Evolutionary Techniques for Constrained Multiobjective Optimization Problems

Fernando Jiménez
Dept. Informática, Inteligencia Artificial y Electrónica
Universidad de Murcia
e-mail: fernan@dif.um.es

José L. Verdegay
Dept. Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada
e-mail: verdegay@goliat.ugr.es

Antonio F. Gómez-Skarmeta
Dept. Informática, Inteligencia Artificial y Electrónica
Universidad de Murcia
e-mail: skarmeta@dif.um.es

Abstract

Evolutionary computation techniques for constrained and multiobjective optimization problems are combined in this paper. Simulation results show a good performance by this method for several test problems with different structure.

1 INTRODUCTION

We consider the following general constrained multiobjective optimization problem:

$$\begin{array}{lll} \text{Minimize} & f_i(x,y,z), & i=1,\ldots,n \\ \text{subject to} & g_j(x,y,z) \leq 0, & j=1,\ldots,m \end{array} \tag{1}$$

where $x = (x_1,\ldots,x_p) \in R^p$ is a p dimensional real-valued parameter vector, with $l_i^x \leq x_i \leq u_i^x$ ($l_i^x, u_i^x \in R$, $i=1,\ldots,p$, $p \geq 0$), R stands for the real line, and no real-valued parameter is assumed when $p=0$; $y=(y_1,\ldots,y_q) \in Z^q$ is a q dimensional integer-valued parameter vector, with $l_i^y \leq y_i \leq u_i^y$ ($l_i^y, u_i^y \in Z$, $i=1,\ldots,q$, $q \geq 0$), Z represents the integer number set, and no integer-valued parameter is assumed when $q=0$; $z=(z_1,\ldots,z_r) \in \{0,1\}^r$ is an r dimensional boolean-valued parameter vector ($r \geq 0$), and no boolean-valued parameter is assumed when $r=0$; $f_i(x,y,z)$ ($i=1,\ldots,n$, $n \geq 1$), $g_j(x,y,z)$ ($j=1,\ldots,m$, $m \geq 0$) are linear or nonlinear arbitrary functions, and no constraint is assumed when $m=0$. In this paper we show an Evolutionary Algorithm (EA) to find multiple nondominated solutions (Srinivas and Deb, 1995) to general constrained multiobjective optimization problems as in (1), in which a problem-independent technique for constraint satisfaction is used.

2 THE EVOLUTIONARY ALGORITHM

Our EA for constrained multiobjective problems is basically a Genetic Algorithm (Goldberg, 1989) because of overall sequence of operations, but a "natural" representation of solutions is used (floating-point numbers to represent real-valued parameters, integer numbers to represent integer-valued parameters, and boolean values to represent boolean-valued parameters). Thus, an individual V of the population is represented as $V=\{x,y,z\}$, where $x=(x_1,\ldots,x_p)$, with $x_i \in R$ ($i=1,\ldots,p$), $y=(y_1,\ldots,y_q)$, with $y_i \in Z$ ($i=1,\ldots,q$), and $z=(z_1,\ldots,z_r)$, with $z_i \in \{0,1\}$ ($i=1,\ldots,r$). To handle constraints, and simultaneosly, to identify nondominated solutions in the current population, we consider the following general assumptions: (1) populations are composed of feasible and unfeasible individuals; (2) feasible individuals evolve towards optimality (noninferiority) guided by an *optimality evaluation function* in each objective and using the nondomination concept; (3) unfeasible individuals evolve towards feasibility guided by an *unfeasibility evaluation function*; (4) feasible individuals have greater probability of selection than unfeasible individuals; (5) a niche formation technique is used to maintain diversity in the populations. According to these criteria, the procedure to evaluate a population can be designed as figure 1 shows. In this algorithm POP(s).IND represents the individual which is placed at the position s in the current population, POP(s).EVAL(i) represents the evaluation in the objective i of the feasible individual which is placed at the position s in the current population, POP(s).UNFEAS represents the evaluation of the unfeasible individual which is placed at the position s in the current population, and *popsize* is the population size. Thus, for the feasible individuals, the optimality evaluation function in each objective is used in conjuntion with

```
procedure evaluate_population
begin
   s ← 0
   while s ≤ popsize do
   begin
      s ← s + 1
      V ← POP(s).IND
      if feasible(V) then
         POP(s).EVAL(i) ← f_i(x, y, z),  i = 1, ..., n
         {optimality evaluation function in each objective}
      else
         POP(s).UNFEAS ← max{g_j(x, y, z),  j = 1 ..., m}
         {unfeasibility evaluation function}
   end
end
```

Figure 1: Evaluation of individuals in a population.

the nondomination concept to proporcionate greater probability of selection to the nondominated individuals in the current population than dominated individuals. The probabilities of selection for the unfeasible individuals come given according to the unfeasibility evaluation function which have to be minimized. An individual $V = \{x, y, z\}$ is feasible if $g_j(x, y, z) \leq 0$ for all $j = 1, \ldots, m$. However, we allow a violation $\Delta \geq 0$ and then an individual $V = \{x, y, z\}$ is feasible if $g_j(x, y, z) \leq \Delta$ for all $j = 1, \ldots, m$.

We can use different techniques to generate a new population (Jiménez and Verdegay, 1998). The proposed EA uses *tournament selection*, an explicit *sharing function* for niche formation, *complete generational replacement* and *elitism strategy*. In this tournament selection, two individuals are picked at random from the population in order to select a winner in a tournament selection. The tournament is as follows: (1) If both individuals are feasible, then they are compared to the feasible members of a comparison set which contains a specific number *tourn* of individuals picked at random from the population. If one of them is nondominated in the comparison set and the other is dominated, then the nondominated individual is selected. If both are either nondominated or dominated in the comparison set, a niche count is found for each individual in the entire population, and then the individual with the least niche count is selected; (2) If one of them is feasible and the other is unfeasible, then the feasible individual is selected; (3) If both individuals are unfeasible, then they are compared to the unfeasible members of a comparison set which contains a specific number *tourn* of individuals picked at random from the population. If one of them is better than the best unfeasible individual in the comparison set (according to the unfeasibility evaluation function) and the other is worse, then the first individual is selected. If both are either better or worse than the best unfeasible individual in the comparison set, a niche count is found for each individual in the entire population, and then the individual with the least niche count is selected.

Many variation operators have been proposed during the last 30 years. (Michalewicz and Schoenauer, 1996). We finally use *uniform crossover* (applied to all the components of the chromosome), *nonuniform mutation* (applied to the components with floating-point representation) and *uniform mutation* (applied to the components with integer and boolean representation).

3 EXPERIMENTS, CONCLUSIONS AND FUTURE WORKS

The main focus of this work has been to combine a problem-independent evolutionary constraint-handling technique with an evolutionary multiobjective optimization technique to find multiple nondominated solutions to general constrained multiobjective optimization problems. An important set of multiobjective programming problems such as linear, nonlinear, integer, boolean and mixed can be solved by using this evolutionary computation technique. EA have been tested by a large number of test problems and simulation results showed a good performance. All the procedures, the different test problems, and simultation results, can be obtained from the authors on request. As main future works we consider a modality of the proposed constraint-handling technique in which constraints are treated as objectives and nondomination concept is used to make evolve the unfeasible individuals toward the feasible region.

Acknowledgement

The authors thank the CICyT for the support given to this work under the projects TIC97-1343-C02-02 and 1FD97-0255-C03-01(02).

References

Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Massachusetts.

Jiménez, F., Verdegay, J.L. (1998). Constrained multiobjective optimization by evolutionary algorithms. *Procs. of the International ICSC Symposium on Engineering of Intelligent Systems (EIS'98)*, 266-271, University of La Laguna, Tenerife, Spain.

Michalewicz, Z., Schoenauer, M. (1996). Evolutionary Algorithms for constrained parameter optimization problems. *Evolutionary Computation* **4**(1):1-32.

Srinivas, N., Deb, K. (1995). Multiobjective optimization using nondominated sorting in Genetic Algorithms. *Evolutionary Computation* **2**(3):221-248.