ADVANCES IN INTELLIGENT AND *SOFT COMPUTING* 150

Ljupco Kocarev (Ed.)

ICT Innovations 2011

Springer

Advances in Intelligent and Soft Computing 150

Editor-in-Chief

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

For further volumes:
http://www.springer.com/series/4240

Ljupco Kocarev (Ed.)

ICT Innovations 2011

Springer

Editor
Ljupco Kocarev
Macedonian Academy of Sciences and Arts
Skopje
Macedonia
Bio Circuits Institute
University of California San Diego
San Diego, California
USA

ISSN 1867-5662 e-ISSN 1867-5670
ISBN 978-3-642-28663-6 e-ISBN 978-3-642-28664-3
DOI 10.1007/978-3-642-28664-3
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2012932760

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The ICT Innovations conference is the primary scientific action of the Macedonian Society in Information and Communication Technologies (ICT-ACT). The conference provides a platform for academics, professionals, and practitioners to interact and share their research findings related to basic and applied research in ICT.

The ICT Innovations 2011 conference gathered 146 authors from 12 countries reporting their valuable work and experience in developing solutions and presenting novel applications of technology in ICT. Only 33 papers were selected for this edition by 80 Program Committee Members, chosen for their reputation for decency and good decree in the field.

ICT Innovations 2011 was held in Skopje, at the Macedonian Academy of Science and Art, in the period September 14-16, 2011. The conference focused on variety of ICT fields: Complex networks and Cryptography, Wireless Communication, Mobile Application, Machine Learning, High Performance Computing, Software Implementation, and Hardware, Grid and Cloud Computing. The conference was open by Ivo Ivanovski, the Minister of Information Society and Administration of the Republic of Macedonia.

I would like to express sincere gratitude to the authors for submitting their works to this conference, the reviewers providing service for the editor but also for authors, readers, and the ICT community as a whole, and all colleagues from the Department of Computer Science and Engineering, especially Sonja Filiposka and Anastas Misev, for their support in organizing the conference and preparing the book proceedings.

Skopje
September 2011

Ljupco Kocarev

Organization

ICT Innovations 2011 was organized by the Macedonian Society of Information and Communication Technologies (ICT-ACT).

Conference and Program Chair

Ljupco Kocarev	Macedonian Academy of Sciences and Arts Bio Circuits Institute, University of California San Diego Faculty of Computer Science and Engineering, Skopje

Program Committee

Ana Madevska-Bogdanova	UKIM, Macedonia
Anastas Misev	UKIM, Macedonia
Andrea Kulakov	UKIM, Macedonia
Aneta Karaivanova	IICT BAS, Bulgaria
Antun Balaz	Institute of physics Belgrade, Serbia
Attila Szucs	University of California San Diego, San Diego, USA
Betim Cico	University of Tirana, Albania
Branko Marovic	University of Belgrade, Serbia
Dancho Davchev	UIT, Macedonia
Danilo Gligoroski	University of Trondheim, Norway
Dejan Gjorgjevikj	UKIM, Macedonia
Dejan Spasov	UKIM, Macedonia
Dilip Patel	LSBU, UK
Dimitar Trajanov	UKIM, Macedonia
Dragan Bosnacki	TUE, The Netherlands
Emanouil Atanassov	IPP BAS, Bulgaria
Francky Trichet	Nantes University, France
Francois Paris	University of Houston, USA

Gabor Vattay	Etovos Lorand University, Budapest, Hungary
Georgi Stojanov	American University of Paris, France
Gerg Duane	University of Colorado, Boulder, USA
Gianluca Setti	University of Bologna, Italy
Goce Armenski	UKIM, Macedonia
Goran Velinov	UKIM, Macedonia
Harold Sjursen	New York University, USA
Hrachya Astsatryan	IIAP NAS, Armenia
Igor Trajkovski	UKIM, Macedonia
Ivan Chorbev	UKIM, Macedonia
Jasen Markovski	Eindhoven University of Technology, The Netherlands
Jorge Marx-Gomez	Oldenburg University, Germany
Juergen Kurhs	Humboldt University, Berlin, Germany
Julien Bourgeois	University of Franche-Comté, France
Katerina Zdravkova	UKIM, Macedonia
Kosta Mitreski	UKIM, Macedonia
Lev Tsimring	University of California San Diego, San Diego, USA
Liane Haak	University Oldenburg, Germany
Ljiljana Trajkovic	Simon Fraser University, Vancouver, British Columbia, Canada
Ljupcho Antovski	UKIM, Macedonia
Lubomir T.-Chitkuchev	Boston University, USA
Luigi Fortuna	University of Catania, Italy
Margita Kon-Popovska	UKIM, Macedonia
Marija Mihova	UKIM, Macedonia
Mario Biey	Politecnico di Torino, Italy
Mario di Bernardo	University of Napoli, Italy
Marjan Gushev	UKIM, Macedonia
Mile Stojchev	University of Niš, Serbia
Milentijevic Ivan	University of Nis, Serbia
Mirjana Ivanovic	University of Novi Sad, Serbia
Natasha Jonoska	Univ. of South Florida, USA
Nevena Ackovska	UKIM, Macedonia
Nevenka Dimitrova	Philips Research, USA
Nikolai Rulkov	University of California San Diego, San Diego, USA
Noel Keenlyside	University of Bergen, Bergen, Norway
Norbert Grünwald	Hochschule Wismar, Germany
Pece Mitrevski	UKLO, Macedonia
Peter Parychek	Univ. Donau, Austria
Predrag Petkovic	University of Niš, Serbia
Sanja Lazarova-Molnar	UAE University, UAE
Sashko Celakovski	ITgma, Macedonia

Sasho Josimovski	UKIM, Macedonia
Saso Dzeroski	Institut Jozef Stefan, Ljubljana, Slovenia
Shushma Patel	LSBU, UK
Slobodan Bojanic	Universidad Politecnica de Madrid, Spain
Slobodan Kalajdziski	UKIM, Macedonia
Smile Markovski	UKIM, Macedonia
Sonja Filiposka	UKIM, Macedonia
Sonja Gievska-Krilu	GWU, USA
Stephan Olariu	ODU, USA
Suzana Loshkovska	UKIM, Macedonia
Tommaso Addabbo	University of Sienna, Italy
Vanco Litovski	University of Niš, Serbia
Verica Bakeva	UKIM, Macedonia
Vesna Dimitrova	UKIM, Macedonia
Visarath In	SPAWAR Systems Center Pacific, San Diego, USA
Vladimir Radevski	SEEU, Macedonia
Vladimir Trajkovic	UKIM, Macedonia
Wallace Tang	City University Hong Kong, Hong Kong, China
Yiannis Kompatsiaris	Centre for Research and Technology Hellas, Greece
Zamir Dika	SEEU, Macedonia
Zoran Hadzi-Velkov	UKIM, Macedonia

Contents

Invited Keynote Papers

Proceeding Papers

A Jumping Gene Evolutionary Approach for Multiobjective Optimization

Wallace K.S. Tang, Chris S.H. Yeung, and K.F. Man

Abstract. The phenomenon of Jumping genes was initially discovered by Nobel Laureate, Barbara McClintock, in her work on maize chromosome in fifties. The Jumping genes transpose from one position to another in horizontal fashion within the same chromosome or even to other chromosomes. In this paper, it is to present how this genetic transposition, after transforming into a computational method, can enhance the evolutionary multiobjective optimization. The fundamental concept, design of operations, performance justification and applications of the Jumping Gene evolutionary approach will be outlined.

1 Introduction

Genetic Algorithm (GA) was firstly proposed by John Holland [1] in late 1960s. As a major family of evolutionary algorithms inspired by the natural phenomenon of biological evolution, its success relies on the evolution of the chromosomes, following the principle of survival of the fittest.

GA is particularly well-known for its ability in handling multiobjective problems, thanks for the efforts made to effectively transform multiple objectives into a single fitness value [2, 3]. However, despite the numerous successful stories of applying GA, keeping diversity in the population during the GA optimization process is still a major problem.

To resolve this problem, many proposals have been made for various multiple objective genetic algorithms, including the use of high disruptive operations (For example, uniform crossover and multi-point crossover); diversity enhancement (e.g. mating restrictions [4]); and diversity perservation (e.g. crowding operator in NSGA2 [5], tournament selection and fitnesss sharing [6, 7]). However, it is still a

Wallace K.S. Tang · Chris S.H. Yeung · K.F. Man
Department of Electronic Engineering, City University of Hong Kong
e-mail: {eekstang,saiyeung,eekman}@cityu.edu.hk

L. Kocarev (Ed.): ICT Innovations 2011, AISC 150, pp. 1–14.
springerlink.com

very difficult task to obtain a widespread non-dominated solution set and the convergence speed is usually slow.

Recently, inspired by the biological mobile genes (called Jumping genes) [8], a new class of operations, called gene transposition, has aroused many attentions and is also the focus of this paper. It describes the transfer of Jumping gene (JG) existing in chromosome, which results in a migration of genetic information between chromosomes within a generation. This is also referred as horizontal transmission in biology as compared with the vertical transmission of genetic information appears from one generation to its descendants.

This paper is to give an introduction of how the transposition of JG contributes in evolutionary computation. The fundamental concept, design of operations, performance justification and the applications of JG, are to be discussed.

2 Evolutionary Multiobjective Optimization

2.1 Multiobjective Problems

Multiobjective problems (MOPs) are commonly encountered in real life. Unlike the single objective problems, there are two or more objective functions to be optimized in MOP at the same time. With such a nature, it is usually demanded to identify a set of tradeoff solutions, instead of a single global optimal one.

Figure 1 shows an example, in which two functions f_1 and f_2 are to be minimized. There exists a set of optimal solutions, called Pareto optimal set, which are not dominated by any other solutions.

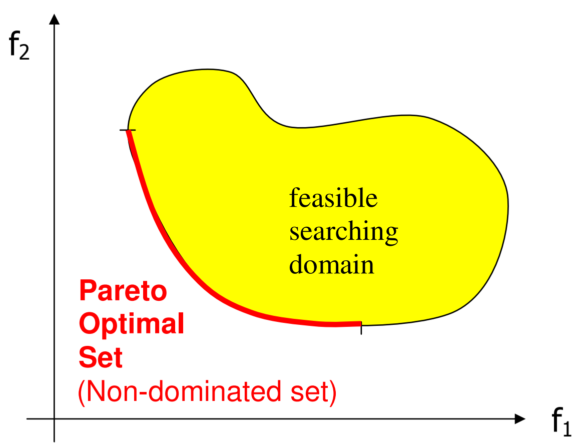

Fig. 1 Pareto optimal set

In mathematics, for a minimization problem (Remark: Similar definition can be given for maximization problem), we say a solution candidate u is dominated by another solution v (or v is more preferable than u) if

$$\begin{aligned} f_i(u) &\geq f_i(v) \quad \forall i = 1, 2, \cdots, n \quad \text{and} \\ f_i(u) &> f_i(v) \quad \exists j = 1, 2, \cdots, n. \end{aligned} \tag{1}$$

Due to optimization and practical reasons, it is more favourable to find the entire Pareto optimal set instead of a single solution. This is usually accomplished by population based methodologies, for which GA is one of them, so that only a single run is sufficient.

2.2 *Overview of GA*

GA is a population-based guided-search methodology, inspired by the rule of "survival of the fittest" and based on the biological DNA structure. As its simplest form, its flow can be described as below and a genetic operational cycle is graphically depicted in Fig. 2:

1. Population creation: Create an initial set of solutions for a problem by encoding each solution into a string of symbols, for example, a binary string, which is called as chromosome.
2. Fitness evaluation: Evaluate each of the chromosomes by the problem specific objective function(s) and a positive fitness value is to be assigned.
3. Selection: Select a subset of chromosomes based on their fitness values. The subset is called a mating pool.
4. Reproduction: New chromosomes (called offspring) are produced by mating chromosomes in the mating pool in pair (called parents). They are created by mixing the genetic information of the parents (the operation is known as crossover) and are then mutated by altering some genes in a random sense (the operation is known as mutation). The set of all offspring created is referred as the subpopulation.
5. Fitness evaluation: Evaluate the chromosomes in the sub-population as in Step 2.
6. Replacement: Some chromosomes in the original population are replaced by the offspring, usually based on their fitness values.
7. Termination: Steps 3 to 6 are repeated until some termination conditions are reached, for example the maximum number of generations is lasted.

Even with some variations, most of the GAs will have the similar above operations. They mimic the natural selection at the Selection and Replacement steps, and simulates the sexual reproduction in the Reproduction step. Further details of GA can be referred to [2, 9].

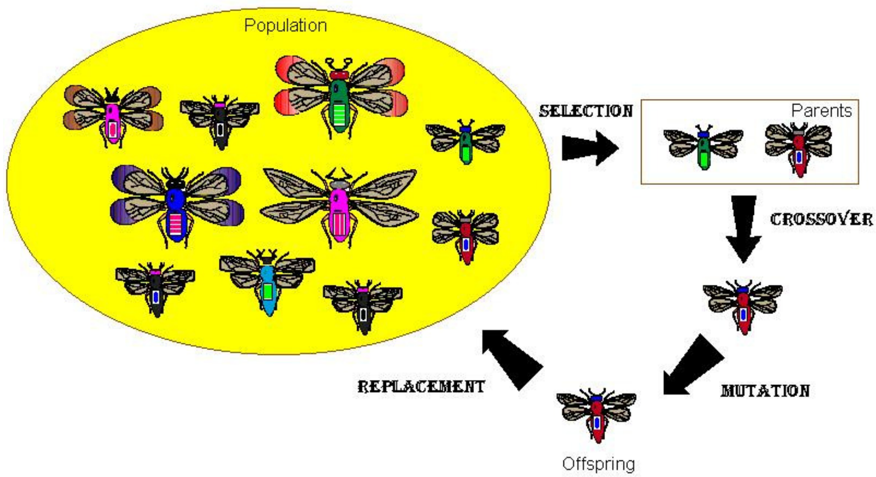

Fig. 2 Genetic algorithm cycle

2.3 Multiobjective Ranking

In order to achieve multiobjective optimization, the key is to assign the fitness of an individual according to its performance on the multiple objectives (i.e. Step 2 in Sec. 2.2). There are two commonly used Pareto ranking schemes which were proposed by Goldberg [2] and Fonseca & Fleming [3] respectively. They are briefly explained in the following subsections.

2.3.1 Goldberg Multiobjective Ranking

The rank of each individual is determined by the following procedures:

1. Set $p = 1$;
2. Determine all the non-dominated individuals in the population, and they are assigned as rank-p
3. Ignore those individuals with ranks having been assigned; increment p by 1 and go to Step 2 until all individuals have been processed.

An example of the ranking result is given in Fig. 3 (a), where the integer in the bracket shows the rank of the corresponding individual.

2.3.2 Fonseca and Fleming Multiobjective Ranking

The rank of each individual is $(1+q)$ where q is the number of other individuals in the population that dominate it. An example is given in Fig. 3 (b).

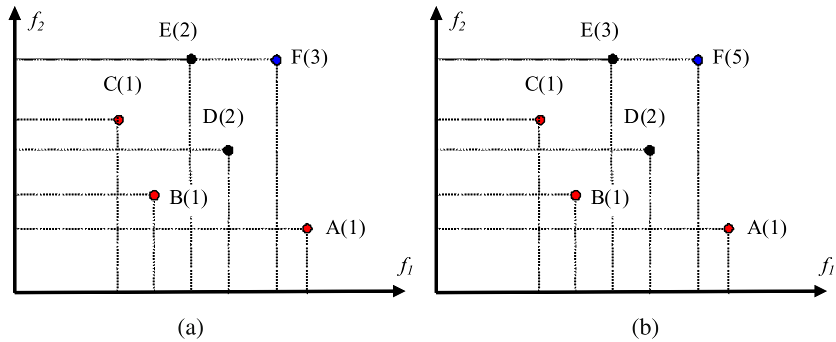

Fig. 3 Multiobjective ranking for minimizing f_1 and f_2: (a) Goldberg method (b) Fonseca and Fleming method.

By using the ranking as the fitness in the conventional GA described in Sect. 2.2, the basic framework of a mutliobjective GA can be formed. However, due to the stochastic nature and the processes, final solutions are likely to be crowded in a small region of the whole Pareto optimal set.

In order to explore the entire Pareto optimal set, diversity has to be maintained in the population. Different enhancements, such as high disruptive crossover, tournament selection, fitness sharing, crowding and so on, have been suggested. However, despite of many efforts being paid, it is still a difficult and challenging problem to find a widely spread solution set, especially the extreme ones, in MOP.

3 Computational Jumping Gene Operations

In 2004, inspired by the biological observation of jumping genes, an innovative optimization technique was proposed [10] to enhance the searching ability of a GA, in particular for MOPs. Two distinct computational operations are designed to mimic the biological gene transposition process.

3.1 Biological Background and Operational Design

The design of computational genetic transpositing operations is rooted onto the biological observation made by the Nobel Laureate in Physiology or Medicine, Barbara McClintock. In her pioneering work on maize chromosome, mobile genetic elements (called Jumping genes) were noticed and their transpositioning consequently leading to the changing patterns of coloration in maize kernels was reported [8].

Similar gene transposition processes have later been reported in different classes of organisms, including bacteria, plants and animals [11, 12]. The mechanism of transposition can be classified as conservative or replicative one. For conservative transposition, the mobile genetic element moves from its original locus to a new one without any replication, while it transposes by replication for the replicative case.

Based on these two mechanisms, two computational Jumping gene (JG) operations, called cut-and-paste and copy-and-paste, were proposed [10]. The schematic diagrams of these two operations are depicted in Figs. 4 (a) and (b), respectively.

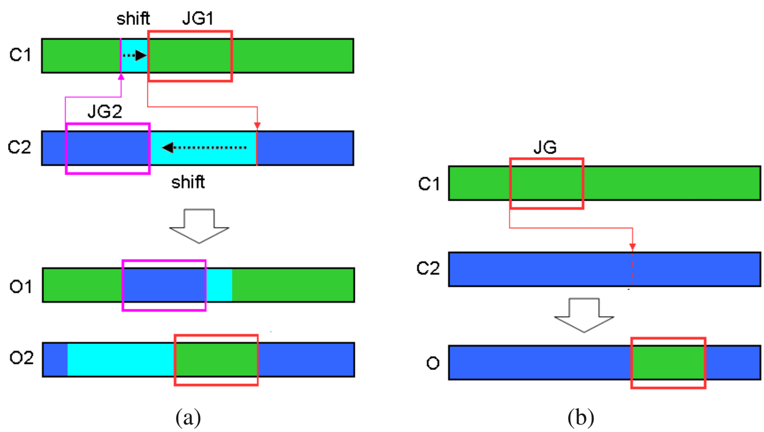

Fig. 4 (a) Cut-and-paste operation (b) Copy-and-paste operation

As shown in Fig. 4 (a), a section of genes (denoted as JG1 and JG2) is removed (or cut) from the original location of each of two chromosomes and then pasted into a new position of another chromosome when cut-and-paste is undergone. For copy-and-paste (see Fig. 4 (b)), the information of the JG is remained unchanged in the original chromosome, while a duplicate is pasted on another chromosome. In either case, the chromosomes and the JG locations are randomly chosen. Thus, these operations may occur within the same chromosome or two different ones.

3.2 Schemata Evolution Equations of JG Operations

In this section, it is to explain why these two JG operations can enhance the optimization process. A mathematical justificaiton is provided using the concept of schema.

Unlike the Holland's or other approximate approaches, the destruction and reconstruction of schemata under the two JG operations are derived, and hence exact schema evolution equations can be obtained [13, 14]. Similar approaches were adopted by Stephens and Waelbroeck, which gave an exact formulation (rather than a lower bound in the approximate approaches) for selection, single-point crossover and mutation [15, 16]. A major advantage of such an exact theorem is that the expected number of strings matching a schema can be predicted over multiple generations, which clearly overcomes the shortcoming of the approximate approaches. Moreover, this has also been used as a starting point for many other results on the behavior of a GA over multiple generations, based on the assumption of infinite population size.

Consider a schema ξ with L symbols, each symbol belongs to the set B={0, 1, *}, where 0 and 1 are called the actual bits while the wild character * is interpreted as a don't-care bit which can be either 0 or 1. Therefore, a schema represents a set of bit strings (i.e. these strings belong to the schema), for example, the bit strings 0100, 0110, 1100, and 1110 all belong to $\xi = *1*0$.

In the followings, some major definitions are given:

Definition 1. A map f_L is defined as $f_L : B^L \to V$, such that for $v \in V$, $v = f_L(\xi)$ returns a vector specifying the locations of all the actual bits in schema $\xi \in B^L$. V is a vector of non-negative integers, and it is assumed that the location begins with zero from left to right.

Definition 2. A map f_T is defined as $f_T : B^L \times V \to B^i$, such that for $\zeta \in B^i$, $\zeta = f_T(\xi, v)$ is formed by copying the symbols from schema $\xi \in B^L$ according to the locations specified in $v \in V$ where $size(v) = i$.

Definition 3. The primary schemata competition set of schema ξ is defined as

$$\Psi_\xi = \left\{\xi_i \in B^L : f_L(\xi_i) = f_L(\xi)\right\} \tag{2}$$

For example, given $\xi = *1*0$, $f_L(\xi) = [1,3]$ and $\Psi_\xi = \{*0*0, *0*1, *1*0, *1*1\}$.

Definition 4. For $x_i, y_i \in B$, the bit distance between two symbols x_i and y_i is defined as:

$$d(x_i, y_i) = \begin{cases} 1 & x_i = * \\ 1 & x_i \neq *, y_i \neq *, x_i = y_i \\ 0 & x_i \neq *, y_i \neq *, x_i \neq y_i \\ 0.5 & x_i \neq *, y_i = * \end{cases} \tag{3}$$

which describes how close y_i matches with x_i.

Definition 5. For $\zeta_1, \zeta_2 \in B^m$, the regional similarity of two strings ζ_1 and ζ_2 is specified by

$$\delta(\zeta_1, \zeta_2) = \prod_{j=0}^{m-1} d(\zeta_1(j), \zeta_2(j)) \tag{4}$$

where $\zeta_i(j)$ is the j-th bit of ζ_i.

Definition 6. For $v_1, v_2 \in V$ and $size(v_1) = size(v_2)$, $\Delta(\xi_1, v_1; \xi_2, v_2)$ is the regional similarity of ξ_1 in region v_1 and ξ_2 in region v_2, defined as

$$\Delta(\xi_1, v_1; \xi_2, v_2) \equiv \delta(f_T(\xi_1, v_1), f_T(\xi_2, v_2)) \tag{5}$$

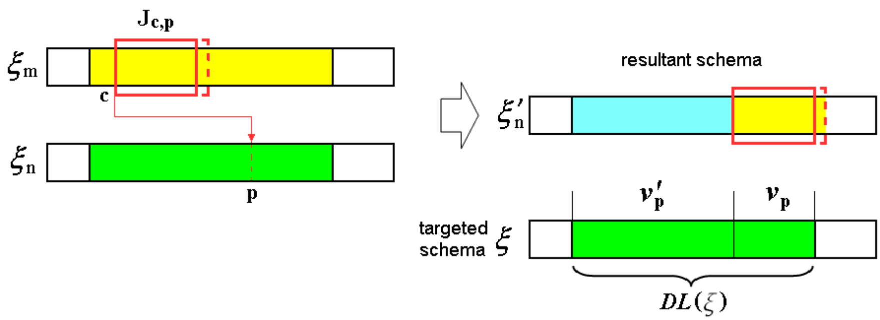

Fig. 5 Copy-and-paste operation with ξ_m and ξ_n, while ξ'_n is the resultant schema and ξ is the targeted schema.

By performing copy-and-paste operation, we can then derive the evolution equation of a particular schema ξ in each generation by studying its construction and destruction possibilities. The following schema evolution equation can be obtained [14]:

$$P(\xi,t+1) = (1-p_1)P(\xi,t) + \frac{p_1}{(L-L_g+1)^2} \sum_{\xi_m \in \Psi_\xi} \sum_{\xi_n \in \Psi_\xi} \sum_{c=0}^{L-L_g} \sum_{p=0}^{L-L_g} \left[\Delta(\xi,v_p;\xi_m,J_{c,p})\Delta(\xi,v'_p;\xi_n,v'_p)\right] P(\xi_m,t)P(\xi_n,t) \qquad (6)$$

where p_1 is the copy-and-paste operational rate; $P(\xi,t)$ is the proportion of ξ at generation t; L_g is the bit length of the JG, Ψ_ξ is the primary schemata competition set of ξ; $J_{c,p}$ specifies the bit locations of selected JG in ξ_m which fall into the defining-length region of the schema ξ (denoted as $DL(\xi)$) after pasted; v_p specifies the bit locations in $DL(\xi)$ where the JG is pasted; and v'_p specifies the bit locations in $DL(\xi)$ but not belong to v_p. The definitions of different regions under copy-and-paste operation are given in Fig. 5.

Similarly, the evolution equation of schema ξ by cut-and-paste operation can be derived [14] as below:

$$P(\xi,t+1) = (1-p_2)P(\xi,t) + \frac{p_2}{(L-L_g+1)^3} \sum_{\xi_m \in \Psi_\xi} \sum_{\xi_n \in \Psi_\xi} \sum_{c_m=0}^{L-L_g} \sum_{c_n=0}^{L-L_g} \sum_{k_n \notin \kappa(c_n)} \Delta(\xi,I_{c_n,p_n};\xi_m,J_{c_m,c_n,p_n})\Delta(\xi,S_{c_n,p_n};\xi_n,S'_{c_n,p_n}) \Delta(\xi,R_{c_n,p_n};\xi_n,R_{c_n,p_n})P(\xi_m,t)P(\xi_n,t) \qquad (7)$$

where p_2 is the cut-and-paste operational rate, J_{c_m,c_n,p_n} indicates the bit locations of the selected JG cut from ξ_m which fall into $DL(\xi)$ after pasted into ξ_n; I_{c_n,p_n} indicates the bit locations within $DL(\xi)$ where the JG is pasted onto ξ_n, S_{c_n,p_n} indicates the bit locations within $DL(\xi)$ where some bits of ξ_n are shifted, S'_{c_n,p_n} indicates the

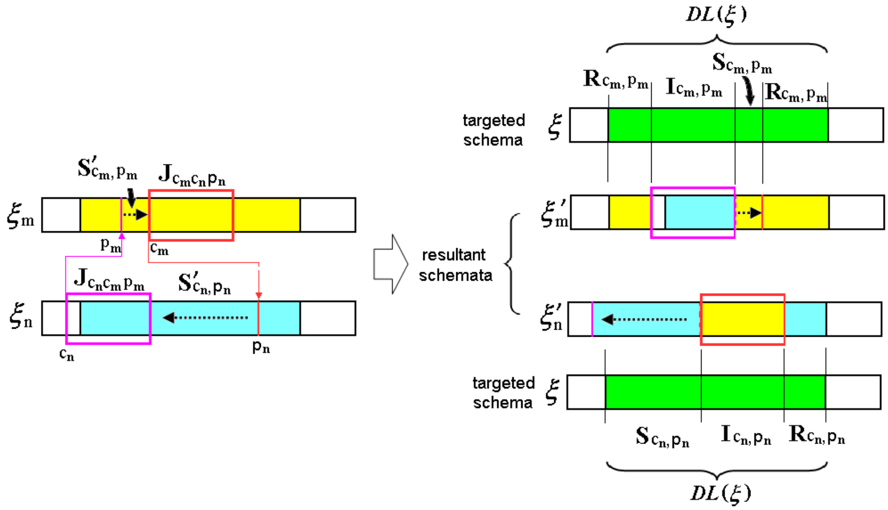

Fig. 6 Cut-and-paste operation with ξ_m and ξ_n, while ξ_m' and ξ_n' are the resultant schemata which are to be compared with the targeted schema ξ.

bit locations in ξ_n shift to S_{c_n,p_n} in ξ_n', R_{c_n,p_n} indicates the bit locations within where $DL(\xi)$ remains unaffected by the operation, and $\kappa(c_n) = (c_n, c_n + L_g]$. Figure 6 depicts the definitions of different regions under cut-and-paste operation.

To further analyze the evolution equations (6) and (7), let's consider any primary competitive group $\Psi = \{\xi_1, \xi_2, \cdots, \xi_k\}$ with $k = 2^{o(\Psi)}$ (Remark: The order of Ψ, $o(\Psi)$, is defined as the number of actual bits in ξ for any $\xi \in \Psi$.) and assume that $p_1 = p_2 = 1$. Define $P(\Psi,t) = \left[P(\xi_1,t)\ P(\xi_2,t)\ \cdots\ P(\xi_k,t) \right]$, (6) or (7) can be rewritten in the form of

$$P(\xi_i, t+1) = P^T(\Psi,t) A_{\xi_i} P(\Psi,t) \tag{8}$$

where A_{ξ_i} is a constant matrix with size of $k \times k$ and $\xi_i \in \Psi$. This constant matrix can be obtained when the JG length L_g and the primary competitive group Ψ are fixed, and it is different for copy-and-paste and cut-and-paste.

Equation (8) defines a discrete-time system of $P(\Psi,t)$ in a quadratic form, and the following theorem can be mathematically proved.

Theorem 1. *For any primary schemata competition group Ψ with $0 < o(\Psi) < L$, the proportion of every schema in Ψ globally asymptotically converges to $\frac{1}{2^{o(\Psi)}}$ with the use of copy-and-paste or cut-and-paste operation. [17, 18]*

Details of the proof can be referred to [13].

3.3 Discussions

Theorem 1 implies that either the copy-and-paste or the cut-and-paste operation will force a uniform distribution of schemata in each primary schemata competition set, despite of its initial distribution, i.e. $P(\xi,\infty) = \frac{1}{2^{o(\Psi)}}\ \forall\xi \in \Psi$. This ensures the occurrence of all the solutions in the population if the population size is sufficiently large, or in the other word, global search is guaranteed.

For a finite population, this driving force also enhances the searching of solutions. An illustrative example is given below, in which the possible outcomes of a 16-bit string generated by different operations are recorded. The initial population only contains 100 chromosomes and typical genetic operations are tested. It can be noticed in Fig. 7 that the cut-and-paste is the best and obtains all the possible outcomes within the smallest generations. The performances of copy-and-paste and mutation with high operational rate are good and similar.

In addition, the inclusion of copy-and-paste and cut-and-paste operation helps to maintain the diversity in a finite population. A simple case is given in Fig. 8. The simulation is based on a simple genetic algorithm with a population size of 100. The chromosome is in 16-bit string while the fitness is assigned only according to the values of 3 specified bit locations in a chromosome. The result clearly demonstrates that the dominance of the best schema is slow-down by the JG operations, while the cut-and-paste presents a more significant effect as compared with copy-and-paste (see Fig. 8 (b) and (c)). This matches with what we obtain from the previous example, where cut-and-paste is more powerful in terms of locating the solutions and promoting the diversity. More simulation results can be found in [14, 18].

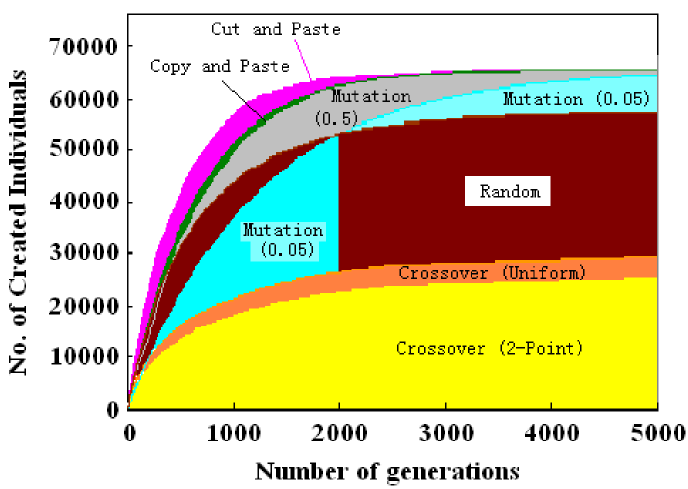

Fig. 7 Example showing the searching abilities of different operations with initial population of size 100 randomly generated.

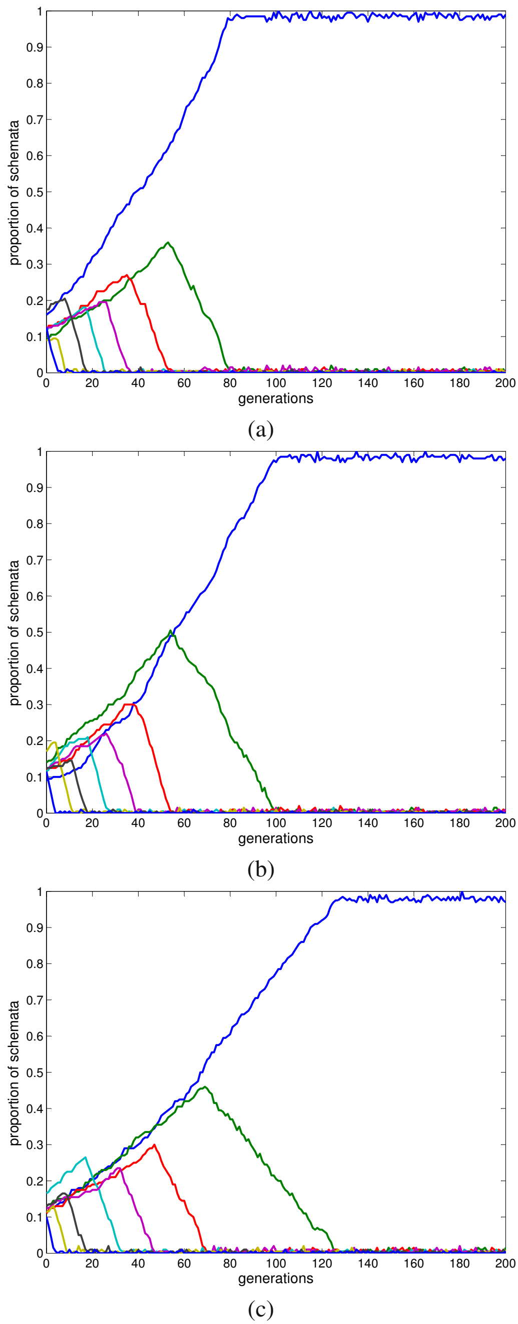

Fig. 8 Example showing the slow-down of the dominance of the best schema when cut-and-paste or copy-and-paste is applied in a standard GA. Proportion of schemata with (a) GA using crossover and mutation (b) GA using crossover, mutation and copy-and-paste (c) GA using crossover, mutation and cut-and-paste

4 Applications

The performance of JG-incorporated GA (known as JGGA) has been numercially studied in [19]. It is noticed that a rich set of non-dominated tradeoff solutions with better convergence and diversity can be obtained as compared with other MO evolutionary algorithms. Its outperformance is also reflected by its long list of applications for many real-world MOPs. Some of them are briefly summarized in the followings while the details can be referred to the cited references.

JGGA has been used to solve the radio-to-fiber repeater placement problem in wireless local loop systems [20]. Optimal locations for the repeaters can be duly determined so that the total repeater cost and total link cost are minimized simultaneously. A similar problem but targetting for placing base-stations for a wireless local area network in an IC factory is given in [21]. By finding a suitable number of base stations and identifying their corresponding locations, a specified quality of service is guaranteed. In [22], the resource management problem in wideband CDMA systems has been considered. It is concluded that JGGA can effectively optimize the total transmission power and the total transmission rate, and its performance is better than those common multiobjective algorithms.

Besides the resource management problems, JGGA has also been successfully applied in power voltage control [23], industrial product designs [24], and the designs of radio-frequency components and devices, such as the trapeizform U-slot folded patch feed antenna [25], the industrial, scientific and medical (ISM) band folded-patch antenna [26], the quarter-wave patch antenna [27], the planar monopole ultrawideband antenna [28] and microwave branch line hybrid coupler [14].

JGGA is capable of handling combinatorial optimization problems as well. In [29], JGGA has been applied to solve the job-shop scheduling problem whose goal is to find the best schedule of activities such that the length of schedule is minimized and the shared resources are efficiently used. In [30], a hybrid JGGA has been proposed to solve a scheduling problem in multiple destination routing. This is a large complex problem as it incorporates the scheduling and routing process of a set of requests which has a single source and multiple destinations in a network.

5 Conclusions

This paper is to summarize the fundamental concept, design of operations, performance justification and applications of the Jumping Gene approach. An overview has been presented and readers are encouraged to refer to the corresponding references for further details.

Acknowledgments. The first and third authors greatly appreciate the contributions from their former research students T.M. Chan, R.J. Yin and H.M. Ma.

References

1. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
2. Goldberg, D.E.: Genetic algorithms in search, optimization, and machine learning. Addison-Wesley (1989)
3. Fonseca, C.M., Fleming, P.J.: Multiobjective optimization and multiple constraint handling with evolutionary algorithms - Part I: A unified formulation. IEEE Trans. Syst. Man Cybern. Part A - Syst. Hum. 28, 26–37 (1998)
4. Fonseca, C.M., Fleming, P.J.: Multiobjective genetic algorithms made easy: Selection, sharing and mating restriction. In: Proc. Genetic Algorithms Engineering Systems: Innovations and Applications, pp. 45–52 (1995)
5. Srinivas, N., Deb, K.: Multiobjective function optimization using nondominated sorting genetic algorithms. Evol. Comput. 2(3), 221–248 (1994)
6. Goldberg, D.E., Richardson, J.: Genetic algorithms with sharing for multimodal function optimization. In: Proc. 2nd Int. Conf. on Genetic Algorithms (1987)
7. Oei, C.K., Goldberg, D.E., Chang, S.J.: Tournament selection, niching and the preservation of diversity. Technical Report (94002), Illinois GA Lab, University of Illinois (1994)
8. Fedoroff, N., Botstein, D. (eds.): The dynamic genome: Barbara McClintock's ideas in the century of genetics, Cold Spring Harbor, New York (1992)
9. Man, K.F., Tang, K.S., Kwong, S.: Genetic algorithms: concepts and applications. IEEE Trans. Ind. Electron. 43(5), 519–534 (1996)
10. Man, K.F., Chan, T.M., Tang, K.S., Kwong, S.: Jumping-genes in evolutionary computing. In: Proc. Annu. Conf. IEEE Ind. Electron. Society, vol. 2, pp. 1268–1272 (2004)
11. Finnegan, D.J.: Transposable elements in eukaryotes. Int. Rev. Cytology 93, 281–326 (1985)
12. Kleckner, N.: Transposable elements in prokaryotes. Annu. Rev. Genet. 15, 341–404 (1981)
13. Yin, R.J.: Schema Theorem for Computational Gene Transposition and Performance Analysis. PhD Thesis, City University of Hong Kong (2010)
14. Tang, K.S., Yin, R.J., Kwong, S., Ng, K.T., Man, K.F.: A theoretical development and analysis of jumping gene genetic algorithm. IEEE Trans. Ind. Inform. 7(3), 408–418 (2011)
15. Stephens, C.R., Waelbroeck, H.: Effective degrees of freedom in genetic algorithms and the block hypothesis. In: Proc. 7th Int. Conf. Genet. Algorithms, pp. 34–40 (1997)
16. Stephens, C.R., Waelbroeck, H.: Analysis of the effective degrees of freedom in genetic algorithms. Phys. Rew. D 57, 3251–3264 (1998)
17. Yin, J.J., Yeung, S.H., Tang, W.K.S., Man, K.S., Kwong, S.: Enhancement of multiobjective search: A Jumping-genes approach. In: Proc. IEEE Int. Symp. Ind. Electron., Vigo, pp. 1855–1858 (2007)
18. Tang, W.K.S., Kwong, S., Man, K.F.: A jumping genes paradigm: theory, verification, and applications. IEEE Circ. Syst. Mag. 8(4), 18–36 (2008)
19. Chan, T.M., Man, K.F., Kwong, S., Tang, K.S.: A jumping gene paradigm for evolutionary multiobjective optimization. IEEE Trans. Evolut. Comput. 12(2), 143–159 (2008)
20. Chan, T.M., Man, K.F., Tang, K.S., Kwong, S.: Multiobjective optimization of radio-to-fiber repeater placement a jumping gene algorithm. In: Proc. IEEE Int. Conf. Ind. Technol., pp. 291–296 (2005)
21. Chan, T.M., Man, K.F., Tang, K.S., Kwong, S.: A jumping-genes paradigm for optimizing factory WLAN network. IEEE Trans. Ind. Inform. 3(1), 33–43 (2007)

22. Chan, T.M., Man, K.F., Tang, K.S., Kwong, S.: A jumping gene algorithm for multiobjective resource management in wideband CDMA systems. Comput. J. 48(6), 749–768 (2005)
23. Ma, H.M., Ng, K.T., Man, K.F.: Multiobjective coordinated power voltage control using jumping genes paradigm. IEEE Trans. Ind. Electron. 55(11), 4075–4084 (2008)
24. Zheng, S.Y., Yeung, S.H., Chan, W.S., Man, K.F., Tang, K.S.: Design of broadband hybrid coupler with tight coupling using jumping gene evolutionary algorithm. IEEE Trans. Ind. Electron. 56(8), 2987–2991 (2009)
25. Yeung, S.H., Man, K.F., Luk, K.M., Chan, C.H.: A trapeizform u-slot folded patch feed antenna design optimized with jumping genes evolutionary algorithm. IEEE Trans. Antennas Propag. 56(2), 571–577 (2008)
26. Yeung, S.H., Man, K.F., Chan, W.S.: Optimised design of an ISM band antenna using a jumping genes methodology. IET Microw. Antennas Progag. 2(3), 259–267 (2008)
27. Sun, S.H., Man, K.F., Wang, B.Z., Wong, T.P.: An optimized wideband quarter-wave patch antenna design. IEEE Antennas Wirel. Propag. Lett. 4(1), 486–488 (2005)
28. Yang, X.S., Ng, K.T., Yeung, S.H., Man, K.F.: Jumping genes multiobjective optimization scheme for planar monopole ultrawideband antenna. IEEE Trans. Antennas Propag. 56(12), 3659–3666 (2008)
29. Ripon, K.S.N., Tsang, C.H., Kwong, S.: Multi-objective evolutionary job-shop scheduling using jumping genes genetic algorithm. In: Proc. Int. Joint Conf. Neural Netw., pp. 3100–3107 (2006)
30. Rahman, M., Mondol, S., Hossain, G.S., Dey, A.K.: A hybrid jumping genes genetic algorithm based request scheduling approach in multiple destination routing. In: Proc. Int. Conf. Inf. Commun. Technol., pp. 331–335 (2007)

Distributed Intelligent MEMS: Progresses and Perspectives

Julien Bourgeois and Seth Copen Goldstein

Abstract. MEMS research has until recently focused mainly on the engineering process, resulting in interesting products and a growing market. To fully realize the promise of MEMS, the next step is to add embedded intelligence. With embedded intelligence, the scalability of manufacturing will enable distributed MEMS systems consisting of thousands or millions of units which can work together to achieve a common goal. However, before such systems can become a reallity, we must come to grips with the challenge of scalability which will require paradigm-shifts both in hardware and software. Furthermore, the need for coordinated actuation, programming, communication and mobility management raises new challenges in both control and programming. The objective of this article is to report the progresses made by taking the example of two research projects and by giving the remaining challenges and the perspectives of distributed intelligent MEMS.

1 Introduction

Microelectromechanical systems (MEMS) have reached a state of design maturity which has led to some interesting prototypes and profitable products. While most MEMS devices have been used as independent elements of a larger system, this article deals with distributed MEMS systems composed of many MEMS devices which work together to achieve a global goal. The distinguishing feature of MEMS devices is that they are small and that they can be efficiently mass-produced. This naturally engender thinking of how they can be used together as a distributed system. Due to their small size, their low-cost and the fact that they can be mass-produced,

Julien Bourgeois
University of Franche-Comté, 1 cours Leprince-Ringuet, 25200 Montbeliard, France
e-mail: Julien.Bourgeois@univ-fcomte.fr

Seth Copen Goldstein
Carnegie Mellon University, Pittsburgh, PA 15213, USA
e-mail: seth@cs.cmu.edu

L. Kocarev (Ed.): ICT Innovations 2011, AISC 150, pp. 15–25.
springerlink.com

millions of units can be used in very small space. For example, a volume of less than 1 m^3 of 1mm-diameter silicon balls contains as many nodes as in the Internet. This characteristic requires paradigm-shifts both in hardware and software parts.

Past research focused on challenges of the engineering process, future challenges will consist in adding embedded intelligence to MEMS systems, so that they will be able to collaborate efficiently. This will require embedding MEMS sensors/actuators, electronics, communication capabilities, control of actuators and programs in the same unit. We suggest the use of the phrase ”distributed intelligent MEMS (diMEMS)” when referring to such systems. DiMEMS systems will certainly contain heterogeneous units. However, to simplify the programming challenge, we consider in this article only systems composed of homogeneous units.

Designing and managing diMEMS inherently requires multiple disciplines (e.g. hardware and software research). The challenges are therefore present in every field of research as well as in the integration of all the parts. In the 90's, DARPA Information Science and Technology funded a study on the state of the art and the perspectives of distributed MEMS. The conclusions of this report [7], published in 1997, were that the challenges involved in realizing diMEMS were mainly in controlling large numbers of MEMS sensors and actuators, the emergence of distributed intelligence, the use of MEMS devices as computational elements and the multiple-energy-domain simulation, analysis, and design. This article examines these challenges and the new ones that have been identified since 1997 using the results of two projects that have been conducted in the field.

2 Challenges

Many of the challenges raised by diMEMS have been studied in isolation in different research fields. However, in diMEMS they must be examined together and they become even more extereme. The scale of diMEMS needs new software paradigms as well as requiring new hardware capabilities. In the main, scaling up is the main concern of software challenges, while scaling down is the main concern of hardware challenges.

2.1 Software Challenges, Scaling Up

Scalability

Scalability is the main concern of diMEMS as the number of units will likely number in the millions. Scalability therefore impacts the way units will communicate. Systems using synchronous communications can't scale as well as those using asynchronous communications because the constraints of synchronization lower the system efficiency [6]. Synchronous communication is therefore difficult to achieve and asynchronous communications have already shown better results.

To ensure scalability, the programming model and the language must hide the complexity from the programmer and the compiler should enable programming the system as a single ensemble. Within the Claytronics project, new languages like LDP [11] and Meld [4] been have developed to cope with this challenge.

As scalability has to be tested up to millions of units, simulation tools also have to scale up. Dprsim [25] which has been developed within the Claytronics project has successfully simulated millions of units.

Uncertainty Tolerance

Faulty behavior is inherent to any diMEMS system. This is due to several factors. The batch process used in MEMS fabrication creates different levels of reliability. While some of the devices will have no defects, most of them have a high percentage of failed units. On the software side, this characteristic has to be handled and fault-tolerance has to be implemented.

In the case of mobile distributed MEMS a logical topology has to be maintained in order to communicate between the units. Maintaining a logical topology over a physical one is the concern of many research topics like P2P [1], swarm intelligence [18, 23], ad hoc networks [26] or wireless sensor networks [29]. Mobile distributed intelligent MEMS is even more complex than these examples. Mobility, scalability, fault-tolerance and limited processing capability are the main challenges to solve in order to create and maintain a logical topology.

Communications

The tradeoff between computation/communication/sensing is a challenge that has already been studied in wireless sensor networks but it needs some adaptation to take into account the scalability factor which is inherent to diMEMS.

Each diMEMS project has its own communication model directly linked to the application. The question here would be to study the cost and the interest of having some abstraction layer.

Control

When each unit of a system is mobile, the changes in the physical topology modify the logical topology by changing network connectivity. This is one of the main concerns of MANET. The inverse is also true, the logical topology can drive the mobility. This is usually done by covering an area which needs to be sensed [27] but it can also be used for modifying the logical topology, for example to keep connectivity in a sensor network [14].

DiMEMS are composed of actuators which needs control and a degree of synchronization. Three synchronization schemes can be used: no synchronization between MEMS units, means that the control loop doesn't have to synchronize with other units, local synchronization means that a MEMS unit has to be synchronized with its neighbors while global synchronization means that all the MEMS units have to act synchronously.

Having actuators to control requires real-time deadlines. Some applications need a very high frequency from the controler. If the control is fully decentralized like it is the case in distributed intelligent MEMS and that different modules requires a local or a global synchronization, the time to communicate has to be very short.

Reliability through Properties Verification

Reliability is difficult to achieve in any information technology project. The approach taken to achieve it often uses modularity which allows one to define interfaces and to segment the causes of failures. In diMEMS, this modularity is limited. Methods to model the whole system with VHDL-AMS and UML/SysML would allow one to verify some properties of the system and to increase its reliability.

CPS and IoT, Relations with Macro World

DiMEMS are systems that can interact with other intelligent systems. This interaction is the focus of cyber-physical systems (CPS) [22] and Internet of Things (IoT) [5]. The new challenge with distributed inteligent MEMS systems is to manage the different density of communication between the macro-objects (low density) and the micro-objects (high density).

2.2 Hardware Challenges, Scaling Down

Seamless Integration of MEMS and Logic

Integrating MEMS with CMOS is still a challenge in the fabrication process [30, 15]. Most of the MEMS-CMOS integration follow a hybrid integration through wire-bonding but this approach is not well-suited for diMEMS which requires too many connections. DiMEMS requires a monolithic integration for two reasons. First of all, only a monolithic approach can guarantee scalable and affordable fabrication process whereas hybrid approaches often requires manual intervention. Secondly, the weight of a hybrid system is more important than a monolithic one. MEMS actuation requires higher voltages than logic which can create problems. Some solutions have been proposed [20] to tackle this problem but a real voltage difference management between actuation and logic is still a challenge.

Designing Robust MEMS

Due to their size, MEMS are very sensitive to external factors (e.g., dust and air quality) which can change the behavior of certain types of MEMS actuators. Modeling and simulation have proven to be efficient to solve design issues [10] but new solutions have to be found to increase MEMS robustness.

Building Micro-communication Devices

DiMEMS needs communication capabilities, but integrating communication and MEMS is still a significant challenge. Some of them are linked to the previous challenges described here like the voltage difference between the actuation and the