POLITECNICO DI TORINO
Repository ISTITUZIONALE

Macromodel-Based Iterative Solvers for Simulation of High-Speed Links With Nonlinear Terminations

(Article begins on next page)

25 April 2024

Macromodel-based iterative solvers for simulation of high-speed links with nonlinear terminations

Salvatore Bernardo Olivadese, Stefano Grivet-Talocia, *Senior Member, IEEE*, Claudio Siviero, Dierk Kaller

Abstract—Data transmission on high-speed channels may be affected by several undesired effects, including coupling from nearby interconnects, dispersion, losses, signal reflections from terminations and from internal discontinuities, and nonlinear/dynamic effects of drivers and receivers. The latter are often neglected, leading to very fast solvers, whose results may however be questionable when driver/receiver nonlinearities are important. This paper presents a framework for the transient analysis of complex high-speed channels with arbitrary nonlinear termination circuits. The approach is based on decoupling channel and terminations through a scattering-based Waveform Relaxation (WR) formulation. The channels are here cast as delay-rational macromodels, which are solved in discrete time domain through fast delayed recursive convolutions. The terminations can be either arbitrary circuits, solved by SPICE, or nonlinear behavioral macromodels, which are here formulated in discrete-time scattering representations. In order to overcome the known convergence issues of standard WR methods, we apply here more general iterative solution schemes, such as GMRES and BiCGSTAB, integrated into inexact Newton iterations, obtaining a set of numerical schemes with guaranteed convergence. The excellent performance of the proposed approach is illustrated on a large set of benchmarks.

Index Terms—High-speed interconnects, Scattering, macro-modeling, rational approximation, delay extraction, behavioral modeling, waveform relaxation, inexact Newton methods

I. INTRODUCTION

The qualification process of a high-speed channel for a target data rate inevitably requires extensive numerical simulations [1]–[3]. In fact, transient link analyses are performed since early design stages in order to assess all possible signal degradation effects, including crosstalk and coupling from nearby interconnects, dispersion, losses, signal reflections from terminations and from internal discontinuities, and nonlinear/dynamic effects of drivers and receivers. The usual computed metrics are vertical and horizontal eye diagram openings and Bit Error Rate (BER) statistical estimates.

The reliability of the above metrics ultimately depends on how representative are the models for the channel and its terminations used in the numerical simulations. Channels are electrically long distributed interconnect structures, whose parasitic effects are best represented in the frequency domain. Conversely, drivers and receivers are best represented by their

Manuscript received ...; revised ...

S.B. Olivadese, S. Grivet-Talocia and C. Siviero are with the Department of Electronics and Telecommunications, Politecnico di Torino, Torino 10129, Italy (e-mail: stefano.grivet@polito.it, salvatore.olivadese@polito.it, claudio.siviero@polito.it).

D. Kaller is with IBM Systems and Technology Group, IBM Deutschland Research & Development GmbH, Böblingen, Germany (e-mail: dkaller@de.ibm.com)

transistor-level schematics. The combination of frequency-domain descriptions of distributed structures with transistor-level termination circuits is well-known to be troublesome for standard circuit solvers.

The above difficulties led to several different approaches to simplify the link simulation process. A very popular approach [4] neglects completely the nonlinear characteristics of drivers and receivers. If linear drivers and receiver models are assumed, the characterization can be completely performed in the frequency domain, followed by an inverse Fourier Transform to compute the transient response to an elementary pulse. Eye diagram and related statistics are then computed in extremely fast runtimes leveraging on linear superposition and convolution. If drivers and receivers can be safely approximated through linear characteristics, no better and faster approach is possible.

In case the nonlinear effects of drivers and receivers are important, the above linear approach looses validity. Superposition does not apply, and the resulting statistics become possibly flawed, if not validated by a more accurate "golden" simulation. In principle, it is possible to perform brute-force transient analyses using transistor-level models of drivers and receivers. This approach is however not practical due to the overwhelming circuit complexity. An example will be provided in Sec. VII. Moreover, transistor-level schematics are rarely available due to intellectual property restrictions.

These difficultes led to many alternative approaches in the field of behavioral driver/receiver modeling. We can cite the various IBIS (lookup table based) models [5], Mπlog [6], [7] and derived/improved models [8]–[10], Volterra-Laguerre models [11], [12], Neural Network models [13], and more recently X-parameter-based models [14]. Most of these approaches are compatible with standard circuit solvers, thus enabling more efficient transient analyses.

Most modern circuit solvers allow the direct inclusion of tabulated frequency-domain Scattering parameters defining a channel multiport model. This data is internally converted to a set of channel impulse responses, which are in turn used as linear convolution kernels during transient analysis. Due to long-tail memory effects induced by losses, dispersion, and reflections from discontinuities, this approach may also be inefficient. For this reason, channel macromodeling has also been pursued with some success, both in the standard rational form [15]–[19] and in the more advanced delay-rational form [20]–[28], the latter being able to account for explicit propagation delays embedded in the model transfer functions. An example of macromodel-based channel simulation using a standard circuit solver is available in [29].

Transient analysis by standard circuit solvers using channel

and termination macromodels may still be inefficient for systematic transient analysis, e.g., due to bad scalability with the number of coupled links. For this reason, Waveform-Relaxation (WR) techniques have been recently revitalized, due to their potential for dramatic speedup. Originally, the WR framework was introduced for general circuit simulation [30]–[37], building on the consideration that weakly coupled subsystems can be efficiently solved independently after a suitable decoupling process, provided that an iterative scheme corrects for the missing coupling terms, cast as relaxation sources. It was then observed that the WR idea is particularly well suited for transmission line structures [38]–[42], whereas the partitioning can be either longitudinal, i.e., decoupling a line from its terminations [43], [44], or transverse, i.e., decoupling a multiconductor line into separate scalar lines [45], [46]. Decoupling enables straightforward algorithm parallelization for deployment on multicore computing architectures [47].

WR-based transient analysis of complex channels (including irregular routing through packages, cards, boards, connectors and via fields) rather than simpler uniform transmission lines has been proposed in [48], [49]. Unfortunately, the analysis in [49] shows that when transverse partitioning is introduced to improve simulation efficiency, the convergence of the WR scheme is only conditional and cannot be guaranteed in all cases, despite some improvements based on possibly frequency-dependent over-relaxation have been documented [50], [51]. This consideration motivated the approach in [52], [53], where more robust iterative solvers have been suggested to replace the fixed point iteration typical of basic WR implementations.

This work extends the preliminary results of [53], by introducing a general framework for iterative solution of high-speed channels with nonlinear terminations. A suitable time discretization process casts the transient simulation problem as the solution of a possibly very large (but very sparse) nonlinear algebraic system. The unknowns collect the discretized scattering wave variables at all channel ports and at all time steps to be computed. The system is never formed explicitly, but the evaluation of the (nonlinear) residual is based on the successive evaluation of a channel operator and a termination operator. The former boils down to fast recursive convolutions arising from a delay-rational channel macromodel. The latter amounts to the solution of the (decoupled) termination circuits excited by known incident scattering waves. The solution of the nonlinear system is achieved through an inexact Newton iteration [54], [55] combined with a Krylov subspace iteration. Two implementations based on the Generalized Minimal RESidual (GMRES) scheme [56] and the BiConjugate Gradient STABilized (BiCGSTAB) scheme [57] have been realized and are here documented. The main new contribution of this formulation with respect to earlier approaches is the guaranteed convergence, albeit with a possibly large number of iterations. However, the numerical results show that, even in the most challenging situations, convergence is always reached within very few (up to 5-6) iterations.

The proposed schemes are based, as basic WR, on decoupling the channel from its terminations. Therefore, dedicated solvers can be used for channel and termination operators.
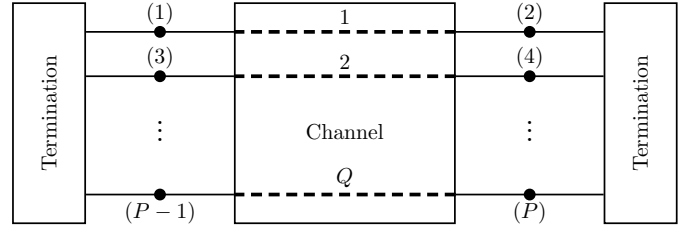

Fig. 1. Topology of the system under investigation. Each dot represents an electrical port.
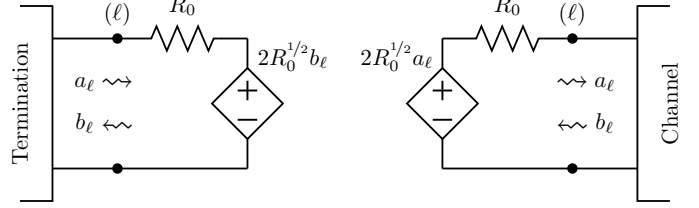

Fig. 2. Definition of scattering waves and longitudinal partitioning sources at a single interface port (ℓ) between channel and terminations.

In particular, we present two different implementations. The first uses an external call to SPICE to solve for the terminations. This approach requires a suitable interface to exchange transient relaxation sources between SPICE and the main iterative solver, thus limiting efficiency. However, this scheme is useful when the terminations are known as complex and even encrypted netlists. We use this implementation to validate the convergence of proposed schemes in the most critical situations, i.e., with strongly nonlinear and mismatched terminations. A novel and more application-oriented implementation is instead based on behavioral macromodels of drivers and receivers. In particular, we adopt a new scattering-based formulation extending the results of [63], by suitably modifying the Mπlog macromodel format [6], [7], in order to obtain termination equations that are compatible with our simulation setup. We show excellent convergence and performance when applying the solver to various industrial channels from [29], driven by production-level driver models.

II. PROBLEM STATEMENT

We consider the general structure depicted in Fig. 1. The channel is formed by Q coupled point-to-point interconnects, with ports $(2i - 1)$ and $(2i)$ denoting near and far end of the i-th link, for $i = 1, \ldots, Q$. The channel is known by frequency samples of its $P \times P$ (with $P = 2Q$) scattering matrix $\hat{\boldsymbol{S}}_\nu = \hat{\boldsymbol{S}}(\mathrm{j}\omega_\nu)$ at frequencies $\{\omega_\nu, \nu = 1, \ldots, L\}$ suitably spread over the frequency band of interest.

No particular structure is imposed on the terminations, which can be formed by any arbitrary circuit, including complex transistor-level schematics. The terminations include some transient sources for launching signals into the channel. Our objective is to compute the transient voltages and currents at the channel ports. More precisely, we will assume a uniform time discretization $t_k = k\,\delta t$, and we will compute the time samples of port voltages $v_{\ell,k} \approx v_\ell(t_k)$ and currents $i_{\ell,k} \approx i_\ell(t_k)$ for $\ell = 1, \ldots, P$ and $k = 0, \ldots, K$ up to a prescribed maximum time $T = K\,\delta t$.

Some remarks on notation. Standard italic fonts x will denote scalar variables, boldface italic fonts $\boldsymbol{x}(t)$ or $\boldsymbol{X}(\mathrm{j}\omega)$ will represent time- or frequency-dependent port variables collected in vectors or matrices with leading size P, whereas boldface normal fonts $\mathbf{x}$ will denote vectors collecting discrete-time samples. Operators on these vectors will be denoted by calligraphic fonts $\mathcal{X}$.

III. FORMULATION

A. Scattering-based formulation

Our formulation is based on scattering wave variables. With reference to Fig. 2, we denote with $a_\ell(t)$ the transient scattering wave that is incident into port (ℓ) of the channel, and with $b_\ell(t)$ the corresponding reflected wave. Clearly, $b_\ell(t)$ can also be regarded as the incident wave into port (ℓ) of the termination network, with $a_\ell(t)$ being the corresponding reflected wave. Upon time discretization, we can form the vectors

$$\mathbf{a}^\ell = (a_{\ell,0}, a_{\ell,1}, \ldots, a_{\ell,K})^\mathsf{T}, \quad \mathbf{b}^\ell = (b_{\ell,0}, b_{\ell,1}, \ldots, b_{\ell,K})^\mathsf{T} \tag{1}$$

collecting all $K+1$ time samples of the scattering waves at port (ℓ). These vectors are then collected for all ports

$$\mathbf{a} = (\mathbf{a}^1; \mathbf{a}^2; \ldots; \mathbf{a}^P), \quad \mathbf{b} = (\mathbf{b}^1; \mathbf{b}^2; \ldots; \mathbf{b}^P), \tag{2}$$

where ";" denotes vertical stacking. We have $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$, where $N = P(K+1)$.

B. Channel characterization

Denoting with $\boldsymbol{S}(s)$ the Laplace-domain scattering matrix of the channel, and with $\boldsymbol{A}(s), \boldsymbol{B}(s) \in \mathbb{C}^P$ the corresponding incident and reflected wave vectors, we have by definition $\boldsymbol{B}(s) = \boldsymbol{S}(s)\boldsymbol{A}(s)$. Inverse Laplace transform leads to the continuous-time convolution

$$\boldsymbol{b}(t) = \int_0^\infty \boldsymbol{h}(t-\tau)\boldsymbol{a}(\tau)\mathrm{d}\tau, \tag{3}$$

where $\boldsymbol{a}(t), \boldsymbol{b}(t) \in \mathbb{R}^P$ are transient scattering wave vectors and where $\boldsymbol{h}(t)$ is the matrix of scattering impulse responses of the channel.

1) Direct convolution: The simplest approach to obtain a discrete-time representation of the channel operator is to discretize (3) with the prescribed time step δt. Writing the resulting expression componentwise leads to

$$b_{\ell,k} = \sum_{m=1}^{P} \sum_{r=0}^{k} \delta t\, h_{\ell,m}(t_k - \tau_r) a_{m,r}, \tag{4}$$

where $\tau_r = r\,\delta t$. The discrete-time impulse response samples can be obtained, e.g., by an inverse Fast Fourier Transform (FFT) applied to the set of raw scattering frequency samples $\hat{\boldsymbol{S}}_\nu$. This operation should be performed with care, in order to minimize aliasing and truncation artifacts due to the finite-size and bandlimited nature of the available samples. We refer the Reader to [64]. In case the time step δt is matched to the highest frequency $\omega_{\max}$ in the input scattering data, we expect a total number of impulse response samples $h_{\ell,m}(t_k)$ that is equal to the number L of raw frequency samples, unless interpolation, extrapolation or other postprocessing operations are performed. This implies that the equivalent "memory" of the convolution is L, since only up to L terms are considered in (4) for each component (ℓ, m).

2) Delayed Rational Macromodeling: A more advanced approach for the discretization of (3) is to resort to an analytic inverse Laplace transform, preceded by a frequency-domain approximation. For electrically-long structures such as the channels under investigation, the state of the art approach is to approximate each component of the scattering matrix $\boldsymbol{S}(s)$ with a Delayed Rational Macromodel (DRM)

$$S_{\ell,m}(s) \approx \sum_{d=1}^{N_d^{\ell,m}} \sum_{n=1}^{N_p^{\ell,m}} \frac{R_{dn}^{\ell,m}}{s - p_{dn}^{\ell,m}} e^{-s\tau_d^{\ell,m}} + D^{\ell,m} \tag{5}$$

whose coefficients (poles $p_{dn}^{\ell,m}$, residues $R_{dn}^{\ell,m}$, and delays $\tau_d^{\ell,m}$) are determined through specialized curve fitting algorithms. This process is well documented in the literature and is not repeated here. We refer the Reader to [20]–[29]. As a result, the continuous-time convolution (3) is approximated as a one-tap recursive convolution

$$b_{\ell,k} = \sum_{m=1}^{P} \sum_{d=1}^{N_d^{\ell,m}} \sum_{n=1}^{N_p^{\ell,m}} \Big\{ \alpha_{dn}^{\ell,m} b_{m,k-1} + \beta_{dn}^{\ell,m} a_{m,k-\bar{k}_d^{\ell,m}} + \gamma_{dn}^{\ell,m} a_{m,k-1-\bar{k}_d^{\ell,m}} \Big\}, \tag{6}$$

where $\bar{k}_d^{\ell,m} = \left\lfloor \tau_d^{\ell,m}/\delta t \right\rfloor$ is the time index corresponding to the individual delay $\tau_d^{\ell,m}$. The precise definition of the coefficients in (6) in terms of the coefficients of (5) is available in [49]. The DRM formulation (6) is competitive with respect to the direct convolution (4) when the total number of coefficients $\{\alpha_{dn}^{\ell,m}, \beta_{dn}^{\ell,m}, \gamma_{dn}^{\ell,m}\}$ is less than $P^2 L$.

3) Discrete-time channel operator: Both the direct convolution formulation (4) and the delayed recursive convolution (6) can be cast in the compact operator form

$$\mathbf{b} = \mathcal{H}\mathbf{a}, \tag{7}$$

where $\mathbf{a}, \mathbf{b}$ are as in (2), and $\mathcal{H} \in \mathbb{R}^{N \times N}$ is a square matrix that represents an algebraic description of the linear channel operator that, applied to a known set of incident discrete-time scattering waves, returns the corresponding reflected wave samples. This matrix description is only formal, since $\mathcal{H}$ is never formed explicitly. Application of this operator is performed componentwise using (4) or (6), depending on the adopted formulation.

C. Terminations

Let us now consider the termination networks. In order to properly setup the interface with the channel, we consider a scattering-based input-output formulation of the termination equations, where the incident scattering waves $\boldsymbol{b}(t)$ are considered as inputs, and the reflected waves $\boldsymbol{a}(t)$ are the corresponding outputs. The proper setup for this characterization is depicted in Fig. 2.

The termination equations can be formulated as nonlinear state-space equations

$$\dot{\boldsymbol{x}}(t) = \mathcal{F}(\boldsymbol{x}(t), \boldsymbol{b}(t), \boldsymbol{u}(t)), \tag{8}$$

$$\boldsymbol{a}(t) = \mathcal{G}(\boldsymbol{x}(t), \boldsymbol{b}(t), \boldsymbol{u}(t)), \tag{9}$$

where $\boldsymbol{x}$ denotes an internal state vector, $\boldsymbol{u}$ is a vector collecting the independent sources, and $\mathcal{F}, \mathcal{G}$ are the nonlinear state and output maps. Considering again a time discretization with uniform step δt, we can solve numerically the nonlinear state equation (8) for the discrete time samples of the state vector $\boldsymbol{x}(t_k)$ which, inserted in the output equation (9), provides the discrete-time samples of the reflected waves $\boldsymbol{a}(t_k)$.

The above solution process can be cast in a compact operator form as

$$\mathbf{a} = \mathcal{T}(\mathbf{b}; \mathbf{u}), \tag{10}$$

where $\mathbf{a}, \mathbf{b}$ are as defined in (2), and where $\mathbf{u}$ collects all time samples of all independent sources over the time span of the simulation. The operator $\mathcal{T}$ is purely algebraic, and represents the sequence of operations that, given a set of discrete time samples of the independent sources embedded in the terminations and the corresponding samples of the incident scattering waves into the terminations, returns the time samples of the corresponding reflected waves. Under a practical standpoint, the operator $\mathcal{T}$ performs the same steps that a circuit solver of the SPICE class would execute when solving for the termination networks loaded at their ports as in Fig. 2 (hence decoupled from the channel), and exporting the computed output scattering waves $\boldsymbol{a}(t)$ at uniformly sampled time steps $t_k = k\,\delta t$.

D. Waveform Relaxation approaches

Collecting discretized channel and termination equations, we obtain the nonlinear system

$$\begin{cases} \mathbf{b} &= \mathcal{H}\mathbf{a}, \\ \mathbf{a} &= \mathcal{T}(\mathbf{b}; \mathbf{u}). \end{cases} \tag{11}$$

Earlier WR formulation approaches, as summarized in [49] setup fixed-point iterative schemes to compute the solution of this system by simple forward evaluations of the channel and termination operator. In particular, Longitudinal Partitioning (LP) approaches start at some initial guess $\mathbf{a}_0$ and apply the two operators sequentially as

$$\begin{cases} \mathbf{b}_\mu &= \mathcal{H}\mathbf{a}_{\mu-1} \\ \mathbf{a}_\mu &= \mathcal{T}(\mathbf{b}_\mu; \mathbf{u}). \end{cases} \quad \mu = 1, 2, \dots \tag{12}$$

until the solution stabilizes. Transverse Partitioning (TP) approaches take further advantage of the small couplings between individual links to split the channel operator $\mathcal{H} = \mathcal{D} + \mathcal{C}$ into a "diagonal" part $\mathcal{D}$ that neglects inter-channel couplings, and a remainder part $\mathcal{C}$. The latter becomes a relaxation (equivalenty, a correction) source $\mathbf{w}$ within the following iterative loop

$$\begin{cases} \mathbf{w}_{\mu-1} &= \mathcal{C}\mathbf{a}_{\mu-1} \\ \mathbf{b}_\mu &= \mathcal{D}\mathbf{a}_\mu + \mathbf{w}_{\mu-1} \\ \mathbf{a}_\mu &= \mathcal{T}(\mathbf{b}_\mu; \mathbf{u}) \end{cases} \quad \mu = 1, 2, \dots \tag{13}$$

where the last two rows are solved concurrently at the μ-th iteration, e.g., by an inner WR-LP iteration, leading to the so-called WR-LPTP scheme [49].

In all above cases, the WR scheme can be interpreted as a fixed point iteration that, starting from some initial condition $\mathbf{a}_0$, produces a sequence of estimates $\mathbf{a}_0 \to \mathbf{a}_1 \to \mathbf{a}_2 \to \dots$ that hopefully converges to the correct solution $\mathbf{a}_*$. Each new iteration can be expressed as

$$\mathbf{a}_\mu = \mathcal{P}(\mathbf{a}_{\mu-1}; \mathbf{u}) \tag{14}$$

where $\mathcal{P}$ is the iteration operator. Convergence holds only if $\mathcal{P}$ is a contraction, as discussed in [30], [36], [49]. This condition is not always verified [50], [51], and more robust approaches are required.

IV. PROPOSED ITERATION

Let us consider the discretized system (11). Upon elimination of vector $\mathbf{a}$, we obtain

$$\mathbf{b} - \mathcal{H}\mathcal{T}(\mathbf{b}; \mathbf{u}) = \mathcal{N}(\mathbf{b}) = \mathbf{0}. \tag{15}$$

where $\mathcal{N}$ is the nonlinear residual or simply the residual. Since the source vector $\mathbf{u}$ can be considered as constant, it will be omitted in the following derivations. The classical approach for the approximate solution of (15) is by means of the Newton sequence

$$\mathbf{b}_{\mu+1} = \mathbf{b}_\mu - [\mathcal{J}(\mathbf{b}_\mu)]^{-1}\mathcal{N}(\mathbf{b}_\mu), \tag{16}$$

with μ iteration number and $\mathcal{J}(\mathbf{b}_\mu)$ the Jacobian matrix of $\mathcal{N}$ evaluated at $\mathbf{b}_\mu$. Equation (16) is the first order approximation for the nonlinear system $\mathcal{N}(\mathbf{b})$ near $\mathbf{b}_\mu$, thus it provides good accuracy only in the neighborhood of the linearization point $\mathbf{b}_\mu$. A more robust implementation of the Newton iteration involves the following steps

1) compute an estimate of $\mathcal{N}(\mathbf{b}_\mu)$;
2) find the numerical solution $\mathbf{s}_n$ of the linear system

$$\mathcal{J}(\mathbf{b}_\mu)\mathbf{s}_\mu = -\mathcal{N}(\mathbf{b}_\mu); \tag{17}$$

3) construct $\mathbf{b}_{\mu+1} = \mathbf{b}_\mu + \lambda_\mu \mathbf{s}_\mu$, where the step length λ_μ is selected to guarantee a decrease in the residual error $\|\mathcal{N}(\mathbf{b}_{\mu+1})\| < \|\mathcal{N}(\mathbf{b}_\mu)\|$.

The second step is the most time consuming, since the evaluation of the Jacobian matrix and its inversion are prohibitive for large-scale systems as in our application. However, an approximate solution for the descent direction $\mathbf{s}_\mu$ can be computed to satisfy the inequality

$$\|\mathcal{J}(\mathbf{b}_\mu)\mathbf{s}_\mu + \mathcal{N}(\mathbf{b}_\mu)\| \le \eta_\mu \|\mathcal{N}(\mathbf{b}_\mu)\| \tag{18}$$

usually denoted as the inexact Newton condition [55], through an iterative scheme that does not even require to compute, store and factor $\mathcal{J}(\mathbf{b}_\mu)$. The action of the Jacobian on a given vector is approximated as through a forward difference. The parameter η_μ is determined as discussed in [54].

Several iterative methods are available to compute the step $\mathbf{s}_\mu$ with (18) as a stop condition. We adopt here two among the most popular techniques, namely the GMRES (Generalized Minimal RESidual) [56] and the BiCGSTAB (BiConjugate Gradient STABilized) [57] methods. Both schemes provide

estimates of the solution based on suitably constructed Krylov subspaces, used both for projection and for the determination of appropriate search directions for residual minimization. The main complexity of these solvers is dictated by the size q of such Krylov spaces. Using the two above methods to solve for (18) leads to two implementations, denoted in the following as WR-NGMRES and WR-BiCGSTAB, respectively. Further, we test the nonlinear preconditioner for inexact iterative methods proposed by [62]. We will see that for the problem at hand, resorting to such a preconditioning technique is generally not advantageous due to the very fast convergence rate that is already achieved with the standard Newton-Krylov iterations.

A. Initialization, convergence, and termination

In order to ensure good convergence properties for the Newton-based iterative schemes, some technical assumptions are required:

1) system (15) admits at least one solution $\mathbf{b}_*$;
2) the nonlinear operator $\mathcal{N}$ is Lipschitz continuous near the solution $\mathbf{b}_*$;
3) the Jacobian $\mathcal{J}(\mathbf{b}_*)$ is nonsingular.

Under the above assumptions, classic or inexact Newton methods converge to the correct solution $\mathbf{b}_*$ if the initial guess $\mathbf{b}_0$ for the iterative scheme is close to $\mathbf{b}_*$.

We ensure a good starting point for the Newton iterations by running a number μ_i of initial iterations of the WR-LP or WR-LPTP schemes outlined in (12)–(13) and in [49]. If μ_i is small, i.e., $\mu_i = 1$ or 2, this results in a very fast and sufficiently accurate initialization, without running into possible convergence issues. In our implementation, we keep on executing these initialization iterations by increasing μ_i, while monitoring the evolution of the error between two subsequent iterates, see (19) below. If a lack of convergence is detected at some iteration $\bar{\mu}$ by an error increase, the next iterations are performed via GMRES or BiCGSTAB schemes, initialized with the solution estimate $\mathbf{b}_{\bar{\mu}-1}$. This process achieves the condition $\mathbf{b}_{\bar{\mu}-1} \approx \mathbf{b}_*$ and standard results [58], [59] on local convergence of the adopted Newton method apply. In order to achieve global convergence, we use the linear search method denoted as the Armijo rule [60], which guarantees a good estimate of the optimal step length λ_μ towards the solution, avoiding oscillations in the convergence process.

In our implementation, the stopping condition of the iterative solver is based on checking the decrease in norm of the nonlinear residual (15) through

$$\|\mathcal{N}(\mathbf{b}_{\mu+1})\| \leq \tau_r \|\mathcal{N}(\mathbf{b}_\mu)\| + \tau_a, \qquad (19)$$

with absolute and relative thresholds. In all numerical experiments, these thresholds are set to $\tau_r = \tau_a = 10^{-4}$. It will be shown that in all tested configurations, the proposed schemes always reach condition (19) in only 5–7 iterations.

B. DC initialization

Most often the terminations force a DC bias along the channel. Examples are differential drivers and/or pull-up/down receivers. For this reason, the transient scattering wave variables at the first time step $k = 0$ assume nonvanishing values

$\boldsymbol{a}_{DC} = \boldsymbol{a}(0)$ and $\boldsymbol{b}_{DC} = \boldsymbol{b}(0)$. We compute these initial DC bias levels before setting up the main iterations, by solving system (11) including only the unknowns at $t = 0$. In this situation, the channel operator $\mathcal{H}$ stores the entries of the scattering matrix of the channel at DC only, whereas the termination operator $\mathcal{T}$ becomes equivalent to a DC SPICE run using the intial conditions imposed by the embedded sources $\boldsymbol{u}(t)$ at $t = 0$.

Once the initial DC bias is computed, we redefine the main variables as the deviation of the transient scattering waves with respect to this bias,

$$\hat{\boldsymbol{a}}(t) = \boldsymbol{a}(t) - \boldsymbol{a}_{DC}, \quad \hat{\boldsymbol{b}}(t) = \boldsymbol{b}(t) - \boldsymbol{b}_{DC}, \qquad (20)$$

and the main solver is setup to calculate the corresponding discrete-time unknown vectors $\hat{\mathbf{a}}, \hat{\mathbf{b}}$. As a result, the first sample at $t = 0$ of the actual variables being computed by the solver is always vanishing, leading to a simplified treatment of the discretized convolution (6).

C. Differences with standard time-stepping solvers

The proposed solver is completely different from standard circuit solvers based on time-stepping, such as SPICE. A SPICE solver computes all circuit variables at each time step by solving a nonlinear algebraic system through Newton iterations. The number of Newton solves equals therefore the number of time steps to be computed. In other words, the time stepping scheme is implemented as an outer loop, whereas the Newton iteration is embedded in an inner loop.

Our solver basically reverses inner and outer loop. The unknowns that are solved for are large-size vectors, collecting all time samples. These unknowns are found through a single outer loop based on inexact Newton iterations combined with the selected Krylov solver. The outcome of each Newton iteration is an estimate for the complete set of time samples, which are all computed at once. Since these samples describe port signals, whose estimates are refined through iterations, we consider the proposed schemes as belonging to the class of WR solvers, albeit based on more robust iterative solvers.

V. VALIDATION RESULTS

In this section, we report a set of numerical results obtained on several test cases. The implementation that is benchmarked here is based on an external circuit solver [65] for solving the termination circuits. Therefore, the termination operator $\mathcal{T}(\mathbf{b}, \mathbf{u})$ consists of the following operations, which are repeated at each iteration

1) export the time samples in vector $\mathbf{b}$ to external files;
2) run an external SPICE solver on the termination network loaded by relaxation sources as in Fig. 2;
3) export from the SPICE solver the computed time samples in vector $\mathbf{a}$;

Since this implementation is far from being optimal, due to the need of exchanging time-dependent relaxation sources (via external files) between circuit and channel solvers, the main focus here is on accuracy and convergence of the outer iteration loop. Results for an all-macromodel based implementation that does not require external circuit solvers (the main objective

of this paper) will be reported in Sec. VII. Moreover, all test cases in this section were executed by disabling the automated initialization scheme (see Sec. IV-A), and by initializing the various Newton schemes via only $\mu_i = 1$ standard WR-LPTP iteration.

A. Channels

Several channel models are used to demonstrate and validate the proposed solver. We use here the same channels already introduced in [29], consisting of intra-node, inter-node, or node-to-peripheral electrical interconnects in high-end server systems. More details on these structures, as well as a complete documentation of the corresponding delay-rational macromodels, are available in [29], [51]. Under the electrical modeling standpoint, each channel consists of $Q = 9$ single-ended coupled links, resulting in $P = 18$ electrical interface ports. These structures are certified up to 2 Gbps data transmission, with proper terminations. The impedance level of each link is close to 50 Ω.

B. Terminations

The termination structures that we use in our benchmarks are deliberately designed to explore the practical behavior of the solver in challenging situations. In particular, we combine

1) strong nonlinear characteristics (close to piecewise linear), that nearly violate the last two assumptions for the local convergence made in Section IV-A; and
2) strongly mismatched terminations, so that undesired signal reflections are induced, causing oscillations in the port voltages and currents and thus making more difficult to obtain a good starting point for the iterative algorithm. In this way, the hypotheses for the global convergence are in danger, and the effectiveness of the Armijo rule [60] can be verified in practice.

Of course, since real drivers and receivers are properly designed to avoid the above effects, many of the proposed terminations may look (and actually are) unrealistic. A more practical simulation setup will be considered in Sec. VI, where the proposed solver is tested with macromodels of real production-level drivers.

Two termination blocks will be repeatedly used in the following. The first, denoted as NLA, is a simple overvoltage protection one-port circuit with a static characteristic depicted in Fig. 3. The second, depicted in Fig. 4 and later denoted as NLB, will be used as a nonlinear dynamic driver. In all cases, the time-varying sources embedded in the driver circuits will be defined as random bit sequences with $V_{\min} = 0$ V and $V_{\max} = 1.1$ V, rise and fall time of $t_r = t_f = 66$ ps, and pulse width $T_b = 500$ ps. Most of the results will be reported showing the first few bits of the computed transient voltages, to enhance visibility on the small waveform features, although all reported computing times refer to a full 500 bits simulation.

C. Test case 1

In this test, we consider a node-to-I/O channel with strongly mismatched terminations. The far end (even-numbered) ports
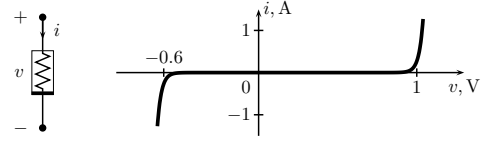

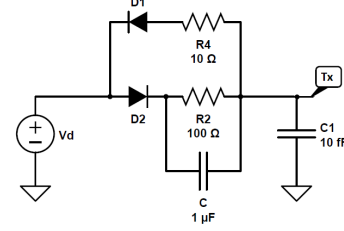Fig. 3. A nonlinear overvoltage protection circuit.

Fig. 4. A dynamic nonlinear driver. D1 and D2 are based on the default SPICE model for diodes.

are terminated into 1pF capacitances. The link is driven by a linear Thevenin data source with series 10Ω resistance on port 9, and by synchronous aggressor clock signals placed at all other near end ports (with internal impedance 1Ω). In addition, two NLA blocks are optionally connected in parallel to ports 2 and 10 to clip the received voltages and to illustrate the corresponding nonlinear effects on the solution.

Figure 5 shows the simulation results (only the first 20 ns are reported for clarity) obtained by the NGMRES solver, compared to reference results obtained by SPICE. This plot clearly illustrates the clipping effect of the nonlinearity. Despite the strong clipping, the accuracy of the proposed solver is excellent. Figure 6 shows the reduction of the nonlinear residual (15) through iterations, until (19) is satisfied. Results are compared for the linear and the nonlinear case, for NGMRES and BiCGSTAB implementations, with and without enabling the preconditioner. It is remarkable that the presence of the preconditioner has little effect on the number of iterations required for convergence. This behavior will be confirmed by all other examples that follow. All implementations show a very fast convergence, which reaches the target stopping threshold in up to 6 iterations.

D. Test case 2

In this test case, the driver circuit (applied to port 9 of the channel) is the nonlinear dynamic element NLB of Fig. 4. Terminations on the aggressor links and receivers are linear but again strongly mismatched with respect to the channel impedance. For the complete list of terminations see Fig. 7. Note the wide variation on the order or magnitude of both resistive and dynamic elements. Also for this case, we illustrate the importance of the nonlinear driver effects through a comparison with a reference linear simulation, obtained by removing from the driver the dynamic elements, the branch through diode D2, and shorting diode D1.

Transient results for the linear and the nonlinear setup are reported and compared to the corresponding reference SPICE solutions in Fig. 8. The evolution of the nonlinear residual (15) is reported in Fig. 9. In this case, we note a remarkably fast
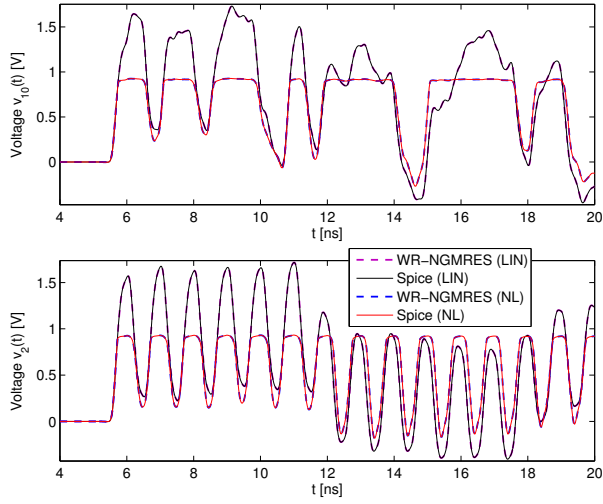
Fig. 5. Voltage signals at ports 2 and 10 of test case 1. Results with (NL) and without (LIN) the NLA clipping circuit are reported from proposed solver (dashed lines) and SPICE reference (solid lines).
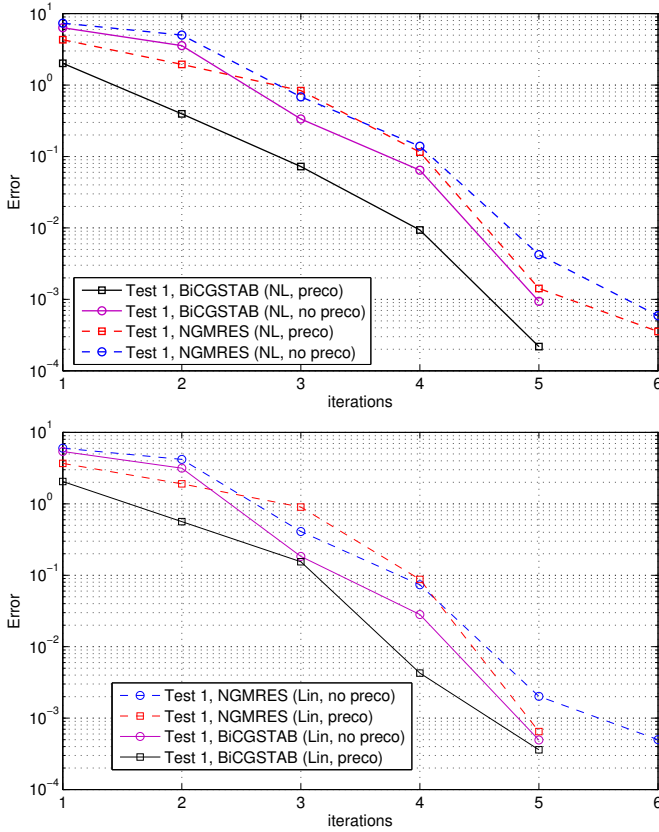

Fig. 7. Simulation deck for test case 2.

Fig. 6. Evolution of the residual error through iterations for various different simulations performed on test case 1.

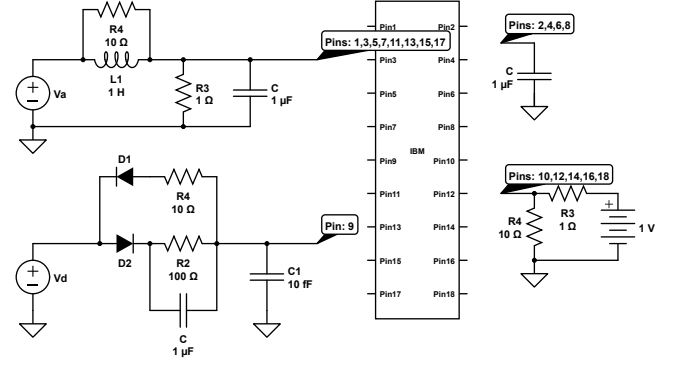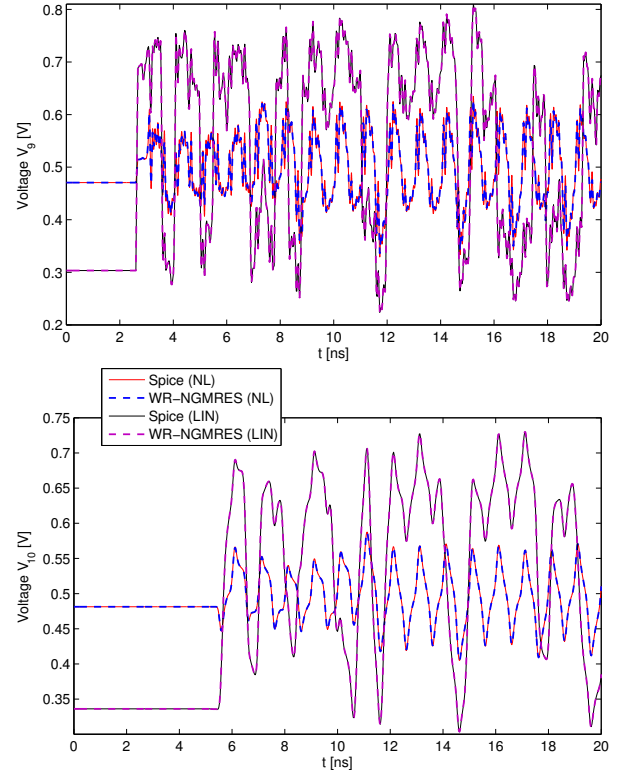Fig. 8. Transmitted (v_9) and received (v_{10}) signals for test case 2. Results with the nonlinear NLB driver (NL) and a corresponding linearized driver (LIN) are reported for proposed solver (dashed lines) and SPICE reference (solid lines).

convergence of the preconditioned BiCGSTAB implementation in only 4 iterations, although all other implementations converged in up to 6 iterations.

E. Test case 3

The termination setup for this test case is illustrated in Fig. 10. A circuit implementaiton of the NLA block is connected to the driver port 9. All other terminations are characterized by varying impedance levels, in order to investigate if the convergence properties suffer from load sensitivity. In addition, we test two different channel models with the above termination scheme, namely the node-to-I/O channel
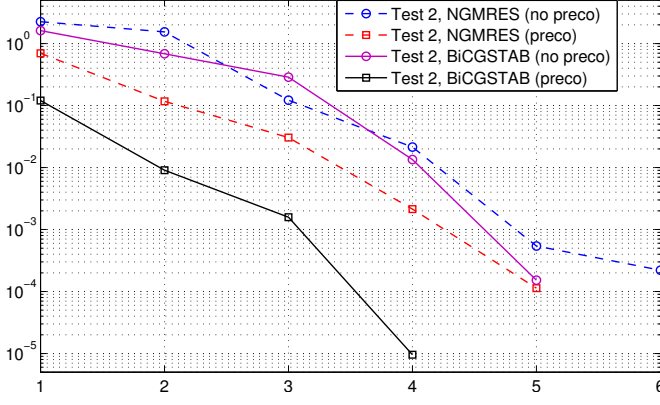
Fig. 9. Evolution of the residual error through iterations for various different simulations performed on test case 2.
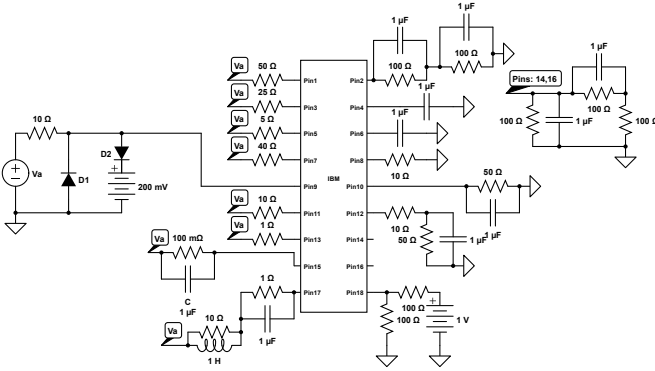

Fig. 10. Simulation deck for test case 3.

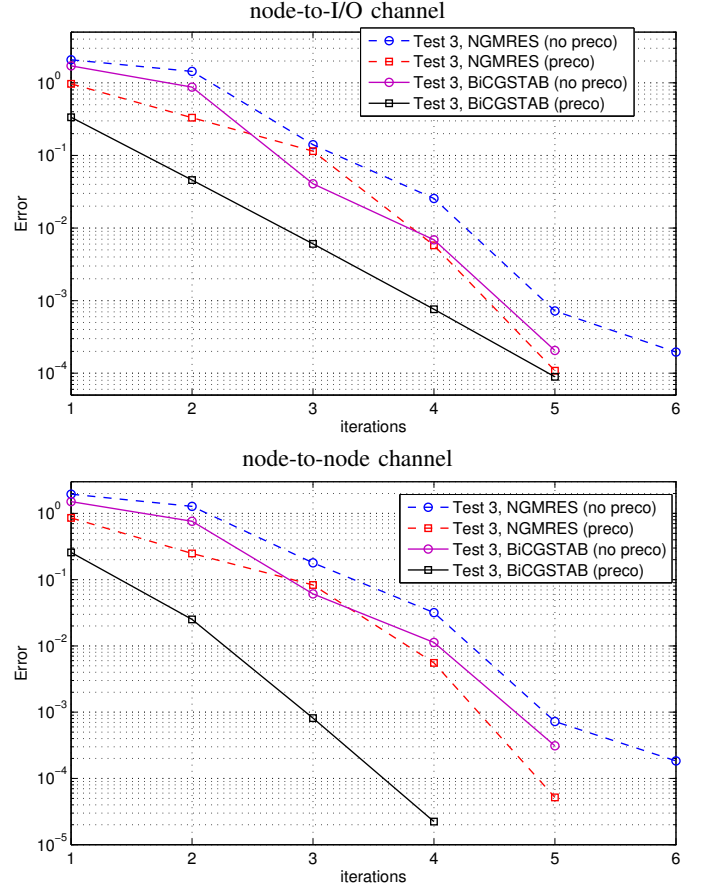node-to-I/O channel

node-to-node channel

Fig. 11. Evolution of the residual error through iterations for various different simulations performed on test case 3. Top and bottom panels refer to a node-to-I/O channel and a node-to-node channel across a backplane, respectively.

already analyzed in the first three test cases, and a node-to-node channel across a backplane (see [29] for details). Figure 11 reports the evolution of the residual errors for both channels, showing that the number of required iterations is only marginally affected. All proposed solver implementations reach convergence in up to 6 iterations.

The convergence is further visualized in Fig. 12, where the solution estimates obtained after the first and the sixth iterations are compared to reference SPICE results. We remark that the first iteration (initialization) is determined by considering all channels decoupled (i.e., the scattering matrix of the channel is approximated by its 2×2 block-diagonal part), and by successive evaluations of decoupled channel and termination circuits. It is therefore expected that the small features of the true response cannot be captured at this stage. The proposed scheme is however able to reduce the residual error after few iterations below the desired convergence threshold.

F. Test case 4

In this test, the node-to-I/O channel is terminated as detailed in Table I, with the two receiver circuits depicted in Fig. 13. This is the case for which convergence was fastest, as demonstrated in Fig. 14. We justify this fact based on the improved matching that is provided by the adopted termination circuits with respect to the other test cases. This example thus indicates that in a practical channel simulation setup,

TABLE I
TERMINATIONS FOR TEST CASE 4

Driver side		Receiver side	
Port	Circuit	Port	Circuit
1	1Ω resistor	2	NLA+RCRC
3	1Ω resistor	4	RCRC
5	1Ω resistor	6	RCRC
7	1Ω resistor	8	RCRC
9	NLA+10Ω resistor	10	NLA+RRV
11	1Ω resistor	12	RCRC
13	1Ω resistor	14	RCRC
15	1Ω resistor	16	RCRC
17	1Ω resistor	18	RCRC

where both drivers and receivers are optimally matched to the channel, it is expected that proposed scheme will be very effective due to fast convergence. A SPICE validation is reported for completeness in Fig. 15, where the strong clipping effects of the nonlinear terminations are clearly visible.

G. Test case 5

The last test case of this section demonstrates the possibilty to include differential terminations. This case is of particular interest due to the widespread adoption of differential signaling for improved speed and immunity in data transmission. We considered both linear and nonlinear terminations, depicted in
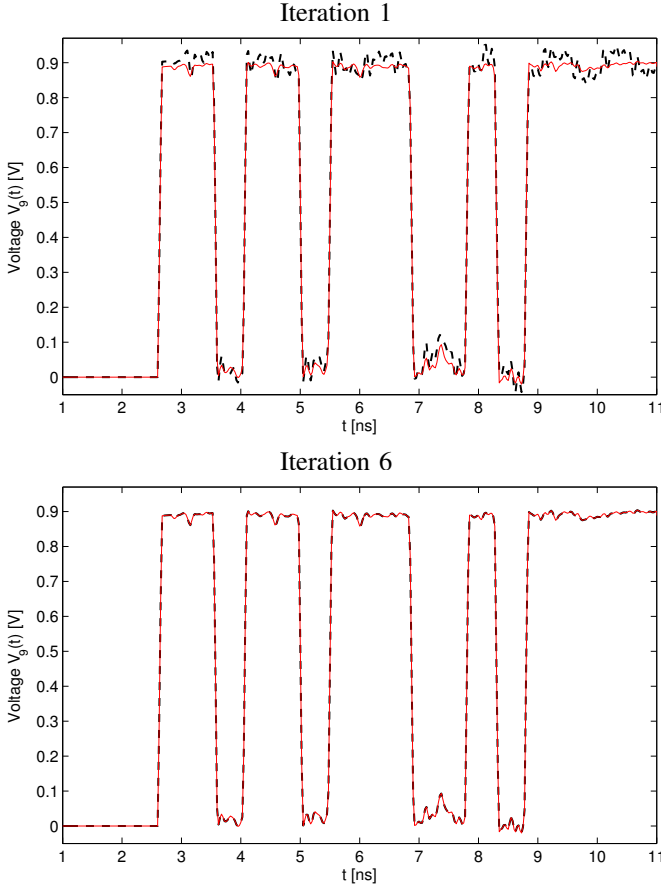
Fig. 12. Transmitted (v_9) signal for test case 3 at iteration 1 (top) and 6 (bottom). The black dashed line is the voltage signal from WR-NGMRES, while the red continuous line is the SPICE reference.
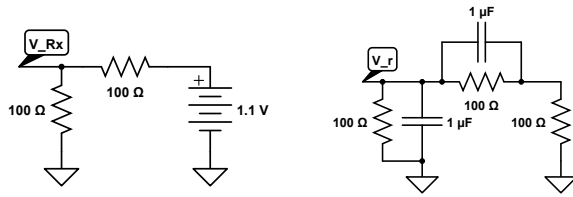

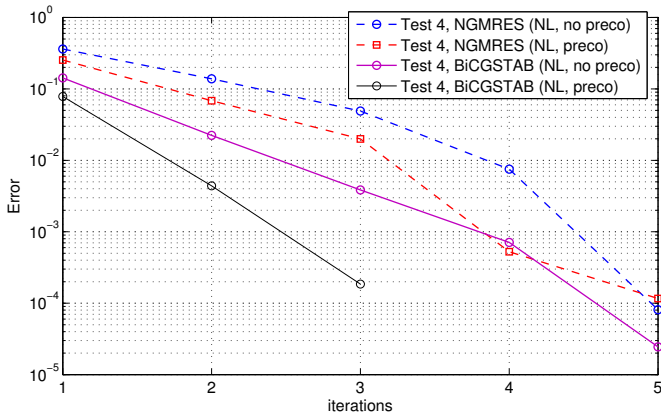Fig. 13. Receiver test circuit codenamed RRV (left) and RCRC (right)

Fig. 14. Evolution of the residual error through iterations for various different simulations performed on test case 4.
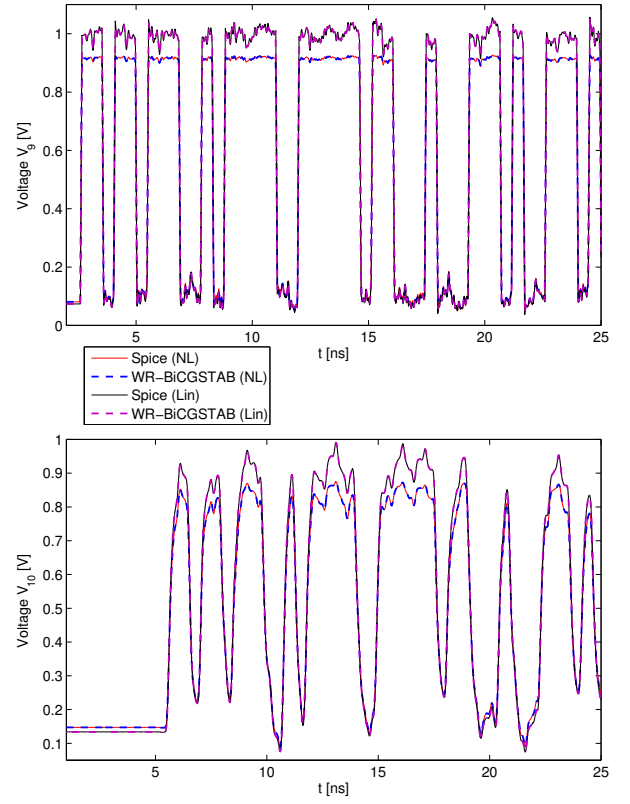

Fig. 15. Transmitted (v_9) and received (v_{10}) signals for test case 4. Results with (NL) and without (LIN) the NLA clipping circuit are reported from proposed solver (dashed lines) and SPICE reference (solid lines).
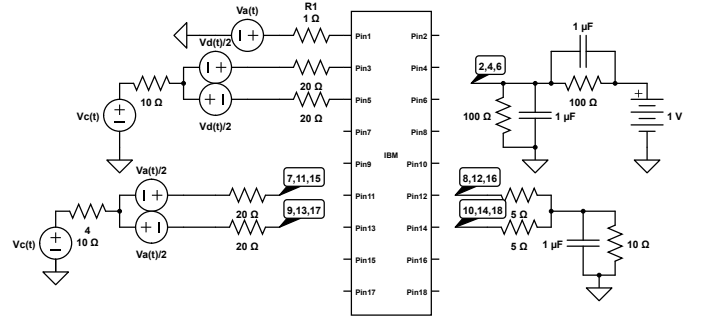

Fig. 16. Simulation deck for test case 5 (linear setup).

Fig. 16 and Fig. 17, respectively. The nonlinear case induces a strong clipping on the resulting waveforms, as demonstrated in Fig. 19, where the differential voltages at both transmitted (ports 7-9) and received (ports 8-10) pairs are compared to a SPICE reference. The corresponding residual error evolutions are reported in Fig. 18 for linear (top) and nonlinear (bottom) setup, respectively.

H. Efficiency and runtime

In this section, we report the runtime required by test cases 1–5 to run a 500-bit transient analysis, using the complete NGMRES-based implementation (without preconditioning) of proposed solver, and including automated WR initialization. The cumulative runtime is reported in Table II. The second column reports the total runtime required by
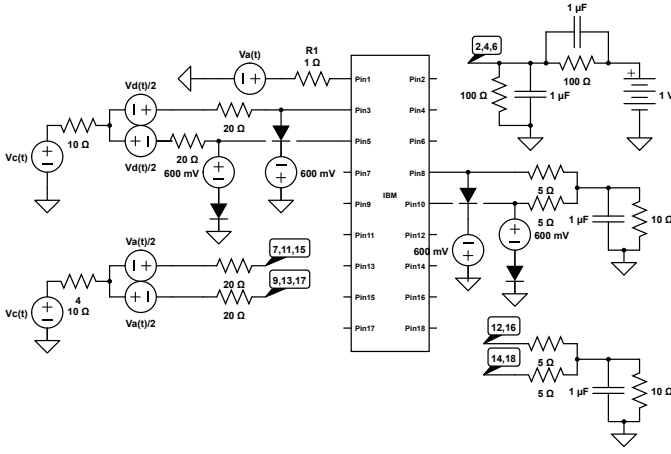
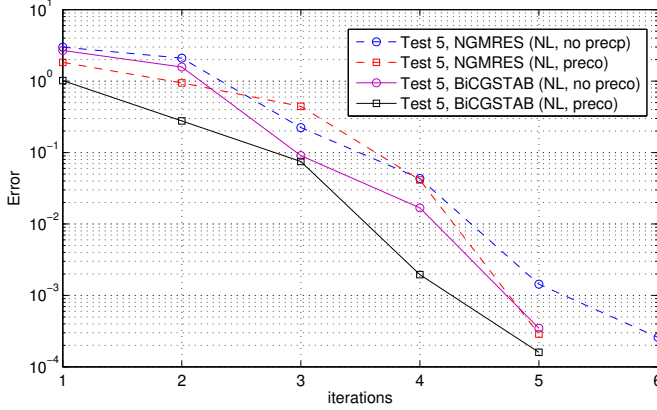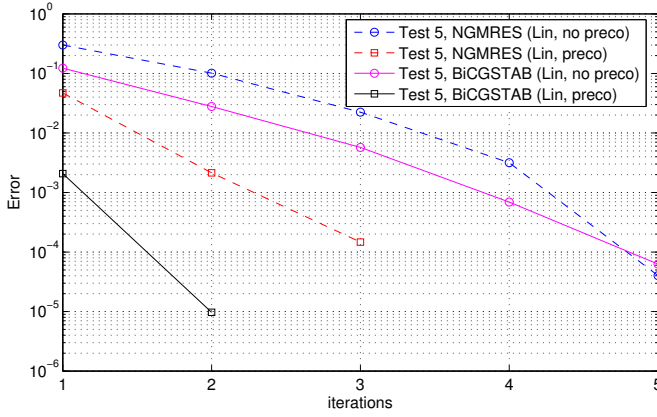Fig. 17. Simulation deck for test case 5 (nonlinear setup).

Fig. 18. Evolution of the residual error through iterations for various different simulations performed on test case 5.

Fig. 19. Transmitted $(v_7 - v_9)$ and received $(v_8 - v_{10})$ signals for test case 5. Results for both linear and nonlinear termination schemes are reported from proposed solver (dashed lines) and SPICE reference (solid lines).

SPICE when applied to a single global netlist including both an equivalent circuit synthesis of the channel macromodel and the corresponding termination circuits, implemented through standard elements. The last two columns report, respectively, the cumulative runtime (all iterations) required by SPICE to solve for the decoupled termination circuits closed on relaxation sources (including overhead due to file-based exchange of relaxation sources), and by our dedicated Matlab code for solving the decoupled channel macromodel (also terminated by relaxation sources) via recursive convolutions. Note that, in our current implementation, all runs are sequential and do
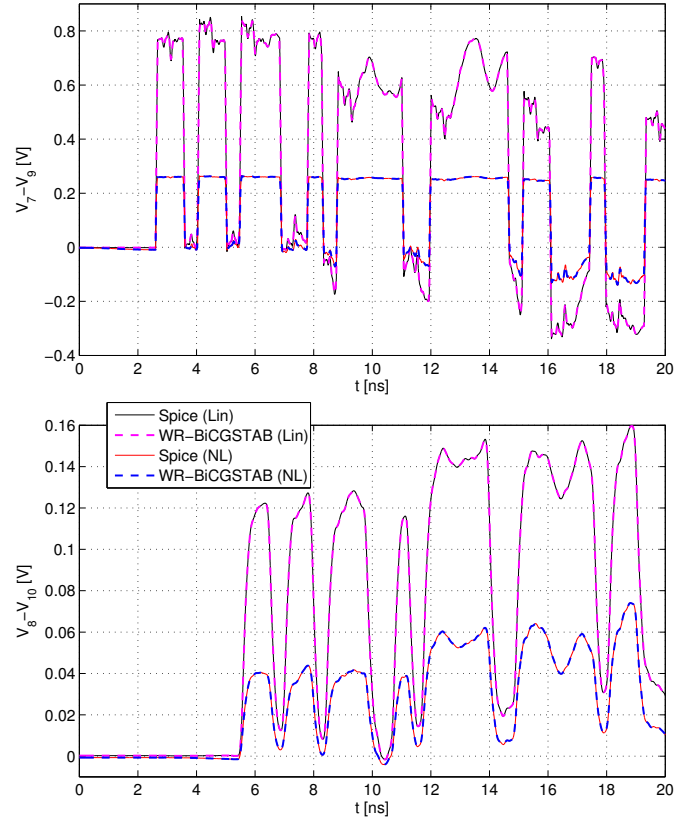
not take advantage of any multicore parallel processing engine.

Table II does not report the additional overhead due to channel macromodeling. Recalling the main results of [29], all analyzed channels required from one to 5 minutes for DRM identification, and from 6 up to 10 minutes for passivity enforcement. Note that this cost is spent only once for macro-model generation, since once available, the macromodel can be reused for any arbitrary subsequent transient simulation.

Table II clearly demonstrates that

1) The full SPICE simulations are much slower than the WR-based simulations;
2) For each WR simulation, the time spent by SPICE to solve for (very simple) decoupled termination circuits is much larger than the time required by the WR engine to solve for the channel and to perform the outer iteration loop.

These results suggest that embedding the termination circuits in the WR solver, e.g., in form of behavioral models, the corresponding decoupled transient analyses could be computed much faster and without resorting to a general-purpose solver like SPICE. This is in fact the approach that we pursue in next Section.

VI. ALL-MACROMODEL IMPLEMENTATION

In this section, we demonstrate the feasibility of an all-macromodel implementation of the proposed WR scheme,

TABLE II
CUMULATIVE RUNTIME FOR TESTCASES 1–5. ALL SIMULATIONS REFER
TO A 500-BIT SEQUENCE. FOR THE WR SCHEME, THE TERMINATION
PARTS ARE SOLVED BY SPICE, WHEREAS THE CHANNEL PARTS ARE
SOLVED BY DEDICATED (MATLAB) CODE.

Test	SPICE	WR	
		terminations	channel
1 (LIN)	2h 19min	1h 11min	22min
1 (NL)	2h 21min	1h 15min	22min
2 (LIN)	2h 6min	37min	9min
2 (NL)	2h 14min	35min	9min
3 (NL)	2h 26min	47min	13min
4 (LIN)	2h 28min	5min	3min
4 (NL)	2h 19min	11min	3min
5 (LIN)	2h 8min	8min	6min
5 (NL)	2h 18min	1h 3min	22min

Fig. 20.   Differential driver: definition of output ports and signals

by adopting behavioral macromodels also for the channel terminations. In the present case of high-speed channel terminations, the Mπlog technique [6] has been demonstrated to be accurate and reliable for behavioral macromodeling of drivers and receivers, both single-ended [6] and differential [7]. This technique is essentially based on parametric models and black-box identification methods. Model parameters are estimated from external port current and voltage waveforms that are available from reference (short) transistor-level simulations. These parametric mathematical descriptions are very general and flexible, and in principle they can be cast to describe the relationship between port variables in any given representation. Therefore, for the development of macromodels that are compliant with the proposed WR solver as in (10), we modify the Mπlog technique to express the macromodel equations directly in the scattering domain, rather than through standard voltages and currents. A short outline is given below for the single-ended and the differential driver cases. Receiver modeling is not critical, so we concentrate only on the driver cases.

A. Single-ended drivers

Let us consider the case of a single-ended (one-port) driver, with port variables defined as in Fig. 2. We recall that the transient scattering waves are defined as (we drop the port index ℓ for simplicity) $a = (v + R_0 i)/(2\sqrt{R_0})$ and $b = (v - R_0 i)/(2\sqrt{R_0})$ with R_0 port reference resistance *e.g.*, $R_0 = 50\Omega$. Consistently with the notation of this paper, the scattering wave $b(t)$ is considered as incident into the driver output (hence outgoing from the channel), whereas $a(t)$ is the scattered (launched) wave form the driver into the channel. With these assumptions, the Mπlog model that describes the driver behavior through these scattering variables assumes the form

$$a(t) = w_H(t)a_H(b(t), d/dt) + w_L(t)a_L(b(t), d/dt). \quad (21)$$

Representation (21) belongs to the so-called two-piece model class and represents a modified version of the classic relation expressed in terms of port voltage and current [6]. Specifically,

- a_H and a_L are parametric submodels describing the nonlinear dynamic behavior of the output port in the fixed High (H) and Low (L) logic states, respectively. Each of
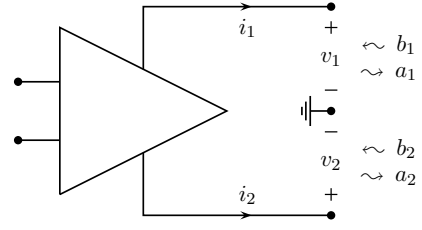
these two submodels is identified as a further superposition of a static nonlinear submodel and a dynamic linear submodel.

- w_H and w_L are time-dependent weights accounting for logic state transitions.

The details of the identification process are discussed in [6].

Since our intention is to plug the model into proposed WR solver, which operates in the discrete-time, the macromodel equation (21) is formulated and its parameters identified directly in discrete time with a time step $t_k = k\,\delta t$, such that the output reflected wave at a given time step t_k can be computed directly from (21) knowing its past samples, as well as present and past samples of the incident wave. This description is compliant with (10) and with our proposed WR solver.

B. Differential drivers

With reference to the basic differential driver structure depicted in Fig. 20, port scattering waves are defined as $a_n = (v_n + R_0 i_n)/(2\sqrt{R_0})$ and $b_n = (v_n - R_0 i_n)/(2\sqrt{R_0})$ for $n = 1, 2$, again with the convention that b_n are the incident waves into the driver output ports, and a_n are the corresponding reflected waves. For differential drivers, we define the common and differential incident waves as $b_c(t) = (b_1(t) + b_2(t))/2$ and $b_d(t) = b_1(t) - b_2(t)$, respectively. As discussed in [63], common and differential incident waves form the input set of choice for the scattering-based Mπlog model for differential drivers, which consists of the two following equations

$$\begin{cases} a_1(t) &= w_{1H}(t)a_{1H}(b_c(t), b_d(t), d/dt) \\ &+ w_{1L}(t)a_{1L}(b_c(t), b_d(t), d/dt) \\ \\ a_2(t) &= w_{2H}(t)a_{2H}(b_c(t), b_d(t), d/dt) \\ &+ w_{2L}(t)a_{2L}(b_c(t), b_d(t), d/dt) \end{cases} \quad (22)$$

where parametric submodels $a_{n\nu}$ for $n = 1, 2$ and $\nu = H, L$ and time-dependent weights $w_{n\nu}$ play the same role as their counterparts in model (21) by retaining the same time discretization scheme. We remark that we choose $b_{c,d}$ as model inputs because under ideal conditions, i.e., when connecting the driver model to a pair of perfectly matched and balanced transmission lines with line impedance R_0, we obtain $b_d = 0$ independent on the driver logic state. Arguing that under realistic operations the equivalent channel impedance will not be far from R_0 (a condition that is targeted in the design), the range of values that will be spanned by b_d will be a small neighborhood of 0, thus making the model outputs $a_{1,2}$ very insensitive to its differential input. Also in this

case, the estimation process of model (22) is standard and follows the same guidelines that apply for the identification of voltage-current models [6]. See [63] for more details on the peculiarities of scattering-based models.

C. Complexity

A few remarks are in order about overall complexity of proposed framework. We note that the cost for application of the channel operator $\mathcal{H}$ scales as $O(\bar{N}_d \bar{N}_p P^2)$, where $\bar{N}_d$ and $\bar{N}_p$ are the average number of delays and poles for all channel responses. Conversely, application of the termination operator $\mathcal{T}$ based on Mπlog macromodels, assuming all ports terminated into single-ended drivers for simplicity, scales only as $O(\kappa P)$, where κ is a constant determined by the overall dynamic order of the H, L submodels. Therefore, we see that a single evaluation of the nonlinear residual $\mathcal{N}(\mathbf{b})$ in (15), which essentially requires one channel and one termination operator application, has a quite favorable scaling with system complexity. The overall WR scheme requires a number of evaluations that scales linearly with the size q of the Krylov subspace used by the approximate linear solvers, with an embedded orthogonalization loop that requires $O(q^2 N)$ operations. The latter is the most computationally demanding step of the algorithm. For this reason, in our implementation we limit the maximum size $q_{\max} = 40$, although this limit was never reached in any simulation.

VII. APPLICATION EXAMPLES

We illustrate the advantages of the all-macromodel WR formulation through three additional test cases based on three different driver devices.

A. Test case 6

The first device we consider is the single-ended driver Fairchild NC7SV126 ($V_{\mathrm{dd}} = 3.3$ V, 750 ps switching time), whose reference description is the HSPICE-encrypted transistor-level model freely available from [66]. In order to get a preliminary feeling on the accuracy of the Mπlog macromodel, a basic validation setup has been devised, consisting of a short 400Mbps bit-sequence launched by the driver into a (50 Ω, 2 ns) ideal transmission line terminated into a 60 Ω resistive load. Figure 21 shows the comparison between the port current signals obtained by running the encrypted transistor-level model in HSPICE and the estimated scattering-based model in the Matlab environment. This excellent match validates the Mπlog model and enables its systematic use for channel analysis.

As a more challenging assessment, the scattering-based Mπlog model has been plugged into the WR solver for carrying out a realistic channel simulation. The setup at hand involves nine instances of the driver connected to the (near end) odd numbered ports of the channel, and nine resistive 50 Ω terminations connected to the (far end) even numbered ports. All the nine drivers transmit the same 400 Mbps pseudo-random bit sequence. Figure 22 depicts the comparison between the voltage at port 9 obtained by running the WR
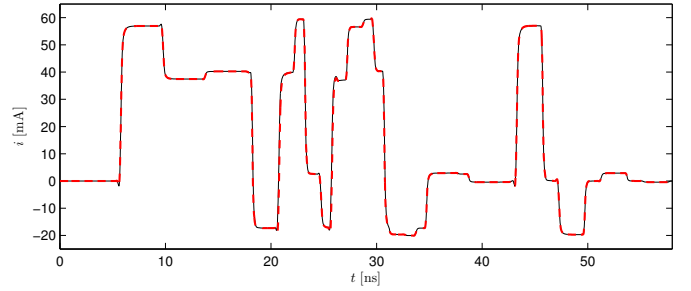

Fig. 21. Validation of the single-ended driver model of test case 6 loaded by a mismatched transmission line load. Reference (solid black line), model (red dashed line).
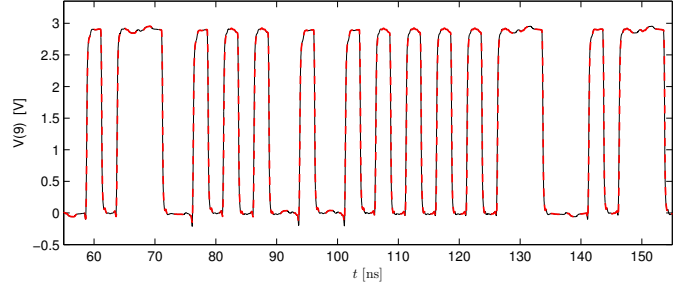

Fig. 22. Channel simulation with single-ended driver terminations (test case 6). Reference (solid black line), macromodel-based WR solver (red dashed line).

solver in Matlab, and the corresponding HSPICE simulation based on the transistor-level driver models connected to a circuit realization of the channel macromodel. The WR solver required only 4 minutes, whereas the reference HSPICE simulation required 50 minutes, with a speedup of about 12.5×. This speedup was achieved with no loss of accuracy, as Fig. 22 confirms.

B. Test case 7

The second considered device is the differential driver Fairchild FIN1019 ($V_{\mathrm{dd}} = 3.3$ V, 1 ns switching time) whose reference description is again a HSPICE encrypted transistor-level model freely available from [66]. We setup a transient simulation with four driver instances connected to four pairs of odd-numbered channel ports, with resistive 50Ω terminations at all remaining ports. The drivers transmit a pseudo-random bit sequence at 200 Mbps rate. Figure 23 reports the comparison between the reference simulation (transistor-level driver models and equivalent circuit realization of channel macromodel solver by HSPICE) and the results obtained by our proposed WR solver. The figure reports common and differential voltages at the receiver ports 10-12, demonstrating a good match. The WR simulation only took 3 minutes, with a 12× speedup with respect to the reference HSPICE simulation, which required 36 minutes.

C. Test case 8

The two above testcases were based on freely available but relatively simple and slow driver devices. Therefore, we turn to one last test case based on a real industrial production
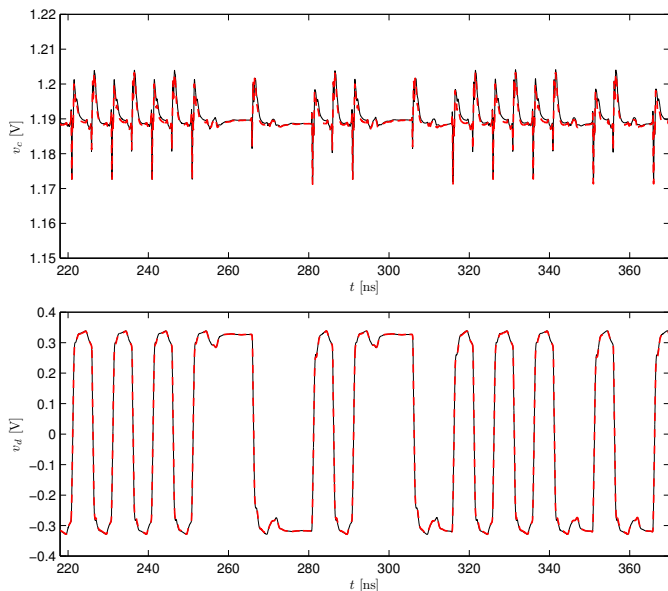
Fig. 23. Channel simulation with differential driver terminations (test case 7). Reference (solid black line), model (red dashed line).
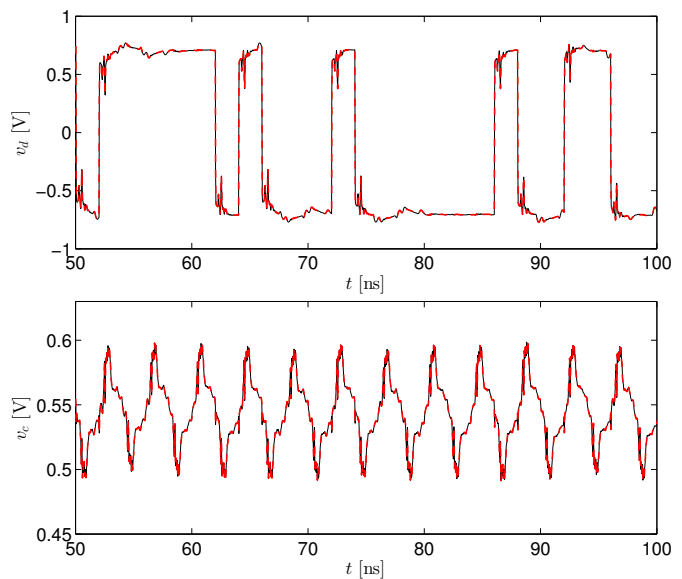

Fig. 24. Full channel/driver combined simulation (see text). Model response (dashed red line) is compared to a full transistor-level reference simulation (black solid line).

level driver used for differential signaling on high speed buses (courtesy of IBM). This differential driver has a nominal supply voltage $V_{\text{dd}} = 1.1$ V, with a switching time of 100 ps. The reference device is the transistor-level schematic coded for the Spectre simulator. More details on this driver and its Mπlog macromodeling process can be found in [63].

As the simulation with the transistor-level turned out to be highly time consuming due to the overwhelming circuit complexity, we consider a simulation setup with a single driver instance, launching a pseudo-random bit sequence at 1 Gbps into one pair of channel ports (ports 9-11). All other terminations consist of synchronous linear aggressors at the near end ports, and resistive loads at the far end ports. Figure 24 compares the differential and common voltage at the driver output ports, highlighting once again an excellent model accuracy. Note also the very accurate prediction of the common voltage periodic perturbation due to the synchronous activity of the aggressors. The WR solver required only 1 minute to complete the simulation, whereas the reference transistor-level simulation (in Spectre) lasted 2 h 30 min. The resulting speedup was 150×.

VIII. DISCUSSION AND LIMITATIONS

The proposed scheme overcomes the convergence issues of earlier WR formulations, and a significant speedup can be achieved with respect to a pure SPICE channel simulation when employing macromodels for all system parts. However, there are some important limitations that should be considered, itemized below.

- The computational requirements for the simulation of a complex channel with nonlinear terminations are such that only bit sequences of few hundreds, up to a few thousands at most, can be handled. Therefore, eye diagram computations for the extraction of statistical information such as BER, down to a level that is required by serial high-speed links (e.g., 10^{-12}), is definitely unfeasible with this approach. This is the price one has to pay for attaining SPICE-level accuracy in treating full nonlinearities.

- The proposed solver is based on a predefined uniform time discretization step δt, which is embedded in the discrete-time operators that represent both channel and terminations. This implies that signal details at a time scale smaller than δt cannot be resolved.

- The most efficient implementation or proposed framework is based on macromodels: both channels and nonlinear terminations present some challenges under this standpoint. On one hand, the channel DRM formulation (5) requires a precise estimation of dominant delays $\tau_d^{\ell,m}$ and poles $p_{dn}^{\ell,m}$, a task that is numerically challenging and that, despite the recent advances, is still not robust as its delayless counterpart, e.g., [15] and its derivatives. For nonlinear driver/receiver macromodels, the situation is even more challenging. In fact, although the adopted Mπlog approach is fairly general, it is constrained to a quite specific model structure. It is not guaranteed that this structure will be able to represent future devices based on new technologies. Should this happen, another class of nonlinear macromodels will have to be developed and suitably incorporated in the WR framework. The approach will remain valid if such models will be available in a nonlinear discrete-time form (10).

We emphasize that the above limitations were not critical in any of the (many) test cases that we analyzed, of which the eight test cases documented in this work form a subset.

IX. CONCLUSIONS

We have presented a Waveform Relaxation (WR) framework for transient analysis of complex channels terminated by nonlinear drivers and receivers. The main advantage of presented

approach is the formulation based on behavioral macromodels for both linear channel and nonlinear terminations. All macromodels are cast in a scattering representation, which enables straightforward WR simulation based on system partitioning and domain decomposition. Global convergence is enforced through a suitable implementation of inexact Newton methods combined with Krylov linear solvers.

The numerical results that are documented in this work show both the excellent convergence and the good speedup with respect to circuit-oriented SPICE simulations that are achieved by our WR solver. Our current implementation is based on a prototypal code in the Matlab environment. However, due to the intrinsic nature of WR schemes, it is expected that a suitably parallelized code deployed on a multicore hardware platform will achieve much faster runtime and good scalability. This will be part of our future research efforts.

REFERENCES

[1] W. T. Beyene, C. Madden, Jung-Hoon Chun, Haechang Lee, Y. Frans, B. Leibowitz, Ken Chang, Namhoon Kim, Ting Wu, G. Yip, R. Perego, "Advanced Modeling and Accurate Characterization of a 16 Gb/s Memory Interface," *Advanced Packaging, IEEE Transactions on*, vol. 32, no. 2, pp. 306–327, May 2009.

[2] G. Balamurugan, B. Casper, J. E. Jaussi, M. Mansuri, F. O'Mahony, J. Kennedy, "Modeling and Analysis of High-Speed I/O Links," *Advanced Packaging, IEEE Transactions on*, vol. 32, no. 2, pp. 237–247, May 2009.

[3] A. Sanders, "Statistical Simulation of Physical Transmission Media," *Advanced Packaging, IEEE Transactions on*, vol. 32, no. 2, pp. 260–267, May 2009.

[4] I/O Buffer Information Specification version 5.1, [Online] Available: http://eda.org/pub/ ibis/ver5.1/ver5-1.pdf

[5] I/O Buffer Information Specication (IBIS), Jan. 2004. Ver. 4.1. [Online]. Available: http://www.eigroup.org/ibis/-ibis.htm

[6] I. S. Stievano, I. A. Maio, and F. G. Canavero, "Mπlog, macromodeling via parametric identification of logic gates," *IEEE Trans. Adv. Packag.*, vol. 27, no. 1, pp. 15-23, Feb. 2004.

[7] I. S. Stievano, I. A. Maio, F. G. Canavero, C. Siviero, "Parametric Macromodels of Differential Drivers and Receivers," *IEEE Trans. Adv. Packag*, Vol. 28, No. 2, pp. 189-196, May, 2005.

[8] W. Dghais, T.R. Cunha, J.C. Pedro, "Reduced-Order Parametric Behavioral Model for Digital Buffers/Drivers With Physical Support," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 2, no. 12, pp. 2071–2079, Dec. 2012

[9] W. Dghais, H.M. Teixeira, T.R. Cunha, J.C. Pedro, "Novel Extraction of a Table-Based IQ Behavioral Model for High-Speed Digital Buffers/Drivers," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 3, no. 3, pp. 500–507, March 2013

[10] W. Dghais, T.R. Cunha, J.C. Pedro, "A Novel Two-Port Behavioral Model for I/O Buffer Overclocking Simulation," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 3, no. 10, pp. 1754–1763, Oct. 2013

[11] M.G. Telescu, I.S. Stievano, F.G. Canavero, N. Tanguy, "An Application of Volterra Series to IC Buffer Models", *14th IEEE Workshop on Signal Propagation On Interconnects, Hildesheim, Germany, May 9-12, 2010*, pp. 93–96.

[12] C. Diouf, M. Telescu, N. Tanguy, P. Cloastre, I.S. Stievano, F.G. Canavero, "Statically constrained nonlinear models with application to IC buffers", *15th IEEE Workshop on Signal Propagation On Interconnects, Napoli, Italy, May 8-11, 2011*, pp 115–118.

[13] Yi Cao, Runtao Ding, Qi-Jun Zhang, "State-Space Dynamic Neural Network Technique for High-Speed IC Applications: Modeling and Stability Analysis," *IEEE Trans. Microw. Th. Tech.*, Vol. 54, No. 6, pp. 23982409, June 2006.

[14] T. M. Comberiate, J. E. Schutt-Aine, "Modeling I/O buffers using X-parameters," *IEEE 22nd Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS), San Jose, CA, USA, 27-30 Oct. 2013*, pp. 25–28.

[15] B. Gustavsen, A. Semlyen, "Rational approximation of frequency responses by vector fitting", *IEEE Trans. Power Delivery*, Vol. 14, N. 3, pp. 1052–1061, July 1999.

[16] B. Gustavsen, A. Semlyen, "A robust approach for system identification in the frequency domain", *IEEE Trans. Power Delivery*, Vol. 19, N. 3, pp. 1167–1173, July 2004.

[17] D. Deschrijver, B. Haegeman, T. Dhaene, "Orthonormal Vector Fitting: A Robust Macromodeling Tool for Rational Approximation of Frequency Domain Responses", *IEEE Trans. Adv. Packaging*, vol. 30, pp. 216–225, May 2007.

[18] S. Grivet-Talocia, M. Bandinu, "Improving the Convergence of Vector Fitting in Presence of Noise", *IEEE Transactions on Electromagnetic Compatibility*, vol. 48, n. 1, pp. 104-120, February, 2006.

[19] W. Beyene, J. Schutt-Ainé, "Accurate Frequency-Domain Modeling and Efficient Circuit Simulation of High-Speed Packaging Interconnects", *IEEE Trans. Microwave Theory Tech.*, vol. 45, pp. 1941-1947, Oct. 1997

[20] S. Grivet-Talocia, "Delay-based macromodels for long interconnects via time-frequency decompositions," in *IEEE* 15$^{\text{th}}$ *Topical Meeting on Electrical Performance of Electronic Packaging, Scottsdale, Arizona*, Oct. 23–25, 2006, pp. 199–202.

[21] A. Charest, D. Saraswat, M. Nakhla, R. Achar, N. Soveiko, "Compact Macromodeling of High-Speed Circuits via Delayed Rational Functions," *IEEE Microwave and Wireless Components Letters* Vol. 17, No. 12, pp. 828–830, Dec. 2007.

[22] A. Charest, M. Nakhla, R. Achar, "Delay Extracted Stable Rational Approximations for Tabulated Networks With Periodic Reflections," *IEEE Microwave and Wireless Components Letters*, Vol. 19, No. 12, Dec 2009, pp. 768–770.

[23] A. Chinea, P. Triverio, S. Grivet-Talocia, "Delay-Based Macromodeling of Long Interconnects from Frequency-Domain Terminal Responses," *IEEE Transactions on Advanced Packaging*, Vol. 33, No. 1, pp. 246–256, Feb. 2010.

[24] P. Triverio, S. Grivet-Talocia, A. Chinea, "Identification of highly efficient delay-rational macromodels of long interconnects from tabulated frequency data," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 3, pp. 566–577, 2010.

[25] A. Charest, M. Nakhla, R. Achar, D. Saraswat, N. Soveiko, I. Erdin, "Time Domain Delay Extraction-Based Macromodeling Algorithm for Long-Delay Networks," *IEEE Transactions on Advanced Packaging*, Vol. 33, No. 1, pp. 219–235, Feb. 2010.

[26] A. Charest, M. Nakhla, R. Achar, "Scattering Domain Passivity Verification and Enforcement of Delayed Rational Functions," *IEEE Microwave and Wireless Components Letters*, Vol. 19, No. 10, pp. 605–607, Oct. 2009.

[27] A. Chinea, S. Grivet-Talocia, D. Deschrijver, T. Dhaene, L. Knockaert, "On the construction of guaranteed passive macromodels for high-speed channels," *Design, Automation and Test in Europe (DATE 10)*, Dresden (Germany), March 8-12, 2010, pp. 1142-1147.

[28] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Passive delay-based macromodels for signal integrity verification of multi-chip links," in *14th IEEE Workshop on Signal Propagation on Interconnects, Hildesheim, Germany*, May 9–12, 2010, pp. 113–116.

[29] A. Chinea, S. Grivet-Talocia, H. Hu, P. Triverio, D. Kaller, C. Siviero, M. Kindscher, "Signal Integrity Verification of Multichip Links using Passive Channel Macromodels," *IEEE Transactions on Advanced Packaging*, Vol. 1, No. 6, pp. 920–933, June 2011.

[30] E. Lelarasmee, "The Waveform Relaxation Method for Time Domain Analysis of Large Scale Integrated Circuits: Theory and Applications", EECS Department University of California, Berkeley Technical Report No. UCB/ERL M82/40, 1982.

[31] E. Lelarasmee, A. E. Ruehli, A. L. Sangiovanni-Vincentelli, "The Waveform Relaxation Method for Time-Domain Analysis of Large Scale Integrated Circuits", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 1, N. 3, July 1982, pp. 131–145.

[32] J. K. White and A. L. Sangiovanni-Vincentelli, *Relaxation Technique for the Simulation of VLSI Circuits*. Norwell, MA: Kluwer Academic, 1987.

[33] A. E. Ruehli, Ed., *Circuit Analysis, Simulation and Design, Part 2*, Elsevier Science, North Holland, 1987.

[34] P. Saviz, O. Wing, "Circuit Simulation by Hierarchical Waveform Relaxation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 12, N. 6, June 1993, pp. 845–860.

[35] A. Lumsdaine, M. W. Reichelt, J. M. Squyres, J. K. White, "Accelerated Waveform Methods for Parallel Transient Simulation of Semiconductor Devices", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 15, N. 7, July 1996, pp. 716–726.

[36] G. D. Gristede, A. E. Ruehli, C. A. Zukowski, "Convergence Properties of Waveform Relaxation Circuit Simulation Methods", *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, Vol.45, No.7, pp. 726–738, July 1998.

[37] Yao-Lin Jiang, "A General Approach to Waveform Relaxation Solutions of Nonlinear Differential-Algebraic Equations: The Continuous-Time and Discrete-Time Cases", *IEEE Transactions on Circuits and Systems-I: Regular Papers*, Vol.51, No.9, pp. 1770–1780, Sept. 2004.

[38] F. Y. Chang, "The generalized method of characteristics for waveform relaxation analysis of lossy coupled transmission lines," *IEEE Trans. Microwave Theory Tech.*, vol. 37, pp. 2028-2038, Dec. 1989.

[39] F. C. M. Lau, "Improvements in the Waveform Relaxation Method Applied to Transmission Lines", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.13, No.11, pp.1409–1412, Nov. 1994.

[40] N. Nakhla, A. Ruehli, M. Nakhla, R. Achar and C. Chen, "Waveform relaxation techniques for simulation of complex interconnects with frequency-dependent parameters", *IEEE 14-th Topical Meeting on Electrical Performance of Electronic Packaging (EPEP)*, Oct. 2005, pp. 47–50.

[41] R. Wang, O. Wing, "Transient Analysis of Dispersive VLSI Interconnects Terminated in Nonlinear Loads", *IEEE Transactions on Computer-Aided Design*, Vol.11, N.10, pp. 1258–1277, Oct. 1992.

[42] N. Nakhla, A. E. Ruehli, M. S. Nakhla, R. Achar, C. Chen, "Waveform Relaxation Techniques for Simulation of Coupled Interconnects With Frequency-Dependent Parameters", *IEEE Transactions on Advanced Packaging*, Vol.30, N.2, pp. 257–269, May 2007.

[43] M. Gander, M. Al-Khaleel, A. E. Ruehli; "Waveform Relaxation Technique for Longitudinal Partitioning of Transmission Lines,"in *IEEE 15th Topical Meeting on Electrical Performance of Electronic Packaging, Scottsdale, Arizona*, Oct. 23–25, 2006, pp. 207–210.

[44] M. J. Gander, M. Al-Khaleel, A. E. Ruehli, "Optimized Waveform Relaxation Methods for Longitudinal Partitioning of Transmission Lines," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 56, no. 8, pp. 1732–1743, Aug. 2009.

[45] N. M. Nakhla, A. E. Ruehli, M. S. Nakhla, R. Achar, "Simulation of coupled interconnects using waveform relaxation and transverse partitioning," *IEEE Transactions on Advanced Packaging*, vol. 29, no. 1, pp. 78–87, Feb. 2006

[46] I. M. Elfadel, "Convergence of Transverse Waveform Relaxation for the Electrical Analysis of Very Wide Transmission Line Buses", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.28, No.8, pp.1150–1161, Aug. 2009.

[47] D. Paul, N. M. Nakhla, R. Achar, M. S. Nakhla, "Parallel Simulation of Massively Coupled Interconnect Networks," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 1, pp. 115–127, Feb. 2010.

[48] W.T.Beyene, "Applications of Multilinear and Waveform Relaxation Methods for Efficient Simulation of Interconnect-Dominated Nonlinear Networks," *IEEE Trans. on Advanced Packaging*, vol. 31, no. 3, pp. 637–648, Aug. 2008.

[49] V.Loggia, S.Grivet-Talocia, H.Hu, "Transient simulation of complex high-speed channels via Waveform Relaxation", *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, pp. 1823–1838, November 2011.

[50] S. Grivet-Talocia, and V. Loggia, "Fast Channel Simulation via Waveform Over-Relaxation," in *15th IEEE Workshop on Signal Propagation on Interconnects, Napoli, Italy*, May 8–11, 2011, pp. 103–106.

[51] H. Hu, A. Chinea, S. Grivet-Talocia, M. Miscuglio, "Fast Iterative Simulation of High-Speed Channels via Frequency-Dependent Over-Relaxation," *Proc. of 20th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS 2011), San Jose, CA (USA)*, October 23-26, 2011, pp. 115–118.

[52] H. Hu, S. Grivet-Talocia, "A Preconditioned Waveform Relaxation Solver for Signal Integrity Analysis of High-Speed Channels," *Int. Symposium on Electromagnetic Compatibility (EMC Europe 2012)*, Rome, 17–21 Sept. 2012, pp. 1–6.

[53] S. B. Olivadese and S. Grivet-Talocia, "Transient analysis of high-speed channels via Newton-GMRES waveform relaxation," in *IEEE 21st Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS 2012), Tempe, AZ, USA*, October 21–24, 2012, pp. 240–243.

[54] C.T. Kelley, "Iterative Methods for Linear and Nonlinear Equations," *SIAM J. Sci. Stat. Comput.*, Vol 16 in Frontiers in Applied Mathematics.

[55] R. Dembo, S. Eisenstat, and T. Steihaug, "Inexact Newton methods," *SIAM J. Numer. Anal.*, Vol. 19, No. 2, pp. 400–408, Apr. 1982.

[56] Y.Saad, M.H.Schultz, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM J. Sci. Stat. Comput.*, Vol. 7, No. 3, July 1986

[57] H. A. Van der Vorst, "Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems," *SIAM J. Sci. Comput.*, vol. 13 (2): 631–644, 1992.

[58] J.M. Ortega, W.C.Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* Accademic Press, New York, 1970.

[59] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations,* Vol. 16 in Classical in Applaied Mathematics, SIAM, Philadelphia, 1996.

[60] L. Armijo, "Minimization of functions having Lipschitz-continuous first partial derivatives," *Pacific J. Math.*, Vol. 16, No. 1, pp 1–3, Nov. 1966.

[61] N. M. Nachtigal, S. C. Reddy, L. N. Trefethen, "How fast are non-symmetric matrix iterations?," *SIAM J. Matrix. Anal. Appl.*, Vol. 13, No. 3, pp. 778–795, July 1992.

[62] Xiao-Chuan Cai, David E. Keys, "Nonlinearly preconditioned inexact Newton algorithms," *SIAM J. Sci. Comput.*, vol. 24, No. 1, pp. 183-200, 2002.

[63] C. Siviero, S. Grivet-Talocia, S.B. Olivadese, D. Kaller, "Scattering-based nonlinear macromodels of high-speed differential drivers," *18th IEEE Workshop on Signal and Power Integrity*, Ghent, May 11-14, 2014, pp. 1–4.

[64] E. O. Brigham, *The Fast Fourier Transform*, Prentice-Hall Inc, Englewood Cliffs, New Jersey, 1974.

[65] LTSPICE IV, *Linear Technology*, available online: www.linear.com

[66] Fairchild Semiconductor, available online: www.fairchildsemi.com

Salvatore Bernardo Olivadese received the Laurea degree (B.Sc) in Information Technology in 2007, the Laurea Specialistica degree (M.Sc) in Electronic Engineering in 2009 and the Ph.D. in Electronic Engineering in 2014, all from Politecnico di Torino, Italy. For his bachelor thesis He developed a Web 2.0 service for scientific parallel applications, and for his master thesis He spent 6 months as visiting researcher at the Interconnect and Packaging Analysis Group of IBM T.J. Watson Research Center, Yorktown, NY, developing an Adaptive Frequency Sampling method. He was recipient of the IBM PhD Fellowship Award for the academic year 2011/12. His research interests concern packaging, circuit theory, Signal and Power Integrity, and parallel algorithms. Part of this research was conducted in collaboration with IBM, Intel, and IdemWorks.

Stefano Grivet-Talocia (M'98–SM'07) received the Laurea and the Ph.D. degrees in electronic engineering from Politecnico di Torino, Italy. From 1994 to 1996, he was with the NASA/Goddard Space Flight Center, Greenbelt, MD, USA. Currently, he is an Associate Professor of Circuit Theory with Politecnico di Torino. His research interests are in passive macromodeling of lumped and distributed interconnect structures, model order reduction, modeling and simulation of fields, circuits, and their interaction, wavelets, time-frequency transforms, and their applications. He is author of more than 120 journal and conference papers. He is co-recipient of the 2007 Best Paper Award of the IEEE Trans. Advanced Packaging. He received the IBM Shared University Research (SUR) Award in 2007, 2008 and 2009. Dr. Grivet-Talocia served as Associate Editor for the IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY from 1999 to 2001. He is co-founder and President of IdemWorks.

Claudio Siviero received his Diploma in electrical engineering in 2003 and his Ph.D. degree in 2007 from the Politecnico di Torino, Italy, where he is now a research assistant with the Electromagnetic Compatibility group. From 2008 to 2013 he was with the IBM Development in Boeblingen, Germany, working as electronic packaging engineer. His research interests concern macromodeling of electrical interconnects for Electromagnetic Compatibility and Signal Integrity problems. In particular, he works on algorithms development for optimization and order reduction of large-scale dynamical systems and on behavioral modeling of nonlinear circuit elements, with specific application to the characterization of digital integrated circuits.

Dierk Kaller is an Advisory Engineer working in the Systems and Technology Group at IBM. He received his Diploma in electrical engineering in 1995 from the University of Hannover, Germany, joining IBM that same year. Since then, he has worked on various System/390 and pSeries system designs in packaging development. He is currently the technical lead for the signal integrity of the first- and second-level packaging of the zSeries high-end servers.