DINCON'09
8TH BRAZILIAN CONFERENCE ON DYNAMICS, CONTROL AND APPLICATIONS
MAY 18-22, 2009

SOLVING THE LOST IN SPACE PROBLEM WITH STAR TRACKER SENSOR USING GENERALIZED EXTREMAL OPTIMIZATION (GEO)

Carlos José Alves Moreira [1],Fabiano Luis de Sousa [2],Roberto V. F. Lopes [3]

[1] Inpe – Instituto Nacional de Pesquisas Espaciais, São José dos Campos - SP, Brasil, cjamoreira@gmail.com
[2] Inpe – Instituto Nacional de Pesquisas Espaciais, São José dos Campos - SP, Brasil, fabiano@dem.inpe.br
[3] Inpe – Instituto Nacional de Pesquisas Espaciais, São José dos Campos - SP, Brasil, roberto@dss.inpe.br

Abstract: In this work, preliminary results of a study to determine the efficacy of using the Generalized Extremal Optimization algorithm (GEO) for star pattern identification for the lost in space problem, are presented. The results of the simulations are presented in graphics and tables which show the identifications and the algorithm performance.

Keywords: Star Tracker Sensor, Generalized Extremal Optimization (GEO), Spacecraft Attitude.

1. INTRODUCTION

The lost in space problem, consist in determine the spacecraft vehicle axis orientations in relationship to the inertial coordinates system without any other information. It is very important for the success of the space mission, because the correct determination of this orientation supply the correct spacecraft attitude to the control system.

In this work, this problem is solved using just information of the stars coordinates observed by the field of view (FOV) of a star tracker sensor attached to the spacecraft.

This sensor determine their attitude comparing the stars coordinates observed by their FOV and the stars coordinates presented in a catalogue stored in the on board memory. This comparison is performed by star patterns algorithms which search in the catalogue similar patterns to the FOV observation.

The evolutionary algorithm GEO is used in this work for realize this identification using an optimization approach, by minimizing an objective function which computes the discrepancy between the stars in the sensor´s FOV with the ones in candidate FOVs in the catalogue.

The simulations are divided in two different sets, where in the first set is used a plan catalogue with randomly distributed stars and in the second set is used a real catalogue with stars in spherical coordinates.

In the second set, with the stars observed by the FOV sensors and the stars identified by the algorithm in the catalogue, is possible calculates the spacecraft attitude using an attitude determination algorithm. In this work is used singular value decomposition (SVD) [1] to determine the attitude.

2. STAR TRACKER SENSOR

The star tracker sensor is an attitude determination sensor that is being largely used in space missions because their accuracy in calculate it.

These sensors, of a more general way, can be divided in two classes [2].

In the first class the star tracker sensors are called not autonomous, where these sensors just inform to the control system of the spacecraft the coordinates of the stars observed by their FOV, the identification of the stars in catalogue and the determination of the attitude is not make by the sensor [2].

In the second class, these sensors are able to identify the stars observed by their FOV and calculate their attitude independently [2].

The star tracker sensors of the second class are the ones addressed in this work.

2.1. Autonomous Star Tracker Operation

In a concise form we can describe the autonomous star tracker operation in four steps [2].

- First step: Image acquisition of the stars by the FOV sensor.
- Second step: mapping of the stars observed in the matrix of the FOV sensor.
- Third step: recognition of the stars observed by the sensor FOV in the star catalogue stored in the sensor’s on board memory by the star pattern algorithm.
- Fourth step: Spacecraft attitude determination.

These four steps can by better observed in the figure 1[2].

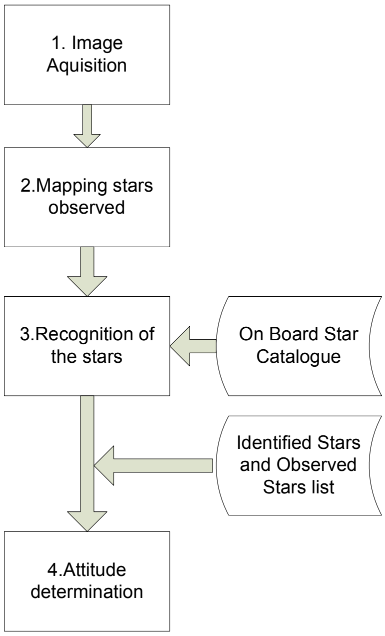

Fig. 1. Autonomous Star Tracker Operation

There is not yet a definitive solution for the star pattern identification problem, in the literature there are many approaches and algorithms proposed, in this work is proposed the use of a new evolutionary algorithm to solve this problem.

3. GENERALIZED EXTREMAL OPTIMIZATION (GEO)

The Generalized Extremal Optimization algorithm (GEO), is a new evolutionary algorithm proposed by De Sousa [3] [4], this algorithm is easy to implementation and can be applied in anyone optimization problem.

In this algorithm a sequence of species represented by a string of bits creates a population, to each species (bit) is attributed a fitness which define what species are more adapted after mutate (flipped bit). The design variables are encoded in the string of bits as show in figure 2[3] [4].

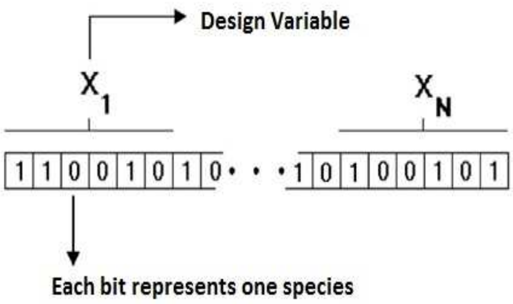

Fig. 2. Design Variable encoded in a binary string

The number (m) of the bits necessary for each design variable (X_i) to have a precision (p) can by calculate by equation 1[3] [4].

$$2^m \geq \left[\frac{X_{i\_\sup} - X_{i\_\inf}}{p} + 1 \right] \quad (1)$$

Where the X_{i_sup} and the X_{i_inf} are the higher and low design variables limits.

The values of the design variables encoded in real number can by calculated by equation 2 [3] [4].

$$X_i = X_{i\_\inf} + (X_{i\_\sup} - X_{i\_\inf}) \cdot \frac{X_{i\_b^{10}}}{2^m - 1} \quad (2)$$

Where $X_{i\_b}^{10}$ is an integer number obtained from the transformation of the design variable X_i in a decimal representation.

The implementation of the GEO algorithm are represented in figure 3 [5]:

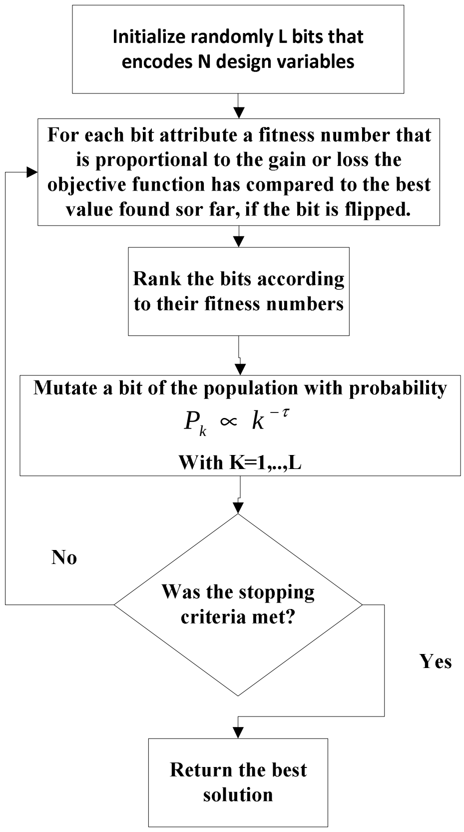

Fig. 3. Canonical GEO flowchart

The parameter τ, is a adjustable parameter which determine how stochastic is the search. If the parameter τ is equal to zero the algorithm is completely stochastic.

With the objective to make the search more efficient in the GEO algorithm De Sousa [3] [4] proposed the $GEO_{var.}$

The change realized in GEO_{var} does not modify the original GEO concept and just occur in the fact that now one bit of each design variable is mutated.

3.1. GEO$_{real}$

With the purpose of the use a real codification for the representation of the design variables, Lopes [5] developed two versions of the GEO using real codification.

This codification avoids has to stipulate a precision for the algorithm.

The mutation in the design variables occur now using the equation 3 [5].

$$X'_i = X_i + N(0,\sigma)\cdot X_i \tag{3}$$

Where the $N(0,\sigma)$ is a random number with zero mean and standard deviation σ.

Lopes also propose the GEO_{real2} differing of the GEO_{real1} in fact of the GEO_{real2} realizes P changes in design variables with P different standard deviations. These standard deviations can be calculated with the equation 4 [5].

$$\sigma_{i+1} = \frac{\sigma_i}{2\cdot i} \tag{4}$$

With i = 1 to P.

Other modification in GEO_{real2} is in the fact that now the mutation occur in all design variables similarly GEO_{var}.

The GEO_{real2} is showed in figure 4 [5].

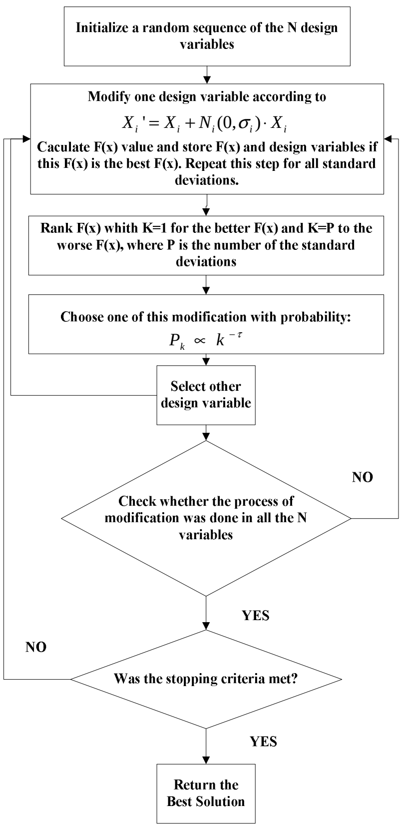

Fig. 4. GEO_{real2} flowchart

In this work, the simulations uses the GEO_{real2} for the star pattern identification.

4. STAR PATTERN INDENTIFICATION

The simulations are divided in two sets, in the first set was used a catalogue with 700 random points distributed in a plan with X_{min} = -150, X_{max} = 150 and Y_{min} = -150, Y_{max} = 150 simulating the stars in sky [6] as shown in figure 5.

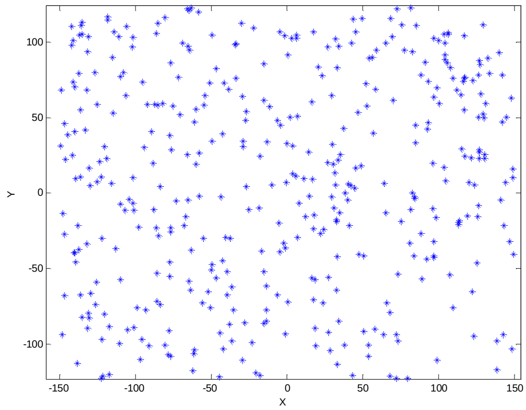

Fig. 5. Random Stars in Flat Catalogue

In this set of the simulation, the objective is to find the coordinates of the center of the FOV simulated and the stars presents in FOV.

For this, is simulated a FOV view in the random catalogue as shown in example in the figure 6.

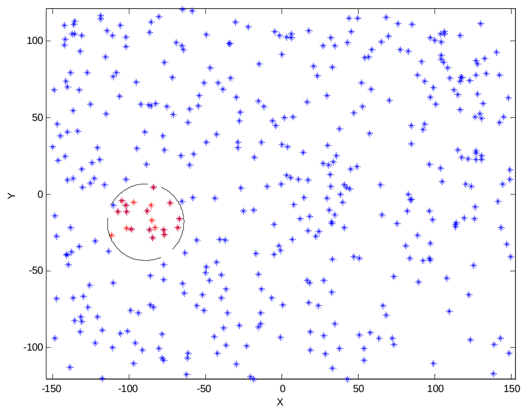

Fig. 6. FOV simulated with coordinates X = -89 and Y = -18

With the FOV simulated is make a vector with the distance between 20 stars closer to the FOV center and the same as shown in figure 7[7] and equation 5[7].

$$D_a = [R_1, R_2, \ldots, R_n] \tag{5}$$

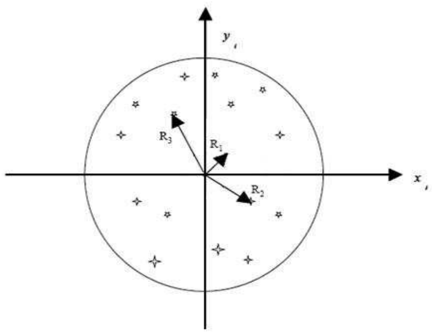

Fig. 7. Distance between stars in FOV and FOV center

Before the FOV simulation, the GEO algorithm starts the search. The design variables are the coordinates (X, Y) of the candidates FOV center.

Similarly is make a vector with stars of the candidates FOVs as shown in equation 6[7].

$$D_i = [R_{i1}, R_{i2}, ..., R_{in}] \tag{6}$$

If the number of the stars in the FOV is less than 20, the vector is completed with zeros.

The GEO algorithm minimizes the objective function which calculates the discrepancy between the stars in the FOV simulated and the stars in candidate FOV. The objective function is calculated with equation 7 [7].

$$C_i = \sum_{k=1}^{n} \left| D_a[R_k] - D_i[R_k] \right| \tag{7}$$

In the second simulations set is used a real catalogue with stars in spherical coordinates. For the simulation of the FOV is make a simulation of the spacecraft attitude and then verify the stars contained in FOV as shown in figure 8.

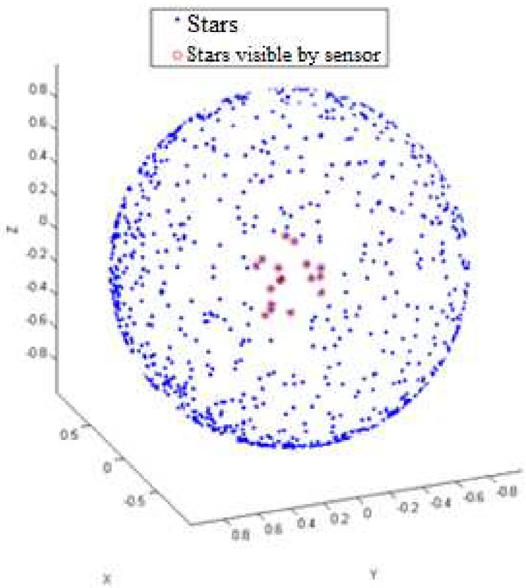

Fig. 8. FOV Simulation with real catalogue

The procedures of this set of simulations are similar to the first set, the difference is that the design variables are now right ascension (α) and declination (δ) which are transformed into (X, Y, Z) in the celestial sphere with the equations 8, 9 and 10:

$$X = \cos(\delta) \bullet \cos(\alpha) \tag{8}$$

$$Y = \cos(\delta) \bullet sen(\alpha) \tag{9}$$

$$Z = sen(\delta) \tag{10}$$

Other modification is that the vector with the distance showed in equations 5 and 6, are transformed in vectors with angular distance between the stars and the FOV center.

The algorithm GEO_{real2} minimizes the equation 7 and returns the stars observed by the FOV indentified in the catalogue.

With the stars observed by the FOV sensors and the stars identified by the algorithm in the catalogue, is possible calculates the spacecraft attitude using an attitude determination algorithm. In this work is used singular value decomposition (SVD) [1] to determine the attitude.

5. RESULTS

In the first simulations set, the FOV was simulate with 25 units of the radius and for each target coordinate was made a mean of the 50 runs.

The results are show in tables 1 to 3 and figures 9 to 11.

- Results with target coordinates (X = 0, Y = 0):

	Mean Value	Mean Time for each execution (s)	Mean of evaluations of the objective function
Coordinate X	0,047	30,3	4.130,3
Coordinate Y	0,009		

Tab. 1. Results with (X=0, Y=0)

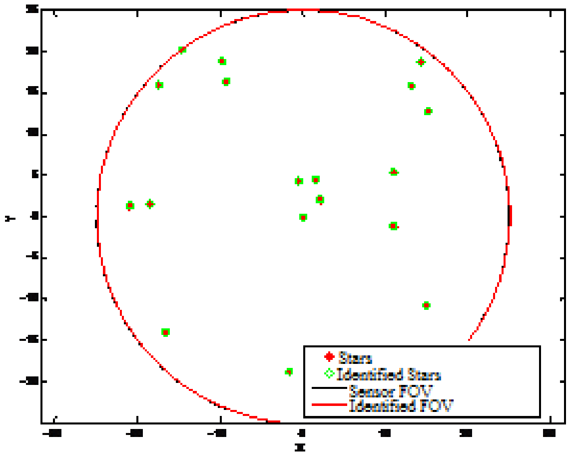

Fig. 9. Graphical Results with (X=0, Y=0)

- Results with target coordinates (X = 40, Y = 75):

	Mean Value	Mean Time for each execution (s)	Mean of evaluations of the objective function
Coordinate X	39,85	59,18	6.721,78
Coordinate Y	75,08		

Tab. 2. Results with (X=40, Y=75)

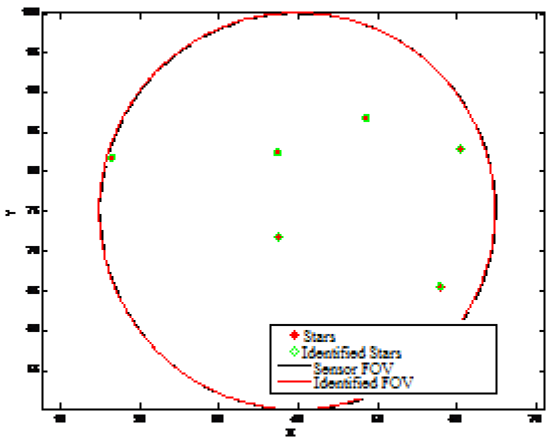

Fig. 10. Graphical Results with (X=40, Y=75)

- Results with target coordinates (X = -89, Y = -18):

	Mean Value	Mean Time for each execution (s)	Mean of evaluations of the objective function
Coordinate X	-89,01	151,42	25.435
Coordinate Y	-17,97		

Tab. 3. Results with (X=-89, Y=-18)

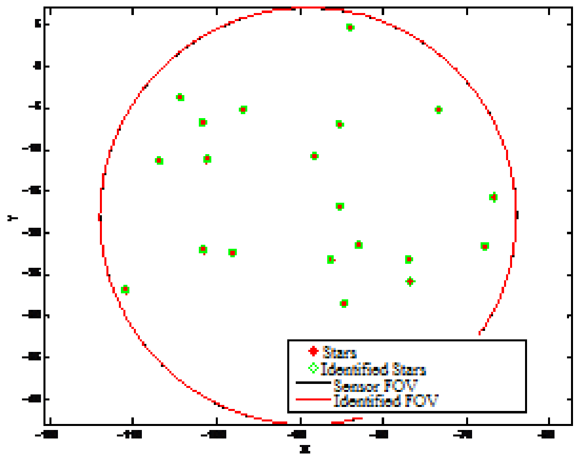

Fig. 11. Graphical Results with (X=-89, Y=-18)

In the second simulation set, was simulated the sensor with the characteristics showed in table 4.

Field of view	25,5º x 25,5º
Precision	≈ 10 arcsec (3σ)
Visual Magnitude	Between 0 e 5
Number of Rastreable stars	16
Output	Triaxial Attitude

Tab. 4. Sensor Configuration

The results of this simulation set was obtained realizing 50 runs with random spacecraft attitudes. These results are showed in table 5.

	Star Pattern with GEO
Mean Time for each execution (s)	34,94 s
Wrong identifications	0
Not identified	0
Standard deviation in X (°)	$2,5 \cdot 10^{-4}$
Standard deviation in Y (°)	$2,39 \cdot 10^{-4}$
Standard deviation in Z (°)	0.002

Tab. 5. Results with real catalogue

6. CONCLUSIONS

The results show that the GEO_{real} is able to identify the stars observed by the FOV sensor.

In the future work shall be done a study to reduce the mean of identification time and realize the comparison of efficiency between the proposed approach and a frequently used approach in this problem.

7. ACKNOWLEDGMENT

The authors are grateful to the cooperation of the CAPES and INPE.

8. REFERENCES

[1] Markley,F. L., (1988), “Attitude Determination using Vector Observations and the Singular Value Decomposition.” Journal of the Astronautical Sciences, Vol. 36, July-September 1988, pp. 245-258.

[2] Fialho, M. A. A.; Lopes, R. V. F.; Saotome, O.;“Um Método de Calibração Automática em Magnitude para Sensores de Estrelas”; 4° Congresso Temático de Dinâmica, Controle e Aplicações – Dincon (2005); Bauru – SP.

[3] De Sousa, F. L., Otimização extrema generalizada: um novo algoritmo estocástico para o projeto ótimo. 2002-09. 142 p. (INPE-9564-TDI/836). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2002.

[4] De Sousa, F. L.; Ramos, F. M.; Paglione, P.; “New Stochastic Algorithm for Design Optimization”; AIAA Jornal, Vol. 41, N° 9, September 2003 pp. 1808-1818.

[5] Lopes, I. M. L., Controle de atitude de satélites rígido-flexíveis usando a otimização extrema generalizada com abordagem multi-objetivo. 26 de Fevereiro de 2008. (INPE--T/). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos.

[6] Moreira, C. J. A., De Sousa, F. L., Lopes, R. V. F. – Identificação de Estrelas Para Determinação de Atitude de Veículos Espaciais Utilizando o Algoritmo da Otimização Extrema Generalizada (GEO). XIV Colóquio Brasileiro de Dinâmica Orbital, Águas de Lindóia, BR, 2008.

[7] Paladugu, L.; Seisie-Amoasi, E.; Williams, B. G. e Schoen, M. P. “Intelligent Star Pattern Recognition For Attitude Determination: “Lost in Space” Problem”. Journal of Aerospace Computing, Information and Communication, Vol. 3, November 2006.