Nodal Analysis for MEMS Design Using SUGAR v0.5

Ningning Zhou* Jason Vaughn Clark** and K. S. J. Pister

*Dept. of Mechanical Engineering, UC Berkeley, nzhou@bsac.eecs.berkeley.edu
**Dept. of Applied Physics (AS&T), UC Berkeley, jvclark@bsac.eecs.berkeley.edu
Berkeley Sensor and Actuator Center, 497 Cory Hall, UC Berkeley, Berkeley, CA

ABSTRACT

This paper presents the development of a simulation program (SUGAR) for planar MEMS devices. The approach is based on nodal analysis to solve coupled nonlinear differential equations. The current version SUGAR v0.5 can perform DC, steady state, and transient analysis. It is implemented in Matlab [1] and the accuracy of the program is comparable with analytical solutions, experimental data and FEM simulations. The preliminary work shows that this approach could lead to an accurate, fast, and high-level simulation package for MEMS.

Keywords: MEMS, SUGAR, nodal analysis, nonlinear model, Matlab

INTRODUCTION

Rapid development of MEMS technology requires CAD tools for support. Most of the research in the MEMS simulation area have focussed on process modeling [3, 4], finite element analysis (FEA) and boundary element methods (BEM) for electromechanical functional modeling [5, 6]. They are generally device and analysis oriented rather than design oriented. When users want to design higher level systems with dozens or even thousands of components, it is far beyond the reach of traditional FEA/BEA based tools.

Nodal analysis has been widely used for formulating system equations in circuit analysis such as SPICE. It is done by decomposing the circuit into N-terminal devices. Each device is modeled by ordinary differential equations (ODEs) with coefficients parameterized by device geometry, and material properties derived from measurements or process specifications. Devices are linked together at their terminals, or nodes, and the resulting coupled differential equations can be solved as a system of nonlinear ODEs using nodal analysis [7].

SUGAR uses a similar approach to simulate MEMS devices. It abstracts the MEMS structures in terms of three basic elements (i.e. beams, gaps, and anchors), and builds the ODE models for each kind. The system equations can be formulated according to the node connectivity information provided in the input file, and solved using nodal analysis.

In the following sections, some details of applying nodal analysis in MEMS design are described. The model implementations and the numerical algorithms for DC, steady state and transient analysis are also presented with some simulation examples.

NODAL ANALYSIS APPROACH

In a MEMS device the law of static equilibrium is applied to each node such that the summation of the forces and moments on the nodes are equal to zero. This is analogous to Kirchhoff's current law in circuit analysis (e.g. forces can be seen as branch quantities like currents while displacements at each node can be thought as node quantities like voltages). The forces and displacements on each node can be related by structural models.

To demonstrate the method of assemblage, the structure shown in Figure 1 is chosen. It contains three anchor elements, one beam element and an electrostatic gap element. Since anchor elements are fixed to the substrate with no degree of freedom, we only need to include beams and gaps in the analysis.

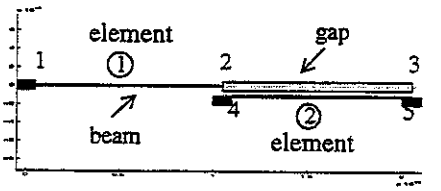

Figure 1. A simple MEMS structure.

To formulate the total system, we must formulate the individual element first. For the beam element that extends from node *1* to node *2* we have

$$f_n^{\,1} = f_n^1(q_1, q_2) \qquad n = 1,2 \qquad (1)$$

and for the gap element (nodes 2,3,4,5) we have

$$f_n^{\,2} = f_n^2(q_2, q_3, q_4, q_5) \qquad n = 2,3,4,5 \qquad (2)$$

where f_n represents the forces $\{F_{x,n}, F_{y,n}, M_n\}$ applied at node n, and q_n represents the node displacements $\{x_n, y_n, \theta_n\}$. The super- and subscripts on f indicate the element number and node number respectively. Each node has three degrees of freedom in the planar case: x, y direction and rotation. Bear in mind that the f's are the internal nodal forces. When two sets of equations are assembled, the sum of such internal forces at each node point is equal to the external load P, which in this example are the equivalent electrostatic forces generated at nodes *2, 3, 4,* and *5*. We must also be aware that nodal displacements at the element levels are the same as those after assemblage.

Following these guidelines, the assembled system equations for each node are

$$
\begin{aligned}
P_1 &= f_1^{\,1}(q_1, q_2) \\
P_2 &= f_2^{\,1}(q_1, q_2) + f_2^{\,2}(q_2, q_3, q_4, q_5) \\
P_3 &= f_3^{\,2}(q_2, q_3, q_4, q_5) \\
P_4 &= f_4^{\,2}(q_2, q_3, q_4, q_5) \\
P_5 &= f_5^{\,2}(q_2, q_3, q_4, q_5)
\end{aligned} \tag{3}
$$

Since the displacements for anchor nodes *1, 4,* and *5* are zero they are removed. The final system equations become

$$
\begin{aligned}
P_2 &= f_2^{\,1} + f_2^{\,2} = f_2^{\,1}(q_2) + f_2^{\,2}(q_2, q_3) \\
P_3 &= f_3^{\,2}(q_2, q_3)
\end{aligned} \tag{4}
$$

By using this method we can formulate the system's equations of motion.

MODELS

Models for linear beam, nonlinear beam, and nonlinear electrostatic gap have been developed. As with SPICE, there are different model levels in SUGAR which allow the user to trade off accuracy and speed.

Linear Beam Model

Traditional finite element analysis attempts to describe the behavior of beams by dividing them up into smaller parts and summing the contribution of each of these sub-elements. Instead of breaking the beam up into numerous finite elements we model simple beams more efficiently by using ODE's which are parameterized by displacements and forces at the beam's end nodes.

Coupled beams will have common deflection and slope at shared nodal points. This condition satisfies the continuity of both the deflection and slope. The transverse deflection $v(x)$, axial displacement $u(x)$, and the angle of rotation $\theta(x)=(dv/dx)$ can be described by three degrees of freedom at each node (Figure 2).

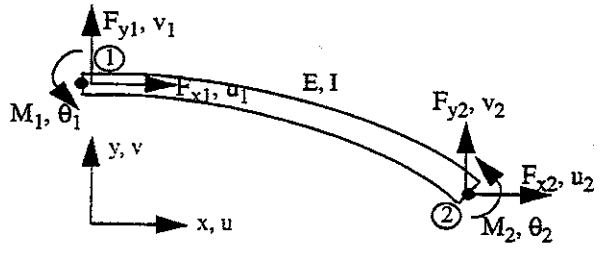

Figure 2. A bent beam showing nodal forces, moments, and coordinates.

In the region between nodes, the equation of equilibrium **[8]** for a beam with uniform cross section is

$$
\frac{\partial^4}{\partial x^4} v(x) = 0 \tag{5}
$$

The solution to Eq. (5) is a cubic polynomial function of x,

$$
v(x) = a_0 + a_1 x^1 + a_2 x^2 + a_3 x^3 \tag{6}
$$

Imposing the boundary conditions at both end nodes yield four equations and four unknowns,

$$
\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \frac{1}{L} \begin{bmatrix} L^3 & 0 & 0 & 0 \\ 0 & L^3 & 0 & 0 \\ -3L & -2L^2 & 3L & L^3 \\ 2 & L & -2 & L \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{bmatrix} \tag{7}
$$

Solving for the coefficients of $v(x)$ and $\theta(x)$ in terms of v_n and θ_n at node $n = 1, 2$, and grouping like-terms results in

$$
v(x) = H_1(x)v_1 + H_2(x)\theta_1 + H_3(x)v_2 + H_4(x)\theta_2 \tag{8}
$$

For a bar element with constant axial stress or strain, the axial displacement $u(x)$ is assumed to be a linear in x,

$$
u(x) = H_5(x)u_1 + H_6(x)u_2 \tag{9}
$$

where the cubic functions $H_i(x)$ are the Hermitian shape functions:

$$
\begin{aligned}
H_1(x) &= 1 - 3\left(\frac{x}{L}\right)^2 + 2\left(\frac{x}{L}\right)^3 & H_2(x) &= x - 2\left(\frac{x^2}{L}\right) + \left(\frac{x^3}{L^2}\right) \\
H_3(x) &= 3\left(\frac{x}{L}\right)^2 - 2\left(\frac{x}{L}\right)^3 & H_4(x) &= -\frac{x^2}{L} + \frac{x^3}{L^2} \\
H_5(x) &= 1 - \frac{x}{L} & H_6(x) &= \frac{x}{L}
\end{aligned} \tag{10}
$$

The transverse (Eq. 8) and axial (Eq. 9) displacements are used to determine the bending and tensile strain energy respectively. For the simplified linear case we have

$$
S = \frac{EI}{2}\int_0^L \left(\frac{\partial^2}{\partial x^2} v(x)\right)^2 dx + \frac{EA}{2}\int_0^L \left(\frac{\partial}{\partial x} u(x)\right)^2 dx \tag{11}
$$

where E is the modulus of elasticity, I the moment of inertia, and A the cross sectional area of the beam. The Young's modulus and layer thickness h are defined in a process file.

Hence, the stiffness for the beam can be obtained by using Castiglianos theorem **[8]**

$$
F_i = \frac{\partial S}{\partial q_i} \tag{12}
$$

where F_i represents the six force components F_{x1}, F_{y1}, M_1, F_{x2}, F_{y2}, $M2$, and q_i represents the six displacement components x_1, y_1, θ_1, x_2, y_2, θ_2. Substituting Eq. (11) into Eq. (12) yields the stiffness

$$
k_{ij} = EI\int_0^L H''_i(x)H''_j(x)dx + EA\int_0^L H'_i(x)H'_j(x)dx \tag{13}
$$

which gives

$$
[k] = \frac{E}{L^3} \begin{bmatrix} AL^2 & 0 & 0 & -AL^2 & 0 & 0 \\ 0 & 12I & 12I & 0 & -12I & 6IL \\ 0 & 6IL & 6IL & 0 & -6IL & 2IL \\ -AL^2 & 0 & 0 & AL^2 & 0 & 0 \\ 0 & -12I & -12I & 0 & 12I & -6IL \\ 0 & 6IL & 6IL & 0 & -6IL & 4IL \end{bmatrix} \tag{14}
$$

for beam structures.

A mass matrix can be found by equating internal and external work due to virtual displacements **[9]**. This results in

$$
m_{ij} = \int_0^L \rho(x)A(x)H_i(x)H_j(x)dx \tag{15}
$$

where $\rho(x)$ and $A(x)$ are the density and cross sectional area at position x respectively. For uniform beams, ρA is constant along the length of the beam and we have

$$[m] = \frac{\rho AL}{420}\begin{bmatrix} 140 & 0 & 0 & 70 & 0 & 0 \\ 0 & 156 & 22L & 0 & 54 & -13L \\ 0 & 22L & 4L^2 & 0 & 13L & -3L^2 \\ 70 & 0 & 0 & 140 & 0 & 0 \\ 0 & 54 & 13L & 0 & 156 & -22L \\ 0 & -13L & -3L^2 & 0 & -22L & 4L^2 \end{bmatrix} \quad (16)$$

For level-1 damping we use a simple Couette flow model considering only the region between the structure and substrate.

We approximate the damping matrix to be

$$c_{ij} = \int_0^L \bar{c} H_i(x) H_j(x) dx \quad (17)$$

where $\bar{c} = \mu w/\Delta$ is the viscous damping per unit length. This results in

$$[c] = \frac{\mu L w}{420\Delta}\begin{bmatrix} 140 & 0 & 0 & 70 & 0 & 0 \\ 0 & 156 & 22L & 0 & 54 & -13L \\ 0 & 22L & 4L^2 & 0 & 13L & -3L^2 \\ 70 & 0 & 0 & 140 & 0 & 0 \\ 0 & 54 & 13L & 0 & 156 & -22L \\ 0 & -13L & -3L^2 & 0 & -22L & 4L^2 \end{bmatrix} \quad (18)$$

where μ is the viscosity of the fluid environment between the device and substrate, Δ is the distance from the device to the substrate, and w is the width. Both μ and Δ are defined in a process file.

Since each structure may have differing orientations, all local coordinates are transformed into global coordinates by a transformation matrix

$$[T] = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

where θ is the orientation of a structure measured counter-clockwise from the positive x axis. For example, to transform the stiffness matrix from local coordinates to global coordinates, the transformation matrix is applied in the following way:

$$[k]_{global} = [T]^T [k]_{local} [T] \quad (20)$$

The assemblage of the set of individual matrices [k], [m], and [c] into collective system matrices, [K], [M], and [C] respectively, where all structures are coupled at common nodes, is accomplished by nodal superposition. Thus, the equation of motion describing the dynamics of the entire system can be expressed in the familiar form

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{F\} \quad (21)$$

where the nodal displacements $\{q\} = \{x_1\ y_1\ \theta_1\ \ldots\ x_N\ y_N\ \theta_N\}$ and nodal forces $\{F\} = \{F_{x1}\ F_{y1}\ M_1\ \ldots\ F_{xN}\ F_{yN}\ M_N\}$ are both $1x3N$ column vectors, and N is the total number of dynamic (non-constrained) nodes.

Nonlinear Beam Model

In the linear beam model, $\{q_i\}$ is proportional to $\{F_i\}$. This model is satisfactory for small deflections; however, it doesn't consider the nonlinearity such as the axial shortening and the effect of axial force on the stiffness of the beam. An axially applied compressive force on a beam decreases it's bending stiffness and an axially applied tensile force increases it's bending stiffness.

A level-2 (nonlinear) beam model was developed for the small rotation condition. Only the geometric nonlinearity is considered. This model starts from cubic and linear interpolation functions for the lateral and longitudinal displacements as in Eq. (8) and Eq. (9), and calculates the strain at every point **[10]**

$$\bar{\varepsilon}(x,\eta) = \frac{du}{dx} + \frac{1}{L}\cdot\int_0^L \left(\frac{1}{2}\cdot\left(\frac{dv}{dx}\right)^2\right)dx + \eta\cdot\frac{d^2v}{dx^2} \quad (22)$$

where η is the distance from the neutral axis. The middle term in above equation is the average of $(dv/dx)^2/2$ along the beam. Compared with the linear model, a new term is introduced to take into account the contribution of the longitudinal displacement to the strain.

Substituting Eq. (22) into the following gives the strain energy

$$S = \int_V \frac{1}{2} E \bar{\varepsilon}^2 dV . \quad (23)$$

Then substituting Eq. (23) into Eq. (12) yields the nodal forces as a function of displacement.

Figure 3a shows a simulation of a clamped-clamped beam with a central concentrated load. Simulation of the load-deflection response (Figure 3b) is in close agreement with ABAQUS **[2]** over a typical range of operation.

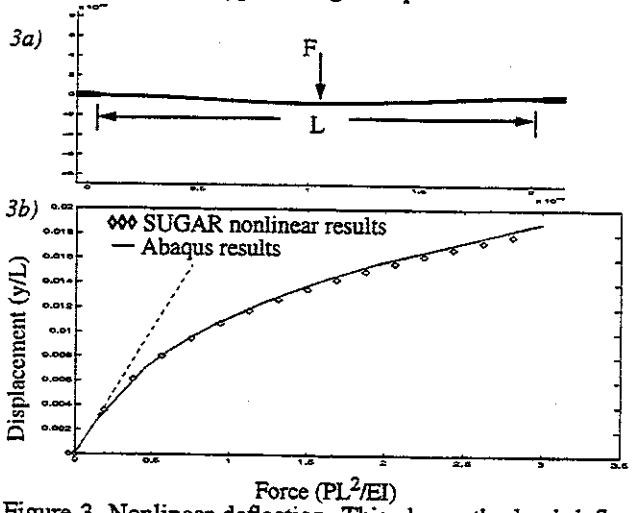

Figure 3. Nonlinear deflection. This shows the load-deflection curve for the clamped-clamped beam with a concentrated load.

Nonlinear Gap Model

Electrostatics forces on parallel conducting plates gener-

ally have a $1/d^2$ dependence for first order approximations. Considering electrostatically attracted parallel beams (gaps), if the distance between the two beams d is much less than the widths of the beams h (layer thickness), the total force generated can be approximated to be simply $(-1/2)\varepsilon_0 V^2 A/d^2$. However, as the gap distance between the beams increase, the attractive force between the beams decrease, and charges that have accumulated near the gap spacing spread out further due to their repulsive forces. This spreading of the charges is not uniform, in that surface charge density near the edge of the beam is greater than the center. This increase in surface charge density at the edges results in a significant fringing field contribution term in the overall capacitance [11]. Taking the fringing field contribution into account, the total force generated by our level-2 gap model is approximated to be

$$F = \frac{-1}{2}\varepsilon_0 V^2 Lh\frac{\alpha(d/h)}{d^2} \qquad (24)$$

where the factor $\alpha > 1$ is the fringing field contribution. For the range of gap distances d such that $0 < d < h$, we assume α to vary linearly. The level-1 gap model is similar to level-2 except the bending and fringing field contributions are not considered.

For conditions where interacting beams are not parallel but are subject to say, bending, the varying gap distance along the length of the beam must be considered. As described previously, the transverse displacements along the length of the beam can be described by nodal coordinates and Hermitian shape functions. Referring to Figure 4,

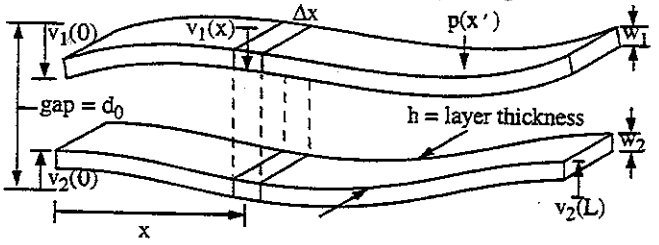

Figure 4. The level-2 model of an electrostatic actuator. There is a distributed electrostatic force p(x) along the length of the beam which varies due to bending.

the distance between the beams at position x is

$$d(x) = d_0 + v_1(x) - v_2(x) \qquad (25)$$

where d_0 is the original gap spacing, and $v_1(x)$ and $v_2(x)$ are the transverse displacements of the beams at position x. The nodal forces due to a distributed force per unit length are

$$F_i = \int_0^L p(x)H_i(x)dx\,. \qquad (26)$$

For the electrostatic case, the force per unit length is approximated to be

$$p(x) = \frac{-1}{2}\varepsilon_0 V^2 h\frac{\alpha(d(x)/h)}{d(x)^2}\,. \qquad (27)$$

In solving the integral in Eq. (26) for a general analytical expression the equivalent nodal forces and moments on each beam are obtained. This analytical expression is parameterized by voltage and nodal coordinates only, as required to fit the nodal analysis scheme of SUGAR. These nodal forces are added to the system force vector in Eq. (21) for further analysis.

Gap Model With Contact Forces

With the linear beam model and nonlinear electrostatic gap model, SUGAR can find the equilibrium position of the structure before and after pull-in. To simulate the contact behavior between the beams of an electrostatic gap, repulsive nodal forces are added to the interacting beams. The gap spacing and contact penetration depth of the beams are determined by Eq. (25). The equal and opposite contact forces prevent the absolute gap distance from approaching zero so that the electrostatic force model will not approach infinity.

A model for contact forces is chosen such that $|F_c| >> |F_e|$ when $d < d_c$ and $|F_c| << |F_e|$ when $d > d_c$, where F_c, F_e, and d_c are the contact force, electrostatic force, and critical gap distance respectively. Since the electrostatic forces are attractive and proportional to d^{-2}, we chose the contact force to be repulsive and proportional to d^{-3}. To avoid convergence problems, we made modifications for this function when d is very close to zero and less than zero.

An electrostatic gap (level-1) pull-in simulation is shown in Figure 5. The test structure is shown in Figure 5a. The pull-in voltage V as a function of beam length L is shown in Figure 5b. A good fit to the experimental data was obtained with a Young's modulus of 140 GPa.

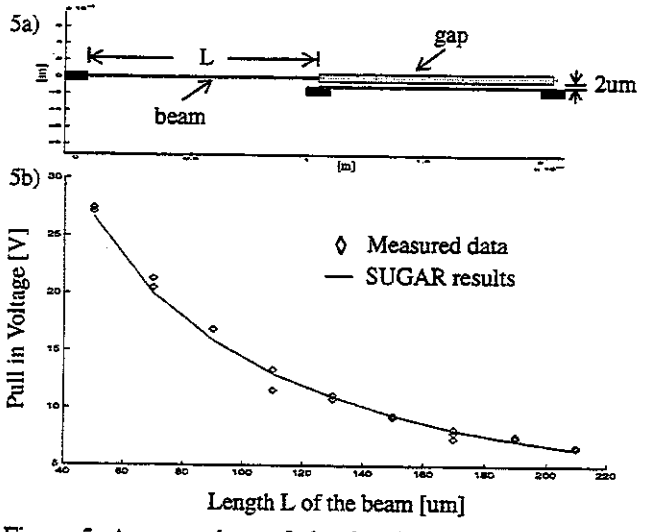

Figure 5: A comparison of simulated pull-in voltages with experimental data.

ALGORITHMS AND RESULTS

This section describes the algorithms implemented in SUGAR. The DC algorithm deals with finding an the equilibrium state of a MEMS device when constant mechanical forces or voltages are applied. The steady state algorithm takes care of the response of a system subjected to damping

and sinusoidal excitation forces. Finally the transient analysis algorithm uncovers the instantaneous state of a system as a function of time.

DC Algorithm

In DC analysis, the equilibrium position due to constant mechanical forces and voltages are calculated according to

$$[K]\{q\} - \{F\} = 0 . \quad (28)$$

Since the electrostatic force and possibly the stiffness are nonlinear functions of displacement, numerical methods are needed to solve Eq. (28).

In general, there is no recognized "best" way for solving nonlinear equations. However, some methods are effective in finding the roots provided we know the approximate locations of those roots. SUGAR uses the Newton-Raphson Method.

Eq. (28) is a special case of the general form

$$f(\{q\}) = 0 \quad (29)$$

First, an initial guess $\{q_0\}$ is taken which is sufficiently near a root. In general, the Newton-Raphson method approaches the solution by the iteration

$$\{q_{n+1}\} = \{q_n\} - [f'(\{q_n\})]^{-1}\{f(\{q_n\})\} \quad (30)$$

where $[f'(\{q_n\})]$ is the system Jacobian matrix. The iteration will proceed until

$$\|\{q_{n+1}\} - \{q_n\}\| < \zeta \quad (31)$$

where ζ is the tolerance.

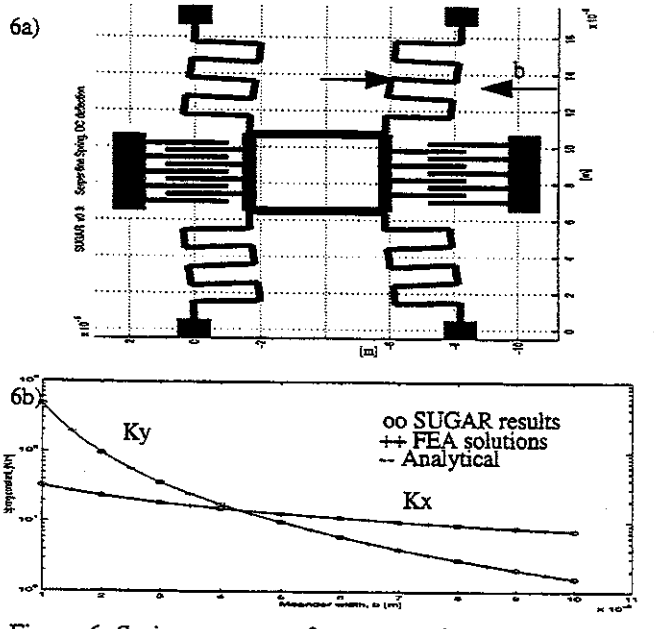

Figure 6. Spring constants for a serpentine spring structure (Fedder [12]). The graph shows stiffness in the x and y directions as a function of meander length b.

DC simulation of common flexural supports indicates that SUGAR can accurately model these structures. SUGAR analysis of serpentine suspensions (Figure 6) is identical to FEA simulations done by Fedder [12]. SUGAR also agrees with Fedder's analytical and measured data. Simulation of crab leg suspensions from [12] is equally accurate.

Steady State Algorithm

In steady state analysis the following equation is solved

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{a_i \cos(\omega t + \beta_i)\} \quad (32)$$

where $\{a_i \cos(\omega t + \beta_i)\}$ is the sinusoidal external excitation.

The solution to this equation is the real part of the following complex equation:

$$[M] \cdot \{\ddot{z}\} + [C] \cdot \{\dot{z}\} + [K] \cdot \{z\} = \{B_i\} \cdot e^{j\omega t} \quad (33)$$

where $\{B_i\} = \{a_i(\cos\beta_i + j\sin\beta_i)\}$. A particular solution of Eq. (33) is of the following form

$$\{z\} = \{V\} \cdot e^{j\omega t} \quad (34)$$

where $\{V\}$, a complex vector, contains the magnitude and phase information of the system response. Substituting Eq. (34) into Eq. (33) gives

$$(-\omega^2[M] + j\omega[C] + [K]) \cdot \{V\} = \{B\} . \quad (35)$$

Once Eq. (35) is solved, the magnitude and phase response for each node is evaluated. The steady state vibration of the structure can be animated.

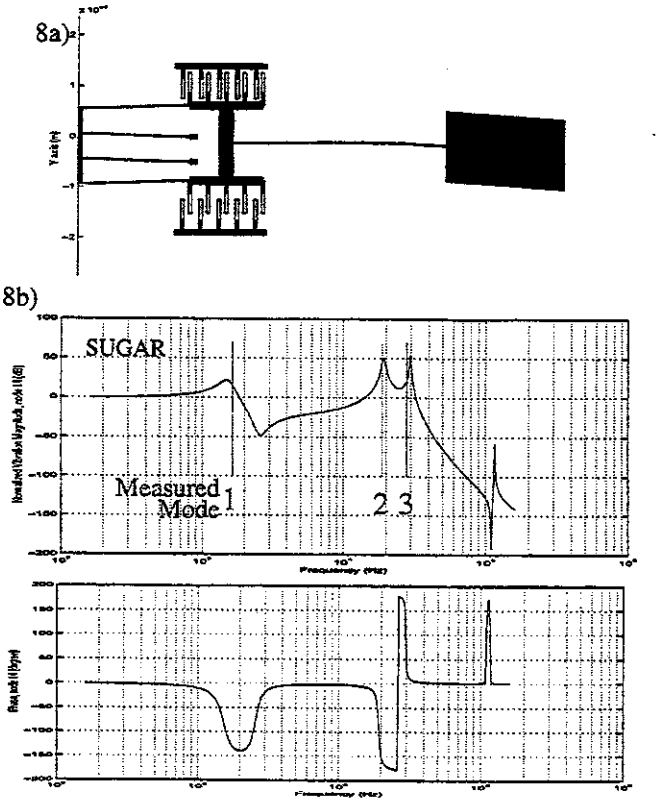

Figure 8. Simulations of the linear-drive multi-mode resonator showing the Bode magnitude and phase plot of the displacement at the base of the semaphore mass as a function of driving frequency.

SUGAR simulations of the multi-mode resonators reported by Brennan et al. [13] show the mode shapes and

Bode plot of the semaphore mass displacement (Figure 8a and 8b). These simulations use a simple Couette flow model for damping under all moving structures. Simulations of the first three modes agree with Brennen's experimental frequency data to within 5%.

Transient Algorithm

For transient behavior, we simulate nodal displacements of the system as functions of time by ordinary differential equation solvers such as (Runge-Kutta) ode45 **[15]** and central difference methods **[14, 7]** such as the summed form and Wilson-θ.

The dynamical equation of motion for a general system may be described by a form similar to Eq. (21) where the matrices may be functions of position, velocity, and time.

The central difference method offers the least complexity. The crux of the main loop is similar to

$$\begin{aligned} \{\ddot{q}\}_t &= [M]^{-1}(\{F(q,\dot{q},t)\} - [C]\{\dot{q}\}_t - [K(q)]\{q\}_t) \\ \{\dot{q}\}_{t+0.5\Delta t} &= \{\dot{q}\}_{t-0.5\Delta t} + \Delta t\{\ddot{q}\}_t \\ \{q\}_{t+\Delta t} &= \{q\}_t + \Delta t\{\dot{q}\}_{t+0.5\Delta t} \end{aligned} \qquad (36)$$

where $\Delta t < \Delta t_{crit}$ is the critical time step for stability **[14]**. For systems that are vastly nonlinear, the time step Δt may need to be adaptive.

The instantaneous nodal positions for the entire system are calculated by integrating the displacements (solutions) of Eq. (21) from the given initial conditions at time t_i to a final time t_f where *[F], [M], [C],* and *[K]* are continuously evaluated and updated throughout the simulation.

CONCLUSION

A practical MEMS simulation program for planar devices has been demonstrated. The algorithms are not as yet optimized for speed as commercially available software is (i.e. ABACUS, SABER). However, SUGAR demonstrates efficient methods for handling nonlinearities due to electrostatics and stiffness for many types of planar devices. Another appealing benefit is that the software is portable across all Unix, PC, and Macintosh platforms; the minimum requirement being Matlab (version 5 or greater). Because the program runs in a Matlab environment it is simple to use and easy to modify. Most importantly, SUGAR gives results that are in agreement with traditional finite element simulation, analytical models, and experimental data. The code for SUGAR v0.5 is available on the web at http://www-bsac.eecs.berkeley.edu.

ACKNOWLEDGEMENTS

Thanks to Dr. A. Parameswaran for helpful review, E. Kruglick for electrostatic pull-in data, and Dr. S. Crary for discussions on fringing fields.

This work is supported in part by the DARPA Composite CAD Program, the National Science Foundation, and a UC Berkeley GP4 Fellowship.

REFERENCES

1. Matlab, *Matlab High-Performance Numeric Computation and Visualization Software Reference Guide*, The Math Works Inc., 24 Prime Park Way Natick, Mass, (1992).
2. ABAQUS, Hibbitt, Karlsson & Sorensen, Inc. Pawtucket, IR, USA.
3. G. M. Koppelman, "Oyster, A Three-dimensional Structural Simulator for Microelectromechanical Design", *Sensors and Actuators, 20*, (1989), p179-185.
4. T. J. Hubbard, E. K. Antonsson, "Design of MEMS via Efficient Simulation of Fabrication", *Proc. of the 1996 ASME Design Engineering Conf.*, August 18-22, (1996), Irvine CA.
5. S. Crary, Y. Zhang, "Software Tools for Designers of Sensor and Actuator CAE Systems", *Solid-State Sensors and Actuators*, San Francisco, CA, June 23-27, (1991), pp. 498-501.
6. J. R. Gilbert, et. al., "Implementation of a MEMCAD System for Electrostatic and Mechanical Analysis of Complex Structures from Mask Descriptions", *IEEE Micro Electro Mechanical Systems Workshop*, (1993), Fort Lauderdale USA, p 207-212.
7. L. W. Nagel, "Computer analysis of nonlinear circuits excluding radiation", Ph. D. dissertation, UC Berkeley (1970).
8. T. Y. Yang, "Finite Element Structural Analysis", Prentice-Hall, Inc (1985).
9. M. Paz, "Structural Dynamics Theory and Computation", Litton Education Publishing, Inc (1980).
10. R. Wen, J. Rahimzadeh, "Nonlinear Elastic Frame Analysis by Finite Element", *Journal of Structural Engineering*, Vol 109, No. 8, August (1983), pp 1952-1971.
11. M. R. Boyd, S. B. Crary, M. D. Giles, "A Heuristic Approach to the Electromechanical Modeling of MEMS Beams", *Solid-State Sensor and Actuator Workshop*, Hilton Head, South Carolina, June 13-16, (1994), pp123-126.
12. G. Fedder, "Simulations of Microelectromechanical Systems", Ph.D. dissertation, UC Berkeley, (1994), p 104-115.
13. R. Brennen, A. Pisano, Tang, "Multiple Mode Micromechanical Resonators", *Proc. IEEE Micro Electro Mechanical Systems Workshop* (1990), Napa, CA USA.
14. pp. 9-13.
    R. R. Craig, Jr., Structural Dynamics: An Introduction to Computer Methods", John Wiley & Sons, Inc. New York (1981).