Multi-criteria optimisation, constraint handling with GAs

Carlo Poloni, V.Pediroda
Dipartimento di Energetica
Università degli Studi di Trieste
Via Valerio 10, 34100 Trieste, ITALY
e-mail: poloni@univ.trieste.it

Introduction

Genetic Algorithms have been used in several engineering problems with clear advantages over other more traditional algorithms. The major advantages of these techniques are mainly related to robustness of the procedure. In simple GA applications however this major advantage is paid with computational cost: a large number of fitness evaluation is needed to reach a satisfactory solution.

If the fitness function is computed by means of complex simulation codes the total cost of the approach may make the problem impossible to be faced.

With the continuing grow of computing resources available the attention of engineers has modified the role of complex simulation that is more and more used directly in the design process. This aspect has also underlined the substantial weakness of traditional optimisation approaches that can usually produce only single-objective optimised solution and only if the objective function satisfy continuity and often derivability conditions. This fact together with the need of multidisciplinary approach to design caused a growing interest into the use of Genetic Algorithms as general purpose optimisers. A large number of examples of engineering application can in fact be found in the literature [1].

Most real-life design procedures are complex tasks that have to deal with multi disciplinary environments, not always clearly defined targets, constraints to be satisfied. In this sense even tough the target of the optimisation could be expressed with a single expression like: "do the best possible design", the optimisation process must consider several different usually conflicting objectives and the compromise obtainable might not be a-priori known. The possibility of looking not only for a single good solution but for a set of solutions (the Pareto Set) [11] that satisfy different levels of compromise might be of great help to the decision maker that must select the most suitable one.

Three main issues makes however GA more than attractive and maybe unique among the aerodynamic design optimisation methods: GA are usually much more robust than gradient based algorithm and can tolerate even approximate or noisy design objectives evaluation, GA can be efficiently parallelised and can therefore take full advantages of the massively parallel computer architecture, GA can directly approach a multi-objective optimisation problem [8], [9], [10].

It must be noted however that the main concern related to the use of Genetic Algorithm for engineering problems involving the use of complex simulation codes is the computational effort needed for the accurate evaluation of a design configuration that, in the case of a crude application of the technique, might leads to unacceptable computer time if compared with other more classical algorithms [7]. With the help of parallel supercomputers [5],[6] and considering the fact that the computational performances of available machines is continuously growing, this problem at first glance might seem to be solvable by the computer technology development. However it is also known that the most powerful today available computer is still far from having sufficient performance even for single "multiphisics" simulation and therefore any effort in the direction of computational cost reduction of the optimisation process should be seen at least as an opportunity to face more challenging design problems.

Two major paths exists in the reduction of computational effort: 1) improve the efficiency of the optimiser and 2) reduce the time needed for the single fitness calculation. The first can be faced by suitable operators the latter by the adoption of approximation techniques.

Optimisation Algorithm

Typical optimisation problems are usually solved by means of "hill climbing procedure" possibly based on local gradients of a stated "cost function". The typical drawback of this approach is the fact that the search for improvements is done efficiently but is done locally.

On the other hand probabilistic optimisation techniques can be used to examine a large but discrete configuration space in order to find a "good solution" possibly close to the global optimum.

In this paper some of the possible operators are briefly illustrated pointing out the most suitable to engineering problems, like aerodynamic optimisation, for which the objective functions topology is not too complex (not-deceptive) but the calculation of the objectives is computationally intensive.

A genetic algorithm in the "generational form" has in the end a structure as follow:

```
do ng generation
     do nind individuals
          translate bits into variables
          compute objectives => interface to analysis
     end do
     do some statistics on the population individuals
     do Create a new population:
          by cross over:
               select  individual
               and reproduce
          by mutation:
               select  individuals
               and mutate
```