Manifold Construction and Parameterization for Nonlinear Manifold-Based Model Reduction

Chenjie Gu and Jaijeet Roychowdhury
{gcj,jr}@eecs.berkeley.edu
EECS Department, University of California, Berkeley

***Abstract*—We present a new manifold construction and parameterization algorithm for model reduction approaches based on projection on manifolds. The new algorithm employs two key ideas: (1) we define an *ideal manifold* for nonlinear model reduction to be the solution of a set of differential equations with the property that the tangent space at any point on the manifold spans the same subspace as the low-order subspace (*e.g.*, Krylov subspace generated by moment-matching techniques) of the linearized system; (2) we propose the concept of *normalized integral curve equations*, which are repeatedly solved to identify an almost-ideal manifold.**

The manifold constructed by our algorithm inherits the important property in [1] that it covers important system responses such as DC and AC responses. It also preserves better local distance metrics on the manifold, thanks to the employment of normalized integral curve equations. To gauge the quality of the resulting manifold, we also derive an error bound of the moments of linearized systems, assuming moment-matching techniques are employed to generate low-order subspaces for linearized systems. The algorithm is also more systematic and generalizable to higher dimensions than the *ad hoc* procedure in [1].

We illustrate the key ideas through a simple 2-D example. We also combine this new manifold construction and parameterization algorithm with maniMOR [1] to generate reduced models for a quadratic nonlinear system and a CMOS circuit. Simulation results are provided, together with comparisons to full models as well as TPWL reduced models [2].

I. Introduction

Dynamical systems are established as models in many disciplines, including electronic circuits, mechanical systems, chemical kinetics, ecosystems, economics, and so forth. Mathematically, they are expressed as differential equations, and various computer programs (such as SPICE [3] for circuits, COPASI [4] for bio-chemical pathways, *etc.*) have been developed to simulate these systems.

A key challenge is that the number of these differential equations can be very large, making simulation extremely slow. For example, a full-SPICE simulation for a circuit with millions of transistors can take days or weeks. As another example, bio-chemical reactions that involve a large number of molecules and reactions also result in a large number of equations.

In this context, model order reduction (MOR) methods become important for simulating large scale dynamical systems. They macromodel a large system into a much smaller one which preserves important input-output responses of the original system. This ambitious objective makes MOR an extremely hard problem, but also an extremely useful technique to simulate large scale systems that are otherwise not affordable to simulate.

MOR for linear systems (especially LTI systems) has witnessed many advances [5]–[8], thanks to elegant linear system theories and wide applicability to real problems, such as RLC networks (in transmission lines and power grids), LTI systems derived from discretization of partial differential equations, *etc.*.

In contrast, MOR for nonlinear systems is much more difficult. One class of methods is based on linearization of nonlinear systems around steady state (DC or periodic) solutions. For example, linear reduced models [9], bilinear reduced models [10] and polynomial reduced models [11]–[13] are based on linearization around the DC operating point of the circuit; linear time-varying reduced models [12], [14] are based on linearization around the periodic steady state of the circuit. Because of the linearization, they are only valid for small signal inputs superimposed on the steady state solution. They also fail to capture many important behaviors of the original nonlinear systems, such as high-order nonlinear harmonic distortions, nonlinear DC responses, oscillatory behaviors, *etc.*.

To address shortcomings of linearization, another class of nonlinear MOR methods, known as trajectory-based methods [2], [15], [16] has been proposed. They generally employ a two-step procedure: (1) training the system and sampling trajectories; (2) building up a low-order subspace, and projecting the original system onto that subspace to generate the reduced order model. Several important issues regarding efficiency, stability and passivity have been studied [17]–[20], and considerable reduction on some practical examples has been reported. However, it still remains unclear how important dynamics, such as DC and AC responses, are preserved in the reduced model.

Recently, a promising variant of trajectory-based methods, maniMOR [1], has been proposed. It consists of three steps: (1) construction of the manifold; (2) parameterization of the manifold; (3) projection of the state space to the manifold. Because it projects the original state space onto a nonlinear manifold, it achieves greater reduction than linear projection. Moreover, it has the important property that it explicitly tries to replicate nonlinear DC responses across a range of inputs, which is achieved by constructing a manifold on which the DC and AC responses lie. Therefore, the key steps in maniMOR are to construct a “good” manifold, and parameterize it. However, the manifold construction and parameterization proposed in [1] are more *ad hoc* than systematic, and potentially computationally expensive, as detailed in the following sections.

In this paper, we present a new method to systematically construct the manifold and to parameterize it. The key insight of our manifold construction scheme is to make sure that the *tangent space* at any point on the manifold spans the same subspace as the low-order subspace (such as Krylov subspace) of the linearized system at that point. Following this key idea, we define an *ideal manifold* as the solution of a set of differential equations, each of which corresponds to *integral curves* on the manifold.

However, we show that such an ideal manifold may not exist, and that only an “optimal” manifold can be constructed, in the sense that the local tangent space at any point on the manifold spans the low-order subspace of the linearized system as much as possible. As a result, we propose a heuristic algorithm to efficiently identify an almost-ideal manifold by finding integral curves, the concept of which is drawn from differential geometry.

The manifold constructed in our algorithm inherits the important property of covering of DC and AC responses. This is achieved by tracing integral curves corresponding to DC and AC responses, which is more efficient and elegant than performing a series of brute-force DC and AC analyses to the system, as is done in maniMOR [1]. Since integral curves are defined by a set of differential equations, various numerical integration methods used in transient simulation can be readily applied.

Besides capturing important responses correctly, another important guideline for constructing and parameterizing the manifold is to preserve local distance metrics, *i.e.*, the local distance between two points on the manifold should be the same as that in the original state space. To preserve local distance metrics, we construct *normalized integral curve equations* for integral curves, and prove that the solutions of normalized integral curve equations overlap with those of regular integral curve equations. Thus, integration of normalized integral curve equations naturally gives an almost perfect parameterization on the manifold that preserves local distance metrics, rather than the *ad hoc* approach in maniMOR [1].

Since the conditions for an ideal manifold are usually not satisfied, a metric to define the “quality” of the manifold is desired. Assuming moment-matching methods are used to generate low-order subspaces at each point on the manifold, we derive an error bound for the moments of impulse responses of linearized systems of the reduced model. This error bound is then used as a metric to assess the “quality” of the manifold.

We verify our manifold construction and parameterization method on a simple example where all the equations for the manifold can be written analytically. We then combine the manifold construction and parameterization method in this paper with maniMOR, and apply it to reduce two systems, including a quadratic nonlinear system and a CMOS circuit. Simulation results are compared with full models as well as TPWL reduced order models.

The remainder of the paper is organized as follows. In Section II, we review the nonlinear projection framework for nonlinear model order reduction, the concept of a manifold, and the manifold construction and parameterization scheme in maniMOR. In Section III, we describe the manifold construction and parameterization method by finding integral curves. In Section IV, we show two examples of application of our method, and compare simulation results to full models and TPWL reduced order models.

978-1-4244-5767-0/10/$26.00 2010 IEEE

II. Background

In this section, we first summarize the nonlinear projection framework, which is the core of many existing MOR methods. Then we briefly review the concepts about manifold as well as the manifold construction and parameterization in maniMOR [1].

A. Nonlinear Projection Framework for Nonlinear Model Reduction

Consider a system described by ordinary differential equations[1]

$$\frac{d}{dt}x + f(x) + Bu(t) = 0, \tag{1}$$

where $x \in \mathbb{R}^n$ are state variables[2], and $u(t)$ are the inputs.

The nonlinear projection framework for reducing (1) consists of three main steps: [1]

1) Construction of the nonlinear manifold
2) Parameterization of the manifold
3) Projection from original state space to the manifold

Given this nonlinear projection framework, the question is how to construct a "good" manifold and parameterize it appropriately.

B. Manifold and its Parameterization

In this subsection, we introduce some concepts that are related to this work from differential geometry [21], in a rather intuitive manner. Readers are recommended to refer to textbooks on this subject for more rigorous definitions and details.

Roughly speaking, manifolds are locally vector spaces and globally curved surfaces. The tangent space at a point on the manifold is a linear space which locally approximates the manifold. The dimension of the tangent space is the same as the dimension of the manifold.

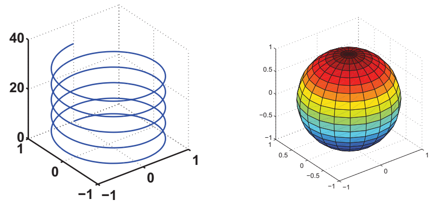
For example, Fig. 1(a) shows a helix, which is defined by

$$x = cos(t), \qquad y = sin(t), \qquad z = t. \tag{2}$$

It is locally a 1-D line, and globally a 1-D manifold (curve). For another example, Fig. 1(b) shows a sphere, which is defined by

$$x = \sin\theta + \cos\phi, \quad y = \sin\theta + \sin\phi, \quad z = \cos\theta. \tag{3}$$

It is locally a 2-D plane, and globally a 2-D manifold (sphere).



(a) Helix. (b) Sphere.

Fig. 1. Examples of 1-D manifold and 2-D manifold.

To be formal, a subset $M \subset \mathbb{R}^n$ is called a *smooth manifold of dimension m* if for each $x \in M$ there is a neighborhood $W \bigcap M$ ($W \subset \mathbb{R}^n$), that is diffeomorphic to an open set $U \subset \mathbb{R}^m$. [21]

Accordingly, a diffeomorphism ψ from U to $W \bigcap M$ is called a *parameterization*, and and its inverse ψ^{-1} is called a *system of coordinates* on $W \bigcap M$. [21] Therefore, suppose $x \in \mathbb{R}^n$ is on the manifold, and $z \in \mathbb{R}^m$ is the global coordinates on the manifold, then $z = \psi^{-1}(x)$ and $x = \psi(z)$. Note that $\psi(\cdot)$ is exactly the $v(\cdot)$ function in maniMOR.

With a parameterization $\psi : U \subset \mathbb{R}^m \mapsto M \subset \mathbb{R}^n$ of a neighborhood $\psi(U)$ of x on manifold M, the *tangent space* to M at $x = \psi(z)$ (denoted by TM_x) is defined to be the image of the map $D\psi_u : \mathbb{R}^m \mapsto \mathbb{R}^n$, where $D\psi_u$ is defined as

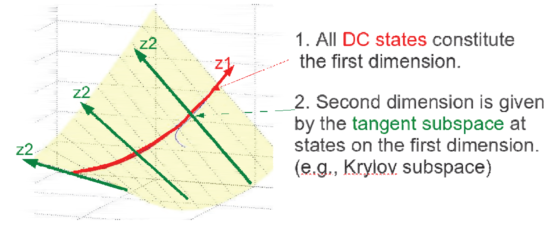
$$D\psi_u(h) = \lim_{t\to 0} \frac{\psi(u+th) - \psi(u)}{t}. \tag{4}$$

In other words, a manifold can be defined as a pair (X, TM_X) where X are the points on the manifold and TM_X are the tangent spaces for all $x \in X$. This definition also validates the usage of piecewise linear approximation to the manifold in maniMOR [1].

C. Manifold Construction and Parameterization in maniMOR [1]

ManiMOR [1] uses two insights to identify the manifold. (1) After the transient behavior is over, the state variable should converge to its DC operating point (DC steady state), if the circuit is not oscillatory or chaotic; (2) Operating at each DC steady state, the local linear subspace around the DC solution can be approximated by a low-order Krylov subspace of the LTI system linearized at that DC operating point. Based on these two insights, maniMOR first performs a series of DC analyses to the circuit, and build up a 1-D manifold by connecting all the DC solutions. Then at each point on the 1-D manifold, the nonlinear system is linearized and Krylov subspace for the linearized system is generated. Finally, the manifold is obtained by stitching together all these Krylov subspaces.

For example, Fig. 2 shows a 2-D manifold for a nonlinear system, where the red curve is composed of all the DC solutions, the green arrows represent the second Krylov basis vector, and the yellow surface is obtained by connecting together all the local tangent spaces.



1. All DC states constitute the first dimension.

2. Second dimension is given by the tangent subspace at states on the first dimension. (e.g., Krylov subspace)

Fig. 2. Illustration of manifold construction in maniMOR.

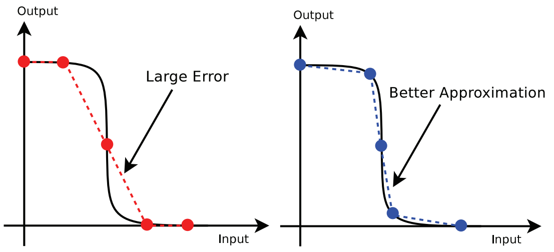
Based on this manifold, maniMOR parameterizes the manifold by trying to preserve the local distances in the original state space. Given two the coordinates of points $x_i, x_j \in \mathbb{R}^n$, and the coordinate of x_i on the manifold (z_i), maniMOR calculates the coordinate of x_j on the manifold to be z_j such that

$$||z_i - z_j||_2 = ||x_i - x_j||_2. \tag{5}$$

There are two main issues associated with the manifold construction and parameterization in maniMOR.

Firstly, the manifold is nonlinear only along the first dimension, which captures system responses mainly around DC operating points. The generalization to make higher dimensions of the manifold to be nonlinear is not straightforward.

Secondly, when computing the DC manifold, the step size for DC sweep analysis is hard to set since two close DC inputs can lead to two DC solutions far apart. In this case, the parameterization procedure can incur large errors. For example, if we perform a DC sweep analysis to an inverter, and want to approximate the DC curve by five points, it is undesirable to have a uniform sampling of the input voltage, as shown in Fig. 3(a). Instead, we want to automatically choose the points equally spaced on the curve, as shown in Fig. 3(b). It is also obvious that the parameterization in the latter case better preserves the distance metric.



(a) Uniform sampling of the input. (b) Uniform sampling on the DC curve.

Fig. 3. DC manifold of a single inverter.

III. Manifold Construction and Parameterization by Finding Integral Curves

In this section, we first give a brief introduction to the concept of integral curve. We then show that the DC manifold constructed in maniMOR can in fact be solved by finding an integral curve. By normalizing the RHS of differential equations for the integral curve, we obtain *normalized integral curve equations*, which are more appropriate for the manifold parameterization application.

By generalizing the idea for finding the DC manifold, we define an *ideal manifold* by the solution of a set of differential equations.

[1] A more general form is differential algebraic equations $\frac{d}{dt}q(x) + f(x) + Bu(t) = 0$. However, for simplicity, we consider only ordinary differential equations in this paper.

[2] Specifically, x represent node voltages and branch currents in modified nodal analysis equations for the circuit.

This manifold has the property that *at each point on the manifold, the tangent space is defined by the low-order subspace of the linearized system at that point*. As an example, we use Krylov-subspace based moment-matching methods to solve for the low-order subspace for each linearized system. However, other subspaces may also be used for the tangent space.

Unfortunately, we show that such an ideal manifold may not exist. Therefore, instead of finding this ideal manifold, we give a heuristic method to find an almost-ideal manifold, which can be solved efficiently by repeatedly finding integral curves. Since the tangent space of this almost-ideal manifold does not span the right Krylov subspace of linearized systems, the moments of the reduced systems linearized at the same points on the manifold are not matched. To assess how well the moments are matched, we also provide an error bound for the moments of the reduced linearized systems.

A. Integral Curve

Integral curve is an important concept in differential geometry. To define an integral curve, a vector field V on the manifold M must be defined first. A *vector field* V essentially defines a vector $v(x) \in TM_x$ at each point $x \in M$. An *integral curve* of a vector field V is then the curve $\gamma \equiv x(t)$ on M, such that

$$\frac{dx}{dt} = v(x). \tag{6}$$

Intuitively, the integral curve can be viewed as the path of a point mass in the state space, and on any point on this path, the velocity of the point mass is determined by the vector field V, as shown in Fig. 4.

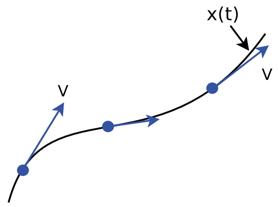


Fig. 4. Integral curve.

Therefore, the solution to (6) gives a construction of a curve consisting of points on $x(t)$ and a parameterization of the curve γ by parameter t.

B. DC Manifold

In maniMOR [1], the DC manifold is defined by the DC equations of the circuit

$$f(x) + Bu = 0. \tag{7}$$

Differentiating (7) with respect to the input u (assuming u is a single input), we obtain

$$\frac{df}{dx}\frac{dx}{du} + B = G(x)\frac{dx}{du} + B = 0. \tag{8}$$

where $G(x) = \frac{df}{dx}$ is the Jacobian matrix of $f(x)$.

Thus, if $G(x)$ is non-singular, we have

$$\frac{d}{du}x = -[G(x)]^{-1}B. \tag{9}$$

which defines an integral curve $x(u)$, parameterized by the input u. We call (9) the *regular integral curve equation*. Note that the RHS defines a vector whose direction is the same as the first Krylov basis vector for systems linearized at x. With (9), theorem 3.1 follows:

Theorem 3.1: Suppose $[G(x)]^{-1}B$ satisfies the Lipschitz condition. Then there exists a unique solution to (9), $x(u)$, which corresponds to DC solutions to (7), if the initial condition is chosen to be a DC solution x_0 when $u = u_0$.

Proof: According to the existence and uniqueness theorem for ordinary differential equations [21], since $[G(x)]^{-1}B$ satisfies the Lipschitz condition, (9) has a unique solution.

Depending on the initial condition, the solutions to (9) have the general form of

$$f(x) + Bu + \vec{c} = 0, \tag{10}$$

where $\vec{c}$ is a constant vector.

Since the initial condition is x_0, which is the DC solution when $u = u_0$, we have

$$f(x_0) + Bu_0 + \vec{c} = \vec{c} = 0. \tag{11}$$

Therefore, $x(u)$ satisfies DC equations (7). ■

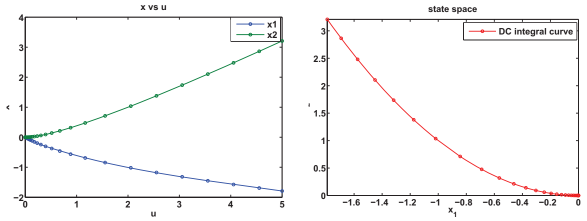
As an illustration, we consider the following 2-D nonlinear system

$$\frac{d}{dt}x_1 = -x_1 + x_2 - u(t), \quad \frac{d}{dt}x_2 = x_1^2 - x_2. \tag{12}$$

Therefore, the equation for the DC integral curve is

$$\frac{dx}{du} = -[G(x)]^{-1}B = \frac{1}{2x_1 - 1}\begin{bmatrix} 1 \\ 2x_1 \end{bmatrix}. \tag{13}$$

In order to solve for the DC curve, we set the initial condition to be a DC solution. Without loss of generality, we choose the DC solution when $u = 0$, which is $x_1 = 0, x_2 = 0$. The DC manifold solved by integrating the integral curve equation is shown in Fig. 5. Fig. 5(a) shows the trajectories of x_1 and x_2 with respect to input u, and Fig. 5(b) shows the DC curve in the state space. It is easily verified that all the points on this integral curve are indeed the DC operating points.



(a) Trajectories with respect to u. (b) DC curve in the state space.

Fig. 5. Constructing the DC manifold by tracing the integral curve.

C. Normalized Integral Curve Equation

There is one major drawback in the above naïve manifold construction and parameterization method: the local distance metric in the original state space is not preserved. Since u is the parameter that parameterizes the DC manifold, we require $|du| = ||dx||_2$, in order to preserve the local Euclidean distance (defined by 2-norm). However, $||[G(x)]^{-1}B||_2 = 1$ is not satisfied for almost all the cases.

Moreover, in Krylov subspace methods, usually only the normalized basis vectors are calculated for each linearized system, in order to retain the numerical stability in generating the projection matrix. This is especially the case when Krylov subspace algorithms are called as a black-box sub-routine. Therefore, the RHS in (9) may not be available.

To circumvent these problems, we build up a *normalized integral curve equation*

$$\frac{dx}{du} = \frac{[G(x)]^{-1}B}{||[G(x)]^{-1}B||_2} \tag{14}$$

where the RHS is normalized (divided by the 2-norm of the RHS). In (14), the 2-norm of the RHS is constantly 1, and therefore, the local distance is preserved on the manifold parameterized by u.

It still remains to show that the solutions to (14) cover the same state space as the solutions to (9), *i.e.*, the DC manifold remains the same, and only parameterization changes. The following theorem gives a way to prove this property:

Theorem 3.2: Suppose $x(t)$ and $\hat{x}(\tau)$ are the solutions to

$$\frac{d}{dt}x(t) = g(x(t)) \tag{15}$$

and

$$\frac{d}{d\tau}\hat{x}(\tau) = \sigma'(\tau)g(\hat{x}(\tau)), \tag{16}$$

respectively, where $t = \sigma(\tau)$ is a function of τ. Then $x(t)$ and $\hat{x}(t)$ span the same state space, *i.e.*$\forall t, \exists \hat{t}$, such that $\hat{x}(\hat{t}) = x(t)$.

Proof: Since $t = \sigma(\tau)$, we have

$$dt = \sigma'(\tau)d\tau \tag{17}$$

Define

$$\hat{x}(\tau) \equiv x(t) = \hat{x}(\sigma(t)). \tag{18}$$

Therefore,

$$\frac{d}{d\tau}\hat{x}(\tau) = \frac{d\hat{x}(\tau)}{dt}\frac{dt}{d\tau} = g(x(t))\sigma'(\tau) = g(\hat{x}(\tau))\sigma'(\tau), \tag{19}$$

i.e.,

$$\frac{d}{d\tau}\hat{x}(\tau) = \sigma'(\tau)g(\hat{x}(\tau)). \tag{20}$$

According to the existence and uniqueness theorem for ordinary differential equations, $x(t)$ and $\hat{x}(\tau)$ are the unique solutions to (15) and (20), respectively.

So $x(t)$ and $\hat{x}(t)$ span the same state space. ■

Based on theorem 3.2, we can prove the following corollary,

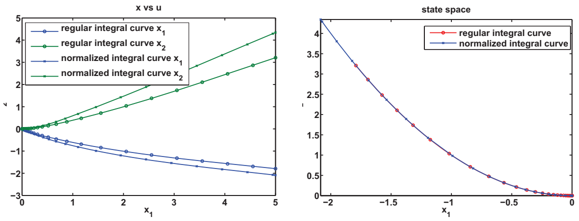
Corollary 3.3: The solutions to the normalized integral curve equation (14) span the same space as the solutions to the regular integral curve equation (9).

Proof: Suppose the solutions to (9) and (14) are is $x(u)$ and $\hat{x}(\hat{u})$, respectively. Let

$$u = \sigma(\hat{u}) = \int_0^{\hat{u}} \frac{1}{||[G(\hat{x}(\mu))]^{-1}B||_2} d\mu. \tag{21}$$

By theorem 3.2, $x(t)$ and $\hat{x}(\hat{u})$ span the same state space. ■

To validate the above theorem and corollary, we perform the integration of (14) for the same 2-D example, and compare the results to those of (9), as shown in Fig. 6. Fig. 6(a) plots trajectories of x_1 and x_2 for both regular integral curve equations and normalized integral curve equations, and we observe that two integral curves give different parameterizations. Fig. 6(b) plots the integral curves in the state space, and it is clear that two integral curves overlap perfectly in the state space.



(a) Trajectories with respect to u. (b) DC curve in the state space.

Fig. 6. Comparison of regular integral curve and normalized integral curve.

The normalized integral curve equation, together with theorem 3.2 and corollary 3.3, are of great value in two aspects:

Firstly, integrating normalized integral curve equations has a much better numerical behavior than integrating regular integral curve equations. For the simplest scalar equation $\dot{x} = x$, we know that the solution $x(t)$ grows exponentially – which is a bad news for parameterization use since as t increases, $|x(t+\Delta t) - x(t)|$ also increases exponentially. However, the solution $\hat{x}(t)$ to the normalized integral curve equation increases only at a constant rate, and thus the points on the integral curve are well spaced.

Secondly, it gives rise to generalization of this integral curve based manifold construction and parameterization method to higher dimensions. Viewing Krylov subspace methods as a black box, the input is an LTI system, and the output is a projection matrix, whose columns are ortho-normalized basis vectors. Therefore, it is natural to explore the manifold by integrating along directions other than the DC direction.

D. Ideal Nonlinear Manifold

Inspired by the idea of parameterizing the DC manifold by finding an integral curve, we further propose an *ideal nonlinear manifold* defined by a set of differential equations.

The ideal nonlinear manifold is derived by the following intuition: For a nonlinear system, if the current state is x^*, then the local behavior of the nonlinear system is determined by the linearized system (at x^*), and the behavior of the linearized system can be efficiently approximated by its reduced order model, by projecting the original state space to a low-order subspace (such as Krylov subspace). On the other hand, by definition, the manifold around x^* (which is on the manifold) is well-approximated by the tangent space at x^*. Therefore, it is desirable for the tangent space at x^* to span the low-order subspace generated for the linearized system (at x^*).

For example, if we use Arnoldi algorithm to generate the q-th order Krylov subspace $V(x) = [v_1(x), \cdots, v_q(x)]$ for linearized systems, it is desirable for the tangent space at x to span the subspace defined by $V(x)$. Let $x \in \mathbb{R}^n$ be the state variable in the original state space, and $z \in \mathbb{R}^q$ be the state variable on the parameterized manifold, then the tangent space is defined by the span of $\frac{\partial x}{\partial z}$. Therefore, we have

$$\frac{\partial x}{\partial z_1} = v_1(x), \qquad \frac{\partial x}{\partial z_2} = v_2(x), \qquad \cdots, \qquad \frac{\partial x}{\partial z_q} = v_q(x). \tag{22}$$

Unfortunately, these differential equations are *over-determined*, and the solution may not exist. (In (22), the number of unknown variables x is n, and the number of equations is $n \times q$.) As an example, we discuss the existence of the 2-dimensional ideal manifold, which we call *AC manifold*.

E. AC Manifold

As just shown, the equations defining the AC manifold are

$$\frac{\partial x}{\partial z_1} = v_1(x), \qquad \frac{\partial x}{\partial z_2} = v_2(x). \tag{23}$$

Suppose the DC manifold is already constructed and parameterized by integrating the first equation. The AC manifold is constructed and parameterized starting from this DC manifold, *i.e.*, when integrating the second equation in (23), the initial condition is set to the points on the DC manifold.

As an example, assume Krylov subspace $\left[[G(x)]^{-1}B, [G(x)]^{-2}B\right]$ is used as tangent spaces. Then the AC manifold is defined by

$$\frac{\partial x}{\partial z_1} = w_1(x), \qquad \frac{\partial x}{\partial z_2} = w_2(x), \tag{24}$$

where $w_1(x) = [G(x)]^{-1}B$ and $w_2(x) = [G(x)]^{-2}B$.

As shown in Section III-C, the RHS should be normalized to obtain better parameterization of the manifold. Furthermore, we ortho-normalize $W(x) = [w_1(x), w_2(x)]$ to obtain $V(x) = [v_1(x), v_2(x)]$ so that the output of Arnoldi algorithm can be directly used.

Therefore, for the same 2-D example in Section III-B, the manifold identified by solving (23) is plotted in Fig. 7, where the red curve is the DC manifold, the blue curves are AC integral curves integrated using different DC operating points as initial conditions. In Fig. 7, the region plotted correspond to $z_1 \in [0, 2]$ and $z_2 \in [-2, 1]$.

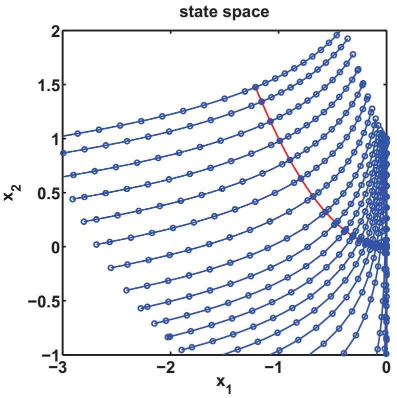
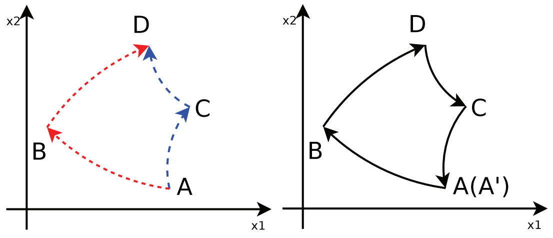


Fig. 7. AC manifold.

However, is this manifold an ideal manifold? To examine this question, we present a test (necessary condition) for ideal manifolds.

In an ideal manifold, there are at least two paths to integrate from one point to another point on the manifold. For example, Fig. 8(a) shows two paths from point A to point D: $A \rightarrow B \rightarrow D$ and $A \rightarrow C \rightarrow D$, where A, B, C, D have coordinates on the manifold being (z_1, z_2), $(z_1 + h, z_2)$, $(z_1, z_2 + h)$, $(z_1 + h, z_2 + h)$, respectively. ($A \rightarrow B$ and $C \rightarrow D$ are on DC integral curves; $A \rightarrow C$ and $B \rightarrow D$ are on AC integral curves.) In other words, no matter what path is used for integration, the same point should be reached.



(a) Two paths from A to D. (b) Test for ideal manifold.

Fig. 8. Test for ideal manifold.

Alternatively, this can be restated as follows: if we parameterize the manifold along the path $A \rightarrow B \rightarrow D \rightarrow C \rightarrow A'$, A and A' should be identical, as illustrated in Fig. 8(b). This serves as a simple test criterion for ideal manifolds.

Since this is generally not the case, we conclude that the ideal manifold may not exist for most cases. (The proof is omitted due to page constraints.)

F. Manifold Construction by Finding Integral Curves

Since an ideal manifold may not exist, we present an algorithm to build an almost-ideal manifold, as shown in algorithm 1. This algorithm starts with an DC operating point in the state space. Using this DC solution as the initial condition, the DC manifold is calculated by integrating DC integral curve equations. Using the DC manifold as initial conditions, the AC manifold is then calculated by integrating AC integral curve equations. Accordingly, using the $(i-1)$-th manifold obtained from previous iterations as initial conditions, the i-th manifold is calculated by integrating the i-th integral curve equations. Finally, a set of points X on the manifold and their parameterizations Z are exported to maniMOR to generate the reduced order model.

Algorithm 1 Manifold Construction by Finding Integral Curves

```
Given the region to be parameterized $(z_{i,min}, z_{i,max}), i \in [1,q]$;
Let $x_0(0,\cdots,0) = x_{DC}$, where $x_{DC}$ is the DC solution when $u = 0$;
$X \leftarrow \{x_0\}$, $Z \leftarrow (0,\cdots,0)$;
for $i = 1$ to $q$ do
    for all $x \in X$ do
        Integrate the integral curve equation
            $\frac{\partial x}{\partial z_i} = v_i(x)$        (25)
        with initial condition $x$;
        $X \leftarrow \{x(z)\}$, $Z \leftarrow z$;
    end for
end for
Output $X$ as the set of points on the manifold;
Output $Z$ as the parameterization of the manifold for each point $x \in X$.
```

Notice that this manifold does not satisfy the condition for ideal manifolds, and therefore we need to derive the tangent space at each point on the manifold, rather than using the low-order Krylov subspace for the linearized system.

For example, consider the AC manifold where two AC integral curves are integrated using two close initial conditions on the DC manifold $x(z_{10},0)$ and $x(z_{10}+\Delta z_1,0)$. By integrating AC integral equations, we obtain two points $x(z_{10},z_2)$ and $x(z_{10}+\Delta z_1,z_2)$. Therefore, the basis vector along the first dimension is

$$v_1 = \lim_{\Delta z_1 \to 0} \frac{x(z_{10}+\Delta z_1,z_2) - x(z_{10},z_2)}{\Delta z_1} = \frac{\partial x(z_1,z_2)}{\partial z_1}. \quad (26)$$

According to the chain rule, we have

$$v_1(z_1,z_2) = \frac{\partial x(z_1,z_2)}{\partial z_1} = \frac{\partial x}{\partial x_0}\frac{\partial x_0}{\partial z_1}. \quad (27)$$

where $\frac{\partial x}{\partial x_0}$ is the sensitivity of the state transition function $\Phi(z_2;x_0,0)$ with respect to the initial condition $x_0 = x(z_{10},0)$[3] and $\frac{\partial x_0}{\partial z_1}$ is the first basis vector along the DC manifold at x_0.

This notion is easily generalized into higher-dimensions. For example, if the dimensionality of the manifold q is 3, assume at any point x on the 3-D manifold but not on the 2-D AC manifold, the basis vector for the tangent space is $[v_1,v_2,v_3]$, where v_3 is the third Krylov basis for the linearized system at that point. Accordingly $v_i,(i=1,2)$ is calculated by

$$v_i = \lim_{\Delta z_i \to 0} \frac{x(z_i+\Delta z_i) - x(z_i)}{\Delta z_i} = \frac{\partial x}{\partial x_0}\frac{\partial x_0}{\partial z_i} \quad (28)$$

where $\frac{\partial x}{\partial x_0}$ is calculated by performing a transient sensitivity analysis, and $\frac{\partial x_0}{\partial z_i}$ is already calculated in previous iterations.

Intuitively speaking, this heuristic is reasonable, since when the state variable is far from its DC solution, the fast dynamics are dominant over slow dynamics. Therefore, it is more desirable for the tangent space to span the basis vectors corresponding to fast dynamics (*e.g.*, which correspond to last few Krylov basis vectors). Indeed, we may reasonably assume that the variables corresponding to slow dynamics do not change.

When the state variables are driven close to the operating point, then the slow dynamics dominate and is also well-modeled by the manifold. These facts make the manifold constructed in algorithm 1 a reasonable one to project onto.

G. Error Bound for the Moments

Since the basis for the tangent space $V = [v_1,\cdots,v_q]$ do not span the low-order Krylov subspace generated by Arnoldi algorithm, not all the moments are matched to the original linearized system.

Since the moments of the linearized system are

$$m_{i-1} = (V^T G V)^{-i} V^T B, \quad (29)$$

which is equivalent to the following iterative definition

$$m_0 = (V^T G V)^{-1} V^T B, \quad m_i = (V^T G V)^{-1} m_{i-1}, \quad \forall i \geq 1. \quad (30)$$

Denote $A = V^T G V$, and $b = V^T B$, then the moments are calculated by a series of $Ax = b$ problems:

$$A m_0 = b, \quad A m_i = m_{i-1}, \quad \forall i \geq 1 \quad (31)$$

Assuming the projection matrix constructed in algorithm 1 is $\hat{V} = V + \Delta V$, then we obtain a perturbed problem

$$(A + \Delta A)x = b + \Delta b, \quad (32)$$

where $\Delta A = \Delta V^T G V + V^T G \Delta V + \Delta V^T G \Delta V$ and $\Delta b = \Delta V^T B$.

On the other hand, for the perturbed system (32), an upper bound [22] for the relative error $\frac{||\Delta x||}{||x||}$ is

$$\frac{||\Delta x||}{||x||} \leq \varepsilon \kappa(A) \left(\frac{||b||}{||A|| \cdot ||x||} + \frac{||y||}{||x||} \right) \quad (33)$$

where $||\Delta A|| \leq \varepsilon ||A||$ and $||\Delta b \leq \varepsilon ||b||$. Therefore, the error bounds for all the moments can be calculated iteratively, and can be used to assess how "good" the manifold is.

IV. Validation

In this section, we replace the manifold construction and parameterization steps in maniMOR [1] with our method, and apply the resulting new maniMOR method to two examples. We validate our approach by comparing the simulation results using this new maniMOR model against the full model and TPWL model.

A. Illustrative 3-D Nonlinear System

We consider an illustrative 3-D nonlinear system

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -10 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ x_1^2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) \quad (34)$$

which has a simple quadratic nonlinearity.

We apply our method to construct and parameterize the manifold, as shown in Fig. 9. In Fig. 9, the red curve is the DC manifold, the blue curves constitute the AC manifold. The black trajectory is the trajectory of a transient simulation of (34) – it lies almost on the manifold we construct.

Using the manifold, we apply maniMOR algorithm to generate a size 2 model for this system, and perform transient simulations. Fig. 10(a) and Fig. 10(b) show simulation results when the system is excited by a multiple-step function $u(t)$ that jump among several DC values. It is observed that maniMOR model tracks the trajectory of the full model better than TPWL. The maximum absolute mean square error of maniMOR model and TPWL model, compared to full model, are 0.0534 and 0.0627, respectively. Although the TPWL model only lead to an error that is slightly larger than maniMOR, its trajectory fails to converge to the right DC operating point, which makes the model unacceptable.

The largest error of maniMOR model happens near time $t = 5$, as shown in Fig. 10(b). This is not desired in the reduced order model – it predicts wrong dynamics of the system. However, although maniMOR captures wrong dynamics of the system in some region due to the reduction, its robustness is also exhibited. After the fast transient behavior is over, the trajectory matches that of the full model, and finally converges to the correct DC solution – the long term DC behavior is correct in maniMOR.

Similar results are observed in Fig. 10(c) and Fig. 10(d), where we apply a two tone sinusoidal input $u(t) = 2.5 + sin(0.1\pi t) + cos(0.4\pi t)$.

[3] The routine to calculate this sensitivity is commonly used in shooting methods which are available in RF simulators.

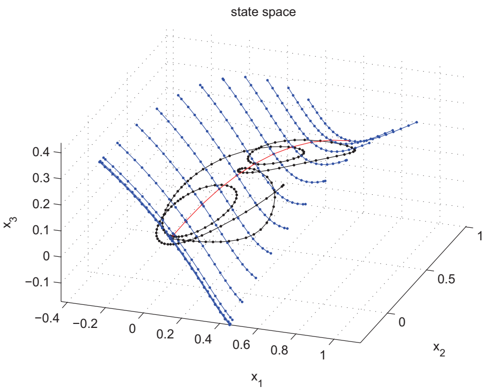
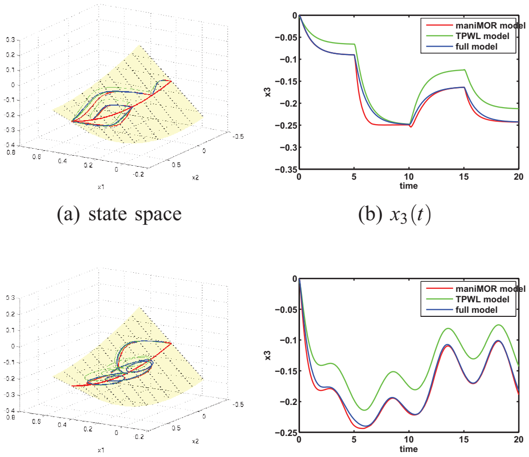


Fig. 9. 2-D manifold for (34).



(a) state space (b) $x_3(t)$

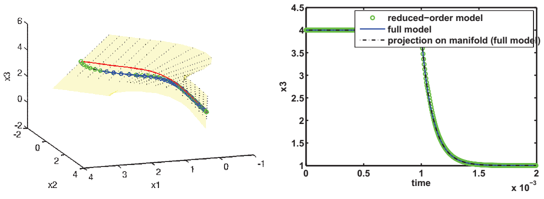
(c) state space (d) $x_3(t)$

Fig. 10. Comparison of maniMOR and TPWL model. (multiple-step input and sinusoidal input) Red, green and blue trajectories represent simulation results of maniMOR, TPWL and full model, respectively.

B. CMOS Ring Mixer

The second example, a simple CMOS ring mixer [23] is highly nonlinear – this can also be seen in Fig. 11(a) that the manifold we constructed is quite twisted.

Again, maniMOR is applied to generate the reduced model. To inspect how well the reduced model captures dynamics and nonlinearities, we apply a step input, and simulate the circuit using full model and maniMOR model. The simulation results are shown in Fig. 11, where we see that the trajectory goes out of the manifold, and finally converges back to another DC operating point. The maximum absolute and relative mean square error in this case is 0.1293 and 0.0492, respectively.



(a) state space (b) $x_3(t)$

Fig. 11. Simulation of maniMOR and full model for the CMOS ring mixer.

V. Conclusion

In this paper, we have presented a new manifold construction and parameterization procedure for nonlinear model order reduction based on projection on manifolds. The manifold we construct inherits good properties of the manifold in maniMOR [1] such as covering DC and AC responses. It also preserves better local distance on the manifold by parameterizing the manifold by integrating normalized integral curve equations, which is a modified version of regular integral curve equations. To gauge the quality of the manifold, we also derive an error bound of the moments for linearized systems when moment-matching MOR methods are used to reduce each linearized system. We have combined this method with maniMOR approach, and applied it to several examples. The resulting reduced models are validated by comparing to full models and TPWL reduced models.

References

[1] C. Gu and J. Roychowdhury. ManiMOR: Model Reduction via Projection onto Nonlinear Manifolds, with Applications to Analog Circuits and Biochemical Systems. In Computer-Aided Design, 2008. ICCAD 2008. IEEE/ACM International Conference on, pages 85–92, 10-13 Nov. 2008.

[2] M. Rewienski and J. White. A Trajectory Piecewise-Linear Approach to Model Order Reduction and Fast Simulation of Nonlinear Circuits and Micromachined Devices. IEEE Transactions on Computer-Aided Design, 22(2), February 2003.

[3] L.W. Nagel. SPICE2: a Computer Program to Simulate Semiconductor Circuits. PhD thesis, EECS department, University of California, Berkeley, Electronics Research Laboratory, 1975. Memorandum no. ERL-M520.

[4] S. Sahle et al. Simulation of biochemical networks ssing copasi: a complex pathway simulator. In Proceedings of the 37th conference on Winter simulation, pages 1698–1706. Winter Simulation Conference, 2006.

[5] E. Chiprout and M.S. Nakhla. Asymptotic Waveform Evaluation. Kluwer, Norwell, MA, 1994.

[6] P. Feldmann and R.W. Freund. Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process. IEEE Transactions on Computer-Aided Design, 14(5):639–649, May 1995.

[7] A. Odabasioglu, M. Celik, and L.T. Pileggi. PRIMA: passive reduced-order interconnect macromodellingalgorithm. In Proceedings of the IEEE International Conference on Computer Aided Design, pages 58–65, November 1997.

[8] J. Phillips and L. M. Silveira. Poor Man's TBR: a simple model reduction scheme. Design, Automation and Test in Europe Conference and Exhibition, 2004.

[9] F. Wang and J. White. Automatic model order reduction of a microdevice using the arnoldi approach. In International Mechanical Engineering Congress and Exposition, pages 527–530, 1998.

[10] J. R. Phillips. Projection-Based Approaches for Model Reduction of WeaklyNonlinear Time-Varying Systems. IEEE Transactions on Computer-Aided Design, 22(2):171–187, 2003.

[11] Y. Chen and J. White. A Quadratic Method for Nonlinear Model Reduction. Proc. of International Conference on Modeling and Simulation of Microsystems, pages 477–480, 2000.

[12] A. Demir and J. Roychowdhury. Nonlinear TVP: Reduced-order modelling of nonlinear systems, June 1998. Draft, Bell Laboratories.

[13] P. Li and L. T. Pileggi. NORM: Compact Model Order Reduction of Weakly Nonlinear Systems. Proceedings of the IEEE Design Automation Conference, 2003.

[14] J. Phillips. Model Reduction of Time-Varying Linear Systems UsingApproximate Multipoint Krylov-Subspace Projectors. In Proceedings of the IEEE International Conference on Computer Aided Design, November 1998.

[15] N. Dong and J. Roychowdhury. Piecewise Polynomial Nonlinear Model Reduction. Proceedings of the IEEE Design Automation Conference, 2003.

[16] N. Dong and J. Roychowdhury. Automated Nonlinear Macromodelling of Output Buffers for High-Speed Digital Applications. Proceedings of the IEEE Design Automation Conference, 2005.

[17] S.K. Tiwary and R.A. Rutenbar. Scalable trajectory methods for on-demand analog macromodel extraction. In Design Automation Conference, 2005. Proceedings. 42nd, pages 403–408, 13-17 June 2005.

[18] S.K. Tiwary and R.A. Rutenbar. Faster, parametric trajectory-based macromodels via localized linear reductions. In Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on, pages 876–883, Nov. 2006.

[19] B. N. Bond and L. Daniel. Stabilizing schemes for piecewise-linear reduced order models via projection and weighting functions. In Proc. IEEE/ACM International Conference on Computer-Aided Design ICCAD 2007, pages 860–867, 4–8 Nov. 2007.

[20] Y. Gheisari, H. R. Shaker, M. A. Torabi, and M. Samavat. Accuracy and efficiency enhanced nonlinear model order reduction. In Proc. IEEE International Symposium on Computer-Aided Control Systems Design, pages 3007–3012, 4–6 Oct. 2006.

[21] S. Sastry. Nonlinear Systems - Analysis, Stability and Control. Springer, 1999.

[22] G. H. Golub and C. F. Van Loan. Matrix Computations. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[23] R.T. Chang et al. Near Speed-of-Light Signaling Over On-chip Electrical Interconnects. Solid-State Circuits, IEEE Journal of, 38:834–838, May 2003.