Evolution Strategy and Hierarchical Clustering

Ch. Magele†, O. Aichholzer‡, F.Aurenhammer‡, B. Brandstätter†,
H. Krasser‡, M. Mühlmann†, W. Renhart†
† Institute for Fundamentals and Theory in Electrical Engineering
‡ Institute for Theoretical Computer Science
Graz University of Technology, Kopernikusgasse 24, A-8010 Graz, Austria
e-mail: magele@igte.tu-graz.ac.at

Abstract— **Multi-objective optimization problems, in general, exhibit several local optima besides a global one. A desirable feature of any optimization strategy would therefore be to supply the user with as many information as possible about local optima on the way to the global solution. In this paper a hierarchical clustering algorithm is implemented into a higher order Evolution Srategy is applied to achieve these goals.**

I. Introduction

Stochastic methods have been widely accepted for the solution of real world optimization problems. A substantial difference to deterministic methods is, that stochastic methods take advantage of "random decisions". A lot of effort has been made to develop stochastic strategies into global optimization methods, being able to locate the absolute minimum with a very high probability. Higher order $(\mu/\rho, \lambda)$ Evolution Strategies (ES) [1] can be found among them as well as Genetic Algorithms, Simulated Annealing Algorithms [2] and many more. But most real world applications, which in general are multi-objective optimization problems, show several (or many) local solutions besides the global one, which are worth being inspected. Analyzing intermediate stages of many stochastic methods on the way to the final solution reveals, that these strategies pass local optima without giving notice to the user. Several "niching" methods have therefore been developed to overcome this problem [5]. In this paper, the application of a hierarchical cluster algorithm based on a complete linkage strategy [3], [4] is extended to detect as many individual sub-populations as possible temporarily gathering around local solutions. This can be performed by analyzing the static and the dynamical behaviour of cluster characteristics in the due course of the optimization process.

This method has successfully be applied to a multimodal two-dimensional test problems, a modified Rosenbrock function. Extended results including electromagnetic problems will be given in the full paper.

II. $[\kappa(\mu(\tau)/\rho, \lambda)]$ Evolution Strategy

A $[\kappa(\mu(\tau)/\rho, \lambda)]$ ES sets up κ sub-populations, where μ is the average number of parents per subpopulation. Each parent survives at most for τ generations. The total number of individuals therefore is $\kappa * \mu * \tau$. In each generation, ρ out of all mating parents take part in the recombination process to produce $\kappa * \lambda$ children. All children have then to undergo the mutation operation. The $\kappa * \mu$ best children then replace the oldest parents and the iteration process continues.

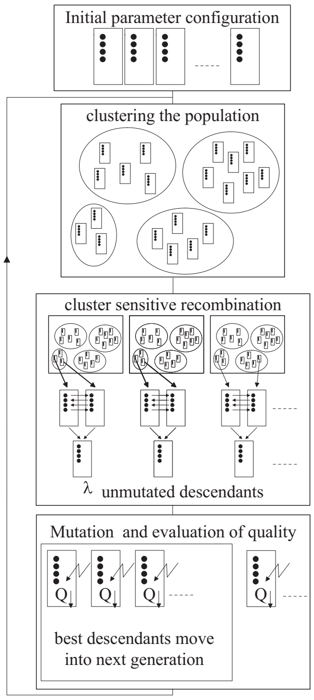

Fig. 1. Higher order $[\kappa(\mu(\tau)/\rho, \lambda)]$ evolution strategy

In general, the number of local solutions is not known beforehand, therefore a method based on the hierarchical clustering of the total population has been developed to construct the parameter κ adaptively. Once the number of clusters (κ) is fixed for the current generation, this information is used to perform the recombination process.

A. Hierarchically Clustering the Population

To apply clustering we treat every parameter configuration as a single object. Grouping these objects together is done by complete linkage (C-link), an agglomerative hierarchical clustering method [3], [4]. This class of methods starts with n single objects (parameter configurations), all seen as their own clusters. Step by step two specific clusters are then merged into a new cluster until, after $n-1$ steps, all objects form a single final cluster, the root of the resulting hierarchical cluster tree (Figure 2). The two clusters to be merged in the next step are chosen in a way that the dissimilarity of the resulting cluster is minimized among all possible cluster pairs. C-link takes the cluster diameter (maximum distance of objects in the cluster) as the measure of dissimilarity. For real world applications, where it is a priori not known how many (i.e. few) clusters are required, parameters like cluster diameter can be used to dynamically determine the number of clusters κ.

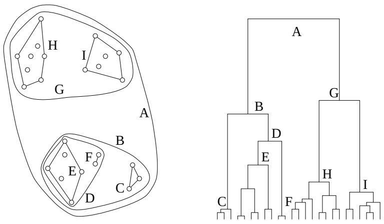

Fig. 2. Hierarchical clustering and corresponding cluster tree

B. Cluster Sensitive Recombination

Since the optimization parameters are floating point numbers, recombination is performed by *arithmetic crossover* [2]. The candidates for recombination are chosen in the following way.

- Choose one of the κ clusters randomly
- Choose one candidate, p_1, from this cluster, taking its fitness into account (roulette wheel) [2]
- Choose the second candidate, p_2, such that members from the current cluster are more likely than members from remote clusters

The last rule is implemented using the C-link cluster tree.

III. Modified Rosenbrock Function

The unimodal Rosenbrock function was modified to show three distinct minima (1).

$$f_{rosen,mod}(\mathbf{x}) = f_{rosen}(\mathbf{x}) - 50((x_1+1)^2 + (x_2-1)^2) \\ -10((x_1-1.5)^2 + (x_2-2.5)^2) + e^{2x_2-5} \quad (1)$$

Fig. 3 shows an intermediate state of the optimization process, when the population is spread over all possible solutions.

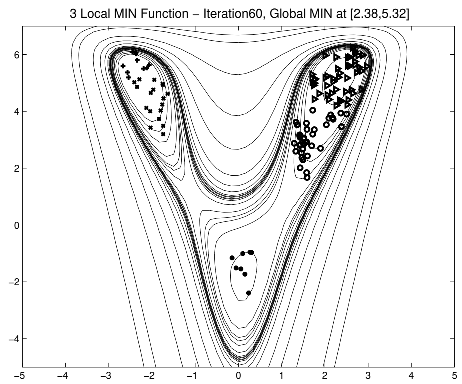

Fig. 3. Modified Rosenbrock function, population spread over three minima

Taking a closer look at the behaviour of the cluster diameter versus the number of clusters (Fig. 4), two peculiarities can be seen. First of all, each diameter function converges

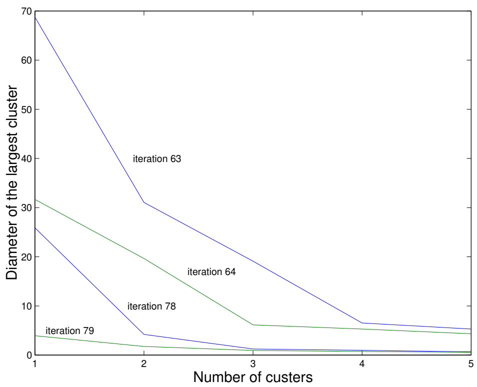

Fig. 4. Diagrams of the largest cluster diameters at different iteration steps

to a final value, which is reached, when the number of detected minima is exceeded (static behaviour). Secondly, the diameter of the largest cluster decreases significantly, if the individuals give up to populate one of the local minima (dynamic behaviour). Using these information and storing the best values of each detected minimum throughout the optimization process allows to identify a lot of solution in the end. This method has also been applied to the 8 parameter SMES optimization problem [6] and numerous local minima were detected. These results will be presented in the full paper.

IV. Conclusion

Evolution Strategies are already very powerful and still easy-to-use methods to solve complex, multimodal optimization problems. Applying a *cluster sensitive recombination*, which requires very little modification to a standard $(\mu/\rho, \lambda)$ Evolution Strategy, allows to identify the number and location of local solutions (if such exist) during the iteration process. This enables the user to choose among a set of solutions rather than being forced to accept the best one. The increasing number of function calls can well be justified by the number of solutions obtained.

References

[1] Rechenberg, I.: "Evolutionsstrategie '94" frommann-holzboog, Stuttgart, 1994

[2] Alotto P.G., Brandstätter B., Cela E., Fürntratt G., Magele Ch., Molinari G., Nervi M., Preis K., Repetto M. and Richter K.R.: "Stochastic Algorithms in Electromagnetic Optimization" IEEE *Trans. Magn.*, vol. 34, No. 5, pp 3674-3684, 1998

[3] H.H. Bock, Automatische Klassifikation, Studia Mathematica, Vandenhoeck & Ruprecht, Göttingen, 1974

[4] J. A. Hartigan, Clustering Algorithms, Wiley Series in Probability and Mathematical Statistics, John Wiley & Suns, NY, 1975

[5] Sareni B., Krähenbühl L.: "Fitness Sharing and Niching Methods Revisited" IEEE *Trans. on Evolutionary Computation*, vol. 2, No. 3, pp 97-106, 1998

[6] Takahashi N., "Optimization of Die Press Model", *Proceedings of the TEAM Workshop in the Sixth Round*, March 20-21, Okayama, Japan, 1996.