HYBRIDIZE, HYBRIDIZE AND HYBRIDIZE AGAIN

Hugues Bersini
IRIDIA - CP 194/6
Universite Libre de Bruxelles
50, av. Franklin Roosevelt
1050 Bruxelles - Belgium

Abstract

These last years two global optimizations methods hybridizing Evolutionary Algorithms (EA, but mainly GA) with hill-climbing methods have been investigated. The first one involves two interwoven levels of optimization: Evolution (EA) and Individual Learning (hill-climbing), which cooperate in the global optimization process. The second one consists of modifying EA by the introduction of new genetic operators or by the alteration of traditional ones in such a way that these new operators reflect basic mechanisms of hill-climbing methods. The Immune Recruitment Mechanism, the simplex-GA and the STEP methods are different instances of this second type of hybridation. They already have been described and tested in previous papers and will be briefly recalled. With respect to previous works putting the emphasis on each hybrid separately, this new work looks at the further possibility of combining these two hybridations. "Hybridize, hybridize and hybridize again" is the basic message to be delivered in this paper. Since we believe these two methods of hybridization to be complementary rather than redundant (the first method makes the hill-climbing perform locally whereas the second globally), a a complete hybridization is advocated and it is experimentally shown that the resulting hybrid of hybrids gives the best results both for two real functions optimization and on the TSP and ATSP problems.

1 INTRODUCTION

A very pragmatic attitude increasingly met and advocated when using EA for resolving problems is not to use them in isolation but to combine them with a lot of existing optimisation heuristics. Well-known existing hybrids are GA + Simulated Annealing [] [] [], GA + Tabu Search [] [] [], GA + Simulated Annealing+Tabu Search [], GA + Simplex [][], GA + problem-dependent local hill-climber .... One basic motivation is that this merge aims at compensating for two basic weaknesses of Evolutionary Algorithms (EA). Firstly, while EA are good at global exploration, they are pretty bad at local exploitation. Secondly, while EA, more than any optimization methods before emphasize the use of a population-based search, they make very little use of the information provided by the relative fitness of the individuals in the population in order to choose the next point to evaluate. In so doing and in contrast with the great majority of optimization methods, they misuse the knowledge gained during the previous evaluations of the function to optimize?

Clearly, standard EA focus on the global facet of optimisation problem and the lack of attention paid to the local facet often prevents them from being of practical interest for many applications. Conversely, traditional hill-climbing methods (gradient descent, Quasi-Newton methods, simplex method) mainly concentrate on local exploitation. As a rule, the more intensive the local exploitation, the stronger the need for specialized information about the function to be minimized. For classical hill-climbers, global exploitation capacities are often non-existent and, consequently, their utility can be greatly reduced if they are used as such for the optimization of functions with numerous local optima. These considerations, highlighting the complementary properties of EA and hill-climbing methods, suggest that hybridization between both approaches may lead to improved performances. As a matter of fact, several forms of hybridization have already been suggested and tested in a large number of works [1] [4] [5] [6] [8] [9] [11] [13] [15] [17].

A first aim of this paper is to present two protocols for designing hybrid methods. The first one has turned to be a very classical one. It involves two interwoven levels of optimization, namely Evolution (EAs) and Individual Learning ('hill-climbing'), which cooperate in the global optimization process. EA generate a population of solutions then each solution in turn is improved by the hill-climber to finally give a new population upon which EA will act all over again. Some existing versions of it can be found in [1]

[9] [15] [17]. The second protocol consists of introducing new genetic operators or modifying the traditional ones (crossover, mutation, selection) in such a way that the new operators implement, in a certain manner, some basic principles of hill-climbing methods. The main motivation behind this second form of hybridation is that EA don't use, even during the global exploration, precious information given by the relative fitness of the individuals constituting the population. This second protocol is more original and less accepted in the EA community since it denatures to some extent the original biological inspiration (typical biological violations are multi-parents crossover, weighted by the relative fitness, simulated-annealing type of selection or memory-based type of search). Indeed less authors have explored this hybridizing approach [3] [4] [7] [11] since it is responsible for a "fuzzification" of the borderline between classical optimization algorithms and EA. IRM, GA-Simplex and STEP all aim at better using this information given by the individuals relative fitness.

In the next section, the first classical hybridation method is reviewed as well as the way it can be implemented both for real function optimization and the TSP problem. In section three the second type of hybridation is discussed. Three possible EA alterations allowing for a more informed global exploration are recalled: IRM, STEP and Simplex-GA. Simplex-GA will receive considerable attention and possible implementations of it for real functions and the TSP problem will be presented in greater details. Finally, the most original part of the paper is the object of section four where it is discussed and analysed the benefit obtained by combining the two hybrids in a final method which captures the respective advantages of each of them. In a nutshell we believe the first method to improve the local exploitation while the second improves the global exploitation, and this is why this further merging is achieved. The better results obtained by this further hybridation both on real function optimisations and TSP practically confirms the improvements allowed by this last complete hybridation.

2. FIRST HYBRID: EA + HILL-CLIMBING

2.1 Basic Principles

This well-known form of hybridation relies on the coupling of EA and hill-climbing methods (HC), using both the explorative capabilities, parallelism and combinative properties of EA, and the local exploitative power of HC. The principle is to let EA make selections based on the fitness at the end of the HC run on every individual of the population. EA are run successively on population further improved by HC. The traits acquired by the individual during her learning (i.e. at the end of the local hill-climbing) are hereditarily transmitted by sexual reproduction (in other words the genetic operations are performed on the individuals improved through HC); this is why such an approach has been labeled 'Lamarckian' in the literature (see also [1]).

2.2. A particular instance of this hybrid for real function optimization

Iteratively, EA produce better solutions using HC as an 'accelerator' mechanism thanks to the local exploitative properties of HC. In our experiments, the HC we used is the Powell's method which is a well-known optimization mechanism for N-dimensional real functions which, without resorting to any derivative of the function, first aims at discovering N mutually conjugate directions and then minimizes by a one-dimensional method the function along the line defined by the N directions. Its computer C code can be found in the numerical recipes [16]. Following every EA generation, there are as many parallel HC runs as individuals in the population. For the selection, crossover and mutation operations, EA work on the new populations of solutions obtained by HC. There is no need for using any termination criterion in order to limit the number of HC steps. Individual hill-climbing can be performed over a pre-defined and limited number of steps for two reasons: first because the main part of the information given by the HC search is acquired during the first steps, then because the search is pursued and refined over the next generation anyway. In practice the hybrid algoritm terminates by an "extended-life", namely the best individual of the last EA generation is improved through HC until to reach the normal Powell' termination criterion (often related to the importance of the progress achieved by the last optimizing step).

2.3 A particular instance of this hybrid for TSP and ATSP

For this famous combinatorial optimization problem, the hybrid works exactly as for the previous application. EA generate successive populations and each individual of the population is then improved using a local and problem-dependent optimisation mechanism. Again only a few steps of the local optimisations are performed. For the TSP, two very classical heuristics have been tested: 2-opt and 3-opt. In 2-opt, two edges of a selected tour are deleted and reconnected in a different way in order to reduce the length of the tour. In 3-opt, three edges are deleted. There are then four possible ways to reconnect these edges in order to obtain a valid tour. It is the one leading to the best improvement which is selected. For every individual, only a few random trials are performed to find an improving move. If none is found after a fixed number of trials, the individual is considered as "locally optimal" and left unchanged. While Powell's local optimisation heuristic can be considered as problem independent (it might work for whatever function optimisation problem), it is clear that 2-opt and 3-opt are local optimisation mechanism perfectly suited for the particular TSP problem, and it is very likely that another combinatorial optimisation problem would demand another more adapted local optimisation mechanism.

3. SECOND HYBRID: MODIFY EA AND MAKE THEM CLOSER TO CLASSICAL OPTIMIZATION METHODS

3.1 Basic principles

The goal here is to speed up and to enhance the global exploration performed by EA by limiting the search to more promising zones of the search space. One way to locate these zones is to better exploit the information provided by the relative fitness of the individuals in the population. As a matter of fact, in standard EA only at the selection stage is this information used and not at all during the generation of new candidates. For instance crossover merges randomly two parents without taking into account their relative fitness. More and more researchers argue for the use of multiparents crossover [][], like it was early proposed in the Immune Recruitment Mechanism [] and in the Simplex-GA [][]. However the importance lies not so much in increasing the materials to be recombined in one shot but in exploiting to its best the relative fitness of all parents.

One first attempt to make better use of this information is the Immune Recruitment Mechanism which tries to locate certain zones of recruitment where it might be a good idea to choose the next candidate to evaluate. The zones proposed then are regions near the good points and far from the bad ones. Before any multiparent offspring can be inserted in the population, it has to pass a recruitment test which forces it to reside in these promising zones. This approach seems to have given, for some problems, interesting results [] but unfortunately is much too vague and too complex to be analysed formally.

STEP (Select the Easiest Point) is a second more promising attempt in this same direction. The basic idea is to determine the promising zones as the ones in which the points, once evaluated, are the most likely to have fitness better than the best point found until now. In order to define the difficulty of exceeding the best point, STEP strongly relies on Brent's analysis and methods []. This is indeed a further illustration of the benefits gained by attentively looking in the good pages of classical optimisation literature in search for hybridizing evolutionary methods with more classical ones. For a one-dimensional optimisation problem, zones are segments and the difficulty of a segment (an easy segment is an interesting zone of recruitment) is given by the second derivative of a parabola which passes through the two extremities of a segment and exceeds the value of the best point found so far. In [], we shown some very good results for one-dimensional function as compared with Grefenstette's GENESIS and above all with Glomin which is one of the best and best-known algorithms for global optimization.

3.2 Simplex-GA for real function optimization

The idea lying behind the Simplex-GA is indeed to use the relative fitness of the individuals in the same way as the Simplex optimisation algorithm does it. Indeed why inventing a new recombination algorithm for real function optimisation when the Simplex recombines the individuals in such an optimal and well-motivated way ? To some extent Simplex and IRM share this same characteristic that the worst point among the ones to be recombined exerts a repulsive effect. While for search in binary space and for combinatorial optimisation, a Simplex form of recombination needs some thinking to be derived, its use is immediate for real function optimisation since it can just replace, in its standard form (the "Simplex" or "Flexible Polyhedron" due to Nelder and Mead [16] and also explained and described in the numerical recipes) whatever crossover already in use. The algorithm will be now briefly recalled (see [] for a more complete description). Let n be the number of variables of the function to be optimized. The Simplex recombination operator acts on groups of n+1 individuals. So first the current population must be subdivided in groups of (n+1) individuals, selected randomly. All n+1 individuals will constitute the parenthood of the future offspring. In each group only one individual is produced and replace the worst individual of its group in a way very similar to the type of replacement strategy found in IRM [][] and in GENITOR []. The standard alternative geometric transformations, reflection, contraction and extension, are attemped for generating an individual better than the ones currently composing the group.

3.3 Simplex-GA for TSP and ATSP

Since the Simplex algorithm was originally introduced for real function optimization, its extension for discrete and combinatorial optimization is far from immediate and indeed a lot of alternatives are possible. Departing from the uniform crossover, one possible form of Simplex crossover recombining three binary strings was already shown to compare favourably with standard crossover on the DeJong Benchmark functions []. Better performance resulting from the use of Simplex-GA for binary strings in the context of the classical neural network architecture discovery problem are also presented in [12]. To keep this idea of a repulsive effect exerted by the third string, the genes taken from this worst parent were just inverted before to be integrated in the offspring. The problem with the TSP is that there is no notion of “opposite” for an edge, and thus a new way of expressing the repulsive effect of the third parent must be imagined.

Three versions of the Simplex crossover were tested for TSP. Like the edge-recombination crossover which has been shown to be the best recombination mechanism, these three Simplex manipulate the edges rather than the cities.

First method: Three parents are selected and sorted according to their fitness. A starting point is randomly chosen. The new tour is then gradually built by selecting edges in the first two parents complying with the following rules:

* if there is an edge shared by the two “attractive” parents, it is selected.
* If no common edge exists, and one of the first two parents has an edge which is not in the third one, this edge is selected
* If no such edge is available, a city is randomly chosen to continue the tour, so that the new edge is not one of the third parent.

Example:

Parent P1: a b c d e f
Parent P2: c e b a f d (fitness(P1) ≥ fitness(P2) ≥ fitness(P3))
Parent P3: d c f b a e

The offspring: **a b e f c d**

Second method: This method works on the same principle, but with four individuals instead, the two last ones having the same repulsive effect. This operator prohibits more edges than he previous one, and is therefore more selective.

Third method: This last method really tries to give a meaning to the notion of “opposite” of an edge. In this case, the third parent does not only eliminate a region of the search space, it also orients the search. The method works as follows. First three parents are selected and sorted according to their fitness. A starting point is randomly chosen. The new tour is then gradually built by selecting edges in the first two parents complying with the following rules:

* if there is an edge shared by the two “attractive” parents, it is selected.
* If no common edge exists, and there are edges remaining in the two parents, then it selects the one leading to the most distant city in the third parent (with the highest number of edges between the two cities in the third parent)
* If no such edge is available, a city is randomly chosen to continue the tour, so that the new edge is not one of the third parent.

Example

Parent P1: a b c d e f
Parent P2: c e b a f d (fitness(P1) ≥ fitness(P2) ≥ fitness(P3))
Parent P3: d c f b a e

The offspring: **a b e f d c**

4. HYBRIDIZING THE HYBRIDS AND EXPERIMENTAL RESULTS

4.1 Basic Principles

One obvious question to be raised following the description of the two hybrids is: "are they equivalent ? Are the improvements brought by the first one the same as those brought by the second one ?". Recently, Kido et al. [9] showed that a good optimization algorithm can result from the combination of multiple local search strategies with GAs. Their algorithm was derived by hybridizing GAs with simulated annealing and TABU search, and their good experimental results justify this combination of various optimization schemes. Indeed hill-climbing can be performed in very different manners justifying thus the combination of these manners. Also a same problem can present different profiles, various local minima with different neighboods, rugged zones and smoother ones, jsutifying then the possible simulatenous use of different optimisation heurisitics.

The easiest way to answer this question on the possible redundancy was by means of further experiments done with a combination of the hybrids: the modified EA + the local optimisation. Importantly enough in the modified verion of the EA we used (an that will be described next more accurately), the simplex crossover was never the only possible recombination algorithm. We maintained several form of crossovers, keeping some less exploitative forms. The choice of one among the possible recombinations was done on probabilistic basis with sometimes very unexpected results for the TSP problems. EA with only the simplex crossover although imporving on EA with the binary crossover was never the optimal proposal. Instead a probabilistic choice among different crossovers always lead to better results.

4.2 The hybrid of hybrids for the real function optimization problems

This new optimisation algorithm uses as EA algorithm the real-coded version of GA originally proposed by Michalewicz with its particular and more adapted mutation and crossover mechanisms. The only change with the original form was the addition of an alternative Simplex crossover described in the previous chapter, but applied with probability ps. Two other crossovers were maintained. The first one applied with probability pd called discrete crossover with the offspring obtained by randomly taking the coordinate value of one or the other parent. The second one, called “continuous crossover” and “intermediate” in the evolutionary strategies papers [], applied with probability pc, and just performing an “averaging” operation on the coordinate values of both parents.

We tested this hybridation on two problems, the first one extracted from Michalewicz:

$$f(x) = \sum_{i=1}^{n} \sin(x_i).\sin^{2m}\left(\frac{i.x_i^{2}}{\pi}\right) \quad \text{for } x_i \in [0,\pi]$$

which has no “cross-terms” and one more challenging the Griewangk’s 10 dimension F8 function:

$$F_8(x) = \sum_{i=1}^{10} x_i^2/4000 - \prod_{i=1}^{10} cos(x_i/\sqrt{i}) + 1$$

For both cases, the best result was achieved by means of the hybridation. It improves on Powell’s method alone which although needs the smallest number of function evaluations, was not reliable at all. It improves on Michelwicz’s GA both in terms of accuracy and speed. Michalewicz’s GA + Powell method was faster than GA alone but slower than Powell alone. However it reaches a satisfactory fitness value much more better than independent parallel hill-climbers which shows the improvements brought by crossover and selection mechanisms. However the complete hybrid seems to reach an ideal compormise between the accuracy (the fitness) and the speed.

F8 has its global minimum f=0 at xi = 0, i=1..10. In 10 dimensions the functions has four suboptimal minima f(x) = 0.0074 at x = $(\pm\pi, \pm\pi\sqrt{2}, 0, 0)$. This function is difficult to optimize because non separable. Further the search algorithm has to climb a hill to get to the next valley. Mülhenbeim et al. have developed a parallel and Lamarckian GA where subpopulations try to locate good local minima. If a subpopulation does not progress after a number of generations, hill-climbing is done. Thus good local minima found by a subpopulation are diffused to neighbouring subpopulations. In the results they gave, the global optima was always found after, in the fastest case: 39240 function evaluations, and in the average: 59520 iterations.

Applying the complete hybridation with 4 mating groups (a group has to contained 11 individuals) the global minima can already be found after 8215 function evaluations which was the fastest performance we could reach. However in order to have a reliable algorithm always discovering the global solution, 10 mating groups were necessary and the speed of convergence turnsed out to be 20338 in the best case and 32223 in average. This was a considerable acceleration with respect to the results presented by Mühlenbeim et al. and convinced us that the combined hybrids algorithm could achieve very good performance by reinforcing the exploitation side of GA both at a global level (the Simplex) and at a local level (Powell's method).

4.3 The hybrid of hybrids for the TSP problems

The EA we used is based on the Genitor algorithm, implemented in the Genitor package provided by D.L. Whitley. Several experiments have shown the superiority of the edge-recombination operator which focuses on manipulating the edges between the cities rathre than the cities themselves. The data structure used in this case is an edge table in which all the edges of the two parents are stored. A new tour is created by selecting edges in ths table giving priority to edges in the following order:

- edges shared by two parents
- edges leading to cities with fewest remaining edges.

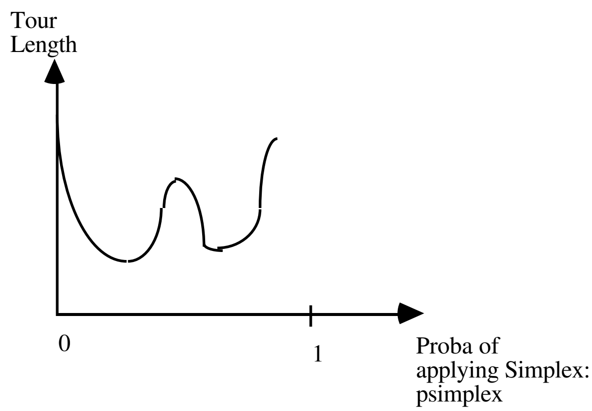
This recombination works better because indeed edges are recombined instead of cities and thus preserve the fact that all information lies in the relative position of the cities and not in their absolute position. The only disadvantage of this operator is its data structure which leads to an higher computation time. Here again we modified the Genitor algorithm by adding a possible crossover alternative, the Simplex-crossover whose selection rather than the original edge-recombination operator is done probabilistically. Again by substituting completely the edge-recombination by the Simplex-croosevr was not a very satisfactory move and maintaining this probabilistic choice was important.

Experiments have been performed on four 100-city problem (kroA100, kroB100, kroC100 and kroD100). The local optimisation heuristic was 2-opt and the Simplex-crossover was the first of the three methods described above.

The experiments give interesting and sometimes surprising results. Here are the experiemntal observations we have done:

- for all four problems, the use of 2-opt gives better results than 3-opt. This is quite interesting since the computational cost of 2-opt is lower than 3-opt.

- for all four problems, the quality of the solution found by the complete hybridation SImplex+2-opt is better than the one found by the edge-recombination +2-opt. However and more surprisingly by varying the probability of applying Simplex-Crossover instead of the original edge-recombination and plotting the final tour length obtained by the algorithms we saw typically the following graph in fig.1 with two gaps of values like for instance psimplex =0.2 and another at p=0.6 so two nearly equivalent optimal tuning one with a major use and one with a minor use of the simplex operation. Often the value 0.5, the same proportion of uses of the two crossover often achieves the worst results. No explanation have been found yet for this troubling phenomenon. We would have expected rather a simple parabola like indeed we found in the real function optimisation problem namely increasing simplex probability (psimplex) imporves the results up to a certain value.



- The good news concern the improving of the performance, sometimes very sensible like you can see on table 1(all these results have been averaged on 20 trials with different initial populations) and it was nearly always possible to find a good simplex probability that allows the discovery of the global optimum.

psimplex	kroA100	kroB100	kroC100	kroD100
0	21310	22320	20815	21390
0.2	21285	22225	20780	21440
0.7	21285	22225	20800	21390
1	21302	22285	20765	21380

- The bad news is that the number of function evaluations increases with the probability to apply the simplex operator so that, in contrast to what we expected, the increased use of the Simplex-operator slows down the algorithm. For instance it needs six times more evaluations to converge as compared with the use of the edge-recombination alone. Aimng simultaneosuly at ea very good results but without waiting too long would rathre favour the use of a small psimplex (± 0.2).

- We tested the two other simplex crossovers (4-parent simplex and the farthest-city simplex) which show poorer results than the previous one.

- We have finaly done some experiments for the Asymmetric TS which is well known to be harder to treat than the TSP and make the use of EA methods easy to argue for. The only possible local optimisation we could use so as to preserve the dierctionality of a tour was a restricted 3-opt, where only one of the possible four 3-opt moves is possible. We astill have to perform further experiemnts in order to confirm the results obtained so far but it seems the best algorithm is now a combination of a SImplex-GA but using the “farthest-city” simplex instead of the first one combined with the restricted 3-opt. Also in contrast with the TSP problem, the number of evuations needed for convergence seems to be stable whatever value of psimplex. Also again the length of the best solution shows a very surprising profile as a function of psimplex and even more irregular than for the TSP case so that it is very hard to independently of the ATSP problem provide the use with a fixed and robust psimplex.
4.4. The hybrid of hybrids fof the ATSP problem

5. CONCLUSIONS

This paper has shown the benefits gained by two possible ways of hybridizing EA with hill-climbing methods. Several works have already indicated the improvements gained by a Lamarckian form of hill-climber/GA coupling [1] [9] [15] [17], the first type of hybrid method proposed in the paper. A growing number of works also aim at improving evolutionary methods by substituting the standard genetic operations: selection, recombination and mutation, with operations reflecting better the classical optimization principles, the second type of hybrid method proposed in the paper. Among others, there are the simulated annealing type of selection [7] [11] and the adaptive hill-climbing type of mutations [2]. However the best experimental results were obtained by a further hybridization between the hybrids. This among other things leads us to think that the ideal optimization algorithm will always be a form of probabilistic and parametrized Meta-GA where not only the solutions are recombined but even the operators generating these solutions. For complex landscapes, in order to get to the highest place in the fastest way, it must be useful to combine different ways of climbing the hill depending on where you stand on the hill. The Darwinian selection might not only try to optimize the individuals but in the same time the way the new individual genomes are obtained by means of recombination and mutation. In a very distant future we can expect to see the number of sexual differentiation increasing such as to increase the number of parents. Wouldn't that be a more entertaining time ?

REFERENCES

[1] Ackley, D.H. and M.L. Littmann, 1993. "A Case for Lamarckian Evolution". In C.G. Langton, editor, Artificial Life III. MA. Addison Wesley - pp. 3-10.

[2] Bäck, T., Hoffmeister, F. and H.-P. Schwefel, 1991. "A survey of evolution strategies". In R.K. Belew and L.B. Booker (eds.) Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 2-9, Morgan Kaufmann.

[3] Bersini, H. and F.J. Varela, 1991. "The immune recruitment mechanism: A selective evolutionary strategy". In R.K. Belew and L.B. Booker (eds.) Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 520-526, San Mateo, CA, Morgan Kaufmann.

[4] Bersini, H. and G. Seront, 1992. "In Search of A Good Evolution-Optimization Crossover" - In Männer, R. and B. Manderick (Eds.) - Parallel Problem Solving From Nature 2. Amsterdam: North Holland. pp. 479-488.

[5] Bersini, H. and G. Seront, 1993. "Optimizing with the Immune Recruitment Mechanism" - Technical Report - TR/IRIDIA/93-6.

[6] Davis, L.D. (ed.), 1991. "The Handbook of Genetic Algorithms". Van Nostrand Reinhold.

[7] De la Maza, M. and B. Tidor, 1993. "An Analysis of Selection Procedures with Particular Attention Paid to Proportional and Boltzmann Selection" - In Forrest (Ed.) Proceedings of the Fifth International Conference on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann pp. 124 - 131.

[8] Goldberg, D.E. , 1989. "Genetic algorithms in search, optimization, and machine learning" Reading, MA: Addison-Wesley.

[9] Kido, T., Kitano, H. and M. Nakanishi, 1993. "A Hybrid Search for Genetic Algorithms: Combining Genetic Algorithms, TABU Search, and Simulated Annealing". In Forrest (Ed.) Proceedings of the Fifth International Conference on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann - pp. 641.

[10] Langerman, S., Seront, G. and H. Bersini, 1994. STEP: The Easiest Way to Optimize a Function - in Proceedings of the First IEEE Conference on Evolutionary Computation - pp. 519-524.

[11] Mahfoud, S.W. and D.E. Goldberg, 1992. "A genetic algorithm for parallel simulated annealing" - In Männer, R. and B. Manderick (Eds.) - Parallel Problem Solving From Nature 2. Amsterdam: North Holland. pp. 301-310.

[12] Maniezzo, V., 1994. "Genetic Evolution of the Topology and Weight Distribution of Neural Networks" - in IEEE Transactions on Neural Networks - Vol.5, No1 - pp. 39-53.

[13] Mansour, N. and G.C. Fox, 1991. "A Hybrid Genetic Algorithm for Task Allocation in Multicoputers". In R.K. Belew and L.B. Booker (eds.) Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 466-473, San Mateo, CA, Morgan Kaufmann.

[14] Michalewicz, Z., 1992. "Genetic Algorithms + Data Structures = Evolution Programs" - Springer Verlag Berlin Heidelberg.

[15] Mühlenbeim, H., Schomisch, M. and J. Born, 1991. "The Parallel Genetic Algorithm as Function Optimizer". In R.K. Belew and L.B. Booker (eds.) Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 271-278, San Mateo, CA, Morgan Kaufmann.

[16] Press, W., Teukolsky, S.A., Vettering, W.T. and B.P. Flannery. "Numerical Recipes in C" The Art of Scientific Computing - Second Edition - Cambridge University Press.

[17] Uckun, S., Bagchi, S., Kawamura, K. and Y. Miyabe, 1993. "Managing Genetic Search in Job Shop Scheduling". In IEEE Expert Intelligent Systems and Their Application - October 1993 - pp. 15 - 24.

[18] Whitley, D., Mathias, K. and P. Fitzhorn, 1991. "Delta Coding: An Iterative Search Strategy for Genetic Algorithms" - In Proceedings of the Fourth International Conference on Genetic Algorithms - Belew, R. and Booker, L.B. (eds.) - pp.77-84.

FIGURES

Figures 1(a,b,c,d): the results of the experimental tests on Michalewicz's test bed are compared in terms of maximum fitness (a), reliability (b), accuracy (c) and computing time (d).
Parameters Setting: ??????
Figures 2(a): the number of function evaluations required for finding the global minima of F8 (for the best and average case) as a function of the population size (= number of mating groups x 11) - 2(b): the percentage of failure as a function of the population size.

Parameters Setting: ?????