

New Numerical Techniques and Tools in SUGAR for 3D MEMS Simulation

Z. Bai*, D. Bindel†, J. Clark‡, J. Demmel†, K. Pister‡, N. Zhou‡

*Department of Computer Science, University of California, Davis, USA, bai@cs.ucdavis.edu

†Department of Computer Science, University of California, Berkeley, USA
dbindel@cs.berkeley.edu, demmel@cs.ucdavis.edu

‡Berkeley Sensor and Actuator Center, University of California, Berkeley, USA
jvclark@bsac.eecs.berkeley.edu, pister@eecs.berkeley.edu, nzhou@bsac.eecs.berkeley.edu

ABSTRACT

SUGAR is a nodal analysis package for 3D MEMS simulation that owes its heritage and its name to the SPICE family of circuit simulation. SUGAR has undergone the stage of proof-of-concept which showed that nodal analysis was in fact just as accurate and much faster than finite element simulation on many MEMS problems. The upcoming major release of SUGAR is version 2.0, which includes a number of new features, such as 3D beam and gap elements, thermal expansion, linearly and rotationally accelerating frames, and user-defined models.

In this paper, we introduce two new numerical techniques and tools to be incorporated in the future release of SUGAR, namely scaling schemes to remedy artificial ill-conditioning and Krylov subspace based reduced-order modeling techniques for efficient transient analysis of dynamical systems.

Keywords: MEMS, SUGAR, modified nodal analysis, scaling, reduced-order modeling

1 Introduction

The potential impact of simulation, verification and synthesis tools to MEMS design, fabrication and applications is hard to overstate [8], [7]. There are a large number of efforts to bring such tools to the market. SUGAR is one of such efforts and it is a modified nodal analysis package for 3D MEMS simulation that owes its heritage and its name to the SPICE family of integrated circuit simulation [4], [3]. It is an open source package. SUGAR has undergone the stage of proof-of-concept which showed that nodal analysis was in fact just as accurate and much faster than finite element simulation on many MEMS problems. The upcoming release of SUGAR is version 2.0, which includes a number of new features, such as 3D beam and gap elements, thermal expansion, linearly and rotationally accelerating frames, and user-defined models.

In this paper, we will introduce two new numerical techniques to be incorporated in the future release of SUGAR, namely scaling schemes to remedy artificial ill-conditioning and reduced-order modeling techniques for efficient transient analysis of dynamical systems.

The coefficients of the system matrices for the ODEs used in SUGAR vary across several orders of magnitude. This poor scaling causes the system to be artificially ill-conditioned, so that when an ODE solver and Lanczos process involve solutions to linear systems associating with these matrices they get undeservedly inaccurate results. Large errors from solving these linear systems contaminate the numerical solution to the ODE, so that the ODE solver must take very small time steps to resolve illusory high-frequency vibrations. To cope with scaling difficulties, we transform the system using two diagonal matrices, or equilibrate it, and remove ill-conditioning associated with poor scaling.

The basic idea of reduced-order modeling of a dynamical system is to replace the original system by an approximating system with much smaller state-space dimension. An accurate and effective reduced-order model can be applied for steady state analysis, transient analysis and sensitivity analysis. As a result, it can significantly reduce design time and allow for aggressive design strategies. Such a computational prototyping tool will let designers try “what-if” experiments in hours instead of days. Krylov subspace methods are emerging numerical techniques for reduced-order modeling of large scale dynamical systems. They have led to a major breakthrough in the field. In this part of work, we will demonstrate how to use Krylov subspace-based reduced-order modeling techniques for transient analysis of the nonlinear ODE systems arising from the SUGAR simulation. We have achieved a factor of 60 speedup for the transient analysis of an electrostatic beam gap-closing actuator device.

2 Simulation Model and Case Study

Currently, SUGAR uses the following matrix representation of the transient dynamics equation [3]:

$$M\ddot{q} + D\dot{q} + Kq = F(t, q) \quad (1)$$

where q is a state vector of length N . M , D and K are the $N \times N$ system matrices, which are analogous to the mass, damping, and stiffness in a purely mechanical systems. $F(t, q)$ is the $N \times 1$ excitation vector for the system. It is a function of time and state. We can write

$F(t, q) = Bu(t, q)$, where B is an $N \times p$ input influence array to indicate the position input excitation, and $u(t, q)$ is the input excitation source including nonlinear electrostatic force.

Furthermore, system (1) can be rewritten as the following multi-input and multi-output (MIMO) form

$$\begin{cases} M\ddot{q} + D\dot{q} + Kq = Bu(t, q) \\ y = L^T q, \end{cases} \quad (2)$$

where $y(t)$ is the output of the system, and L is $N \times m$ output influence array and is chosen to extract the components of state vector of interest. The equation (2) can be equivalently cast in the following form

$$\begin{cases} C\dot{x} + Gx = \hat{B}u(t, x) \\ y = \hat{L}^T x \end{cases} \quad (3)$$

where

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad \hat{L} = \begin{bmatrix} L \\ 0 \end{bmatrix}$$

and

$$C = \begin{bmatrix} D & M \\ I & 0 \end{bmatrix}, \quad G = \begin{bmatrix} K & 0 \\ 0 & -I \end{bmatrix}$$

The system (3) is similar to well-studied linear time-invariant MIMO systems (in that case, $u(t, q)$ is $u(t)$, independent of q). We can try to benefit from techniques developed for the linear MIMO systems, and exploit the idea of so-called “nonlinear dynamics using linear modes” as in [1].

We use a simple electrostatic beam actuator shown in Fig. 1 as an example throughout this paper. The top actuator plates are $5\mu\text{m}$ wide and $100\mu\text{m}$ long at the end of a flexure $2\mu\text{m}$ wide and $100\mu\text{m}$ long. We use a parallel plate approximation to the electrostatic force $F(t, q) = Bu(t, q)$, where B has ones in the components corresponding to the displacement in y direction of nodes b and c , and

$$u(t, q) = -\frac{1}{4}\epsilon_0 A \frac{v(t)^2}{\text{gap}(q)^2},$$

ϵ_0 is permittivity of free space, and A is the area of the plate. All structures are fabricated in a $2\mu\text{m}$ polysilicon layer; M and K use the material properties of this layer, D is based on simple Couette damping, and is proportional to M . L selects the y displacement of the node c . While this model may not be highly physically accurate, it serves to illustrate our techniques. For more detail about modeling of the electrostatic beam actuator, see [3].

3 Scaling

The nonzero coefficients of the matrix C in our example vary across 40 decimal orders of magnitude, causing

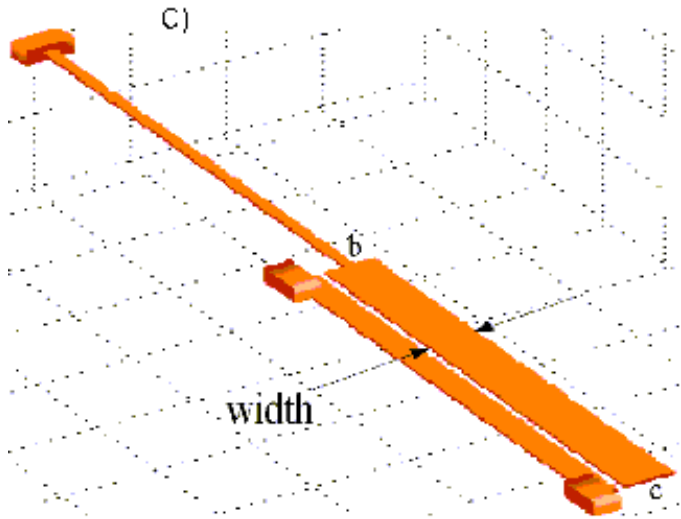


Figure 1: electrostatic beam actuator

C to be very poorly scaled. Consequently, C has an enormous 2-norm condition number of around $7.7 \cdot 10^{23}$. While such a large condition number does not guarantee that a particular linear system involving C will be inaccurately solved, it does raise suspicion, and renders many of the standard error bounds ineffective. Since we solve linear systems involving C when forming the Krylov subspaces for the Lanczos procedure and at each step of the ODE solver, these large condition numbers cause concern us.

There are several standard techniques for dealing with ill-scaled problems [6]. The simplest is *equilibration*, which involves multiplying C on the left by a diagonal matrix which scales each row of the matrix to have unit norm. Multiplying the matrix by diagonal matrices on both the left and the right can improve the condition number even further, but undoing the column scaling subsequent to solving a linear problem with C can undo whatever benefits were conferred by the column scaling. Another technique which often remedies the effects of ill-scaling is iterative refinement, which involves computing the residual $r = b - C\hat{x}$ for the computed solution x , solving a linear system to find the approximate error $C(x - \hat{x}) = r$, and then adding the approximate error to obtain a corrected solution.

4 Reduced-Order Modeling

Krylov subspace methods are emerging numerical techniques for reduced-order modeling of large scale dynamical systems. They have led to a major breakthrough in the field, see the survey paper [5] and the references therein. The need and challenges of reduced-order modeling techniques for simulating MEMS devices are presented in [8], [7]. In this part of work, we will demonstrate how to use Krylov subspace-based reduced-order modeling techniques for transient analysis of the nonlin-

ear system (3).

The idea of nonlinear dynamics using linear modes is presented in [1]. A selected set of eigenvectors (modes) of the matrix $M^{-1}K$ are used and the damping term D is neglected. In our approach, we use a Krylov-subspace based technique for the reduced-order modeling, specifically, the basis vectors of Krylov subspaces computed via the efficient Lanczos process are directly used without the need of further computing for getting eigenvectors (modes). The damping term is included in our approach.

The Krylov subspace based methods for reduced-order modeling have two steps, namely, Lanczos or Arnoldi procedure for generating the bases of the underlying Krylov subspaces, and the model order reduction by coordinate transformation using Krylov bases. We will use Lanczos process here. For simplicity, we only present the single vector Lanczos process, i.e., for the SISO ($p = m = 1$) system of (3). For the detail of Lanczos process including the multi-vector Lanczos process, see [2, secs 7.8, 7.9 and 7.10].

Let $A = -(G + s_0C)^{-1}C$ and $R = (G + s_0C)^{-1}B$, where s_0 is chosen to be an expansion close to the frequency of interest. The governing equations of the Lanczos process with A , R and L can be summarized compactly in matrix form as follows:

$$AV_n = V_nT_n + \rho_{n+1}v_{n+1}e_n^T \quad (4)$$

$$A^TW_n = W_n\tilde{T}_n + \eta_{n+1}w_{n+1}e_n^T \quad (5)$$

where T_n and \tilde{T}_n are tridiagonal matrices, and the columns of $N \times n$ matrices W_n and V_n are called left and right Lanczos vectors and span the so-called left and right Krylov subspaces

$$\mathcal{K}_n(A, R) = \text{span}\{R, AR, \dots, A^{n-1}R\}$$

$$\mathcal{K}_n(A^T, L) = \text{span}\{L, A^TL, \dots, (A^T)^{n-1}L\}$$

and furthermore they satisfy the biorthogonality condition

$$W_n^TV_n = \Delta_n, \quad W_n^Tv_{n+1} = 0, \quad V_n^Tw_{n+1} = 0. \quad (6)$$

where Δ_n is a diagonal matrix. From (4) and (5), it follows that

$$W_n^TAV_n = \Delta_nT_n \quad (7)$$

We note that for the implementation of Lanczos process, the matrix A is involved only in the form of matrix-vector multiplication, hence the structure and sparsity of A , and correspondingly the matrices M , D and K can be exploited for memory saving and computational efficiency.

This Krylov-Lanczos process can be used as a building block for reduced-order modeling and applications for steady-state, transient and sensitivity analysis of dynamical systems. In this paper, we will focus on how to do transient analysis of the system (3).

Again, let $A = -(G + s_0C)^{-1}C$ and $R = (G + s_0C)^{-1}B$, then the equation (3) can be written as

$$\begin{cases} -A\dot{x}(t) + (I + s_0A)x(t) = Ru(t, x), \\ y(t) = L^Tx(t), \end{cases}$$

Let V_n be the right Lanczos vectors generated by the above Lanczos procedure, and let

$$x(t) \approx V_nz(t),$$

where $z(t)$ is the new state vector of length n , then an approximate dynamical system is

$$\begin{cases} -AV_n\dot{z}(t) + (I + s_0A)V_nz(t) = Ru(t, V_nz), \\ \tilde{y}(t) = L^TV_nz(t). \end{cases}$$

Multiplying W_n^T from the left, we have

$$\begin{cases} -W_n^TAV_n\dot{z}(t) + W_n^T(I + s_0A)V_nz(t) = W_n^TRu(t, V_nz), \\ \tilde{y}(t) = L^TV_nz(t). \end{cases}$$

Therefore, by (7), it yields an n -th reduced-order model

$$\begin{cases} -\Delta_nT_n\dot{z}(t) + (\Delta_n + s_0\Delta_nT_n)z(t) = B_nu(t, V_nz), \\ \tilde{y}(t) = L_n^Tz(t). \end{cases} \quad (8)$$

where $B_n = W_n^TR$ and $L_n = V_n^TL$.

As a result, instead of solving the original system (3) of dimension N for $y(t)$, one can solve the reduced-order model (8) of dimension n for \tilde{y} as an approximation of $y(t)$. The strength of this approach derives from the fact that the value of n is much less than N in many cases. The constructed Krylov subspaces contain the necessary information for the dominant modes to capture the dynamic response of the system.

5 Results and Discussion

For the case study of the gap-closing actuator described in section 2, an initial voltage $v(t)$ is applied across the gap, which increase linearly in time. The voltage $v(t)$ rams from 5V at $t = 10\mu\text{sec}$ to 12V at $t = 500\mu\text{sec}$, and then release. The displacement in y direction of the node c is selected to be observed. The initial voltage step starts the device to resonate. As the voltage increases at a linear rate, the gap decreases at a nonlinear rate due to the electrostatic force increasing proportionally to $1/\text{gap}(q)^2$. This force also causes the period of oscillation to increase. Once the voltage is removed, the actuator exponentially decays back to equilibrium due to viscous air damping. These phenomena are captured in numerical simulation. Fig. 2(a) shows the displacement versus time. Fig. 3 show the decays back to equilibrium. In Fig. 2(a), the displacement curves of the original system and the reduced-order models are overlapped. The accuracy of the reduced-order model can be seen in Fig. 2(b). There relative

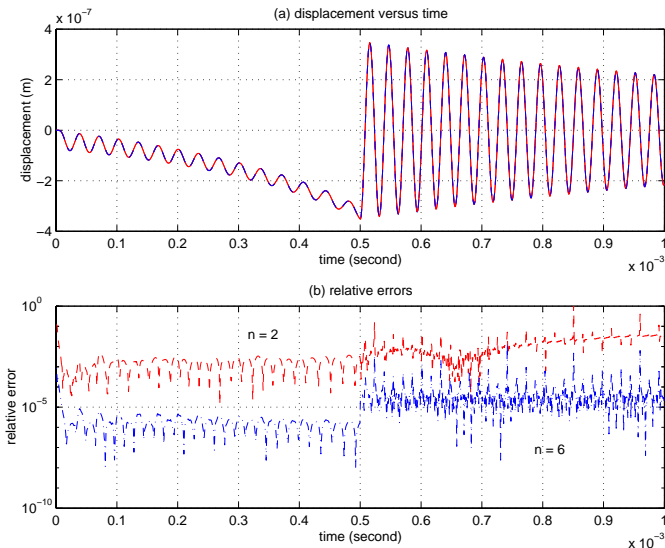


Figure 2: (a) displacement vs. time of full and reduced-order systems, (b) relative errors of reduced-order systems

errors $|y(t) - \tilde{y}(t)|/|y(t)|$ are plotted for the 2nd and 6th order of the reduced-order model. The order of the original model is $N = 30$.

The solution of the original full system for time domain from 0 to 10^{-3} second took 28533 seconds. However, the construction of the 2nd and 6th order system and the solution of the reduced-order systems took 2 seconds and 428 seconds respectively. Therefore, with the satisfactory accuracy of the 6th order system, we have achieved a factor of 60 speedup for the transient analysis of the electrostatic beam gap-closing actuator.

Although M , D and K are all symmetric positive definite, they are ill-scaling and ill-conditioned. Let us examine the matrix C in detail. The condition number of the matrix C is $7.7 \cdot 10^{23}$. Its largest singular value is 1 and the smallest singular value is $1.3 \cdot 10^{-24}$. With row equilibration, the condition number is reduced to $8.2 \cdot 10^9$ and the largest and smallest singular values are 2.3 and $2.8 \cdot 10^{-10}$, respectively. Furthermore, with both row and column equilibrations, we have a nearly perfect conditioned matrix, with condition number 48, and the largest and smallest singular values are 2.1 and $4.4 \cdot 10^{-2}$, respectively. The issue concerning to the ill-scaling and artificial ill-conditioning are pointed out. The impact of scaling and equilibration to the reduced-order modeling methods and ODE solvers are still under investigation.

All numerical simulations were run on a SUN 440MHz Ultra 10 workstation. Matlab is used as the computing environment. The results of ODE solver ode15s is reported in this paper.

In the future, we plan to develop a reduced-order modeling technique for a more general case where the right-hand side of equation (1) is a function of q , \dot{q} , and

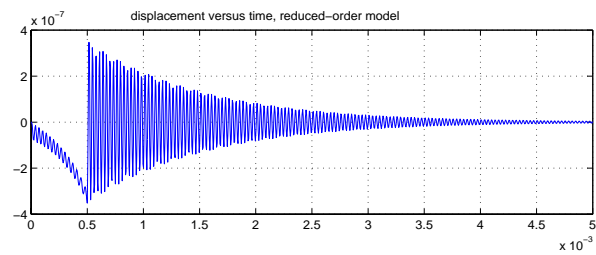


Figure 3: displacement vs. time, back to equilibrium, using the 6th reduced-order model

t :

$$M\ddot{q} + D\dot{q} + Kq = F(q, \dot{q}, t).$$

Here, the function $F(q, \dot{q}, t)$ may also include the variations in M , D , and K matrices that depend on state and time.

REFERENCES

- [1] G. K. Ananthasuresh, R. K. Gupta, and S. D. Senturia. An approach to macromodeling of MEMS for nonlinear dynamic simulation. In *Microelectromechanical Systems (MEMS), ASME Dynamics Systems & Control (DSC) ser.*, volume 59, pages 401–407, 1996.
- [2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, Philadelphia, 2000. xxx+410 pages.
- [3] J. V. Clark, N. Zhou, D. Bindel, L. Schenato, W. Wu, J. Demmel, and K. S. J. Pister. 3D MEMS simulation using modified nodal analysis. In *Proceedings of Microscale Systems: Mechanics and Measurements Symposium, Orlando, Florida, pp. 68-75, June 8, 2000*, pages 68–75.
- [4] J. V. Clark, N. Zhou, and K. S. J. Pister. MEMS simulation using SUGAR v0.5. In *Proc. Solid-State Sensors and Actuators Workshop, Hilton Head Island, SC, USA. June 8-11, 1998*, pages 191–196.
- [5] R. W. Freund. Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation. In *Applied and Computational Control, Signals, and Circuits, Vol. 1*, pages 435–498. Birkhäuser, Boston, 1999.
- [6] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, Philadelphia, PA, 1996.
- [7] S. D. Senturia. CAD challenges for microsensors, microactuators, and microsystems. In *Proceedings of the IEEE*, volume 86, pages 1611–1626, 1998.
- [8] S. D. Senturia, N. Aluru, and J. White. Simulating the behavior of MEMS devices: Computational methods and needs. *IEEE Computational Science and Engineering*, 4:30–43, 1997.