A Simple Evolution Strategy to Solve Constrained Optimization Problems

Efrén Mezura-Montes and Carlos A. Coello Coello

CINVESTAV-IPN
Evolutionary Computation Group (EVOCINV)
Departamento de Ingeniería Eléctrica
Sección de Computación
Av. Instituto Politécnico Nacional No. 2508
Col. San Pedro Zacatenco
México D.F. 07300, MÉXICO
emezura@computacion.cs.cinvestav.mx
ccoello@cs.cinvestav.mx

1 Our Approach

In this paper, we argue that the self-adaptation mechanism of a conventional evolution strategy combined with some (very simple) tournament rules based on feasibility similar to some previous proposals (e.g., [1]) can provide us with a highly competitive evolutionary algorithm for constrained optimization. In our proposal, however, no extra mechanisms are provided to maintain diversity. In order to verify our hypothesis, we performed a small comparative study among five different types of ES: $(\mu \overset{+}{,} \lambda)$-ES with and without correlated mutation and a $(\mu + 1)$-ES using the "$1/5$-success rule". The tournament rules adopted in the five types of ES implemented are the following: Between 2 feasible solutions, the one with the highest fitness value wins, if one solution is feasible and the other one is infeasible, the feasible solution wins and if both solutions are infeasible, the one with the lowest sum of constraint violation is preferred. To evaluate the performance of the five types of ES under study, we decided to use ten (out of 13) of the test functions described in [2]. The $(\mu+1)-ES$ had the best overall performance (both in terms of the best solution found and in terms of its statiscal measures). The algorithm of the type of ES adopted (due to its simplicity, we decided to call it Simple Evolution Strategy, or SES) is presented in Figure 1.

Compared with respect to other state-of-the-art techniques (due to space limitations we only compare with respect to [2]), our algorithm produced very competitive results (See Table 1). Besides being a very simple approach, it is worth reminding that SES does not require any extra parameters (besides those used with an evolution strategy) and the number of fitness function evaluations performed (350,000) is the same used in [2].

Acknowledgments. The first author acknowledges support from the mexican Consejo Nacional de Ciencia y Tecnología (CONACyT) through a scholarship to pursue graduate studies at CINVESTAV-IPN's Electrical Engineering Department. The second author acknowledges support from (CONACyT) through project number 32999-A.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2723, pp. 640–641, 2003.

Begin
  t=0
  Create a random initial solution x^0
  Evaluate $f(x^0)$
  For t=1 to MAX_GENERATIONS **Do**
    Produce μ mutations of $x^{(t-1)}$ using:
      $x_i^j = x_i^{t-1} + \sigma[t] \cdot N_i(0,1)\ \forall i \in n, j = 1, 2, \ldots, \mu$
    Generate one child x^c by the combination of the μ mutations using
      m=randint$(1, \mu)$
      $x_i^c = x_i^m, \forall i \in n$
    Evaluate f(x^c)
    Apply comparison criteria to select the best individual x^t between $x^{(t-1)}$ and x^c
    $t = t + 1$
    If $(t \mod n = 0)$ **Then**

$$\sigma[t] = \begin{cases} \sigma[t-n]/c & if\ p_s > 1/5 \\ \sigma[t-n] \cdot c & if\ p_s < 1/5 \\ \sigma[t-n] & if\ p_s = 1/5 \end{cases}$$

    End If
  End For
End

Fig. 1. SES algorithm (n is the number of decision variables of the problem)

Table 1. Comparison of results between our approach (SES) and Stochastic Ranking (SR) [2].

		Best Result		Mean Result		Worst Result	
Problem	**Optimal**	**SES**	**SR**	**SES**	**SR**	**SES**	**SR**
g01	−15.000000	−15.000000	−15.000	−14.848614	−15.000	−12.999997	−15.000
g02	0.803619	0.793083	0.803515	0.698932	0.781975	0.576079	0.726288
g03	1.000000	1.000497	1.000	1.000486	1.000	1.000424	1.000
g04	−30665.539000	−30665.539062	−30665.539	−30665.441732	−30665.539	−30663.496094	−30665.539
g06	−6961.814000	−6961.813965	−6961.814	−6961.813965	−6875.940	−6961.813965	−6350.262
g07	24.306000	24.368050	24.307	24.702525	24.374	25.516653	24.642
g08	0.095825	0.095825	0.095825	0.095825	0.095825	0.095825	0.095825
g09	680.630000	680.631653	680.630	680.673645	680.656	680.915100	680.763
g11	0.750000	0.749900	0.750	0.784395	0.750	0.879522	0.750
g12	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

References

1. Kalyanmoy Deb. An Efficient Constraint Handling Method for Genetic Algorithms. *Computer Methods in Applied Mechanics and Engineering*, 186(2/4):311–338, 2000.
2. Thomas P. Runarsson and Xin Yao. Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation*, 4(3):284–294, September 2000.