Evolving MEMS Resonator Designs for Fabrication

Gregory S. Hornby[1], William F. Kraus[2] and Jason D. Lohn[2]

[1] U.C. Santa Cruz, Mail Stop 269-3, Moffett Field, CA 94035, USA,
`Gregory.S.Hornby@nasa.gov`,
`http://ti.arc.nasa.gov/people/hornby/`

[2] Carnegie Mellon University, Building 221, Moffett Field, CA 94035, USA,
`Jason.D.Lohn@nasa.gov`

Abstract. Because of their small size and high reliability, microelectromechanical (MEMS) devices have the potential to revolution many areas of engineering. As with conventionally-sized engineering design, there is likely to be a demand for the automated design of MEMS devices. This paper describes our current status as we progress toward our ultimate goal of using an evolutionary algorithm and a generative representation to produce designs of a MEMS device and successfully demonstrate its transfer to an actual chip. To produce designs that are likely to transfer to reality, we present two ways to modify evaluation of designs. The first is to add *location noise*, differences between the actual dimensions of the design and the design blueprint, which is a technique we have used for our work in evolving antennas and robots. The second method is to add *prestress* to model the warping that occurs during the extreme heat of fabrication. In future we expect to fabricate and test some MEMS resonators that are evolved in this way.

1 Introduction

Microelectromechanical systems (MEMS) are devices that generally range from 20 micrometers to 1 millimeter. These micro-sized devices are of great interest in the aerospace community because of their small size and high reliability [13]. Along with the desire for improved performance and increased complexity of MEMS devices, there is a corresponding need for automated design and optimization methodologies that allow a designer to explore more of the potential design space.

Already automated design systems based on evolutionary algorithms (EAs) have been used for antennas [8], flywheels, load cells [15], trusses [12], robots [4], and more [1, 2]. In the domain of MEMS, there have been examples of using EAs for automated mask-layout and process synthesis [6, 7, 10] and to produce the designs of simulated MEMS devices [5, 16–18]. Both Kamalian et al. [5] and Zhou et al. [16, 17] employed an EA to optimize the size, orientation and number of segments that make up the legs of a meandering resonator. While the simulated performance of the evolved designs was impressive, the designs themselves were characterized by acute angles between the beams comprising the resonator legs, making them unsuitable for fabrication. In addition, constraints placed on intersections between beams meant that not all genotypes produced valid designs, reducing the overall effective population size. For this work they used an open-source MEMS simulation system called SUGAR and did not fabricate evolved designs.

Previously we evolved a MEMS meandering resonator using a generative representation [9] and compared our evolved solutions with that of other work [5, 16, 17]. In this paper we describe our work toward evolving MEMS designs for fabrication. To improve the likelihood of a successful transfer to reality, we present two methods for evaluating candidate designs. The first method is to include fabrication noise in evaluation and this consists of evaluating a design multiple times with perturbations in the design to reflect the slight errors in where parts and edges are located when a design is manufactured. We have used this technique previously in evolving antennas and robots in simulation that have been successfully transferred to reality [4, 8]. The second method is a kind of domain specific version of manufacturing noise that is specific to the domain of MEMS devices and consists of adding the effects of prestress to a design to model the warping that occurs during the extreme heat of fabrication. Using combinations of with/without these two types of manufacturing noise we have evolved designs of a MEMS resonator. Having reached this stage, the next step for future work is having some of our evolved designs fabricated and then testing these designs to determine how well they match the simulated results.

2 Method

To demonstrate the effectiveness of using EAs to automatically design MEMS devices we selected the problem of producing the design of a 2D meandering resonator. A meandering resonator consists of a central mass suspended by up to four 'legs,' each leg is comprised of a series of beams terminated by an anchor. This device is a useful first-step toward demonstrating the use of automated design algorithms for producing MEMS designs since it is fairly simple to fabricate and does not require extensive circuitry to test. Further, a MEMS resonator is a step on the path toward producing a MEMS gyroscope, which would be of great use for NASA missions. In the rest of this section we describe the generative representation for encoding MEMS devices, the method by which a MEMS design was tested and evaluated, and the EA used for the design automation.

2.1 Generative Representation

A MEMS resonator consists of a proof mass and its supporting legs. Here we fixed the size of the proof mass to a 492 um square and evolved the design for the legs supporting it. Specifically, a generative representation was used to encode the construction of a single leg design and then four copies of this were attached to fixed points on the proof mass. At the other end of each "leg" a 100 um square anchor was placed. The entire device was assumed to have a fixed thickness of 25 um and to have a Silicon-On-Insulator (SOI) material composition.

The generative representation for a resonator's legs consisted of a variable length sequence of leg-construction operators. These operators are: `go_right`, `go_left`, `go_straight`, `go_straight_w`, `terminate`, `begin_branch`, `end_branch`, `begin_repeat`, and `end_repeat`. The operators `go_straight_w`, `go_left`, and

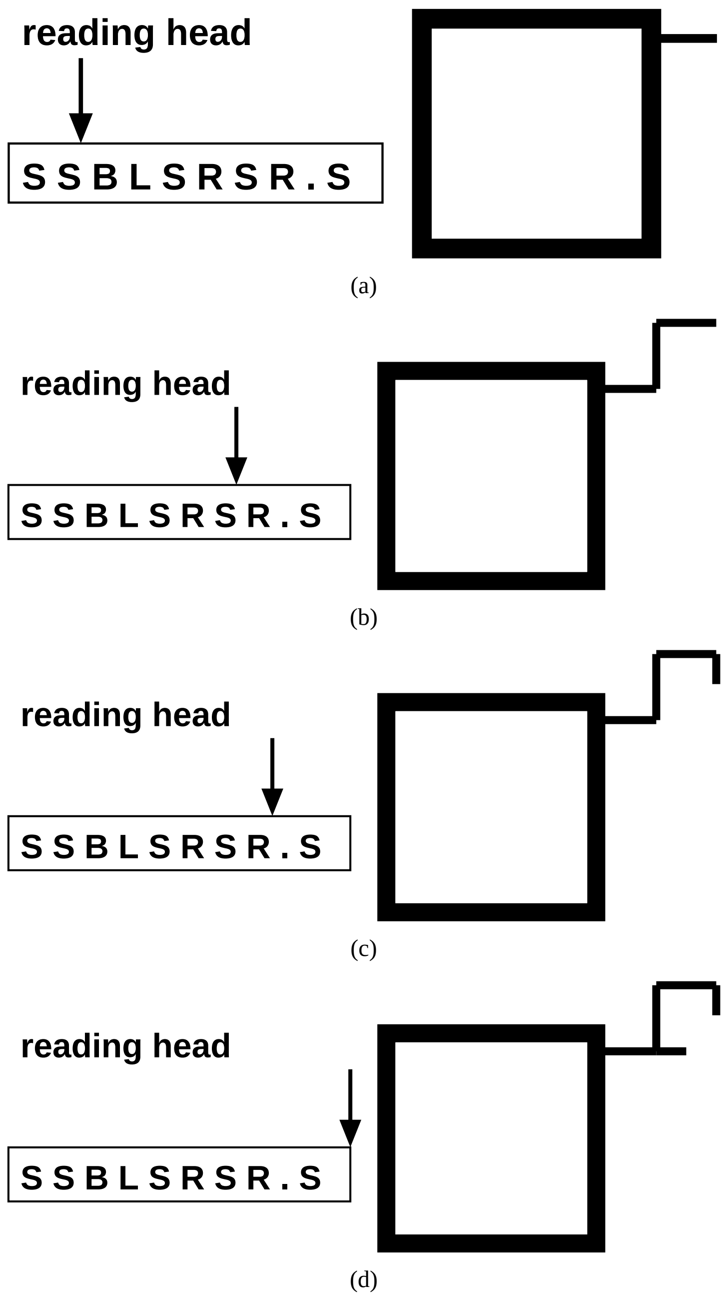


Fig. 1. The "legs" of the resonator were constructed by sequentially reading operators in the evolved encoding and executing them. Once one leg was produced, three additional copies were placed about the central mass.

`go_right` each contain a 4-bit beam width specification, which was allowed to vary between 4 and 20 um. The operator `go_straight` does not allow for a width change and was biased to have a higher frequency than `go_straight_w` since excessive changes in beam width are less amenable to fabrication. In order to maintain Manhattan (right angle) geometry, these construction operators create extensions straight ahead, turn right or turn left. Each leg was constructed by executing the sequence of operators, resulting in the outward construction of a leg from a predetermined corner of the center mass. Construction would end when the reading head encountered either the end of the operator sequence or a `terminate` instruction. Once all the operators are executed, a 100 um square anchor is placed at the distal terminus of the leg and then three additional copies of this leg are place at the other corners of the proof mass.

The initial frequency of each instruction in the population, as well as the likelihood that any given byte would mutate into a given instruction, was arbitrarily set at 0.5625 for go_straight instruction, and 0.0625 for all other instructions. Note that the higher combined probability of the go_straight and go_straight_w instructions meant that any given section of a leg was more likely to go straight than to turn. Also, preliminary work indicated that good designs were more likely to be achieved if there was a bias towards left turns, hence, the initial population (and any subsequent mutations) was so biased.

In the process of executing a leg-construction program, there are some conditions under which operators were ignored. Operators that will create a collision with the central mass or trace over an existing pather were ignored. But operators that create loops resulting from the intersection of two beams at right angles were allowed. To meet the size constraints of possible fabrication processes, the topology of an individual was constrained to a square area 3.5 times the width of the proof mass and if, while constructing the leg, the boundary of this area is encountered, the instruction was ignored. In addition, legs were not allowed to extend into the area where the finger structures resided.

An example of creating a design using this language for constructing resonator legs is shown in the sequence of images in Figure 1 and is the result of executing the following sequence of operators:

```
go_straight go_straight begin_branch go_left go_straight
go_right go_straight go_right branch_end go_straight
```

The first two operators will create an initial extension for the leg, Figure 1(a). Next, the `begin_block` operator marks this location as a branch point. Executing the operators `go_left go_straight go_right go_straight` adds two more segments, Figure 1(b). The next operator, `go_right`, adds another segment to the leg, Figure 1(c), and then the `end_branch` operator moves the location for new construction back to the branch point. Finally, the execution of `go_straight` adds a branched segment to the leg and results in the design shown in Figure 1(d).

Randomly created individuals consisted of a sequence of 20 random operators and this size was allowed to vary through recombination. In addition, because the instruction set constrained the orientation of beams to right angles, all evolved devices exhibited Manhattan geometry. This not only simplified the beam intersection tests, but also produced designs that were more amenable to fabrication, avoiding the potential kinks and sharp acute angles found in previous evolved designs [5, 16–18].

2.2 Cost Function and Evaluation

To evaluate resonator designs, ANSYS Multiphysics is used to model them. After a given individual's encoding is converted into a geometry, this design is translated into a set of ANSYS commands specifying the device structure and the material properties as well as the fixed anchor locations. Each time an individual was evaluated, an ANSYS geometry file would be created and a command line version of ANSYS would be invoked by the EA to evaluate it. At the end of an analysis, ANSYS writes out the resonance frequency and displacement normal of the center mass, which was then read in by the EA and this analysis was used to calculate the cost value for the individual.

To reduce the amount of computational time necessary to evaluate a given design, a simplification was made to the non-evolved part of the device. Resonator designs are intended to have 100 fingers attached to the proof mass, with 50 fingers on opposite sides. On the simulated designs, this was reduced to 10 fingers, with 5 on each side. So that this change in the number of fingers would have minimal impact on the analysis, the area of each finger was increased to maintain the same mass and moment of inertia as the fully specified, 100 finger case. With this simplification, simulating a single MEMS resonator design took approximately 30 seconds and one of our evolutionary runs took between 1 to 2 days.

Once simulated, a design was evaluated using the following cost function:

$$\text{Score} = \frac{|F_o - F_t|}{F_t} + \frac{1 - D_Y}{\sqrt{D_X^2 + D_Y^2 + D_Z^2}} \tag{1}$$

Here, F_t is the target frequency (10 kHz), F_o is the observed frequency, and D_X, D_Y, and D_Z are the observed axes displacements. With this cost function, scores can range from 0.0 (ideal) to 2.0 (worst).

2.3 Location Noise and Prestress

One of the main challenges with evolving physical designs in simulation is setting up the system such that evolved designs successfully transfer to the real world. Since there is no prior work demonstrating this, the objective of this work was to test different strategies to determine the best approach for producing designs that successfully transfer. To improve the likelihood that evolved designs would work in reality comparable to how they are predicted to perform by the modeling system, we tried two different approaches of putting manufacturing variability in evaluating a design.

In fabricating a MEMS device there is some location noise between where an edge is located in the design and where it ends up occurring in the physical device. Previously, we have had experience in evolving physical objects and have overcome this manufacturing noise by including it in part of the evaluation [4, 8]. To include location noise in the evaluation of an individual, its design is evaluated multiple times, each time with a slight variation to it based on a model of the expected variation in manufacturing or fabrication. The assigned cost of an individual is the worst score of the different trials and provides a lower-bound expectation of what to expect should this design actually be manufactured.

Here we implemented our noise model by testing newly created individuals three times, each time with a different variation in fabrication. Existing, high quality individuals that are still in the population are regularly re-tested since it is desirable to get a more accurate estimate of the best designs but only necessary to get a rough estimate of the less promising designs.

In addition to variation in the location of edges, another variability that occurs in the MEMS fabrication process is a kind of warping, or stress, on the chip due to the heat of the fabrication process. To mimic this deformation due to fabrication (*prestress*), we used ANSYS to create a temperature gradient across the Z-axis of the structure to create the required prestress deformation.

Evaluation with prestress and location noise both increased evaluation time significantly. While a typical MEMS design is evaluated in about 30s, adding the prestress calculation doubled that time to roughly 60s. Adding location noise to the evaluation process meant multiple simulations, and this increased evaluation time linearly: 3 simulations of the device took 3 times as long as a single simulation.

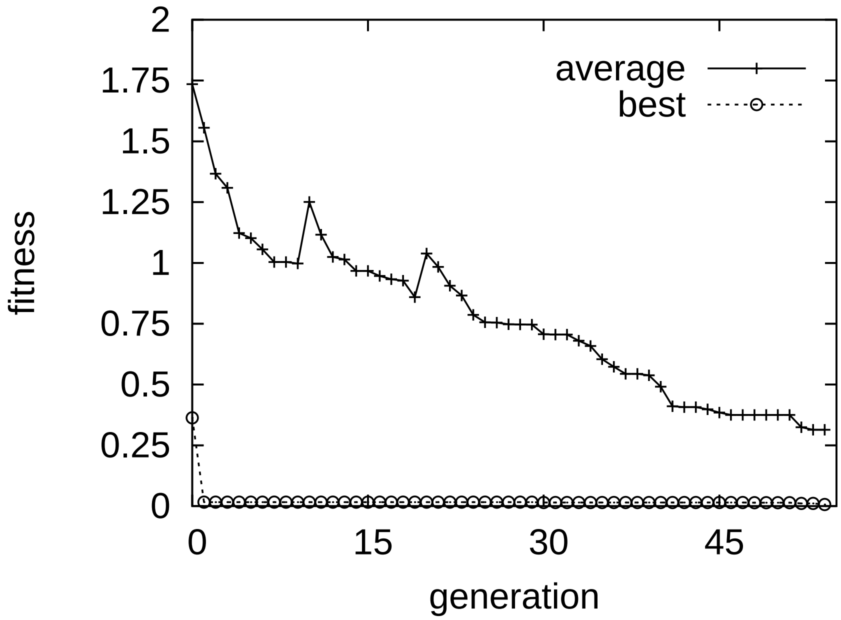
2.4 ALPS

The EA used for the experiments is the Age-Layered Population Structure (ALPS) [3]. Unlike a traditional EA, ALPS maintains several layers of individuals of different age levels and continuously introduces new, randomly generated individuals into the first layer. It has been shown to work better than the canonical EA by better avoiding premature convergence [3, 11, 14]. Using it here allowed us to use a small population size, necessary to minimize the number of expensive evaluations, while avoiding problems of the population prematurely converging on a poor solution.

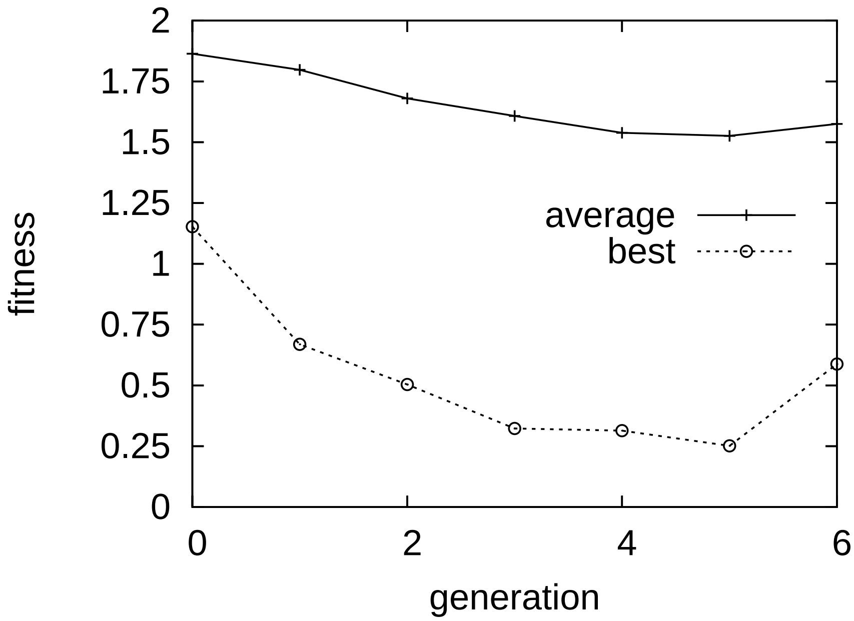
The ALPS EA which we used consisted of a steady-state paradigm with 5 age layers. The evolutionary runs using simple evaluation, prestress, and both prestress and location noise used a population size of 25 individuals. We intended for the evolutionary run with location noise to also use 25 individuals but, because of a typo in the configuration file, it was set to 35 individuals.

3 Evolutionary Results

The goal of our experiments was to evolve a design for a MEMS meandering resonator whose proof mass resonated at 10 KHz in the x direction (first mode). Because the overall objective of this project is to evolve designs that successfully transfer from simulation to reality, we are interested in finding a method for evaluating a MEMS device in simulation that accurately predicts its performance in reality. In the previous section we described two modifications to the basic method of evaluating a MEMS resonator in ANSYS: putting noise in the design and using prestress. Since the use of location noise and prestress was independent of each other, candidate individuals could be evaluated in four different ways: no noise and no prestress; noise but no prestress; no noise with prestress; and with both noise and prestress. Due to time constraints, our experiments consisted of one run of each of these four configurations.

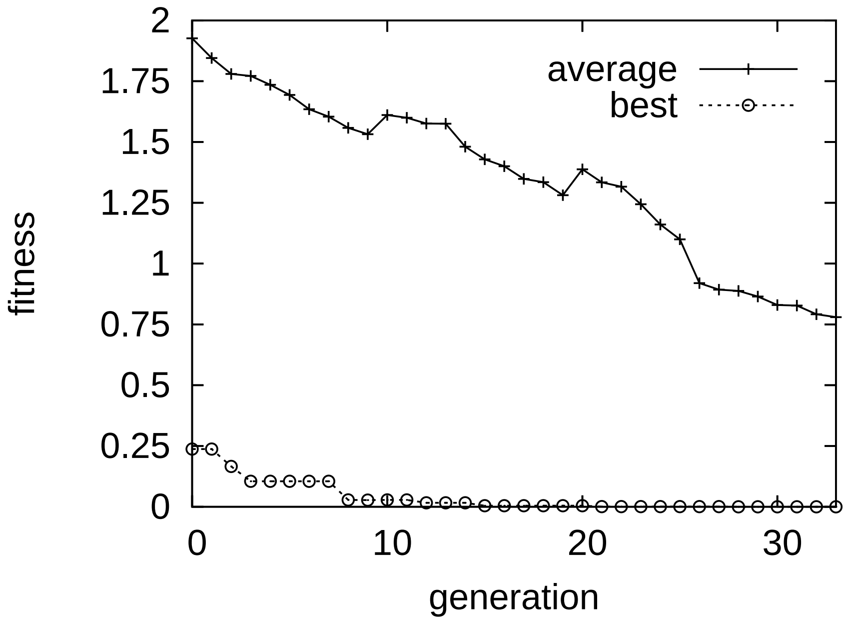


(a)

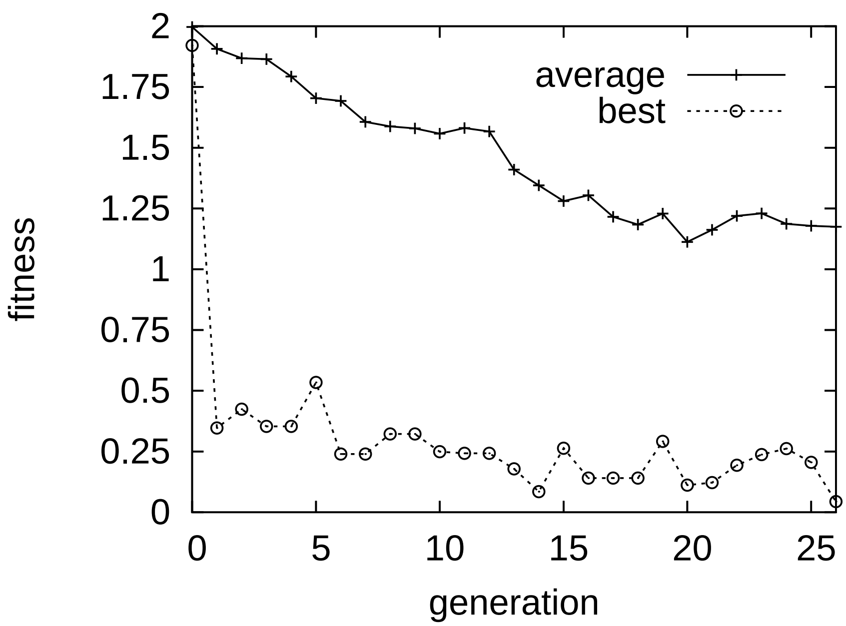


(b)

Fig. 2. Graphs of two of the four different evolutionary runs for evolving a MEMS resonator: (a) simple evaluation model; and (b) evaluation with location noise but no prestress.



(a)



(b)

Fig. 3. Graphs of two of the four different evolutionary runs for evolving a MEMS resonator: (a) evaluation with prestress but no location noise; and and (b) evaluation with location noise and prestress.

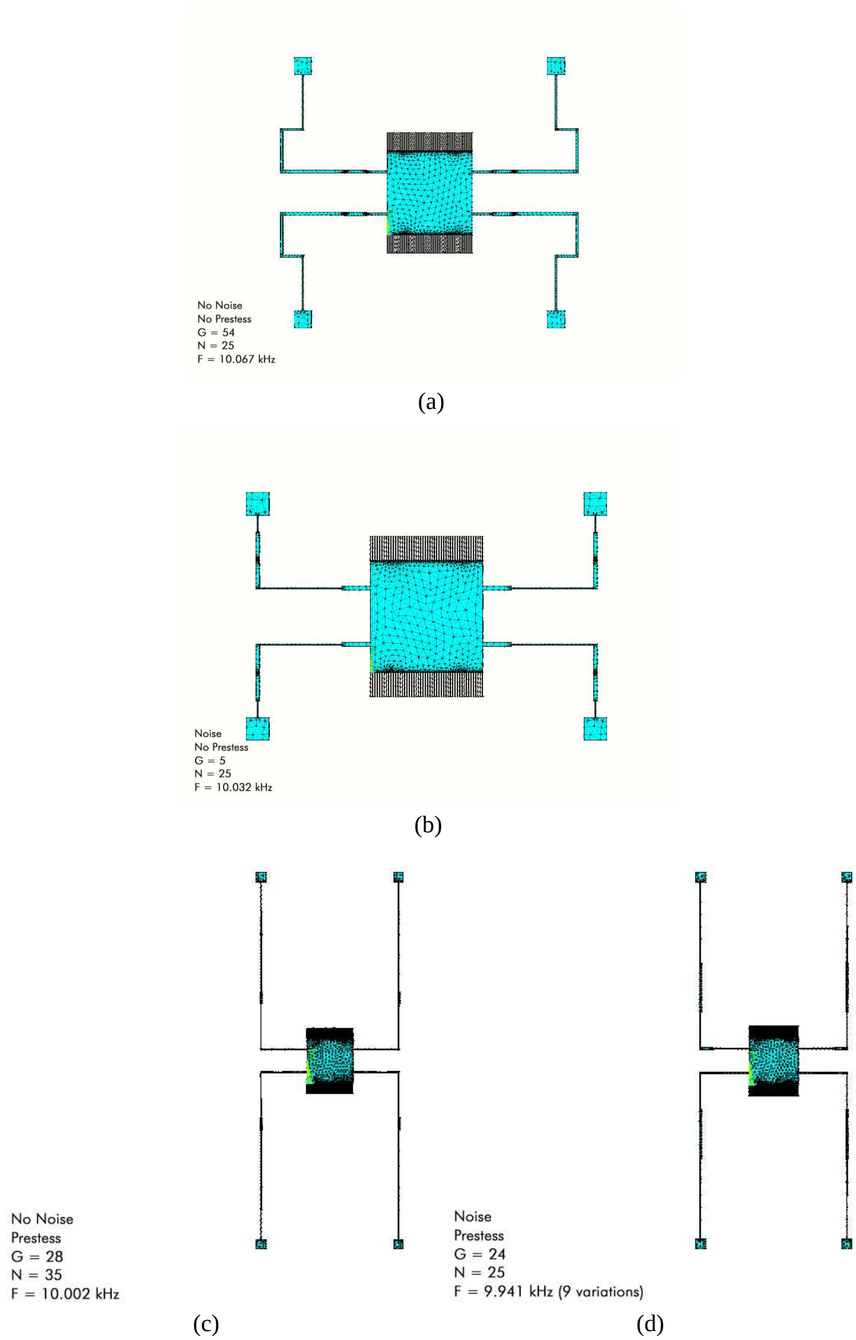


Fig. 4. Evolved resonator designs: (a) no location noise and no prestress; (b) location noise and no prestress; (c) no location noise with prestress; (d) location noise and prestress.

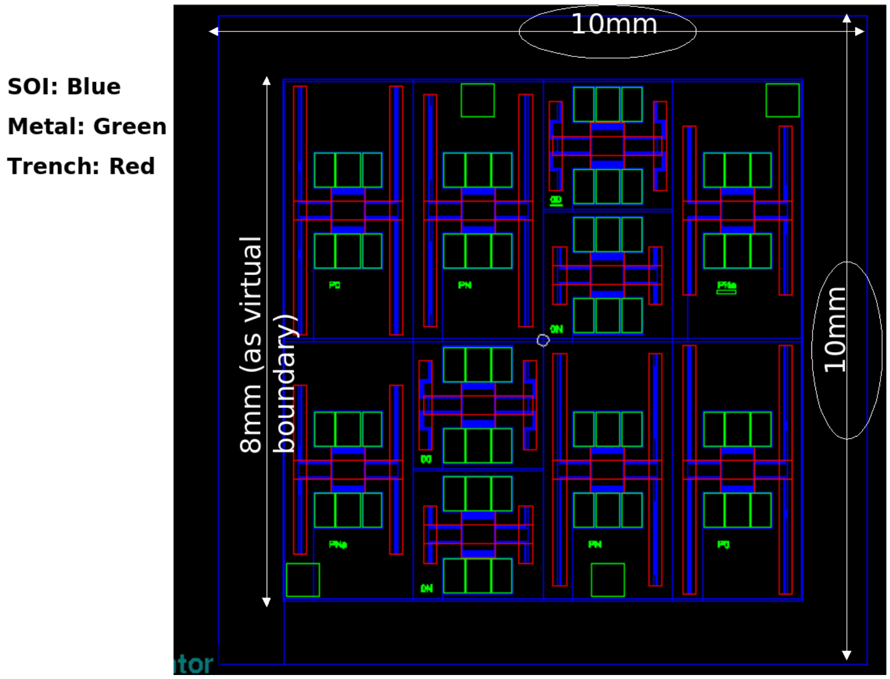


Fig. 5. The complete die layout, suitable for fabricated, with two copies of each evolved design.

The graphs in Figure 2 and 3 show both the best and the average performance of evolutionary runs using each combination of with/without location noise and prestress. The graph in Figure 2(a) plots the performance using the simplest evaluation model, neither location noise or prestress. This graph shows that after only a couple of generations a good resonator design is found. The results of an evolutionary run using the approach of multiple evaluations with location noise are shown in Figure 2(b). Again, good results were obtained after only a few generations, but it can be seen that the best individual from generation 5 receives a much worse cost score as the result of re-testing. Also, this evolutionary run had the fewest number of trials because it was run on our slowest processor. The results of using prestress to evaluate designs is shown in Figure 3(a). In this graph we can see that after about 8 generations a close to optimum design is found. Finally, the results of an evolutionary run using both noise and prestress are shown in the graph in Figure 3(b). The best designs evolved with each of the four methods for evaluating a design are shown in Figure 4.

4 Future Work

The next step in this project is to select a fabrication process – either commercial or academic – and have some of these designs fabricated. To prepare for this next step we selected the best resonator designs from each of the four trials (see Figure 4) and put two copies of each in a sample MEMS layout file. In addition, since there was additional space available, a fifth design was included and this was the second best design from the location noise and prestress run. An image of the overall sample layout is shown in Figure 5, and this shows the 10 resonators laid out on an 8mm by 8mm design space. In future we hope to have this, or something similar, fabricated and then have the resulting chips tested to determine how well they transferred to reality.

5 Summary

MEMS technologies appeal to many areas of engineering because of their low-cost, miniature form-factor, high precision and reliability. We are working toward demonstrating the ability to produce actual MEMS devices in which the design was created by an evolutionary algorithm and a generative representation. Our current stage is that of evolving designs in simulation. To improve the likelihood that evolved MEMS designs successfully transfer to reality, we presented two modifications for evaluating candidate designs. The first method was to evaluate a design multiple times with differences in the design dimensions. This *location noise* in the dimensions is intended to encompass the actual likely differences between what is specified in the blueprint and what is produced in the fabrication process. The second was to apply *prestress* to the design, to incorporate the effects of warping due to the high temperatures involved in manufacturing.

For the four combinations of with/without location noise and prestress, we performed evolutionary runs to evolve a MEMS meandering resonator. We presented results for a single run of each setting which showed that all four approaches are capable of producing a resonator design of the appropriate frequency.

The next step of this project is to demonstrate the ability to successfully go from designs produce from evolution in simulation to correctly working designs in reality. We showed a sample layout file with resonator designs produced by the different evaluation methods and now we are ready to select a fabrication process and have our chips tested. In future, we expect to report on how well our methods work for producing real-world MEMS chips.

Acknowledgements

This work was funded by NASA's Radiation Hardened Electronics for Space Environments Project in the Exploration Systems Mission Directorate. We gratefully acknowledge helpful conversations with Stephen Bart and Ningning Zhou, as well as assistance in testing our designs from Darrell Niemann and Cattien Nguyen.

References

1. P. J. Bentley, editor. *Evolutionary Design by Computers.* Morgan Kaufmann, San Francisco, 1999.
2. P. J. Bentley and D. W. Corne, editors. *Creative Evolutionary Systems.* Morgan Kaufmann, San Francisco, 2001.

3. G. S. Hornby. ALPS: The age-layered population structure for reducing the problem of premature convergence. In M. K. et al., editor, *Proc. of the Genetic and Evolutionary Computation Conference, GECCO-2006*, pages 815–822, Seattle, WA, 2006. ACM Press.
4. G. S. Hornby, H. Lipson, and J. B. Pollack. Generative representations for the automatic design of modular physical robots. *IEEE Transactions on Robotics and Automation*, 19(4):703–719, 2003.
5. R. Kamalian, N. Zhou, and M. Agogino. A comparison of mems synthesis techniques. In *Proceedings of the 1st Pacific Rim Workshop on Transducers and Micro/Nano Technologies*, pages 239–242, 2002.
6. H. Li and E. K. Antonsson. Genetic algorithms in MEMS synthesis. In *Proceedings of IMECE'98 1998 ASME International Mechanical Engineering Congress and Expositions*, 1998.
7. H. Li and E. K. Antonsson. Mask-layout synthesis through an evolutionary algorith. In *Modeling and Simulation of Microsystems, Semiconductors, Sensors and Actuators*. IEEE, 1999.
8. J. D. Lohn, G. S. Hornby, and D. S. Linden. Rapid re-evolution of an X-band antenna for NASA's space technology 5 mission. In T. Yu, R. L. Riolo, and B. Worzel, editors, *Genetic Programming Theory and Practice III*, volume 9 of *Genetic Programming*, chapter 5, pages 65–78. Springer, Ann Arbor, 2005.
9. J. D. Lohn, W. F. Kraus, and G. S. Hornby. Automated design of a MEMS resonator. In D. Srinivasan and L. Wang, editors, *IEEE Congress on Evolutionary Computation*, pages 3486–3491, Singapore, 2007.
10. L. Ma and E. K. Antonsson. Automated mask-layout and process synthesis for MEMS. In *Proceedings of the International Conference on Modeling and Simulation of Microsystems*, pages 20–23, 2000.
11. T. McConaghy, P. Palmers, G. G. E. Gielen, and M. Steyaert. Genetic programming with reuse of known designs for industrially scalable, novel circuit design. In R. L. Riolo and B. Worzel, editors, *Genetic Programming Theory and Practice V*, chapter 10. Kluwer, Ann Arbor, 2007.
12. Z. Michalewicz, D. Dasgupta, R. G. L. Riche, and M. Schoenauer. Evolutionary algorithms for constrained engineering problems. *Computers and Industrial Engineering Journal*, 30(2):851–870, 1996.
13. R. Osiander, M. A. G. Darrin, and J. L. Champion, editors. *MEMS and Microstructures in Aerospace Applications*. Taylor and Francis Group, Boca Raton, FL, 2006.
14. S. Patel and C. Clack. ALPS evaluation in financial portfolio optimisation. In D. Srinivasan and L. Wang, editors, *IEEE Congress on Evolutionary Computation*, pages 813–819, Singapore, 2007.
15. G. Robinson, M. El-Beltagy, and A. Keane. Optimization in mechanical design. In P. J. Bentley, editor, *Evolutionary Design by Computers*, chapter 6, pages 147–165. Morgan Kaufmann, San Francisco, 1999.
16. N. Zhou. *Simulation and Synthesis of MicroElectroMechanical Systems*. PhD thesis, University of California, Berkeley, 2002.
17. N. Zhou, A. M. Agogino, and K. S. J. Pister. Automated design synthesis for micro-electromechanical systems (MEMS). In *Proceedings of the ASME Design Automation Conference*, 2002.
18. N. Zhou, B. Zhu, A. M. Agogino, and K. S. J. Pister. Evolutionary synthesis of microelectromechanical systems (MEMS) design. In *Proceedings of ANNIE 2001: IEEE Neural Networks Council and Smart Engineering Systems Laboratory*, volume 11, pages 197–202. ASME Press, 2001.