Automated Mask-Layout and Process Synthesis for MEMS

Lin Ma and Erik K. Antonsson*

California Institute of Technology
*Mail Code:104-44, 1200 E. California Blvd
Pasadena, CA 91125, USA, erik@design.caltech.edu

Abstract

This paper presents a method for automated mask-layout and process synthesis for MEMS. The synthesis problem is approached by use of a genetic algorithm. For a given desired device shape, and several fabrication process choices, this synthesis tool will produce one or more mask-layouts and associated fabrication process sequences (which when used can generate shapes close to the desired one). For complicated device shapes and wide range of fabrication process possibilities, it is hard for the designer to produce the right mask-layout and fabrication procedure by experience and trial and error. An automated synthesis tool like this will benefit the MEMS fabrication and synthesis by both eliminating the expense and time of the design and increasing the design accuracy.

Keywords: MEMS, design, synthesis, genetic algorithm, bulk wet etching

1 INTRODUCTION

Though there have been many remarkable advances made in microelectromechanical systems (MEMS) design and fabrication during the past decade, as the complexity of MEMS devices and the scope of MEMS applications increases, the need for structured design methods also increases. Many of the most interesting and useful mechanical devices intrinsically rely on 3-dimensional behavior and 3-dimensional shapes. Complex 3-dimensional shapes can be generated using fabrication procedures such as multiple process wet etching, but the mapping from the mask-layout to the final shape is usually non-intuitive and complicated. For a given 3-D device shape, it is difficult for the designer to produce a proper mask-layout as well as correct fabrication procedures by experience and trial and error. An automatic design tool which can automate the shape-to-mask-and-fabrication-process synthesis will be helpful to the development of new MEMS devices, since with its help the designer can be relieved from considering the details of fabrication and instead can focus on the functional design of the devices. Previously reported work [7] utilized a genetic algorithm to synthesize mask-layouts for a single process. Here we introduce an approach to realize mask-layout <u>and</u> fabrication process synthesis through a genetic algorithm.

For our problem of mask and process synthesis, a genetic algorithm iteration loop is constructed to evolve the optimal mask and process sequence as shown in Figure 1. An initial population of solution candidates (mask-layouts and process parameters) are randomly generated. The fabrication of each mask-layout using the associated process parameters is then simulated by a specified fabrication simulator to produce a 3-dimensional shape. The performance of each solution candidate is measured through the shape comparison between the produced shape and the user defined final shape. During each evolutionary loop, genetic operations are applied to control the stochastic searching behavior such that the best performing solution candidates are more likely to survive and evolve even better children for the next generation. The iteration is stopped when one satisfying solution is found.

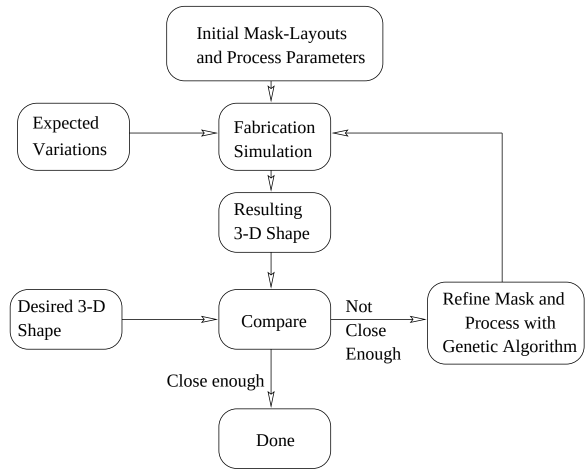

Figure 1: A schematic representation of an genetic algorithm MEMS synthesis technique.

2 THE GENETIC ALGORITHM

Genetic algorithms are a global stochastic optimization technique which is based on the adaptive mechanics of natural selection evolution [5]. The algorithm maintains a population of solution candidates. Each individual in the population is encoded into a string of characters or digits, through which the original solution space is converted to an encoded space, which is easier for the algorithm to search for the global optimum. The genetic algorithm works as an iteration loop. First, an initial population is generated randomly. Then all the individuals in the initial population are evaluated by an objective function measurement program (which is problem specific), and a performance value (called a fitness value or FV) will be calculated for each individual. Then the whole population goes through genetic operations such as selection, crossover, etc., to form individuals of the next generation. Because of the strong converging characteristics of GA, the new individuals will generally have better performance than the ones in last generation. Such iteration continues until an individual is found whose performance is good enough, and this individual will be taken as the solution. In general, genetic algorithms are a non-problem specific technique which can be applied to virtually any optimization problem if its objective function measurement is available. What makes genetic algorithms distinct from other stochastic optimization techniques is the involvement of a generation of individuals. Such participation provides efficient solution information which can be used by genetic operations to control the bias for the searching region as well as increase the robustness to overcome deception traps during the searching. Genetic algorithms have been shown to successfully solve problems in various complex domains [3].

In our problem, the solution space is comprised of a mask-layout space and a fabrication process space. Mask-layouts are treated geometrically as 2-dimensional simple polygons. In the initial test stage, we keep the process space relatively simple. For example, for a multiple process wet etching application, the process space will be all the fabrication procedures that can be used, and each individual solution candidate includes parameters such as the etchant number(s) to be used, the etch time duration value for each applied etchant, *etc*. The process space is application specific, and is simpler than the mask-layout space. The mask-layout space is more complex and therefore requires more attention during encoding and manipulation, which are described below.

2.1 Coding Scheme

An appropriate coding scheme is needed to encode a 2-D polygon to a string which can be easily manipulated by genetic operations. There are many ways to describe a 2-D polygon. Here we chose edge length and edge directional angle (turning angle) to describe each edge of a polygon [1], [7]. A mask polygon is encoded into two real strings. One string contains edge directional angles and the other contains edge lengths. The size of each string is equal to the number of polygon sides. Two elements from each string with the same element position describe an edge of the polygon. In GA's, the physical objects in the solution space which need to be encoded are called phenotypes, and the encoded strings are called genotypes. The searching and manipulation are conducted on genotypes. The two real strings are genotypes for the mask-layouts in our GA. A schematic illustration of the coding scheme is shown in Figure 2.

There are generally two kinds of coding schemes, binary coding, and real coding. We chose real coding scheme

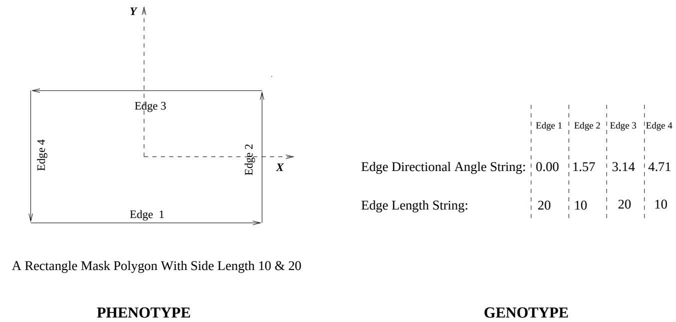

Figure 2: A schematic illustration of the GA coding scheme.

because it provides adequate precision with a short string length. The edge length and turning angle representation of 2-D polygon provides a convenient way to design the associated crossover and mutation schemes.

2.2 Selection Scheme

The core of a selection scheme is the method to assign sampling rate to each individual in the generation [4]. The sampling rates decide the possibilities for each individual to be selected for crossover, and the individual with larger sampling rate has more chance to survive and evolve. In our application, rank-based selection algorithm is chosen to assign the sampling rates based on individual's rank among the whole population [2] through the following formula:

For individual whose rank is i,

$$\text{Sampling rate} = 1.0 + \text{bias} - 2 * \text{bias} * i/(\text{size} - 1)$$

$$\begin{array}{rcl} i & = & \text{individual's rank, from } 0 \text{ to size-1} \\ \text{size} & = & \text{population size} \\ \text{bias} & = & \text{extra fitness awarded to} \\ & & \text{the top-ranked individual} \\ & & \text{compared to the average-ranked} \\ & & \text{individual whose fitness is } 1.0 \end{array}$$

The bias value is in general between 0.0 and 1.0, and it can be used to control the selection pressure. In our application, the bias value is gradually increased as the genetic iteration proceeds to control the rate of convergence. During the initial stage the selection pressure is light so that premature convergence can be prevented, and during the final stage the selection pressure is heavy so that the population will effectively converge to the final result.

2.3 Crossover and Mutation

The genetic operators are applied to genotypes in the GA. The mask genotypes in our problem are the two real strings of edge directional angles and edge lengths. Two genotypes are randomly chosen paired according to the sampling rates for each genotype. Simple crossover is separately applied to the pair of edge directional angle strings and the pair of edge length strings. For each string pair, a random site is chosen among the string bits and string elements right to the random site are exchanged. Two new children are generated and become the members of the next generation. Mutation is applied on string element basis. If mutation is applied to a string element, the element value will be set as a real random number uniformly generated between a specified lower and upper bounds. Typically, crossover serves as the major genetic operator and the crossover rate is set to 0.6. The usage of mutation is highly restricted by setting the mutation rate to 0.005, and each time only one element (at most) of a string is allowed to be mutated.

3 APPLICATION

A specific application of our method on multiple process wet etching synthesis has been conducted. The bulk wet etching mask-layout and process synthesis was chosen as an example because of its high level of geometric complexity. For bulk wet etching, there is no simple geometrical correlation between the masks and process sequences and the evolved 3-D structures. A multiple process wet etching simulator called SEGS [6] was used as the fabrication simulation and performance evaluation program.

The synthesis problem is stated as follows. A desired device shape is given, and we have three different etchants that can be chosen. The fabrication procedures allowed here are two wet etching steps, which means two etchants can be applied sequentially, each for a predetermined time duration, to generate the final shape. The goal is to find a mask-layout, and the order and time duration of the two etchants to use to produce a desired 3-D shape. A multi-process wet etching procedure can generate highly geometrically complicated shapes and will require automated design tools. The two-process example shown here illustrates the feasibility of our approach.

3.1 Shape Comparison

We need a way to calculate the closeness between the evolved 3-D shape and the specified target shape. The shapes are represented by a stack of polygonal layers. The etching simulation program ensures that the polygon layers of both shapes vertically match. The mismatch between two 3-D shapes (contours of the target shape and a candidate evolved shape at the same vertical position) is decomposed into pure shape mismatch and size mismatch. For these two polygons, we calculate the size mismatch as the length ratio of the two polygons. The pure shape mismatch between two polygons is obtained using the method introduced by Arkin *et al*. [1]. The pure shape mismatch of the two shapes is calculated as the *weighted* sum of the mismatch between the two polygon layers in each vertical level. The weights need to be carefully chosen such that the important shape information describing the overall vertical characteristics of the shape which can not be found from the shape mismatch of two polygon layers is conserved. We use the following formulas to calculate the pure shape mismatch and size mismatch between the two 3-D shapes:

$$\mathrm{SizeMismatch} = \sum_{i=1}^{n} \frac{\mathrm{SiM}(i)}{n}$$

$$\mathrm{PureShapeMismatch} = \sum_{i=1}^{n} \frac{\left(\mathrm{ShM}(i) * \frac{\mathrm{SiM}(i)}{\mathrm{ASiM}}\right)}{n}$$

where n is the total number of layers, and ShM and SiM are pure shape mismatch value and size mismatch value of two polygon layers in the same level, and ASiM is average size mismatch value of all layers.

Using shape mismatch and size mismatch to represent the closeness between two 3-D shapes has a disadvantage: the difference between the side wall slopes of the two shapes is not taken into consideration. For some devices with small z-direction dimension compared with xy-plane dimensions, two shapes with similar xy-plane cross section, but different side wall slopes, may be considered very close using our shape and size mismatches criteria, but the difference in the side wall angles of these two shapes may make one device an unacceptable replacement for the other. Due to the preceding consideration, a scheme to represent side wall slope using contours of a shape is constructed. For two contours with a vertical distance of h,

$$\mathrm{SideWallSlope} = \mathrm{ArgTan}\left(\frac{h * 2\pi}{\mathrm{L_{upper}} - \mathrm{L_{lower}}}\right)$$

where $\mathrm{L_{upper}}$ and $\mathrm{L_{lower}}$ are perimeter lengths of the upper layer and lower layer contours. For two shapes, we construct the slope mismatch at a particular depth as the difference between the side wall slopes at this depth calculated as shown above. Then the average of the slope mismatches at all depths is considered as the overall slope mismatch between these two shapes.

3.2 Fitness Evaluation

The fitness value of each individual is evaluated in the following steps:

1. The 3-D etching simulation (SEGS [6]) is run on each candidate solution. Each candidate solution consists of a mask polygon and process parameters including two etchant numbers and two etch time duration values. The simulation is run using the mask, two etchants for those two etch time durations, and as a result we get an evolved 3D shape represented by a stack of polygonal layers.

2. The pure shape mismatch value, size mismatch value, and slope mismatch value between each of the evolved 3D shapes and the specified target shape are calculated. These values are stored as the objective function values of each individual in the generation.

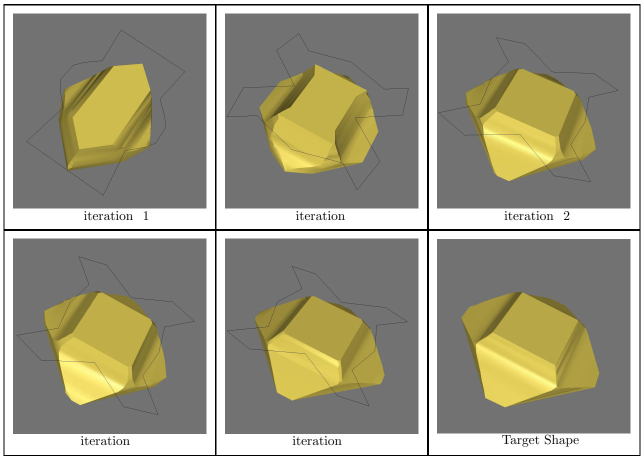

Figure 3: Mask-layouts and evolved shapes

3. After obtaining the objective function values of all the individuals in the generation, the fitness values are calculated from the objective function values. This is a typical multiple objective optimization problem, in which we need to construct an overall performance value from three different objectives. We tried several different ways to construct the fitness value, and the non-compensating maximum-of-all scheme seems to work best:

$$\mathrm{itness}(i) = \mathrm{Ma}\left(\frac{\mathrm{AShM}}{\mathrm{ShM}(i)}\ \frac{\mathrm{ASiM}}{\mathrm{SiM}(i)}\ \frac{\mathrm{AS1M}}{\mathrm{S1M}(i)}\right)$$

where ShM, SiM and SlM are shape, size, and slope mismatch values of individual i, and AShM, ASiM, and ASlM are the respective average values of the generation.

3.3 Result

The results of our synthesis application are shown in Figure 3 and Table 1. In Figure 3, the last frame shows the target shape, and frame 1-5 shows the best candidate mask-layouts (the black polygons) at five different iterations during the synthesis loop, and the simulated 3-D shapes. The convergence of the evolved shapes to the target shape can be easily observed. Detailed information is shown in Table 1. We can see that the shape, size and slope mismatch values between the simulated shape and the target shape for each iteration are getting smaller as the synthesis proceeds, and the etchant numbers and etch times are converging to 2, 1 and 6, 5 respectively. By iteration 46, though the mask shape is nonintuitive (actually the mask shape shows corner compensation), the resulting shape closely approaches the target shape.

4 CONCLUSIONS

A mask-layout and fabrication process synthesis technique combining a genetic algorithm and forward fabrication simulation is introduced here, and its feasibility is shown by an application of multiple process wet etching synthesis. Successful results have been obtained for challenging tests. For now, the edge number of the masks for a synthesis task is fixed during the evolvement and is set by the user, a modification of the program so that it can handle the evolvement and interaction between masks of different edge numbers is under way. As an extension of these methods, we plan to introduce expected variations into the fabrication simulation (*e.g.*, mask mis-alignment and process variations). By combining these variations into the synthesis iteration loop, the stochastic optimization will produce mask-layouts and process sequences that are least sensitive to these variations. Such robust designs will greatly benefit real device development.

teration	tchant o.s	tch Times	Shape Match	Size Match	Slope Match
1	2 1		0.	0.1 2	0.210
	2 1		0.1 00	0.0	0.1 2
2	2 1		0.0 2	0.0	0.0 1 2
	2 1		0.0 1	0.020 1	0.0 2
	2 1		0.0 2	0.00 0	0.021

Table 1: Synthesis data showing convergence

REFERENCES

[1] Esther M. Arkin, L. Paul Chew, Daniel P. Huttenlocher, Klara Kedem, and Joseph S. B. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:209–215, 1991.

[2] James E. Baker. Adaptive selection methods for genetic algorithm. In John Grefenstette, editor, *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, pages 101–111, 1985.

[3] David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[4] John J. Grefenstette and James E. Baker. How genetic algorithms work: A critical look at implicit parallelism. In *The Third International Conference on Genetic Algorithms*, 1989.

[5] John H. Holland. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, 1975.

[6] Ted J. Hubbard and Erik K. Antonsson. Design of MEMS via Efficient Simulation of Fabrication. In *Design for Manufacturing Conference*. ASME, August 1996.

[7] Hui Li and Erik K. Antonsson. Mask-Layout Synthesis Through an Evolutionary Algorithm. In *MSM'99, Modeling and Simulation of Microsystems, Semiconductors, Sensors and Actuators*, San Juan, Puerto Rico, April 1999. IEEE.