

MEMS SIMULATION USING SUGAR v0.5

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

Berkeley Sensor and Actuator Center

University of California at Berkeley

497 Cory Hall, Berkeley, CA 94720

OVERVIEW

CONTEXT

Integrated Circuit (IC) fabrication technology development over the last few decades has been phenomenal. We have also seen the development of design and simulation tools which enable engineers to capture the full capabilities of the technology. MEMS fabrication technology has leveraged a huge fraction of the IC fabrication technology, but virtually none of the CAD support that goes with it. As a result, we can all envision MEMS which could be fabricated if they were designed, but which can not be designed due to the lack of support tools.

Rectifying this disparity in the design and fabrication technology is the subject of active research in the MEMS CAD community. To date, MEMS design tools have focussed mostly on process modeling [1, 2] and Finite Element and Boundary Element Analysis (FEA and BEA) for electromechanical functional modeling [3, 4]. This work has led to several products currently on the market [5, 6, 7]. These tools generally take as input a process flow and mask geometry from which they generate a solid model. The solid model is then meshed and simulated.

All of these approaches are geared toward process and device development, and they are generally analysis oriented rather than design oriented. While these activities will continue to be critical to the success of MEMS, there is a growing community of designers who need support at a higher level than device simulation. These users are designing systems with dozens or even thousands of components, far beyond the reach of FEA/BEA based tools.

In an attempt to leverage some of the CAD technology developed for the IC industry, several groups are now working toward a VLSI inspired approach to MEMS design, with an emphasis on hierarchical design, system simulation, and layout synthesis [8, 9]. Initial success in this area includes the synthesis of 1D resonators [10], optimization-based synthesis of 2D resonators [11], nodal analysis of multiple degree of freedom structures in Saber [12] and inertial sensors [13]. Some of this work has also become commercially available [14].

The success of this approach is critically dependent on the ability to quickly and accurately simulate large numbers of interconnected MEMS devices. In the IC environment, this is done by abstracting the physical world in terms of 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 [15].

A similar approach can also be used to abstract the behavior of MEMS devices so that MEMS can also be quickly and accurately simulated. This is a critical capability in the increasingly important framework of hierarchical design, simulation, and synthesis for MEMS.

SUGAR is a collection of Matlab [24] routines which implements a nodal analysis approach to MEMS simulation. A wide variety of planar electromechanical systems can be simulated. The user provides a text file describing the geometry and connectivity of the system, and then calls one or more analysis routines to determine static displacement, transient response, etc. Graphical results are displayed using Matlab function calls, and numerical results are available for additional Matlab processing if desired.

INPUT

The user provides a text file that describes the building blocks in terms of geometry, voltage, or applied force. The format for a typical input line is block type, node list, geometry list, and parameter list. Currently, block types for planar beams (b), anchors (a), electrostatic gaps (g), and force generators (f) have been defined. An example of a netlist is shown in Figure 1.

```
%Substrate anchor
a 1 2 5e-6 180 10e-6
%Horizontal beam: 34x2 microns
b 1 3 34e-6 0 2e-6
%Vertical beam: 70x2 microns
b 3 4 70e-6 -90 2e-6
%Electrostatic actuator, and the rest...
g 4 5 6 7 50e-6 0 4e-6 4e-6 10e-6 445
b 7 8 70e-6 0 2e-6
a 8 9 5e-6 0 10e-6
```

Figure 1. Input netlist. For example the fourth line describes a beam that extends from node 1 to 3 for a length of $34\mu\text{m}$, at an angle of 0 degrees from the horizontal, and the beam's width is $2\mu\text{m}$.

The layer thickness and Young's modulus are defined in a separate process file. Using the information provided by the netlist and process file, SUGAR creates individual stiffness $[k]_s$, mass $[m]_s$, and damping $[c]_s$ matrices for each structural block s . Since each structure in the netlist may have differing orientations, all local coordinates are transformed into global coordinates. The assemblage of the set of individual matrices into collective system matrices, $[K]$, $[M]$, and $[C]$, where all structures are coupled at common nodes, is accomplished by superposition. Element indices in the system matrix directly correspond to nodal coordinate components. Structures with common nodes have additive contributions to that particular matrix element. Therefore the size of the entire system matrices only depend on the number of distinct nodes. The equation of motion describing the dynamics of the entire system can be expressed in a familiar form [16]

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{F\} \quad (1)$$

where the nodal displacement $\{q\} = \{x_1 y_1 \theta_1 \dots x_N y_N \theta_N\}$ is a $1 \times 3N$ column vector and N is the number of nodes.

ANALYSIS ROUTINES

The analysis of planar MEMS devices in SUGAR is accomplished by a collection of Matlab functions. This includes DC analysis, steady-state analysis, and transient analysis solvers.

Invoking the command $dq = cho_dc(netlist)$, where *netlist* is a string containing the name of the file, will solve for an equilibrium state for the system (not necessarily unique) and assign nodal displacements $\{\Delta x_n \Delta y_n \Delta \theta_n\}$ to $\{dq\}$ for each node *n*. The $cho_dc_display(netlist, dq)$ function displays the resulting structure (Figure 2).

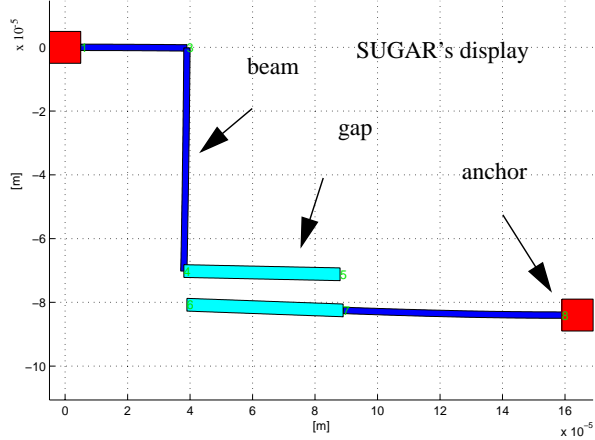


Figure 2. Output display of a DC solution from the netlist in Figure 1. It shows bending due to a voltage applied across the electrostatic actuator (gap).

Resonant mode calculation and display is accomplished using the $cho_mode(netlist, mode, scale)$ function, where *mode* is the mode number and *scale* magnifies the amplitudes of the mode shapes in the display.

$Xw = cho_ss_calc(netlist, W)$ computes the steady state response for an applied sinusoidal set of forces defined in *netlist*.

Animation of the steady state response is accomplished by the $cho_ss_animation(netlistfile, \omega)$ function, where ω is the angular frequency vector.

The $cho_bodeplot(\omega, magnitude, phase)$ function calculates and displays a Bode plot.

Transient analysis is provided by $cho_ta(time\ interval, initial\ conditions, netlist, parameter\ list)$ function, which simulates the system dynamics for a system for a given initial condition and time varying force.

MODELS

The performance of many planar MEMS devices can be modeled with beams and electrostatic gaps. Linear and nonlinear beam models are implemented for small to large deflections, and the nonlinear behavior in the gap model is essential for accurate simulation of electrostatic actuation. As with SPICE, there are different model levels in SUGAR which allow the user to trade off accuracy and speed.

LINEAR BEAM MODEL

Any two coupled beams in SUGAR 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 3 degrees of freedom at each node (Figure 3).

In the region between nodes, the equation of equilibrium for a beam is

$$\frac{\partial^4 v(x)}{\partial x^4} = 0 \quad (2)$$

[17]. Solving this equation for $v(x)$ and imposing the boundary conditions at both end nodes yield 4 equations and 4 unknowns. By solving for the coefficients of $v(x)$ and $\theta(x)$ in terms of v_n and θ_n and grouping like-terms results in

$$v(x) = H_1(x)v_1 + H_2(x)\theta_1 + H_3(x)v_2 + H_4(x)\theta_2 \quad (3)$$

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

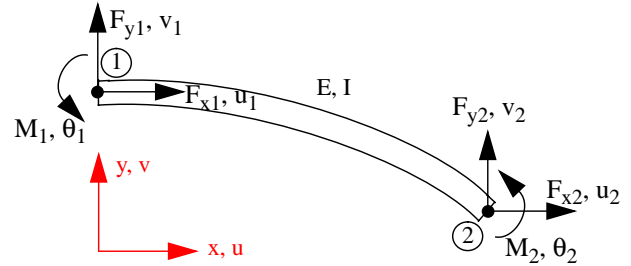


Figure 3. A bent beam showing nodal forces, moments, and coordinates.

The stiffness for this beam can be obtained by using Castiglianos theorem

$$F_i = \frac{\partial S}{\partial q_i} \quad (4)$$

where F_i is the force or moment, and q_i is the displacement coordinate for degree of freedom $i = 1, 2, 3$ for $q = x, y, \theta$ respectively.

For the linear case, the strain energy S for a beam with uniform cross section is

$$S = \frac{EI}{2} \int_0^L \left(\frac{\partial^2 v(x)}{\partial x^2} \right)^2 dx + \frac{EA}{2} \int_0^L \left(\frac{\partial u(x)}{\partial x} \right)^2 dx \quad (5)$$

The substitution of Eq. (5) into Eq.(4) yields the stiffness

$$k_{ij} = EI \int_0^L H''_i(x) H''_j(x) dx + EA \int_0^L H'_i(x) H'_j(x) dx \quad (6)$$

which gives

$$[k]_s = \frac{E}{L^3} \begin{bmatrix} AL^2 & 0 & 0 & -AL^2 & 0 & 0 \\ 0 & 12I & 12EI & 0 & -12I & 6IL \\ 0 & 6IL & 6EIL & 0 & -6IL & 2IL \\ -EAL^2 & 0 & 0 & AL^2 & 0 & 0 \\ 0 & -12I & -12EI & 0 & 12I & -6IL \\ 0 & 6IL & 6EIL & 0 & -6IL & 4IL \end{bmatrix} \quad (7)$$

A mass matrix is found by equating internal and external work due to virtual displacements [16]. This results in

$$m_{ij} = \int_0^L \rho(x) A(x) H_i(x) H_j(x) dx \quad (8)$$

where $\rho(x)$ and $A(x)$ are the density and cross sectional area at position x along the beam. If ρA is constant we have

$$[m]_s = \frac{\rho AL}{420} \begin{bmatrix} 140 & 0 & 0 & 70 & 0 & 0 \\ 0 & 156 & 22L & 0 & 54 & -13L \\ 0 & 22L & 4L^2 & 0 & 13L & -3L^2 \\ 70 & 0 & 0 & 140 & 0 & 0 \\ 0 & 54 & 13L & 0 & 156 & -22L \\ 0 & -13L & -3L^2 & 0 & -22L & 4L^2 \end{bmatrix} \quad (9)$$

where ρ is the density and A is the cross sectional area of the beam.

Using a simple Couette flow model we approximate the damping matrix (similar to Eq. (8)) as

$$[c]_s = \frac{\mu L w}{420 \Delta} \begin{bmatrix} 140 & 0 & 0 & 70 & 0 & 0 \\ 0 & 156 & 22L & 0 & 54 & -13L \\ 0 & 22L & 4L^2 & 0 & 13L & -3L^2 \\ 70 & 0 & 0 & 140 & 0 & 0 \\ 0 & 54 & 13L & 0 & 156 & -22L \\ 0 & -13L & -3L^2 & 0 & -22L & 4L^2 \end{bmatrix} \quad (10)$$

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

NONLINEAR GAP MODEL

For a parallel electrostatic gap where the length of the beams L is much greater than the gap spacing d , and the layer thickness $h \sim d$, the total force on each beam is

$$F = \frac{-1}{2} \epsilon_0 V^2 L h \frac{\alpha(d/h)}{d^2} \quad (11)$$

where the factor $\alpha > 1$ is the fringing field contribution [19].

As described previously, transverse displacements along the length of the beam can be described by nodal coordinates and Hermitian shape functions. Referring to Figure 4,

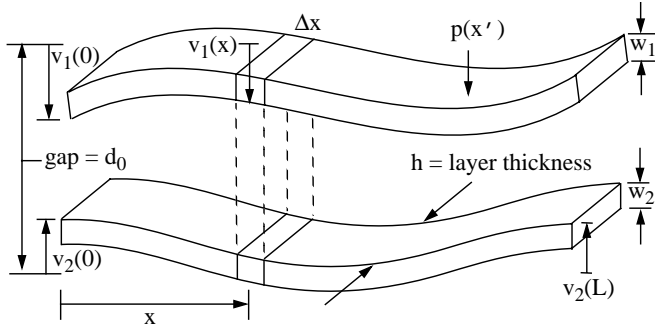


Figure 4. The level-2 model of an electrostatic actuator. There is a distributed electrostatic force $p(x)$ along the length of the beam which varies due to bending.

the distance between the beams at position x is

$$d(x) = d_0 + v_1(x) - v_2(x) \quad (12)$$

where d_0 is the original gap spacing, and $v_1(x)$ and $v_2(x)$ are the transverse displacements Eq. (3) of the beams at position x . The nodal forces due to a distributed force per unit length are

$$F_i = \int_0^L p(x) H_i(x) dx. \quad (13)$$

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

$$p(x) = \frac{-1}{2} \epsilon_0 V^2 h \frac{\alpha(d(x)/h)}{d(x)^2}. \quad (14)$$

In solving the integral in Eq. (13) for a general analytical expression the equivalent nodal forces and moments on each beam is 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. (1) for further analysis. The level-1 gap model is similar to level-2 except the bending and fringing field contributions are not considered.

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. (12). 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 $|F_c| \gg |F_e|$ when $d < d_c$ and $|F_c| \ll |F_e|$ 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.

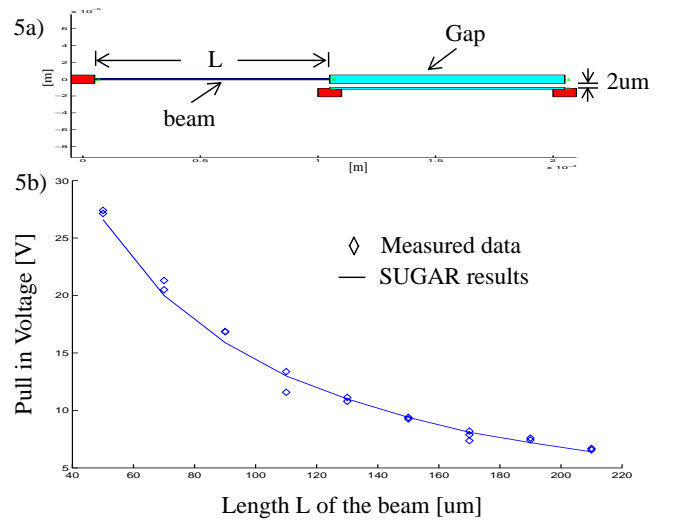


Figure 5: A comparison of simulated pull-in voltages with experimental data.

NONLINEAR BEAM MODEL

In the linear beam model, $\{q_i\}$ is proportional to $\{F_i\}$. This model is satisfactory for small deflections; however, there are nonlinear effects when deflections are large. For example, there may be changes in the projected length due to bending and changes in stiffness due to axial forces.

A level-2 (nonlinear) beam model was developed for the small rotation condition. Only the geometric nonlinearity is considered. This model starts from a linear interpolation for the longitudinal displacement and a cubic interpolation for the lateral displacement

$$u = u_1 + \frac{(u_2 - u_1)x}{L} \quad (15)$$

$$v = v_1 + \theta_1 x + \frac{(-2\theta_1 - \theta_2 + 3\theta_0)x^2}{L} + \frac{(\theta_1 + \theta_2 - 2\theta_0)x^3}{L^2} \quad (16)$$

where $\{u_1 \ v_1 \ q_1\}$ and $\{u_2 \ v_2 \ q_2\}$ are the nodal displacement vectors (Figure 3), and $\theta_0 = (v_2 - v_1)/L$.

The strain at every point on the beam can be calculated from the displacements [25]

$$\tilde{\epsilon}(x, \eta) = \frac{du}{dx} + \frac{1}{L} \cdot \int_0^L \left(\frac{1}{2} \cdot \left(\frac{dv}{dx} \right)^2 \right) dx + \eta \cdot \frac{d^2v}{dx^2} \quad (17)$$

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.

Substituting Eq. (17) into the following gives the strain energy

$$S = \int_V \frac{1}{2} E \tilde{\epsilon}^2 dV. \quad (18)$$

Then substituting Eq. (18) into Eq. (4) yields the nodal forces as a function of displacements. This model takes into account the contribution of the longitudinal displacement to the strain. Figure 6a shows a simulation of a clamped-clamped beam with a central concentrated load. Simulation of the load-deflection response (Figure 6b) is in close agreement with Abaqus [18] over a typical range of operation.

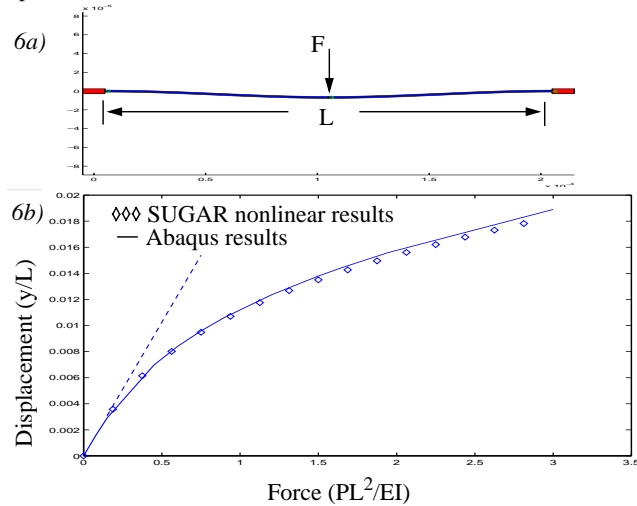


Figure 6. Nonlinear deflection. This shows the load-deflection curve for the clamped-clamped beam with a concentrated load.

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

$$[K]\{q\} - \{F\} = 0. \quad (19)$$

Since the electrostatic force and possibly the stiffness are nonlinear functions of displacement, numerical methods are needed to solve Eq. (19).

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. (19) is a special case of the general form

$$f(\{q\}) = 0 \quad (20)$$

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

$$\{q_{n+1}\} = \{q_n\} - [f'(\{q_n\})]^{-1} \{f(\{q_n\})\} \quad (21)$$

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

$$\|\{q_{n+1}\} - \{q_n\}\| < \zeta \quad (22)$$

where ζ is the tolerance.

DC simulation of common flexural supports indicates that SUGAR can accurately model these structures. SUGAR analysis of a folded flexure suspension (Figure 7) agrees exactly with the Abaqus finite element simulations done by Judy [20]. Similarly, SUGAR analysis of serpentine suspensions (Figure 8) is identical to FEA simulations done by Fedder [21]. SUGAR also agrees with Fedder’s analytical and measured data. Simulation of crab leg suspensions from [21] is equally accurate.

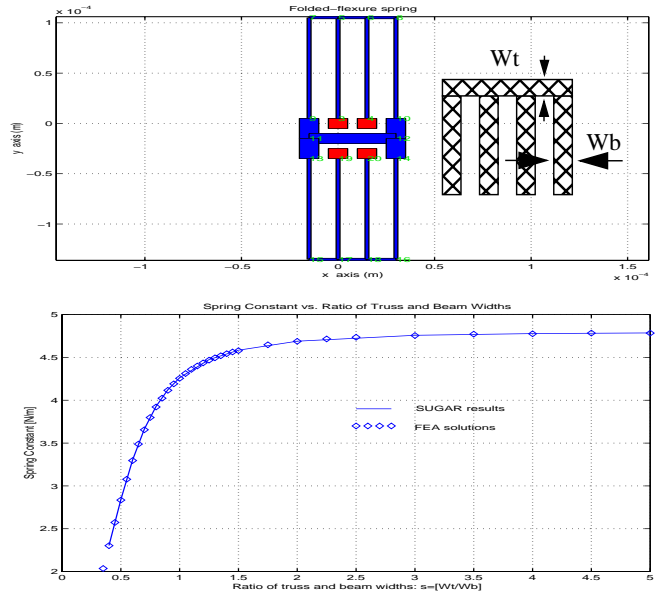


Figure 7: Comparison of SUGAR results and FEA solution for the lateral spring constant of a folded-flexure (from Judy [20]) versus the ratio of the truss width W_t and beam width W_b .

STEADY STATE ALGORITHM

In steady state analysis the following equation is solved

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{a_i \cos(\omega t + \beta_i)\} \quad (23)$$

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:

$$[M] \cdot \{\ddot{z}\} + [C] \cdot \{\dot{z}\} + [K] \cdot \{z\} = \{B_i\} \cdot e^{j\omega t} \quad (24)$$

where $\{B_i\} = \{a_i(\cos\beta_i + j\sin\beta_i)\}$. A particular solution of equation (24) is of the following form

$$\{z\} = \{V\} \cdot e^{j\omega t} \quad (25)$$

where $\{V\}$, a complex vector, contains the magnitude and phase information of the system response. Substituting Eq. (25) into Eq. (24) gives

$$(-\omega^2[M] + j\omega[C] + [K]) \cdot \{V\} = \{B\} \quad (26)$$

Once Eq. (26) is solved, the magnitude and phase response for each node is evaluated. The steady state vibration of the structure can be animated.

SUGAR simulations of the multi-mode resonators reported by Brennan et al. [22] show the mode shapes and Bode plot of the semaphore mass displacement (Figure 9 and 10). These simulations use a simple Couette flow model for damping under all moving structures. Simulations of the first three modes agree with Brennan's experimental frequency data to within 5%.

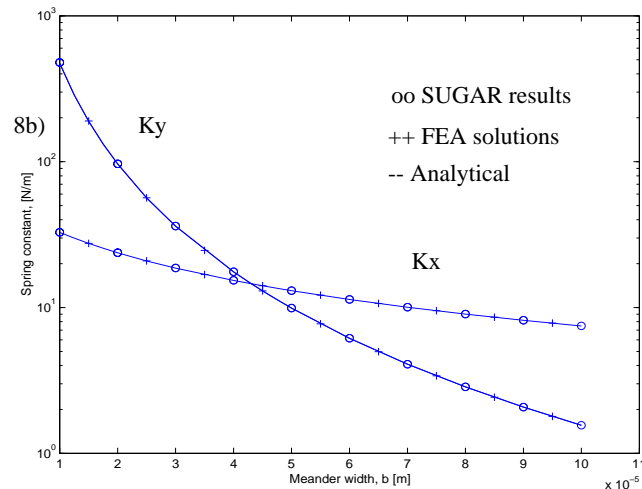
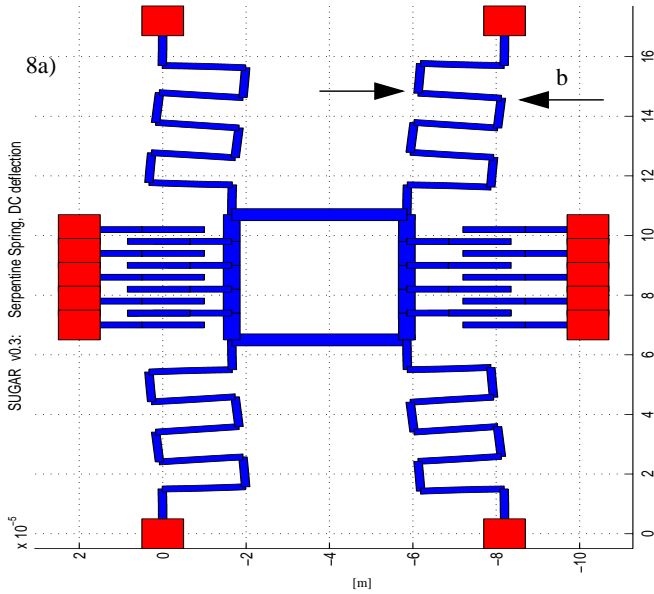


Figure 8: Horizontal and vertical spring constants for a serpentine spring structure (Fedder [21]). The graph shows stiffness in the x and y directions as a function of meander length b .

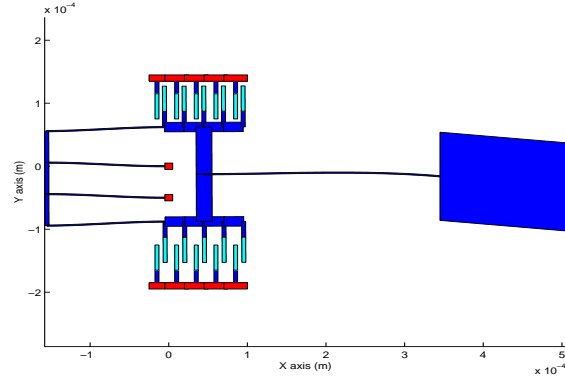


Figure 9: Simulations of the linear-drive multi-mode resonator structure of Brennan [22] shows the second resonant mode shape at 19.177 kHz.

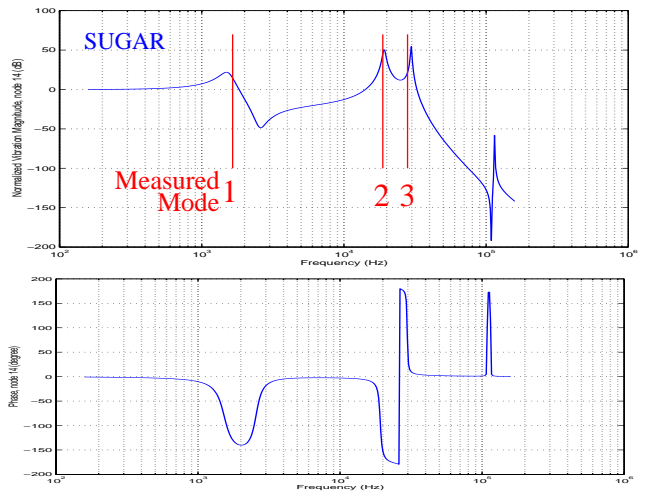


Figure 10: Simulations of the linear-drive multi-mode resonator showing the Bode magnitude and phase plot of the displacement at the base of the semaphore mass as a function of driving frequency. Experimental data from [22] match the predicted three modes to within 5%.

TRANSIENT ANALYSIS

For transient behavior, we simulate nodal displacements of the system as functions of time by ordinary differential equation solvers such as (Runge-Kutta) ode45 [24] and central difference methods [23, 16] 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 (1) 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

$$\{\ddot{q}\}_t = [M]^{-1}(\{F(q, \dot{q}, t)\} - [C]\{\dot{q}\}_t - [K(q)]\{q\}_t) \quad (27)$$

$$\{\dot{q}\}_{t+0.5\Delta t} = \{\dot{q}\}_{t-0.5\Delta t} + \Delta t\{\ddot{q}\}_t \quad (28)$$

$$\{q\}_{t+\Delta t} = \{q\}_t + \Delta t\{\dot{q}\}_{t+0.5\Delta t} \quad (29)$$

where $\Delta t < \Delta t_{crit}$ is the critical time step for stability [23].

The instantaneous nodal positions for the entire system are calculated by integrating the displacements (solutions) of Eq. (1) 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. An example is shown in Figure 12.

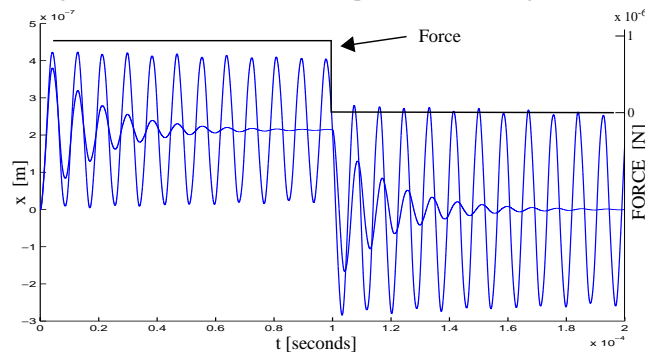


Figure 12: Transient simulations of the resonator from Figure 7. The top trace show the applied force ($1 \mu\text{N}$ for $100 \mu\text{s}$). The bottom trace shows the simulated response in air and water.

CONCLUSION

A simulation program for the nodal analysis of many types planar MEMS devices has been demonstrated. SUGAR is simple to use, executes quickly, and gives results that are in agreement with traditional finite element simulation, analytical models, and experimental data. SUGAR is portable across all Unix, PC, and Macintosh platforms running Matlab (version 5 or greater). The free software is available on the web at <http://www-bsac.eecs.berkeley.edu>.

ACKNOWLEDGEMENTS

Thanks to A. Parameswaran for helpful review and thanks to E. Kruglick for electrostatic pull-in data.

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

BIBLIOGRAPHY

- G. M. Koppelman, "A Three-dimensional Structural Simulator for Microelectromechanical Design", *Sensors and Actuators*, 20, (1989), p179-185.
- 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.
- 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.
- 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.
- MEMCAD version 3., Microcosm, 201 Willesden Drive, Cary, NC 27513.
- Y. He, R. Harris, G. Napadensky, and F. Maseeh, "IntelliCAD: A virtual prototype manufacturing software system for MEMS," *Proc. IEEE Micro Electro Mechanical Systems Workshop*, San Diego, CA, February (1996).
- DRBEM, Coyote Systems, 2740 Van Ness Ave #210, San Francisco, CA 94109.
- N. Lo, E. Berg, S. Quakkelaar, J. Simon, M. Tachiki, H. Lee, K. Pister, "Parameterized Layout Synthesis, Extraction, and SPICE Simulation for MEMS", *Intl. Symp. on Circuits and Systems*, Atlanta, GA, May (1996).
- T. Mukherjee, and G. K. Fedder, "Structured Design of Micro-electromechanical systems", *DAC '97*, Anaheim, CA, June 9-13, (1997).
- E.C. Berg, N.R. Lo, J.N. Simon, H.J. Lee, K.S.J. Pister, "Synthesis and Simulation for MEMS Design", *ACM SIGDA Physical Design Workshop*, Reston, VA, April (1996), pp. 67-70.
- S. Iyer, T. Mukherjee, and G. K. Fedder, "Automated Optimal Synthesis of Microresonators," *Solid-State Sensors and Actuators*, Chicago, IL, June 12-19, (1997).
- J. Vandemeer, M. Kranz, G. Fedder, "Nodal Simulation of Suspended MEMS Multiple Degrees of Freedom", *DSC-Vol. 62 MEMS ASME* (1997), pp 113-118.
- J.E. Vandemeer, M. S. Kranz, and G. K. Fedder, "Hierarchical Representation and Simulation of Micromachined Inertial Sensors", *Modeling and Simulation of Microsystems*, Santa Clara, CA, April 6-8, (1998).
- M. Maher and H. Lee, "MEMS Systems Design and Verification Tools", *SPIE 5th Annual Int. Symp. on Smart Structures and Materials*. March 1-5 (1998). San Diego, CA.
- L. W. Nagel, "Computer analysis of nonlinear circuits excluding radiation", Ph. D. dissertation, UC Berkeley (1970).
- M. Paz, "Structural Dynamics Theory and Computation", Litton Education Publishing, Inc (1980).
- T. Y. Yang, "Finite Element Structural Analysis", Prentice-Hall, Inc (1985).
- ABAQUS, Hibbitt, Karlsson & Sorensen, Inc. Pawtucket, IR, USA.
- 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.
- M. Judy, "Micromechanisms Using Sidewall Beams", Ph.D. dissertation, UC Berkeley, (1994), p 62-67.
- G. Fedder, "Simulations of Microelectromechanical Systems", Ph.D. dissertation, UC Berkeley, (1994), p 104-115.
- R. Brennen, A. Pisano, Tang, "Multiple Mode Micromechanical Resonators", *Proc. IEEE Micro Electro Mechanical Systems Workshop* (1990).
- R. R. Craig, Jr., *Structural Dynamics: An Introduction to Computer Methods*, John Wiley & Sons, Inc. New York (1981).
- Matlab, *Matlab High-Performance Numeric Computation and Visualization Software Reference Guide*, The Math Works Inc., 24 Prime Park Way Natick, Mass, (1992).
- R. Wen, J. Rahimzadeh, "Nonlinear Elastic Frame Analysis by Finite Element", *Journal of Structural Engineering*, Vol 109, No. 8, August (1983), pp 1952-1971.