $m - 1$ bits, which can be expressed as a nonlinear function of both time and

Manuscript received August 27, 2020; revised January 22, 2021, April 6, 2021, and April 12, 2021; accepted April 15, 2021. Date of publication June 8, 2021; date of current version October 13, 2021. This work was supported by a Grant from Intel Corporation. (*Corresponding author: Dan Jiao.*)

Dan Jiao and Yuhang Dou are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: djiao@ecn.purdue.edu; dou16@purdue.edu).

Jin Yan, Jianfang Zhu, and Adam Norman are with Intel Corporation, Hillsboro, OR 97124 USA (e-mail: jin1.yan@intel.com; olenaj.zhu@intel.com; adam.j.norman@intel.com).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TEMC.2021.3074923>.

Digital Object Identifier 10.1109/TEMC.2021.3074923

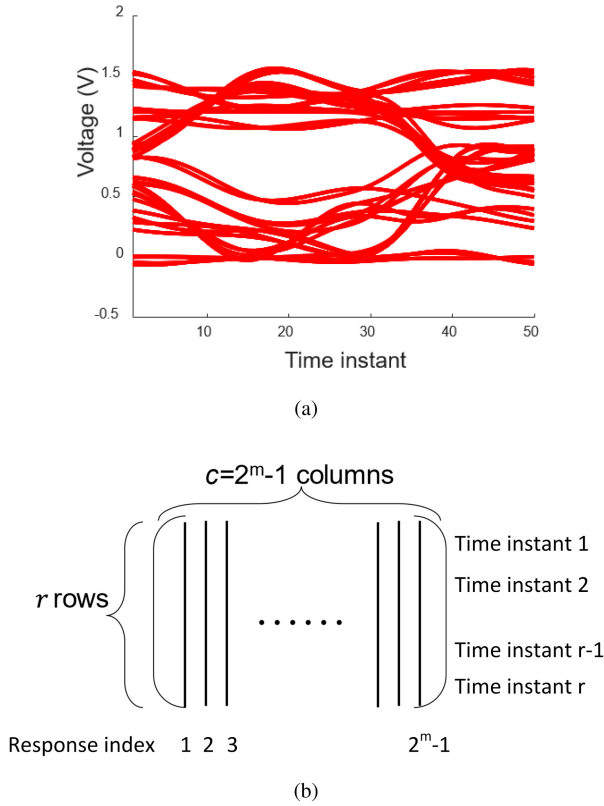


Fig. 1. (a) Illustration of an eye diagram. (b) Matrix \mathbf{E} representation of the eye diagram.

m bits, $y(\text{bit}(-m+1), \dots, \text{bit}(-1), \text{bit}(0), t)$. However, if the bit stream is complete for testing, it is equivalent to sending 2^m kinds of bit sequences to the channel, as the bit more than m bits away from the current bit has a negligible effect. To examine the signal integrity of a nonlinear channel of m -bit memory by eye diagram, thus a rigorous but brute-force solution is to find all $2^m - 1$ responses at current bit. The challenges of this solution approach are obvious, and it is even computationally prohibitive for the large-scale signal integrity analysis where m is large. However, such a large number of bits is frequently encountered in today's high-speed design due to crosstalk, jitter, equalization, etc. Consider a nonlinear channel whose channel memory is 100; also consider one Hspice simulation time to be 8 min, which is typical for simulating a relatively large circuit. For such a scenario, the total number of nonlinear simulations required is 2^{100} . Using Hspice to perform exhaustive simulations would take about 10^{25} years, which is computationally prohibitive.

III. PROPOSED WORK

A. Essential Idea

Consider a channel that has an m -b memory, if we record all the 2^m responses received at the current bit, we obtain an eye diagram of the channel, an example of which is shown in Fig. 1(a). Based on Jiao and Zhu [12], this eye diagram can be stored in a response matrix shown in Fig. 1(b), denoted by \mathbf{E} , where the row dimension of this matrix r is the number of samplings per bit width, and the column dimension c is $2^m - 1$ excluding response to zero input. Each column of \mathbf{E} is one response received at the

current bit, with column i corresponding to the response to bit sequence i (in decimal number). For example, for $m = 3$, there are seven columns in \mathbf{E} , which correspond to responses received at current bit due to bit sequence 001, 010, 011, 100, 101, 110, and 111, respectively.

The response matrix has a rank bounded by its row dimension r , which is much smaller than the column dimension c , and furthermore independent of m . In theory, we can find r linearly independent vectors from \mathbf{E} , using which any column of \mathbf{E} can be represented as a weighted sum of these vectors. However, for nonlinear problems, we do not know the weighting coefficients of the r vectors, since LTI principles are not applicable.

Nevertheless, if we use the number of distinct waveforms (for accuracy ϵ) in the eye diagram as the rank to be found, then the problem becomes solvable. First of all, this number is much smaller than 2^m . This is because the number of distinct voltage levels in the dynamic range of the received signal is a small number at any time instant. It is only related to the resolution used to sample the voltage, regardless of the number of nonlinear simulations. For example, for a voltage range between 0 and 1, dividing it into 100 segments already makes the difference between two adjacent voltage levels no greater than $\epsilon = 1\%$. For r samples in one bit width, the maximum number of distinct waveforms is hence $100r$ for accuracy ϵ , which is a bounded number, and much smaller than 2^m . This is because at each time instant, one just need to perform 100 simulations, i.e., find responses to 100 bit sequences, to capture the 100 distinct voltage values. Since there are r time instants in 1-bit width (i.e., UI, unit interval), in total, there are $100r$ simulations, and hence $100r$ waveforms in the eye diagram. In practice, many of these simulations are shared in common across the time instants, and hence the unique number of simulations one has to do is even smaller. Second, if we can find the k distinct waveforms hence columns in \mathbf{E} , then there is no need to simulate other $2^m - 1 - k$ nonlinear cases, since their waveforms in the eye diagram overlap with the k waveforms that have been simulated (within an error tolerance).

Based on the aforementioned findings, the original problem of finding the eye diagram can be converted to an equivalent problem of finding k distinct waveforms in the eye diagram, by which a low-rank k representation of \mathbf{E} can be obtained. Certainly, if the row number r is smaller than k , the rank of \mathbf{E} will be bounded by the row number r . Although this rank is smaller, from the r linearly independent columns of \mathbf{E} , we do not know how to use them to obtain the other $2^m - r - 1$ columns without invoking a number of nonlinear simulations. Therefore, in this case, we should increase the row number by increasing the sampling rate so that the k distinct waveforms can be found.

To find the low-rank representation of \mathbf{E} efficiently, we utilize [13, Th. 3.1], which is summarized as follows.

Theorem 1: For a matrix \mathbf{R} that satisfies $\|\mathbf{R} - \mathbf{M}\| \leq \epsilon$, and $\text{rank}(\mathbf{M}) \leq k$ for accuracy ϵ , there exist k rows and k columns of the matrix which determine a cross approximation of \mathbf{R} as follows:

$$\tilde{\mathbf{R}} = \mathbf{R}_{:,j}(\mathbf{R}_{i,j})^{-1}\mathbf{R}_{i,:}, \quad (1)$$

and $\|\mathbf{R} - \tilde{\mathbf{R}}\|$ is within error tolerance.

In (1), $\hat{J} = \{j_1, j_2, \dots, j_k\}$ denotes the indexes of the k columns, $\hat{I} = \{i_1, i_2, \dots, i_k\}$ denotes the indexes of the k rows, $\mathbf{R}_{:,j}$ denotes the k columns of \mathbf{R} whose column indexes are in the set \hat{J} , $\mathbf{R}_{\hat{I},:}$ represents the k rows of \mathbf{R} , whose indexes belong to \hat{I} , and $\mathbf{R}_{\hat{I},\hat{J}}^{-1}$ is the inverse of the submatrix in \mathbf{R} formed by the k rows and k columns.

The significance of (1) is that we can use a small set of the original columns and rows of a matrix to construct its low-rank representation. In the context of this article, we seek for the following rank- k representation of the eye diagram:

$$\tilde{\mathbf{E}} = \mathbf{E}_{:, \hat{J}} (\mathbf{E}_{\hat{I}, \hat{J}})^{-1} \begin{bmatrix} \mathbf{E}_{\hat{I}, \hat{J}} & \mathbf{E}_{\hat{I}, J \setminus \hat{J}} \end{bmatrix} \quad (2)$$

where J denotes the whole set of column index from 1 to $2^m - 1$. In the above, the column space of $\mathbf{E}_{\hat{I}, J \setminus \hat{J}}$ is included in the column space of $\mathbf{E}_{\hat{I}, \hat{J}}$, since its columns overlap with the k distinct waveforms, and hence does not need to be computed.

B. On the FCA and Adaptive Cross Approximation (ACA)

The key of constructing (2) is obviously to find the row pivots \hat{I} and column pivots \hat{J} such that the resulting approximation is accurate. It can also be seen from (2) that the choices of such pivots are not unique, as long as $(\mathbf{E}_{\hat{I}, \hat{J}})^{-1}$ is not singular, or ill-conditioned. However, since now the column dimension is $O(2^m)$, if one randomly selects $O(k)$ columns from the whole column set, the accuracy of (2) cannot be guaranteed. It is even more so when it comes to find the worst-case eye height (EH) and eye width (EW) of the eye diagram, since the probability for them to appear is $1/2^m$.

One approach to determine these pivots is the cross approximation algorithm with full pivoting [14], also called FCA, which is shown in Algorithm 1. In this algorithm, we first find the maximal entry in the given matrix \mathbf{E} , whose row and column indexes determine the first set of row and column pivots, (i_1, j_1) . Then, the j_1 th column of \mathbf{E} is stored as a_1 ; and the i_1 th row of \mathbf{E} divided by $\mathbf{E}(i_1, j_1)$ makes the row vector b_1^T . The resultant rank-1 matrix, $a_1 b_1^T$, is then subtracted from the original matrix \mathbf{E} to obtain a residual matrix. In this residual matrix, again, we find the maximal entry, whose row and column indexes determine the second set of row and column pivots, (i_2, j_2) . The a_2 vector is then found from the j_2 column of the residual matrix, whereas b_2^T vector is the i_2 row of the residual matrix divided by the entry of the residual matrix at the i_2 th row and j_2 th column. Such a successive rank-1 approximation is continued until the actual rank of the matrix is revealed, at which δ (the value of the residual matrix at the selected row and column pivots) is zero or close to zero. The whole process results in the following rank- k model of \mathbf{E} :

$$\tilde{\mathbf{E}} = \sum_{\nu=1}^k a_{\nu} b_{\nu}^T \quad (3)$$

which can be proved to be the same as (2) if \mathbf{E} is a rank- k matrix [14]. However, using this algorithm, first, one has to know the entire eye diagram; second, the computational cost is prohibitive, which scales as $O(k \times r \times 2^m)$, neither of which is feasible or affordable.

Algorithm 1: Full Cross Approximation (FCA) (Complexity: $O(k \times r \times 2^m)$ in time and $O(r \times 2^m)$ in memory).

```

1 for  $\nu = 1 : k$  do
2   Find the entry with the largest modulus in  $\mathbf{E}$ , set
    $(i_{\nu}, j_{\nu})$  to be its row and column index; and
    $\delta = \mathbf{E}(i_{\nu}, j_{\nu})$ .
3   if  $\delta \neq 0$  then
4     Compute  $a_{\nu} = \mathbf{E}(:, j_{\nu})$ ;  $b_{\nu}^T = \mathbf{E}(i_{\nu}, :)/\delta$ 
5      $\mathbf{E} = \mathbf{E} - a_{\nu} \times b_{\nu}^T$ 
6   else
7     exit

```

Another way to find (2) is the ACA algorithm [14]. Although this algorithm is more efficient than the FCA shown in Algorithm 1, it also requires a computational cost of $O(k^2(r + 2^m))$, in addition to the cost of obtaining the $2^m - 1$ entries in the k -rows from nonlinear simulations, which is not feasible for large m . In addition, compared to FCA that uses the largest entry in the residual matrix to determine pivots, ACA's choice of pivots is heuristic.

C. Proposed Fast Algorithm

In this article, we develop a fast algorithm to reduce the computational complexity of FCA, for finding the row pivots \hat{I} and column pivots \hat{J} of (2), down to $O(k^2 r)$ in time, and $O(kr)$ in memory while retaining FCA's accuracy. The number of nonlinear simulations required is only $O(k)$. As a result, large-scale nonlinear signaling analysis becomes feasible. In addition, we separate the response matrix when current bit is 1 or 0 and construct two rank- k models, i.e.,

$$\mathbf{E} = [\mathbf{E}(\text{bit}(0) = 1) \quad \mathbf{E}(\text{bit}(0) = 0)] \quad (4)$$

In this way, the accuracy can be further improved. We notice that high accuracy is desired especially for precisely capturing the EH and EW. For example, the two vectors in the response matrix may be very close to each other in terms of linear dependence, however, they can yield a difference at a millivolt level in the EH, which matters to practical design. Next, we will use $\mathbf{E}(\text{bit}(0) = 1)$ as an example to illustrate the proposed algorithm and simplify the notation as \mathbf{E} , whereas $\mathbf{E}(\text{bit}(0) = 0)$ can be handled in the same way.

In this algorithm, we first initialize \mathbf{E} to be \mathbf{E}^0 . There are many ways to generate \mathbf{E}^0 . A simple and effective way is to use an LTI-based channel response. This can be obtained by recording the m responses at the current bit due to a bit 1 located at $-1, -2, \dots$, and $-m + 1$ positions, respectively, for current bit located at position 0. Let $\mathbf{S}_0 \in \mathbb{R}^{r \times m}$ denote these seed responses. Using \mathbf{S}_0 , \mathbf{E}^0 can be obtained readily as follows:

$$\mathbf{E}^0 = \mathbf{S}_0 \mathbf{B} \quad (5)$$

where $\mathbf{B} \in [0, 1]^{m \times (2^m - 1)}$ is the union of all the $2^m - 1$ input bit sequences. Its j th column denotes the j th bit sequence, whose value is the m -b binary representation to decimal number j .

Below is an example of \mathbf{B} for $m = 3$

$$\mathbf{B} = \begin{pmatrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \text{bit 1} & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \text{bit 2} & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ \text{bit 3} & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}. \quad (6)$$

We start from \mathbf{E}^0 to find the row pivots \hat{I} and column pivots \hat{J} . In this matrix, we go through each row, and find the largest entry in each row, out of which we find the entire matrix's maximal modulus. The row and column index of this entry is chosen as the first pair of pivots (i_1, j_1) . Using Algorithm 1, the aforementioned computation would cost $O(r \times 2^m)$ operations, which is prohibitive. However, we find that we can use the sign of each row of \mathbf{S}_0 to determine the maximal modulus in each row of \mathbf{E}^0 in $O(m)$ operations. Basically, for an arbitrary row of \mathbf{E}^0 , i , its maximum modulus can be obtained by adding up either all the positive $\mathbf{S}_0(i, :)$ entries, or all the negative $\mathbf{S}_0(i, :)$ ones, whichever is larger as shown in the following:

$$\max_{1 \leq j \leq 2^m - 1} (|\mathbf{E}^0(i, j)|) = \max(V_p(i), V_m(i)), \quad (1 \leq i \leq r) \quad (7)$$

where

$$V_p(i) = \sum_{j \in [1, m], \mathbf{S}_0(i, j) > 0} \mathbf{S}_0(i, j) \quad (8)$$

$$V_m(i) = \sum_{j \in [1, m], \mathbf{S}_0(i, j) < 0} |\mathbf{S}_0(i, j)|. \quad (9)$$

After finding the maximum in each row, we choose the maximum out of all row maximums to determine the matrix maximum, and hence

$$\max(|\mathbf{E}^0|) = \max_{1 \leq i \leq r} \max(V_p(i), V_m(i)) \quad (10)$$

whose row and column pivots are selected as i_1 and j_1 . The whole computation costs $O(r \times m)$ only, as compared to the cost of $O(r \times 2^m)$ if using Algorithm 1.

Since in the eye diagram generation, both V_p and V_m are important, we split \mathbf{E}^0 into a positive part $\mathbf{E}^{0,+}$, and a negative one $\mathbf{E}^{0,-}$, so that the bit sequences that produce a positive maximum and a negative maximum both can be considered. Thus

$$\begin{aligned} \max(|\mathbf{E}^{0,+}|) &= \max_{1 \leq i \leq r} (V_p(i)) \\ \max(|\mathbf{E}^{0,-}|) &= \max_{1 \leq i \leq r} (V_m(i)). \end{aligned} \quad (11)$$

After finding (i_1, j_1) , we proceed to perform the nonlinear simulation for the bit sequence corresponding to j_1 . The result, which is $\mathbf{E}_{:,j_1}$, is the a_1 vector in the rank- k model being pursued. The i_1 th row of \mathbf{E} divided by \mathbf{E}_{i_1,j_1} is the b_1^T , hence

$$a_1 = \mathbf{E}_{:,j_1} \quad (12)$$

$$b_1^T = \mathbf{E}_{i_1,:} / \mathbf{E}_{i_1,j_1}. \quad (13)$$

However, the row vector b_1^T is not completely known but at the j_1 th column. Hence, at this step, we only need to obtain $b_1^T(j_1)$,

which is

$$b_1^T(j_1) = \mathbf{E}_{i_1,j_1} / \mathbf{E}_{i_1,j_1} = 1. \quad (14)$$

Next, we deduct the rank-1 matrix $a_1 \times b_1^T$ from the original \mathbf{E} to obtain the first residual matrix \mathbf{E}_1 , i.e.,

$$\mathbf{E}_1(:, \hat{J}_1) = \mathbf{E}(:, \hat{J}_1) - a_1 \times b_1^T(\hat{J}_1) \quad (15)$$

where \hat{J}_1 denotes the set of column pivots that have been computed at step 1, which is j_1 . In \mathbf{E}_1 , we need to determine the maximum entry whose row and column indexes are chosen as (i_2, j_2) . However, since \mathbf{E}_1 is not completely known, we use the residue matrix \mathbf{E}_1^0 to estimate \mathbf{E}_1 . Thus, we compute

$$\mathbf{E}_1^0 = \mathbf{E}^0 - a_1^0 \times b_1^{0T} \quad (16)$$

where

$$\begin{aligned} a_1^0 &= \mathbf{E}_{:,j_1}^0 \\ b_1^{0T} &= \mathbf{E}_{i_1,:}^0 / \mathbf{E}_{i_1,j_1}^0. \end{aligned} \quad (17)$$

For each row of \mathbf{E}_1^0 , a brute-force calculation would take $2^m - 1$ operations to determine the maximal entry in this row. Here, again we can reduce it to $O(m)$ by using the sign of seed responses, \mathbf{S}_1 , of \mathbf{E}_1^0 . To derive \mathbf{S}_1 , (16) can be rewritten as

$$\begin{aligned} \mathbf{E}_1^0 &= \mathbf{S}_0 \mathbf{B} - a_1^0 \times c_1^{0T} \mathbf{B} = (\mathbf{S}_0 - a_1^0 \times c_1^{0T}) \mathbf{B} \\ &= \mathbf{S}_1 \mathbf{B} \end{aligned} \quad (18)$$

where c_1^{0T} can be found by using (17) as

$$c_1^{0T} = \mathbf{S}_0(i_1, :) / a_1^0(i_1). \quad (19)$$

From \mathbf{S}_1 , we can use its sign to determine the maximal modulus of \mathbf{E}_1^0 in each row, as shown from (7) to (9). Here, \mathbf{S}_1 can be viewed as the m seed responses of the residual matrix \mathbf{E}_1^0 . Using the sign of $\mathbf{S}_1(i, j)$ for each bit j , we can reversely find out among $2^m - 1$ bit sequences, which bit sequence produces the largest value in the i th row of \mathbf{E}_1^0 in one operation. This is much more efficient than computing (16) as it is, and then find the maximum modulus.

After finding the matrix maximum of \mathbf{E}_1^0 , its row and column pivots are used to determine (i_2, j_2) . We then proceed to perform the nonlinear simulation for the bit sequence corresponding to j_2 . Let \hat{J}_2 denote the two column pivots that have been computed, which are j_1 and j_2 now. The residue matrices \mathbf{E}_1 in (15) and b_1^T in (13) are updated by adding one column for j_2 .

Then, the a_2 and b_2^T vector in the rank- k model of \mathbf{E} can be found as

$$a_2 = \mathbf{E}_1(:, j_2)$$

$$b_2^T(\hat{J}_2) = \mathbf{E}_1(i_2, \hat{J}_2) / \delta_2 \quad (20)$$

where $\delta_2 = a_2(i_2)$, and the resultant residual matrix is

$$\mathbf{E}_2(:, \hat{J}_2) = \mathbf{E}_1(:, \hat{J}_2) - a_2 \times b_2^T(\hat{J}_2). \quad (21)$$

To determine next pivots, again, we compute

$$\mathbf{E}_2^0 = \mathbf{E}_1^0 - a_2^0 \times b_2^{0T} \quad (22)$$

which can be written as

$$\mathbf{E}_2^0 = \mathbf{S}_2 \mathbf{B} \quad (23)$$

in which

$$\mathbf{S}_2 = \mathbf{S}_1 - a_2^0 \times c_2^{0T} \quad (24)$$

where

$$c_2^{0T} = \mathbf{S}_1(i_2, :)/a_2^0(i_2). \quad (25)$$

We then use the signs of \mathbf{S}_2 to determine the maximum entries in \mathbf{E}_2^0 , the row and column pivots of which are chosen as (i_3, j_3) .

The aforementioned computation is performed step by step. At an arbitrary k' th step, we use the sign of seed responses $\mathbf{S}_{k'-1}$ to find the pivots $(i_{k'}, j_{k'})$ of the maximum modulus by using (7). Then, update $\hat{J}_{k'}$ by adding $j_{k'}$ upon $\hat{J}_{k'-1}$. After we perform the nonlinear simulation for the bit sequence corresponding to $j_{k'}$, we can obtain the residue matrix $\mathbf{E}_{k'-1}$ as

$$\mathbf{E}_{k'-1}(:, \hat{J}_{k'}) = \mathbf{E}(:, \hat{J}_{k'}) - \sum_{\nu=1}^{k'-1} a_{\nu} \times b_{\nu}^T(\hat{J}_{k'}). \quad (26)$$

The vectors $a_{k'}$ and $b_{k'}$ for the rank- k model can be obtained as

$$a_{k'} = \mathbf{E}_{k'-1}(:, j_{k'}) \quad (27)$$

$$b_{k'}^T(\hat{J}_{k'}) = \mathbf{E}_{k'-1}(i_{k'}, \hat{J}_{k'})/\delta_{k'} \quad (28)$$

where

$$\delta_{k'} = a_{k'}(i_{k'}). \quad (29)$$

At step- k' ($k' = 1, 2, \dots$), we estimate the error of the current rank- k model using the following formula:

$$\text{error}_{k'} = \frac{\|a_{k'}\| \|b_{k'}^T\|}{\sqrt{\sum_{\nu=1}^{k'-1} \|a_{\nu}\|^2 \|b_{\nu}^T\|^2}} \quad (30)$$

where two-norm is used for each vector. Matrixwise, the denominator of the above is the Frobenius norm of the rank- $(k' - 1)$ model obtained at the $(k' - 1)$ th step, and the numerator is the Frobenius norm of the newly added rank-1 matrix at the k' th step. When the error reaches the prescribed value, we stop the overall process, at which the rank- k model of desired accuracy is found.

D. Algorithm Summary and Complexity Analysis

We summarize the aforementioned algorithm in Algorithm 2. This algorithm is run twice. One for \mathbf{E}^+ , and the other for \mathbf{E}^- , where the current bit is 0 and 1, respectively. The computation performed on \mathbf{E}^0 , shown in Algorithm 2 from Steps 10 to 11, essentially realizes the following rank- k approximation:

$$\mathbf{E}^0 = \mathbf{A}_k \mathbf{C}_k \mathbf{B} \quad (31)$$

where $\mathbf{A}_k \in \mathbb{R}^{r \times k}$ contains the k -vectors of c_{ν}^{0T} , $\mathbf{C}_k \in \mathbb{R}^{k \times m}$ contains the k -vectors of c_{ν}^{0T} , as shown in (19) and (25). The procedure of finding these vectors can be summarized as the following:

$$a_{\nu}^0 = \mathbf{S}_{\nu-1} \mathbf{B}(:, j_{\nu}) \quad (32)$$

Algorithm 2: Proposed Algorithm of $O(k^2r)$ Time Complexity and $O(kr)$ Memory Complexity.

```

1 Initialize  $\mathbf{S}_0$ 
2 for  $\nu = 1, 2, \dots$  do
3   while Accuracy not reached do
4     Using the sign of  $\mathbf{S}_{\nu-1}$  to determine the largest
       modulus in  $\mathbf{E}_{\nu-1}^0$  based on (7), choose its row
       and column index as  $(i_{\nu}, j_{\nu})$ ; and
        $\delta^0 = \mathbf{E}_{\nu-1}^0(i_{\nu}, j_{\nu})$ ,  $\delta = \mathbf{E}_{\nu-1}(i_{\nu}, j_{\nu})$ .
5     if  $\delta \neq 0$  then
6       Augment  $b_{\mu}^T$  ( $\mu = 1, 2, \dots, \nu - 1$ ) by one
         entry at  $j_{\nu}$ 's column
7       Compute  $a_{\nu} = \mathbf{E}_{\nu-1}(:, j_{\nu})$ ;
          $b_{\nu}^T(\hat{J}_{\nu}) = \mathbf{E}_{\nu-1}(i_{\nu}, \hat{J}_{\nu})/\delta$ 
8        $\mathbf{E}_{\nu-1}(:, \hat{J}_{\nu}) = \mathbf{E}_{\nu-1}(:, \hat{J}_{\nu}) - a_{\nu} \times b_{\nu}^T(\hat{J}_{\nu})$ 
9       Compute Error using (30)
10    if  $\delta^0 \neq 0$  then
11      Compute  $a_{\nu}^0 = \mathbf{E}_{\nu-1}^0(:, j_{\nu})$ ,
          $c_{\nu}^{0T} = \mathbf{S}_{\nu-1}^T(i_{\nu}, :)/\delta^0$ , and
          $\mathbf{S}_{\nu}^T = \mathbf{S}_{\nu-1}^T - a_{\nu}^0 \times c_{\nu}^{0T}$ 

```

$$c_{\nu}^{0T} = \mathbf{S}_{\nu-1}(i_{\nu}, :)/a_{\nu}^0(i_{\nu}) \quad (33)$$

$$\mathbf{S}_{\nu} = \mathbf{S}_{\nu-1} - a_{\nu}^0 \times c_{\nu}^{0T} \quad (34)$$

in which ν starts from 1 and finishes at k , and the signs of \mathbf{S}_{ν} at each step are used to determine next pivots (i_{ν}, j_{ν}) based on (7).

The computational cost of (32) for k steps is $O(k \times r \times m)$; the computational cost of (33) for k steps is $O(k \times m)$; and that for k steps of (34) is $O(k \times r \times m)$. As for the pivot determination, it is of $O(r \times m)$ at each step. Hence, the total time complexity of the proposed algorithm for computing (31) is $O(k \times r \times m)$. This is the cost from Steps 10 to 11 in Algorithm 2 together with the cost of Step 4. As for the cost from Steps 5 to 9, it is of $k^2 \times r$. Since k is larger than m in general, the total time complexity of the proposed algorithm is $O(k^2r)$. The number of nonlinear simulations required to be computed is also $O(k)$. As for the memory cost, it is of $O(kr)$, dominated by the cost of storing a_{ν} vectors. Hence, overall, the proposed algorithm has a low complexity independent of 2^m in both time and memory.

It is worth mentioning that for $m < r$ (which is true in general), the mathematical rank of (31) is no greater than m , i.e., the number of bits. However, the algorithm developed here is to find distinct waveforms (columns) of \mathbf{E} for a desired accuracy, not to find a low-rank approximation of \mathbf{E} in a traditional sense. As mentioned in Section III-A, if we do so, we do not know how to use the obtained k columns to synthesize other columns in \mathbf{E} since it is made of nonlinear responses. So one can view the proposed method as *repurposing* the FCA-based rank-revealing algorithm to “find distinct waveforms (columns) of \mathbf{E} ,” or view the rank to be found here as the number of distinct waveforms. In doing so, we utilize one property of FCA: if two columns are very close to each other, when one column is zeroed out in the FCA procedure, so is the other column (in the sense of having a small residual). To facilitate finding a complete set of distinct

waveforms for prescribed accuracy, in addition to choosing the row dimension r no less than k , there is no accuracy parameter used in the procedure of \mathbf{E}^0 's approximation, so that the program can continue to run to find new bit patterns, which are used to generate nonlinear simulation results from which the accuracy of approximated \mathbf{E} is evaluated. In the following section, we also explain why a bit pattern that can result in a distinct waveform in \mathbf{E}^0 would also lead to a distinct waveform in \mathbf{E} .

IV. CONVERGENCE AND ACCURACY

Since \mathbf{E} has a rank- k representation as analyzed in the aforementioned section, the proposed algorithm has a guaranteed convergence. This is because after every step, the rank is increased by 1 in the rank- k model. In other words, the rank of the residual matrix is reduced by 1. After $O(k)$ steps, the rank- k model is found, and the residual matrix becomes zero. This can be proved based on Börn *et al.* [14]. Basically, we can write the residual matrix obtained at each step as

$$\mathbf{E}_{k'} = \mathbf{E} - \sum_{\nu=1}^{k'} a_{\nu} \times b_{\nu}^T \quad (35)$$

where $k' = 0, 1, 2, \dots$. Hence, we have

$$\mathbf{E}_{k'+1} = \mathbf{E}_{k'} - a_{k'+1} \times b_{k'+1}^T. \quad (36)$$

We can prove

$$\text{rank}(\mathbf{E}_{k'+1}) = \text{rank}(\mathbf{E}_{k'}) - 1. \quad (37)$$

This is because first, $a_{k'+1}$ is a column vector in the residual matrix $\mathbf{E}_{k'}$, and hence the rank of $\mathbf{E}_{k'+1}$ is at most the rank of $\mathbf{E}_{k'}$. Second, let $u_{i_{k'+1}}$ be a unit vector that has a 1 appearing at the $i_{k'+1}$ th row, then we have $u_{i_{k'+1}}^T \mathbf{E}_{k'+1} = 0$, but $u_{i_{k'+1}}^T \mathbf{E}_{k'} \neq 0$. Hence, the rank of $\mathbf{E}_{k'+1}$ cannot be equal to $\mathbf{E}_{k'}$. Third, since the two matrices are different by a rank-1 matrix, the difference in their rank is at most 1. As a result, we obtain (37). Using it, we can see that at every step in the proposed algorithm, the rank of the residual matrix is reduced by 1. If \mathbf{E} is of rank k , then after k steps, \mathbf{E}_k is of rank zero, and hence, the method converges. When it converges, the resultant rank- k model is also an accurate representation of \mathbf{E} .

One may wonder if unfortunately, at many steps, the new response we add overlaps with the responses already simulated, although the proposed algorithm would discard that response, the whole procedure can become very inefficient. This is the same question one would raise to a random choice of k bit sequences among $2^m - 1$ ones. If one does so, many bit sequences may result in the same response (for a given error tolerance). It is even more challenging to find the bit sequence that produces the worst-case 1 or 0 at each time instant, whose probability is approximately one out of 2^m . In other words, what makes the proposed algorithm so effective in finding distinct responses? This can be analyzed as follows.

Let

$$\text{bit} = [\text{bit}(-m + 1), \text{bit}(-m + 2), \dots, \text{bit}(0)] \quad (38)$$

be the set of all m bits, where $\text{bit}(0)$ denotes the current bit. This set can be divided into two subsets: one has a significant effect

on the current bit, and the other set's contribution is insignificant (such as 50% smaller than the largest response received at the current bit or even smaller). In general, the former is composed of the bits closest to the current bit, and the smallest set of which is the current bit. Let

$$\overline{\text{bit}} = [0, 0, \dots, \text{bit}(0)] \quad (39)$$

i.e., the current bit being $\text{bit}(0) = 1, 0$, whereas other bits are zero. The response at current bit can be approximately viewed as

$$y(\text{bit}, t) = y(\overline{\text{bit}}, t) + y'(\overline{\text{bit}}, t)\Delta(\text{bit}) + y''(\overline{\text{bit}}, t)\Delta(\text{bit})^2/2 + \dots \quad (40)$$

in which $\Delta(\text{bit})$ is the difference between bit and $\overline{\text{bit}}$. The superscript $'$ denotes a first-order derivative, and $''$ denotes a second-order derivative. The first-order derivative term for all $2^m - 1$ bit sequences is nothing but

$$\mathbf{E}^0 = \mathbf{S}_0 \mathbf{B}. \quad (41)$$

Hence, the bit sequence that generates a distinct waveform in \mathbf{E}^0 is expected to produce a distinct waveform in the nonlinear response $y(t)$, although the two waveforms can be completely different. If we include a few more significant bits in $\overline{\text{bit}}$, this would be even more obvious. This explains why the proposed algorithm can find the distinct k -waveforms effectively in the nonlinear case. Yes, the nonlinear response is not equal to (40). However, the proposed algorithm does not use that either. Instead, it calls for the nonlinear simulation to obtain the exact response for the selected bit sequence. As for the true worst-case 1 and 0, as well as the complete eye diagram, they are the natural outcome of the proposed algorithm after the rank- k model is found.

V. VERIFICATION METHOD OF THE PROPOSED WORK

For smaller m , we can validate the proposed work by exhaustive nonlinear simulations. However, for large m , such as 20 or larger, it becomes not feasible to run the exhaustive simulations. We hence propose to use BO [15], [16], a global optimization method for black-box functions, to validate this article. After the program of the proposed algorithm finishes, we continue to search for patterns that may make the eye even worse. This optimization problem is formulated as follows.

- 1) Optimization variables: each bit in an m -bit sequence.
- 2) Optimization target: the voltage at the desired slice of the eye formed by the waveform of any bit sequence.
- 3) Optimization initialization: all the bit sequences (patterns) \hat{J} found in this article.
- 4) Optimization stopping criteria: when the number of iterations exceeds the user-specified iteration limit.

For better accuracy, we run BO for the worst-case 1 and 0 separately. A more powerful BO-SDP [16] is also employed for cases that have aggressors. We use BO for verification only because its run time is too long to tolerate in practical applications. It should be noted that there could exist more advanced BO than the one available for use in this article.

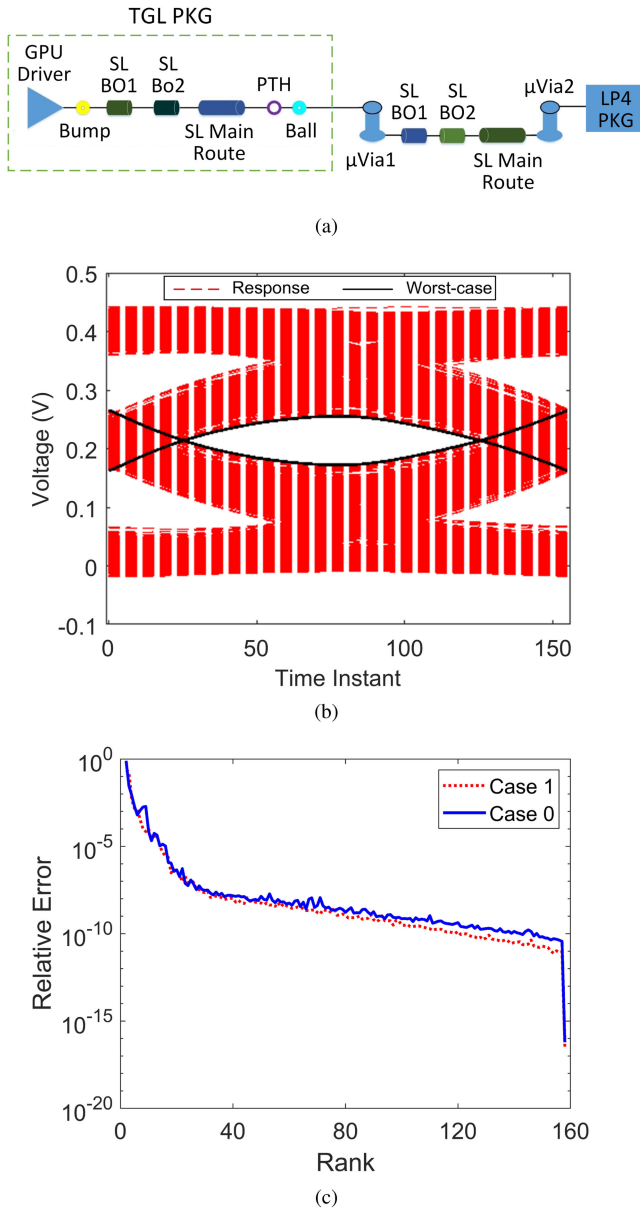


Fig. 2. (a) Channel with one victim and seven aggressors, whose bit number is 184. (b) Eye diagram and worst-case 1 and 0 generated by the proposed algorithm. (c) Relative error of the proposed algorithm for case 1 and 0.

VI. SIMULATION RESULTS

A. Low Power Double Data Rate fifth Generation (LPDDR5) Channel With 184 B

This example is for DQ (double input/output) write direction. It is an LPDDR5 channel, which is illustrated in Fig. 2(a). The driver is linear with $R_{on} = 33 \Omega$. The channel memory m for this example is 22 and the data rate is 6400 MT/s. We include one victim and seven aggressors to study both intersymbol interference (ISI) and crosstalk effect. To generate a complete eye diagram, 1 bit after current bit is added to each channel. Hence, the total number of bits considered in this simulation is 184. Although this channel is a linear channel, we treat it as a nonlinear one to test the capability of the proposed algorithm.

This example also serves a good testing case since the bit number is large while the reference result is known. In Fig. 2(b), we plot the eye diagram generated by the proposed method with the worst-case 1 and 0 highlighted in black. A brute-force simulation would require $2^{184} - 1$ simulations to produce the eye diagram. In contrast, the proposed algorithm only simulates 518 cases (determined by (30) no greater than 10^{-15}), costing 9.32 s and 43 MB memory only. The EH and EW are found to be 83.108210 mV, and 100.286885 ps, respectively, which are identical to the reference results. The relative error of the proposed algorithm, shown in (30), is plotted in Fig. 2(c) at each step for the current bit being 1 (Case 1) and 0 (Case 0), respectively. As can be seen, the error decreases step by step, and reaches a level of 10^{-15} within 160 simulations for each case. The computation stops when a prescribed accuracy is reached, and in this example, 10^{-15} accuracy.

B. Nonlinear LPDDR5 Channel With 13 B

The second example is a nonlinear LPDDR5 channel shown in Fig. 3(a). The receiver has an on-die termination (ODT) and a DRAM die resistance and a capacitance. The channel memory for this example is 13 and the data rate is 6400 MT/s. The sampling rate is 1 ps. We only include the victim in the analysis so that only ISI is considered.

In Fig. 3(b), we plot the eye diagram generated by the proposed method in comparison with that obtained from a brute-force nonlinear simulation, where blue color is for the proposed method, and red one is for the brute-force solution obtained from 8191 nonlinear simulations. They are shown to agree very well with each other. In the proposed method, only 163 nonlinear simulations are performed, which is much smaller than the 8191 nonlinear simulations. The exhaustive nonlinear simulation is feasible in this example because the channel memory is not large. In Fig. 3(b), we also show the worst-case 1 and 0 (black line) as compared to the reference solution (yellow line), which are in excellent agreement. The EH and EW of the proposed method are 248.841000 mV and 115.831671 ps, whereas the EH and EW of the reference solution are 248.841000 mV, and 116.448666 ps, respectively. Again, excellent agreement is observed. In this example, it only takes the proposed algorithm 1.04 s, and 9.6-MB memory to finish the simulation excluding the time of nonlinear circuit simulations. As for the time for nonlinear simulations, the speedup of the proposed method is the ratio of 8191 to 163, which is ~ 50 , since only 163 nonlinear simulations are performed. In Fig. 3(c), the relative error of the proposed algorithm is plotted at each step where rank is increased by 1. The relative error is shown to decrease with the rank, and within 80 simulations, the error is reduced to the level of 10^{-15} , which also indicates the good accuracy of the proposed algorithm.

C. Nonlinear Double Data Rate Fifth Generation (DDR5) Channel With 75 B

The third example is a 1DPC [one DIMM (dual in-line memory module) per channel] DDR5 channel for a test chip comprising package, board, connector, dimm, etc., as shown

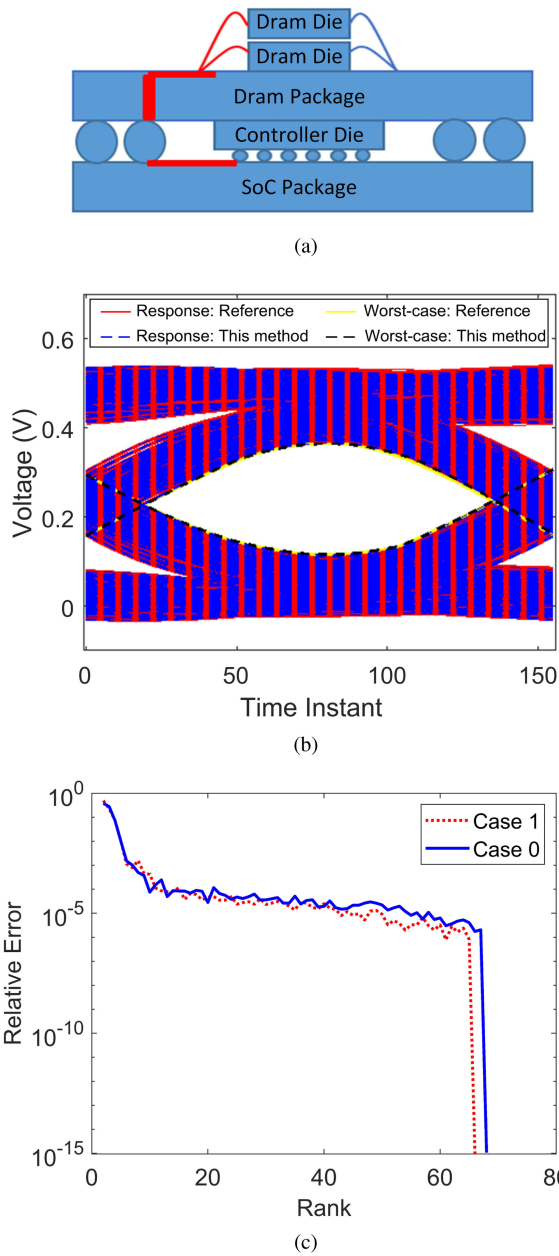


Fig. 3. (a) Nonlinear channel with a 13-b channel memory. (b) Eye diagram and worst-case 1 and 0 generated by the proposed algorithm overlaid with brute-force nonlinear simulations. (c) Relative error of the proposed algorithm.

in Fig. 4(a). This example has a driver whose nonlinearity can be adjusted by a parameter p . If $p = 1$, it means the driver is linear. If $p = 0$, then the driver is extremely nonlinear. We study both $p = 0.1$ and $p = 0.5$ to validate the performance of the proposed algorithm. The receiver has an ODT and a DRAM die resistance and a capacitance. The channel memory for this example is 25 and the data rate is 5600 MT/s. Each bit width is divided into 100 samples. We include one victim and two aggressors in the analysis to consider both ISI and crosstalk effects. Hence, the total number of bits is 75. Note that the number of bits in aggressors is not required to be the same as that of the victim. Each aggressor can have a different channel memory, whereas the proposed algorithm is equally applicable.

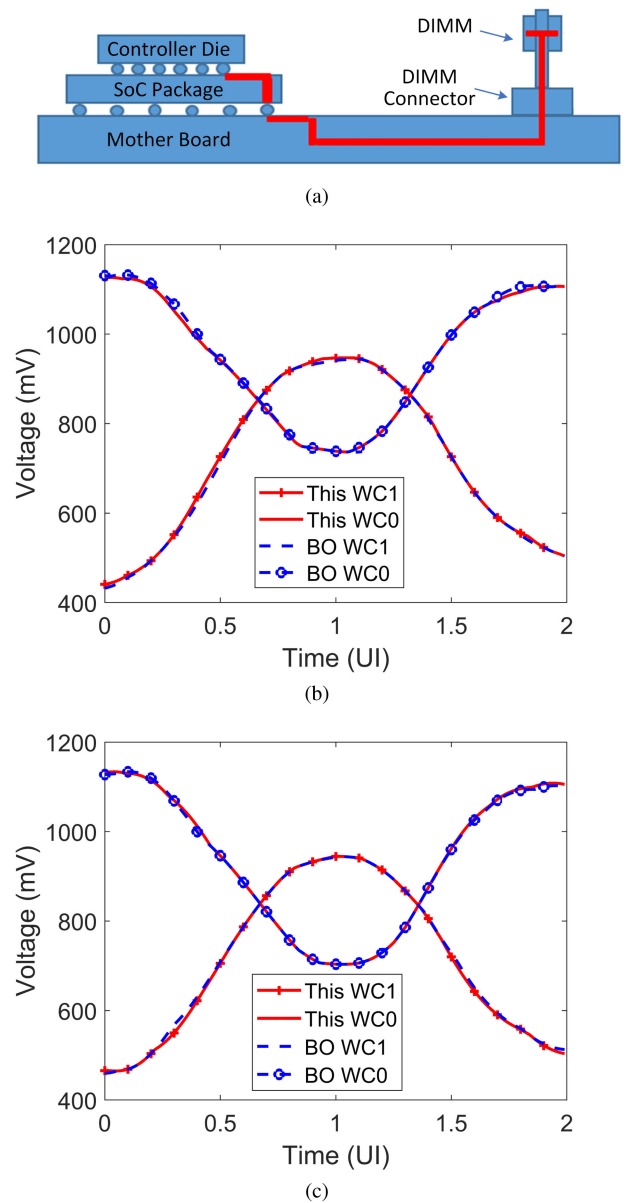


Fig. 4. (a) Nonlinear channel with one victim and two aggressors, whose bit number is 75. (b) Worst-case 1 and 0 generated by the proposed algorithm as compared with BO results for $p = 0.5$ (UI in the label means unit interval). (c) Worst-case 1 and 0 generated by the proposed algorithm as compared with BO results for $p = 0.1$.

Since an exhaustive nonlinear simulation is not feasible in this example, the BO is used to validate the proposed method. As shown in Fig. 4(b) and (c), for both $p = 0.5$ and $p = 0.1$, the proposed algorithm shows an excellent agreement with the BO results. The rank found for the $p = 0.5$ and $p = 0.1$ case is 380 and 402, respectively. Hence, using the proposed algorithm, one only needs to perform a few hundred nonlinear simulations, which can also be done in parallel, whereas a brute-force solution would require $2^{75} - 1$ nonlinear simulations. The time taken by the BO is also much larger than the time cost by the proposed algorithm. For example, for the $p = 0.1$ case, the BO used in this article takes about 50 times longer CPU time since it requires more nonlinear simulations to optimize the result.

VII. CONCLUSION

A fast method is developed for large-scale signal integrity analysis of nonlinear circuits. In this algorithm, we find the nonlinear responses to the $2^m - 1$ bit sequences of a channel can be represented by a rank- k matrix for a prescribed accuracy. Here, k is the number of distinct waveforms (responses) in the eye diagram. k is found to be in hundreds even for large m . After finding these responses, other $2^m - 1 - k$ responses do not need to be simulated since they overlap with the k responses in the eye diagram. We then develop a fast algorithm to find such a rank- k representation with guaranteed accuracy and a low complexity of $O(k^2r) \sim O(k^3)$ independent of 2^m . We have applied the proposed method to real-world large-scale nonlinear signaling analysis whose brute-force simulations are not feasible. These applications clearly demonstrate the accuracy, efficiency, and capacity of the proposed method. The method also significantly outperforms a black-box-type BO based technique in CPU run time and memory usage.

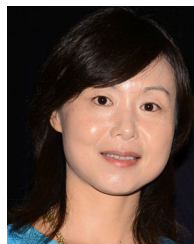
ACKNOWLEDGMENT

The authors would like to express their sincere thanks to Dr. S. Le Borne at the Hamburg Institute of Mathematics, University of Technology, Germany, for her valuable feedback and suggestions during the review of this article.

REFERENCES

- [1] B. K. Casper, M. Haycock, and R. Mooney, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling schemes," in *Proc. Dig. Tech. Papers IEEE Symp. VLSI Circuits*, Jun. 2002, pp. 54–57.
- [2] V. Stojanovic and M. Horowitz, "Modeling and analysis of high-speed links," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2003, pp. 589–594.
- [3] R. Shi, W. Yu, Y. Zhu, C. Cheng, and E. S. Kuh, "Efficient and accurate eye diagram prediction for high speed signaling," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 2008, pp. 655–661.
- [4] A. Tsuchiya, M. Hashimoto, and H. Onodera, "Optimal termination of on-chip transmission-lines for high-speed signaling," *IEICE Trans. Electron.*, vol. E90-C, no. 6, pp. 1267–1273, Jun. 2007.
- [5] H. Zhu, C. K. Cheng, A. Deutsch, and G. Katopis, "Predicting and optimizing jitter and eye-opening based on bitonic step response," in *Proc. IEEE Elect. Perform. Electron. Packag.*, Oct. 2007, pp. 155–158.
- [6] K. Xiao, B. Lee, and X. Ye, "A flexible and efficient bit error rate simulation method for high-speed differential link analysis using time-domain interpolation and superposition," in *Proc. IEEE Int. Symp. Electromagn. Compat.*, 2008, pp. 1–6.
- [7] Y. Chu, R. Chakraborty, R. Friar, and Z. Yang, "Analytical bit-error-rate analysis for multi-tone sinusoidal jitter from power supply noise," in *Proc. IEEE Int. Symp. Electromagn. Compat.*, 2016, pp. 267–270.
- [8] Y. Chu *et al.*, "FastBER: A novel statistical method for arbitrary transmitter jitter," DesignCon, 2017.
- [9] W. T. Beyene, "Peak distortion analysis of nonlinear links," in *Proc. IEEE 22nd Conf. Elect. Perform. Electron. Packag. Syst.*, 2013, pp. 169–172.
- [10] 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.
- [11] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. New York, NY, USA: Wiley, 1980.
- [12] D. Jiao and J. Zhu, "Fast method for accurate and efficient nonlinear signaling analysis," *IEEE Trans. Electromagn. Compat.*, vol. 59, no. 4, pp. 1312–1319, Aug. 2017.
- [13] S. A. Goreinov, E. E. Tyrtyshnikov, and N. L. Zamarashkin, "A theory of pseudoskeleton approximations," *Linear Algebra Appl.*, vol. 261, pp. 1–22, 1997.
- [14] S. Börm, L. Grasedyck, and W. Hackbusch, "Hierarchical matrices," *Lecture Note 21, Max Planck Inst. Math. Sci.*, Leipzig, Germany, 2003.

- [15] Bayesian Optimization. [Online]. Available: <https://www.mathworks.com/help/stats/bayesian-optimization-algorithm.html>
- [16] R. Baptista and M. Poloczek, "Bayesian optimization of combinatorial structures," 2018, *arXiv:1806.08838*.



Dan Jiao (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2001.

She then was with the Technology Computer-Aided Design (CAD) Division, Intel Corporation, until September 2005, as a Senior CAD Engineer, Staff Engineer, and 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 three book chapters and more than 300 papers in refereed journals and international conferences. Her current research interests include computational electromagnetics, high-frequency digital, analog, mixed-signal, and RF integrated circuit 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 bioelectromagnetics.

Dr. Jiao has served as the reviewer for many IEEE journals and conferences. She is an Associate Editor for the IEEE TRANSACTION ON COMPONENTS, PACKAGING, AND MANUFACTURING TECHNOLOGY, and an Associate Editor for the IEEE JOURNAL ON MULTISCALE AND MULTIPHYSICS COMPUTATIONAL TECHNIQUES. She served as the General Chair of the 2019 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization, Boston, MA, USA. She was selected as an IEEE MTT-Society Distinguished Microwave Lecturer in 2020. She was a recipient of Intel's 2019 Outstanding Researcher Award and 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. She was among the 21 women faculty selected across the country as the 2014–2015 Fellow of Executive Leadership in Academic Technology and Engineering at Drexel, a national leadership program for women in the academic STEM fields. She has been named a University Faculty Scholar by Purdue University since 2013. She was among the 85 engineers selected throughout the nation for the National Academy of Engineering's 2011 US Frontiers of Engineering Symposium. She was the recipient of 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 Divisional Achievement Award, the Intel Corporation's Technology CAD Divisional Achievement Award, the 2002 Intel Corporation's Components Research the Intel Hero Award, the Intel Corporation's LTD Team Quality Award, and the 2000 Raj Mittra Outstanding Research Award presented by the University of Illinois at Urbana-Champaign.

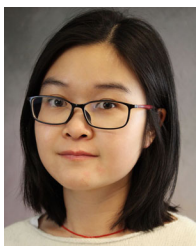


Yuhang Dou (Member, IEEE) received the B.S. degree from the Nanjing University of Science and Technology, Nanjing, China, in 2012, and the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, in 2019, both in electrical engineering.

She is currently a Postdoctoral Fellow with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA. Her current research interests include fast signal integrity analysis of large-scale nonlinear/linear circuits and systems, minimal-order model of large-scale electromagnetic

problems, and physics-based circuit-domain modeling methods for radiation and high-speed microwave problems.

Dr. Dou was the recipient of the first runner up awards of the IEEE Hong Kong AP/MTT Postgraduate Conference in 2015, 2018 and second runner up awards in 2016. She was also the recipient of an Honorable Mention Award from the 2015 IEEE NEMO conference.



Jin Yan received the B.S. degree in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 2012, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 2016.

She has been a Software Engineer with the Intel Corporation, Hillsboro, OR, USA, since 2016. Her current research interests include signal and power integrity, computational electromagnetics, high-performance very-large-scale integration CAD,

and fast and high-capacity numerical methods.

Dr. Yan was a recipient of the Honorable Mention Award from the IEEE International Symposium on Antennas and Propagation in 2015 and the Best Student Paper Award Finalist from the IEEE MTT-S International Microwave Symposium in 2016.



Jianfang (Olena) Zhu 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 currently a Principal Engineer with Intel Corporation, Hillsboro, OR, USA, and a System Architect focusing on end-to-end computer system design optimization and innovation. She initiated and co-leads the corporate-level strategy on augmented

intelligence (machine + human intelligence) to transform end-to-end design tools and methodologies. She spearheads the teams across organizations to drive highly optimized design at scale, through augmented intelligence, from silicon to intelligent end point device. Her team has deployed multiple augmented-intelligence-based tools and reduced the design time by over 90% in areas, such as SoC floorplan and board component placement. She also oversees the Intel Evo Platform/Project Athena battery life pillar and drives power and performance optimization. Her team has increased the power efficiency to support 20+ hours of local video play back, delivered the real-world all-day battery life experience on 22+ Project Athena systems in 2019 and launched the Intel Evo platform in 2020. She also spearheads multiple university collaborations to develop advanced algorithms for signal integrity analysis of nonlinear high-speed IOs and high-dimensional global optimization for system design optimization. She has authored one book chapter and 40+ papers in refereed journals and international conferences. She has 20+ patents granted or pending.



Adam Norman was born in 1969. He received the B.S., M.S., and Ph.D. degrees, all in electrical engineering from Michigan State University (MSU), East Lansing, MI, USA, in 1992, 1993, and 1996, respectively.

He performed research in theoretical and experimental transient electromagnetic scattering, as a graduate student with MSU. He has been a Principal Engineer with Intel Corporation, Hillsboro, OR, USA, since 2000, where he has worked in the areas of signal integrity/power integrity and EMC. His primary research interest includes development of the tools and methodologies for efficient

Signal/power integrity analysis.