MULTI-OBJECTIVE GENERALIZED EXTREMAL OPTIMIZATION WITH REAL CODIFICATION AND ITS APPLICATION IN SATELLITE ATTITUDE CONTROL

Igor Mainenti, rogimainenti@gmail.com
Luiz Carlos Gadelha DeSouza, gadelha@dem.inpe.br
Fabiano L. De Sousa, fabiano@dem.inpe.br
Ana Paula Curti Cuco, apcuco@gmail.com
Nacional Institute for Space Research – INPE. Av. dos Astronautas, 1758 – 12201-970 São José dos Campos – SP – Brazil.

Abstract. *In this work a new multi-objective optimization algorithm is presented. This new evolutionary algorithm, called M-GEO$_{real}$, is based on the M-GEO algorithm. The motivation to develop this algorithm is to increase the efficiency and efficacy of the previous one. As a brand new algorithm, it is necessary to perform some tests with well-known test functions. In this test, the performance of the M-GEO$_{real}$ will be compared with some commonly used optimization algorithm. Besides, in order to test the M-GEO$_{real}$ algorithm in a real problem, it is used to determining the gains of a non-linear control law type, which controls the attitude of a rigid-flexible satellite. A multi-objective approach is employed in order to minimize, simultaneously, the time and the energy during the satellite attitude control. The use of a multi-objective approach allows that a set of optimized trade-off solutions (non-dominated solutions) be determined and become available to the designer for posterior choice of an individual solution to be implemented. The non-dominated solutions set in the design space (Pareto optimal set) and in the objective functions space (Pareto front) were obtained through this new multi-objective version of GEO.*

Keywords*: rigid-flexible satellite, non-linear attitude control law, evolutionary algorithm, multi-objective optimization, generalized extremal optimization.*

1. NOMENCLATURE

EA – Evolutionary Algorithm;
GEO – Generalized Extremal Opmization;
M-GEO – Multi-objective GEO;
GEO$_{real}$ – GEO with real codification;
M-GEO$_{real}$ – Multi-objective GEO$_{real}$;
L – The length of the M-GEO string of bits;
N – Number of design variables;
m – Number of objective functions;
$F_m(\boldsymbol{x})$ – The value of the m-th objective function;
$\boldsymbol{x}$ – Vector of the design variables;
A_i – The adaptability of the i-th bit;
k_i – The rank of the i-th bit;
τ – Free parameter of M-GEO and M-GEO$_{real}$;
rt – The number of restarts of M-GEO and M-GEO$_{real}$ algorithms;
w_m – The weight related to $F_m(\boldsymbol{x})$;
x_i – The i-th design variable;
x'_{ij} – The j-th variation of x_i;
A_j – The adaptability of x'_{ij};
k_j – The rank of x'_{ij};
P – Number of different values of x'_{ij} per iterations of M-GEO$_{real}$;
$N_j(0,\sigma_j)$ – A random number with Gaussian distribution;
σ_j – The standard deviation;
NSGAII – Non-Dominated Sorting Genetic Algorithm II;
p – The modal state of the satellite beam;
θ – The rotation angle of the satellite central body;
ω – The beam first mode of vibration;
ξ – The torque that controls the satellite attitude;
K_1, K_2 and K_3 – The gains of the control law;
A_h – Cross section of the satellite beam;
ρ – Alumininum density;
l – Beam length;
E – Young's modulus;
$a_1 l$ – Eigen value associated to the beam first mode of vibration;
I_h – Moment of inertia of the beam cross section about the neutral axis;
I_o – Satellite's main body moment of inertia about its center of mass;
ω – Beam first mode of vibration;
R_1 – Half of the satellite central body edge.

2. INTRODUCTION

Evolutionary algorithms (EAs) are stochastic methods of optimization that is based on nature process. It is widely used to tackle engineering and scientific optimization problem (Davis *et al.*, 1999; Bäck and Schwefel, 1993). This kind stochastic method employs a population of candidate solutions that is "evolved" during the search as better individuals (new solutions) are generated from previous ones in the sense that they are closer to the global minimum. The main advantage of the evolutionary algorithms is the capacity to avoid local optimal solutions, allowing searching for the global optimum. In fact, evolutionary algorithms are very robust methods. They are capable to tackle problems with non-linearities in the objective functions. They can easily deal with constrains and their non-linearities, and also deal well with problems that have differently kinds of design variable.

Recently, a new EA, called M-GEO, capable to tackle multi-objective problem was proposed (Galski, 2006; Galski *et al*., 2005). This algorithm is the first multi-objective version of the Generalized Extremal Optimation (GEO) algorithm (De Sousa *et al*., 2003). It was developed to obtain the Pareto Front maintaining the main characteristics of GEO algorithm. That is, the easy implementation to optimization problems; derivatives are not used during the search; can be applied to unconstrained or constrained problems and non-convex or even disjoint design spaces, in the presence of any combination of continuous, discrete or integer design variables.

Although, GEO and M-GEO have showed good performance to tackle optimization problems, they codify the variables with strings of bits. This characteristic imposes a precision to the variables and this can lead to a sub-optimal solution if the bit coding does not capture the variable optimal values. In order to avoid this limitation a new version of GEO was developed dealing directly with real variables (Mainenti-Lopes, 2008; Mainenti-Lopes *et al*., 2008). Called GEO_{real}, this new version showed better performance than previously versions of GEO by tackle test functions. Meanwhile, the GEO_{real} cannot tackle multi-objective problems.

In this context, a new version of M-GEO algorithm is present in this work. Called M-GEO_{real}, this new algorithm is based on GEO_{real} and it was develop to tackle multi-objective problems in the same way of M-GEO, but using real variables. As a preliminary test, its efficiency and efficacy will be test by tackling two well know test functions for multi-objective algorithm, ZDT1 and TNK. Besides, a real problem will be used to test this algorithm. Considering that, satellite attitude control is one of the most complex subsystems of the satellite; the chosen problem was to optimize the gains of the satellite's control law. In most satellites, this subsystem makes use of fuel. Therefore, it is very important to minimize the energy spent by this subsystem. Knowing that the time to control the satellite needs to be minimized too, this multi-objective problem becomes a good test for M-GEO_{real} to prove its efficiency and efficacy in real problems.

The paper is divided in several sections. In section 2, the M-GEO_{real} algorithm is described. Following by the test functions characteristics and the M-GEO_{real} performance in section 3. In section 4, it is presented the satellite attitude control problem and, in section 5, is described how the M-GEO_{real} tackles this problem including the results. Finally, the section 6 concludes this work.

3. M-GEO AND M-GEO_{real} ALGORITHMS

Multi-objective optimizations problems consist in optimize simultaneously two or more conflicting objectives. Because the objectives are conflicting, it is impossible to obtain one solution that optimizes all objectives. Therefore, one can obtain a set of solutions that, for each solution, it is impossible to optimize one objective without losing optimality in the others. This set of solution in the design space is called Pareto Set and in the objective space is called Pareto Front. The main goal of an algorithm capable to tackle multi-objective problems is to obtain Pareto Set and Pareto Front.

M-GEO_{real} was based on the second algorithm presented by Mainenti-Leal *et al.* (2008), called GEO_{real2}. The main difference between GEO_{real2} and M-GEO_{real} is how each one deals with the best solution. As a mono-objective algorithm, GEO_{real2} stores the best solution along the run and returns only one solution. While M-GEO_{real} stores the non-dominated solutions along the run and for each new solution a test is made to determine which solution will be kept and which will be discarded. The following steps give this test that will be called Pareto Front Test in this work:

(i) test if the new solution is dominated by any solution in the stored Pareto Front. That is, if any solution in Pareto Front is at least equal in all objective functions except for one that is better than the new solution. If the new solution is dominated, keep the Pareto Front and go to the step (iii). Otherwise, include the new solution and go to the next step;

(ii) determine all solutions that the new solution dominate and discard them from the Pareto Front;

(iii) finish the Pareto Front Test.

M-GEO_{real} was developed to recover the Pareto Front and the Pareto Set maintaining the main characteristic of GEO algorithm. This new multi-objective version will compared to previously one, called M-GEO. The M-GEO algorithm can be described using the following steps:

(i) initialize randomly a string of L bits, which codifies N design variables. Calculate the value of all functions $F_m(\boldsymbol{x})$ with this set of variables, where m is the number of objective functions. Store $F_m(\boldsymbol{x})$ in Pareto Front and $\boldsymbol{x}$ in Pareto Set;

(ii) set the value of the index i to 1 and chose a integer number j between 1 and m;

(iii) flip the i-th bit, calculate all functions $F_m(\boldsymbol{x})$ using that new set of bits and run the Pareto Front Test. Assign the adaptability A_i to that bit as $F_j(\boldsymbol{x})$. Return the original value of the i-th bit, increment the value of i and repeat this step until $i > L$.

(iv) assign a rank k_i to each bit according to the A_i value, $k_i = 1$ to the best value and $k_i = L$ to the worst value;

(v) chose with uniform probability one of the bits, accept this choice with probability equal to $k_i^{-\tau}$, where τ is a free parameter. If the choice was accepted flip the chosen bit and go to step (vi). In the other wise, repeat this step;

(vi) test a stopping criterion. If it is accepted go to step (viii). Otherwise, test a population restart criterion. If it is accepted go to step (vii). Otherwise, go back to step (ii);

(vii) initialize randomly a string of L bits that codifies N design variables, calculate all objective function value $F_m(\boldsymbol{x})$ with this set of variables and run the Pareto Front Test. Go back to step (ii);

(viii) return the Pareto Front and the Pareto Set.

The flowchart of M-GEO is presented in Fig. 1.

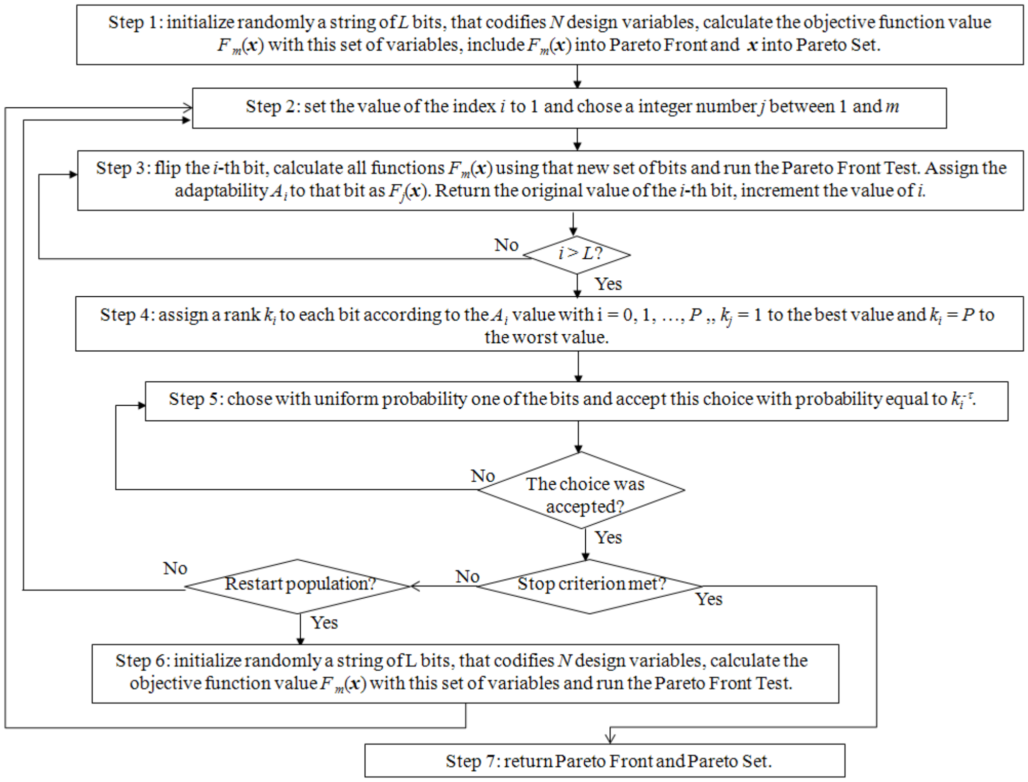

Figure 1. M-GEO flowchart.

The M-GEO$_{real}$ can be described by the following steps:

(i) initialize randomly a string of N design variables, calculate the value of all functions $F_m(\boldsymbol{x})$ with this set of variables, where m is the number of objective functions. Store $F_m(\boldsymbol{x})$ in Pareto Front and $\boldsymbol{x}$ in Pareto Set;

(ii) set the value of the index i to 1;

(iii) set the value of the index j to 1;

(iv) generate randomly m weight w_m between 0 and 1, each one associate to each objective function and calculate the adaptability of $\boldsymbol{x}$ given by

$$A_j = \frac{\sum_{k=1}^{m} w_k F_k(x)}{\sum_{k=1}^{m} w_k}. \tag{1}$$

with $j = 0$. A_0 represents the probability to maintain the variable value unchanged. That is, $x'_{i0} = x_i$. Therefore, there is a chance to keep the variable value if it is a good value. This is one of the differences between this version and the mono-objective version;

(v) Change the value of the variable x_i using a equation given by

$$x'_{ij} = x_i + N_j(0, \sigma_j)x_i \tag{2}$$

calculate $F_m(\boldsymbol{x})$ using the value of x'_{ij} instead of x_i and run the Pareto Front Test. Calculate the adaptability of x'_{ij} using the Eq. (1), where $N_j(0,\sigma_j)$ is a random number with Gaussian distribution and σ_j is the standard deviation;

(vi) return the value x_i to the vector $\boldsymbol{x}$, increment the value of j, return to step (iv). Repeat this sequence until $j > P$;

(vii) assign a rank k_j to each x'_{ij} according to the A_j value with j = 0, 1, …, P , where k_j = 1 to the best value and k_j = P + 1 to the worst value;

(viii) chose with uniform probability one of x'_{ij} (including the solution unchanged represented by x'_{i0}), accept this choice with probability equal to $k_j^{-\tau}$. If the choice was accepted store the choose x'_{ij}, but do not change the value of x_i yet, and continue to next step. In the other wise, go back to step (viii);

(ix) increment the index i and go back to step (iii). Repeat this process until $i > N$.

(x) change each element x_i of the vector $\boldsymbol{x}$ according to the value x'_{ij} chosen in step (vii). Calculate $F_m(\boldsymbol{x})$ using the new vector $\boldsymbol{x}$ and run the Pareto Front Test. Test a stopping criterion. If it is accepted go to step (xii). Otherwise, test a population restart criterion. If it is accepted go to step (xi). Otherwise, go back to step (ii);

(xi) initialize randomly a string of N design variables, calculate all objective function value $F_m(\boldsymbol{x})$ with this set of variables and run the Pareto Front Test. Go back to step (ii);

(xii) return the Pareto Front and the Pareto Set.

The flowchart of M-GEO$_{real}$ is presented in Fig. 2.

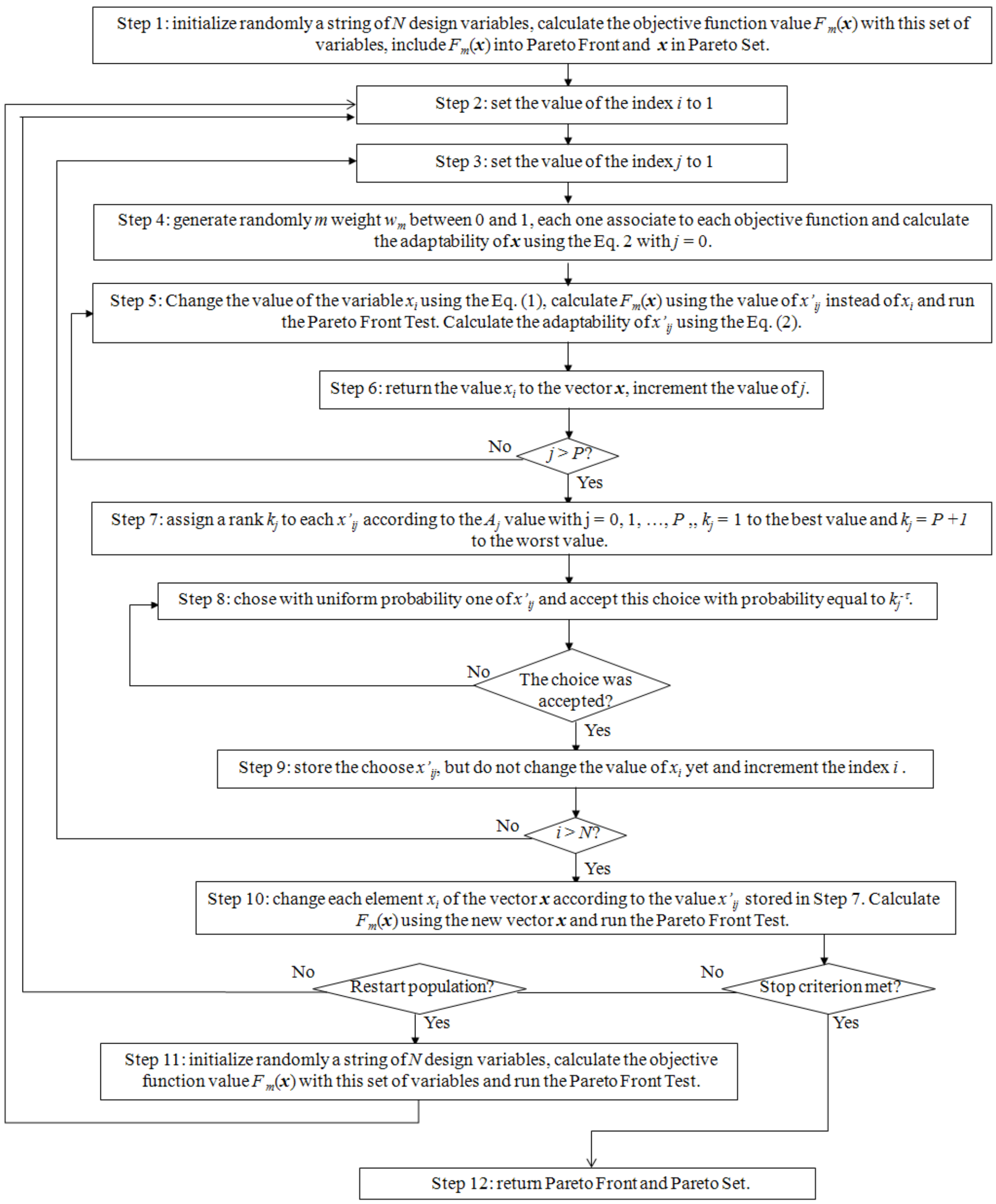

Figure 2. M-GEO$_{real}$ flowchart.

The population restart test is made to increase the algorithm capacity to recover all Pareto Front. In this work, the criterion to restart the population was given by the free parameter *rt* that represents the number of restarts along the search.

A disadvantage of M-GEO$_{real}$ compared with M-GEO is the increase of free parameters. In M-GEO algorithm, there is only two parameter; the value of τ and *rt*. In the M-GEO$_{real}$, there is P+3 new free parameters (P standard deviations, the P, τ and *rt* values). The intention of using several values of standard deviation for a same variable is to capacitate the algorithm to search in a greater range of value in a single iteration. Therefore, it is interesting to select high and low values of σ_j. To reduce the amount of free parameters the following rule it was adopted.

$$\sigma_{i+1} = \frac{\sigma_i}{s.i} \tag{3}$$

where i = 1, 2, …P and *s* is a arbitrary number greater than one. In this work, it was chosen s = 2. In that way, it is enough define σ_1 and all the other values of σ_j will be automatically defined. Therefore, there are as many high values as low values of σ. Now, it is needed to define four free parameters σ_1, P, τ and *rt*.

However, M-GEO$_{real}$ algorithm can change all variables per iteration. While M-GEO change only one bit, that is, it can change just one variable per iteration. Besides, M-GEO chooses one function $F_m(x)$ to calculate the adaptability per iteration. This procedure leads the algorithm to find solution at the edge of the Pareto Front. While, M-GEO$_{real}$ uses a weight sum of the functions.

4. TEST FUNCTIONS AND M-GEO$_{real}$ PERFORMANCE

The performance of M-GEO$_{real}$ was compared to the performance obtained by the multi-objective algorithms M-GEO and NSGAII (Deb *et al*., 2000) using 2 test functions: ZDT1 and TNK. NSGAII is a multi-objective evolutionary algorithm widely used in engineering and scientific problems. One used the modeFRONTIER software to find the Pareto Front of the test functions using the NSGAII algorithm. It was used default parameters of NSGAII given by that software. The parameters of M-GEO used was τ = 1.5 rt = 50 for ZDT1 and τ = 4.25 and rt = 50 for TNK.

The main features of the test functions are presented in Tab. 1.

Table 1. Main features of the test functions.

Name	Functions	Constraints
ZDT1	$F_1 = x_1$ $F_2 = x_2$	$C_1(\vec{x}) = x_1^2 + x_2^2 - 1 - 0.1\cos\left(16\arctan\frac{x_1}{x_2}\right) \geq 0$ $C_2(\vec{x}) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$ $0 \leq x_1 \leq \pi$ $0 \leq x_2 \leq \pi$
TNK	$F_1 = x_1$ $F_2 = 1 - \sqrt{\frac{f_1}{g}}$, with $g = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i$	$0 \leq x_i \leq 1$ with *i* = 1, 2, 3, ..., 30.

The reason of this choice was that: ZDT1 presents a great number of variables, while it is easy to find an analytical result; and TNK presents a non-convex region and discontinuity in its Pareto Front. The Pareto Fronts of these test functions are presented in Fig 3. The right graph is the Pareto Front for ZDT1 and the left is for TNK.

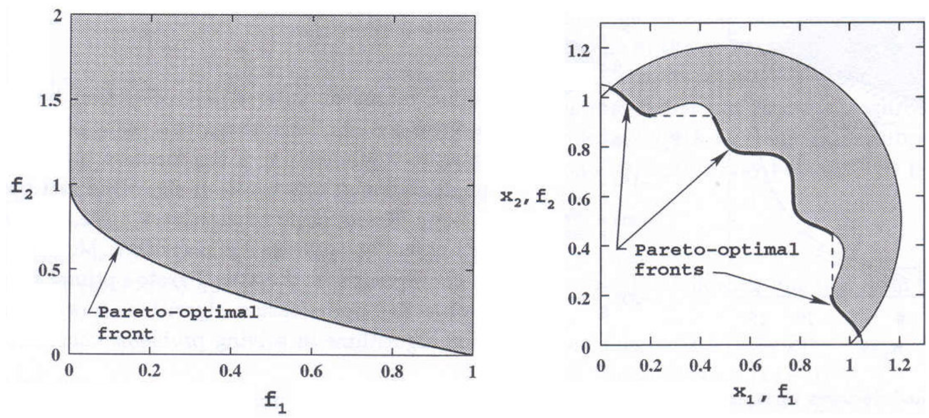

Figure 3. In the right graph is the Pareto Front for ZDT1 and in the left is for TNK.

One used 10^5 objective function evaluations. For test function ZDT1, it was tested the τ values between 0.5 and 10 with variation of 0.5 among one test and another. A new test was accomplished using values between the best values of τ found in the previous test (0.5 and 2.5). Meanwhile, the variation between one test and another this time was 0.1. The chosen value was 1.8. After that, it was tested the following values: $P = 3$; 4 and 5; $\sigma_l = 1$, 2 and 3; $rt = 10$, 20, 30 and 50.

The selected values were: $P = 3$; $\sigma_l = 2$; $rt = 20$.

For test function TNK, the tests to set the best parameters follow the same proceeds of ZDT1, except for the second test of τ value with variation of 0.1. The selected values were: $\tau = 9.5$; $P = 3$; $\sigma_l = 1$; $rt = 20$.

The side constraints were treated rejecting all solutions that did not respect them. The constraints C_1 and C_2 was treated adding a penalty to the objective function given by the following equations

$$F_p = F_m(\vec{x}) + p_1(1 - C_1)^2 \quad (4)$$

if $C_1 < 0$ and

$$F_p = F_m(\vec{x}) + p_2(C_2 + 0.5)^2 \quad (5)$$

if $C_2 > 0.5$.

The performance of M-GEO$_{real}$ is presented in Fig. 4 for ZDT1 and in Fig. 5 for TNK.

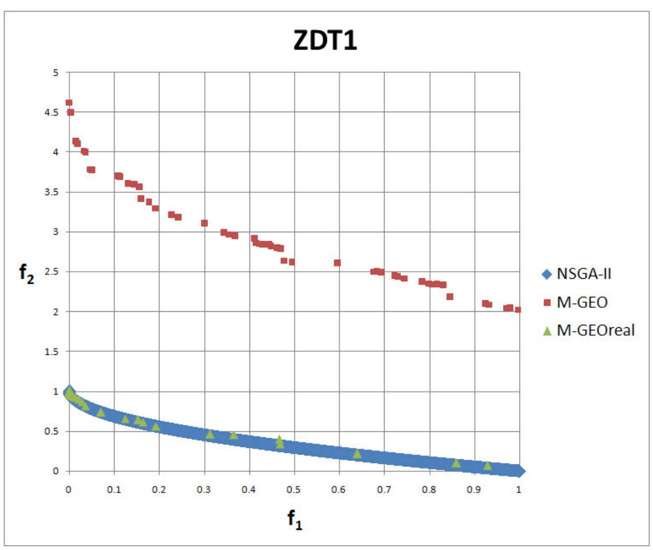

Figure 4. ZDT1 Pareto Front. In blue is the Pareto Front given by NSGAII; in red by M-GEO; in green by M-GEO$_{real}$

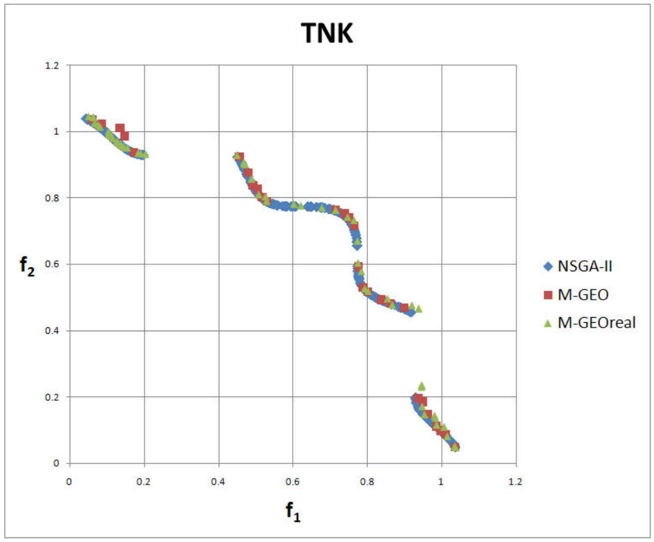

Figure 5. TNK Pareto Front. In blue is the Pareto Front given by NSGAII; in red by M-GEO; in green by M-GEO$_{real}$

M-GEO shows the worst performance in all test functions. Although, M-GEO recovers the shape of the TNK Pareto Front, there are lacks in some regions of the Pareto Front and some solutions are in dominated region of the objective space. In ZDT1 test function, the algorithm cannot find the Pareto Front. This occurred because of the great number of variables of the test function. At each iteration, M-GEO performed 240 objective function evaluations. Therefore, M-GEO had few iterations to recover the Pareto Front of ZDT1.

M-GEO$_{real}$ algorithm was capable to recover the shape of the Pareto Fronts of the test functions, especially for TNK. M-GEO$_{real}$ returns few solutions in ZDT1 because the algorithm had difficult to deal with the great number of variables. However, NSGAII was capable to return much more solutions. Therefore, the M-GEO$_{real}$ algorithm needs to be developed to become a competitive algorithm.

5. MATHEMATIC MODEL OF THE RIGID-FLEXIBLE SATELLITE

In this test the M-GEO$_{real}$ algorithm was used to optimize, simultaneously, the time to control a rigid-flexible satellite and the energy spent by the attitude control to perform this task. The satellite model used to check the M-GEO$_{real}$ performance was based in a model used by Hassmann e Fenili (2007). The satellite was modeled as a rigid cube with edge equal to 1.5 m with a flexible beam of 2 m clamped in one of its edge. The satellite is free to rotate only in the XY plane. A graphic representation of the satellite model is presented in Figure 9.

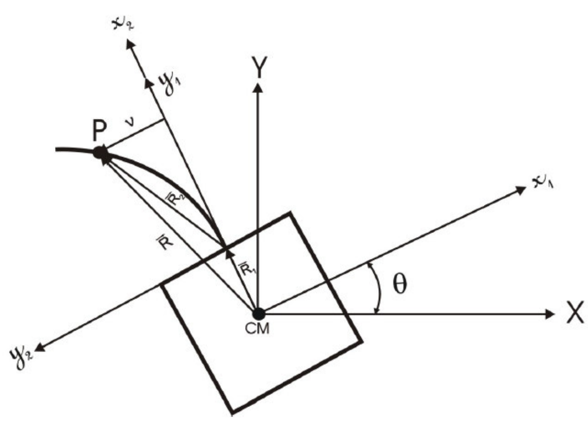

Figure 6. Graphic representation of the rigid-flexible satellite.

In order to model the flexible beam, it was used an Euler-Bernoulli formulation (Rezende *et al.*, 2004). Considering only the first vibration mode of the beam, this model can be described by the following equations

$$\begin{cases} 2C_3\ddot{\theta} + C_2\ddot{p} + 2C_1p^2\ddot{\theta} + 4C_1\dot{\theta}p\dot{p} = \xi \\ 2C_1\ddot{p} + C_2\ddot{\theta} - 2C_1\dot{\theta}^2 p + 2C_1\omega^2 p = 0 \end{cases} \tag{6}$$

where p is the modal state, θ is the rotation angle of the central body, ω is the beam first mode of vibration and C_1, C_2 and C_3 are constants given by

$$C_1 = \frac{\rho A_h}{2} \tag{7}$$

$$C_2 = \rho A_h R_1 \lambda + \rho A_h \mu \tag{8}$$

$$C_3 = \frac{1}{2}\left[\rho A_h\left(R_1^2 l + R_1 l^2 + \frac{l^3}{3}\right) + I_o\right] \tag{9}$$

That system of equations in the state space form of 1º order can be written as

$$\dot{x}_1 = x_2 \tag{10}$$

$$\dot{x}_2 = \frac{2C_1C_2\omega_1^2 x_3 - 2C_1C_2x_2^2x_3 - 8C_1^2x_2x_3x_4 + 2C_1\xi}{4C_1^2x_3^2 + C_4} \tag{11}$$

$$\dot{x}_3 = x_4 \tag{12}$$

$$\dot{x}_4 = \frac{4C_1C_3x_2^2x_3 + 4C_1^2x_2^2x_3 - 4C_1C_3\omega_1^2x_3 - 4C_1^2x_3^3 + 4C_1C_2x_2x_3x_4 - C_2\xi}{4C_1^2x_3^2 + C_4} \tag{13}$$

where $C_4 = 4C_1C_3 - C_2^2$, $x_1 = \theta$, $x_2 = d\theta/dt$, $x_3 = p$ and $x_4 = dp/dt$. The full mathematic development of these equations can be found in the work of Mainenti-Lopes (2008). In this work, one used a non-linear control law (Rietz and Inman, 2000) given by

$$\xi = -K_1x_1 - K_2x_2 - K_3x_1x_2 \tag{14}$$

where K_1, K_2 and K_3 are gains of the control law. The parameters value of the satellite and a brief description of each parameter is presented in Tab 2.

Table 2. Parameters value of the satellite.

Parameter	Description	Value	Unit
A_h	Cross section of the beam	7.5×10^{-4}	m^2
ρ	Aluminimum density	2700	kg/m^2
l	Beam length	2.0	m
E	Young's modulus	7×10^{10}	N/m^2
a_1l	Eigen value associated to the beam first mode of vibration	1.878	-
I_h	Moment of inertia of the beam cross section about the neutral axis	1.5625×10^{-9}	m^4
I_o	Satellite's main body moment of inertia about its center of mass	1125	$kg.m^2$
ω	Beam first mode of vibration	18,0001	rad/s
R_1	Half of the central body edge	0,75	m

6. M-GEO$_{real}$ PERFORMANCE IN OPTIMAL SATELLITE ATTITUDE CONTROL

In this performance test, the M-GEO$_{real}$ was used to identify the set of gains (K1, K2 and K3) that minimize the time to control the satellite and the energy spent by the controller, considering that the initial attitude angle as 28.65º and the final attitude angle needs to be 0º.

In order to integrate the equations of motion, one uses the fourth order Runge-Kutta algorithm (RK4). For each set of gains tried by M-GEO$_{real}$, it called the Runge-Kutta algorithm that allowed to calculate the value of energy spent and time to control de satellite using the following equations

$$F_1 = T \tag{15}$$

$$F_2 = \sum_{i=0}^{hT} \left| \left(K_1 x_{1i} + K_2 x_{2i} + K_3 x_{1i} x_{2i} \right)\left(x_{1i} - x_{1i+1} \right) \right| \tag{16}$$

where T is the instant that the satellite is controlled. In other words, the instant that fulfill the following conditions: $|\theta| < 1{,}75 \times 10^{-3}$ rad and $|\dot{\theta}| < 5{,}23 \times 10^{-4}$ rad/s. h is the step of RK4, x_{1i} and x_{2i} are the angle shift and angle velocity of each i-th iteration of RK4, respectively. That is, the energy spent by the controller. Equation 14 and 15 were minimized considering $0 < K_1 < 20000$, $0 < K_2 < 20000$ and $0 < K_3 < 20000$. The constraints were treated discarding all solutions unfeasible.

The M-GEO$_{real}$ performance was compared to its previous version M-GEO and the LQR method (Stengel, 1994). The LQR method can be treated as weighed sum method. Therefore, the LQR can only returns all Pareto Front if the weighs is changed several times. In this work, only one solution of LQR will be used. The M-GEO Pareto Front and the LQR solution was obtained by Mainenti-Lopes (2008). Figure 7 shows the comparison of performance between these three algorithms.

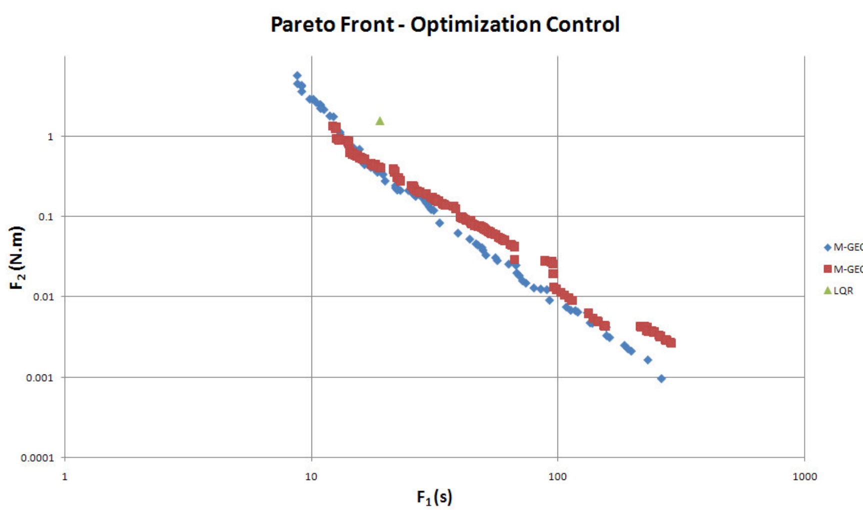

Figure 7. Pareto Front of the optimal control problem. In blue is M-GEO$_{real}$ solution, in red, the M-GEO and in green is the LQR solution.

M-GEO$_{real}$ shows good performance in this problem. It recovers a larger extension of the Pareto Front than its previously version and its solutions dominated a great number of solutions given by M-GEO and the LQR solution. Besides, M-GEO$_{real}$ uses only 3.2×10^5 function evaluations, while M-GEO uses 6.6×10^5.

7. CONCLUSION

In this work, a new multi-objective was proposed. Called M-GEO$_{real}$, this evolutionary algorithm was based on its mono-objective version (GEO$_{real}$) and the multi-objective version of GEO algorithm (M-GEO). M-GEO$_{real}$ algorithm uses real codification of the design variables; on the contrary of M-GEO that uses binary codification.

As brand new algorithm, it becomes necessary to test its performance against test functions. The chosen test functions were ZDT1 and TNK. ZDT1 shows a great difficult because of its large number of variables and TNK has discontinuities and non-convexities in its Pareto Front. M-GEO$_{real}$ performance was compared to NSGAII algorithms.

In this test, M-GEO$_{real}$ algorithm shows good performance recovering all extension of the Pareto Front, especially for TNK. However, M-GEO$_{real}$ returns few solutions in ZDT1 because the algorithm had difficult to deal with the large number of variables. NSGAII was capable to return much more solutions than M-GEO$_{real}$.

A second test of performance made in work had as goal the optimization of an attitude control of satellite. A rigid-flexible satellite model with a non-linear control law was used. M-GEO$_{real}$ was used to optimize, simultaneously, the time to control the satellite and energy spent by the controller in this task. M-GEO$_{real}$ results were compared to M-GEO performance and with LQR solution.

M-GEO$_{real}$ shows great performance to tackle this satellite attitude control problem. The solutions in its Pareto Front dominated several solutions given by M-GEO and the LQR solution. Besides, it was capable to recover a large extension of the Pareto Front and it takes less functions evaluation to do this job.

8. ACKNOWLEDGEMENTS

The authors acknowledge the financial support provided by CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

9. REFERENCES

Bäck, T., and Schwefel, H. P., 1993, “An Overview of Evolutionary Algorithms for Parameter Optimization”, Evolutionary Computation, Vol.1, No. 1, 1993, pp. 1 –23.

Davis, L. D., De Jong, K., Vose, M. D., and Whitley, L. D., 1999, “Evolutionary Algorithms”, Mathematics and Its Applications, Vol. 111, Springer-Verlag, Berlin, Germany.

Deb *et al*., 2000. Deb, K., Agrawal, S., Pratap, A. and Meyarivan T., 2003, “A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II”, Parallel Problem Solving from Nature PPSN VI, Vol. 1917/2000, pp. 849-858, Springer Berlin, Heidelberg.

De Sousa, F. L., Ramos, F. M., Paglione, P. and Girardi, R. M., 2003, “New Stochastic Algorithm for Design Optimization”, AIAA Journal, Vol. 41, No. 9, pp. 1808 –1818.

Galski, R. L., De Sousa, F. L. and Ramos, F. M., 2005, “Application of a New Multiobjective Evolutionary Algorithm to the Optimum Design of a Remote Sensing Satellite Constellation”, Proceedings of the 5th International Conference on Inverse Problems in Engineering: Theory and Practice, Cambridge, UK.

Galski, R. L., 2006, “Development of Improved, Hybrid, Parallel and Multiobjective Versions of the Generalized Extremal Optimization Method and Its Application to the Design of Space Systems (in portuguese)”, Doctor Thesis, Nacional Institute for Space Research (INPE).

Hassmann and Fenili, 2007, “Attitude and Vibration Control of a Satellite with a Flexible Solar Panel Using LQR Tracking with Infinite Time”, Proceedings of the 19th International Congress of Mechanical Engineering, Brasília, DF, Brazil.

Mainenti-Lopes, I., 2008, “Satellite Attitude Control of Rigid-Flexible Satellite Using the Generalized Extremal Optimization with a Multi-Objective Approach (in portuguese)”, Master Thesis, Nacional Institute for Space Research (INPE).

Mainenti-Lopes I., De Sousa, F. L., Souza, L. C. G., 2008, “The Generalized Extremal Optimization with Real Codification”, Proceedings of the EngOpt 2008 - International Conference on Engineering Optimization, Rio de Janeiro, Brazil.

Rietz R. W., Inman, D. J., 2000, “Comparison of Linear and Nonlinear Control of a Slewing Beam”. Journal of vibration and control, Vol. 6, No. 2, pp. 309 - 322.

Rezende, C. P., Fenili, A.; Souza, L. C. G., 2004, “Modelagem e Controle de Estruturas Flexíveis - Abordagem Ideal e Não Ideal: Caso Linear”, Proceedings of the 3rd Brazilian Conference on Dynamics, Control and Applications - DINCON, Ilha Solteira.

Stengel, R. F., 1994, Optimal Control and Estimations, Dover publications, inc.

10. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.