XXXVI Ibero-Latin American Congress on Computational Methods in Engineering

CILAMCE

Rio de Janeiro, 22-25 Nov 2015

HANDLING OPTIMIZATION PROBLEMS WITH CONSTRAINTS OF DIFFERENT MAGNITUDES USING EVOLUTIONARY ALGORITHMS

Rafael de P. Garcia

Beatriz S. L. P. de Lima

Breno P. Jacob

garcia816@gmail.com

bia@coc.ufrj.br

breno@lamcso.coppe.ufrj.br

Federal University of Rio de Janeiro / COPPE / LAMCSO

Pedro Calmon Avenue, 21941-596, Rio de Janeiro - RJ, Brazil

Afonso Celso de Castro Lemonge

afonso.lemonge@ufjf.edu.br

Federal University of Juiz de Fora

José Lourenço Kelmer Street, 36036-900, Juiz de Fora - MG, Brazil

Abstract. *This paper proposes a rank based constraint handling technique called Multiple Constrained Ranking Technique (MCR). This technique is a modification of the ranking method by Ho-Shimizu (HS), where a population of individuals is sorted using three ranks as following: the value of the objective function, the squared sum of the violations values and the number of violated constraints. Our proposal changes the second rank, which is the sum of the violations values by considering many ranks separately, one for each constraint of the problem. The aim of this new constraint handling technique is to overcome the difficulty shown by other techniques when faced with constraint violation values of different magnitudes. We employ the MCR technique with a genetic algorithm using the 2006 CEC benchmark functions and some well-known engineering problems. The results suggest that the MCR technique is competitive, especially when applied to problems defined by constraints of very different magnitudes.*

Keywords: *Constrained Ranking Technique, Evolutionary algorithms, genetic algorithm*

1. INTRODUCTION

Constraints with different magnitudes often describe many engineering optimization problems. As an example, declivity constraints can restrict a submarine pipeline routes optimization problem, which is valued in degrees, and by minimal distance constraints, which establish a minimal distance between oil wells and its violations are calculated in billions of dollars (Lucena et al., 2014). In this case, previously constraint handling techniques (CHT) coupled with a population-based evolutionary algorithm tend to consider more important to find individuals that do not violate constraints with high value, despising constraints with low value. Although normalization is an option this technique does not work to problems handled in this paper, since you do not know the maximum and minimum values of the constraints. So, rank-based constraint handling techniques have been proposed to deal with different orders of magnitude, but the existing ones are still poor.

This paper proposes a rank based constraint handling technique called Multiple Constrained Ranking technique (MCR). The MCR orders in lines the individuals of the population and calculates the fitness of each individual by the sum of its position in each line. There is a different ordering criterion for each line. Those criteria are the objective function value, the number of violated constraints and the violation in each constraint. Thus, in addition to balance the objective function against the violation of each constraint, the MCR handles each constraint with equal importance.

A technique proposed by Ho & Shimizu (2007) has inspired MCR. The difference between those two techniques is that the MCR considers a ranking for each constraint, while the Ho & Shimizu technique does a unique ranking considering as order criteria the sum of violations. Therefore, although Ho & Shimizu is a ranking based CHT, it still does not work well with different magnitude constraints.

This paper compares the MCR with three others ranking based CHTs. All of them were coupled in a genetic algorithm to solve 26 benchmarks optimization problems. The results were confronted using performance profiles, which suggest that the MCR have better performance than the others in both the two metrics of performance utilized and both stop criterion used.

2. CONSTRAINT HANDLING TECHNIQUE

The aim of an optimization problem is to find the best value of x that maximize or minimize a function $f(x)$. Constrained optimization problem (COP) is defined when constraints are added in an optimization problem. Generally, COPs are formulated mathematically as:

$$\text{Min/max } f(x) \tag{1}$$

$$\text{s. a} \quad g_i(x) \leq 0 \tag{2}$$

$$h_j(x) = 0 \tag{3}$$

$$x^L \leq x < x^U \tag{4}$$

Where x is an n-dimensional vector, $f(x)$ is the objective function, $g_i(x)$ $(i = 1, \dots, q)$ are inequality constraints, $h_j(x)$ $(j = q+1, \dots, m)$ are equality constraints, x^L and x^U are lower and upper bounds, respectively. Usually, equality constraints are transformed in inequality ones adding a small tolerance ϵ. In this paper, $\epsilon = 0.0001$.

Bio-inspired evolutionary algorithms either sum a penalty function $P(x)$ to objective function or take some rankings positions to define the fitness function $F(x)$. The fitness function is important to evolution because it qualifies each individual according to its value during the evolution. Thus, individuals with small fitness value have more probability to be selected when compared to an individual with higher value.

2.1 Stochastic Ranking (SR)

An approach that balances the dominance of the objective and penalty function is proposed by Runarsson and Yao (2000). They introduce a probability parameter P_f for comparisons in ranking using only the objective function in the infeasible regions of the search space. In other words, given any pair of two adjacent individuals, the probability of they are compared according to the objective function is 1 if both of them are feasible. Otherwise, it is P_f. The aim motivation to Runarsson and Yao propose the SR comes from the need for balancing objective and penalty function directly.

In review, this approach decides how compare two adjacent individuals to creating the ranking, either by objective function or by sum of violations.

2.2 Global Competitive Ranking (GCR)

The same authors of the SR method proposed another ranking based CHT later [Runarsson and Yao (2002)], the Global Competitive Ranking (GCR). The GCR balances the objective function and the sum of constraints violations according to Eq. (9).

$$F(x) = P_f \frac{rank(f(\mathrm{x})) - 1}{N-1} + \left(1 - P_f\right) \frac{rank\left(\sum_{j=1}^{m} v_j(\mathrm{x})\right) - 1}{N-1} \qquad (9)$$

Where P_f represents a balance parameter, N is the amount of individuals in the population and $rank\big(f(x)\big)$ and $rank\left(\sum_{j=1}^{m} v_j(\mathrm{x})\right)$ represent the current ranking position of the solution x based on its objective value and the sum of its violation (v) in the constraints, respectively. The authors suggest that P_f varies between zero and 0.5.

2.3 Ho-Shimizu Ranking (HSR)

Ho & Shimizu (2007) have developed a ranking based technique such that a population of individuals is sorted using three ranks. The rankings are built according to the value of the objective function, the squared sum of the violations value and the number of violated constraints. The fitness value is evaluated as follows:

$$F(x) = \begin{cases} R_{nv} + R_s, \text{ if there are not feasible individuals} \\ \qquad R_f + R_{nv} + R_s, \text{ otherwise} \end{cases} \qquad (10)$$

Where R_s, R_{nv} and R_f represent the rank position of the individual x based on the squared sum of violations, the number of violated constraints and the objective function value, respectively.

2.4Multiple Constrained Ranking technique (MCR)

This paper proposes a rank based CHT, called Multiple Constrained Ranking Technique (MCR). The MCR is inspired in the HSR and the difference between them is that MCR separates the rank of the violations R_s by considering m ranks, one for each constraint of the problem. Thus, the fitness of each individual is evaluated according to (11).

$$F(x) = \begin{cases} R_{nv} + \sum_{i=1}^{m} R_i \text{ , if there are not feasible individuals} \\ R_f + R_{nv} + \sum_{i=1}^{m} R_i \text{ , otherwise} \end{cases} \quad (11)$$

Where R_i represents the position value of an individual in the constraint i.

We can observe each rank varies between 1 and the number of variable of the population. Therefore if there are not feasible individuals in the population, the maximum fitness value of an individual will be $N(1+m)$, while otherwise will be $N(2+m)$. Thus, the MCR shows a good way to deal with the difficulty shown by some techniques when faced with constraint violation values of different magnitudes.

An optimization problem with three inequality constraints, G1, G2 and G3, is shown in table 1 to illustrate the method. Five individuals (x1, x2, x3, x4 and x5) are used and their objective functions values and the violations (*v*i) are shown .

Table 1. Applying the MCR to a three-inequality constraint optimization problem

Individual	Objective function	*v*1	*v*2	*v*3
x1	5.41	6.24	3.4×10^5	0.002
x2	1.13	7.8	0	0.03
x3	8.7	3.1	4.3×10^5	0.0012
x4	2	0	0	0.04
x5	10	0	0	0

We can note in Table 1 that the individual x5 is feasible since it presents zero value in all constraints. Also, this problem shows violations with high different magnitudes, mainly in constraint G2. Ranks according to the objective values, violations in each constraint, number of constraint violated and the fitness are shown in the Table 2.

Table 2. Ranks table of the example presented in the Table 1

	Rank according to					
Individual	Objective Function	*v*1	*v*2	*v*3	Number of constraints violated	Fitness
x1	3	4	4	3	4	18
x2	1	5	1	4	3	14
x3	4	3	5	2	4	18
x4	2	1	1	5	2	11
x5	5	1	1	1	1	9

In order to use the MCR to evaluate the fitness value of each individual, first we need to note if there is a feasible individual. So, the fitness value of each individual is shown in the last column of Table 2. They are calculated just adding each previously line in the Table 2. The individual x5 have received the lowest fitness value, indicating it will have more chances to be selected.

3. EXPERIMENTS

All CHTs described in section 2 were compared within the same evolutionary algorithm – a elitist genetic algorithm with real coding representation, initial population randomly generated with 50 individuals, selection by ranking, crossover with 0.9 rate under BLX-α method [Eshelman *et al.*, 1993] (α = 0.15) and gaussian mutation equal to 0.03. Two stop criterions were adopted to finish the evolution: when fewer generations are used and when more generations are used.

The performance of the constraint handling techniques were assessed when they were executed 10 times using twenty six functions, 24 taken from a set of benchmark constrained optimization problem known as the G-suite problems (Liang et al., 2006), and two benchmark engineering problems, the pressure vessel problem and the welded beam design problem both described by Mezura et al. (2003).

When fewer generations are used as stop criterion, 500 generations are used to evolution of the G-suite problems, 1600 generations of the pressure vessel problem and 4000 generation of the welded beam design problem. By the other hand, when more generations are used as stop criterion, 1000 generations are used to evolution of the G-suite problems, 3200 generations of the pressure vessel problem and 8000 generations of the welded beam design problem.

Performance profiles suggested by Dolan and Moré (2002) were used to compare the results obtained. The analysis of the performance profiles also were done according the ideas proposed by Barbosa et al. (2010). Two metrics were defined: the best solution found among the runs and the average of the runs divided by feasibility rate. The second one is a strategy suggested by Lian et al. (2010) which in addition to give an idea of average solutions also give us an idea of how good the algorithm is to find feasible solutions.

Tables 4 and 5 show the results obtained. In the first column the problems are named and its best solution ever found is presented. In the third, fourth, fifth and sixth columns the best solution, the average, the worst and the feasibility rate are shown to the MCR, HSR, SR and GCR methods, respectively. Each bold value indicates the best method.

Table 4. Results obtained after fewer generations

		MCR	HSR	SR	GCR
G1	Best	-14.8613	-14.8985	**-14.9156**	-14.8806
	Average	-14.5725	-14.326	-14.7172	**-14.8402**
(-15.000)	Worst	-12.7031	-11.9797	-14.5482	**-14.7929**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G2	Best	**-0.791024**	-0.781999	-0.77788	-0.783348
	Average	**-0.768311**	-0.761847	-0.76519	-0.751835
(-0.80361)	Worst	**-0.746304**	-0.738966	-0.724835	-0.696871
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G3	Best	-0.944933	-0.935853	-0.686617	**-0.968627**
	Average	-0.831732	-0.825203	-0.364498	**-0.861343**
(-1.0005)	Worst	-0.639601	-0.68924	-0.112364	**-0.69925**
	Feasibility	7/10	6/10	**10/10**	**10/10**
G4	Best	-30439	-30564.5	**-30600.6**	-30502.6
	Average	-30286	-30252.3	-30229	**-30294**
(-30665.5)	Worst	**-30098.1**	-29920.6	-29827.6	-29904.1
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G5	Best	-	-	-	-
	Average	-	-	-	-
(5126.5)	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G6	Best	-6661.6	**-6693.31**	-6095.78	-6343.62
	Average	-4997.02	**-5186**	-5050.9	-4892.68
(-6961.81)	Worst	-3936.88	-4172.5	**-4273.56**	-1564.57
	Feasibility	**10/10**	9/10	9/10	**10/10**
G7	Best	29.2078	27.2396	**26.0599**	27.873
	Average	37.9835	33.0876	**31.6805**	35.5993
(24.3062)	Worst	62.562	42.4572	**39.2438**	44.7814
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G8	Best	**-0.095825**	**-0.095825**	**-0.095825**	**-0.095825**
	Average	-0.062484	-0.0885995	-0.0891531	**-0.089154**
(-0.095825)	Worst	**-0.0291438**	-0.0235709	**-0.0291438**	-0.0291431
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G9	Best	**683.049**	685.863	683.967	684.406
	Average	695.131	696.432	**693.451**	698.332
(680.63)	Worst	732.452	**709.801**	706.5	723.302
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G10	Best	**8232.92**	8884.36	8886.93	-
	Average	**10064.3**	10889.6	10836.4	-
(7049.25)	Worst	**11673.7**	12831	13100	-
	Feasibility	**10/10**	**10/10**	9/10	0/10

G11	Best	0.830905	**0.750018**	0.760072	0.751647
	Average	0.899674	0.884713	0.834093	**0.80131**
(0.7499)	Worst	0.94888	0.983733	0.975515	**0.925879**
	Feasibility	**10/10**	**10/10**	**10/10**	9/10
G12	Best	**-1**	**-1**	**-1**	**-1**
	Average	**-1**	-0.999999	-0.999999	**-1**
(-1.000)	Worst	**-1**	-0.999999	-0.999999	**-1**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G13	Best	**0.608958**	0.955396	-	-
	Average	**0.8687254**	1.02945	-	-
(0.05394)	Worst	**0.995264**	1.10351	-	-
	Feasibility	**5/10**	2/10	0/10	0/10
G14	Best	-45.6922	-44.8946	-45.9383	**-46.6637**
	Average	-43.433962	-44.0048	**-44.6389**	-44.5052
(-47.76488)	Worst	-42.1829	-42.9949	**-43.9639**	-42.8258
	Feasibility	**8/10**	**8/10**	7/10	6/10
G15	Best	962.548	962.388	**961.769**	-
	Average	**962.548**	964.446	964.262	-
(961.7150)	Worst	**962.548**	966.505	966.002	-
	Feasibility	1/10	2/10	**3/10**	0/10
G16	Best	-1.6771	-1.57054	-1.69466	**-1.76041**
	Average	-1.369708	-1.25626	-1.24171	**-1.40167**
(-1.905155)	Worst	**-1.03611**	-0.962999	-0.996301	-0.970783
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G17	Best	-	-	-	-
	Average	-	-	-	-
(8853.539)	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G18	Best	**-0.859334**	-0.855946	-0.765799	-0.854763
	Average	**-0.742577**	-0.658592	-0.569836	-0.701567
(-0.86602)	Worst	-0.485178	-0.496972	**-0.499994**	-0.493507
	Feasibility	**10/10**	8/10	6/10	6/10
G19	Best	-	-	-	-
	Average	-	-	-	-
(32.6555)	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G20	Best	-	-	-	-
	Average	-	-	-	-
[unknown]	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G21	Best	**266.527**	-	-	-
	Average	**266.527**	-	-	-
(193.725)	Worst	**266.527**	-	-	-
	Feasibility	**1/10**	0/10	0/10	0/10
G22	Best	-	-	-	-
	Average	-	-	-	-
(236.431)	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10

G23	Best	169.132	**-61.0293**	-	-
	Average	169.132	**-61.0293**	-	-
(-400.055)	Worst	169.132	**-61.0293**	-	-
	Feasibility	**1/10**	**1/10**	0/10	0/10
G24	Best	-5.46687	-5.48688	-5.48879	**-5.4949**
	Average	-5.32674	**-5.36825**	-5.24006	-5.26076
(-5.50801)	Worst	**-5.08931**	-4.83939	-4.9232	-4.89454
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
Pressure vessel	Best	6412.48	**6199.96**	6252	6269.42
	Average	**6696.84**	6853.84	6810.4	6868.71
	Worst	**7055.69**	7545.78	7075.67	7360.22
(6059.71)	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
Welded Beam	Best	2.21011	2.55562	**1.78591**	2.91692
	Average	3.27323	3.54144	**3.07434**	3.53031
	Worst	4.1398	5.87404	**3.97162**	4.61691
(1.29044)	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**

About fewer generations as stop criterion, Figure 1 and Figure 2 are the performance profiles of each handling technique and its areas under its curves, respectively, considering the best solution found among the runs as metric criterion. The Fig. 1 shows that the MCR found the best solution in more problems (around 40%) in relation to the others. The curve corresponding to MCR method is always above the other ones. Consequently it is the method with largest area, as showed in Fig. 2. In this scenario, the performance profiles indicate MCR has better performance than the others. The MCR achieves all its best values with small range of τ (between 1.2 and 1.3). It suggests that even MCR do not find the best solution in a problem, its best solution do not is so far. The high performance of the MCR in face to others techniques is noticeable by the Table 4, since MCR finds the best solution in 9 problems, while the GCR achieves in 7 problems, the SR in 5 and the HSR in 4.

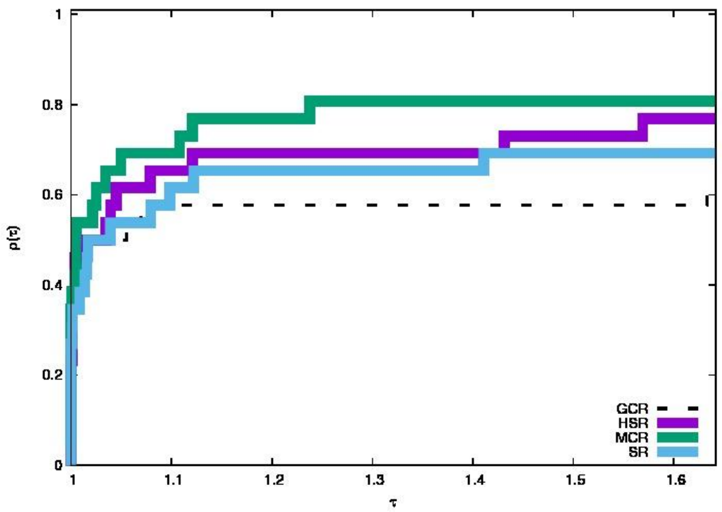

Figure 1. Performance profiles considering the best solution as metric for fewer generations

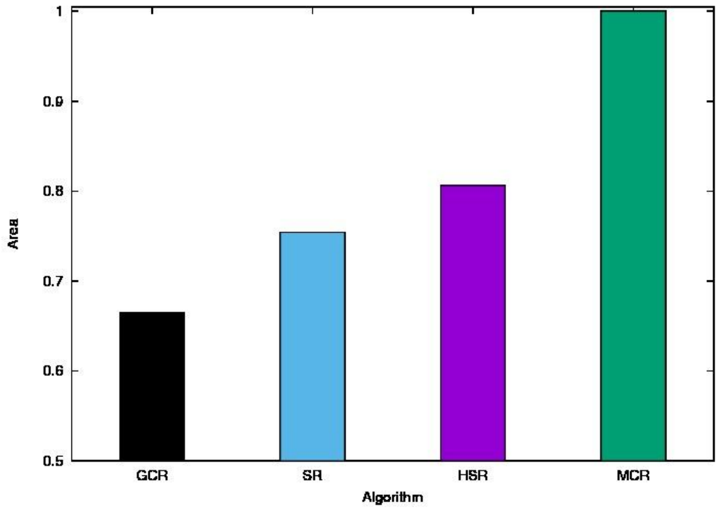

Figure 2. Areas of performance profiles of Fig. 1

The second metric utilized to compare the results is the module of the average of the best solutions divided by the feasibility rate. This metric was adopted because there are problems that are hard to find feasible solutions. In this metric the less feasible solutions are found, the greater the value resulting from dividing between the average and feasibility rate. Thus, it permits us to understand as about the average value as feasibility rate. In this scenario, the MCR continues being the method with better performance according to the performance profiles, as we can see in Fig. 4. However, the difference between its areas and the technique with second best performance, HS, was reduced when we compare the Fig. 2 against Fig. 4.

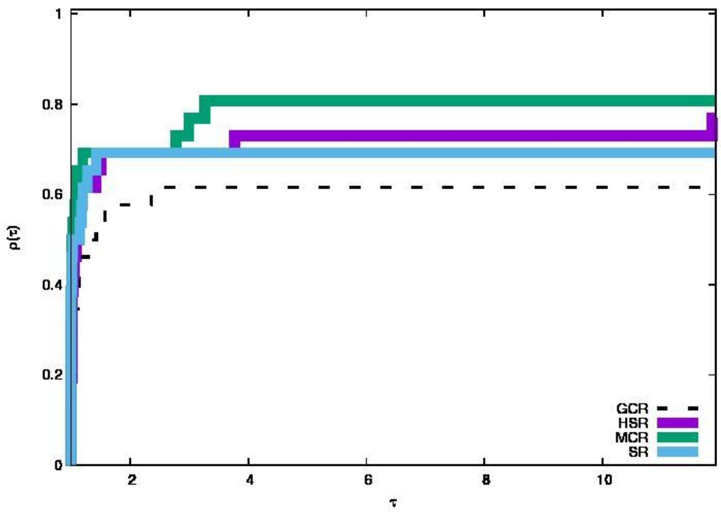

Figure 3. Performance profiles considering the average divided to feasibility rate as metric for fewer generations

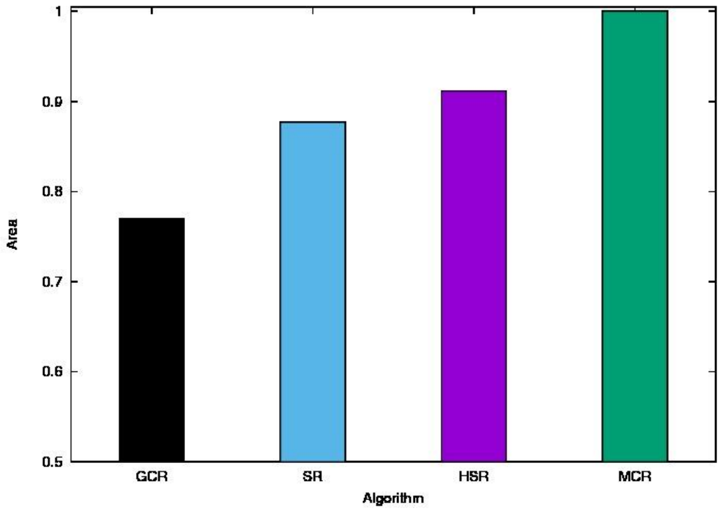

Figure 4. Areas of performance profiles of Fig. 3

The next table, show us the results obtained by the constraint handling techniques when it adopted more generations as stop criterion.

Table 5. Results obtained after more generations

		MCR	HSR	SR	GCR
G1 (-15.000)	Best	**-14.9821**	-14.9737	-14.9451	-14.9751
	Average	-14.3263	-14.2836	-14.7026	**-14.9593**
	Worst	-12.9167	-12.1769	-12.8504	**-14.9392**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G2 (-0.80361)	Best	**-0.799442**	-0.795924	-0.794512	-0.794777
	Average	**-0.784039**	-0.775149	-0.773695	-0.778638
	Worst	**-0.757359**	-0.738877	-0.750139	-0.75227
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G3 (-1.0005)	Best	-0.988247	**-0.992293**	-0.597038	-0.963411
	Average	-0.913219	**-0.9462**	-0.382153	-0.86964
	Worst	-0.811038	**-0.897663**	-0.101408	-0.74034
	Feasibility	8/10	8/10	**10/10**	**10/10**
G4 (-30665.5)	Best	-30472.7	-30466.6	-30359	**-30493.6**
	Average	-30166.7	**-30282.7**	-30203.6	-30208.1
	Worst	-29839.9	**-30055.5**	-29720.7	-29877.4
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G5 (5126.5)	Best	**5203.97**	-	-	-
	Average	**5203.97**	-	-	-
	Worst	**5203.97**	-	-	-
	Feasibility	**1/10**	0/10	0/10	0/10
G6 (-6961.81)	Best	-6075.26	-5568.61	-5904.33	**-6179.4**
	Average	-4667.23	-5188.5	-5266.98	**-5278.02**
	Worst	-1245.46	**-4793.26**	-4440.47	-4522.84
	Feasibility	8/10	9/10	**10/10**	8/10
G7	Best	25.5965	25.6791	**25.5506**	27.5774

(24.3062)	Average	33.6045	**29.6473**	34.4024	31.4062
	Worst	47.1116	38.1064	61.7376	**37.7896**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G8 (-0.095825)	Best	**-0.095825**	**-0.095825**	**-0.095825**	**-0.095825**
	Average	-0.0691101	**-0.0958237**	-0.0891549	-0.0891565
	Worst	-0.0291436	**-0.0958134**	-0.0291438	-0.0291438
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G9 (680.63)	Best	683.177	683.662	**682.386**	683.43
	Average	689.821	**687.149**	693.503	696.513
	Worst	711.497	**699.161**	706.317	769.099
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G10 (7049.25)	Best	**8321.44**	8453.97	8610.24	8909.84
	Average	9658.14	10140.9	10291.6	**8946.19**
	Worst	12148.9	13257.6	12871.1	**8982.54**
	Feasibility	**10/10**	**10/10**	**10/10**	2/10
G11 (0.7499)	Best	0.774367	0.750277	0.75582	**0.75021**
	Average	0.891887	0.866827	0.901552	**0.784394**
	Worst	0.982086	0.971706	0.986105	**0.84427**
	Feasibility	9/10	8/10	**10/10**	8/10
G12 (-1.000)	Best	**-1**	**-1**	**-1**	**-1**
	Average	**-1**	**-1**	**-1**	**-1**
	Worst	**-1**	**-1**	**-1**	**-1**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
G13 (0.05394)	Best	**0.489673**	0.99755	-	-
	Average	2.92607	**0.99755**	-	-
	Worst	9.35983	**0.99755**	-	-
	Feasibility	**4/10**	1/10	0/10	0/10
G14 (-47.76488)	Best	**-46.1729**	-45.481	-44.4148	-46.0368
	Average	**-44.8787**	-44.3639	-43.661	-44.6379
	Worst	-43.0721	-43.1572	-42.8195	**-43.3484**
	Feasibility	8/10	7/10	6/10	**10/10**
G15 (961.7150)	Best	965.28	**961.818**	962.121	-
	Average	965.527	965.774	**962.354**	-
	Worst	965.773	971.422	**962.587**	-
	Feasibility	2/10	**6/10**	2/10	0/10
G16 (-1.905155)	Best	**-1.71671**	-1.65758	-1.70677	-1.6404
	Average	**-1.364729**	-1.32735	-1.26629	-1.35998
	Worst	**-1.10874**	-1.06722	-1.00815	-1.0258
	Feasibility	**10/10**	**10/10**	9/10	**10/10**
G17 (8853.539)	Best	**9177.21**	-	-	-
	Average	**9177.21**	-	-	-
	Worst	**9177.21**	-	-	-
	Feasibility	**1/10**	0/10	0/10	0/10
G18 (-0.86602)	Best	-0.787001	-0.839002	-0.825313	**-0.845778**
	Average	-0.621812	**-0.645104**	-0.529894	-0.557229
	Worst	-0.466952	-0.490838	-0.239467	**-0.498843**
	Feasibility	**10/10**	**10/10**	9/10	6/10
G19	Best	-	**253.235**	-	-

(32.6555)	Average	-	**253.235**	-	-
	Worst	-	**253.235**	-	-
	Feasibility	0/10	**1/10**	0/10	0/10
G20 [Unknown]	Best	-	-	-	-
	Average	-	-	-	-
	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G21 (193.725)	Best	**289.894**	-	-	-
	Average	**302.654**	-	-	-
	Worst	**315.414**	-	-	-
	Feasibility	**2/10**	0/10	0/10	0/10
G22 (236.431)	Best	-	-	-	-
	Average	-	-	-	-
	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G23 (-400.055)	Best	-	-	-	-
	Average	-	-	-	-
	Worst	-	-	-	-
	Feasibility	0/10	0/10	0/10	0/10
G24 (-5.50801)	Best	-5.36907	-5.43366	-5.48875	**-5.50482**
	Average	-5.18345	-5.28997	-5.29807	**-5.40663**
	Worst	-4.96912	-4.96312	-5.01449	**-5.28795**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
Pressure (6059.71)	Best	**6075.87**	6146.73	6418.15	6393.87
	Average	6894.33	7028.64	6985.69	**6756.12**
	Worst	7547.16	7553.79	7544.59	**7332.83**
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**
Welded Beam (1.29044)	Best	**1.82630624**	1.97089	2.58335	1.8648
	Average	3.15213327	3.14909	2.86345	**3.08911**
	Worst	3.91508872	4.09286	**3.11853**	4.38199
	Feasibility	**10/10**	**10/10**	**10/10**	**10/10**

About more generations as stop criterion, the Figure 5 and Figure 6 represent the performance profiles of each handling technique and its areas under its curves, respectively, considering the best solution found among the runs as metric criterion. As we can see in Fig. 5, increasing the number of generations produced a similar performance between the methods when Figures 1 and 5 and Figures 2 and 6 are compared. In other words, even increasing the number of generations the MCR continues to finding and improving its best ever solution.

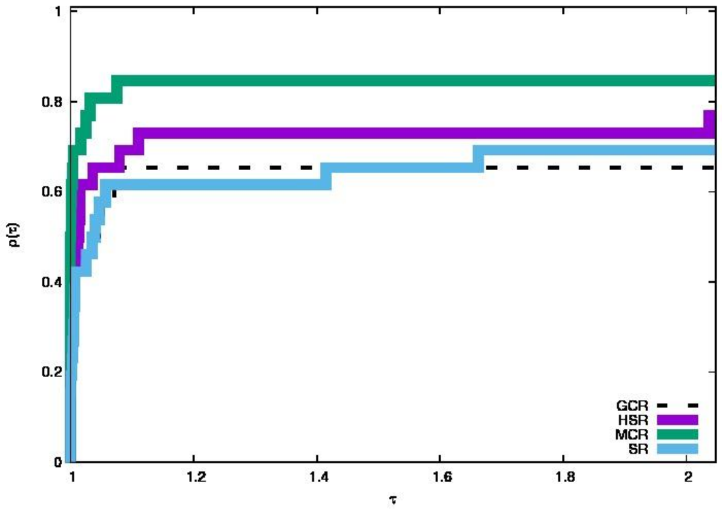

Figure 5. Performance profiles considering the best solution as metric for more generations

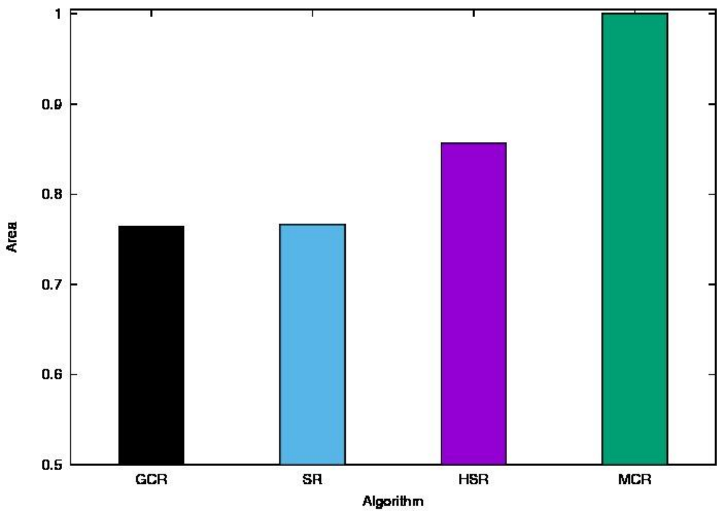

Figure 6. Areas of performance profiles of Fig. 5

The second metric utilized to compare the results is the module of the average of the best solutions divided to the feasibility rate. The performance profiles to this metric and its under areas are shown in the Figure 7 and Figure 8, respectively. The performance curves of MCR and HSR are closer when we compare the two performance metrics, as we can observed in Figures 5 and 7. It indicates us that HSR has a good average performance or a good rate feasibility performance in spite of MCR continues being the technique with best performance in this sense.

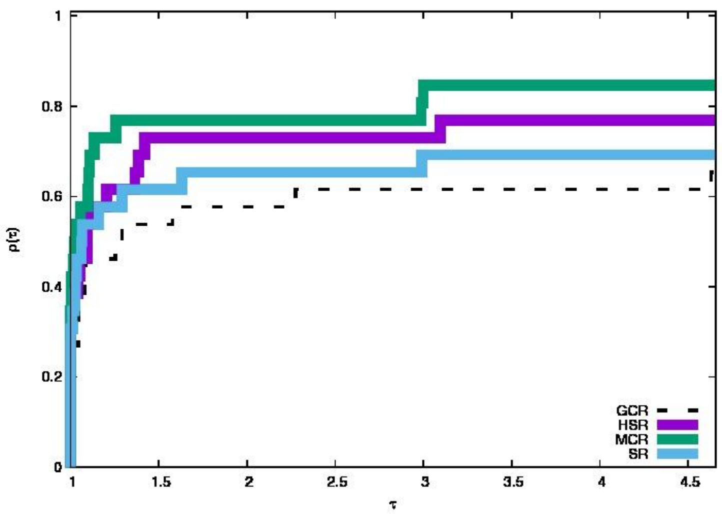

Figure 7. Performance profiles considering as metric the average divided to feasibility rate for more generations

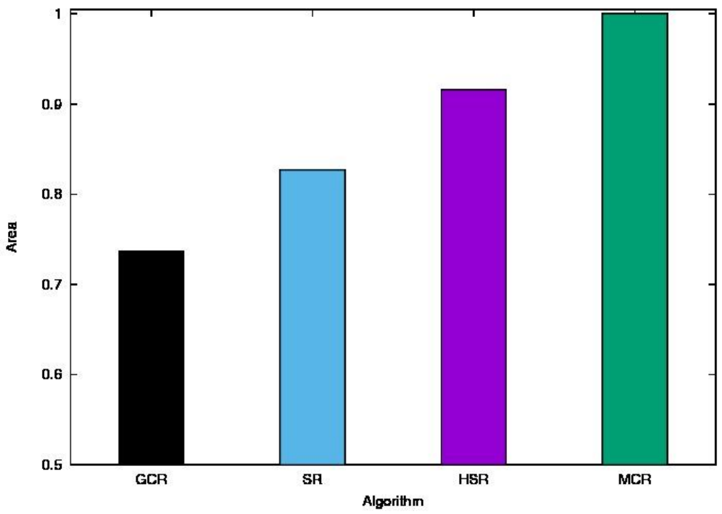

Figure 8. Areas of performance profiles of Fig. 7

The problems G10, pressure vessel and welded beam were analyzed carefully because they are characterized by constraints with different magnitudes. For example, the problem G10 has three constraints with order of magnitude of ten to the seventh power and other three with order of magnitude of ten. The pressure vessel and the welded beam follow the same scheme. They have constraints varying between ten to the minus one to ten to the eighth power. We can see in Table 4 and Table 5 that MCR is the constraint handling technique that finds the best results to G10 in both stop criterion and to Pressure Problem and Welded Beam Problem when more generations are used. By the other hand, when less generations are used the HSR find the best solution to the Pressure Problem while the SR find the best solution to the Welded Beam Problem.

4. CONCLUSIONS

We compare four based ranking constraint handling technique coupled in a genetic algorithm. One of those techniques, MCR, was proposed by this work in order to deal with optimization problems with different-magnitude constraints. All techniques were submitted to 26 optimization problem and the results were compared using performance profiles. Two stop criterions were considered.

The results suggest that MCR is the technique with best performance in relation to the others when two performances metric are adopted – the best solution and the module of the average divided by feasibility rate. The problems G10, Pressure Vessel and Welded Beam, which have constraints with different magnitudes, confirm that the utilization of the MCR can improve the results in face of constraints with different magnitudes.

ACKNOWLEDGEMENTS

The authors thank the Brazilian agencies CNPq and FAPEMIG (grant no. TEC PPM 528/11, TEC PPM 388/14).

5. REFERENCES

Barbosa, H. J. C, Bernardino, H. S., & Barreto, A. M. S., 2010. Using performance profiles to analyze the results of the 2006 CEC constrained optimization competition. *Evolutionary Computation (CEC), 2010 IEEE Congress on*. IEEE.

Dolan, E. D., & Moré, J. J., 2002. Benchmarking optimization software with performance profiles. *Mathematical programming* 91.2: 201-213.

Eshelman, L. J., Schaffer J. D., & Whitley L. D., 1993. Foundations of genetic algorithms 2. *Real Coded Genetic Algorithms and Interval Schemata*: 5-17.

Ho, P.Y., & Shimizu, K., 2007. Evolutionary constrained optimization using an addition of ranking method and a percentage-based tolerance value adjustment scheme. *Information Sciences*, vol. 177, n. 14, pp. 2985-3004.

Liang, J. J., et al., 2006. Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization. *Journal of Applied Mechanics* 41.

Lucena, R. R., et al., 2014. Optimal design of submarine pipeline routes by genetic algorithm with different constraint handling techniques. *Advances in Engineering Software*, *76*, 110-124.

Mezura-Montes, E., Coello, C. A. C., & Landa-Becerra, R., 2003. Engineering optimization using simple evolutionary algorithm. *Tools with Artificial Intelligence. Proceedings. 15th IEEE International Conference on*. IEEE.

Runnarson, T., & Yao, X., 2002. Constrained evolutionary optimization–the penalty function approach. *Evolutionary Optimization*: 87-113.