EI-MOR: A Hybrid Exponential Integrator and Model Order Reduction Approach for Transient Power/Ground Network Analysis

Cong Wang, Dongen Yang, Quan Chen
{12132472,12132483}@mail.sustech.edu.cn,chenq3@sustech.edu.cn
School of Microelectronics, Southern University of Science and Technology
Shenzhen, Guangdong, China

ABSTRACT

Exponential integrator (EI) method has been proved to be an effective technique to accelerate large-scale transient power/ground network analysis. However, EI requires the inputs to be piece-wise linear (PWL) in one step, which greatly limits the step size when the inputs are poorly aligned. To address this issue, in this work we first elucidate with mathematical proof that EI, when used together with the rational Krylov subspace, is equivalent to performing a moment-matching model order reduction (MOR) with single input in each time step, then advancing the reduced system using EI in the same step. Based on this equivalence, we next devise a hybrid method, EI-MOR, to combine the usage of EI and MOR in the same transient simulation. A majority group of well-aligned inputs are still treated by EI as usual, while a few misaligned inputs are selected to be handled by a MOR process producing a reduced model that works for arbitrary inputs. Therefore the step size limitation imposed by the misaligned inputs can be largely alleviated. Numerical experiments are conducted to demonstrate the efficacy of the proposed method.

KEYWORDS

exponential integrator, model order reduction, power/ground network, transient simulation

1 INTRODUCTION

Power/ground (P/G) network simulation is an important task for signal/power integrity in modern IC design. The main challenges facing P/G network analysis come from the huge problem size, e.g., on the order of tens or even hundreds of millions, and a large number of ports easily going beyond thousands. The former imposes tremendous challenges on SPICE-like simulators based on traditional backward differentiation formula (BDF) time integration methods such as backward Euler and trapezoidal. The latter hinders the use of projection-based model order reduction (MOR) techniques such as PRIMA [11], since they are known to be less efficient for many-input systems.

The exponential integrator (EI) method is another attempt to accelerate transient P/G network analysis [7, 16–18]. It is based on the analytical solution of an ordinary differential equation (ODE) in terms of a matrix exponential, instead of the polynomial approximation of the time derivatives as done in traditional backward differential formula (BDF) approaches. The key computation–products of a matrix exponential with a vector–is evaluated efficiently by Krylov subspace approximation. Advantages of EI over BDF, including better scalability, higher accuracy order, massive parallelizability and convenient time-step adaptivity have been reported in various studies [3, 4, 6, 7, 15, 16, 18, 19].

However, EI is also facing the issue of input breakpoints misalignment in transient P/G network simulation. One major benefit brought by EI is the ability of using large step sizes, but with the requirement that *all* inputs must be piece-wise linear (PWL) within one step. If the inputs waveforms are poorly aligned, they will generate a large number of breakpoints, forcing EI to take much smaller time steps, which may largely diminish the performance advantages of EI. The problem is more severe for P/G networks that typically have many input ports (from device currents).

The input misalignment is less a concern for traditional BDF methods with lower accuracy orders, in that the maximum step size is typically limited by the local truncation error (LTE) instead of the breakpoints. It is the high accuracy order (or the moment-matching property to be detailed in later sections) of EI that makes breakpoints a major limiter of step size. There is few works addressing directly the misaligned input problem in the EI context. [18] proposed a technique to divide the breakpoints into global and local transition spots that are handled in different manners. However, the total number of breakpoints is not reduced, and speed-up can be achieved only when the solutions at the global transition spots can be computed in parallel across multiple compute nodes.

Seeing this challenge, we aims to develop a new method to address the input misalignment problem in this work. Our main contribution is two-fold:

1. We prove, for the first time, that EI can be viewed as a series of moment-matching MOR. It applies a simplified MOR in each time step based on rational Krylov subspace with a single input, then advances the reduced system using again the EI approach in the same step. Under certain conditions, m moments of the original solution will be matched in EI if $(m+1)$-dimensional Krylov subspace is used. A discussion on the equivalence and differences between EI and MOR is also provided.
2. Motivated by the equivalence, we devise a hybrid method EI-MOR to tackle the input misalignment problem in EI for transient P/G network simulation. The method divides the inputs into a majority group of well-aligned inputs and a minority group of misaligned inputs. The well-aligned group is handled by EI as usual. The misaligned one is treated by MOR to produce an input-independent reduced order model (ROM) whose simulation cost is not sensitive to the number of time points. The EI and MOR solutions are then superimposed to generate the final solution at the

required time points. This way, the number of time points involved in the EI simulation can be effectively reduced.

The paper is organized as follows: Section 2 briefly reviews the background of MOR and EI. Section 3 proves the equivalence theorems for EI and MOR. Section 4 details the hybrid EI-MOR method. Numerical results are presented in Section 5 and a conclusion drawn in Section 6.

2 BACKGROUND

2.1 Krylov subspace based model order reduction

A transient linear circuit simulation requires solving the following time-domain differential-algebraic equation (DAE)

$$\mathbf{C}\dot{x}\left(t\right)+\mathbf{G}x\left(t\right)=\mathbf{B}u\left(t\right), \tag{1}$$

where $\mathbf{C}\in\mathbb{R}^{n\times n}$ and $\mathbf{G}\in\mathbb{R}^{n\times n}$ represent the linear capacitance inductance matrix and the conductance matrix, respectively. $\mathbf{B}\in\mathbb{R}^{n\times p}$ is the input matrix. $x\left(t\right)\in\mathbb{R}^{n\times 1}$ denotes the solution vector and $u\left(t\right)\in\mathbb{R}^{p\times 1}$ the p input sources.

Conventional Krylov subspace based MOR tackles (1) in the frequency-domain[11]. Taking the Laplace transform of (1) and assuming a zero initial condition $x_0=0$ yields

$$s\mathbf{C}x\left(s\right)+\mathbf{G}x\left(s\right)=\mathbf{B}u\left(s\right). \tag{2}$$

We define $\mathbf{M}=-\mathbf{G}^{-1}\mathbf{C}$ and $\mathbf{R}=\mathbf{G}^{-1}\mathbf{B}$. The transfer function of (2) is given as

$$\mathbf{Y}\left(s\right)=\left(\mathbf{I}_n-s\mathbf{M}\right)^{-1}\mathbf{R}, \tag{3}$$

Note that the output matrix $\mathbf{L}$ does not appear in the above formulation, since in P/G network analysis we are interested in the entire solution vector instead of some selected outputs.

Mainstream MOR methods then start with computing an orthonormal basis $\mathbf{V}\in\mathbb{R}^{n\times mp}$ of the mth order block Krylov subspace $\mathcal{K}_m\left(\mathbf{M},\mathbf{R}\right)=span(\mathbf{R},\mathbf{MR},\mathbf{M}^2\mathbf{R},...,\mathbf{M}^{m-1}\mathbf{R})$ using the block Arnoldi algorithm, which satisfies the following relation

$$\begin{aligned}\mathbf{M}\mathbf{V}_M&=\mathbf{V}_M\mathbf{H}_M+v_{m+1}h_{m+1,m}e_m^T, \\ \mathbf{V}_M^T\mathbf{M}\mathbf{V}_M&=\mathbf{H}_M \\ \mathbf{V}_M^T\mathbf{V}_M&=\mathbf{I_m}\end{aligned} \tag{4}$$

where $\mathbf{H}_M\in\mathbb{R}^{mp\times mp}$ is an upper Hessenberg matrix.

Applying the change of variable $x=\mathbf{V}_Mz$ and multiplying (1) by $\mathbf{V}^T\mathbf{G}^{-1}$, one obtains the $m\times m$ reduced system

$$\begin{aligned}\mathbf{V}_M^T\left(-\mathbf{G}^{-1}\mathbf{C}\right)\mathbf{V}_M\dot{z}\left(t\right)&=z\left(t\right)-\mathbf{V}_M^T\mathbf{G}^{-1}\mathbf{B}u\left(t\right), \\ \mathbf{H}_M\dot{z}\left(t\right)&=z\left(t\right)-\mathbf{V}_M^T\mathbf{G}^{-1}\mathbf{B}u\left(t\right), \\ \dot{z}\left(t\right)&=\mathbf{H}_M^{-1}z\left(t\right)-\mathbf{H}_M^{-1}\mathbf{V}_M^T\mathbf{G}^{-1}\mathbf{B}u\left(t\right)\end{aligned} \tag{5}$$

where the last equation represents an $mp\times mp$ ordinary differential equation (ODE) that is largely reduced in size and can be simulated in the time domain at a much lower cost.

It is well known that transfer function $\hat{\mathbf{Y}}$ of the reduced system (5)

$$\hat{\mathbf{Y}}\left(s\right)=\mathbf{V}_M\left(\mathbf{I}_m-s\mathbf{H}_M\right)^{-1}\mathbf{V}_M^T\mathbf{R} \tag{6}$$

matches the first m moments of $\mathbf{Y}$ of the original system (3) at the DC expansion point. It should also be noted that the ROM size mp grows proportionally with the total number of moments to be matched m, and the number of inputs p.

2.2 Exponential integrator

EI solves the circuit equation (1) solely in the time-domain. If the matrix $\mathbf{C}$ is non-singular, (1) can be converted to an ODE

$$\dot{x}\left(t\right)=\mathbf{A}x\left(t\right)+\mathbf{C}^{-1}\mathbf{B}u\left(t\right), \tag{7}$$

with $\mathbf{A}=-\mathbf{C}^{-1}\mathbf{G}$. Given x_n, the analytical solution of ODE (7) for x_{n+1} can be expressed in terms of matrix exponential

$$x_{n+1}=e^{\mathbf{A}h}x_n+\int_0^h e^{\mathbf{A}(h-\tau)}\mathbf{C}^{-1}\mathbf{B}u(\tau)\,d\tau, \tag{8}$$

where h is the time step size. If $\mathbf{C}$ is singular, the regularization technique in [5] can be applied to eliminate algebraic constraints and retain an ODE similar to (7).

Eq. (8) is the core equation of EI for linear circuits. It has an important implication that, provided the matrix exponential vector product (MEVP) and the nonlinear integral are evaluated exactly, the step size h can be arbitrarily large. In reality, however, one must approximate the the source term $u(\tau)$ in (8) to *analytically* evaluate the integral. If we apply a first-order piece-wise-linear (pwl) approximation, i.e.,

$$u\left(\tau\right)=u_n \tag{9}$$

Eqn (8) can be analytically integrated to

$$x_{n+1}=e^{\mathbf{A}h}x_n+h\varphi_1\left(\mathbf{A}h\right)\mathbf{C}^{-1}\mathbf{B}u_n \tag{10}$$

The φ functions of order 0, 1 and 2 are defined as

$$\varphi_0\left(x\right)=e^x,\ \varphi_1\left(x\right)=\frac{e^x-1}{x},\ \varphi_2\left(x\right)=\frac{e^x-x-1}{x^2}$$

Normally the multiple φ functions in (10) would be combined to a one-exp formulation of a slightly augmented matrix [1]

$$x_{n+1}=\exp\left(\begin{bmatrix}\mathbf{A} & \mathbf{C}^{-1}\mathbf{B}u_n\\ 0 & 0\end{bmatrix}h\right)\begin{bmatrix}x_n\\ 1\end{bmatrix}=\exp\left(\tilde{\mathbf{A}}h\right)\tilde{x}_n, \tag{11}$$

so that the computation boils down to evaluate one MEVP.

Due to the potentially very large matrix size (on the order of millions in, e.g., post layout verification), Krylov subspace approximation is preferred to numerically evaluate (11). The method applies an orthogonal projection of the MEVP onto the Krylov subspace $\mathcal{K}_m(\mathbf{A},v)=span(v,\mathbf{A}v,\mathbf{A}^2v,...,\mathbf{A}^mv)$ with $m\ll N$ [12], i.e.,

$$e^{\mathbf{A}h}v\approx\mathbf{V_m}e^{(\mathbf{V_m}^T\mathbf{A}\mathbf{V_m})h}\mathbf{V_m}^Tv=\beta\mathbf{V_m}e^{\mathbf{H_m}h}e_1 \tag{12}$$

where $\beta=\|v\|_2$, $\mathbf{V}_m$ is an orthonormal basis of $\mathcal{K}_m(\mathbf{A},v)$ and $\mathbf{H}_m$ a small upper Hessenberg matrix. $e^{\mathbf{H}_mh}$ can be conveniently evaluated by a dense matrix approach [9]. A posteriori residual estimate is available to evaluate the approximation quality [20]

$$res=\beta h_{m+1,m}\mathbf{C}_mv_{m+1}e_m^T\varphi_1(\mathbf{H}_m)e_1. \tag{13}$$

An alternative Krylov subspace is the rational (or shift-and-invert) Krylov subspace $\mathcal{K}_m\left((\mathbf{I}-\gamma\mathbf{A})^{-1},v\right)$, where γ is a shift parameter. The MEVP approximation becomes

$$e^{\mathbf{A}h}v\approx\beta\mathbf{V}_me^{(I-\mathbf{H}_m^{-1})h/\gamma}e_1=\beta\mathbf{V}_me^{\mathcal{H}_mh}e_1, \tag{14}$$

where $\mathbf{V}_m$ and $\mathbf{H}_m$, similarly, are the orthonormal basis and the corresponding orthnormalization coefficient matrix of the rational Krylov subspace $\mathcal{K}_m\left((I-\gamma\mathbf{A})^{-1},v\right)$. The matrix $\mathcal{H}_m=\left(I-\mathbf{H_m}^{-1}\right)$

can be considered as an approximation of $\mathbf{A}$ in that subspace. The corresponding posteriori residual is given by

$$res = \beta h_{m+1,m}(\mathrm{C}/\gamma + \mathrm{G})v_{m+1}e_m^T\mathbf{H_m}^{-1}e^{\mathcal{H}_m h}e_1. \tag{15}$$

3 RELATIONS BETWEEN EI AND MOR

3.1 Main Theorem

While serving similar goals of accelerating transient circuit simulation, MOR and EI have long been considered two separate research domains. In this section we will show that they are in fact intimately related. Specifically, we will examine two solutions computed by MOR and EI in the following manners

(1) Using Krylov subspace based MOR to reduce the original DAE system in the frequency domain, then advancing the reduced system one step in the time domain with the EI formulation.
(2) Using EI to directly advance the original DAE system one step in the time domain;

To simplify the derivation, we assume that $\mathbf{C}$ is non-singular and all inputs are constant within one time step. Next we prove the main theorem.

Lemma 3.1. *With zero initial condition $x_0 = 0$, a single-column $\mathbf{b}$ matrix and a real expansion point $s_0 = 1/\gamma$, let $\mathbf{V}_M \in \mathcal{R}^{N\times m}$ be an m dimensional orthonormal basis with*

$$\begin{aligned} colspan\{\mathbf{V}_M\} &= \mathcal{K}_m\left(\mathbf{M},\ \mathbf{R}\right) \\ &= \mathcal{K}_m\left(-(s_0\mathbf{C}+\mathbf{G})^{-1}\mathbf{C},\ \ (s_0\mathbf{C}+\mathbf{G})^{-1}\mathbf{b}\right) \end{aligned} \tag{16}$$

and $\mathbf{V}_E \in \mathcal{R}^{N\times(m+1)}$ be an $m+1$ dimensional orthonormal basis with

$$colspan\{\mathbf{V}_E\} = \mathcal{K}_m\left(\left(\mathbf{I}-\gamma\tilde{\mathbf{A}}\right)^{-1},\ \tilde{x}_n\right) \tag{17}$$

then

$$\mathbf{V}_E = \begin{bmatrix} \mathbf{o} & \mathbf{V}_M \\ 1 & \mathbf{o}^T \end{bmatrix} \tag{18}$$

where $\mathbf{o} \in \mathcal{R}^{N\times 1}$ is a zero column vector.

Proof. Denote $\mathbf{K} = s_0\mathbf{C} + \mathbf{G}$ and assume without loss of generality that $\left\|\mathbf{K}^{-1}\mathbf{b}\right\| = 1$. Let $\mathbf{V}_M = \left[v_0^M, v_1^M, \cdots v_{m-1}^M\right]$ and $\mathbf{V}_E = \left[v_0^E, v_1^E, \cdots v_m^E\right]$, then $v_0^M = \mathbf{K}^{-1}\mathbf{b}$ and $v_0^E = \begin{bmatrix} o^T & 1 \end{bmatrix}^T$ since $x_n = 0$.

To obtain v_1^E, we first calculate

$$w_1^E = \left(\mathbf{I}-\gamma\tilde{\mathbf{A}}\right)^{-1}v_0^E = \begin{bmatrix} \mathbf{K}^{-1}\mathbf{b} \\ 1 \end{bmatrix} \tag{19}$$

Orthogonalization with respect to v_0^E yields

$$v_1^E = w_1^E - \left(\left(w_1^E\right)^T v_0^E\right)v_0^E = \begin{bmatrix} \mathbf{K}^{-1}\mathbf{b} \\ 0 \end{bmatrix} = \begin{bmatrix} v_0^M \\ 0 \end{bmatrix} \tag{20}$$

For v_2^E, we calculate by some derivation

$$w_2^E = \left(\mathbf{I}-\gamma\tilde{\mathbf{A}}\right)^{-1}v_1^E = \begin{bmatrix} (s_0\mathbf{K}^{-1}\mathbf{C}+\mathbf{I})\mathbf{K}^{-1}\mathbf{b} \\ 1 \end{bmatrix} \tag{21}$$

Orthogonalization with respect to v_0^E and v_1^E yields

$$\begin{aligned} w_2^E &= w_2^E - \left(\left(w_2^E\right)^T v_0^E\right)v_0^E - \left(\left(w_2^E\right)^T v_1^E\right)v_1^E \\ &= \begin{bmatrix} \left(s_0\mathbf{K}^{-1}\mathbf{C}+\mathbf{I}\right)v_0^M \\ 0 \end{bmatrix} - s_0\left[\left(v_0^M\right)^T\left(s_0\mathbf{K}^{-1}\mathbf{C}+\mathbf{I}\right)v_0^M\right]\begin{bmatrix} v_0^M \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} s_0\left(\mathbf{K}^{-1}\mathbf{C}v_0^M - \left[\left(v_0^M\right)^T\left(\mathbf{K}^{-1}\mathbf{C}\right)v_0^M\right]v_0^M\right) \\ 0 \end{bmatrix} \end{aligned} \tag{22}$$

It is key to note that the upper element in the last equation is the recursive generation formula of v_1^M. After normalization, we have

$$v_2^E = \begin{bmatrix} v_1^M \\ 0 \end{bmatrix} \tag{23}$$

Continuing the above process, we have

$$v_i^E = \begin{bmatrix} v_{i-1}^M \\ 0 \end{bmatrix}, 0 < i \le m \tag{24}$$

which completes the proof. □

Theorem 3.2. *With the assumptions stated in Lemma 3.1, EI, when solved with a $(m+1)$-dimensional rational Krylov subspace with γ, is equivalent to applying a Krylov subspace based MOR, with the expansion point $s_0 = 1/\gamma$ and the subspace dimension m, to the original system, and advancing the reduced system one step in the time domain using again the EI approach.*

Proof. We start from the MOR side. Rewrite (1) as

$$-\mathbf{K}^{-1}\mathbf{C}\dot{x} - \left(\mathbf{I} - s_0\mathbf{K}^{-1}\mathbf{C}\right)x = -\mathbf{K}^{-1}\mathbf{b}u \tag{26}$$

By (4) and (16), we have

$$-\mathbf{K}^{-1}\mathbf{C}\mathbf{V}_M = \mathbf{V}_M\mathbf{H}_M + v_{m+1}h_{m+1,m}e_m^T \tag{27}$$

$$\mathbf{V_M}^T\left(-\mathbf{K}^{-1}\mathbf{C}\right)\mathbf{V_M} = \mathbf{H_M} \tag{28}$$

Applying (28) and the variable transform $x = V_M z$ to (26) yields

$$\begin{aligned} \mathbf{H_M}\dot{z} &= \left(\mathbf{I_M} + s\mathbf{H_M}\right)z - \mathbf{V_M}^T\mathbf{K}^{-1}\mathbf{b}u \\ \dot{z} &= \left(s\mathbf{I_M} + \mathbf{H_M}^{-1}\right)z - \mathbf{H_M}^{-1}\mathbf{V_M}^T\mathbf{K}^{-1}\mathbf{b}u \end{aligned} \tag{29}$$

If advancing (29) for one time step from $t_0 = 0$ to $t_1 = h$ using EI as the time integration method, and noting that $z_0 = 0$ and $u(t) = u_0$, we have

$$z_1 = e^{\left(s_0\mathbf{I_M}+\mathbf{H_M}^{-1}\right)h}z_0 - h\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{H_M}^{-1}\mathbf{V_M}^T K^{-1}bu_0 \tag{30}$$

$$= -h\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{H_M}^{-1}e_1u_0 \tag{31}$$

and

$$x_1^{MOR} = \mathbf{V_M}z_1 = -h\mathbf{V_M}\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{H_M}^{-1}e_1u_0 \tag{32}$$

On the EI side, the new solution is obtained by (34), which applies (12) to evaluate (11) with V_E defined in (18). Note that to obtain the EI approximation in (14), we effectively exploit the following approximation

$$\mathbf{V_M}^T\mathbf{A}\mathbf{V_M} \approx s_0\mathbf{I} + \mathbf{H_M}^{-1} \tag{33}$$

$$
\begin{aligned}
x_1^{EI} &= \exp\left(\tilde{\mathbf{A}}h\right)\tilde{v} \approx \mathbf{V_E}\exp\left(\mathbf{V_E}^T\tilde{\mathbf{A}}\mathbf{V_E}h\right)\mathbf{V_E}^T\tilde{v} \\
&= \begin{bmatrix} \mathbf{o} & \mathbf{V_M} \\ 1 & \mathbf{o}^T \end{bmatrix} \exp\left(\begin{bmatrix} \mathbf{o}^T & 1 \\ \mathbf{V_M}^T & \mathbf{o} \end{bmatrix}\begin{bmatrix} \mathbf{A} & \mathbf{C}^{-1}\mathbf{b}u_0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \mathbf{o} & \mathbf{V_M} \\ 1 & \mathbf{o}^T \end{bmatrix}h\right)\begin{bmatrix} \mathbf{o}^T & 1 \\ \mathbf{V_M}^T & \mathbf{o} \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{o} & \mathbf{V_M} \\ 1 & \mathbf{o}^T \end{bmatrix} \exp\left(\begin{bmatrix} 0 & 0 \\ \mathbf{V_M}^T\mathbf{C}^{-1}\mathbf{b}u_0 & \mathbf{V_M}^T\mathbf{A}\mathbf{V_M} \end{bmatrix}h\right)\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{o} & \mathbf{V_M} \\ 1 & \mathbf{o}^T \end{bmatrix}\begin{bmatrix} 1 & 0 \\ h\varphi_1\left(\mathbf{V_M}^T\mathbf{A}\mathbf{V_M}h\right)\mathbf{V_M}^T\mathbf{C}^{-1}\mathbf{b}u_0 & \exp\left(\mathbf{V_M}^T\mathbf{A}\mathbf{V_M}h\right) \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= h\mathbf{V_M}\varphi_1\left(\mathbf{V_M}^T\mathbf{A}\mathbf{V_M}h\right)\mathbf{V_M}^T\mathbf{C}^{-1}\mathbf{b}u_0
\end{aligned}
\tag{25}
$$

which drops the rank-one update term on the right hand side of (27) [8]. Then (25) is approximated by

$$
\begin{aligned}
x_1^{EI} &\approx h\mathbf{V_M}\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{V_M}^T\mathbf{C}^{-1}\mathbf{b}u_0 \qquad (34) \\
&= h\mathbf{V_M}\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{V_M}^T\mathbf{C}^{-1}\mathbf{K}\mathbf{K}^{-1}\mathbf{b}u_0 \\
&= h\mathbf{V_M}\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{V_M}^T\left(\mathbf{K}\mathbf{C}^{-1}\right)^{-1}\mathbf{V_M}e_1u_0 \\
&= -h\mathbf{V_M}\varphi_1\left(\left(s_0\mathbf{I_M} + \mathbf{H_M}^{-1}\right)h\right)\mathbf{H_M}^{-1}e_1u_0
\end{aligned}
$$

where the last equality utilizes again the approximation of (33). Comparing (32) and (34), one can see that

$$
\mathbf{x}_1^{MOR} = \mathbf{x}_1^{EI}, \tag{35}
$$

which completes the proof. □

Corollary 3.3. *Theorem 3.2 also holds for inhomogeneous initial condition $x_0 \neq 0$.*

Proof. For a system with nonzero x_0, it is common to apply a variable transformation $\hat{x}(t) = x(t) - x_0$ and convert (1) into

$$
\mathbf{C}\dot{\hat{x}}(t) + \mathbf{G}\hat{x}(t) = \mathbf{B}u(t) - \mathbf{G}x_0, \tag{36}
$$

with $\hat{x}(t) = 0$.

The term $\mathbf{G}x_0$ can be viewed as an extra fixed input to the system [2]. One can then decompose (36) into two sub-systems, one with input $\mathbf{B}u(t)$ and one with $\mathbf{G}x_0$, and add up their solutions at the same time point exploiting the linear superposition property. The extra system to solve is

$$
\mathbf{C}\dot{\hat{x}}(t) + \mathbf{G}\hat{x}(t) = -\mathbf{G}x_0, \tag{37}
$$

for which the equivalence between EI and MOR follows the same proof of Theorem 3.2. □

Corollary 3.4. *The EI solution obtained by* (25) *using the $m+1$-dimensional $\mathbf{V}_E$ in* (18) *matches the first m moment of the exact solution around the expansion point s_0.*

Proof. It is well known that reduced system in (29) with the m-dimensional $\mathbf{V}_M$ from (16) matches the first m moments in transfer function of the original system (1) around s_0, see the proof in e.g. [13]. Since EI with $\mathbf{V}_E$ yields the same x_1, it also matches the first m moments of the original solution around s_0. □

3.2 Discussions

The central difference between MOR and EI is that MOR adopts a "once-and-for-all" strategy, which targets to produce a reduced model that maintains the original I/O relation for *any possible* inputs. Such input (and initial condition) independence sometimes lead to an overclaim because inputs in reality are usually not arbitrary. The computational effort for finding the desired reduced model may thus be unnecessarily high, especially for many-input and wide-spectrum systems.

EI, on the other hand, operates in an "economic" mode that a simplified equivalent MOR is applied in each time step. Each MOR works with only one (different) input and the reduced model is expected to be accurate only within that step, which significantly lowers the difficulty and cost of finding a quality reduced model. In other words, EI sacrifices re-usability–the projection basis and the ROM cannot be re-used even in the next time step since the starting vector is changed–in exchange of MOR efficiency. A few further remarks pertinent to their applications in transient circuit simulation are in order:

(1) EI does not do MOR explicitly, but the numerical procedure of EI amounts to performing the two key steps (reduction and ROM simulation) from an MOR perspective. The ability of EI to match $m-1$ moments of the original solution with an (m)-dimensional subspace in each step provides an alternative perspective to understand its advantages over other BDF-type methods that generally do not match moments.

(2) MOR reduces a system in the frequency-domain but uses the reduced model in the time-domain. It cannot foresee the actual transient input waveform the ROM will be fed with when performing the reduction. Therefore, the automatic selection of key parameters (e.g., the number and location of expansion points and the number of moments to match), plus the error control, have been long-standing issues for traditional MOR. In contrast, EI is a pure time-domain method and has well-defined residual/error estimates, thus the step size and subspace dimension (i.e., the number of moments to match) can be easily and wisely determined.

(3) The stability and passivity of the implicit MOR process performed by EI are not investigated in this work. However, it is less likely a concern in practice since the reduced model is used in only *one* time step instead of the entire transient simulation in the conventional MOR context.

(4) Re-usability is one major advantage of MOR over EI. The ROM can be efficiently used in transient simulation for different inputs and time point settings without re-generating the Krylov subspace. This feature actually motivates the development of a new method in the next section.

4 HYBRID EI AND MOR METHOD FOR INPUT MISALIGNMENT

In P/G network analysis, the contribution from nonlinear devices are modeled as a large number of independent current sources with typically pulse or PWL waveforms. The typical current source waveforms of a P/G network are visualized in Fig. 1.

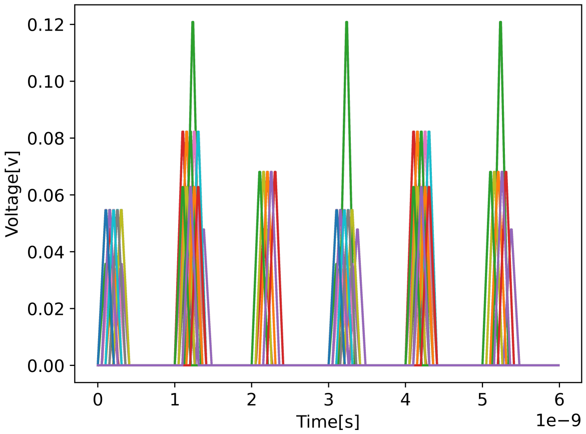

Figure 1: Current source waveforms in P/G network analysis.

The input misalignment problem can be illustrated by a simple example with 3 pulse inputs in Fig. 2 (a). In principle, EI only needs one step for each breakpoint with a sufficiently large subspace dimension. However, the misaligned inputs result in 12 distinct breakpoints in the whole simulation span. To fulfill the PWL requirement that *every* input is linear within the same step, EI will need at least 12 steps. If input 1 and 2 are aligned, the required number of steps would be reduced to 8. Considering the thousands of inputs with very different waveform characteristics in a P/G network, the input misalignment will induce many breakpoints that severely limit the max step size.

In Section 3 we reveals that EI is advantageous for handing many-input systems because all the inputs, regardless of their number, will be condensed into a single vector in each time step as the input matrix for the equivalent MOR process. The cost of generating the projection matrix is therefore independent of the amount of I/O ports. However, the cost of EI is sensitive to the number of time points since a new, full-scale "MOR" process is needed in each time step. To the contrary, the cost of MOR is sensitive to the number of ports, but insensitive to the time points in subsequent simulations because the reduced system is typically small.

Motivated by the differences and underlying equivalence between EI and MOR, we develop a hybrid EI-MOR approach to address the misaligned input issue. We first divide the original inputs $u(t)$ into a majority group of well-aligned inputs $u_{EI}(t)$, which will be handled by EI, and a minority group of misaligned inputs $u_{MOR}(t)$ to be used for MOR. The input matrix $\mathbf{B}$ is also partitioned correspondingly into $\mathbf{B}_{EI}$ and $\mathbf{B}_{MOR}$. By removing the MOR inputs $u_{MOR}(t)$, the number of breakpoints induced by the EI inputs u_{EI} is reduced. For instance, in Fig. 2 (b) and (c), Inputs 2 is selected as the MOR input, and inputs 1 and 3 are left in the EI input set. Hence the number of time points in EI is reduced from 12 to 8. The MOR input matrix $\mathbf{B}_{MOR}$ is used in a MOR process to generate a reduced model, which will be simulated individually. The EI solution and MOR solution are finally superimposed at the corresponding time points to generate the final solution.

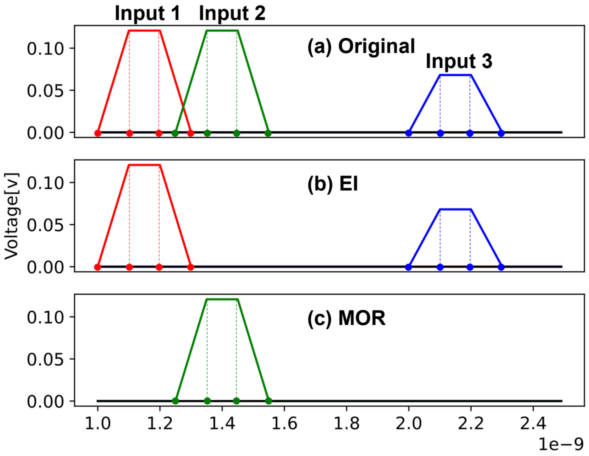

Figure 2: Illustration of input grouping.

Note that the terms "well-aligned" and "misaligned" are of a relative sense. A simple heuristic for selecting MOR inputs is to pick those inputs that would reduce the number of breakpoints by the most extent if removed from the original input set. Let k_0 denote the total number of breakpoints dictated by the current set of inputs and k_i be the number of breakpoints if input i is removed, then the input selected for MOR is

$$i_{MOR} = \arg\max_{i} \left(k_0 - k_i\right). \tag{38}$$

The term "minority" implies that only a few inputs should be selected for MOR to avoid performance degradation. Note that (38) is merely one way to select MOR inputs, other heuristics based on different considerations are also possible.

To combine the solutions from EI and MOR to form the final solution, let $\mathcal{S}_{All}$ denote the set of breakpoints from the original inputs and $\mathcal{S}_{EI}$ be that from the EI inputs. The reduced system from MOR will be simulated at every breakpoints in $\mathcal{S}_{All}$. For each breakpoint $i \in \mathcal{S}_{All} \cap \mathcal{S}_{EI}$, an EI solution is computed by (14) using a new Arnoldi process. For each breakpoint $i \in \mathcal{S}_{All} \backslash \mathcal{S}_{EI}$, we do not perform a new Arnoldi process but obtain the EI solution by scaling the solution from the nearest next breakpoint in $\mathcal{S}_{EI}$ at nearly no cost, cf. (14)

$$x_i^{EI} = \beta \mathbf{V}_m^* e^{\mathcal{H}_m^* h_i} e_1, \tag{39}$$

where $\mathbf{V}_m^*$ and $\mathcal{H}_m^*$ are the Krylov subspace matrices that have been generated for point i^*, the nearest next time point to i and $i^* \in \mathcal{S}_{EI}$. h_i is the step size from the previous EI point to point i. Eq. (39) utilizes the scaling invariant property of Krylov subspace [7] and the fact that the input u_i lies within the same PWL interval with u_i^*. Finally, all the EI and MOR solutions are superimposed at $\mathcal{S}_{All}$. A flowchart of the EI-MOR method is shown in Fig. 3.

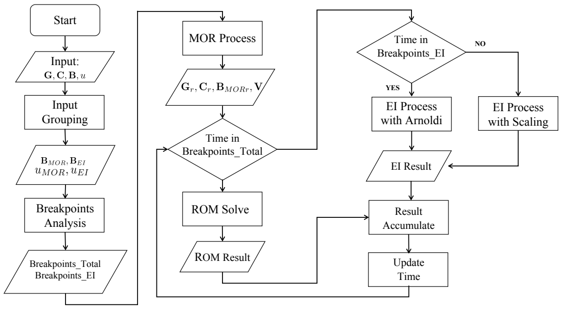

Figure 3: EI-MOR Algorithmic Flowchart.

Table 1: Testcases Specification

Case	Node	VSource	ISource
ibmpg1t	39681	14308	5883
ibmpg2t	164238	330	18535
ibmpg3t	1041535	955	114191

5 NUMERICAL RESULTS

We implement the EI-MOR method in a Python-based circuit simulator Ahkab [14]. We use a set of IBM P/G network netlists [10] as test cases with the specifications in Table 1 and PRIMA [11] as the MOR engine. The subspace dimension in EI m_{EI} is dynamically determined with a relative tolerance $reltol_{EI} = 10^{-3}$. The subspace dimension of MOR m_{MOR} is pre-determined for each case. The reduced system is solved by the EI solver with the matrix exponential term directly evaluated by the expm function from Scipy.

We start with a numerical verification of Theorem 3.2. We use ibmpg1t as the testcase with one current source and an all-one x_0. Three expansion points $s_0 = 10^7, 10^8, 10^9$ are used for MOR (corresponding to $\gamma = 10^{-7}, 10^{-8}, 10^{-9}$ in EI), each matching 4, 6, 8 moments. The solution x computed by EI and MOR with $h = 10^{-11}$ are compared in Table 2. The very small differences under all tested combinations confirm Theorem 3.2 that the two methods indeed generate the same solution.

Next, we confirm the validity of the “divide-and-superposition” scheme of EI-MOR described in Section 4. Case ibmpg1t is used again with 5883 current sources, among which 3 sources are selected for MOR and 5880 left to EI. The transient simulation waveforms are shown in Fig 4, where EI_{all} refers to the solution from EI using the original inputs (the reference), EI the solution from EI using the EI inputs only, MOR the solution from the reduced system simulated in the time-domain, and $EI+MOR$ the solution from superimposing the EI and MOR solutions. One can clearly see that both EI and MOR inputs have non-trivial contribution to the transient responses, and produce the correct total solution when combined properly by the proposed EI-MOR method.

Table 2: Numerical Verification of MOR-EI Equivalence

Expansion Point	Moments	$\frac{\lVert x^{EI}-x^{MOR}\rVert}{\lVert x^{EI}\rVert}$
1e7	4	2.5418×10^{-14}
	6	8.1615×10^{-14}
	8	1.2139×10^{-13}
1e8	4	2.3310×10^{-14}
	6	5.9894×10^{-14}
	8	1.0884×10^{-13}
1e9	4	4.6395×10^{-14}
	6	7.7582×10^{-13}
	8	6.3915×10^{-12}

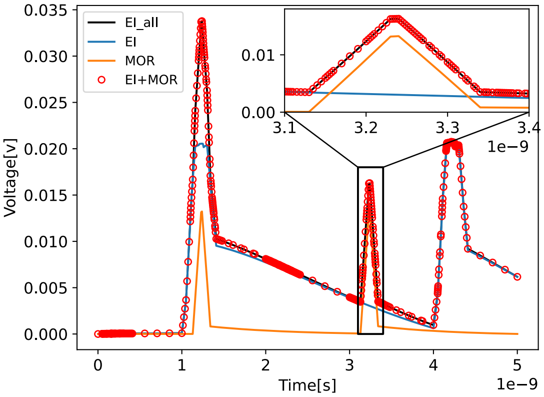

Figure 4: Transient waveform of EI-MOR.

Finally, we demonstrate the performance of the hybrid EI-MOR method. EI refers to using EI with all the original inputs, and EI-MOR is the proposed hybrid method with different numbers of MOR inputs. For all the three cases, we select 3 and 6 inputs for MOR based on (38) and compare their performance data in Table 3. bp_{All} and bp_{EI} denote the numbers of breakpoints due to the original inputs and the remaining EI inputs after the MOR inputs are removed. $Arnoldi_{EI}$ refers to the number of Arnoldi iterations, the most computationally intensive step in EI, consumed in different settings. m_{MOR} and ROM Size indicate the number of moments matched around a fixed expansion point $s_0 = 10^8$ and the size of the obtained reduced model. t_{EI}, t_{MOR}^{Reduce}, t_{MOR}^{Sim} and t_{total} denote the simulation time with the EI inputs only, the MOR reduction time, the simulation time of the reduced system, and the total runtime, respectively.

Several observations are in order. First, one can see that removing some MOR inputs substantially reduces the number of breakpoints involved in the EI simulation, by nearly 50% with 6 MOR inputs.

Table 3: Performance of EI-MOR

Case	Method	Isource	bp_{All}	bp_{EI}	$Arnoldi_{EI}$	$t_{EI}(s)$	m_{MOR}	ROM Size	$t_{MOR}^{Reduce}(s)$	$t_{MOR}^{Sim}(s)$	$t_{total}(s)$
ibmpg1t	EI	5883	517	517	16006	225.39	-	-	-	-	225.39
	EI-MOR	5880(EI)3(MOR)	517	342	12570	177.14	15	45	1.08	17.31	195.52
	EI-MOR	5877(EI)6(MOR)	517	277	8496	122.41	15	90	3.48	28.60	154.48
ibmpg2t	EI	18535	533	533	15840	1173.51	-	-	-	-	1173.51
	EI-MOR	18532(EI)3(MOR)	533	389	11509	865.06	15	45	5.32	86.77	957.16
	EI-MOR	18529(EI)6(MOR)	533	277	7700	634.85	15	90	17.12	84.71	736.68
ibmpg3t	EI	114191	509	509	13882	8420.87	-	-	-	-	8420.87
	EI-MOR	114188(EI)3(MOR)	509	375	10067	6306.15	20	60	93.33	471.59	6871.07
	EI-MOR	114185(EI)6(MOR)	509	277	7318	4733.61	20	120	194.13	672.05	5599.79

The number of Arnoldi iterations and the runtime of EI simulation are also proportionally decreased. Second, the first 3 MOR inputs are more effective than the second 3 in reducing the amount of breakpoints since they are picked according to their ability for this purpose. For instance, in case 1 removing the first 3 MOR inputs reduce 175 breakpoints, while the second 3 reduce only 65 points. Lastly, the MOR reduction time is not significant because of the relative small model size. For the same reason, the runtime for simulating the reduced model is mainly consumed by the backward projection $\mathbf{V}_M z$ with a linear complexity of ROM size.

6 CONCLUSION

This paper presented a new approach to address the input misalignment problem of EI for transient P/G network simulation. We first proved that EI is equivalent to a moment-matching MOR in each time step, with the same expansion point and m matched moments if $(m+1)$-dimensional Krylov subspace is used. This equivalence provides new sights to bridge the two separate domains of EI and MOR, and hopefully will facilitate further understanding and development in transient circuit simulation. Next we proposed a hybrid EI-MOR method which divides the inputs into the EI part and the MOR part. The two sets of inputs are handled separately by EI and MOR, taking advantage of their own distinct characteristics, to reduce the number of time points in the EI solution. Numerical results have demonstrated the accuracy and efficiency of the proposed method. The EI-MOR method can also be easily extended to other linear circuit simulation scenarios.

ACKNOWLEDGMENTS

The authors thank the financial support of the Special Funds of the National Natural Science Foundation of China (Grant No. 62141410), the Key Area Research and Development Program of Guangdong Province, China (Grant No. 2021B1101270003) and the Key Program of the National Natural Science Foundation of China (Grant No. 62034007).

REFERENCES

[1] A. H. Al-Mohy and N. J. Higham. 2011. Computing the Action of the Matrix Exponential, with an Application to Exponential Integrators. *SIAM Journal on Scientific Computing* 33, 2 (2011), 488–511.

[2] Christopher Beattie, Serkan Gugercin, and Volker Mehrmann. 2017. Model reduction for systems with inhomogeneous initial conditions. *Systems & Control Letters* 99 (2017), 99–106.

[3] Pengwen Chen, Chung-Kuan Cheng, Dongwon Park, and Xinyuan Wang. 2018. Transient Circuit Simulation for Differential Algebraic Systems using Matrix Exponential. In *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. 1–6. https://doi.org/10.1145/3240765.3264636

[4] Quan Chen. 2020. A Robust Exponential Integrator Method for Generic Nonlinear Circuit Simulation. In *2020 57th ACM/IEEE Design Automation Conference (DAC)*. 1–6. https://doi.org/10.1109/DAC18072.2020.9218556

[5] Quan Chen. 2022. EI-NK: A Robust Exponential Integrator Method With Singularity Removal and Newton–Raphson Iterations for Transient Nonlinear Circuit Simulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41, 6 (2022), 1693–1703. https://doi.org/10.1109/TCAD.2021.3098749

[6] Quan Chen, W. Schoenmaker, Shih-Hung Weng, Chung-Kuan Cheng, Guan-Hua Chen, Li-Jun Jiang, and Ngai Wong. 2012. A fast time-domain EM-TCAD coupled simulation framework via matrix exponential. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. 422–428.

[7] Q. Chen, S. Weng, and C. Cheng. 2012. A Practical Regularization Technique for Modified Nodal Analysis in Large-Scale Time-Domain Circuit Simulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 31, 7 (July 2012), 1031–1040.

[8] Tanja Göckler. 2014. *Rational Krylov subspace methods for phi-functions in exponential integrators*. Ph.D. Dissertation. Karlsruhe, Karlsruher Institut für Technologie (KIT), Diss., 2014.

[9] Nicholas J. Higham. 2005. The Scaling and Squaring Method for the Matrix Exponential Revisited. *SIAM J. Matrix Anal. Appl.* 26, 4 (April 2005), 1179–1193. https://doi.org/10.1137/04061101X

[10] Sani R Nassif. 2008. Power grid analysis benchmarks. In *2008 Asia and South Pacific Design Automation Conference*. IEEE, 376–381.

[11] Altan Odabasioglu, Mustafa Celik, and Lawrence T Pileggi. 1998. PRIMA: Passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on computer-aided design of integrated circuits and systems* 17, 8 (1998), 645–654.

[12] Yousef Saad. 1992. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.* 29, 1 (1992), 209–228.

[13] Behnam Salimbahrami and Boris Lohmann. 2006. Order reduction of large scale second-order systems using Krylov subspace methods. *Linear Algebra Appl.* 415, 2-3 (2006), 385–405.

[14] Giuseppe Venturini. 2015. Ahkab: an open-source SPICE-like interactive circuit simulator. https://ahkab.readthedocs.io/en/latest/

[15] Shih-Hung Weng, Quan Chen, and Chung-Kuan Cheng. 2012. Time-Domain Analysis of Large-Scale Circuits by Matrix Exponential Method With Adaptive Control. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 31, 8 (2012), 1180–1193.

[16] Shih-Hung Weng, Quan Chen, Ngai Wong, and Chung-Kuan Cheng. 2012. Circuit simulation via matrix exponential method for stiffness handling and parallel processing. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, 407–414.

[17] Hao Zhuang, Shih-Hung Weng, and Chung-Kuan Cheng. 2013. Power Grid Simulation using Matrix Exponential Method with Rational Krylov Subspaces. In *IEEE 9th International Conference on ASIC (ASICON)*. IEEE, 369–372.

[18] Hao Zhuang, Shih-Hung Weng, Jeng-Hau Lin, and Chung-Kuan Cheng. 2014. MATEX: A Distributed Framework for Transient Simulation of Power Distribution Networks. In *IEEE/ACM Design Automation Conference (DAC)*. IEEE, 81:1–81:6. https://doi.org/10.1145/2593069.2593160

[19] Hao Zhuang, Wenjian Yu, Ilgweon Kang, Xinan Wang, and Chung-Kuan Cheng. 2015. An Algorithmic Framework for Efficient Large-scale Circuit Simulation Using Exponential Integrators. In *IEEE/ACM Design Automation Conference (DAC)*.

IEEE, 163:1–163:6.

[20] H. Zhuang, W. Yu, S. Weng, I. Kang, J. Lin, X. Zhang, R. Coutts, and C. Cheng. 2016. Simulation Algorithms With Exponential Integration for Time-Domain Analysis of Large-Scale Power Delivery Networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 35, 10 (Oct 2016), 1681–1694.