What Makes a Constrained Problem Difficult to Solve by an Evolutionary Algorithm *

Efrén Mezura-Montes and Carlos A. Coello Coello
CINVESTAV-IPN
Evolutionary Computation Group (EVOCINV)
Electrical Eng. Department, Computer Science Dept.
Av. Instituto Politécnico Nacional No. 2508
Col. San Pedro Zacatenco, México D.F. 07300, MÉXICO
emezura@computacion.cs.cinvestav.mx
ccoello@cs.cinvestav.mx

Abstract

An empirical study about the features that prevent an Evolutionary Algorithm to reach the feasible region or even get the global optimum when it is used to solve global optimization constrained optimization problems is presented. For the experiments we use a Simple Multimembered Evolution Strategy which provides very competitive results in the well known benchmark of 13 test functions. Also, we add 11 new problems which have features we hypothesize that decrease the performance of the algorithm (nonlinear equality constraints and dimensionality). The results seems to agree with our idea and they give some insights to develop more robust EA's for global optimization mainly for real world problems which have the features analyzed in this work.

1 Introduction

Evolutionary algorithms (EAs) have been successfully used to solve different types of optimization problems [1]. However, in their original form, they lack an explicit mechanism to handle the constraints of a problem. This has motivated the development of a considerable number of approaches to incorporate constraints into the fitness function of an EA [11, 2]. Particularly, in this paper we are interested in the general nonlinear programming problem in which we want to:Find $\vec{x}$ which optimizes $f(\vec{x})$ subject to: $g_i(\vec{x}) \leq 0, \;\; i = 1, \ldots, n \; h_j(\vec{x}) = 0, \;\; j = 1, \ldots, p$ where $\vec{x}$ is the vector of solutions $\vec{x} = [x_1, x_2, \ldots, x_r]^T$, n is the number of inequality constraints and p is the number of equality constraints (in both cases, constraints could be linear or nonlinear). This work

*Technical Report EVOCINV-01-2004, Evolutionary Computation Group at CINVESTAV, Sección de Computación, Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, México, February 2004.

Problem	n	Function	ρ	LI	NI	LE	NE
g01	13	quadratic	0.0003%	9	0	0	0
g02	20	nonlinear	99.9973%	1	1	0	0
g03	10	nonlinear	0.0026%	0	0	0	1
g04	5	quadratic	27.0079%	0	6	0	0
g05	4	nonlinear	0.0000%	2	0	0	3
g06	2	nonlinear	0.0057%	0	2	0	0
g07	10	quadratic	0.0000%	3	5	0	0
g08	2	nonlinear	0.8581%	0	2	0	0
g09	7	nonlinear	0.5199%	0	4	0	0
g10	8	linear	0.0020%	3	3	0	0
g11	2	quadratic	0.0973%	0	0	0	1
g12	3	quadratic	4.7697%	0	9^3	0	0
g13	5	nonlinear	0.0000%	0	0	1	2

Table 1: Values of ρ for the first 13 test problems.

covers a first approach to empirically find out what features of a problem, which are not fully covered in the most used benchmark to test constraint handling techniques in EAs, decrease the good performance of one of them. Our study starts by using an EA which actually provides a very competitive performance on the benchmark previously mentioned. 11 new test functions that include characteristics that the current benchmark lack, like nonlinear equality constraints and a high dimensionality is presented. The algorithm is tested on them and the results provided are analyzed and discussed. This paper is organized as follows: In Section 2 we describe previous work of analysis of features of constrained problems. In Section 3 we show our empirical experiments and we detail the 11 new test functions proposed; also, we list the features analyzed in this study. Section 4 provides the results obtained and a discussion of them. We conclude and enumerate our future work in Section 5.

2 Previous Work

The idea of having a set of problems with different characteristics to test evolutionary algorithms to solve constrained problems was initially proposed by Michalewicz & Schoenauer [11]. This set consisted on eleven problems with different features, like type of type of objective function (linear, quadratic, nonlinear), type of constraints (linear, nonlinear, equality or inequality) and dimensionality. Besides, they proposed a metric to approximate the proportion of the feasible region with respect to the whole search space called "ρ". Koziel & Michalewicz [8] added one function to the original benchmark. The main feature of this new function is its disjoint feasible region. Runnarson & Yao proposed another function to the benchmark [12]. This function has three equality constraints (two of them are nonlinear) and the objective function is also nonlinear. These two new functions [8, 12] addressed two features the benchmark lacked (disjoint feasible region and combination of linear and nonlinear equality constraints). The goal of this benchmark is to have a reliable mean to test the quality and robustness of constraint handling techniques in evolutionary algorithms. Michalewicz [10] proposed a Test Case Generator for constrained parameter optimization techniques. This Generator allows to generate test problems by varying several features like: dimen-

Problem	n	Type of function	ρ	LI	NI	LE	NE
g14	10	nonlinear	0.00%	0	0	3	0
g15	3	quadratic	0.00%	0	0	1	1
g16	5	nonlinear	0.0204%	4	34	0	0
g17	6	nonlinear	0.00%	0	0	0	4
g18	9	quadratic	0.00%	0	12	0	0
g19	15	nonlinear	33.4761%	0	5	0	0
g20	24	linear	0.00%	0	6	2	12
g21	7	linear	0.00%	0	1	0	5
g22	22	linear	0.00%	0	1	8	11
g23	9	linear	0.00%	0	2	3	1
g24	2	linear	79.6556%	0	2	0	0

Table 2: Values of ρ for new 11 test problems.

sionality, multimodality, number of constraints, connectedness of the feasible region, size of the feasible region with respect to the whole search space and ruggedness of the objective function. This first version had some problems because the generated functions were very symmetric. Therefore a new version called TCG-2 was proposed [13]. Both versions were used to test a steady-state EA with real representation using a static penalty function to deal with constraints. The results obtained in both TCG' versions share some similarities and also have differences. The similarities are that the high dimensionality and multimodality are parameters that decrease the performance of the EA with the static penalty function. For the first TCG, decreasing the connectivity of the feasible region also affected the good performance of the algorithm. For the TCG-2 the width of peaks had the same undesired effect. Among the parameters with no effect in the performance of the EA for both versions were the size of the feasible region with respect to the whole search space. For the first TCG, the parameters with no effect were the number of constraints and the ruggedness of the objective function. Finally, for the TCG-2 the complexity of the function and the number of active constraints presented little importance in the performance of the EA.

3 Our Empirical Study

The motivation of this work is to determine which characteristics of a global nonlinear optimization constrained problem makes it difficult to solve by an EA. It can help researches to develop even more robust and more applicable to real world problems. We then hypothesized that the current benchmark lack of two main important features: high dimensionality and a considerable (more than three) nonlinear equality constraints. As a second set of features we include the number of nonlinear inequality constraints (more than ten at least), nonlinear objective function and a disjoint feasible region (only one function with this feature is included in the current benchmark [12]. The detail of this benchmark is shown in Table 1 where n is the number of decision variables, LI is the number of linear inequalities, NI the number of nonlinear inequalities, LE is the number of linear equalities and NE is the number of nonlinear equalities. To get a measure of the difficulty of solving each of these problems, a ρ metric (as suggested by Koziel and Michalewicz [8]) was computed using the following expression:

j	1	2	3	4	5
e_j	-15	-27	-36	-18	-12
c_{1j}	30	-20	-10	32	10
c_{2j}	-20	39	-6	-31	32
c_{3j}	-10	-6	10	-6	-10
c_{4j}	32	-31	-6	39	-20
c_{5j}	-10	32	-10	-20	30
d_j	4	8	10	6	2
a_{1j}	-16	2	0	1	0
a_{2j}	0	-2	0	4	2
a_{3j}	-3.5	0	2	0	0
a_{4j}	0	-2	0	-4	-1
a_{5j}	0	-9	-2	1	-2.8
a_{6j}	2	0	-4	0	0
a_{7j}	-1	-1	-1	-1	-1
a_{8j}	-1	-2	-3	-2	-1
a_{9j}	1	2	3	4	5
a_{10j}	1	1	1	1	1

Table 3: Data set for test problem g19

$\rho = |F|/|S|$ where $|F|$ is the number of feasible solutions and $|S|$ is the total number of solutions randomly generated. In this work, $S = 1,000,000$ random solutions.

Unlike Michalewicz TCG approach, we do not want to provide the user the best EA to use depending of his problem. We want to detect features that difficult an EA to find the feasible region of a search space an even more, the feasible global optimum.

Our experiment design was the following: (1) First, we selected test functions (either artificial or from real world problems) that have at least one of the features mentioned before. We selected seven functions from Himmelblau's book [7] (**g14**, **g15**, **g16**, **g17**, **g18**, **g19**, **g20**) two are from heat exchanger network problems detailed in [3] and tested in [4] (**g21**, **g22**). One more was proposed by Xia [15] (**g23**) and the last one was taken from Floudas et al. Handbook [5] (**g24**). Problems selected with high dimensionality were: **g19**, **g20** and **g22**. Test functions with more than three nonlinear equality constraints were: **g17**, **g20**, **g21** and **g22**. For the secondary set of features problems with more than ten nonlinear inequality constrains were problem **g16** and **g18**. Problems with a nonlinear objective function were **g14**, **g16**, **g17** and **g19**. Finally, a test function with a feasible region consisting on two disconnected sub-regions was **g24**. For completeness, we also included two functions that seems to be easy to solve because they have only one nonlinear equality constraint and a quadratic and linear objective function (**g15**and **g23**). The characteristics of each problem is summarized in Table 2

The details of each functions are presented below below below below below below below:

- **g14**:
  Minimize: $f(\vec{x}) = \sum_{i=1}^{10} x_i \left(c_i - \ln \frac{x_i}{\sum_{j=1}^{10} x_j} \right)$
  subject to:
  $h_1(\vec{x}) = x_1 + 2x_2 + +2x_3 + x_6 + x_{10} - 2 = 0$
  $h_2(\vec{x}) = x_4 + 2x_5 + x_6 + x_7 - 1 = 0$
  $h_3(\vec{x}) = x_3 + x_7 + x_8 + 2x_9 + x_{10} - 1 = 0$

where the bounds are $0 \leq x_i \leq 10$ ($i = 1, \ldots, 10$), and $c_1 = -6.089$, $c_2 = -17.164$, $c_3 = -34.054$, $c_4 = -5.914$, $c_5 = -24.721$, $c_6 = -14.986$, $c_7 = -24.1$, $c_8 = -10.708$, $c_9 = -26.662$, $c_{10} = -22.179$,. A feasible local minimum is at $x^* = (0.0350, 0.1142, 0.8306,$
$0.0012, 0.4887, 0.0005, 0.0209, 0.0157, 0.0289, 0.0751)$ where $f(x^*) = -47.751$.

- **g15**:
  Minimize: $f(\vec{x}) = 1000 - x_1^2 - 2x_2^2 - x_3^2 - x_1x_2 - x_1x_3$

  subject to:
  $h_1(\vec{x}) = x_1^2 + x_2^2 + +x_3^2 - 25 = 0$
  $h_2(\vec{x}) = 8x_1 + 14x_2 + 7x_3 - 56 = 0$

  where the bounds are $0 \leq x_i \leq 10$ ($i = 1, \ldots, 10$). A feasible local minimum is at $x^* = (3.512$,0.217,$3.552)$ where $f(x^*) = 961.715$.

- **g16**:
  Maximize: $f(\vec{x}) = 0.0000005843y_{17} - 0.000117y_{14} - 0.1365 - 0.00002358y_{13}$
  $- 0.0000011502y_{16} - 0.0321y_{12} - 0.004324y_5 - 0.0001\frac{c_{15}}{c_{16}} - 37.48\frac{y_2}{c_{12}}$

  subject to:
  $g_1(\vec{x}) = y_4 - \frac{0.28}{0.72}y_5 \geq 0$
  $g_2(\vec{x}) = 1.5x_2 - x_3 \geq 0$
  $g_3(\vec{x}) = 21 - 3496\frac{y_2}{c_{12}} \geq 0$
  $g_4(\vec{x}) = \frac{62,212}{c_{17}} - 110.6 - y_1 \geq 0$
  $g_5(\vec{x}), g_6(\vec{x}) = 213.1 \leq y_1 \leq 405.23$
  $g_7(\vec{x}), g_8(\vec{x}) = 17.505 \leq y_2 \leq 1053.6667$
  $g_9(\vec{x}), g_{10}(\vec{x}) = 11.275 \leq y_3 \leq 35.03$
  $g_{11}(\vec{x}), g_{12}(\vec{x}) = 214.228 \leq y_4 \leq 665.585$
  $g_{13}(\vec{x}), g_{14}(\vec{x}) = 7.458 \leq y_5 \leq 584.463$
  $g_{15}(\vec{x}), g_{16}(\vec{x}) = 0.961 \leq y_6 \leq 265.916$
  $g_{17}(\vec{x}), g_{18}(\vec{x}) = 1.612 \leq y_7 \leq 7.046$
  $g_{19}(\vec{x}), g_{20}(\vec{x}) = 0.146 \leq y_8 \leq 0.222$
  $g_{21}(\vec{x}), g_{22}(\vec{x}) = 107.99 \leq y_9 \leq 273.366$
  $g_{23}(\vec{x}), g_{24}(\vec{x}) = 922.693 \leq y_{10} \leq 1286.105$
  $g_{25}(\vec{x}), g_{26}(\vec{x}) = 926.832 \leq y_{11} \leq 1444.046$
  $g_{27}(\vec{x}), g_{28}(\vec{x}) = 18.766 \leq y_{12} \leq 537.141$
  $g_{29}(\vec{x}), g_{30}(\vec{x}) = 1072.163 \leq y_{13} \leq 3247.039$
  $g_{31}(\vec{x}), g_{32}(\vec{x}) = 8961.448 \leq y_{14} \leq 26844.086$
  $g_{33}(\vec{x}), g_{34}(\vec{x}) = 0.063 \leq y_{15} \leq 0.386$
  $g_{35}(\vec{x}), g_{36}(\vec{x}) = 71084.33 \leq y_{16} \leq 140000$
  $g_{37}(\vec{x}), g_{38}(\vec{x}) = 2802713 \leq y_{17} \leq 12146108$

  where:
  $y_1 = x_2 + x_3 + 41.6$
  $c_1 = 0.024x_4 - 4.62$
  $y_2 = \frac{12.5}{c_1} + 12$

$c_2 = 0.0003535x_1^2 + 0.5311x_1 + 0.08705y_2x_1$
$c_3 = 0.052x_1 + 78 + 0.002377y_2x_1$
$y_3 = \frac{c_2}{c_3}$
$y_4 = 19y_3$
$c_4 = 0.04782(x_1 - y_3) + \frac{0.1956(x_1-y_3)^2}{x_2}$
$c_5 = 100x_2$
$c_6 = x_1 - y_3 - y_4$
$c_7 = 0.950 - \frac{c_4}{c_5}$
$y_5 = c_6c_7$
$y_6 = x_1 - y_5 - y_4 - y_3$
$c_8 = (y_5 + y_4)0.995$
$y_7 = \frac{c_8}{y_1}$
$y_8 = \frac{c_8}{3798}$
$c_9 = y_7 - \frac{0.0663y_7}{y_8} - 0.3153$
$y_9 = \frac{96.82}{c_9} + 0.321y_1$
$y_{10} = 1.29y_5 + 1.258y_4 + 2.29y_3 + 1.71y_6$
$y_{11} = 1.71x_1 - 0.452y_4 + 0.580y_3$
$c_{10} = \frac{12.3}{752.3}$
$c_{11} = (1.75y_2)(0.995x_1)$
$c_{12} = 0.995y_{10} + 1998$
$y_{12} = c_{10}x_1 + \frac{c_{11}}{c_{12}}$
$y_{13} = c_{12} - 1.75y_2$
$y_{14} = 3623 + 64.4x_2 + 58.4x_3 + \frac{146.312}{y_9+x_5}$
$c_{13} = 0.995y_{10} + 60.8x_2 + 48x_4 - 0.1121y_{14} - 5095$
$y_{15} = \frac{y_{13}}{c_{13}}$
$y_{16} = 148000 - 331000y_{15} + 40y_{13} - 61y_{15}y_{13}$
$c_{14} = 2324y_{10} - 28740000y_2$
$y_{17} = 14130000 - 1328y_{10} - 531y_{11} + \frac{c_{14}}{c_{12}}$
$c_{15} = \frac{y_{13}}{y_{15}} - \frac{y_{13}}{0.52}$
$c_{16} = 1.104 - 0.72y_{15}$
$c_{17} = y_9 + x_5$

and where the bounds are $704.4148 \leq x_1 \leq 906.3855$, $68.6 \leq x_2 \leq 288.88$, $0 \leq x_3 \leq 134.75$,
$193 \leq x_4 \leq 287.0966$ and $25 \leq x_5 \leq 84.1988$. A feasible local minimum is at $x^* = (705.06, 68.6,$
$102.9, 282.341, 35.627)$ where $f(x^*) = 1.905$.

- **g17**:
  Minimize: $f(\vec{x}) = f(x_1) + f(x_2)$

  subject to:
  $f_1(x_1) = \begin{cases} 30x_1 & 0 \leq x_1 < 300 \\ 31x_1 & 300 \leq x_1 < 400 \end{cases}$
  $f_2(x_2) = \begin{cases} 28x_2 & 0 \leq x_2 < 100 \\ 29x_2 & 100 \leq x_2 < 200 \\ 30x_2 & 200 \leq x_2 < 1000 \end{cases}$
  $h_1(\vec{x}) = x_1 = 300 - \frac{x_3x_4}{131.078}\cos(1.48477 - x_6) + \frac{0.90798x_3^2}{131.078}\cos(1.47588)$
  $h_2(\vec{x}) = x_2 = -\frac{x_3x_4}{131.078}\cos((1.48477 + x_6) + \frac{0.90798x_4^2}{131.078}\cos(1.47588)$

$h_3(\vec{x}) = x_5 = -\frac{x_3 x_4}{131.078}\sin((1.48477 + x_6) + \frac{0.90798 x_4^2}{131.078}\sin(1.47588)$
$h_4(\vec{x}) = 200 - \frac{x_3 x_4}{131.078}\sin((1.48477 - x_6) + \frac{0.90798 x_3^2}{131.078}\sin(1.47588)$

where the bounds are $0 \leq x_1 \leq 400$, $0 \leq x_2 \leq 1000$, $340 \leq x_3 \leq 420$, $340 \leq x_4 \leq 420$,
$-1000 \leq x_5 \leq 1000$ and $0 \leq x_6 \leq 0.5236$. A feasible local minimum is at $x^* = (107.81, 196.32,$
$373.83, 420, 21.31, 0.153)$ where $f(x^*) = 8927.5888$.

- **g18**:
  Maximize: $f(\vec{x}) = 0.5(x_1x_4 - x_2x_3 + x_3x_9 - x_5x_9 + x_5x_8 - x_6x_7)$

  subject to:
  $g_1(\vec{x}) = 1 - x_3^2 - x_4^2 \geq 0$
  $g_2(\vec{x}) = 1 - x_9^2 \geq 0$
  $g_3(\vec{x}) = 1 - x_5^2 - x_6^2 \geq 0$
  $g_4(\vec{x}) = 1 - x_1^2 - (x_2 - x_9)^2 \geq 0$
  $g_5(\vec{x}) = 1 - (x_1 - x_5)^2 - (x_2 - x_6)^2 \geq 0$
  $g_6(\vec{x}) = 1 - (x_1 - x_7)^2 - (x_2 - x_8)^2 \geq 0$
  $g_7(\vec{x}) = 1 - (x_3 - x_5)^2 - (x_4 - x_6)^2 \geq 0$
  $g_8(\vec{x}) = 1 - (x_3 - x_7)^2 - (x_4 - x_8)^2 \geq 0$
  $g_9(\vec{x}) = 1 - x_7^2 - (x_8 - x_9)^2 \geq 0$
  $g_9(\vec{x}) = x_1x_4 - x_2x_3 \geq 0$
  $g_{10}(\vec{x}) = x_3x_9 \geq 0$
  $g_{11}(\vec{x}) = -x_5x_9 \geq 0$
  $g_{12}(\vec{x}) = x_5x_8 - x_6x_7 \geq 0$

  where the bounds are $-10 \leq x_i \leq 10$ ($i = 1, \ldots, 8$) and $0 \leq x_9 \leq 20$. A feasible local minimum is at $x^* = (0.9971, -0.0758, 0.5530, 0.8331, 0.9981, -0.0623,$
  $0.5642, 0.8256, 0.0000024)$ where $f(x^*) = 0.8660$.

- **g19**:
  Maximize: $f(\vec{x}) = \sum_{i=1}^{10} b_i x_i - \sum_{j=1}^{5}\sum_{i=1}^{5} c_{ij} x_{(10+i)} x_{(10+j)} - 2\sum_{j=1}^{5} d_j x_{(10+j)}^3$

  subject to:
  $g_j(\vec{x}) = 2\sum_{i=1}^{5} c_{ij} x_{(10+i)} + 3d_j x_{(10+j)}^2 + e_j - \sum_{i=1}^{10} a_{ij} x_i \geq 0 \qquad j = 1, \ldots, 5$

  where $\vec{b} = [-40, -2, -.25, -4, -4, -1, -40, -60, 5, 1]$ and the remaining data is on Table 3. The bounds are $0 \leq x_i \leq 10$ ($i = 1, \ldots, 15$). A feasible local minimum is at $x^* = (0, 0, 5.1740, 0, 3.0611, 11.8395, 0, 0,$
  $0.1039, 0, 0.3, 0.3335, 0.4, 0.4283, 0.2240)$ where $f(x^*) = -32.386$.

- **g20**:
  Minimize: $f(\vec{x}) = \sum_{i=1}^{24} a_i x_i$

  subject to:
  $h_i(\vec{x}) = \frac{x_{(i+12)}}{b_{(i+12)}\sum_{j+13}^{24}\frac{x_j}{b_j}} - \frac{c_i x_i}{40 b_i \sum_{j=1}^{12}\frac{x_j}{b_j}} = 0 \qquad i = 1, \ldots, 12$

$h_{13}(\vec{x}) = \sum_{i=1}^{24} x_i - 1 = 0$
$h_{14}(\vec{x}) = \sum_{i=1}^{12} \frac{x_i}{d_i} + f \sum_{i=13}^{24} \frac{x_i}{b_i} - 1.671 = 0$
$g_i(\vec{x}) = \frac{-(x_i + x_{(i+12)})}{\sum_{j=1}^{24} x_j + e_i} \geq 0 \qquad i = 1, 2, 3$
$g_i(\vec{x}) = \frac{-(x_{(i+3)} + x_{(i+15)})}{\sum_{j=1}^{24} x_j + e_i} \geq 0 \qquad i = 4, 5, 6$

where $f = (0.7302)(530)(\frac{14.7}{40})$ and the data set is detailed on Table 4. The bounds are $0 \leq x_i \leq 10$ ($i = 1, \ldots, 24$). A feasible local minimum is at $x^* = (9.53E-7, 0, 4.21eE-3, 1.039E-4, 0, 0, 2.072E-1, 5.979E-1, 1.298E-1, 3.35E-2, 1.711E-2, 8.827E-3, 4.657E-10, 0, 0, 0, 0, 0, 2.868E-4, 1.193E-3, 8.332E-5, 1.239E-4, 2.07E-5, 1.829E-5)$ where $f(x^*) = 0.09670$.

- **g21**:
  Minimize: $f(\vec{x}) = x_1$

  subject to:
  $g_1(\vec{x}) = -x_1 + 35x_2^{0.6} + 35x_3^{0.6} \leq 0$
  $h_1(\vec{x}) = -300x_3 + 7500x_5 - 7500x_6 - 25x_4x_5 + 25x_4x_6 + x_3x_4 = 0$
  $h_2(\vec{x}) = 100x_2 + 155.365x_4 + 2500x_7 - x_2x_4 - 25x_4x_7 - 15536.5 = 0$
  $h_3(\vec{x}) = -x_5 + \ln(-x_4 + 900) = 0$
  $h_4(\vec{x}) = -x_6 + \ln(x_4 + 300) = 0$
  $h_5(\vec{x}) = -x_7 + \ln(-2x_4 + 700) = 0$

  where the bounds are $0 \leq x_1 \leq 1000$, $0 \leq x_2, x_3 \leq 40$, $100 \leq x_4 \leq 300$, $6.3 \leq x_5 \leq 6.7$, $5.9 \leq x_6 \leq 6.4$ and $4.5 \leq x_7 \leq 6.25$. A feasible local minimum is at $x^* = (193.7783493, 0, 17.3272116, 100.0156586,$
  $6.684592154, 5.991503693, 6.214545462)$ where $f(x^*) = 193.7783493$.

- **g22**:
  Minimize: $f(\vec{x}) = x_1$

  subject to:
  $g_1(\vec{x}) = -x_1 + x_2^{0.6} + x_3^{0.6} + x_4^{0.6} \leq 0$
  $h_1(\vec{x}) = x_5 - 100000x_8 + 1 \times 10^7 = 0$
  $h_2(\vec{x}) = x_6 + 100000x_8 - 100000x_9 = 0$
  $h_3(\vec{x}) = x_7 + 100000x_9 - 5 \times 10^7 = 0$
  $h_4(\vec{x}) = x_5 + 100000x_{10} - 3.3 \times 10^7 = 0$
  $h_5(\vec{x}) = x_6 + 100000x_{11} - 4.4 \times 10^7 = 0$
  $h_6(\vec{x}) = x_7 + 100000x_{12} - 6.6 \times 10^7 = 0$
  $h_7(\vec{x}) = x_5 - 120x_2x_{13} = 0$
  $h_8(\vec{x}) = x_6 - 80x_3x_{14} = 0$
  $h_9(\vec{x}) = x_7 - 40x_4x_{15} = 0$
  $h_{10}(\vec{x}) = x_8 - x_{11} + x_{16} = 0$
  $h_{11}(\vec{x}) = x_9 - x_{12} + x_{17} = 0$
  $h_{12}(\vec{x}) = -x_{18} + \ln(x_{10} - 100) = 0$
  $h_{13}(\vec{x}) = -x_{19} + \ln(-x_8 + 300) = 0$
  $h_{14}(\vec{x}) = -x_{20} + \ln(x_{16}) = 0$
  $h_{15}(\vec{x}) = -x_{21} + \ln(-x_9 + 400) = 0$

$h_{16}(\vec{x}) = -x_{22} + \ln(x_{17}) = 0$
$h_{18}(\vec{x}) = -x_8 - x_{10} + x_{13}x_{18} - x_{13}x_{19} + 400 = 0$
$h_{19}(\vec{x}) = x_8 - x_9 - x_{11} + x_{14}x_{20} - x_{14}x_{21} + 400 = 0$
$h_{20}(\vec{x}) = x_9 - x_{12} - 4.60517x_{15} + x_{15}x_{22} + 100 = 0$

where the bounds are $0 \leq x_1 \leq 20000$, $0 \leq x_2, x_3, x_4 \leq 1 \times 10^6$, $0 \leq x_5, x_6, x_7 \leq 4 \times 10^7$, $100 \leq x_8 \leq 299.99$, $100 \leq x_9 \leq 399.99$, $100.01 \leq x_{10} \leq 300$, $100 \leq x_{11} \leq 400$, $100 \leq x_{12} \leq 600$, $0 \leq x_{13}, x_{14}, x_{15} \leq 500$, $0.01 \leq x_{16} \leq 300$, $0.01 \leq x_{17} \leq 400$, $-4.7 \leq x_{18}, x_{19}, x_{20}, x_{21}, x_{22} \leq 6.25$. A feasible local minimum is at $x^* = (12812.5, 722.1602494, 8628.371755, 2193.749851, 9951396.436, 18846563.16, 11202040.4, 199.5139644, 387.979596, 114.8336587, 27.30318607, 127.6585887, 52.020404, 160, 4.871266214, 4.610018769, 3.951636026, 2.486605539, 5.075173815)$ where $f(x^*) = 12812.5$.

- **g23**:
  Minimize: $f(\vec{x}) = -9x_5 - 15x_8 + 6x_1 + 16x_2 + 10(x_6 + x_7)$

  subject to:
  $h_1(\vec{x}) = x_1 + x_2 - x_3 - x_4 = 0$
  $h_2(\vec{x}) = 0.03x_1 + 0.01x_2 - x_9(x_3 + x_4) = 0$
  $h_3(\vec{x}) = x_3 + x_6 - x_5 = 0$
  $h_4(\vec{x}) = x_4 + x_7 - x_8 = 0$
  $g_1(\vec{x}) = x_9x_3 + 0.02x_6 - 0.025x_5 \leq 0$
  $g_2(\vec{x}) = x_9x_4 + 0.02x_7 - 0.015x_8 \leq 0$

  where the bounds are $0 \leq x_1, x_2, x_6 \leq 300$, $0 \leq x_3, x_5, x_7 \leq 100$, $0 \leq x_4, x_8 \leq 200$ and $0.01 \leq x_9 \leq 0.03$.

- **g24**:
  Minimize: $f(\vec{x}) = -x_1 - x_2$

  subject to:
  $g_1(\vec{x}) = -2x_1^4 + 8x_1^3 - 8x_1^2 + x_2 \leq 0$
  $g_2(\vec{x}) = -4x_1^4 + 32x_1^3 - 88x_1^2 + 96x_1 + x_2 - 36 \leq 0$

  where the bounds are $0 \leq x_1 \leq 3$ and $0 \leq x_2 \leq 4$. The feasible global minimum is at $x^* = (2.3295, 3.17846)$ where $f(x^*) = -5.50796$.

(2) We selected an algorithm that provides very good results for the 13 test functions of the current benchmark. It is a Simple Multimembered Evolution Strategy (SMES) proposed by Mezura & Coello [9]. The SMES does not use a penalty function. Instead of using a penalty function, SMES uses simple feasibility rules and an also simple diversity mechanism to maintain infeasible solutions close to the boundaries of the feasible region to bias the search to find the global optimum of a problem. The results were competitive compared to those provided by three state-of-the-art approaches [9].

i	a_i	b_i	c_i	d_i	e_i
1	0.0693	44.094	123.7	31.244	0.1
2	0.0577	58.12	31.7	36.12	0.3
3	0.05	58.12	45.7	34.784	0.4
4	0.2	137.4	14.7	92.7	0.3
5	0.26	120.9	84.7	82.7	0.6
6	0.55	170.9	27.7	91.6	0.3
7	0.06	62.501	49.7	56.708	
8	0.1	84.94	7.1	82.7	
9	0.12	133.425	2.1	80.8	
10	0.18	82.507	17.7	64.517	
11	0.1	46.07	0.85	49.4	
12	0.9	60.097	0.64	49.1	
13	0.0693	44.094			
14	0.0577	58.12			
15	0.05	58.12			
16	0.2	137.4			
17	0.26	120.9			
18	0.55	170.9			
19	0.06	62.501			
20	0.1	84.94			
21	0.12	133.425			
22	0.18	82.507			
23	0.1	46.07			
24	0.09	60.097			

Table 4: Data set for test problem g20

(3) After that, we solved the new set of 11 problems using the SMES and exactly the same parameters previously defined to solve the first 13 test functions.

We performed 30 independent runs for each test function. The learning rates values were calculated using the formulas proposed by Schwefel [14] (where n is the number of decision variables of the problem): $\tau = (\sqrt{2\sqrt{n}})^{-1}$ and $\tau' = (\sqrt{2n})^{-1}$. For the experiments we used the following parameters: $(100+300)$-ES, number of generations = 800, number of objective function evaluations = $240,000$. To deal with equality constraints, a parameterless dynamic mechanism originally proposed in ASCHEA [6] and used in [9] is adopted. The initial ϵ_0 was set to 0.001.

4 Results and Discussion

The statistical results of the SMES for the first set of 13 test functions are summarized in Table 5 and for the new set of 11 functions they are presented in Table 6.

As described in Table 5, for the first 13 test problems the SMES was able to find the global optimum in seven (**g01**, **g03**, **g04**, **g06**, **g08**, **g11** and **g12**) and it found solutions very close to the global optimum in the remaining six (**g02**, **g05**, **g07**, **g09**, **g10**, **g13**). These results show a competitive approach based on the current benchmark.

Now we analyze the results for the new 11 test functions. The SMES had not problem to solve problem **g16** despite its low value of ρ, **g16** involves a considerable number of nonlinear inequalities (34) combined with 4 linear inequality constraints and a nonlinear objective function. The problem has a low dimensionality (5 decision variables). The SMES also solved quite well problems **g14** and **g18**. In both problems the algorithm found the optimum reported in Himmelblau's book. Problem **g14** has a nonlinear objective function and 3 linear equality constraints. Problem **g18** has a quadratic objective function and 12 nonlinear inequality constraints. Both problems have a value of $\rho = 0\%$ and a higher dimensionality (10 and 9 decision variables

Problem	Statistical Results of the SMES for the first 13 Problems					
	Optimal	**Best**	**Mean**	**Median**	**Worst**	**St. Dev.**
g01	-15.000000	-15.000000	-15.000000	-15.000000	-15.000000	0.000000
g02	0.803619	0.803601	0.785238	0.792549	0.751322	0.016757
g03	1.000000	1.001038	1.000989	1.001017	1.000579	0.000209
g04	-30665.539000	-30665.539062	-30665.539062	-30665.539062	-30665.539062	0.000000
g05	5126.498000	5126.599609	5174.492301	5160.197754	5304.166992	50.057854
g06	-6961.814000	-6961.813965	-6961.283984	-6961.813965	-6952.481934	1.851141
g07	24.306000	24.326715	24.474926	24.426246	24.842829	0.132385
g08	0.095825	0.095826	0.095826	0.095826	0.095826	0.000000
g09	680.630000	680.631592	680.643410	680.641571	680.719299	0.015528
g10	7049.25	7051.902832	7253.047005	7253.603027	7638.366211	136.023716
g11	0.750000	0.749090	0.749358	0.749357	0.749830	0.000152
g12	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
g13	0.053950	0.053986	0.166385	0.061873	0.468294	0.176855

Table 5: Statistical results obtained by this new version of the SES for the 13 test functions with 30 independent runs.

Problem	Statistical Results of the SMES for the new 11 Problems					
	Optimal	**Best**	**Mean**	**Median**	**Worst**	**St. Dev.**
g14	-47.656000	-47.534851	-47.367386	-47.385674	-47.053207	0.133386
g15	961.715000	$*961.698120$	963.921753	964.058350	967.787354	1.791314
g16	1.905000	1.905155	1.905155	1.905155	1.905155	0.000000
g17	8927.588800	$*8890.182617$	$*8954.136458$	$*8948.685547$	$*9163.676758$	40.826101
g18	0.866000	0.866002	0.715698	0.673722	0.647570	0.081901
g19	-32.386000	-34.222656	-37.208255	-36.429800	-41.251328	2.102102
g20	0.096700	$*0.211364$	$*0.251130$	$*0.252439$	$*0.304414$	0.023365
g21	193.778349	$*347.980927$	$*678.392445$	$*711.847260$	$*985.782166$	158.493960
g22	12812.500000	$*2340.616699$	$*9438.254972$	$*9968.156250$	$*17671.535156$	4360.887012
g23	0.000000	$*-1470.152588$	$*-363.508270$	$*-333.251541$	$*177.252640$	316.115639
g24	-5.507960	-5.508013	-5.508011	-5.508013	-5.507959	0.000010

Table 6: Statistical results for the SMES with the 11 new test functions "*" means infeasible

respectively).

A value close to the optimum and a low value of the standard deviation were give by the SMES for problem **g19**. The algorithm was less robust to this problem with a nonlinear objective function and 5 nonlinear inequality constraints. It is interesting to note that, despite its ρ value of 33.4761% (which means a large feasible region), a low number of constraints (5) and no equality constraints, the SMES could not find the best solution reported. Nevertheless, this problem has 15 decision variables.

For problem **g15**, the best value found by the SMES is better than the solution reported by Himmelblau, but it is infeasible. Also, in about 35% of the 30 runs, the SMES could not find feasible solutions. This problems has one linear and one nonlinear equality constraints. The objective function is quadratic and the ρ values is 0%. The problem has only 3 decision variables.

Problems **g17**, **g20**, **g21** and **g22** have one common aspect: they have more linear equality constraints than any other problems (4, 12, 5 and 11 respectively) and the SMES could not find feasible solutions in any single run for all of them as well. The dimensionality is different for each of these four problems (6, 24, 7 and 22 respectively). For three problems the objective function is linear (**g20**, **g21** and **g22**). Only **g17** has a nonlinear objective function. All this suggests that the difficult comes from the number of nonlinear equality constraints. It is worth reminding that none of the 13 original test functions have more than 3 of them. Furthermore, none problem with equality constraints have more than 5 decision variables.

The results suggest that the combination of an increasing dimensionality and and high number of nonlinear equality constraints makes a problems more difficult to solve by the SMES. In fact, just one feature is enough to give some problems like in function **g19** which does not have equality constraints, but it has 15 decision variables, the performance of the SMES is degraded. A similar degradation of performance is observed in problem **g17**, with a low dimensionality (6 decision variables) but with 4 nonlinear equality constraints. The performance degrades the most when the problems combine nonlinear equality constraints and a high dimensionality, as in problems **g20 g22**.

It is important to mention that the sum of constraint violation of the final results for problems **g17**, **g20**, **g21** and **g23** is not high. For problem **g23** the best results was far from the feasible region.

There are two test problems that only have one nonlinear equality constraint: **g15** and **g23** with a quadratic and linear objective function respectively. The dimensionality is different (3 and 9 decision variables). Both of them have a quite small feasible region compared with the whole search space. Besides, both have linear equality constraints (1 for **g15** and 3 for **g23** which has 2 nonlinear inequality constraints). For **g23** the dimensionality coupled with the combination of linear and nonlinear equality constraints and the nonlinear inequality constraints should influence the SMES to do not reach the feasible region. For problem **g15** it is important to remark that the best value found is better than the solution reported by Himmelblau, but it is slightly infeasible. Also, in about 35% out of the 30 runs, the SMES could not find feasible solutions. However for the remaining runs, feasible solutions close to the global optimum were found. Therefore, the dimensionality also plays a role of affecting the performance of the algorithm.

Finally, Problem **g24** with a disjoint and quite large feasible region but with a low

dimensionality of 2 represented no problem for the SMES.

To summarize, the overall results suggest that the two main factors that affect the performance of our EA is the dimensionality (like Michalewicz & Schmidt concluded for the static penalty function approach [10, 13]) and the increasing number of nonlinear equality constraints. The factors that do not seem to decrease the performance of our EA were a high number of inequality constraints (even nonlinear), and, quite interesting, the type of objective function. For some problems, despite a linear one, the problems resulted difficult to solve (even reach the feasible region). Finally, Disjoint feasible regions with a considerable size with respect to the search space and a low dimensionality seem to be not difficult to reach for an EA. We need to test other functions with disjoint feasible regions but with a higher dimensionality and with nonlinear constraints to get more information about. This small study is far from being conclusive, but it gives some insights about the factors that difficult an EA to provide good results solving global optimization constrained problems. This information may help to develop even more robust and general EA's.

5 Conclusions and Future Work

A preliminary empirical study about factors that difficult an EA to provide good results when solving global optimization constrained problems was presented. The results show that the number of nonlinear equality constraints as well as a high dimensionality affects the performance of an evolutionary algorithm (in our case a Simple Multimembered Evolution Strategy SMES) when solving global optimization constrained problems. These two features combined can prevent the EA to reach the feasible region of the search space. This study started off from the premise that the SMES provided very competitive results when tested in the well known benchmark of 13 test functions [12]. The features that did not show any mayor impact in the performance were a high number of inequality constraints and the type of objective function. Further and more deep research is necessary to establish with more certainty the features that requires more attention when developing EA's for global optimization problems. Our future work involves a deep analysis of each function and also to analyze other important aspects that are not covered in this papers as nonconvex feasible regions and the size of the search space defined by the bounds of each decision variable.

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. The second author acknowledges support from CONACyT through project number 34201-A.

References

[1] Thomas Bäck. *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New York, 1996.

[2] Carlos A. Coello Coello. Theoretical and Numerical Constraint Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art. *Computer Methods in Applied Mechanics and Engineering*, 191(11-12):1245–1287, January 2002.

[3] T.G.W. Epperly. Global optimization test problems with solutions. Available at http://citeseer.nj.nec.com/147308.html.

[4] T.G.W. Epperly and R.E. Swaney. Branch and Bound for Global NLP: Iterative LP Algorithm & Results. In Ignacio E. Grossman, editor, *Global Optimization in Engineering Design*, Nonconvex Optimization and its Applications, chapter 2, pages 37–73. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1996.

[5] Christodoulos A. Floudas, Panos M. Pardalos, Claire S. Adjiman, William R. Esposito, Zeynep H. Gümüs, Stephen T. Harding, John L. Klepeis, Clifford A. Meyer, and Carl A. Schweiger. *Handbook of Test Problems in Local and Global Optimization*. Nonconvex Optimization and its Applications. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1999.

[6] Sana Ben Hamida and Marc Schoenauer. ASCHEA: New Results Using Adaptive Segregational Constraint Handling. In *Proceedings of the Congress on Evolutionary Computation 2002 (CEC'2002)*, volume 1, pages 884–889, Piscataway, New Jersey, May 2002. IEEE Service Center.

[7] David M. Himmelblau. *Applied Nonlinear Programming*. Mc-Graw-Hill, USA, 1972.

[8] Slawomir Koziel and Zbigniew Michalewicz. Evolutionary Algorithms, Homomorphous Mappings, and Constrained Parameter Optimization. *Evolutionary Computation*, 7(1):19–44, 1999.

[9] Efrén Mezura-Montes and Carlos A. Coello Coello. A Simple Multimembered Evolution Strategy to Solve Constrained Optimization Problems. Technical Report EVOCINV-04-2003, Evolutionary Computation Group at CINVESTAV, Sección de Computación, Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, México D.F., México, 2003. Available in the Constraint Handling Techniques in Evolutionary Algorithms Repository at http://www.cs.cinvestav.mx/˜constraint/.

[10] Zbigniew Michalewicz, Kalyanmoy Deb, Martin Schmidt, and Thomas Stidsen. Test-Case Generator for Nonlinear Continuous Parameter Optimization Techniques. *IEEE Transactions on Evolutionary Computation*, 4(3):197–215, September 2000.

[11] Zbigniew Michalewicz and Marc Schoenauer. Evolutionary Algorithms for Constrained Parameter Optimization Problems. *Evolutionary Computation*, 4(1):1–32, 1996.

[12] Thomas P. Runarsson and Xin Yao. Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation*, 4(3):284–294, September 2000.

[13] Martin Schmidt and Zbigniew Michalewicz. Test-Case Generator TCG-2 for Nonlinear Parameter Optimisation. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo, and H.-P. Schwefel, editors, *Proceedings of 6th Parallel Problem Solving From Nature (PPSN VI)*, pages 539–548, Heidelberg, Germany, September 2000. Paris, France, Springer-Verlag. Lecture Notes in Computer Science Vol. 1917.

[14] Hans Paul Schwefel. *Evolution and Optimal Seeking*. John Wiley & Sons Inc., New York, 1995.

[15] Quanshi Xia. Global optimization test problems. Available at http://www.mat.univie.ac.at/ neum/glopt/xia.txt.