Title
Case-Based Reasoning and Object-Oriented Data Structures Exploit Biological Analogs to Generate Virtual Evolutionary Linkages

Authors
Cobb, Corie L.
Zhang, Ying
Agogino, Alice M.
et al.

Publication Date
2007

Case-Based Reasoning and Object-Oriented Data Structures Exploit Biological Analogs to Generate Virtual Evolutionary Linkages

Corie L. Cobb, Ying Zhang, Alice M. Agogino, and Jennifer Mangold

***Abstract*—Multiobjective Genetic Algorithms (MOGA) and Case-based Reasoning (CBR) have proven successful in the design of MEMS (Microelectromechanical Systems) suspension systems. Object-oriented data structures of primitive and complex genetic algorithm (GA) elements have been developed to restrict genetic operations to produce feasible design combinations as required by physical limitations or practical constraints. Thus, virtual linkage between genes and chromosomes are coded into the properties of pre-defined GA objects. A new design problem requires selecting the right primitive elements, associated data structures, and linkages that promise to produce the best gene pool for new functional requirements. In this paper, biomimetics is proposed as a means to examine and classify functional requirements so that case-based reasoning algorithms can be used to map design requirements to promising initial conceptual designs and appropriate GA primitives. The concept is demonstrated using micro-mechanical resonators.**

I. Introduction

Microelectromechanical Systems (MEMS) are small micro-machines or micro-scale electro-mechanical devices that are fabricated with processes adapted from Integrated Circuits (ICs). Although still a relatively new research field, MEMS devices are being developed and deployed in a broad range of application areas, including consumer electronics, biotechnology, automotive systems and aerospace. As these devices grow in complexity, there is a greater need to reduce the amount of time MEMS designers spend in the initial conceptual stages of design by employing efficient computer-aided design (CAD) tools.

Working with a multidisciplinary research team at the Berkeley Sensor and Actuator Center (BSAC), our work with Evolutionary Computation (EC) is focused on the conceptual design of MEMS devices. Zhou et al. [1] were the first to demonstrate that a multi-objective genetic algorithm (MOGA) can synthesize MEMS resonators and produce new design structures. SUGAR [2], a MEMS simulation tool, was used to perform function evaluations on constraints and fitness values. Kamalian et al. [3] extended Zhou's work and explored interactive evolutionary computation to integrate human design expertise into the synthesis process. They also fabricated and tested the emergent designs in order to characterize their mechanical properties and identify deviations between simulated and fabricated features [4]. Zhang et al. [5],[6] implemented a hierarchical MEMS synthesis and optimization architecture, using a component-based genotype representation and two levels of optimization: global genetic algorithms (GA) and local gradient-based refinement. Cobb et al. [7] created a case-based reasoning (CBR) tool to serve as an automated knowledge base for the synthesis of MEMS resonant structures, integrating CBR with MOGA [8] to select promising initial designs for MOGA and to increase the number of optimal design concepts presented to MEMS designers.

In other related research areas, Muhkerjee et al. [9] have conducted work on MEMS synthesis for accelerometers using parametric optimization of a pre-defined MEMS topology. They expanded the design exploration within a multidimensional grid in order to find the global optimal solution. Wang's [10] approach to MEMS synthesis utilized bond graphs and genetic programming with a tree-like structure of building blocks to incorporate knowledge into the evolutionary process, similar to work by Zhang [6]. Li et al. [11] concentrated on developing automatic fabrication process planning for surface micromachined MEMS devices that releases the designers from the tedious work of process planning so that they can concentrate on the design itself. MEMS CAD has matured to the point that there are now commercial CAD programs, such as Comsol® and IntelliSuite®, that offer MEMS designers pre-made modules and cell libraries, but there is little automatic reasoning in place for the user on how and when these components should be used.

Our EC method employs a genetic algorithm as the evolutionary search and optimization method. GAs were introduced by John Holland [12] to explain the adaptive processes of evolving natural systems and for creating new artificial systems in a similar way, and Goldberg [13] further demonstrated how to use them in search, optimization, and machine learning. Chen et al. [14] noted that traditional GAs require users to possess prior domain knowledge in order for genes on chromosomes to be correctly arranged with respect to the chosen operators. The performance of a GA is heavily dependent upon its encoding scheme. When prior domain

Manuscript received March 30, 2007.

This work was supported in part by NSF grant CCR-DES/CC-0306557 and a Bell Labs Graduate Research Fellowship.

Corie L. Cobb is a Ph.D. Candidate in the Mechanical Engineering Department at the University of California, Berkeley, CA 94720, USA (e-mail: ccobb@me.berkeley.edu)

Ying Zhang is on the faculty of the School of Electrical and Computer Engineering at Georgia Institute of Technology, Savannah, GA 31407, USA (email: yzhang@gatech.edu).

Alice M. Agogino is on the faculty of the Mechanical Engineering Department at the University of California, Berkeley, CA 94720, USA (e-mail: agogino@berkeley.edu).

Jennifer Mangold is a graduate student in the Mechanical Engineering Department at the University of California, Berkeley, CA 94720, USA (e-mail: jam@me.berkeley.edu)

1-4244-1340-0/07/$25.00 ©2007 IEEE

knowledge is available, the design problem can be solved using traditional genetic algorithms. However, that is not always the case, and this is when methods such as linkage learning are needed. Chen [15] and Harik [16] both focused research efforts on the linkage learning genetic algorithm (LLGA) so that a GA, on its own, can detect associations among genes to form building blocks [15].

Linkage is an important part of GA performance. Tightly linked genes are synonymous with building blocks, but higher level linkage amongst building blocks is also necessary to ensure successful design solutions are reached. This paper proposes an integrated MEMS design synthesis system which combines CBR with biologically inspired classifications and an evolutionary algorithm, MOGA, to help generate more varied conceptual MEMS design cases for a designer and her/his current design application. The concept is explained using mechanical resonators as an example.

II. MEMS SENSOR DESIGN WITH EVOLUTIONARY COMPUTATION

To date, our resonator synthesis examples (Fig. 1) have consisted of a fixed center mass (either with or without electrostatic comb drives) connected to four 'legs', each made up of multiple beam segments called "meandering springs." We have run our MOGA synthesis program for several sets of performance objectives all calculated using the SUGAR simulation program.

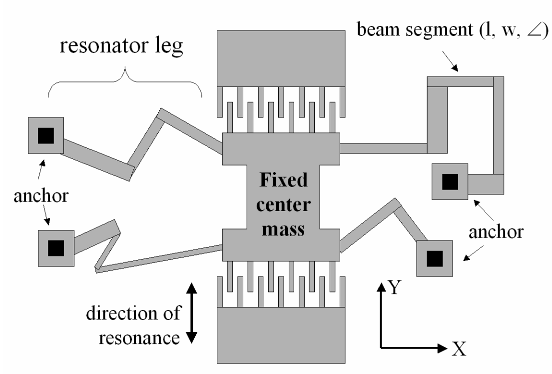


Fig. 1. Schematic of resonator synthesis example problem. The geometry of the center mass is fixed, while the number of beam segments per leg and the size and angle of each segment is variable [3].

SUGAR [2] is an open-source MEMS simulation tool based on modified nodal analysis (MNA), allowing a designer to quickly prototype and simulate several complex MEMS structures for preliminary design applications. Finite element analysis (FEA) calculations could take hours per simulation, making them infeasible for iterative design processes on complex systems. SUGAR and other similar lumped parameter nodal analysis simulation tools can perform these functional calculations with reasonable accuracy at a fraction of the time and can therefore allow the MEMS designer to explore larger design spaces. FEA and parametric optimization can then be used to refine the most promising of the design concepts produced by the MOGA evolutionary process.

As we are designing resonators, the most significant performance objective for all structures is the resonant frequency which is a function of the device's mass and the stiffness of the suspensions. Other performance objectives we have used for synthesis include the stiffness of the structure in the x or y-direction as well as the device area (defined by a bounding rectangle around the device). A schematic of a MEMS resonator and its component decomposition are shown in Fig. 1.

A. Linkage with Component-based Genotype Representation

Genetic linkage, in biological terms, refers to the relative position of two genes on chromosomes. Two genes are linked if they are on the same chromosome and are tightly linked if they are physically close to each other on the same chromosome. Genes that are closely linked are usually inherited together from parent to offspring [14]. Our MOGA data structure can be classified as "linkage adaptation" if we use the same terminology as Chen [14]. Linkage adaptation refers to specifically designed representations, operators, and mechanisms for adapting genetic linkage along with the evolutionary process. Chen states that linkage adaptation techniques are closer to biological metaphors of evolutionary computation because of their representations, operators, and mechanisms.

Our component-based genotype representation for MEMS design synthesis is supported by a hierarchical extendible design component library [6]. Each MEMS design component type is represented by a gene. This gene carries all salient information about this component: its geometric layout parameters, as well as constraints on how it can be modified and what genetic operations can be applied to it. Each gene has external nodes through which components are connected and registered to one another. Two genes are on the same chromosome, which represents a design cluster or a simple MEMS design, if one of them can be reached from the other through any linkage path in the chromosome. Two genes are tightly linked if they share the same external node. A designer can predefine what gene types are allowed to be closely linked to a specific gene type and whether a position on the chromosome is a crossover point during the evolutionary process by associating special properties to certain linkage nodes in the chromosome. Based on predefined rules, the mutation operation can be applied at either the gene level or the chromosome level that provides a probability of changing linkage with mutation operation during the evolutionary process.

III. BIOMIMETICS: THE ROLE OF SYMMETRY AND RESONANCE IN MEMS STRUCTURES

Applying Manhattan geometries (90 degree angles) and symmetry constraints greatly reduces the search space and allows MOGA to optimize its search over a more manageable size. However, when MOGA runs unconstrained or with only symmetry constraints, the results produce designs that greatly differ from those designed by humans (see Fig. 2). Upon observation, these designs have an uncanny appearance to spiders, insects, and other organisms observed in nature. This prompted us to examine the biological analogies between our EC generated resonators and biological organisms to help us understand which symmetry and geometric constraints might be an evolutionary advantage of natural life forms that use vibration or natural frequencies to survive.

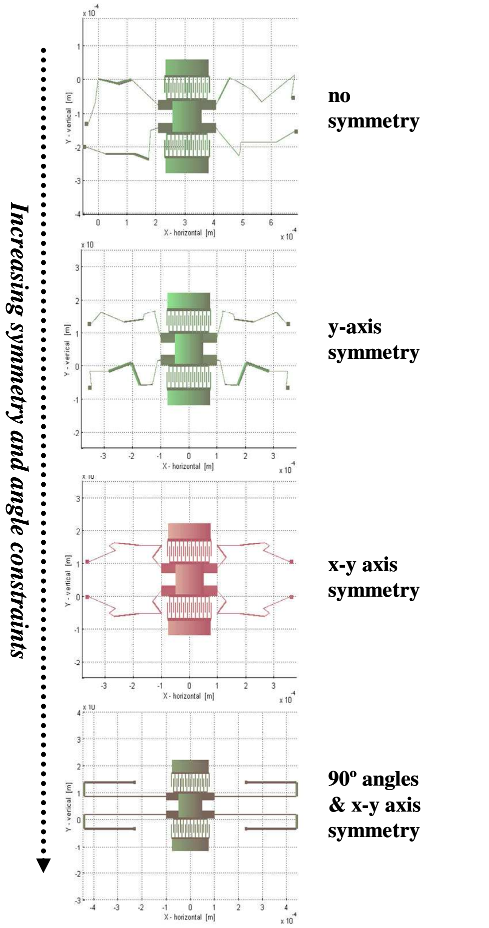


Fig. 2. Examples of resonator designs with increasing constraints

A. *Symmetry and Geometric Constraints*

Symmetry is evident throughout the natural world – a butterfly's wings, a spider's web, and even physicists observe symmetry in distant galaxies. Symmetry has been used to try to understand the physical world since ancient times [17]. In the animal kingdom, bilateral symmetry is found in more complex species, where different parts of the animal's body perform different functions. Radial symmetry can be found in simpler life forms, such as starfish, where the entire body performs most of the life functions.

Symmetry has typically been a sign of quality in nature, and symmetry perception has been demonstrated in humans, animals, and insects. Many studies have concluded that humans and other species find symmetrical patterns more favorable than asymmetrical ones. It has been suggested that preferences for symmetry adapted for reasons related to mate choice. For several species, females prefer a mate that has more symmetrical characteristics [18]; experiments performed with insects and birds found that females prefer to mate with males who have the most symmetrical ornaments [19]. Enquist and Arak [20] suggest that the preference for symmetry has evolved from the need to recognize objects no matter what their position or orientation may be. This preference for symmetry is prominent in the MEMS world where many designers highly favor symmetrical layouts and Manhattan style geometry. In previous work, some of our nontraditional MEMS designs were fabricated and characterized to help improve EC algorithms, and it was shown that the fabricated design behaved within reasonable agreement to simulation results [4].

MEMS designers are tasked with developing physical forms that satisfy multiple functional requirements. It is tempting to think that simple designs with 90 degree angles are better than designs with irregular or nontraditional layouts. This can be the case in macroscale designs where non-perpendicular and parallel designs can be time consuming and expensive from a manufacturing point of view. But in MEMS fabrication, lithography processes enable a designer to create almost any geometrical layout and all are equally easy to fabricate, with the only obstacle being the resolution capabilities of the lithography process, impacting the minimum size of features that can be fabricated.

Kamalian et al. [3] previously noted that optimal MEMS designs with multiple competing objectives need not have full symmetry or Manhattan angles. Similar to our EC generated MEMS resonators, spiders have a large central mass and a similar number of legs on either side of their body. Spiders have evolved to have some degree of symmetry around the longitudinal axis, but none around the horizontal axis, similar to our y-symmetric resonator designs shown in Fig. 2. All species of spiders have a broad range of leg shapes, but none of them have Manhattan geometries and most exhibit symmetry about only one axis.

B. *Purpose of Resonance and Vibration*

We can further examine the spider as a biological analog to a resonator in its ability to detect prey by resonating with their vibrations. Vibration cues have been used by insects and spiders to locate and kill their prey. Without the use of vibration recognition, it may be difficult for insects to find their prey, because dense vegetation may limit their visual abilities. Vibration signals are also important, because many of the insect's or spider's prey produce vibrations through movement or feeding, which enables them to be located more easily [21].

Bola spiders catch their prey by mimicry, emitting the pheromones of the prey species. The wing-beat vibrations of the moths that fall victim to the bola spiders stimulate the spider to make a bolas in which to capture the moth [22]. Generally all web-spinning spiders detect and find prey in their webs through the vibrations generated by their prey. This is especially important because most species of web spiders do not have a strong sense of smell or good vision. Peters (1931) found that the spiders did not respond to a dead fly placed gently in its web. If, however, the fly arrived in the web with a jerk or if, once in the web, it was stimulated in some way, the spider responded [23]. There is a good deal of evidence that spiders discriminate between different types of signals. There have been several studies that demonstrated how spiders move towards vibrations of various frequencies, similar to the way MEMS resonators and bandpass filters attempt to hone in on certain frequencies for communication purposes. Resonators, which are basic building blocks of MEMS filters, are designed to reject certain frequencies from a wide range of signals and only allow a particular frequency band to pass through.

The type of stabilimenta, a structural characteristic of the web, a spider chooses to spin is also influenced by vibration characteristics. A spider that is in need of food will spin a spiral pattern, as oppose to a linear one, because the higher tension transmits high frequencies better than the linear pattern. The high frequencies are generally produced by smaller prey. This would suggest that the spider structures its web according to its food needs [24], similar to the way MEMS designers layout a resonant device in order to achieve high frequencies in the GHz range for communication devices.

An import aspect of resonance in MEMS and nature is movement. Blickhan and Full [25] conducted a study of multi-legged locomotion in animals as diverse as cockroaches and kangaroos in order to develop a model of "legged terrestrial locomotion." They found that the dynamics of movement depend on the number of legs one has and the gait or movement pattern. Four- and six-legged creatures had greater whole body stiffness than two-legged creatures. The greater whole body stiffness in the four- and six-legged creatures resulted in higher natural frequencies, just as a higher overall stiffness results in a higher natural frequency in MEMS. Spiders generally have eight legs while insects have six legs. In MEMS, we mostly observe resonators with four main legs for stability. There are resonators with only two legs, but these tend to be slightly unstable with a tendency towards out of plane movement. In spiders, eight legs can enable them to move faster and give them the ability to travel in different directions easily. Some insects with six legs have a tendency to move forward more and not backwards and sideways as quickly as spiders. In our MEMS resonator design, we only want to move in one direction based on the comb drive actuation, hence four legs provides more balance and stability than two legs. Additional legs are not needed because in these MEMS resonator designs, motion in multiple directions is undesirable. However, if we look more broadly at other MEMS designs, such as micro-robots, more legs can be desirable to enable quick and easy movement in multiple directions.

After 3.8 billion years of "research and development," nature has discovered what works, what does not, and what is life sustaining, essentially perfecting its designs to meet the necessary functional needs. These "successful designs" are ever-changing to meet environmental requirements and are driven by an ultimate challenge: survival. Nature's solutions are not perfect; they are solutions that are as good as they need to be to serve their purpose.

IV. Case-Based Reasoning for MEMS Using Biomimetic Ontology

Case-based Reasoning (CBR) is an artificial intelligence method that utilizes knowledge from a past situation to solve current problems. Shank's dynamic memory model [26] is regarded as the beginnings of CBR. Kolodner used Shank's model to create the first CBR system called CYRUS which was a basic question and answer system [27]. CBR is analogous to the human cognition and thought process. With this analogy, cases can be regarded as "memories," while retrieval is similar to "reminding" one of a particular instance, and case representation is the way one's memories are organized. CBR involves indexing past knowledge, in the form of "cases" to enable effective retrieval of solutions for a current problem. Indexing and case representation are the two initial and most important stages of CBR, determining the ultimate performance of a CBR program.

In the context of our work, CBR takes advantage of previous human knowledge in the form of successful MEMS design cases to help guide humans and computational design tools towards more optimal design concepts. Previous work [8] has shown that the integration of a CBR knowledge base with a multi-objective genetic algorithm (MOGA) can increase the number of optimal solutions generated for a given MEMS design problem. CBR is used to help select the best candidates to be evolved in an evolutionary process such as MOGA. In the following sections, we will examine the biological analogs of case representation and indexing as well as how they can support linkage in MOGA.

A. *Creating Evolutionary Linkage with CBR*

In linkage, as defined by Chen [14], by placing related genes close together on a chromosome, the GA programmer seeds the GA with initial designs with implicit linkages. The GA programmer may be adding her/his expertise to the codification in this process. This may be difficult to do, however, on new problems in which the programmer has limited experience.

Applying the aforementioned definition to MEMS synthesis, we use the concept of linkage to refer to how closely MEMS building blocks should be linked in an evolutionary process. With the integration of CBR and MOGA, CBR defines the linkages for the user with an automated case-based library of previous MEMS designs. CBR takes away from the user the burden of defining the problem by automatically selecting and optimizing design structures based on a few inputted design specifications.

CBR pulls out the best design cases for a given scenario and ranks them according to the user's performance criteria. The designs are then encoded in the component-based genotype representation to enable the evolutionary process. Incorporating other powerful computational tools, such as CBR, with MOGA can help MOGA converge faster and more efficiently to optimal design concepts. The linkage problem is alleviated in our MOGA program because CBR inherently defines linkage for MOGA with its case examples. Shown in Fig. 3 and Fig. 4 are examples of tight linkage generated by our integrated CBR and MOGA program.

CBR assists MOGA by propagating the linkage of effective building blocks. In one experiment [8], for each MOGA synthesis run, we used a population of 400 for 50 generations. Each constraint case of (1) no symmetry, (2) y-symmetry, and (3) xy-symmetry had 5 runs of the MOGA process in order to see a good spread of designs. We found that when MOGA is seeded with good starting designs from CBR, in some instances, y-symmetry and xy-symmetry constraints generate more pareto optimal designs. A resonator with an enclosed frame mass and crab-leg suspensions (two beams with a local 90 degree angle) was selected as the starting design for a given resonant frequency (f_0), stiffness ratio (K_x/K_y) and area constraint (Table I and Fig. 3). The aforementioned constraints correspond to the following design requirements: f_0 = 24.7 kHz, $K_x/K_y \geq 47.9$, $Area \leq 1.455\text{e-}7$ m^2. For this scenario, the mass and comb drives remained fixed while the crab-leg suspensions (which have the largest impact on the performance objectives) were allowed to change in width, length, and global orientation, but the crab-leg suspensions retained their local 90 degree angle.

As one can see in Fig. 3, the initial design in Fig. 3a generated an optimal design (Fig. 3b) which had the leg suspensions rotated outside of the frame mass. One would assume that if the objective is to minimize area, the suspensions would remain inside the mass, similar to the initial design in Fig. 3a. However, because frequency and stiffness were also part of the optimization problem, MOGA determined that a design with the suspensions outside of the mass could produce a better resonant frequency and stiffness ratio. This design again resembles some of the asymmetrical spider designs we discussed previously. The resonator design in Fig. 3b may have not been considered by a human MEMS designer, but due to the linkage knowledge CBR gave MOGA, the design is a good candidate for further analysis and fabrication.

TABLE I: EXAMPLE MOGA REPRESENTATION FOR INITIAL DESIGN IN FIG. 3

MOGA Design Representation		
Gene Type	**MEMS Component**	1 | 5 | 1 — 9 > 10 < 9 — 1 1 — 9 / | \\ 9 — 1 5 | 1
10	Frame Mass	
9	Crab-leg Suspension	
5	Comb Drive	
1	Anchor	

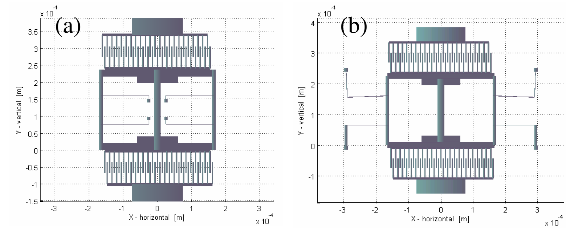


Fig. 3. Resonant frequency = 23.8 kHz for initial MOGA design (a); Resonant frequency = 24.8 kHz for a pareto optimal y-symmetric design generated by MOGA (b)

TABLE II: Example MOGA Representation for Initial Design in Fig. 4a

MOGA Design Representation		
Gene Type	**MEMS Component**	1 | 5 | 1 — 2 > 15 < 2 — 1 1 — 2 / | \\ 2 — 1 5 | 1
15	Hollow Ring Mass	
2	Serpentine Suspension	
5	Comb Drive	
1	Anchor	

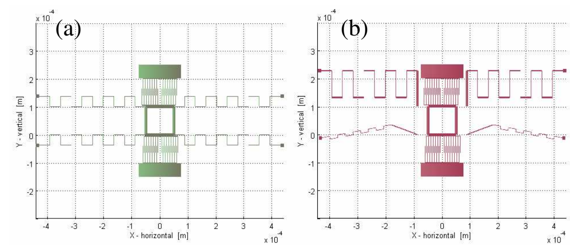


Fig. 4. Resonant frequency = 6969.3 Hz for initial MOGA design (a); Resonant frequency = 8299.9 Hz for a pareto optimal y-symmetric design generated by MOGA (b)

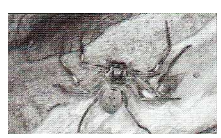
Fig. 4 shows another example of tight linkage in our MOGA process. The initial design was optimized with MOGA to meet the following design requirements: f_0 = 8.3kHz, $K_x/K_y \geq 29$, $Area \leq 3.7\text{e-}7$ m^2. In this particular case, the initial design selected by CBR consisted of a hollow squared shaped mass with four serpentine springs. Again, the mass and comb drives remained fixed while the serpentine suspension blocks were free to mutate in length, width, number of loops, and their global angle orientation. This scenario also generated designs which had a similar appearance to spiders and insects (aside from the inherent manhattan geometry in the building blocks). Minimizing area is our main design objective for all cases, and the y-symmetry constraint runs had the smallest design area average (2.608e-7 m^2) with a standard deviation of 7.031e-8 m^2.

B. Case Representation and Biological Taxonomy

Biological classification or taxonomy is a means by which biologists group and classify organisms. Taxonomy helps one identify evolutionary relationships and links between certain species, and in the case of MEMS, certain design structures. Taxonomy began with classifying organisms based on shared physical traits, but these classifications have been modified over the years to reflect Darwinian evolutionary relationships. As an example, biologists have classified over 40,000 species of spiders, but they believe there are still thousands of species which have not yet been identified and named. As more species are discovered the current biological classification system can expand and change.

The classification of animals and plants is inherently hierarchical; similar to the way our MEMS case library is hierarchical to demonstrate the relationships between designs. The 40,000 species of classified spiders are further divided into 3 suborders with 38 families and 111 subfamilies. The groups described by taxonomy get more specific as one goes from the kingdom classification all the way down to the species group. Kingdom is the largest unit of classification (with approximately 5 kingdoms), phylum is the next unit of classification which further divides each kingdom, and this pattern continues down to the species level, forming a tree like hierarchy of organism representation. As an example, shown in Table III is the classification of the *Isopeda insignis* spider. No two species of spiders, or any plant or animal, will have the same scientific name (defined by the genus and species). The scientific name is a unique identifier just as each unique MEMS design component has a distinct identification number and gene type to distinguish it from other designs and enable efficient case retrieval.

TABLE III: CLASSIFICATION OF AN ISOPEDA INSIGNIS SPIDER [28]

Kingdom	*Animalia*	
Phylum	*Arthropoda*	
Class	*Arachnida*	
Order	*Araneae*	
Family	*Sparassidae*	
Genus	*Isopeda*	
Species	*insignis*	

MEMS is still an exploratory field and new designs and pieces of the MEMS hierarchy are constantly being added, similar to the way newly discovered species of organisms are expanding the biological taxonomy system everyday. Others [29] have noted that it is still premature today to create a robust categorization due to the fact that many MEMS devices are still in the research phases and have not matured for every application. MEMS categorization has often focused on fabrication methods and materials selection, geometry, or application areas [30]. There are a broad array of MEMS sensors and actuators available today. Bell et al. [31] categorized MEMS by considering work-producing actuators, force sensors and displacement sensors fabricated by surface or bulk micromachining in their work and did an in depth classification of these devices.

In MEMS, designs are often classified based on their performance and functional characteristics. Sensors and actuators are the two most broad and commonly agreed upon categories of MEMS which can be divided further into families and classes. Similar to the work of Bell et al. [31], for our purposes, we will have two kingdoms in our classification system: sensors and actuators. Sensors and actuators can each be further divided into phylum or classes based upon their operating domains. For our purposes, we will assume six operating domains based upon input and output signals MEMS devices utilize: (1) Magnetic, (2) Thermal, (3) Electrical, (4) Mechanical, (5) Chemical, and (6) Optical.

Imagine the aforementioned domains placed in a 6 by 6 matrix (with all six categories lined up on the rows and columns) to enable multiple input and out combinations. For example, a thermal-mechanical sensor might take a thermal input and have a mechanical deflection as its output. For a piezoelectric sensor, it will output a voltage in response to an applied mechanical stress, enabling a further categorization of the mechanical-electrical class. Because the user of our CBR program may be searching for designs based on input and output domains or application areas, it is important to index cases by both. Our MEMS hierarchy starts with sensors and actuators, and then branches out to the various input and output mechanisms, and under each of these are specific application areas (RF MEMS, Micro-fluidics, BioMEMS, Optical MEMS, etc.), and then divided further are whole MEMS devices, which are broken down into their various components and primitive elements.

Our work focuses currently on resonant structures, such as resonators, accelerometers and filters, thus traversing the MEMS hierarchy, our work falls under the electrical (input and output domain) where electrostatics are primarily used. Fig. 5 is a condensed graph, and is not inclusive of all MEMS devices. The portion shown demonstrates how the classification leads to accelerometers, filters, and resonators – the focus of our work. Resonators, basic components of filters, can be further decomposed into masses, springs, comb drives, and anchors. Each one of the aforementioned components would have a unique identifier to distinguish them from others. Nguyen [32] classifies MEMS filters based on their ability to achieve a certain frequency range, and import part of being able to develop RF communication devices.

A hierarchy for biological organisms was created just as a hierarchy for CBR needs to be created in order to sort information and efficiently pull the most relevant primitives and designs for evolutionary computation. Our current CBR hierarchy classifies designs based on their shared functionality and performance.

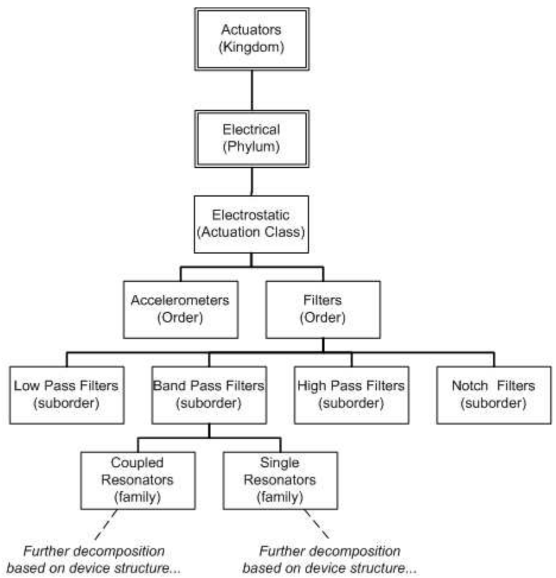


Fig. 5. MEMS hierarchy with biological analogy

Ontology is a way to represent knowledge in a specific domain, helping an artificial intelligence (AI) program to define and retrieve objects. A general hierarchy or structure of ontology is the following [33]: objects, classes of objects, attributes of objects, and relations between objects. Shown in Fig. 6 is our current MEMS case library ontology. Using entity-relationship diagram notation, one can observe how objects such as MEMS resonators and filters are related together. In the diagram, 'd,p' indicates a disjoint/distinct and partial relationship between classes, in order to account for designs that have not yet been created or added to the library. Attributes of each object include indices for quick retrieval and overall device performance.

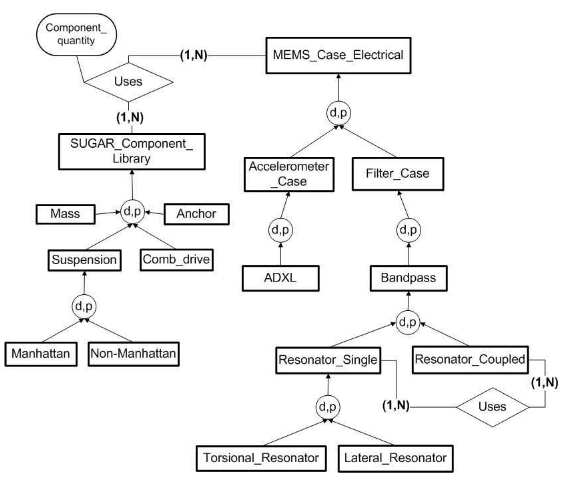


Fig. 6. CBR entity-relationship database schematic for case library with filters added

The database diagram in Fig. 6 represents how information is stored in our case-based library. This new case library will be integrated into our current MEMS design synthesis program, shown in Fig. 7, to further generate and optimize new MEMS design structures.

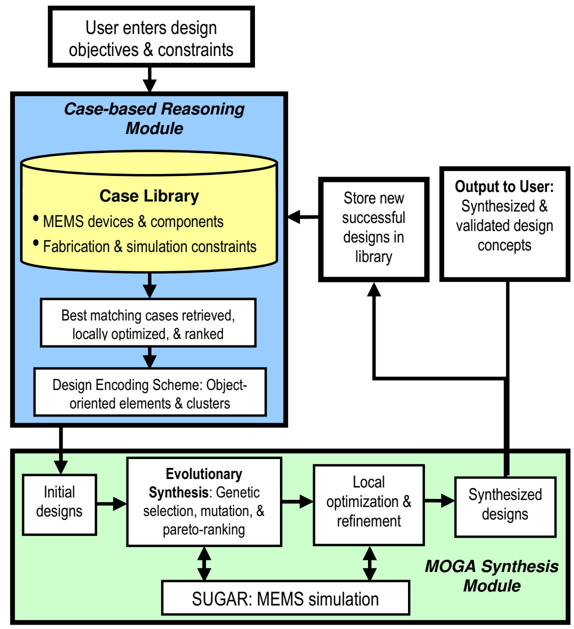


Fig. 7. MEMS design synthesis flowchart

V. Conclusions and Future Work

Our MEMS synthesis architecture with integration of MOGA and CBR deals with the concept of linkage by using a component-based genotype representation and a CBR automated knowledge base. CBR provides MOGA with good linkage information through past design cases while MOGA inherits linkage information through our component based genotype representation.

As part of our future research plan, we will examine how linkage learning can be integrated with MOGA when CBR may not be able to select a good initial design. Further exploring biomimetics and the ties to MEMS synthesis algorithms is another area to pursue, investigating how increasing the number of leg components can create optimal designs in other MEMS areas such as micro-robots. Currently, we are moving towards creating a broader MEMS classification scheme and building up a case library of MEMS filter designs and their accompanying components to further expand the range of designs covered by our program.

References

[1] N. Zhou, B. Zhu, A. M. Agogino, K. S. J. Pister, "Evolutionary synthesis of MEMS (MicroElectronicMechanical Systems) design," In *Proc. of the Artificial Neural Networks in Engineering Conference*, 2001, pp.197-202.

[2] http://www-bsac.eecs.berkeley.edu/cadtools/sugar/sugar/

[3] R. H. Kamalian, A. M. Agogino, and H. Takagi, "The role of constraints and human interaction in evolving MEMS designs: microresonator case study," In *Proc. of 2004 ASME Design Engineering Technical Conferences and Design Automation Conference*, 2004. #DETC2004-57462.

[4] R. Kamalian, Y. Zhang and A. M. Agogino, "Microfabrication and characterization of evolutionary MEMS resonators," In *Proc. of the Symposium of Micro- and Nano-Mechatronics for Information-based Society*, IEEE Robotics & Automation Society (ISBN # 0-7803-9482-8), Nov. 2005, pp. 109-114.

[5] Y. Zhang, R. Kamalian, A.M. Agogino, C. H. Séquin, "Hierarchical MEMS synthesis and optimization," In *Proc. Smart Structures and Materials 2005 Smart Electronics, MEMS, BioMEMS, and Nanotechnology*, SPIE Vol. 5763, pp. 96-106. CD ROM, Paper # 5763_12.

[6] Y. Zhang, R. Kamalian, A. M. Agogino, and C.H. Séquin, "Design synthesis of Microelectromechanical Systems using genetic algorithms with component-based genotype representation," In *Proc. of 2006 Genetic and Evolutionary Computation Conference*. Seattle, July 8-12, 2006. ISBN 1-59593-187-2. Vol. 1, pp.731-738.

[7] C. L. Cobb and A. M. Agogino, "Case-based reasoning for the design of Micro-Electro-Mechanical Systems," In *Proc. of 2006 ASME International Design Engineering Technical Conferences & the Computers and Information in Engineering Conference*, September 10-13, 2006, #DETC2006-99120.

[8] C. L. Cobb, Y. Zhang, and A. M. Agogino, "MEMS design synthesis: integrating case-based reasoning and multi-objective genetic algorithms," In the *Proc. of 2006 SPIE Smart Materials, Nano- and Micro-Smart Systems*, SPIE Vol. 6414, #641419, ISBN: 9780819465221 (Invited Paper).

[9] T. Mukherjee, Y. Zhou and G. Fedder, "Automated optimal synthesis of microaccelerometers," *Technical Digest of the 12th IEEE International Conference on Micro Electro Mechanical Systems (MEMS '99)*, pp. 326-331, 1999.

[10] J. Wang, Z. Fan, J. P. Terpenny, and E. D. Goodman, "Knowledge interaction with genetic programming in mechatronics systems design using bond graphs," *IEEE Transactions on System, Man, and Cybernetics−Part C: Applications and Reviews*, Vol. 35, No. 2, May 2005.

[11] J. Li, S. Gao and Y. Liu, "Solid-based CAPP for surface micromachined MEMS devices," *Computer-Aided Design*, Vol. 39, No. 3, pp. 190-201, 2007.

[12] Holland, J.H., *Adaptation in Natural and Artificial Systems*, Ann Arbor: The University of Michigan Press, 1975.

[13] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Longman, Boston, MA, 1989.

[14] Y.-P. Chen and D. E. Goldberg, "Convergence time for the linkage learning genetic algorithm," *Evolutionary Computation*, Vol. 13, No. 3, Pages 279-302, Fall 2005.

[15] Y.-P. Chen and D.E. Goldberg, "Introducing subchromosome representations to the linkage learning genetic algorithm," *Lecture Notes in Computer Science (LNCS)*, Vol. 3102, pp. 971–982, 2004.

[16] G.R. Harik, *Learning gene linkage to efficiently solve problems of bounded difficulty using genetic algorithms*, Doctoral Dissertation, University of Michigan, Ann Arbor, 1997.

[17] I. Stewart, *Why Beauty Is Truth: A History of SymmetryBook,* Joat Enterprises: Basic Books, 2007.

[18] R. A. Johnstone, "Female preferences for symmetrical males as a by product of selection for mate recognition," *Nature* **372**, pp 172-175 , 1994.

[19] J. P. Swaddle and I. C. Cuthill, "Preference for symmetric males by female zebra finches," *Nature* **367**, 165-166, 1994.

[20] M. Enquist and A. Arak, "Symmetry, beauty, and evolution," *Nature* **372**, pp. 169- 172, 1994.

[21] R. S. Pfannenstiel, R. E. Hunt, and K.V. Yeargan, "Orientation of a hemipteran predator vibrations produced by feeding caterpillars," *Journal of Insect Behavior*, Vol. 8, No. 1, 1995.

[22] K. F. Haynes, K. V. Yeargan, and C. Gemenol, "Detection of prey by a spider that aggressively mimics pheromone blends," *Journal of Insect Behavior, Vol. 14, No. 4, 2001.*

[23] D. A. Parry, "The signal generated by an insect in a spider's web," *Journal of Experimental Biology* **43**, pp. 185-192, 1965.

[24] T. Watanabe, "Web tuning of an orb-web spider, *Octonoba sybotides*, regulates prey-catching behavior," *Proceedings of The Royal Society of London Series B* **267**, p. 565-569, 2000.

[25] R. Blickhan and R. J. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *Journal of Comparative Physiology A*. (1993) **173**, pp. 509-517.

[26] R. Shank., *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*. Cambridge, 1982.

[27] J. Kolodner, *Case-Based Reasoning*, Morgan Kaufmann Publishers, Inc., 1993.

[28] T. J. Hawkeswood, *Spiders of Australia: An introduction to their Classification, Biology and Distribution*. ISBN 9546421928, Pensoft Publishers, Sofia-Moscow. 2003.

[29] V.K. Varadan, *RF MEMS and Their Applications*, John Wiley and Sons, 2002.

[30] J.A. Walraven, "Introduction to applications and industries for microelectromechanical systems (MEMS)," In *Proc. of 2003 International Test Conference (ITC)*, Sept. 30-Oct. 2, 2003. pp. 674- 680.

[31] D.J. Bell, T.J. Lu, N.A. Fleck, and S.M. Spearing, "MEMS actuators and sensors: observations on their performance and selection for purpose," *Journal of Micromechanics and Microengineering*, Vol. 15, No. 3, July 2005.

[32] C. T. -C. Nguyen, "RF MEMS in wireless architecture," In *Proc. of the 42nd Design Automation Conference*, Anaheim, CA, June 13-17, 2005, pp. 416-420.

[33] N. F. Noy and D. L. McGuinness, "Ontology development 101: a guide to creating your first ontology," Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, March 2001.