# SugarCube: An Online CAD Tool for Parametrically Investigating the Performance of Ready-Made MEMS

Prabhakar Marepalli[1], Jason V. Clark[2]

[1,2]School of Mechanical Engineering
[2]School of Electrical and Computer Engineering
Birck Nanotechnology Center
Purdue University, USA

## ABSTRACT

We present an online tool called SugarCube that facilitates the parameterization of ready-made micro electro mechanical Systems (MEMS). Our tool is available online at nanoHUB.org through a web interface and remote computation. A web browser on the user's computer or cell phone is all that is required to run SugarCube. This tool has a hierarchical library which consists of multitude of complete networks of popular or useful MEMS that are based on the published work of experts. Once a user selects a system, default values of its key geometric, material and loading parameters are displayed. These parameters can either be held constant or modified. With these parameters, the user can perform static, modal, or transient analysis. The deflected system and numerical values of deformation are shown. What is new and different from previous efforts is that SugarCube does not require reading of manuals, programming, or any MEMS expertise to use. This makes SugarCube amenable to novice users at various educational levels to explore what-if scenarios of MEMS, and to experts who do not have time or desire to learn a traditional tool for modeling and simulating a device based on a common system. SugarCube may also be useful for manufacturers in surveying the performances of various MEMS for use in their new product.

***Keywords***: *S*ugarCube, Sugar, NanoHUB, parameterization

## 1. INTRODUCTION

With the growing use of MEMS in the consumer market, research, and education, the need for novice and expert designers to easily and quickly predict the performance characteristics of commonly used MEMS continues to increase [1], [2]. Many CAD for MEMS tools are available which are successful in simulating the behavior of common and unique devices [7]-[10]. But the use of these tools often require extensive training through tutorials or reading manuals, tech. support, and advanced education in science or engineering to determine if the computed result makes sense. The use of such tools in studying the behavior of unique devices is extremely valuable. However, for the larger number of novice and expert users who would like to explore common MEMS, no tool has adequately addressed such needs.

We created SugarCube to address these needs. Its main purpose is to reduce complexity thereby making it amenable to include a wide range of users, especially novices. It has a simple interface and a hierarchical library of commonly used MEMS which were coded by experts. Most of the models are from published work in the MEMS literature. Users can save and download their simulation data for further use. Since SugarCube uses Sugar's models and simulation engine [2], it benefits from Sugar's computational efficiency in modeling MEMS as previously verified and validated in [2]-[4].

The rest of the paper is organized as follows. In Section 2, we describe SugarCube's framework. In Section 3 we describe SugarCube's Graphical User Interface. And in Section 4 and 5 we provide SugarCube examples by parametrically exploring a thermal actuator, resonator, accelerometer, and gyroscope.

## 2. SUGARCUBE FRAMEWORK

SugarCube adds a simple and intuitive graphical user interface (GUI), parameterization, and optimization features to Sugar. Sugar is a system-level design, modeling, and simulation tool for MEMS. In [2] we reported on Sugar’s ability to fairly accurately and most efficiently emulate a wide variety of MEMS. However, similar to other lumped-element analysis tools (e.g. SPICE [5] and NODAS [6]), Sugar relies on predefined lumped-element models, which limits its applicability to MEMS that can be decomposed into its predefined elements. Otherwise, the lacking element model must be created for the tool. No such limitation exists for tools based on distributed-element analysis such as Coventorware [7], Intellisense [8], Ansys [9], and Comsol [10]. Therefore, distributed-element tools are usually better for modeling

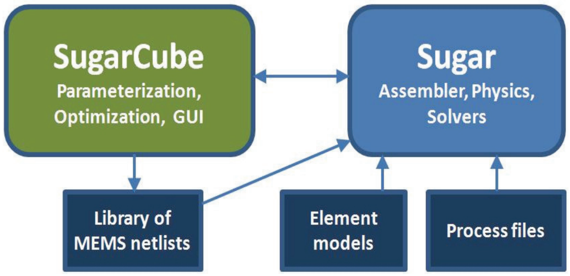
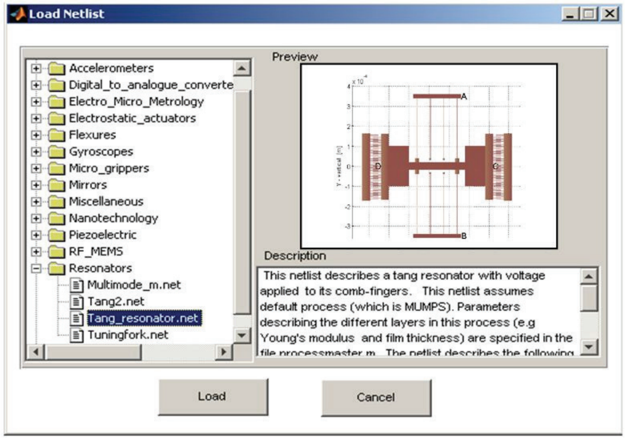


**Figure 1: SugarCube framework.** SugarCube is an extension of Sugar. SugarCube includes parameterization and optimization features as well as a simple graphical user interface to address the needs of novices. Sugar is used as the simulation engine for SugarCube. The arrows indicate directions of data flow.



This netlist describes a tang resonator with voltage applied to its comb-fingers. This netlist assumes default process (which is MUMPS). Parameters describing the different layers in this process (e.g Young's modulus and film thickness) are specified in the file processmaster.m. The netlist describes the following

**Figure 2: SugarCube MEMS library.** A hierarchical library displays images and descriptions of commonly-used MEMS. Each model is parameterizable for exploring performance as a function of geometry or geometry as a function of performance.

MEMS that have irregular geometries, fluidic fields, complex 3D electromagnetic fields, and higher-order effects in general.

*SugarCube versus Sugar.* Although Sugar can be used to efficiently configure wide variety of MEMS, the user is required to learn Sugar's unique netlist programming language in order to configure the systems. In addition, to efficiently and accurately model advanced systems, the user must develop a certain level of skill with netlist programming. With SugarCube, we eliminate netlist programming for the user, and supply ready-made models of MEMS that were coded by experienced Sugar users. Although SugarCube uses Sugar as its simulation engine, SugarCube is easier to use than Sugar at the expense of some limitations. Of course, the expert user may use Sugar to avoid such limitation.

## 3. GRAPHICAL USER INTERFACE

SugarCube has three main windows: a MEMS display window, a parameterization window, and a plot window. See Figure 3. The details follow.

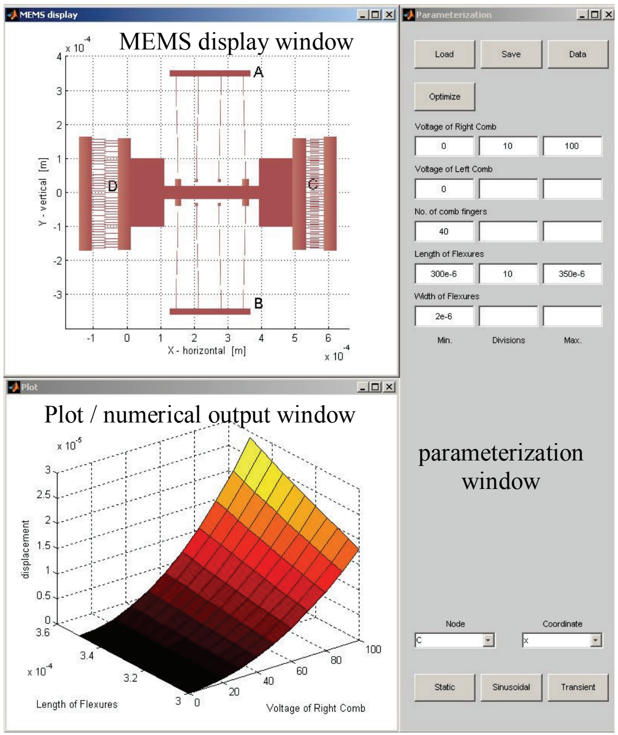
### 3.1 MEMS display window

The models loaded into the tool are displayed in this window. Each model in SugarCube is a network of predefined lumped elements which are connected together at nodes. Some of these key nodes are displayed at their corresponding locations in the model (see Figure3). After simulation, the deflected shape of the MEMS is displayed in this window.

### 3.2 Parameterization window

In the parameterization window, the user selects which parameters of the design space to explore. Through this window the user is also able to load a model, simulate and save results, and download simulation data.

Upon clicking the *Load* button, a window appears showing hierarchical library of ready-made MEMS models. See Figure 2. By selecting a category, a list of corresponding MEMS is displayed. An image preview and description accompany each selection in the library for easy identification. Upon selecting a MEMS from the library, the device is displayed in main MEMS display window and predefined parameters associated with this model appear in the parameter window. These parameters may include geometries, material properties, and dynamic performances, all with common default values i.e. the user can run each device in the library at once. If the user chooses to modify default values, each parameter has editable minimum, divisions, and maximum entry fields for parameter sweeps. Single-valued parameters are identified by a first-field entry only.



| | Min. | Divisions | Max. |
|---|---|---|---|
| Voltage of Right Comb | 0 | 10 | 100 |
| Voltage of Left Comb | 0 | | |
| No. of comb fingers | 40 | | |
| Length of Flexures | 300e-6 | 10 | 350e-6 |
| Width of Flexures | 2e-6 | | |

Node: C
Coordinate: x

**Figure3. SugarCube parameterized control and output windows.**

Solver options are given at the bottom of the parameterization window. Three kinds of solvers are available: static, modal, and transient analyses. The desired lumped-element node and coordinate of the output is also selectable for plotting.

### 3.3 Plot window

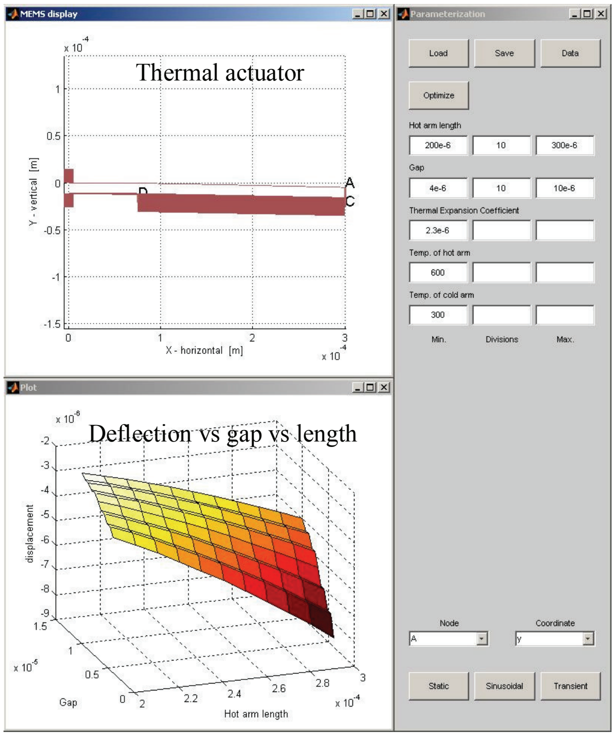
Output values of static parameters, 2D curves, or 3D manifolds of parameter sweeps appear in this window. Higher dimensional data may be downloaded for further analysis. For modal analysis, a drop down menu appears which lists different mode numbers, where each mode, frequency, and deformed shape is displayed in this window.

## 4 STATIC ANALYSIS EXAMPLES

Few examples of static analysis in SugarCube are demonstrated in this section.

### 4.1 Thermal actuator

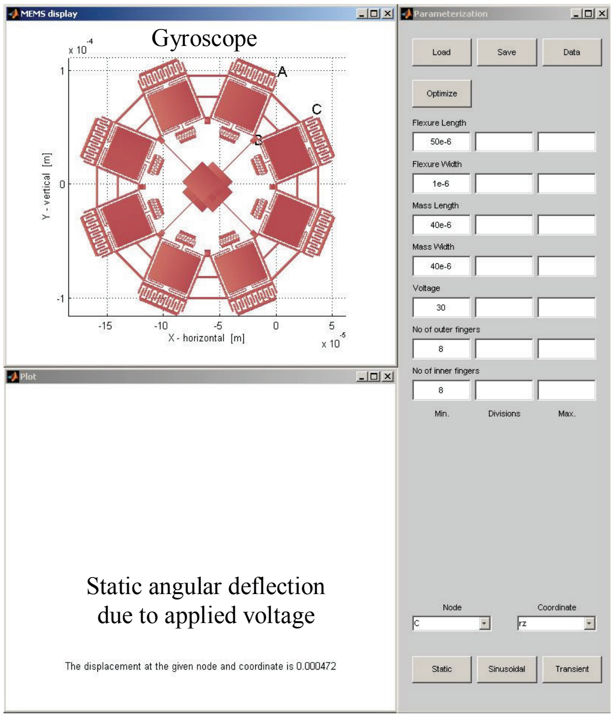
As a first example, a thermal actuator is considered. A thermal actuator is configured in a u-shape and it is actuated



| | Min. | Divisions | Max. |
|---|---|---|---|
| Hot arm length | 200e-6 | 10 | 300e-6 |
| Gap | 4e-6 | 10 | 10e-6 |
| Thermal Expansion Coefficient | 2.3e-6 | | |
| Temp. of hot arm | 600 | | |
| Temp. of cold arm | 300 | | |

Node: A
Coordinate: y

**Figure4: Thermal actuator.** The deflection in the device is because of the temperature difference across slender and thick beam. The behavior plot reveals the design space where geometry may be optimized for better performance



| | Min. | Divisions | Max. |
|---|---|---|---|
| Flexure Length | 50e-6 | | |
| Flexure Width | 1e-6 | | |
| Mass Length | 40e-6 | | |
| Mass Width | 40e-6 | | |
| Voltage | 30 | | |
| No of outer fingers | 8 | | |
| No of inner fingers | 8 | | |

Node: C
Coordinate: rz

Static angular deflection
due to applied voltage

The displacement at the given node and coordinate is 0.000472

**Figure 5: Gyroscope.** Distributed mass gyroscope taken from [11]. With the given input parameters, the static simulation shows a rotation of 0.0036 degrees at the tip of outer fingers.

by thermal expansion. If one side of the u-shape has a smaller cross section, then a passing electric current cause this side to have a greater thermal expansion than the other side. In doing so, the free end of the actuator deflects laterally. We experimentally validated our thermal actuator model in [4].

We show SugarCube's representation of a thermal actuator in Figure 4. Key nodes are displayed at corresponding positions. And key parameters are given in the parameterization window. A parametric sweep is made for the displacement at the tip as a function of gap between the beams and the length of the slender beam. Range of values for which the sweep has to be performed is entered in the Min. and Max. fields with the resolution entered in the Divisions field. The result of static analysis is given in the plot window. Such sweeps may reveal where geometry can be optimized for maximum performance.

### 4.2 Gyroscope

For the second example, we consider a microscale distributed mass gyroscope designed by Acar and Shkel [11]. This gyroscope has multiple drive mode oscillators distributed symmetrically around the center of the supporting flexures. The drive mode oscillators are driven in phase towards the center of the device. Upon a disturbance in the angular rotation rate about the z-axis, a Coriolis force at the drive frequency is induced on each proof mass in the direction orthogonal to each drive mode oscillations. This force generates a net torque on the device which excites the supporting frame into torsional oscillations about z-axis, which are detected by sense capacitors for angular rate measurement [12].

We show the SugarCube representation of this gyroscope in Figure 5. Taking advantage of symmetry, the model was initially configured in Sugar using its subnet feature where only a fourth of the entire device was coded into the netlist while the rest is replicated about the center anchor to create the above gyroscope.

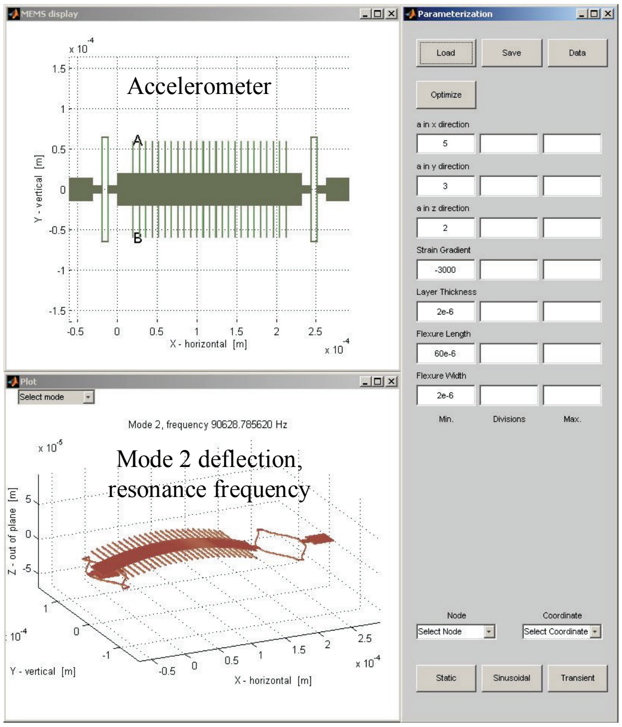
The key parameters are identified to be length and width of flexures and mass, voltage applied to outer comb fingers and number of comb fingers on each side. The electrostatic comb drive forces are applied to the outer comb fingers, which generate a resulting torque on the supporting frame [12]. We show the deformed structure and the numerical value of deflection in Figure 5.

## 5 MODAL ANALYSIS EXAMPLES

In this section we demonstrate a couple of examples using the modal analysis solver in SugarCube.

### 5.1 Resonator

For the first modal analysis test case, we explore a most common MEMS resonator (see Figure 3) designed by Tang and Howe [13], which was the first device to make use of the popular comb drive sensor/actuator. Upon loading the model in SugarCube, the key parameters are given as voltage of left and right comb fingers, length of flexures, width of flexures, and number of comb fingers on each side. The available



Mode 2, frequency 90628.785620 Hz

a in x direction: 5
a in y direction: 3
a in z direction: 2
Strain Gradient: -3000
Layer Thickness: 2e-6
Flexure Length: 60e-6
Flexure Width: 2e-6

**Figure 6: Modal analysis of an accelerometer.** The accuracy of modal analysis of SUGAR compared to experimental values is shown in [4]. The device is observed bending out of the plane. This behavior is the result of residual stresses in the device.

output nodes are identified on the structure in the MEMS display window. We select a flexure length and width of 300 $\mu m$ and 2$\mu m$, and choose sinusoidal (modal) analysis. The first five Eigen modes are computed in SugarCube (not shown). Upon selecting mode 2, SugarCube shows the characteristic deflection and a resonance frequency of 3218Hz.

## 5.2 Accelerometer

The accelerometer has been successfully used in the transportation, communications, and entertainment industries. As for the popular ADXL accelerometer, we have previously validated the strain gradient that results from the device being fabricated in a BiCMOS process using Sugar [14]. In SugarCube, the user is able to explore the performance as a function of acceleration in x, y, and z directions, out-of-plane strain gradient, thickness of the device, and the length and width of flexures.

We show SugarCube's resultant Eigen mode 2 in Figure 6 with a resonant frequency of 90628 Hz. As also seen in the figure, SugarCube shows the warping effect due to a strain gradient causing the device bending out of plane.

## 6 CONCLUSION

In this paper we introduced a new online tool called SugarCube, which adds a simple and easy to use graphical user interface and parameterization features upon a Sugar modeling and simulation engine. SugarCube does not require any programming or extensive training to use. It allows novice users to easily explore what-if scenarios of ready-made by changing various governing parameters. It is also useful for experts who have little time or interest to learn how to master more sophisticated simulation software. For test cases we demonstrated the parameterization of a thermal actuator, gyroscope, resonator, and an accelerometer. Expert users that have created MEMS using Sugar may upload their models into SugarCube for public access.

## ACKNOWLEDGEMENT

This work is sponsored by the NSF-supported nanoHUB, Network for Computational Nanotechnology.

## REFERENCES

[1] S.D. Senturia, "CAD Challenges for Microsensors, Microactuators and Microsystems," *Proceedings of the IEEE*, Vol. 86, No. 8, (1998)

[2] Jason Vaughn Clark and Kristofer S. J. Pister , "Modeling, Simulation, and Verification of an Advanced Micromirror Using SUGAR", *Journal of Microelectromechanical Systems, Vol. 16,* No. 6, December, 2007, pp.1524-1536.

[3] J. V. Clark, N. Zhou, D. Bindel, L. Schenato, W. Wu, J. Demmel, K.S.J. Pister, 3D MEMS Simulation Modeling Using Modified Nodal Analysis, *Proceedings of the Microscale Systems: Mechanics and Measurements Symposium*, June 8, (2000), Orlando FL, USA, pp. 68-75

[4] J. V. Clark, D. Bindel, N. Zhou, S. Bhave, Z. Bai, J. Demmel, K. S. J.Pister, "Sugar: Advancements in a 3D Multi-Domain Simulation Package for MEMS," *Proceedings of the Microscale Systems: Mechanics and Measurements Symposium*, June 4 (2001), Portland OR, USA

[5] L. W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," *ERL Memo*. No. UCB/ERL, Vol. M75/520 (1975)

[6] G. K. Fedder and Q. Jing, "NODAS 1.3: Nodal Design of Actuators and Sensors", *Proc. IEEE/VIUF Int. Workshop on Behav. Mod. and Sim.*, (1998)

[7] Coventorware, 951 Mariners Island BLVD, Suite 205, San Mateo CA 94404, http://www.coventor.com

[8] IntelliSense Corporation, 600 W. Cummings Park Suite 2000, Woburn MA 01801. (http://www.intellisense.com)

[9] ANSYS, Inc. Southpointe, 275 Technology Drive, Canonsburg, PA 15317, www.ansys.com

[10] Comsol, 1 New England Executive Park, Suite 350 Burlington, MA 01803, USA, www.comsol.com

[11] C. Acar and A. M. Shkel "An Approach for Increasing Drive-Mode Bandwidth of MEMS Vibratory Gyroscopes" Journal of Microelectromechanical Systems Vol. 14, No. 3, pp. 520-528, (2005)

[12] A. M. Shkel, C. Acar and C. Painter, "Two Types ofMicromachined Vibratory Gyroscopes", pp. 531-536, *IEEESensors*, Irvine, CA, USA (2005)

[13] W. C. Tang, T.-C. H. Nguyen, and R. T. Howe, "Laterally driven polysilicon resonant microstructures," *Sensors and Actuators*, vol. 20, pp. 25-32, (1989)

[14] F. Li, J. V. Clark, "SUGARCube – ADXL Accelerometer," DOI: 10254/nanohub-r6730.2 (2009)