

A More Effective Technique of Design Synthesis for MEMS with Expected Performance

Shuxun Chen

Department of Mechanical Engineering, Guangxi University, Nanning, China

E-mail: chenshx@gxu.edu.cn

Received December 17, 2009; revised January 23, 2010; accepted February 26, 2010

Abstract

A design synthesis technique based on sensitivity for Micro-Electro-Mechanical Systems (MEMS) proposed. This new technique can be called Sensitivity-Based Direct Solution Algorithm (DSA) of design synthesis for MEMS with expected performance. Design synthesis with expected performance is regarded as a reverse problem of MEMS analysis. Behavior equation group can be deduced from analysis equations. Solving the behavior equation group only need L design variables, L is number of desired behaviors. This behavior equation group can be solved using any solution algorithm of non-linear equation group. Newton Iteration Method based on sensitivity is adopted. Comparing with Genetic Optimization Algorithm (GA) and Simulated Annealing Optimization Algorithm (SA), computational workload of DSA is greatly decreased. For instance, synthesis computation of a meandering resonator only needs 4 iterations (17 analyses); computational time is decreased from 7~8 hours to less than 30 seconds.

Keywords: MEMS, Design Synthesis, Direct Solution Algorithm, Genetic Algorithms, Simulated Annealing, Comparing

1. Introduction

Estimating Micro-Electro-Mechanical Systems (MEMS) are miniaturized mechanical devices and components, often integrated or interfaced with electronics and fabricated on silicon wafers. They are systems in multi energy domains including electrical, mechanical, thermal etc. So their analysis and design is more complex than simple systems in single energy domain e.g. structural systems or circuit system. With rapid development of micro-electronic industry, techniques of analysis, synthesis and optimization of MEMS are in great demand to automate the MEMS design process.

MEMS synthesis usually begins with high-level specifications that describe the expected behaviors of devices [1]. This kind of design synthesis problem of MEMS devices can be described as: design variables are physical configuration and geometries; it is design objectives to find values of the design variables so that same behaviors equal expected values; it is design constraints that values of design variables are not allowed beyond given bounds.

Optimization problem of MEMS can be described as:

Project 50965002 supported by NSFC.

design variables are physical configuration and geometries; it is design objective to find values of the design variables so that some behavior achieves its the optimum value; design constraints are bounds of the design variables and some performance restriction.

There has been some progress on MEMS device Synthesis [2–5]. However, the present synthesis methods are tailored to specific tasks, are not general-purpose synthesis tools.

From 1998, research group leaded by Prof. Alice M. Agogino have developed an analysis simulation tool for MEMS devices—SUGAR [6–8]. It is based on nodal analysis techniques from the world of integrated circuit simulation, includes beams, anchor, electrostatic gaps, circuit elements, etc. MEMS are modeled as coupled systems of differential equations. Using SUGAR, static, dynamic, and steady-static and transient analyses simulation of performance behaviors of MEMS devices can be conveniently processed. So SUGAR is a good base of advanced techniques of general-purpose synthesis and optimization of MEMS devices.

Because the problem of design synthesis can be regarded as the optimization problem in which it is design objectives to find values of the design variables so that difference of same behaviors and their expected values

achieves it's the minimum values–zeros. On the basis of SUGAR, the research group firstly developed some optimization algorithms to solve the design synthesis problem of MEMS devices. Dr. Zhou etc. adopted Genetic Algorithm (GA) [9]. Kamalian etc. adopted Simulated Annealing Algorithm (SA) [11]. They can be called Simulated Nature Algorithms (SNA). It is shown by engineering examples that these SNA are effective for design synthesis problems with many variables including topological variables, but its computational workload are enormous.

Regarding the above-mentioned design synthesis problem as a design problem of MEMS devices with expected performance behaviors, Prof. Chen proposes a Direct Solution Algorithm (DSA) of based ob sensitivity. For the same engineering example, comparing SNA, computational workload of the Direct Solution Algorithm is greatly decreased from thousands analysis to less than 30 analysis. Its computational time is greatly reduced from hours to less than 30 seconds, and only a few design variables are necessary, other design parameters can be decided according to convenience.

In this paper, we introduce the above-mentioned analysis method and analysis tool, SNA design synthesis Algorithms, proposes the Direct Solution Algorithm (DSA), compare them through an engineering example, and discuss various design strategy and their development.

2. Analysis Method and Analysis Tool

2.1. Analysis Equation

In multi energy domains, such as electrical circuits, mechanical, thermal etc., analysis equation group of micro-electro-mechanical systems can be described as the following modified nodal equation group:

$$\mathbf{M} \mathbf{q}'' + \mathbf{C} \mathbf{q}' + \mathbf{K} \mathbf{q} = \boldsymbol{\Sigma} \mathbf{F} \quad (1)$$

where the three dimensional node vector and excitation vectors for a system of N nodes ($6N$ freedom degree) are:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ \vdots \\ q_N \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ \vdots \\ F_N \end{bmatrix}, \quad q_i = \begin{bmatrix} q_{xi} \\ q_{yi} \\ q_{zi} \\ q_{0xi} \\ q_{0yi} \\ q_{0zi} \end{bmatrix}, \quad F_i = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_{xi} \\ M_{yi} \\ M_{zi} \end{bmatrix} \quad (2)$$

Any electrical elements are appended onto these mechanical elements, creating vectors of length $6N$ Mechanical + N Electrical. The \mathbf{q} is node vector that consists of unknown node variables, such as mechanical displacements, electrical potentials and thermal tem-

peratures etc. \mathbf{q}'' and \mathbf{q}' is the second derivative and the first derivative of \mathbf{q} to time respectively. The \mathbf{F} is excitation vector that consists of known external loads, such as forces, currents and heat flux. The equation group expresses: At each node of MEMS, sum of forces equals zero, sum of currents equals zero, sum of heat flux equals zero.

The coefficients \mathbf{M} , \mathbf{C} , and \mathbf{K} represent the mass matrix, damping matrix and stiffness matrix respectively. They are composed of element mass matrices \mathbf{M}_i , element damping matrices \mathbf{C}_i and element stiffness matrices \mathbf{K}_i after coordinate transformation. Literature [8] presented \mathbf{M}_i , \mathbf{C}_i , and \mathbf{K}_i of a linear beam element, and presented modified computational expressions of \mathbf{F} and \mathbf{q} for current thermal expansion, residual stress, electrostatic gap and accelerating base.

2.2. Analysis Algorithm

2.2.1. Static Analysis

Given a system subject to nonlinear excitations of various energy domains (e.g. thermal, electrical and mechanical), when the system is in static equilibrium state, the equation group (1) is tenable. It can be briefly expressed as:

$$\mathbf{f}(\mathbf{q}) = \mathbf{0} \quad (3)$$

where the function $\mathbf{f}(\mathbf{q})$ contains the modified nodal equation for all energy domain. To solve the nonlinear equation group, Newton-Raphson algorithm is adopted. Its iteration formula is:

$$\mathbf{q}^{(n+1)} = \mathbf{q}^{(n)} - [\partial \mathbf{f}(\mathbf{q}) / \partial \mathbf{q}]^{-1} \mathbf{f}(\mathbf{q}^{(n)}) \quad (4)$$

where $[\partial \mathbf{f}(\mathbf{q}) / \partial \mathbf{q}]$ is Jacobian matrix composed of sensitivities.

2.2.2. Steady-State Analysis

To determine the steady-state response, firstly linearize the system of ordinary differential equation at the point of static equilibrium. The high order system is then converted into the first order form given by

$$\begin{aligned} \mathbf{x}' &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \end{aligned} \quad (5)$$

where \mathbf{x} is the system dynamic state variable, \mathbf{u} is the sinusoidal excitation, and \mathbf{y} is the system dynamic response. \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} are the system, input coupling, output, and feed forward matrices respectively. The solution of Equation (5) provides Bode plots as well as modal analysis.

2.2.3. Transient Analysis

To calculate the transient response of a MEMS device, which may contain nonlinear elements and excitations that are functions of time t and state vector \mathbf{q} , several

solvers can be available. These numerical methods include an implicit second order Rosenbrock solver for stiff problems where low accuracy is acceptable, an explicit Runge-Kutta 4th-5th order solver for non-stiff systems, an implicit multi-step integration method of varying order for stiff problems requiring higher accuracy, and a simple explicit Euler algorithm. The transient solvers require the modified nodal equations to be in first order form. We do this in the standardized way [12] by introducing a new state vector Q where

$$\dot{Q}' = \frac{d}{dt} \begin{bmatrix} q \\ q' \end{bmatrix} = f(t, q) \quad (6)$$

where t is time.

2.3. Analysis Software – SUGAR

From 1998, on the basis of the above-mentioned analysis equation and analysis algorithm, the research group developed a general-purpose analysis simulation software system of MEMS devices—SUGAR.

SUGAR is a collection of MATLAB algorithms that take a netlist description of MEMS devices and perform static, steady-state, model, and transient analyses of three-dimensional mechanical structures and electrical circuits. The code, demo files and manual of the newest version SUGAR are downloadable from the Berkeley Sensor and Actuator Center at the University of California at Berkeley.

3. Design Synthesis Using Simulated Nature Algorithms

3.1. Design Synthesis Problem of a Resonator

Design synthesis problem of MEMS devices can be described as: To find values of the design variables (physical configuration and geometry), so that same of values of performance behaviors equal expected values, subject to bound constraints of design variables.

Figure 1 shows a micro-resonator with a center mass (length L , width W) supported by four serpentine springs at 1, 2, 3, 4 node. Each spring is composed of N beams and fixed by an anchor at end of the spring. There are two sets of electrostatic comb drives on both sides to drive this resonator vibrating horizontally.

Design synthesis problem of the meandering resonator is described as: Design variables are lengths l_i , widths w_i and angles θ_i of every beam, and total number N of beam in every spring. Design objectives are as: the lowest frequency f equals 93723 Hz; x-directional Stiffness K_x equals 1.90 N/m. y-directional stiffness K_y equals 0.56 N/m. Bound constraints of design variables are as **Table 1**:

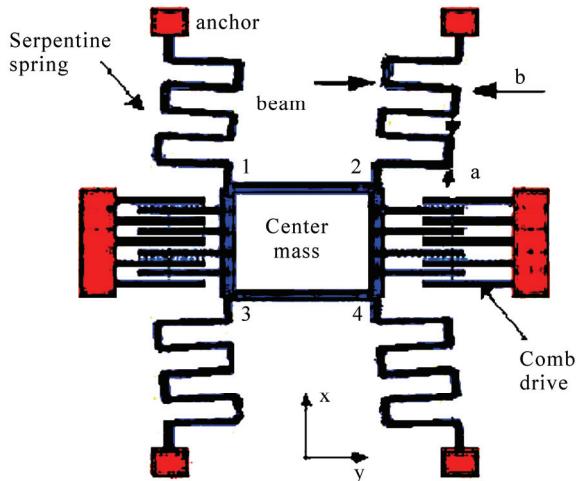


Figure 1. A MEMS resonator with four meandering springs.

Table 1. Design constraints for meandering resonator.

w_{min}	w_{max}	l_{max}	θ_{min}	θ_{max}	N_{max}
2 μm	20 μm	400 μm	-90	90	6

3.2. Synthesis Using Genetic Algorithm

Because this synthesis problem of the serpentine resonator can be regarded as the multi-objective optimization problem in which it is objectives to find values of the design variables so that difference of same behaviors and their expected values achieves it's the minimum values – zeros. To solve this synthesis problem, Dr. Zhou adopted Genetic Algorithm (GA) [9] used in Multi-Objective optimization. The basis of GA synthesis is the coding scheme that represents the features of a problem. Many devices can be represented as a rooted acyclic tree of building blocks. ‘Rooted’ means that there exists a distinguished node as the reference node. All other structures are laid out with respect to this node. MEMS GA building blocks are connected together at their nodes to form a tree. A building block is fully described by the following information:

- A building block type identifier;
- A list of ports or nodes through which this block can be connected to others;
- Governing parameters.

This design of the serpentine resonator is encoded as a 5×3 cell array as follows.

[Center mass]	[1 2 3 4]	[$L \ W$]
[spring block 1]	[1]	[$l_1 \ w_1 \ \theta_1 \ l_2 \ w_2 \ \theta_2 \ \dots$]
[spring block 2]	[2]	[$l_1 \ w_1 \ \theta_1 \ l_2 \ w_2 \ \theta_2 \ \dots$]
[spring block 3]	[3]	[$l_1 \ w_1 \ \theta_1 \ l_2 \ w_2 \ \theta_2 \ \dots$]
[spring block 4]	[4]	[$l_1 \ w_1 \ \theta_1 \ l_2 \ w_2 \ \theta_2 \ \dots$]

Each row of the cell describes one building block. The first column of the cell represents the building block type.

The second column of the cell represents the building block node connection. The third column represents the building block parameters. Crossover and mutations operators are carried out. The arithmetic crossovers are defined as the linear combination of two parents. This crossover ensures that each offspring will maintain the required functional configuration of one mass and four springs. Results of GA synthesis are as follows:

The initial design parameters are: center mass beam $120 \times 12.5 \times 2 \text{ } \mu\text{m}$, each spring beam cross-section $2 \times 2 \text{ } \mu\text{m}$, $a = 60 \text{ } \mu\text{m}$, $b = 80 \text{ } \mu\text{m}$, their material is poly with $2 \text{ } \mu\text{m}$ thickness. The GA configuration parameters were taken as: the population size $n_{pop} = 400$, the number of generation $n_{gen} = 30$, the crossover probability $P_c = 0.7$, the total percentage of elite $P_e = 5\%$, the mutation probability and $P_m = 0.1$, $\lambda = 0.3$. No significant improvement was found in performance after 25–30 iterations taking about ten thousands analyses and 4–5 hours. A Pareto optimal set of 26 designs is found. **Figure 2** shows some of the ‘equally good’ results. Designers will choose the right design from the Pareto set that reflects their multiple objective trade-offs.

3.3. Synthesis Using Simulated Annealing Algorithm

Kamalian etc. adopted the Simulated Annealing Algorithm (SA) [11] to solve the design synthesis problem of the serpentine resonator. As the name implies, simulated annealing exploits an analogy between the way in which a heated metal cools into a minimum energy state and a stochastic optimization algorithm that slowly “lowers the temperature” in stages to eventually “freeze” at the global optimum. SA randomly perturbs a given initial design, whose variations are accepted as the new design with a threshold probability, known as the Metropolis Condition that decreases as the computation proceeds. The slower the rate of probability decrease, the more likely the algorithm is to find an optimal or near optimal solution.

Generally, SA is only applicable to a single objective optimization. It is the design objective for this example to have a lowest natural frequency $f = 93723$.

For the SA implementation, the probability of changing any single beam parameter was set to 10%, as was the probability of adding or deleting a beam segment from each of the four springs. Maximum number of iterations was set to 5000, but a conditional statement can stop the synthesis if the objective error is below an accepted value (frequency error $< 100 \text{ Hz}$).

Using SA, a single objective design result (shown in **Figure 3**) of the meandering resonator was reached quickly, only taking approximately 1 hour. For different starting points, the average time was in the range of 4 to 5 hours. **Figure 4** shows a result with forced Manhattan geometry and symmetry.

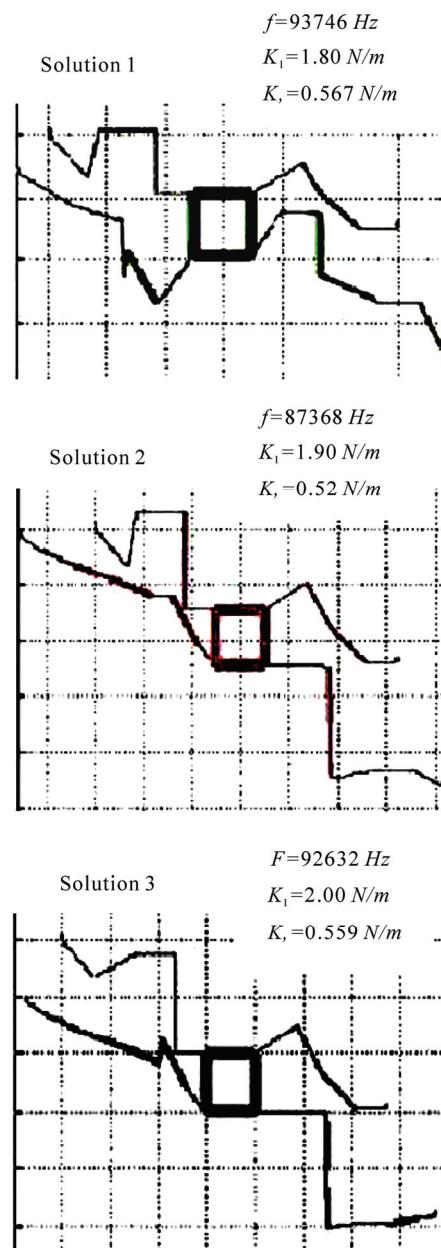


Figure 2. Synthesis results of a serpentine resonator using the Genetic Algorithm (GA).

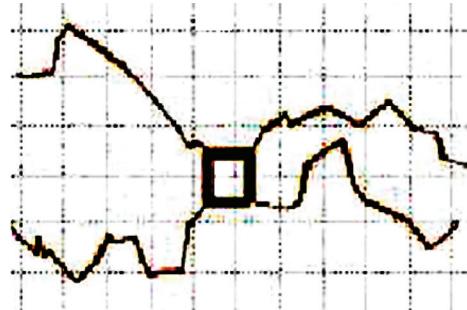


Figure 3. Result 1 of SA.

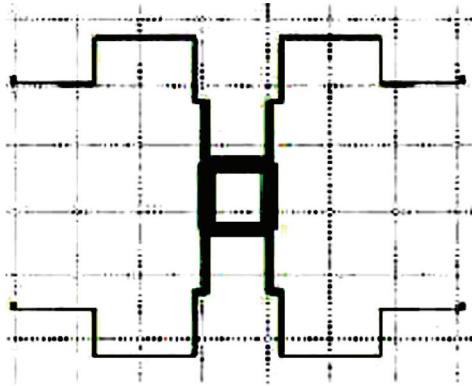


Figure 4. Result 2 of SA.

The results show that Simulated Annealing can in some cases synthesize valid designs faster than Genetic Algorithms. However its dependence on a single objective function and the difficulty in finding the global optimum indicate that it is a less robust method for many MEMS synthesis problems.

4. Design Synthesis Using Direct Solution Algorithms

4.1. Analysis Equations and Design Equations

Regarding the above-mentioned design synthesis problem as a design problem of MEMS devices with expected performance behaviors, Prof. Chen proposes a Direct Solution Algorithm (DSA) of MEMS devices. DSA is a solution technique of equations based on sensitivity analysis.

Analysis Equation (1) of MEMS can be expresses as:

$$\mathbf{M}(\mathbf{X}) \mathbf{q}'' + \mathbf{C}(\mathbf{X}) \mathbf{q}' + \mathbf{K}(\mathbf{X}) \mathbf{q} = \boldsymbol{\Sigma} \mathbf{F}(\mathbf{X}) \quad (7)$$

where \mathbf{q} is the node variables; \mathbf{X} is the design variables describing physical configuration and geometries of MEM design. Mass matrix \mathbf{M} , damping matrix \mathbf{C} , stiffness matrix \mathbf{K} , even force vector \mathbf{F} are functions of design variables \mathbf{X} . Brief expression of (7) is:

$$\mathbf{f}(\mathbf{X}, \mathbf{q}) = \mathbf{0} \quad (8)$$

When \mathbf{X} is given, to find \mathbf{q} , these equations are analysis equations. When \mathbf{q} is given, to find \mathbf{X} , these equations are a base of design equations. 1987, Prof. Chen educed and solved linear design equations of truss structures:

$$\mathbf{D}(\mathbf{q}) \mathbf{X} = \mathbf{F}, \quad \mathbf{X} = [\mathbf{D}(\mathbf{q})]^{-1} \cdot \mathbf{F} \quad (9)$$

A relative paper [13] was embodied by EI 8809-091423.

Any performance behavior of MEMS can be expresses as function of design variables $s_i(\mathbf{X})$. It is task of design synthesis of MEMS to make every behavior $s_i(\mathbf{X})$ equals its expected value s_i^0 . All of L expected behaviors decides the following behavior equation group:

$$\left. \begin{array}{l} s_1(\mathbf{X}) / s_1^0 = 1 \\ s_2(\mathbf{X}) / s_2^0 = 1 \\ \dots \\ s_L(\mathbf{X}) / s_L^0 = 1 \end{array} \right\} \quad (10)$$

Its brief expression is:

$$\mathbf{S}(\mathbf{X}) = \mathbf{I} \quad (11)$$

The behavior equation group of the meandering resonator can be expressed as follows:

$$\left. \begin{array}{l} s_1(\mathbf{X}) = f(\mathbf{X}) / 93723 = 1 \\ s_2(\mathbf{X}) = K_x(\mathbf{X}) / 1.90 = 1 \\ s_3(\mathbf{X}) = K_y(\mathbf{X}) / 0.56 = 1 \end{array} \right\} \quad (12)$$

4.2. Selection of Design Variables

L design variables are enough to solve the equation group composed of L behavior equations. For instance, in behavior equation group (12) of the meandering resonator, L equals 3. So the necessary design variables are few.

L independent continuous design parameters are generally selected as design variables from all design parameters. Design variables selected should be mainly related to the expected behaviors. Other design parameters can be decided according to convenience.

Design variables of the meandering resonator can have many selection schemes. For instance, the first selection is as:

$$x_1 = a = l_{xi}, \dots, l_{xi} \text{ is length of all } x \text{ directional beams;}$$

$$x_2 = b = l_{yi}, \dots, l_{yi} \text{ is length of all } y \text{ directional beams.}$$

$$x_3 = c = w_i, \dots, w_i \text{ is width of all spring beam.}$$

Given parameters are: $N = 6$, $\theta_i \in \{0^\circ, 90^\circ, -90^\circ, 180^\circ\}$. Center mass beams is: $l = 120 \text{ um}$, $w = 12.5 \text{ um}$, $h = 2 \text{ um}$; In Figure 5, an initial design is shown: $x_1 = 60 \text{ um}$, $x_2 = 80 \text{ um}$, $x_3 = 2 \text{ um}$.

The second selection is:

$$x_1 = a = b = l_i, \dots, l_i \text{ is length of all spring beams;}$$

$$x_2 = c = w_i, \dots, w_i \text{ is width of all spring beams;}$$

$$x_3 = \alpha, \dots, \alpha \text{ is spring angle shown in Figure 6.}$$

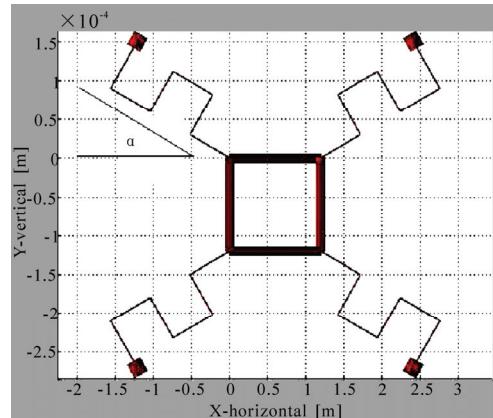


Figure 5. The first section of design variables.

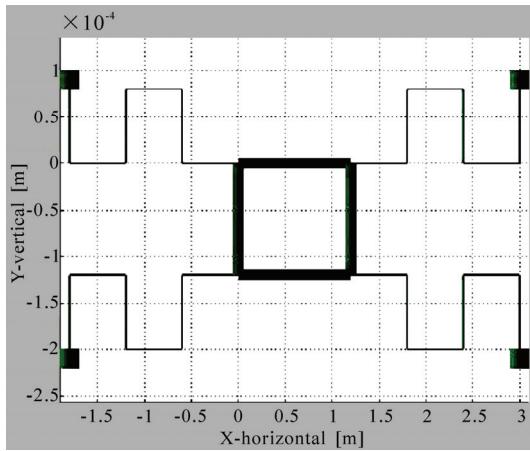


Figure 6. The second section of design variables.

Given parameters are: $N = 6$, $\theta_i \in \{0^0 \pm \alpha, 90^0 \pm \alpha, -90^0 \pm \alpha, 180^0 \pm \alpha\}$. In **Figure 6**, initial design is shown: $x_1 = 60 \text{ um}$, $x_2 = 2 \text{ um}$, $x_3 = 30^0$.

4.3. Solution Algorithm of Behavior Equation Group

Generally behavior equation group is nonlinear; it can be solved using any algorithm of nonlinear equation group. Of course, they can be solved them using Newton method iteratively:

$$\mathbf{X}^{(n+1)} = \mathbf{X}^{(n)} + [\partial \mathbf{S}(\mathbf{X}^{(n)}) / \partial \mathbf{X}]^{-1} [\mathbf{I} - \mathbf{S}(\mathbf{X}^{(n)})] \quad (13)$$

$[\partial \mathbf{S} / \partial \mathbf{X}]$ is Jacobian matrix that is composed of sensitivities *i.e.* derivatives of the behaviors to the design variables.

There two kind of computational methods of sensitivity analysis: analytic sensitivity analysis and difference sensitivity analysis. Analytic sensitivity analysis is often used in structural optimization, is real derivatives of behaviors to design variables. We provisionally adopt difference sensitivity analysis. It is approximate derivatives:

$$[\partial s / \partial x_i] \approx \Delta s / \Delta x_i = [s(x_i + \Delta x_i) - s(x_i)] / \Delta x_i \quad (14)$$

It is merit of difference sensitivity analysis that program is simple and reliable. It is disadvantage that its computational workload is more than analytic sensitivity analysis. Difference sensitivity analysis of every design variable needs a re-analysis.

Flow chart of the direct solution algorithm (DSA) is shown in **Figure 7**.

The total number of necessary analysis of DSA can be calculated with the following formulate:

$$N_a = (1 + N_d) \times N_i + I \quad (15)$$

where N_a is number of necessary analysis; N_d is number of the design variables; N_i is number of necessary iteration.

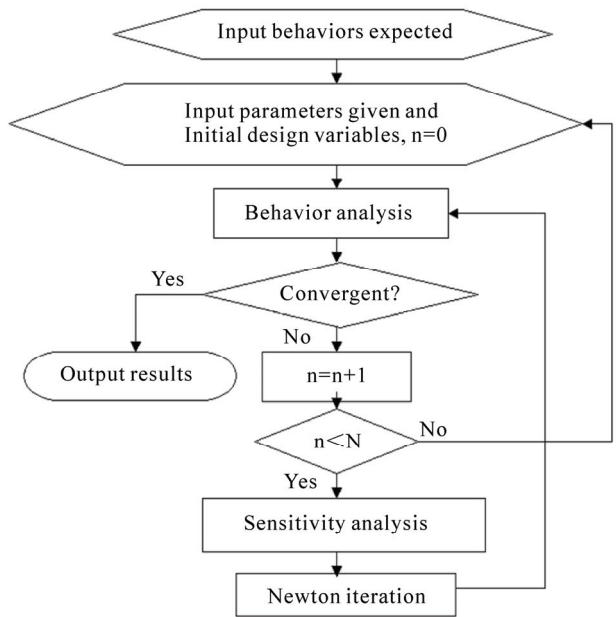


Figure 7. Flow chart of DSA.

4.4. Result of the Meandering Resonator Using DSA

On the basis of SUGAR, program software CSX 1.0 of direct solution algorithm (DSA) of MEMS synthesis has already developed.

For the first selection scheme of design variables, when the initial design is as **Figure 5**, *i.e.* $x_1 = 60 \text{ um}$, $x_2 = 80 \text{ um}$, $x_3 = 2 \text{ um}$. After 4 iterations *i.e.* 17 analysis's, the iteration is convergent. Its synthesis results shown in the up-left figure of **Figure 8** are:

$$x_1 = 295.30 \text{ um}, x_2 = 360.39 \text{ um}, x_3 = 8.7651 \text{ um}$$

If the initial design is as: $x_1 = 20 \text{ um}$, $x_2 = 60 \text{ um}$, $x_3 = 2 \text{ um}$, after 7 iterations, 29 analysis, the iteration is convergent to the above-mentioned results too. Can get the same result from different starts, Robustness of DSA is validated. The behaviors of the above-mentioned results equal their expected values exactly. Relative computational error is less than 0.0001:

$$f = 93723 \text{ Hz}, K_x = 1.90 \text{ N/m}, K_y = 0.56 \text{ N/m}$$

In **Figure 8**, up-right, down-left and down-right figure is iteration history of frequency, x-stiffness and y-stiffness respectively.

For the second selection scheme of design variables, the initial design is as **Figure 6** *i.e.* $x_1 = 60 \text{ um}$, $x_2 = 2 \text{ um}$, $x_3 = 30^0$. After 5 iterations *i.e.* 21 analysis's, the iteration is convergent. Its synthesis results are shown **Figure 9**. If the initial design is as: $x_1 = 60 \text{ um}$, $x_2 = 2 \text{ um}$, $x_3 = 0^0$, after 6 iterations, 25 analysis, the iteration is convergent to the above-mentioned results too. The behaviors of the above-mentioned results equal their expected values exactly. Relative computational error is less than 0.0001:

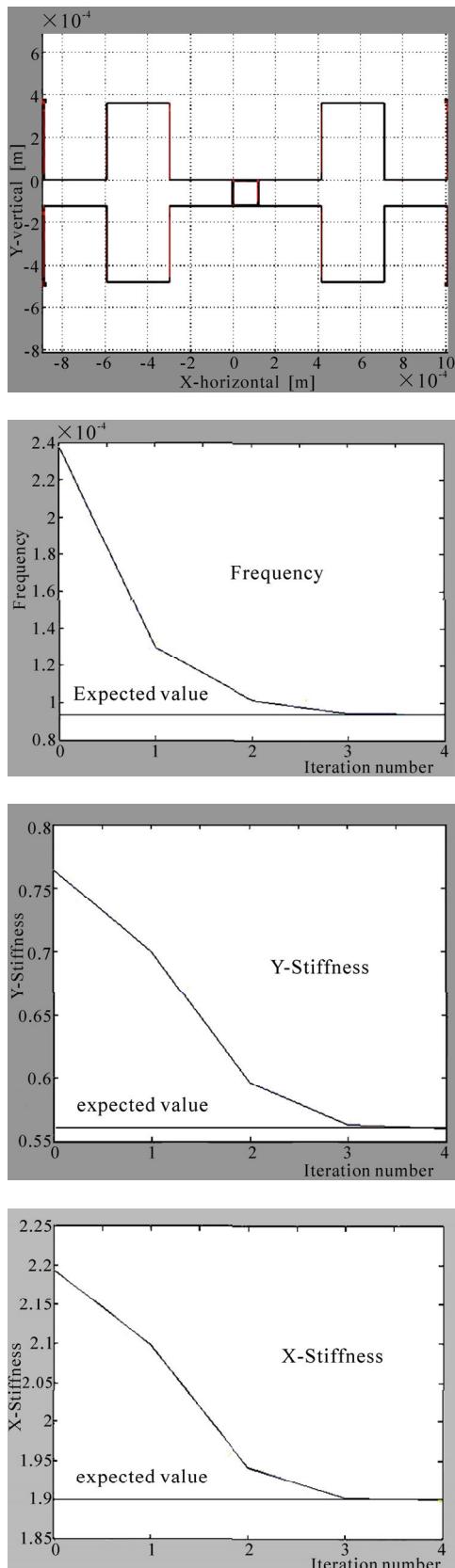


Figure 8. Synthesis result of design variable scheme 1.

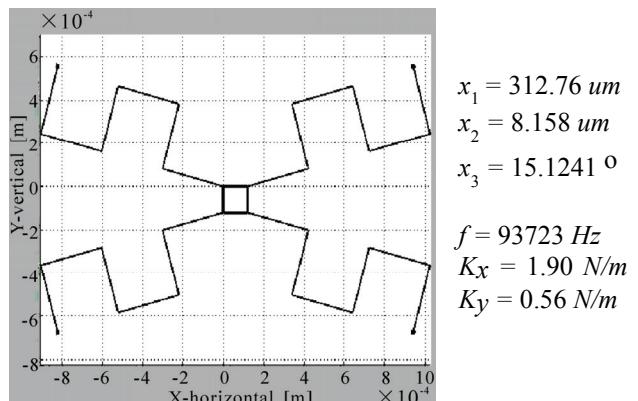


Figure 9. Synthesis result of design variable scheme 2.

For the meandering resonator, computational time of a synthesis is less than 30 seconds, using Pentium III PC.

5. Discussion

While MEMS designers often resort to back-of-the-envelope calculations, SUGAR provides a quick effective analysis tool. Development of synthesis algorithms provides tools of automatic design of MEMS devices.

However, Genetic Algorithm, Simulated Annealing Algorithm, Nerve Net Algorithm, and Mini Wave Algorithm etc. can only be called as Simulated Nature Algorithms (SNA). It is main merit of SNA that objectives and constraints functional relationship and sensitivity analysis are not necessary. SNA is more applicable to global optimization and topology optimization etc. complex optimization problems. When it is difficult to solve these complex optimization problems using traditional algorithms based on sensitivity analysis, SNA have to be adopted. Deathful disadvantage of SNA consists in its enormous computational workload. In design synthesis of MEMS, objectives and constraints functional relationship and sensitivities of few design variables can be gotten. So algorithms based on sensitivity are more applicable to it. In engineering example of the meandering resonator, it is shown that Direct Solution Algorithm proposed here is very effective to design synthesis of MEMS. Its main merit is that computational workload is greatly decreased from thousands analysis's to less than 30 analysis's, computational time is greatly decreased from hours to less than 30 seconds.

In design synthesis problem of MEMS with expected performance behaviors, objective extreme is known beforehand. So the design synthesis problem of MEMS is a solution problem of equation group rather than it is an optimization problem. In design of MEMS, if need get conditional extreme of some behaviors, it becomes optimization problem, e.g. minimize the area of an electrostatic actuator. For an optimization problem of MEMS, there many optimization algorithms based on sensitivity

can be adopted. Guide-Weight Method [14] proposed by Prof. Chen is very effective one of them, and has already been adopted on electronic mechanical, civil, aeronautical and astronautically structural optimization.

6. References

- [1] N. Zhou and M. Agogino, "Automated design synthesis for Micro-Electro-Mechanical Systems (MEMS)," The Proceedings of the ASME Design Automation Conference, Montreal, Canada, 2002.
- [2] Q. Jing, H. Luo, T. Mukherjee, L. R. Carley, and G. Federer, "CMOS micro-mechanical bandpass filter design using a hierarchical MEMS circuit library," Proceedings IEEE Thirteenth Annual International Conference on Micro Electro Mechanical Systems, Miyazaki, Japan, pp. 187–192, 2000.
- [3] E. Goldberg, "Genetic algorithms in search, optimization, and machine learning," Addison-Wesley, 1989.
- [4] H. Li and E. Antonsson, "Evolutionary techniques in MEMS synthesis," Proceedings of DETC'98, 1998 ASME Design Engineering Technical Conferences, Atlanta, GA, 1998.
- [5] T. Lacksonen, "Empirical comparison of search algorithms for discrete event simulation," Computers & Industrial Engineering, Vol. 40, No. 1–2, pp. 133–48, June 2001.
- [6] N. Zhou, V. Clark, and K. Pister, "Nodal simulation for MEMS design using SUGER v.0.5," In 1998 International Conference on Modeling and Simulation of Microsystems Semiconductors, Sensors and Actuators Santa Clara, CA, April 6–8, pp. 308–313, 1998.
- [7] J. Clark, N. Zhou, S. Bhate, Z. Bai, J. Demmel, and K. Pister, "Sugar: Advancements in a 3D multi-domain simulation package for MEMS," In Proceedings of the Microscale Systems: Mechanics and Measurements Symposium, Portland, OR, June 4, 2001.
- [8] J. Clark, N. Zhou, D. Bindel, L. Schenato, W. Wu, J. Demmel, and K. Pister, "3D MEMS simulation modeling using modified nodal" In Proceedings of the Microscale Systems: Mechanics and Measurements Symposium, pp. 68–75, 2005.
- [9] N. Zhou, B. Zhu, A. Agogino, K. Pister, "Evolutionary synthesis of micro-electro-mechanical systems (MEMS)," Proceedings of ANNIE 2001: IEEE Neural Networks Council and Smart Engineering Systems Laboratory. ASME Press, St. Louis, Missouri, Vol. 11, pp. 197–202, November 4–7, 2003.
- [10] S. Narayanan and S. Azarm, "On improving multi-objective genetic algorithms for design optimization," Structural Optimization, Vol. 18, pp. 146–155, 1999.
- [11] S. Chen, "Design & optimization of truss with expected deformed shape," China Civil Engineering Journal, Vol. 20, No. 2, pp. 1–9, 1987.
- [12] S. Chen and S. Ye, "A guide-weight criterion method for the optimum design of antenna structures," Engineering Optimization, Vol. 10, No. 3, pp. 199–127, 1987.
- [13] S. Chen, "Modern design methods of precise complex structures," Press of Beijing University of Aeronautics and Astronautics, Beijing, 1992.