

***Newton*: A Library-Based Analytical Synthesis Tool for RF-MEMS Resonators**

Michael S. McCorquodale

James L. McCann

Richard B. Brown

Mobius Microsystems, Inc.
 Detroit, MI 48226-1686
 Tel: +1-313-420-5400x114
 Fax: +1-313-420-5404
 email: mccorquodale@mobiustmicro.com

Carnegie Mellon University
 Pittsburgh, PA 15213-3891
 Tel: +1-412-268-8525
 Fax: +1-412-268-5576
 email: jmccann@cs.cmu.edu

University of Utah
 Salt Lake City, UT 48112
 Tel: +1-801-585-7498
 Fax: +1-801-581-8692
 email: brown@coe.utah.edu

Abstract—*Newton* is a library-based CAD tool with an analytical synthesis engine which has been developed to support the direct synthesis of the physical design and an electromechanically equivalent model of RF-MEMS resonators based on process parameters and performance metrics. *Newton* provides accuracy comparable to finite element analysis while requiring a fraction of the computation and design time. A comparison of results from synthesis with *Newton*, design with FEA, and test results from fabricated devices is presented.

Newton is then presented and a sample derivation of the analytical expression for one key performance metric for one supported device is shown. The user interface is illustrated as a sample synthesis problem is solved, thus further describing the tool framework and its use. Experimental results from devices that have been fabricated from synthesis with *Newton* are presented along with FEA results. Good agreement is found between the specification and the measured results.

I. INTRODUCTION

II. MEMS SIMULATION APPROACHES

Since its inception, the general trend in the field of design automation has been to consider the design activity from an increasingly abstract level, the advantages of which are many. First, design productivity can be increased substantially. Second, design effort can be focussed on functional, logical, and performance verification as opposed to physical design. Lastly, with the use of an APR tool, the entire physical design process can be automated. Of course, significant physical design verification effort is still required, but design iteration can be achieved in a fraction of the time that is required for full-custom physical design.

A. Finite Element Analysis

FEA is an analysis technique by which an object is dissected into a finite number of elements and analyzed based on mechanical constraints, or boundary conditions, that are placed upon specific elements. The technique requires significant computation time for both finite element generation, or meshing, and solution convergence. Additionally, significant design effort is required to generate the 3D model of the structure to be simulated. Mesh and simulation time are largely contingent upon the mesh size and type. As the number of nodes in the mesh is increased, the accuracy is increased, but so is the computational complexity and simulation time. Each simulation typically requires several hours and a typical design effort can span days for a single device.

Certainly, a similar approach could be applied to the synthesis of microelectromechanical systems (MEMS) and the advantages would be similar. However, MEMS synthesis presents certain implementation challenges. Consider that synthesis of MEMS is not as straight-forward as analogous problems in the field such as digital logic synthesis. Further consider that MEMS devices must typically be modeled as continuous systems while logic can be described by a finite set of states. Also, a variety of MEMS topologies and manufacturing process technologies exist. Consequently, the standard approach to MEMS design has been with finite element analysis (FEA). With FEA, arbitrary structures can be designed in any material and simulated for a given set of boundary conditions. Though these tools are clearly indispensable for MEMS design, their use requires significant knowledge of MEMS device physics and manufacturing. Circumventing FEA would substantially increase design productivity. Similarly, some type of direct silicon compilation would reduce physical design time.

B. Nodal Analysis

In [1], a nodal analysis approach, similar to that implemented in SPICE, was developed in *Matlab* for MEMS simulation. Here, basic MEMS elements including beams, gaps, and anchors are parameterized based on geometry and can be coupled together. The law of static equilibrium is applied to each node and DC, steady-state, and transient results can be found by solving the resulting set of coupled ordinary differential equations using nodal analysis. With this approach, simulation time can be reduced dramatically as compared to FEA, though design iteration is still required.

In this work, an alternative library-based analytical synthesis tool, *Newton*, is presented for use in the design of radio frequency (RF)-MEMS resonators. *Newton* supports the direct synthesis of the physical design and an electromechanically equivalent model for RF-MEMS resonators from process parameters and performance metrics. Moreover, *Newton* provides performance accuracy comparable to FEA while requiring a fraction of the computation time and design effort.

III. MEMS SYNTHESIS APPROACHES

A. Automated Design Synthesis

Recent work has leveraged the speed at which MEMS can be simulated with the tool presented in [1] and lead to the exploration of automated synthesis approaches. A comparison of such approaches has been presented in [2] and includes the evolutionary approach presented in [3] where multi-objective genetic algorithms have been employed. Such an approach is general-purpose and allows the design space to be explored rapidly, but is suited to the design of arbitrary devices and not targeted at specific topologies of specific utility.

In the sections that follow, current MEMS simulation and synthesis approaches are described in greater detail where design effort and time are considered. The approach utilized in

B. Library-Based Analytical Synthesis

A parameterized layout generator for low frequency lateral MEMS resonators was presented in [4]. Here, parameters such as the device resonant frequency were used to solve simple mechanical relationships from which the device geometry could be determined, thus determining the physical design. Additionally, an electromechanical analogy was employed to build a SPICE model for system-level simulation and optimization. Though this work demonstrated reasonably good agreement between measurements from fabricated devices and the specified resonant frequency, system-level design iteration was required and extension of the framework was not addressed. Moreover, it would likely be difficult to achieve high specification accuracy due to the simplicity of the mechanical relationships that were employed.

In contrast to this previous work, *Newton* has been developed with an extensible component library framework. The reality of MEMS design and development is that although an infinite number and variety of devices can be conceived and developed with FEA or evolutionary synthesis, only a finite number possess any practical use. For example, RF-MEMS resonators have recently been demonstrated successfully in oscillator circuits [5]. Moreover, the numerous MEMS foundries that emerged throughout the mid and late 1990's are now being consolidated and standardized, thus reducing the variety of available devices and process technologies. Considering these trends, *Newton's* library contains previously and analytically designed components that can be selected and synthesized based on performance parameters. This finite component library is certainly not a limitation of the tool as *Newton* is an extensible software framework into which more components can be integrated easily. Components currently supported include RF-MEMS varactors and RF-MEMS resonators of two topologies: clamped-clamped beam (CCB) and free-free beam (FFB).

The back-end of *Newton* solves analytical expressions which have been derived for the synthesis of the supported MEMS components. Thus solutions can be obtained directly from the performance specification and process parameters. With this approach, the costly simulation and design time associated with FEA is reduced substantially. As compared to other analytical approaches, MEMS-specific design knowledge is not required for synthesis, and convergence to a specification is achieved by *Newton*, not the designer. Lastly, in this approach, lumped-parameter equivalent circuits have been derived, which provide significant utility for co-simulation of MEMS and transistor circuitry. Moreover, the total time required for physical synthesis and model generation is mere seconds.

In the following section, an example of one of the derived performance-driven analytical expressions utilized in the back-end of *Newton* is presented for illustrative purposes. How such expressions are utilized within the framework of *Newton* is presented following the derivation.

IV. EXAMPLE ANALYTICAL EXPRESSION AND COMPUTATIONAL ALGORITHM

Here, the derivation of the analytical expression for synthesis for one performance metric (the resonant frequency) of one supported component (a CCB resonator) is shown.

A. CCB Resonator Design Overview

A CCB resonator is illustrated in Fig. 1 and has been presented previously in [6], though without complete analysis for design synthesis purposes. It is comprised of a prismatic beam that is fixed at both ends and suspended over an electrode with an air gap between the electrode and the beam. The device is typically utilized as a frequency reference for oscillator and filter applications. Thus the most significant performance metric is the mechanical resonant frequency of the device. Two general physics-based harmonic analysis approaches can be applied in an effort to calculate the resonant frequency of this device and the corresponding lumped-parameter electrical model. These approaches include the Euler-Bernoulli Method [7] and Timoshenko's method [8], the former of which is presented next. The results from this analysis provide the general purpose analytical synthesis expression for synthesizing CCB resonators based on the resonant frequency.

B. Euler-Bernoulli Method and Solution

For transverse vibrations of a simple beam, the mode shape equation takes the following form [9],

$$u(x) = C_1(\cos kx + \cosh kx) + C_2(\cos kx - \cosh kx) - C_3(-\sin kx - \sinh kx) + C_4(\sin kx - \sinh kx) \quad (1)$$

where $C_1, C_2, C_3,$ and C_4 are constants; k is the wave number; and x is distance along the beam with length L . For the clamped-clamped case, the boundary conditions are:

$$u(0) = 0, \left. \frac{du}{dx} \right|_{x=0} = 0, u(L) = 0, \text{ and } \left. \frac{du}{dx} \right|_{x=L} = 0 \quad (2)$$

The first two conditions are satisfied if $C_1 = C_3 = 0$. The remaining two conditions determine a set of equations which can be solved by evaluating the determinant. After some algebra, the determinant becomes,

$$\frac{1}{\cosh kL} = \cos kL \quad (3)$$

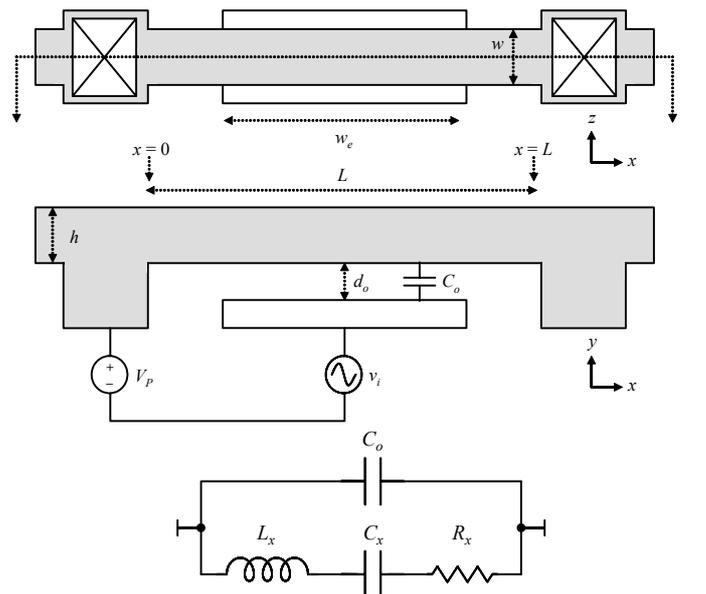


Fig. 1. Simple top and cross-sectional illustration of a CCB resonator along with equivalent lumped-parameter electromechanically equivalent circuit model at resonance.

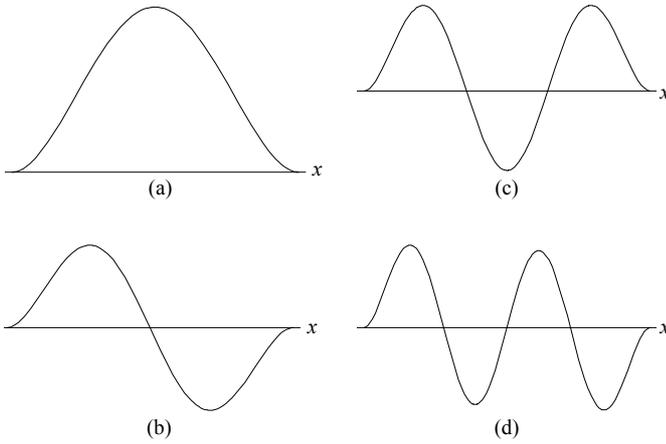


Fig. 2. First 4 mode shapes of a CCB resonator with length L : (a) Fundamental mode. (b) 2nd mode. (c) 3rd mode. (d) 4th mode.

There are multiple kL roots to (3). The first nontrivial root represents the fundamental mode shape of the device. This root, and others, are pictured in Fig. 2. Because L does not change for any of these mode shapes, it is useful to denote the wave number by k_n where n denotes the mode number. For each mode, C_1 and C_3 can be solved uniquely, thus $u(x)$ can be determined.

The fundamental resonant frequency, f_o , for any mechanically resonant device can be determined by solving the differential equations of simple harmonic motion. Without damping, the solution is [9],

$$f_o = \frac{1}{2\pi} \sqrt{\frac{k_m}{m}} \quad (4)$$

where k_m is the mechanical stiffness of the resonating body and m is its mass. In [6], it has been shown how (3) and (4) can be evaluated for the case of the CCB and an arbitrary mode,

$$f_n = \frac{(k_n L)^2}{2\pi L^2} \sqrt{\frac{EI}{\rho A}} \quad (5)$$

where f_n is the n^{th} resonant mode, E is Young's modulus, I is the moment of inertia, ρ is the density of the material, and A is the cross-sectional area where $A = W_r h_r$. The moment of inertia for a simple prismatic beam is given by, $I = (1/12)W_r h_r^3$, and substituting yields,

$$f_n = \frac{(k_n L)^2}{2\pi \sqrt{12}} \sqrt{\frac{E h_r}{\rho L^2}} \quad (6)$$

At first glance, this expression appears to be sufficient for analytical evaluation. However, consider that the CCB actually experiences two spring effects: mechanical (k_m) and electrical (k_e). The mechanical spring constant is determined by the stiffness of the material and the anchors. The electrical spring constant is associated with the DC voltage that is applied across the beam which effectively "softens" the spring constant associated with the system because this force is subtractive. Considering this effect, a modified version of (6) can be written as,

$$f_n' = \frac{1}{2\pi} \sqrt{\frac{k_m - k_e}{m}} = f_n \sqrt{1 - \frac{k_e}{k_m}} = \frac{(k_n L)^2}{2\pi \sqrt{12}} \sqrt{\frac{E h_r}{\rho L^2}} \left(1 - \frac{k_e}{k_m}\right)^{1/2} \quad (7)$$

The force on the beam can be calculated by considering the energy stored between the beam and the electrode. The differential of this force, with respect to displacement, is the electrical spring constant. Referring to Fig. 1, displacement is in the y direction.

$$F_e = \frac{\partial E}{\partial y} = \frac{1}{2} \frac{\partial C V_p^2}{\partial y} = \frac{1}{2} \frac{C_o V_p^2}{d_o} \quad (8)$$

$$k_e = \left| \frac{\partial F_e}{\partial y} \right| = \frac{C_o V_p^2}{d_o^2} = \frac{\epsilon A V_p^2}{d_o^3} \quad (9)$$

Consider that both k_m and m are functions of x , the position along the beam. Return to the general expression for the mechanical resonant frequency in (6) and consider the following relationship,

$$f_n = \frac{(k_n L)^2}{2\pi \sqrt{12}} \sqrt{\frac{E h_r}{\rho L^2}} = \frac{1}{2\pi} \sqrt{\frac{k_m}{m}} \quad (10)$$

and solve for the position-dependent mechanical stiffness, $k_m(x)$.

$$k_m(x) = \left[\frac{(k_n L)^2 h_r}{\sqrt{12} L^2} \right]^2 \frac{E}{\rho} m(x) \quad (11)$$

Now using (9) and (11) derive an expression for k_e/k_m over a differential length of the beam, dx , for a beam of length L . Such an expression becomes,

$$\frac{k_e}{k_m}(L) dx = \frac{V_p^2 \epsilon W_r}{d_o^3} \frac{\rho L^4}{(k_n L)^4 \frac{12}{E h_r^2} m(x)} dx \quad (12)$$

The expression in (12) should be integrated over the region in which there exists a beam-electrode overlap because this is the region in which a DC bias exists across the device, thus softening the system spring constant. Assuming that the beam is positioned in x between $x = 0$ and $x = L$ and that the electrode is centered under the beam, as illustrated in Fig. 1, (12) becomes,

$$\frac{k_e}{k_m}(L) = \frac{V_p^2 \epsilon W_r \rho L^4}{(k_n L)^4 E h_r^2 d_o^3} \int_{\frac{L-W_e}{2}}^{\frac{L+W_e}{2}} \frac{1}{m(x)} dx \quad (13)$$

At this point (13) could be evaluated by integration, except that no expression for $m(x)$ has been determined yet. This expression is determined next using a generalized equivalent mass technique [10].

The kinetic energy, KE , for a body in motion is given by,

$$KE = \frac{1}{2} m v^2 \quad (14)$$

where v is the velocity of the mass. If the equivalent mass of the body varies along the position of the body, then so must the equivalent velocity, such that energy is conserved. Thus, (14) can be rewritten as an integral in the form,

TABLE I
CCB RESONATOR PROCESS AND PERFORMANCE VARIABLES

Design variable	Type	Description
ρ	Process	Density
E	Process	Young's Modulus
h	Process	Beam height
d_o	Process	Beam-electrode gap
k_n	Performance	Determined by mode
V_p	Performance	Bias voltage
W_r	Performance	Beam width
W_e	Performance	Electrode width
f_o	Performance	Resonant frequency

$$m_{eq}(x) = \frac{KE}{\frac{1}{2}v^2(x)} = \frac{\frac{1}{2}\rho A}{\frac{1}{2}v^2(x)} \int_0^L v^2(x) dx \quad (15)$$

Velocity is the differential of position. Thus for an object at resonance, the velocity can be described in phasor form by, $v(x) = j\omega u(x)$, where ω is the radian resonant frequency and $u(x)$ is the displacement of the beam in the y direction at position x . Now (15) can be rewritten as:

$$m_{eq}(x) = \frac{\frac{1}{2}\rho A \int_0^L [j\omega u(x)]^2 dx}{\frac{1}{2}[j\omega u(x)]^2} = \frac{\rho A}{u^2(x)} \int_0^L u^2(x) dx \quad (16)$$

Now with an expression for $m_{eq}(x)$, an expression for $k_e/k_m(L)$ can be determined,

$$\frac{k_e}{k_m}(L) = \frac{V_p^2 \epsilon W_r \rho L^4}{(k_n L)^4} \frac{L + W_e}{2} \left(\frac{\rho A}{u^2(x)} \int_0^L u^2(x) dx \right)^{-1} dx \quad (17)$$

and finally (7) can be determined with the substitution of (17). Thus, an accurate physics-based analytical expression for the resonant frequency of the device has been determined. Moreover, the effect of spring softening on the resonant frequency of the device has also been included.

In a typical application, ρ , E , h_r , and d_o are determined by the fabrication process. Performance parameters k_n , V_p , W_r , W_e , and f_o are determined by design and ϵ and π are constants. Thus L can be synthesized from the design parameters using a numeric integration technique for solving (7) in L with (17) substituted.

Table I summarizes these process and performance parameters, while a summary of constants and derived variables is presented in Table II. The solutions to these expressions are determined in *Newton* via scripts written in *Mathematica*, a symbolic mathematics package. *Mathematica* was selected due to the fact that it supports symbolic integration, which appears often in this analysis. The computation algorithm involves using the design and process parameters in order to synthesize the device length, L , based upon convergence to the

TABLE II
CCB RESONATOR CONSTANTS AND DERIVED VARIABLES

Design variable	Value/Expression	Description
ϵ	8.85×10^{-12} F/m	Permittivity of free space
A	$A = W_r h_r$	Beam cross-sectional area
I	$I = (1/12) W_r h_r^3$	Moment of inertia
$u(x)$	Defined by (1)	Mode shape function
L	From (17) in (7)	Beam length

correct root of (7) with (17) substituted. The remaining parameters are simply calculated from the respective definitions. This analysis demonstrates one example of the typical back-end computations performed by *Newton* for a specific device.

V. TOOL FRAMEWORK

Newton is partitioned into a graphical user interface (GUI) and a synthesis engine in order to facilitate the development and addition of new devices to the library. The tool has a uniform GUI shared by all components and individual synthesis scripts for each component. The interaction between the GUI and the synthesis engine is illustrated in Fig. 3. To add a new component, one need only add a new component script to the engine. This modularity greatly facilitates the ability to expand the capabilities of the tool. The synthesis engine is simply comprised of the synthesis scripts which have been coded based upon performance analyses such as the analysis presented in the previous section.

Fig. 4 and Fig. 5 outline the synthesis procedure for a CCB resonator where the user interaction with the GUI can be described in two phases. In the first phase the user selects a device from the library component browser and specifies the performance and process-dependent parameters within the component parameter interface. Then synthesis is initiated.

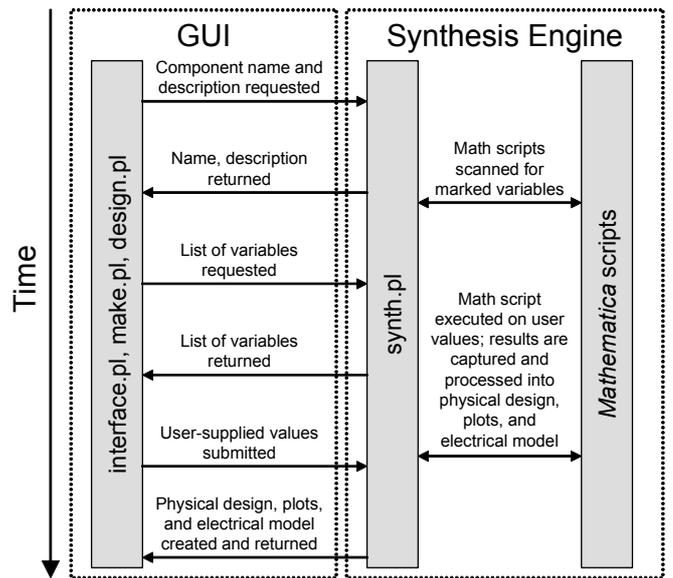


Fig. 3. The *Newton* framework partitioned into a GUI and a synthesis engine. The chronology of steps involved with a typical synthesis session is indicated by the time axis.

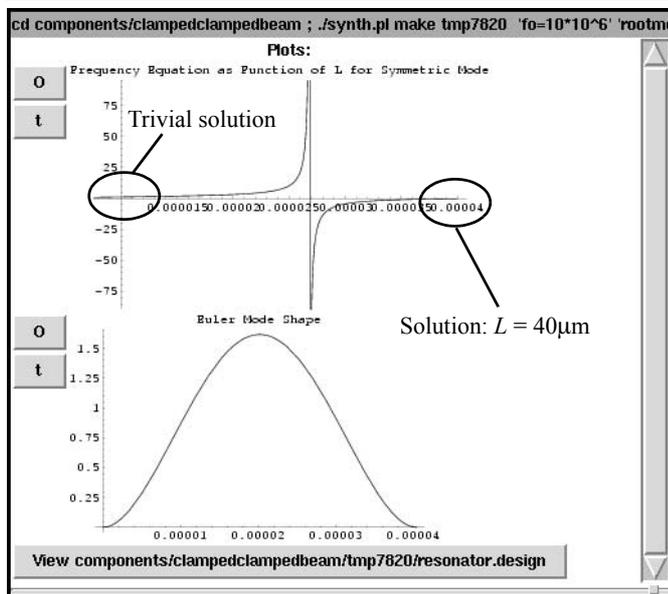
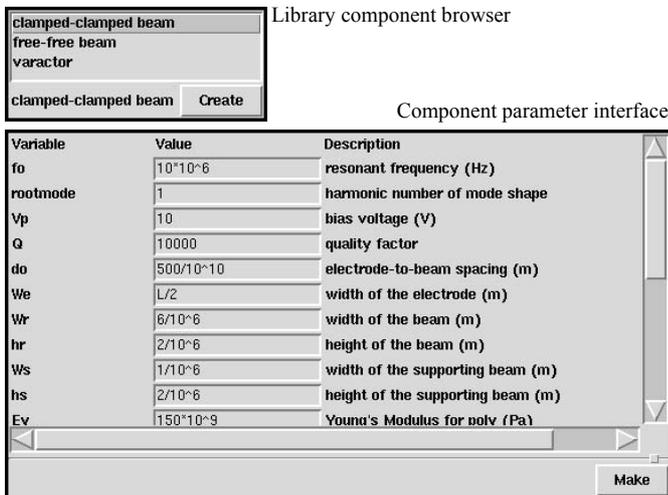


Fig. 4. Screenshot of *Newton*'s GUI and framework. Post synthesis results are shown for a 10MHz CCB resonator.

Results can be verified by examining the frequency equation graph and the mode shape as shown in Fig. 4. In the second phase, the user is presented with a graphical display of the mask set, shown in Fig. 5, for the component and allowed to adjust those parameters that are not fixed by the desired performance characteristics. These include options such as interconnect position and size. Once the design is complete, the user may export both the physical design in CIF for fabrication and the electromechanically equivalent model of the component in *SPICE* format for simulation with other MEMS devices or transistors.

A. Graphical User Interface

Perl and the GUI package Perl/Tk were utilized to develop *Newton*'s GUI since these packages are easily portable across operating environments and allow for rapid software development. The partition of *Newton*, however, allows new MEMS library component synthesis scripts to be written in any programming language. In this work, *Newton* was compiled for the *SUN Solaris* operating environment.

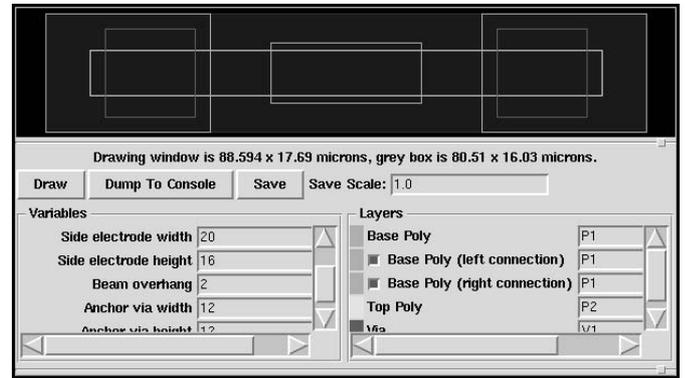


Fig. 5. *Newton*'s physical design viewpoint for a 10MHz CCB resonator, which is shown fabricated in Fig. 6.

B. Synthesis Engine

As described previously, synthesis scripts have been developed with *Mathematica* but could be eventually coded directly into the design framework with the aid of a math package. This would eliminate the need for a license for *Mathematica*. However, the synthesis scripts were developed with these tools due to the symbolic nature of the derived analytical expressions. Some challenges reside in porting the code for these analytical expressions to the code supported by a numeric math package, particularly in instances where intermediate symbolic solutions must be replaced with numeric solutions. For these reasons, this activity was not pursued in this work.

VI. A SYNTHESIS EXAMPLE

As a benchmark and synthesis example, a 10MHz polysilicon (poly-Si) CCB resonator was synthesized with *Newton* using the Euler-Bernoulli Method. The procedure and data from this design example are exactly those shown in Fig. 4 and Fig. 5. First, the CCB library part was selected from the library component browser. The process and performance parameters are entered into the synthesis form; the values for this example are listed for the reader's convenience in Tables III and IV. Synthesis was initiated by selecting the *Make* button. The frequency equation was plotted against L and displayed along with the mode shape equation. In Fig. 4, it can be seen that indeed the synthesis script has converged to the first nontrivial solution of the frequency equation and that the mode shape corresponds to the first resonant mode. The solution was $L = 40\mu\text{m}$ as shown. Once design convergence was

TABLE III
PERFORMANCE PARAMETERS FOR THE SYNTHESIS OF A 10MHz CCB RESONATOR.

Performance-Driven	Value
Resonant frequency, f_o	10MHz
Resonant mode number, n	1
Resonator width, W_r	$6\mu\text{m}$
Bias voltage, V_p	10V
Electrode width, W_e	$L/2$

TABLE IV
PROCESS PARAMETERS FOR THE SYNTHESIS OF A 10MHz CCB RESONATOR.

Process-Dependent	Value
Density, ρ	2330kg/m ³
Young's modulus, E	150GPa
Resonator height, h_r	2 μ m
Resonator-electrode gap, d_o	500Å

verified, the physical design was edited in real time as shown in Fig. 5. The physical design was generated automatically and displayed within a form where attributes that do not affect performance can be modified. These parameters were adjusted appropriately and the design was exported to CIF for mask generation. The device was fabricated using a custom polysilicon surface micromachining process in the University of Michigan's Solid State Electronics Laboratory. An electron micrograph of fabricated device is shown in Fig. 6.

VII. EXPERIMENTAL RESULTS

Prior to testing the fabricated device, the physical design from *Newton* was imported into the FEA tool *Coventorware*. An accurate 3D model was generated from process parameters that corresponded to the fabrication process for the device. A manhattan brick mesh of the 3D device was created and harmonic analysis was performed. The simulated fundamental mode resonant frequency was 10.20MHz as shown in Fig. 7.

The resonant frequency of the fabricated CCB device was determined by measuring the transmission scattering parameter, S_{21} , with an HP4195 network analyzer. The device under test was placed under a vacuum pressure of approximately 100mTorr. The measured spectrum is shown in Fig. 7, where the resonant frequency peak is at 9.93MHz and the quality factor is approximately 1,500.

Results from both *Newton* and *Coventorware* are close to the measured resonant frequency for the device. The simulated resonant frequency within *Coventorware* was 2.7% in error as compared to fabricated devices while the *Newton* design was in error by only 0.70%. Accuracy differences between the two packages likely arise from the fact that the synthesis scripts in *Newton* account for electrical frequency pulling, while the modal analysis in *Coventorware* is a mechanical analysis that does account for this phenomenon.

In addition to the CCB resonator, all of the remaining devices in *Newton*'s component library have been synthesized, fabricated, and tested. Results are not presented here in the

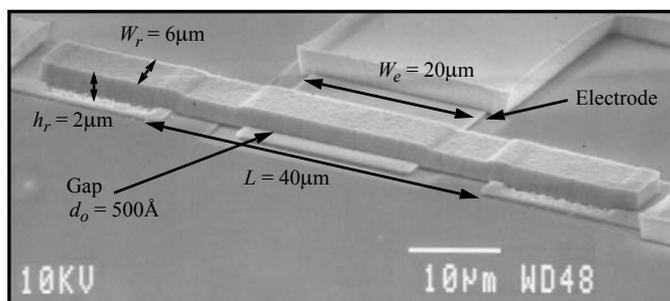


Fig. 6. Electron micrograph of a fabricated surface micromachined 10MHz CCB poly-Si resonator.

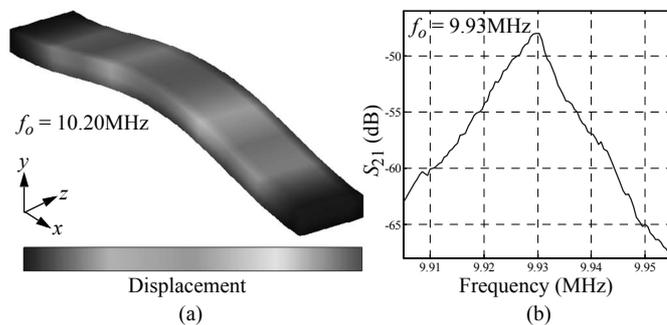


Fig. 7. (a) FEA simulation results. (b) Measured S_{21} response for the fabricated 10MHz CCB poly-Si resonator shown in Fig. 6.

interest of space, but performance is similar to that for the CCB resonator shown here.

VIII. CONCLUSION

Newton is the first complete CAD tool that supports library-based analytical synthesis of RF-MEMS resonators from a performance specification directly to physical design, while also generating an equivalent lumped-parameter electrical model for simulation with other devices. *Newton* has been shown to be a fast and accurate tool for the design of RF-MEMS resonators. As a benchmark, a 10MHz poly-Si CCB resonator was synthesized with *Newton* and the measured resonant frequency was in error by only 0.70%.

Newton could certainly be developed into a much more substantial tool through expansion of the component library. Each device would require analysis and derivation of critical performance parameters. The developed analytical expressions would then need to be coded into a math package, after which synthesis accuracy could be determined via fabrication and test. Beyond expanding the component library, the tool could be developed to increase automation. For example, process parameters could be set based on a selected foundry. The authors intend to explore such opportunities in future work.

REFERENCES

- [1] N. Zhou, J. V. Clark, and K. S. J. Pister, "Nodal analysis for MEMS design using SUGAR v0.5," *Proc. of Modeling and Simulation of Microsystems, Semiconductors, Sensors, and Actuators*, 1998.
- [2] R. Kamalian, N. Zhou, and A. M. Agogino, "A comparison of MEMS synthesis techniques," *Proc. of 1st Pacific Rim Workshop on Transducers and Micro/Nano Technologies*, 2002.
- [3] R. H. Kamalian, H. Takagi, and A. M. Agogino, "Optimized design of MEMS by evolutionary multi-objective optimization with interactive evolutionary computation," *Proc. of the Genetic and Evolutionary Computation Conference*, 2004.
- [4] N. R. Lo, et al., "Parameterized layout synthesis, extraction, and SPICE simulation for MEMS," *Proc. of IEEE International Symposium on Circuits and Systems*, 1996.
- [5] Y.-W. Lin, S. Lee, S.-S. Li, Y. Xie, Z. Ren, and C. T.-C. Nguyen, "60-MHz wine-glass micromechanical-disk reference oscillator," *Proc. of IEEE Int. Solid-State Circuits Conf. Dig. of Tech. Papers*, 2004.
- [6] C. T.-C. Nguyen, *Micromechanical Signal Processors*, Ph.D. Dissertation, University of California-Berkeley, 1994.
- [7] K. Wang, *Microelectromechanical Resonators and Filters for Communications Applications*, Ph.D. Dissertation, University of Michigan, 1999.
- [8] R. A. Anderson, "Flexural vibrations in uniform beams according to the Timoshenko theory," *J. of Applied Mechanics*, 1953.
- [9] W. Weaver, et al., *Vibration Problems in Engineering*, 5th ed., New York: John Wiley & Sons, 1990.
- [10] M. E. Frerking, *Crystal Oscillator Design & Temperature Compensation*, New York: Van Nostrand Reinhold Company, 1978.