A Multiset Genetic Algorithm for Real Coded Problems

António Manso[1,2]
1 - Instituto Politécnico de Tomar
Quinta do Contador - Estrada da Serra
2300-313 Tomar - Portugal
manso@ipt.pt

Luís Correia[2]
2 - LabMAg - Laboratório de Modelação de Agentes
Faculdade de Ciências da Universidade de Lisboa
Campo Grande, 1749-016 Lisboa – Portugal
Luis.Correia@di.fc.ul.pt

ABSTRACT
The Multiset Genetic Algorithm (MuGA) was adapted to real coded problems, tested in benchmark functions, and compared to competitive algorithms. Genetic operators were adapted to take into account the multiset representation of the population, which is the main distinctive feature and advantage of MuGA. The new operators extend existing ones, incorporating the influence of the number of copies each multi-individual has. Preliminary results obtained, without particular tuning efforts, position MuGA close to the best results obtained by other approaches. Future work will improve limitations found in maintaining a high genetic diversity.

Categories and Subject Descriptors
I.2.8 **[Artif cial Intelligence]**: Problem Solving, Control, Methods and Search.
D.2.2 [**Design Tools and Techniques**]: Evolutionary prototyping.

General Terms
Algorithms, Performance, Experimentation.

Keywords
Multiset, genetic algorithms, operators, real coding problems.

1. INTRODUCTION
MuGA is an approach designed towards producing a good balance between exploration and exploitation in evolutionary algorithms (EA), as well as a resource rationalization. It is described in more detail in section 2, but its most distinctive feature is to use multiset representation for populations [1].

Previous work on MuGA has concentrated on showing the benefits of simply replacing the conventional representation of population as a collection by the multiset representation [2]. However, using multisets opens a wide domain of new operators that only make sense with such a population representation. This paper presents the first initiative into the exploration of new operators specific for multiset based populations in real coded optimization problems.

2. MULTISETS GENETIC ALGORITHM (MUGA)
A multiset (or multiple membership set) is a collection of objects, called elements, which are allowed to repeat. We can define the multiset as a set of ordered pairs <*copies*, *element*> where *copies* is the cardinality associated to the *element.* MuGA (Figure 1) is a genetic algorithm in which populations are represented by multisets, called MultiPopulations (MP) and individuals represented by pairs <*copies*, *genotype*> called MultiIndividuals (MI). Introduction and removal of repeated individuals in a MP is done by incrementing and decrementing the number of copies of corresponding MI. By design, the algorithm, Figure 1, preserves the genetic diversity by maintaining constant the number of MI in the parents population, MP_0. Next we describe the new operators adapted to capitalize on the multiset representation.

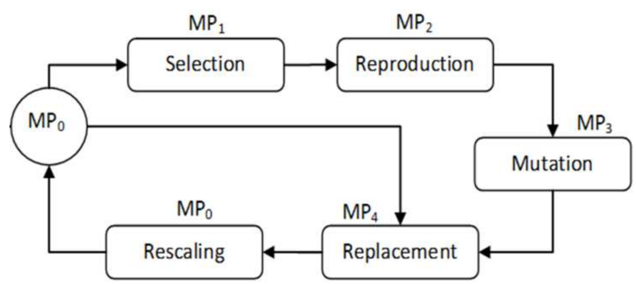


Figure 1- Operators and MultiPopulations in MuGA

Multiset Selection
Tournament selection is widely used in EA and presents unique features that allow an effective control of the selection pressure. The operator is not dependent on the range of values of the fitness function and avoids fitness scaling. These properties make this operator the right choice for MuGA selection. The selection of individuals for the tournament is done with uniform distribution over the elements of the population. The number of copies of an individual increases its probability of selection.

Multiset Reproduction
The main drawback of EA is the loss of genetic diversity by the repeated application of genetic operators over the population. Thus, MuGA employs a crossover operator that introduces genetic diversity in the population, using the number of copies of the MI to expand the range where offspring are created. The number of copies of individuals reflects its adaptation, and the higher it is, the more the offspring creation region is expanded in that direction. The line connecting the two parents is expanded in each direction by a factor of the number of copies of the corresponding parent. Offspring are selected from the extended line with a uniform random distribution. As a consequence, better fit individuals (with multiple copies) may generate offspring in wider areas, contributing in this way to increase genetic diversity.

GECCO'11, July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07

Multiset Mutation

The mutation operator uses the number of copies of the MI to reduce the range of mutation and better exploit its neighborhood. To execute a mutation the MI is expanded to an array of single individuals. The index of the array is used to shrink the range where genes peek their values. We use a Gaussian function with zero mean and standard deviation equal to the domain of the gene divided by the index of the array. Highly adapted individuals have many copies and therefore its mutants are progressively more close to the original MI as the index number increases. Individuals with poor fitness contribute with fewer copies and the mutation operator does not concentrate so much around them.

Multiset Replacement

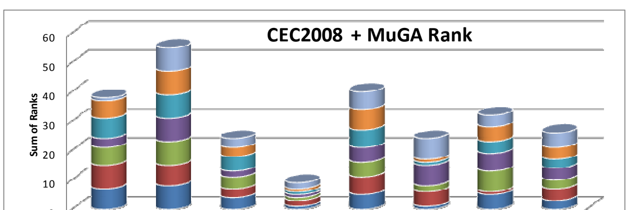
EA have the ability to explore large regions of the search space but are less successful in the exploitation of these promising regions. The hybridization with local search algorithms improves significantly the results of genetic evolution. In this work we use the local search in a novel way: a local search embedded in the replacement operator using Nelder-Mead Simplex method (NMS). We combine the fresh genetic material of MP_3 and the good individuals produced by the evolutionary process, MP_0, in the same simplex, and perform a limited local search. The individuals resulting from NMS go to the next generation.

Multiset Rescaling

The introduction of repeated elements in the MP causes an increase in the number of copies of the corresponding MI and the rescaling operator avoids the best individuals getting too many copies. In order to control the number of repeated elements, the rescaling operator divides the number of copies of each MI by a factor. The operator ensures that each MI has at least one copy and the total of individuals in MP is not greater than a constant defined as the double of the number of MI. Experimental results show that this value is a good compromise between selection pressure and genetic diversity.

3. EXPERIMENTAL RESULTS

To evaluate the performance of MuGA we chose a benchmark that poses difficulties and we compare our algorithm to state of the art optimizers. In order to maintain computation times within reduced values we opted for CEC 2008 benchmark problems of the Special Session & Competition on Large Scale Global Optimization with 100 variables [3]. This benchmark has seven functions with different degrees of difficulty. All except F7 have zero as minimum value. F7 is a fractal function without known minimum.



	MLCC	EPUS-PSO	JDEdynNP-F	MTS	DewSAcc	DMS-PSO	LSEDA-gl	MUGA
F7	1	8	3	2	6	7	4	5
F6	6	8	3	1	7	1	5	4
F5	7	8	5	1	6	1	4	3
F4	3	8	2	1	5	7	6	4
F3	6	8	4	1	5	2	7	3
F2	8	7	3	2	6	5	1	4
F1	7	8	4	1	5	1	5	3

Figure 2 – Rank of the CEC 2008 Algorithms and MuGA

Table 1. Statistics of MuGA results from the CEC 2008 benchmark functions after 500.000 function evaluations.

Statistic	F1	F2	F3	F4	F5	F6	F7
Best	2,76E-26	5,27E-01	5,64E+01	8,70E-14	0,00E+00	3,24E-14	-1,49E+03
Median	9,55E-26	6,54E-01	1,06E+02	1,39E-12	1,11E-16	5,73E-14	-1,46E+03
Worst	4,75E-25	8,06E-01	2,20E+02	1,35E-09	1,22E-01	4,17E-12	-1,43E+03
Mean	1,36E-25	6,66E-01	1,20E+02	9,07E-11	1,92E-02	2,43E-13	-1,46E+03
Std	1,22E-25	7,75E-02	4,04E+01	2,70E-10	3,62E-02	8,21E-13	1,47E+01

Table 1 shows the statistics of 25 runs of MuGA over the CEC2008 benchmark functions after 500.000 function evaluations. Figure 2 shows the rank of MuGA against the algorithms that competed in CEC2008. Overall MuGA is on second tier group. It has a robust behavior, patent in the consistency and quality of results over all benchmark functions. The rank of MuGA is only once below average (F7) and in the other functions it is always above that, although slightly.

4. CONCLUSION

This paper presents a first approach to real coded function optimization with a multiset based evolutionary algorithm, incorporating local search. We adapted each operator to the multiset representation, in order to take advantage of the number of copies in each multi-individual. The number of copies of each individual allowed modifications of the genetic operators so that a better balance between exploration and exploitation is achieved without compromising population stability.

Real problem optimization is known to need hybrid solutions and we have incorporated local search in MuGA in a novel way. The Nelder-Mead simplex search used in the replacement operator can take profit of a number of individuals higher than the population size. This is important to explore the fitness landscape in a wider range than using only one population, being it the original of a generation or the offspring population.

This first approach to MuGA in real coded problems was configured based on a set of different operators that were chosen to match with each other. This choice took into account exploration and exploitation capabilities of the operators as well as diversity maintenance in the population. Next efforts will be directed to identify the causes of limitations of MuGA observed in some functions of the used benchmark.

REFERENCES

[1] Aparício, J., Correia, L., Moura-Pires, F. *Populations are Multisets – PLATO.* W. Banzhaf, J. Daida, A. Eiben, M. Garzon, V. Honavar, M. Jakiela, and R. Smith, eds., Proc. GECCO, vol. Morgan Kaufmann, 1999, 1845–1850.

[2] Manso, A., Correia,L., *Genetic algorithms using populations based on multisets*. L. Seabra Lopes, N. Lau, P. Mariano, L. Rocha, eds., New Trends in Artificial Intelligence, EPIA 2009, 2009, 53–64.

[3] Tang, K., Yao, X., Suganthan, P., Mac-Nish, C., Chen, Y., Chen, C., Yang, Z., *Benchmark Functions for the CEC'2008 Special Session and Competition on LargeScale Global Optimization*, Nature Inspired Computation and Applications Laboratory, USTC, China, 2007.