Projection frameworks for model reduction of weakly nonlinear systems

Joel R. Phillips
Cadence Berkeley Laboratories, San Jose, CA 95134

Abstract

In this paper we present a generalization of popular linear model reduction methods, such as Lanczos- and Arnoldi-based algorithms based on rational approximation, to systems whose response to interesting external inputs can be described by a few terms in a functional series expansion such as a Volterra series. The approach allows automatic generation of macromodels that include frequency-dependent nonlinear effects.

1 Introduction

Model reduction refers to the procedure of automatic generation of system macromodels from detailed descriptions. These macromodels can be used to perform rapid system-level simulation of engineering designs that are too complicated to analyze at the detailed component level. The advantage of the reduction approach is that because the macromodels are generated from detailed physical descriptions of the system components, the influence of complicated second-order physical effects can be included at the system level. Thus an essential feature of reduction approaches is a relatively thorough control and assessment of approximation errors that is gained by formal analysis of the reduction algorithms.

The most successful algorithms for reduction of large-scale linear systems have been projection-based approaches[1, 2]. Algorithms such as PVL[3], Arnoldi methods[4], and PRIMA[5] obtain reduced models by projecting the linear equations describing the LTI model system into a subspace of lower dimension. The subspace chosen determines the approximation properties of the reduced model. These algorithms exploit the connection between Krylov subspaces and rational approximation to develop algorithms that have a known relationship to the frequency-domain characteristics of the system, for example, matching the transfer function and some of its derivatives at various points in the complex plane. Projection approaches for large-scale systems have recently been extended to time-varying linear models [6, 7].

Linear reduction algorithms are useful for many problems, for example, simulation of electrical interconnect and analysis of noise in RF systems, but fail totally in other contexts. For example, “adjacent channel power ratio” (ACPR) is a figure of merit for the distortion properties of digitally modulated RF transmission systems and therefore by definition requires the utilization of nonlinear models[8]. Microelectromechanical (MEMS)[9, 10] and power systems[11] also require nonlinear macromodeling approaches. However, very few results are available for reduction of nonlinear systems.

To illustrate some of the difficulties in developing nonlinear reduction algorithms, let us contrast the behavior of generic linear and non-linear state-space models under projection-based reduction. First consider the linear time-invariant state-space model for time $t > 0$

$$E \frac{dx}{dt} = Ax + Bu \quad y = Cx$$

(1)

where $E, A \in \mathbb{R}^{n \times n}, x(t) \in \mathbb{R}^{n}, B \in \mathbb{R}^{n \times p}, u(t) \in \mathbb{R}^{p}, y(t) \in \mathbb{R}^{q}, n$ is the system order, and $p$ and $q$ are the number of system inputs and outputs respectively. This system of differential equations defines a linear functional $L: [0, t]^{p} \rightarrow \mathbb{R}^{q}$ that maps $u(t)$ on the past time interval $[0, t]$ to the output $y(t)$ at time $t$. A projection based reduction scheme involves selecting a matrix $V$ whose columns span a “useful” subspace, and drawing an approximation to $x$, $\hat{x}$, from this subspace as $\hat{x} = Vz$. Equations for the reduced system are obtained by defining the residual $R \equiv Ax + Bu - E\hat{x}$ and requiring the residual to be orthogonal¹ to the approximation space $V$, $V^{T}R = 0$. Reduced equations are then given by

$$E \frac{d\hat{x}}{dt} = \hat{A}\hat{x} + \hat{B}u \quad \hat{y} = \hat{C}\hat{x}$$

(2)

where

$$\hat{A} \equiv V^{T}AV \quad \hat{B} \equiv V^{T}B \quad \hat{E} \equiv V^{T}EV \quad \hat{C} \equiv CV$$

(3)

The Krylov-subspace based projection schemes work for large-scale linear systems for four reasons. First, good choices for the subspace defined by $V$ exist. Rational approximation paradigms suggest matching properties of the transfer function. Second, the columns of the matrix $V$ can be efficiently obtained. Only products with $A$ or solution of linear systems $Ax = b$ are required, and thus any sparsity or special structure of the underlying linear system can be exploited. Therefore, the reduced models can be efficiently obtained, since in the worst case, only a small number of matrix-vector products are needed. Finally, the reduced models can be efficiently simulated. This is guaranteed by construction, since the reduced models have the same form as the original equations, i.e., linear systems of differential equations, but of much lower dimensionality.

In contrast, consider a system with a nonlinear state-evolution function

$$E \frac{dx}{dt} = f(x) + Bu \quad y = Cx$$

(4)

where $f: \mathbb{R}^{n} \rightarrow \mathbb{R}^{n}$. This system of differential equations also defines a functional $K: [0, t]^{p} \rightarrow \mathbb{R}^{q}$ that maps $u(t)$ on the past time interval $[0, t]$ to the output $y(t)$ at time $t$, but $K$ is nonlinear because

¹For simplicity we will work with only orthogonal projections throughout this paper, that is, models defines by the condition $V^{T}R = 0$. 
\( f(x) \) is not a linear function. We may formally apply the projection recipe to this system of equations to obtained a “reduced” model
\[
\dot{\tilde{x}} = V^T f(Vz) + B\tilde{u} \quad \tilde{y} = \tilde{C}\tilde{x}
\]
where \( \tilde{E}, \tilde{B}, \tilde{C} \) are as before. Several difficulties with this approach are apparent.

First, it is not at all clear how to choose \( V \), and even less is known about efficient computation. Approaches based on analysis of the linearized model have been proposed\cite{12, 8}, but by definition such approaches do not include information about nonlinear model properties. Heuristic approaches based on the singular value decomposition of a statistically representative sampling of the state-space \( \phi(t) \) have met with some success\cite{9, 13, 14}, but the computations are extensive and little control over model accuracy is available. Balancing based procedures exist in theory\cite{15}, but it is not clear how they may be computed. More importantly, in the general case, interpreting the term \( V^T f(Vz) \) as a “reduced” model is problematic. Since \( f \) is a nonlinear function, \( V \) may not be generally passed through the parentheses, and the only recourse to computing \( V^T f(Vz) \) may be to (1) explicitly construct \( \dot{\tilde{x}} = Vz \), (2) evaluate \( \hat{f}(\tilde{x}) = f(Vz) \), (3) compute \( V^T \hat{f}(\tilde{x}) \). As a result, efficient simulation is not guaranteed. For example, in nonlinear circuit simulation, even for circuits with tens of thousands of nodes, roughly half the simulation time is spent in evaluation of the nonlinear function \( f \). Thus, regardless of the reduction in the size of the state space, if the original function \( f \) must be evaluated, the efficiency gain in moving from the detailed to reduced model will be at most a factor of two or three.

This paper is dedicated to addressing these barriers to constructing nonlinear model reduction algorithms. The principle contributions is to extend the rational approximation paradigm to the nonlinear case, thereby providing a sound theoretical basis for selection of the subspace spanned by \( V \). Our emphasis is not so much on proposing a specific algorithm as in showing that, for sufficiently well-restricted classes of nonlinear systems, algorithms with provable approximation properties and finite computation time exist, and that these algorithms generate models with reduced dimensionality and model complexity. It is beyond the scope of this paper to provide details of a numerical implementation, but we will show how algorithms may be constructed using tools familiar from the linear model reduction problem.

### 2 Projection methods for LTI model reduction

In this section we will review results on projection and Krylov-subspace methods for model reduction\cite{1, 2}. As a prelude to model reduction, we define Krylov subspaces.

**Definition 1 (Krylov subspace)** The Krylov subspace \( K_m(A, p) \) generated by a matrix \( A \) and vector \( p \), of order \( m \), is the space spanned by the set of vectors \( \{p, Ap, A^2p, \ldots, A^{m-1}p\} \).

The essential elements of Krylov-subspace based reduction are given by

**Theorem 2.1** Suppose \( K_m(A^{-1}, p) \subset \mathcal{R}(V) \), where \( \mathcal{R}(V) \) denotes the range or column-space of \( V \), then \( V(V^TAV)^{-1}V^TB = V\tilde{A}^{-1}B \), for \( k < m \).

*Proof.* See \cite{2}.

Now return to the linear time-invariant multi-input, multi-output (MIMO) linear system of Equation (1), which is equivalent to the Laplace-domain description \( y(t) = H(s)u(t) \) where the transfer function \( H(s) \) is given by \( H(s) = C(sI - A)^{-1}B \). Projection generates rational approximants
\[
\tilde{H}(s) = \tilde{C}(s\tilde{E} - \tilde{A})^{-1}\tilde{B},
\]
and the most well known rational approximation methods (e.g., Padé approximants\cite{16}) have the property that they match the transfer function and some of its derivatives with respect to \( s \). Connecting the moments to the projection matrix \( V \) is the key to the model reduction procedure. Note that the \( k \)th derivative, or moment, of the transfer function is given by \( C\tilde{A}^{-k+1}B \).

**Theorem 2.2 (Krylov Subspace Model Reduction)** If the columns of \( V \) span \( K_m(A^{-1}E, A^{-1}B) \) then the reduced order transfer function \( \tilde{C}^T(s\tilde{E} - \tilde{A})^{-1}\tilde{B} \) matches the first \( m \) moments of the original transfer function \( C(sI - A)^{-1}B \).

*Proof.* Follows from the Taylor expansion of \( (sE - A)^{-1} \) and Theorem 2.1\cite{1, 2}.

### 3 Functional series expansions

The output \( y(t) \) of our LTI system may be expressed in the time-domain using the convolution representation
\[
y(t) = \int_0^t h(s)u(t - s)ds \tag{7}
\]
where \( h(s) \) is the kernel, or in the frequency domain, as \( Y(s) = H(s)U(s) \) where \( H(s) \) is the Laplace transform of \( h(t) \). For a nonlinear system, the analogous quantities come from a functional series expansion of the output \( y(t) \) that has the general form\cite{17, 18}
\[
y(t) = \sum_{n=1}^{\infty} y_n(t) \tag{8}
\]
where \( y_n(t) \) is the \( n \)th-order response
\[
y_n(t) = \int_0^t \cdots \int_0^t h_n(s_1, \ldots, s_n)u(t - s_1) \cdots u(t - s_n)ds_1 \cdots ds_n \tag{9}
\]
The \( n \)th term in Eq. 9 represents an \( n \)-dimensional convolution of \( n \) products of the input \( u \) with an \( n \)-dimensional kernel \( h_n(s_1, \ldots, s_n) \). Series expansions of the form (9) can be shown to exist for a broad class of nonlinear systems\cite{19}. Volterra series, for example, may be considered a Taylor series expansion of the nonlinear functional \( K : [0, t]^p \to R^q \) that maps \( u(t) \) on the past time interval \([0, t]\) to the output \( y(t) \) at time \( t \).

The Laplace transform \( Y(s) \) of the output \( y(t) \) is likewise given by a sum of terms,
\[
Y(s) = \sum_{n=0}^{\infty} Y_n(s) \tag{10}
\]
where each \( Y_n \) is related to a multi-dimensional response function by
\[
Y_n(s) = \frac{1}{(2\pi i)^{k}} \int_{s_1 - i\infty}^{s_1 + i\infty} \cdots \int_{s_n - i\infty}^{s_n + i\infty} Y_n(s - s_1 - \cdots - s_n, s_1, \ldots, s_n)ds_{n-1} \cdots ds_1 \tag{11}
\]
with the responses related to the input by the frequency-domain kernels $H_n$,
\[ Y_n(s_1, \ldots, s_n) = H_n(s_1, \ldots, s_n)U(s_1) \cdots U(s_n) \tag{12} \]
and we have defined the multidimensional Laplace transform
\[ \mathcal{L}(h_n(t_1, \ldots, t_n)) = \int_0^\infty \cdots \int_0^\infty h(t_1, \ldots, t_n) e^{-s_1 t_1 - \cdots - s_n t_n} dt_1 \cdots dt_n \tag{13} \]
From Eq. 11 it is clear that the $n$th kernel $H_n$ represents the $n$th order distortion products that result in inputs at frequencies $s_1, \ldots, s_n$ generating an output response at the frequency $s = s_1 + \cdots s_n$. If we construct a reduced model whose $n$th kernel $\hat{H}_n(s_1, \ldots, s_n)$ matches the original system kernel $H_n(s_1, \ldots, s_n)$, then $\hat{y}$ will approximate $y$ for (at least) terms in the $n$th order. If the nonlinearity of the system is sufficiently weak and/or the input sufficiently small, matching the first few kernels will allow the reduced order system to approximate the full nonlinear system.

4 Bilinear systems analysis

Now we consider nonlinear systems of a special form, those that are bilinear in the state $x$ and the inputs $u$. A $p$-input bilinear system has the form (where we will take $E = I$ in the rest of the paper to simplify notation)
\[ \frac{dx}{dt} = Ax + \sum_{j=1}^p N_j xu_j + Bu \quad y =Cx \tag{14} \]
where $N_j \in \mathbb{R}^{m \times n}$ and $u_j(t)$ denotes the $j$th component of $u(t)$. The terms $N_j xu_j$ are responsible for the nonlinear response of the system. Bilinear systems are interesting because calculation of the kernels in the functional series expansion is fairly simple. As the nonlinear systems of interest in this paper can be embedded in a bilinear system of higher dimensionality, analysis of bilinear systems provides the formal tools needed for analysis of more general nonlinear ones. For the moment, in order to simplify notation, let us restrict the analysis to single-input, single-output systems, and further impose the restriction that $x(0) = 0$. A non-zero initial state will lead to an additional series in what follows. To analyze bilinear systems we construct a functional series representation of the state response $x(t)$.

**Theorem 4.1** The Laplace transform of the $n$th order kernel describing the response $x(t)$ of the SISO bilinear system to input $u(t)$ is given by
\[ H_n(s_1, \ldots, s_n) = C((s_n \cdots s_1 I - A)^{-1} N \cdots N(s_1 I - A)^{-1} B \tag{15} \]

**Proof.** See [17]. \qed

It will be convenient to express the frequency-domain kernels in what is called the regular form [17],
\[ H_n(s_1, \ldots, s_n) = H_n^{\text{reg}}(s_1 + s_2 + \cdots + s_n, s_1 \cdots s_n) \tag{16} \]
so we have
\[ H_n^{\text{reg}}(s_1, \ldots, s_n) = C(s_n I - A)^{-1} N(s_{n-1} I - A)^{-1} N \cdots N(s_1 I - A)^{-1} B \tag{17} \]

5 Bilinear model reduction

Assuming that an appropriate $q$-dimensional subspace has been identified as $R(V)$ with $V \in \mathbb{R}^{m \times q}$, reduction by orthogonal projection proceeds precisely as in the linear case, by first identifying the approximate state $\hat{x} = \hat{V}z$, and then defining the residual $R \equiv A\hat{x} + N\hat{x}u + Bu + \frac{dz}{dt}$ and imposing the Galerkin condition $V^T R = 0$. We obtain a reduced model
\[ \hat{E} \frac{dz}{dt} = \hat{A}z + \hat{N} \hat{z}u + \hat{B}u \quad \hat{y} = \hat{C} \hat{x} \tag{18} \]
where
\[ \hat{A} \equiv V^T AV \quad \hat{N} \equiv V^T NV \quad \hat{B} \equiv V^T B \quad \hat{E} \equiv V^T \hat{E} \quad \hat{C} \equiv CV \tag{19} \]
. The reduced bilinear system is also bilinear, but of dimension $q$, and if $q \ll n$, substantial computational savings can be achieved. Now we address the choice of the matrix $V$.

To see how to extend the rational approximation properties of the Krylov-subspace methods to bilinear systems, consider the second-order regular kernel, given by
\[ H_2^{\text{reg}}(s_1, s_2) = C(s_2 I - A)^{-1} N(s_1 I - A)^{-1} B \tag{20} \]
Similarly, the kernel for the reduced system is
\[ \hat{H}_2^{\text{reg}}(s_1, s_2) = \hat{C}(s_2 I - \hat{A})^{-1} \hat{N}(s_1 I - \hat{A})^{-1} \hat{B} \tag{21} \]
The natural generalization of moment-matching is to require that $\hat{H}(s_1, s_2)$ and $H(s_1, s_2)$ agree to terms in $s_1^q s_2^q$, i.e., that $\hat{H}$ be a multidimensional partial Padé approximation of $H$. The two-dimensional Taylor series expansion of $H_2^{\text{reg}}(s_1, s_2)$ is
\[ H_2^{\text{reg}}(s_1, s_2) = \sum_{k=0}^{\infty} \sum_{\ell=0}^{\infty} C_{\ell k} N^\ell A^{-k} B s_1^\ell s_2^k \tag{22} \]
We call terms of the form $CA^{-\ell} NA^{-k} B s_1^\ell s_2^k$ a multilinear moment. We now show how Krylov-subspace algorithms lead to reduced models that match multimoments.

Suppose we construct a $V_1$ such that $A^{-k} B \in \text{colsp}(V_1)$ for $k = 0 \ldots q_1$, i.e., $R(V_1) \supseteq K_{q_1}(A^{-1} B)$. This is the condition for the first order kernels, i.e., the transfer function of the linearized model, to match up to terms in $s_1^q$ (see Theorems 2.1, 2.2), or in other words, for $A^{-k} B = V_1 A_1^{-k} B_1$, $k \leq q_1$ where $A_1 = V_1^T A V_1$, $B_1 = V_1^T B$. Now construct $V_2$ such that $K_{q_2}(A^{-1} N, V_1) \subset R(V_2)$, and construct $V \in R^{m \times q}$ such that $R(V) \supseteq R(V_1) \cup R(V_2)$. Then we must have
\[ A^{-k} B = V \hat{A}^{-k} \hat{B} \quad k \leq q_1 \tag{23} \]
where now $\hat{A} = V^T AV$, $\hat{B} = V^T B$, because $R(V_1) \subset R(V)$. In addition, if we take any $x \in R(V_1)$, so that $x = V_1 z_1 = V z$ for some $z \in R^q$, we have
\[ A^{-l} N x = V \hat{A}^{-l} \hat{N} V z = V \hat{A}^{-l} \hat{N} z \quad l \leq q_2. \tag{24} \]
The key point is that, by construction, $A^{-k} B \in R(V_1)$ for $k \leq q_1$. Thus
\[ A^{-l} N A^{-k} B = V \hat{A}^{-l} \hat{N} A^{-k} \hat{B} \quad k \leq q_1, l \leq q_2 \tag{25} \]
so we may conclude that
\[ H_2^{\text{reg}}(s_1, s_2) = \hat{H}_2^{\text{reg}}(s_1, s_2) = O(s_1^{q_1} s_2^{q_2}) \tag{26} \]
and that $\hat{A}, \hat{N}, \hat{B}$, etc. is the desired model. This result is easily generalized.
Theorem 5.1 Given
\[
\frac{dx}{dt} = Ax + \sum_{j=1}^{p} N_j x u_j + Bu
\]
where \( A \in \mathbb{R}^{n \times n} \), \( x(t) \in \mathbb{R}^n \), \( B \in \mathbb{R}^{n \times r} \), \( N_j \in \mathbb{R}^{n \times n} \), \( u(t) \in \mathbb{R}^r \), \( C \in \mathbb{R}^r \), suppose
\[
R(V_1) \supset K_{q_1}(A^{-1}, B)
\]
and
\[
R(V_j) \supset \bigcup_{k=1}^{p} K_{q_j}(A^{-1}, N_k V_{j-1}) \quad j > 1
\]
Then if
\[
\hat{A} \equiv V^T A V \quad \hat{N} \equiv V^T N V \quad \hat{B} \equiv V^T B \quad E \equiv V^T V \quad \hat{C} \equiv CV,
\]
and \( H_{n_i}^{(\text{reg})} \) are the order-\( n \) regular kernels of the models \((A, N, B, C)\) and \((\hat{A}, \hat{N}, \hat{B}, \hat{C})\) respectively,
\[
H_n^{(\text{reg})}(x_1, \ldots, x_n) - H_n^{(\text{reg})}(x_1, \ldots, x_n) = O(x_1^n \cdots x_n^n)
\]
for \( n \leq J \).

Proof. Follows directly from Theorems 2.1 and 2.2 via the procedure above. \qed

Clearly, generating models that match many moments of high order kernels could be numerically expensive, particularly for systems with many inputs, and practical implementations will require careful selection of the minimum number of moments to be matched at each nonlinear order, deflation procedures, principal component selection of the spaces \( V_j[1] \), and so forth.

6 Reduction of nonlinear systems

Again consider the system of Equation (4), and for simplicity, assume \( E = I \). Assume \( f(x) \) may be expanded in a multidimensional polynomial series
\[
f(x) = \sum_{k=0}^{\infty} \phi_k(x, \ldots, x)
\]
where each \( \phi_k \) is a \( k \)-multilinear form. For example \( \phi_1(x) \) is linear in the argument \( x \), and may be written as a matrix \( \phi_1(x) = A_1 x \). \( \phi_2(x, y) \) is bilinear in each argument, that is, \( \phi_2(\alpha x + \beta y, z) = \alpha \phi_2(x, z) + \beta \phi_2(y, z) \) and similarly for the second argument. It is possible[8, 20] to form reduced models by using a projection formalism to construct reduced \( \hat{\phi} \) such that
\[
\hat{\phi}(z, \ldots, z) = V^T \hat{\phi}(Vz, \ldots, Vz),
\]
and the Kronecker notation below will considerably simplify the interpretation of the reduced multilinear forms, but for now the analysis will proceed by constructing a bilinear representation of the system \((f(x), B, C)[17] \). Our goal is to guarantee accurate nonlinear representation by including information about the higher-order nonlinear terms explicitly in the reduction process.

6.1 Bilinearization

A concrete representation of the \( \phi_k \) may be obtained by using Kronecker forms. In particular, define
\[
\begin{align*}
\phi^{(1)}(x) &\equiv x, \\
\phi^{(2)}(x) &\equiv x \otimes x, \\
\phi^{(3)}(x) &\equiv x \otimes x \otimes x,
\end{align*}
\]
then the expansion for \( f(x) \) is
\[
f(x) = A_1 x^{(1)} + A_2 x^{(2)} + A_3 x^{(3)} + \cdots
\]
so that
\[
\frac{dx}{dt} = A_1 x^{(1)} + A_2 x^{(2)} + A_3 x^{(3)} + \cdots + Bu
\]
where \( A_k \in \mathbb{R}^{n \times n^k} \). The bilinear model is obtained by defining a new state variable, \( x^\circ \),
\[
x^\circ \equiv \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ \vdots \end{bmatrix}
\]
To do this, note that the time-derivative of a term \( x^{(k)} \) is related to the time-derivative of the original state, \( \dot{x} \), and a term of one order lower, \( x^{(k-1)} \). For example,
\[
\frac{d}{dt} x^{(2)} = \frac{d}{dt} (x^{(1)} \otimes x^{(1)})
\]
\[
= \dot{x} \otimes x^{(1)} + x^{(1)} \otimes \dot{x}
\]
\[
= [A_1 x^{(1)} + A_2 x^{(2)} + \cdots + B u] \otimes x^{(1)} +
\]
\[
\dot{x}^{(1)} \otimes [A_1 x^{(1)} + A_2 x^{(2)} + \cdots + B u]
\]
\[
= [(A_1 \otimes I)(x^{(1)} \otimes x^{(1)}) + \cdots + (B \otimes I)x^{(1)}u +\]
\[
+[I \otimes A_1](x^{(1)} \otimes x^{(1)}) + \cdots + (I \otimes B)x^{(1)}u]
\]
\[
= [(A_1 \otimes I + I \otimes A_1)x^{(2)} + \cdots + (B \otimes I)x^{(1)}u]
\]
\[
= A_2 x^{(2)} + \cdots + B x^{(1)}
\]
where \( A_{21} = (A_1 \otimes I + I \otimes A_1) \) and \( B_{20} = (B \otimes I) \). Continuing this process, we will obtain a bilinear realization \((A^\circ, N^\circ, B^\circ, C^\circ)\)
\[
\frac{dx^\circ}{dt} = A^\circ x^\circ + N^\circ x^\circ u + B^\circ u.
\]
for the original nonlinear system, where
\[
A^\circ = \begin{bmatrix} A_{11} & A_{12} & \cdots \\ A_{21} & A_{22} & A_{23} \\ \vdots & \ddots & \ddots \end{bmatrix}
\]
\[
N^\circ = \begin{bmatrix} 0 & 0 & \cdots \\ B_{20} & 0 & 0 \\ \vdots & \ddots & \ddots \end{bmatrix}
\]
\[
B^\circ = \begin{bmatrix} B \\ \vdots \end{bmatrix}, \quad C^\circ = [C \ 0 \ \cdots]
\]
6.2 Multimoments-matching

The multimoments of the nonlinear system can be expressed in terms of the bilinear system quantities as

\[
m_k^{(1)} = C^\circ(A^\circ)^{-k}B, \quad m_k^{(2)} = C(A^\circ)^{-1}N^\circ(A^\circ)^{-k}B, \ldots
\]  

(46)

Even though the matrix \( A^\circ \) has infinite dimension, due to its special structure, multimoment calculation is feasible. Because \( A^\circ \) is block-upper-triangular, and the matrix \( N^\circ \) is non-zero only on the first lower block sub-diagonal, multimoments \( m_{k_{j_{1}...j_{k}}}^{(r)} \) depend only on the submatrices in the first \( k \times k \) size blocks. In particular, note that the \( k \)th kernel \( H_k \) depends only on the first \( k \) terms in the series expansion, so a bilinear approximate system obtained by dropping terms of order higher than \( k \) in the expansion of \( f(x) \) will agree with the original system in the first \( k \) kernels \( H_k \). It is important to keep the distinction between the order of the kernels and the order of the system in mind, however. For example, we have seen that a general bilinear system will possess kernels \( H_k \) of all orders and thus even if \( f(x) \) can be expressed exactly as, say, a second-order multinomial, powers of the input \( u \) to every order may be significant in the output.

Computing the multimoments of the \( k \)th kernel will involve (1) inversion of the diagonal blocks \( A_{k1} \), and (2) products with the off-diagonal matrices \( A_{kj} \) for \( j > 1 \). Since the second operation essentially involves products with the tensor series terms, terms that are sparse in many applications of interest, we expect the bulk of the numerical effort to be in the first step.

It is not so hard to see that the primary difficulty with the bilinear forms is that the dimension of the state space grows exponentially with the order of nonlinear approximation. Solving linear systems of equations whose size grows exponentially is generally considered difficult. However, it is not actually necessary for the model reduction procedure to compute the exact matrix solutions, only the subspaces needed for projection. The linear equations involved in the proposed paradigm are highly structured, and efficient algorithms can be developed to solve them. For example, consider inverting the second diagonal block in \( A^\circ \). This block has the form \( A \otimes I + I \otimes A \). Solving this equation is the same as solving the Lyapunov equation \( AX + XA^T = C \) for some \( C \). It has recently been shown how to construct low-rank solutions of Lyapunov equations [22] for low-rank right-hand-sides, which are the sort that occur in the model-reduction procedure. The virtue of the low-rank solutions is that if they can be constructed for all the diagonal blocks, then the projection vectors can be represented with a number of degrees of freedom that is \( O(N^2) \), where \( N \) is the total number of multimoments matched and \( N \) the dimension of the original nonlinear system. It is still true that \( Q \) will grow exponentially with the degree of nonlinear approximation order, but this is unavoidable for functional series representations.

7 Examples

One of the difficulties in testing nonlinear reduction schemes is demonstrating that the algorithm genuinely reduces nonlinear complexity, not just a linear piece of the problem. A network with many linear elements can be reduced much more easily than if every element is nonlinear. Ultimately we are interested in studying problems such as RF circuits (mixers, power amplifiers, etc.) under time-varying bias conditions [20], but to illustrate the concepts in this paper, we adapt an example familiar from the linear model reduction literature, the RC line. We introduce strong global nonlinearity by connecting a diode in parallel with each resistor. This example is motivated by a problem in [8]. We drive the circuit with a sine wave at one end and observed the transmitted signal. By measuring distortion, an intrinsically nonlinear phenomenon, in the frequency-domain, we can separate the effects of the linear and nonlinear model contributions more cleanly that by observing time-domain simulations. Note that in our problem nonlinearity and capacitance is distributed throughout the network and so the distortion will vary with frequency in a way that may be difficult to approximate by approaches that do not explicitly consider the nonlinearity in the model-reduction procedure. Harmonic balance was used to calculate the response of the original and reduced system. The original line has thirty RC sections.

Figure 1 shows computed results. An order-8 model was computed based on the bilinear form that included the vectors \( B, A^{-1}B, \ldots, A^{-5}B, NA^{-1}B, A^{-1}NA^{-1}B \). The higher-order distortion terms show some degree of match to the full model. Next, to increase the order of nonlinear approximation, we add in additional Krylov-subspace components, \( A^{-2}NA^{-1}B, NA^{-2}B, A^{-1}NA^{-2}B, A^{-2}NA^{-2}B \) to match more second-order multimoments. The re-
sult is shown in Figure 2. The higher-order distortion products are now almost as well approximated as the linear terms. Note that the linear (first-order) response is essentially that of an order-8 Arnoldi-based model. We consider this empirical evidence that our nonlinear approach should have comparable properties of accuracy and computational complexity as the linear procedures.

Finally, we wish to point out that the line was driven sufficiently hard to produce 2nd-order distortion terms 20dB down from the primary signal, and -40dB 3rd-order terms. This is enough distortion to indicate that the approach presented here is capable of treating nonlinear effects that occur in practical examples.

8 Conclusion

We have presented a projection-based approach to model reduction of nonlinear circuits. By using a bilinearization formalism, we demonstrated reduction approaches that general models that match moments of Volterra-like kernels. It is possible to develop algorithms that are similar in complexity to linear schemes, yet can compute models that exhibit nonlinear, frequency-dependent behavior. We expect that the bilinearization formalism will also be useful in understanding the advantages and limitations of nonlinear reduction approaches based purely on linear information (e.g., [8]). One of the drawbacks of the bilinearization is that, in the general case, it may produce models of higher order than necessary. It is possible to avoid this by working directly with the Taylor series matrices [20], resulting in an improvement in model size and a reduction in algorithmic complexity.

References