Schematic-Based Lumped Parameterized Behavioral Modeling for Suspended MEMS

Qi Jing†, Tamal Mukherjee† and Gary K. Fedder†*
†Department of Electrical and Computer Engineering and *The Robotics Institute
Carnegie Mellon University, Pittsburgh, PA 15213-3890 USA

Abstract

Schematic-based lumped parameterized behavioral modeling and simulation methodologies have become available since the emergence of analog HDLs. They greatly ease iterative hierarchical multi-domain simulation, which is critical to the design of MEMS. NODAS is one of such tools, with models written in VerilogA and simulation performed within the Cadence framework.

This paper focuses on several key modeling issues in NODAS, including schematic representation, element communication, linear, nonlinear and multi-domain modeling, and extensibility to new physical effects, processes and physical domains. A nonlinear beam model and an electrostatic gap model are discussed as examples. Simulation comparison to finite element analyses and experimental data verifies the accuracy of the models and validates the simulation methodology.

Keywords*: MEMS, schematic-based, lumped, parameterized, behavioral, nonlinear beam, electrostatic gap*

I. Introduction

Suspended MEMS

MicroElectroMechanical Systems (MEMS) include various process technologies, each spanning a large design space. Suspended MEMS, in which all the movable parts of a device are suspended by springs attached to fixed anchor points, covers a wide range of applications including accelerometers, micromirrors, resonator filters, and RF switches. Our discussion will be restricted to the design space of suspended MEMS, with mechanical and electrostatic effects as the focus, for two reasons. First, many of these devices have been commercialized. Secondly, unlike other emerging MEMS areas such as microfluidics and bioMEMS, there is a tight integration with electronics.

Similar to CAD for integrated circuits, a performance evaluation method for suspended-MEMS should have the characteristics of good accuracy, fast speed, ease of iterative evaluation and the ability to handle complexity. In addition, as the interactions between multiple physical domains, for instance, the mechanical domain and the electrical domain, are critical to the performance of MEMS, CAD tools for MEMS are also expected to be able to handle mixed-domain simulation, especially the co-simulation with electrical circuits.

Existing Simulation Methodologies

Finite and boundary element analyses (FEA/BEA) are the most commonly used methods for numerical mechanical and electrostatic simulations. Commercial tools commonly used by the MEMS design community include ANSYS [1], ABAQUS [2], Maxwell [3], and Coventorware [4]. These methods are accurate for fine meshes. However, as they are layout-based, any change to the geometric sizes requires a new mesh, leading to inconvenient design iteration. Continuum simulation has been extended to enable multi-physics simulation, by using self-consistent convergence solutions or by creating a composite system matrix for both mechanical and electrostatic FEA. However, such solutions still do not include the interactions between transducers and transistor-level interface circuits.

Reduced-order modeling speeds up system simulation and meets the co-simulation requirement [1][5]. Macromodels for the entire system are based on prioritized parameters extracted from a batch of FEA simulations, then instantiated in analog Hardware Description Languages (AHDL). The AHDL models enable co-simulation with circuits. The drawback is that as the models are based on a prioritized motion shape subset, they are only valid for specific structures. If there is any change to the geometric sizes or topologies, new models will have to be recreated from the beginning.

Circuit-level behavioral nodal simulation has been proposed to meet the need for ease of iterative design evaluation. Existing circuit-level representations for suspended MEMS include MEMSPro [6], ARCHITECT [7], SUGAR [8], MEMSMaster [9] and NODAS [10]. In these methods, systems are represented from structure using atomic elements such as anchors and beams. The system matrices formed are much smaller than those from FEA. Each atomic element has a lumped behavioral model with geometric parameters which can be specified individually. The geometric parameterization simplifies the evaluation of changes in size on the device performance in each design iteration. The models are implemented in analog HDLs or directly in element matrices, both inherently supporting simulations in mixed physical domains. The major distinctions among these tools are the representation styles, the design frameworks and the availability of specific model libraries, leading to varied capability of handling co-simulation and other important design steps such as layout editing and extraction.

This paper presents the modeling and simulation methodology in NODAS, including schematic representation, element communication, linear, nonlinear and multi-domain modeling. The discussion is followed by simulation examples of a CMOS-MEMS bandpass filter and a RF-MEMS switch, and a section about the extensibility of the methodology.

II. Modeling and Simulation in NODAS

The modeling and simulation methodology in NODAS is based on the composition property of suspended-MEMS: a large variety of suspended MEMS devices, despite different topology and complexity, can be structured from a small set of atomic ele-

0-7803-7607-2/02/$17.00

ments, including anchors, beams, plates, and electrostatic gaps. Each element has a lumped behavioral model, a symbol, a set of process-dependent parameters, and a set of process-independent geometric design parameters.

The models in NODAS are written in a mixed-domain analog HDL, VerilogA [11], for simulation in Spectre [12]. Design with NODAS starts from schematic entry, where MEMS and circuit elements, such as beams and transistors, can be wired together. A composite netlist for the entire system is generated and sent to the circuit simulator. Both mixed-domain simulation and transistor-level co-simulation are thus supported.

Complicated MEMS devices, including accelerometers, gyroscopes and high-order filters, have been designed and fabricated with the aid of NODAS. Fig. 1 shows a CMOS-MEMS bandpass filter composed of three identical crab-leg resonators [13], coupled by "O" springs. Differential comb drives are employed for capacitive driving and sensing. The schematic is fully compatible with the electrical interface circuit. There is one-to-one correspondence between the schematic and the layout.

Schematic Representation

In a NODAS schematic, connection terminals of element instances are represented by groups of pins. Each pin has an associated discipline determining its physical nature.

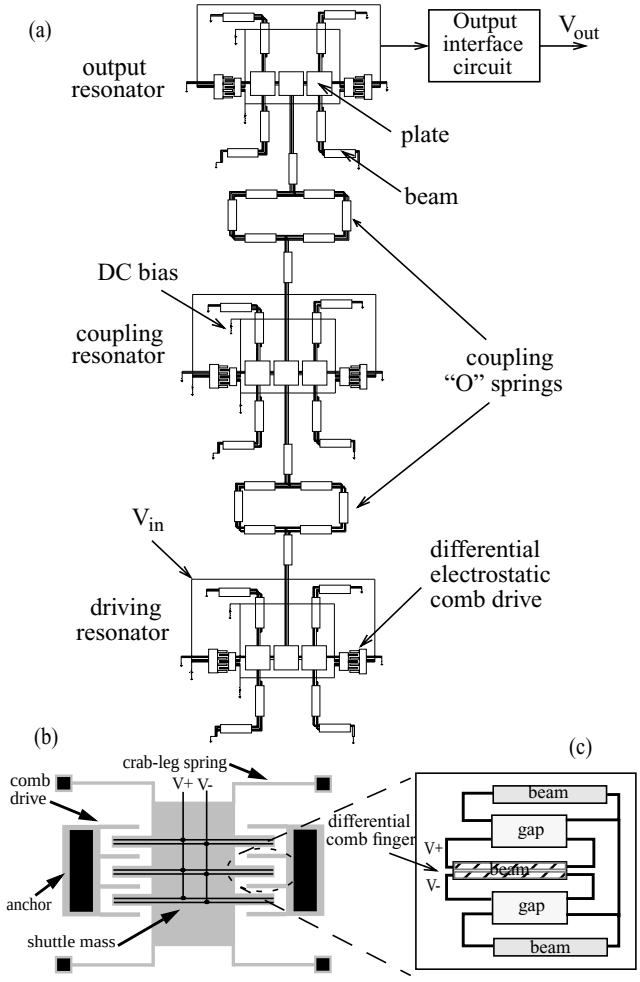

Fig. 1: (a) Top-level schematic of a bandpass filter (b) constitutional crab-leg resonator layout (c) atomic-level schematic of differential comb drive composed by gap and beam elements.

Pin definition affects both schematic composition and behavioral modeling. Fig. 2(a) and (b) explain the pin definitions using a cantilever beam as an example. The beam behavior is lumped at terminal *a* and terminal *b*. Each terminal has three translational pins, named as *x*, *y*, *z*, to represent the translational motions along the *X*, *Y* and *Z* axes, and three rotational pins, named as ϕ_x, ϕ_y, and ϕ_z, to represent the rotational motions about the *X*, *Y* and *Z* axes. The beam is also an electrical conductor, which is crucial for electrostatic actuation and sensing. Thus, electrical pins v_a and v_b are needed. In addition, as the initial position and orientation of the beam in the chip layout are important for modeling mechanics, the coordinates (X_c, Y_c) of the beam center relative to the layout origin and the angle Θ of the beam relative to the *X*-axis in the chip frame are included in the schematic representation. As the initial layout positions are static values which do not change during dynamic simulation, they are represented as parameters. A topological connectivity analysis algorithm calculates the layout coordinate values based on the topology and the element sizes in the schematic.

Schematic assembly consumes much effort and is prone to error when the schematic has many elements. As indicated in Fig. 2(b), at least seven wires are needed for each MEMS element terminal. In digital circuit schematics, buses are used for compaction of schematic representation. Similarly, analog buses are used in NODAS, as shown in Fig. 2(c). Due to the existing limitation in the analog HDLs, only pins of the same discipline are allowed to be grouped as one bus, resulting in three buses for each terminal (translational, rotational, and electrical, respectively). This compact terminal representation reduces wiring effort as well as the probability of wiring errors. Behavioral blocks that convert scalar wires to bus wires ("splitters" in Figure 2(c)) are used to apply stimuli and monitor simulation results at individual degrees of freedom.

When the chip experiences dynamics, the same amount of external acceleration and rotational rate must be applied to every element on the chip. Hence, global acceleration and rotational rate pins are used and shared by all elements, in combination with hierarchical schematic for each model element, to reduce clutter in the schematic. As seen in Fig. 2(d), these pins are not shown in the top-level symbol. Instead, they are defined as pins in the associated bottom-level schematics.

Another representation issue is the definition of across and through variables. Mechanical systems in translational motion must be in dynamic equilibrium: $\sum_{k=1}^{n} F_k = 0$. Using the analogy of force balance to KCL, forces are defined as through variables for the mechanical translational discipline. There are three choices for the definition of across variables for the translational discipline: positions, displacements or velocities. They all satisfy KVL. Displacements are preferred in suspended MEMS as they are usually of primary interest to be directly observed. Using displacements also enables the kinematics to be modeled explicitly. For cases where velocities are the primary interest, using velocities as across variables may be a better choice. Velocities are also convenient for cases where power is the main interest since the product of force and velocity is the mechanical power as an analogy to $V*I$ in electrical domain.

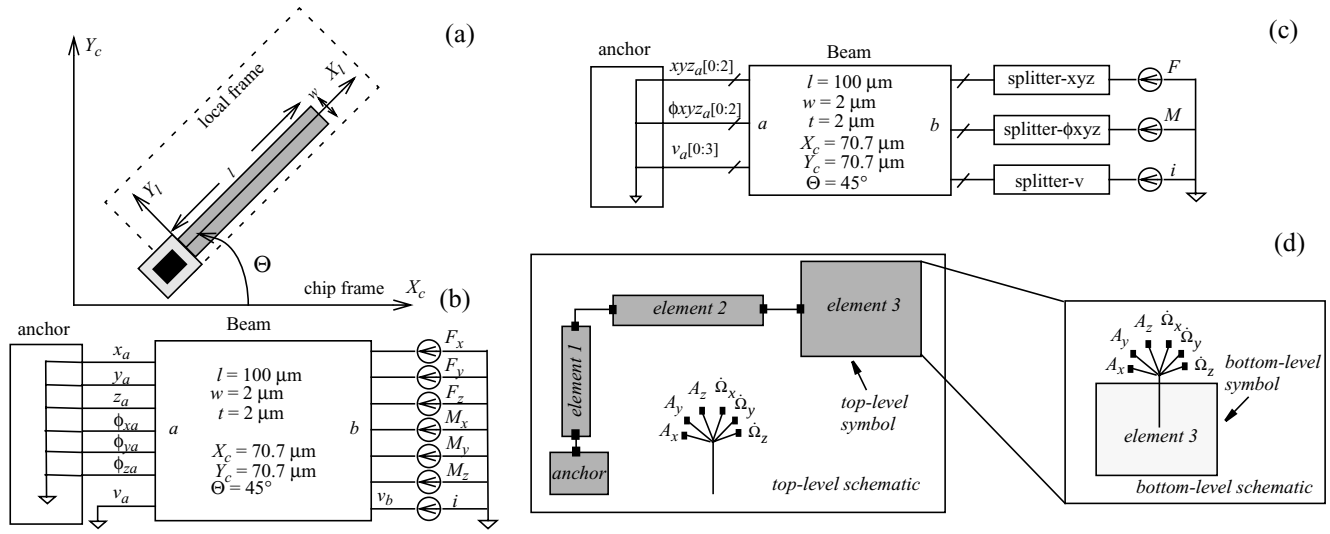

Fig. 2: Pin definition in NODAS (a) a cantilever beam on a chip (b) schematic with individual pins (c) schematic with bus pins (d) common external acceleration pins and hierarchical schematic.

Coordinate Transformation

The transformation of forces and displacements are dependent on the geometric orientation and position of each elements. As shown in Fig. 2(a), local frames of reference are attached to each element to decouple physics modeling from the geometric transformations. The chip frame is where the layout position of each element is defined. The across and through variables at the connection terminals represent the displacements, forces and moments in the chip frame and are used to form the system matrix [10]. The system matrix is then solved in the chip frame, following Kirchhoff's network laws specified by the topology.

The coordinate transformation between frames of reference is done through a 3D rotation matrix, which can be defined in many different ways [14]. A transformation matrix based on equivalent rotation about a single axis is used in NODAS because of its unique representation and independence on the order of rotations. In this method, element orientation is represented as an equivalent rotation about a rotation axis p by a rotation angle of ϕ, as shown in Fig. 3. This method is commonly used in finite element packages.

The corresponding rotation matrix is:

$$[R] = \begin{bmatrix} up_x^2 + c & up_xp_y + sp_z & up_xp_z - sp_y \\ up_xp_y - sp_z & up_y^2 + c & up_yp_z + sp_x \\ up_xp_z + sp_y & up_yp_z - sp_x & up_z^2 + c \end{bmatrix}, \text{ where } p=[p_x\ p_y\ p_z]^T,$$

$c=cos\phi$, $s=sin\phi$, and $u=1-cos\phi$. When $[R]$ is used to describe the rotation between the chip frame and the local frame for an in-plane beam element with an orientation angle Θ, $p=[0\ 0\ 1]^T$, and the rotation matrix simply reduces to:

$$[R] = \begin{bmatrix} cos\Theta & sin\Theta & 0 \\ -sin\Theta & cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

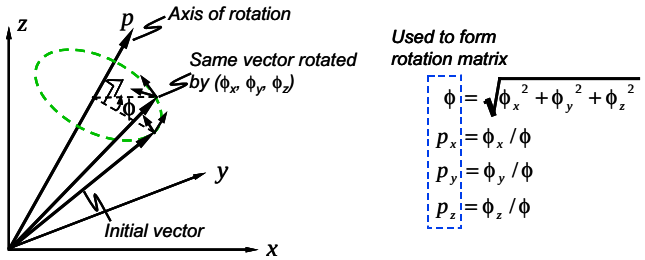

Fig. 3: Equivalent rotation about a single axis

Lumped Parameter Modeling

Geometrically parametrized lumped models are the core of the circuit-level simulation methodology. They are derived from structural matrix analysis, which lumps the distributed element behavior to a limited number of terminal nodes and describes the physics of the element in the form of matrices [15].

As an example, the linear beam model includes linear beam bending and torsion, inertial forces and damping. External forces and moments are restricted to concentrated loads at the beam ends. Using energy methods, beam bending and torsion are modeled as a stiffness matrix $[k]$, derived from beam mechanics equations [15]. Beam inertia is modeled as a mass matrix $[m]$ and damping is modeled as a damping matrix $[B]$, derived by assuming that the static mode shapes are effective for dynamic motion [15][16]. The relation between the force/moment and the displacement vector is established as: $[F_{beam}] = [m][\ddot{x}] + [B][\dot{x}] + [k][x]$, where $[F_{beam}] = [F_{xa}\ F_{ya}\ F_{za}\ M_{xa}\ M_{ya}\ M_{za}\ F_{xb}\ F_{yb}\ F_{zb}\ M_{xb}\ M_{yb}\ M_{zb}]^T$, $[x] = [x_a\ y_a\ z_a\ \phi_{xa}\ \phi_{ya}\ \phi_{za}\ x_b\ y_b\ z_b\ \phi_{xb}\ \phi_{yb}\ \phi_{zb}]^T$, and node a and b are the terminals at beam ends.

This lumped beam model is composable. It has been verified that using more beam elements to model a physical beam gives better simulation accuracy. The comparison of NODAS simulation to FEA simulation verifies the accuracy of the lumped model [10]. Lumped parameterized models for other atomic-level elements are derived in a similar way. The physical equations can be implemented in AHDLs in multiple alternative ways. The inclusion of internal states for the acceleration node representation has been shown to adversely affect the convergence property of the simulation [17]. These states are unnecessary and should not be included.

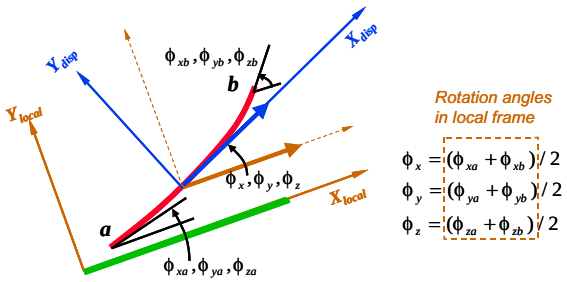

Fig. 4: Dynamic rotation between displaced frame and local frame

Nonlinear Beam Modeling

This section will discuss several specific issues about the nonlinear modeling in NODAS, taking the nonlinear beam model as an example.

The stiffness matrix in the linear beam model is only good for applications where the displacements are small. Although the linear model covers the basic beam mechanics existing in devices such as low-frequency resonators and accelerometers, it is unsuitable for simulation of large stroke actuators and RF-MEMS switches where the beam nonlinearity is non-negligible. Due to the complexity of the nonlinear beam mechanics and the inability to exploit superposition in nonlinear problems, analytical solutions are only available for several simple cases [18]. Finite element methods can analyze the nonlinear beam, but they do not fully support hierarchical multi-physics co-simulations, especially fast transient analysis and analysis with electronics. A nonlinear behavioral beam model has been developed in NODAS for use within conventional analog behavioral simulators [19]. The model is reusable and composable hence can be used as a building block for complicated systems. It captures geometric nonlinearity, which is the main nonlinear beam mechanism.

To handle large deflections, a dynamic coordinate transformation is introduced between the local frame and the displaced frame, as shown in Fig. 4. The transformation includes a translation about the center of the displaced beam, followed by a rotation about an rotation axis by an angle of ϕ, which is derived from the average of angular displacements at beam ends *a* and *b*. The angle ϕ varies in time along with the beam bending dynamics, thus the coordinate transformation between the displaced frame and the local frame is a dynamic transformation. This is in contrast to the static coordinate transformation between chip frame and local frame, which is only a function of initial orientation angles.

The effective beam length along the displaced beam, L', is a critical variable for modeling the nonlinearity. L' is calculated based on the cubic beam bending shape functions, which is only good for small deflection. Since the nonlinear stiffness matrix is based on the effective beam length, spring forces and moments should be calculated in the displaced frame based on variables in that frame, then be transformed back to the local frame. The inertial forces and moments are calculated in the fixed local frame rather than the displaced frame because they occur in the inertial frame while the displaced frame is a non-inertial dynamic frame. The same principle applies to the calculation of damping forces and moments. These three types of forces/moments are then summed up in the local frame and transformed into the chip frame altogether to join the analysis of system matrix, as shown in Fig. 5.

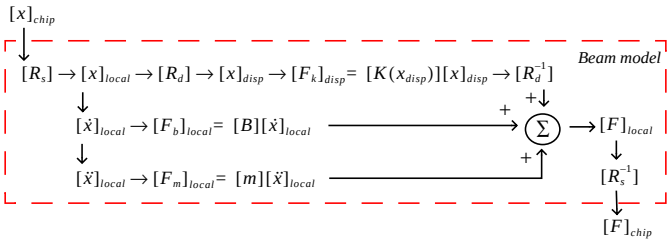

Fig. 5: Algorithmic structure of the nonlinear beam model

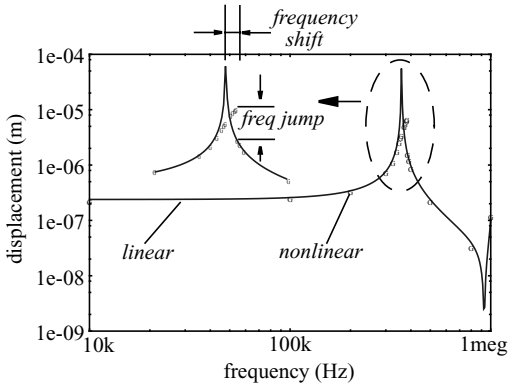

Fig. 6: Duffing effect simulations using linear beam model and nonlinear beam model.

The accuracy of the NODAS nonlinear beam model has been verified by detailed simulations and comparison to FEA results [19]. As is the case with FEA, using multiple beam elements to compose a physical beam allows the element cubic shape functions to more closely fit the actual shape function of the nonlinear beam, thus improves the simulation accuracy.

Due to the limit of small-deflection shape functions, an incremental loading method is needed to obtain accurate simulation results for large deflections. A nonlinear numeric method, such as Newton-Raphson, is also needed to solve the nonlinear equations. These techniques have to be included in the numeric solver, as is done in commercial FEA tools and in behavioral simulation tools with self-maintained numeric solvers such as SUGAR. Since the NODAS models are embedded in Cadence, we take advantage of the incremental loading and Newton-Raphson methods already included in the Spectre simulator. Accurate nonlinear solution can be obtained either by sweeping the load or by setting nodal displacement values at critical nodes.

Fig. 6 shows the simulation of a folded-flexure resonator excited by a large sinusoidal force. The steady-state envelope of the displacement magnitude obtained from a series of transient analyses with varying excitation frequencies are given. The existence of the duffing effect, a well-known phenomenon caused by beam stiffening nonlinearity, is clearly shown. The amplitude of the force is set to be such that the displacements are small at low frequencies but are large enough to cause the nonlinearity at frequencies near to the resonance.

Electrostatic Gap

Electrostatic effects are widely used in MEMS for actuation and sensing. This section takes the electrostatic gap model as an example to discuss the modeling of interactions between multiple physical domains.

The gap model with rigid parallel-plate approximation is the most simplified gap model. It only captures the electrostatic effect

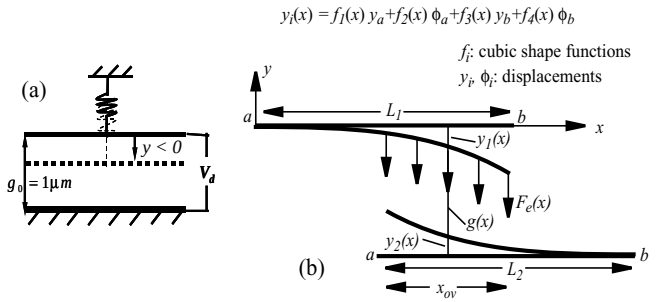

Fig. 7: (a) Gap model with rigid parallel plates (b) Gap model based on beam shape functions.

with acceptable accuracy in those cases where the comb fingers or electrodes are very rigid and only move in parallel without rotations. Consider a one-dimensional electrostatic gap for which the electrodes forming the gap are rigid plates in parallel, as in Fig. 7(a). The capacitance is $C = \frac{\varepsilon t L}{g_0 + y}$, where t is the thickness of the electrodes, L is the length of the overlap region, g_0 is the initial gap, and y is the variance of gap due to displacement. In MEMS, electrodes are normally suspended by springs, thus y represents the electrode displacement due to spring deformation. When a voltage, V, is applied, an electrostatic force will be generated: $F_{ey} = \frac{d}{dy}\left(\frac{1}{2}C(y)V^2\right) = -\frac{1}{2}\frac{\varepsilon t L}{(g_0 + y)^2}V^2$. The gap model is nonlinear and the electrostatic force is an attractive force.

Since y is not an arbitrary value but a variable determined by the balance between the spring restoring force and the electrostatic force, it's critical for the gap model to capture the interaction between the mechanical and the electrical domain to solve for the self-consistent solution. Fig. 8 describes pin definitions of the gap model. Fig. 9 illustrates the algorithmic structure procedure of the gap model in NODAS. The displacements of the electrodes are in the chip frame, transformed into the local frame. Equivalent parallel-plate capacitance and electrostatic force are then calculated. The electrostatic force is applied as through variables on the mechanical pins to join the analysis of the system matrix. The simulator iterates till a self-consistent solution is found for the system matrix.

Fig. 7(b) shows an improved gap model which captures the deflection of flexible electrodes (modeled as beams) thus suitable for simulation of comb drives with non-rigid fingers and MEMS switches composed with beams. Cubic beam bending shape functions are employed to capture the non-uniform electrical field in the overlapping region. The gap along the bent beams is: $g(x) = g_0(x) + y(x) = g_0(x) + y_1(x) - y_2(x)$, where $y_1(x)$ and $y_2(x)$ are the displacements of the upper and lower electrodes, respectively. Capacitance per unit length is thus $\tilde{C}(x) = \frac{\varepsilon t}{g(x)}$, and the

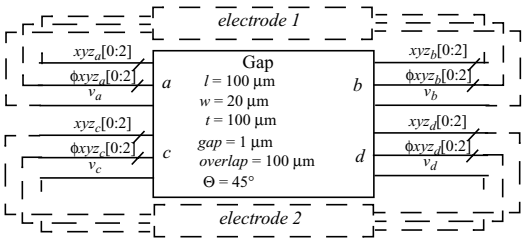

Fig. 8: Gap model symbol and wiring to electrodes

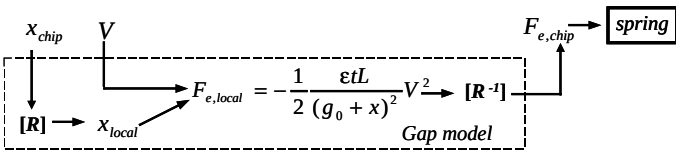

Fig. 9: Algorithmic structure of the simple parallel-plate gap model

total capacitance is obtained by integrating along the overlapping region: $C(x) = \int_{(L_1 - ov)}^{L_1} \tilde{C}(x)dx$. The electrostatic force per unit length along the y-axis is obtained by taking the derivative about y: $\tilde{F}_{ey}(x) = \frac{d}{dy}\left(\frac{1}{2}\tilde{C}(x)V^2\right)$. This force is a distributed force rather than a concentrated load, therefore, the force is lumped based on beam bending shape functions to transform into equivalent forces and moments at the ends of electrodes [20]. The loads for the upper electrode are:

$$F_{ey,a1} = \int_{(L_1 - x_{ov})}^{L_1} f_1(x, L_1)\tilde{F}_{ey}(x)dx\ ,\quad M_{ez,a1} = \int_{(L_1 - x_{ov})}^{L_1} f_2(x, L_1)\tilde{F}_{ey}(x)dx\ ,$$

$$F_{ey,b1} = \int_{(L_1 - x_{ov})}^{L_1} f_3(x, L_1)\tilde{F}_{ey}(x)dx\ ,\quad M_{ez,b1} = \int_{(L_1 - x_{ov})}^{L_1} f_4(x, L_1)\tilde{F}_{ey}(x)dx\ .$$

The loads for the lower electrode and the electrostatic force along the x-direction are obtained in a similar way.

Another crucial physical phenomenon to be captured by the gap model is the snap-in effect due to the nonlinear relation between displacement and attractive electrostatic force. When the voltage is larger than the pull-in threshold, the two electrodes come so close that the electrostatic force begins to increase at a faster rate than the spring force. The electrodes will then snap towards each other. In practice, if no measure is taken, the snap-in will cause short-circuit problems, hence a mechanical stop is normally designed to prevent electrical shorting. In simulation, the snap-in effect has the potential to cause instability and convergence problems, which are solved through proper structural definition in the schematic.

As shown in Fig. 10, the contact mechanism is modeled in NODAS by a stiff spring deforming in the direction of snap-in formed by the contact region of the electrodes, and by assuming the existence of an oxide insulator coated on both electrodes. Normally, the electrodes are made of metals or silicon, which will oxidize in the air. The thickness of the oxide insulator is parameterized with a default setting of 20 nm. When the gap is in normal operation, the contact spring force does not add into the dynamics. Once the gap is in contact, the contact spring begins to generate a large restoring

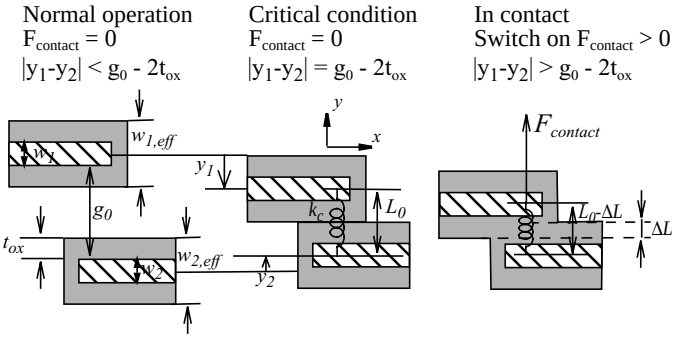

Fig. 10: Contact mechanism in gap model (top view)

force, $F_{contact}$, to work together with the restoring force from external elements to balance the large electrostatic force. The spring constant of the contact spring is: $k_c = \frac{E \cdot ov \cdot t}{\frac{w_{1,eff} + w_{2,eff}}{2}} = \frac{E \cdot ov \cdot t}{\frac{w_1 + w_2}{2} + 2t_{ox}}$, where E is the Young's Modulus of the electrodes, t is the thickness of the electrodes, w_1 and w_2 are the width of upper and lower electrodes, t_{ox} is the thickness of oxide insulator. The contact force is: $F_{contact} = k_c \cdot \Delta L = k_c \cdot (2t_{ox} - (g_0 + y_1 - y_2))$, where ΔL is the deformation of the contact spring.

III. Simulation Examples

In this section, simulation examples are presented to verify the accuracy of the beam and gap models and to validate the methodology.

Fig. 11(a) shows the static analysis of a gap composed with a cantilever beam over a plane. Results from NODAS are compared to the simulation results from Coventorware, a combined FEA/BEA tool which solves for the self-consistent solution through alternative mechanical and electrostatic analyses. Snap-in voltages agree to within 8%. Fig. 11(b) is the result of a transient simulation in NODAS, which shows the four states involved in the gap models: normal operation, snap-in, contact and peel-off. The simulation takes about 1 hour to finish, while in self-consistent FEA/BEA each data points takes about 30 mins and dynamic simulation is not supported.

Fig. 12(a) illustrates a MEMS switch composed of a fixed-fixed beam and an anchored surface. Each electrode is represented by two beam elements, in order to access the displacement node at the center of the beam. Accordingly, the physical gap is also composed of two elements. A sinusoidal voltage of 1 kHz is applied to the structure. Results from transient simulations with linear and nonlinear beam models are given in Fig. 12(c). With the nonlinear beam model, the beam stiffening effect is captured and the snap-in voltage increases consequently. Since the electrostatic force is proportional to V^2, the second harmonic is the primary displacement term when V is a sinusoidal voltage.

Fig. 13 shows the frequency response of the CMOS-MEMS bandpass filter (output voltage of interface circuit) given in Fig. 1, with comparison to experimental data [13]. Simulations with a comb drive macromodel, in which the comb fingers are assumed to be rigid, is compared to simulation with gap and beam models, in

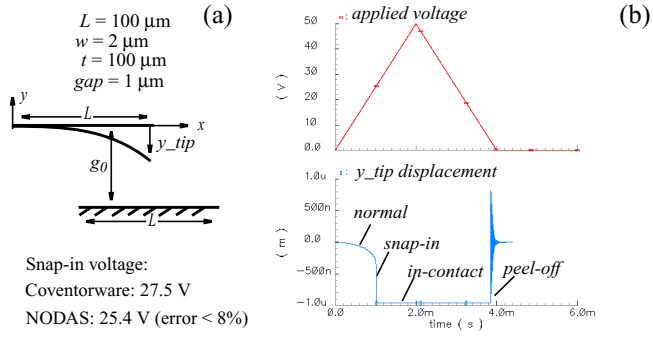

Fig. 11: Simulation of a cantilever beam over a plane (a) Static analysis (b) transient analysis

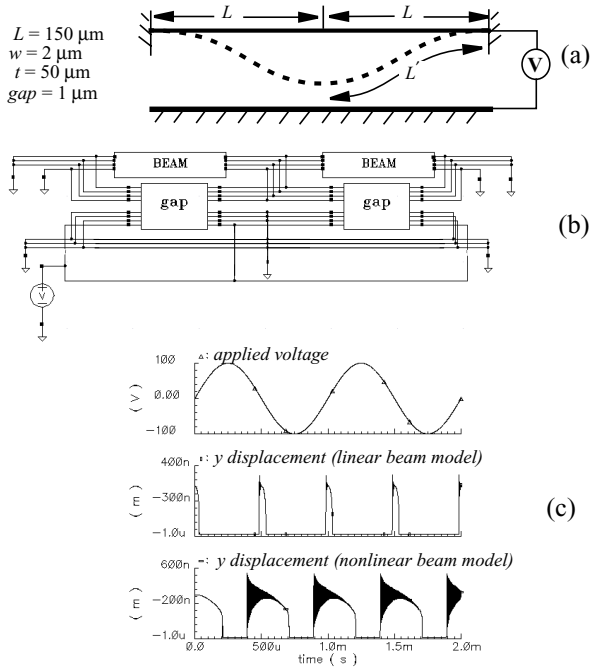

Fig. 12: (a) Structure of a MEMS switch (b) NODAS schematic (c) transient analysis.

which the compliance of comb fingers is considered. The comparison shows that the simulations matches the experimental results to within 3%. With gap and beam models, the frequency reduction due to the compliance of comb fingers is captured.

IV. Extensibility

A single cell library can not be expected to cover every desired feature. A useful simulation methodology is desired to be applicable to a wide range of applications and be extensible to new physical effects, new processes and new domains.

Extensibility to New Physical Effects

Any model is an abstraction of the real entity and cannot cover all the physical effects involved in the actual physical elements. Physical effects which are negligible in certain design spaces may become non-negligible in some other spaces. For instance, slender

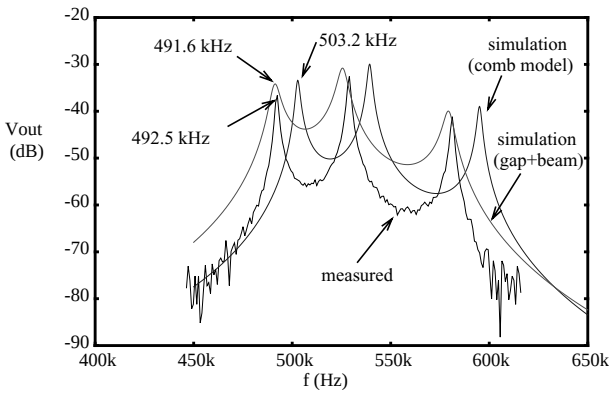

Fig. 13: Measured output voltage of sensing interface circuit and output voltage from NODAS simulations with rigid combdrive macromodel and gap model.

beams are widely used in low-frequency MEMS designs, but for recently emerging designs of RF-MEMS, shorter beams are used for higher frequency operation. The shear effect in beams thus becomes non-negligible. The nonlinear beam model can be extended to capture the shear effect by adding the energy due to shear deformation to the strain energy caused by beam bending. The shape functions are changed accordingly [15]. Nonlinear stiffness and mass matrices with the shear effect have been developed, following the same energy method as in the derivation of linear beam model.

Several other physical effects, including residual stress in microstructures [21] and finite sidewall angles due to etching have also been added to the beam model. The effect of residual stress, σ, is modeled as an axial force $F = \sigma A$ applied to the beam ends, where A is the cross-sectional area of the beam. The effect of sidewall angle is modeled by changing the moment of inertia for rectangular cross-section to that for a trapezoidal cross-section. The addition of these new features are fully compatible with the base beam model, as long as the fundamental algorithmic structure of the model stated in Fig. 5 is not changed.

Extensibility to New Processes

The models mentioned above are for a single structural layer process. In fact, there exist much more complicated processes, raising specific modeling issues. The models are expected to be extensible to new processes. For example, the cross-sectional structure in CMOS MEMS is a stack of metals and oxides. The moment of inertia for the composite structure is one of the key parameters which has to be captured. The misalignment of the multiple layers due to manufacturing variations also causes changes in the moment of inertia [22]. Moreover, since every metal layer is electrically conducting, corresponding electrical pins must be created for each layer. These issues are solved by properly modifying the constructive variables and equations on basis of the original extensible model structure.

Extensibility to New Domains

Physical domains involved in MEMS are not restricted to mechanical, electrical and electrostatic domains. For instance, different thermal expansion coefficients between the layer stacks of CMOS-MEMS devices cause curling problems. The NODAS beam model has been successfully extended to the thermal domain to capture this effect [21]. New pins associated with the temperature discipline are defined and the stresses caused by thermal expansion and compression are added into the beam model as an extra component of internal forces.

CONCLUSION

The modeling and simulation methodology in NODAS has the advantage in easy iterative hierarchical multi-domain co-simulation. The current NODAS model library is oriented to the mechanical and electrostatic physical effects involved in the specific design space of suspended MEMS. The accuracy of the existing models have been verified by comparing to FEA results and experimental data. This methodology is extensible to new physical effects, new processes and new physical domains.

ACKNOWLEDGMENT

This research effort is sponsored by the Defense Advanced Research Projects Agency (DARPA) and U.S. Air Force Research Laboratory, under agreement number F30602-99-2-0545 and F30602-01-2-0987 and in part by the National Science Foundation (NSF) Award CCR-9901171.

REFERENCES

[1] ANSYS User Manuals, ANSYS Inc. Global Headquarters, Canonsburg, PA.
[2] ABAQUS/Standard User's Manual, Hibbitt, Karlson and Sorenson Inc., Pawtucket RI.
[3] Maxwell User Manuals, Ansoft Corporation, Four Station Square, Pittsburgh, PA.
[4] Coventorware User Manual, Coventor Inc., Cambridge, MA.
[5] L.D. Gabbay, *et.al,* "Automatic generation of Dynamic Macro-Models Using Quasistatic Simulations in Combination with Modal Analysis", *Solid-State Sensors and Actuators Workshop*, Hilton Head Is., SC, June 7-11, 1998, pp.197-200.
[6] M.A. Maher, *et.al.*, "MEMS Systems Design and Verification Tools", *SPIE Smart Structures and Materials*, San Diego, CA, March 1998, pp. 40-48.
[7] G. Lorenz and R. Neul, "Network-Type Modeling of Micromachined Sensor Systems," *MSM'98*, Santa Clara, CA, April 6-8, 1998.
[8] J. Clark, K.S. Pister, *et.al.*, "Addressing the Need for Complex MEMS Design", *MEMS'02*, Jan. 20-24, 2002, Las Vegas, NV, USA, pp 204-209.
[9] D. Moulinier, *et.al.*, "MEMSMaster: a new approach to prototype MEMS", *DTIP'01*, April, 2001, pp.165-174.
[10] G.K. Fedder, *et.al.*, "A Hierarchical Circuit-Level Design Methodology for Micro-electro-mechanical Systems," *IEEE TCAS-II,* Vol. 46, No. 10, Oct. 1999, pp. 1309-1315.
[11] VerilogA Reference Manual, Cadence Design Systems, Inc., http://www.cadence.com.
[12] Spectre Reference Manual, Cadence Design Systems, Inc., http://www.cadence.com.
[13] Q. Jing, *et.al.*,"CMOS micromechanical bandpass filter design using a hierarchical MEMS circuit library," *MEMS '00*, Miyazaki, Japan, Jan. 23-27, 2000, pp. 187-192.
[14] A. Glassner, "Graphics Gems", Academic Press, Boston, 1990.
[15] S. P. Przemieniecki, "Theory of Matrix Structural Analysis," McGraw-Hill, New York, 1968.
[16] N. Zhou, *et.al.*, "Nodal Analysis for MEMS Design Using SUGAR v0.5", *MSM'98*, San Clara, CA, USA, April 6-8, 1998, pp 308-313.
[17] S. Iyer,*et.al.*, "Convergence and Speed Issues in Analog HDL Model Formulation for MEMS", *MSM'01*, March 19-21, 2001, Hilton Head, SC, USA.
[18] R. Frisch-Fay, "Flexible Bars", Butterworths Scientific Publications, Washington, D.C., 1962.
[19] Q. Jing, *et.al.*, "Large-Deflection Beam Model for Schematic-Based Behavioral Simulation in NODAS", *MSM'02*, April 21-25, 2002, San Juan, Puerto Rico, USA, pp 136-139.
[20] P. Mario, "Structural Dynamics, Theory and Computation", Van Nostrand Reinhold, New York, 1980.
[21] H. Lakdawala, *et.al.*, "Analysis of Temperature-dependent Residual Stress Gradients In CMOS Micromachined Structures", *Transducers'99*, June 7-10, 1999, Sendai, Japan.
[22] S. Iyer, *et.al.*, "Simulation of Manufacturing Variations in a Z-axis CMOS-MEMS Gyroscope", *MSM'02*, April 21-25, 2001, San Juan, Puerto Rico, USA, pp 186-189.