STAR PATTERN IDENTIFICATION FOR SPACECRAFT ATTITUDE DETERMINATION USING GENERALIZED EXTREMAL OPTMIZATION

Carlos José Alves Moreira, cjamoreira@gmail.com
Fabiano Luis de Sousa, fabiano@dem.inpe.br
Instituto Nacional de Pesquisas Espaciais – INPE, Av. dos Astronautas 1758, 12227-010 São José dos Campos, SP - Brasil

Roberto V. F. Lopes, roberto@dss.inpe.br
Instituto Nacional de Pesquisas Espaciais – INPE, Av. dos Astronautas 1758, 12227-010 São José dos Campos, SP - Brasil

Abstract. *In this work, the results of a study to determine the efficiency of using the Generalized Extremal Optimization (GEO) algorithm for star pattern identification for the lost in space problem, are presented. The objective is to find the attitude of a spacecraft by identifying in a star catalogue, the stars observed in the field of view (FOV) of a star sensor attached to the spacecraft. This problem is tackled by GEO 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. Comparison of efficiency between the proposed approach and a frequently used approach called triangle algorithm is made. This algorithm realizes the identification comparing the triangle geometry shaped by three stars present in the sensor´s FOV with similar geometries presented in the catalogue. A real star catalogue in spherical coordinates is used for the simulations.*

Keywords*: Star Tracker Sensor, Star Pattern Identification, Generalized Extremal Optimization, Spacecraft Attitude*

1. INTRODUCTION

The problem of the spacecraft attitude determination, consists in calculating the spacecraft axis orientation in relation to the inertial system axis. This is very important for the success of the space mission, because the correct determination of this orientation supplies the correct spacecraft attitude to its control system.

To solve this problem it is necessary to dispose of the reference vectors whose directions are known in relation to the spacecraft. These vectors are obtained from measurements taken by attitude sensors. These sensors can be horizon sensors, solar sensors, magnetometers, star tracker sensors, etc.

In this work, this problem is solved by using just information of the stars coordinates observed by the field of view (FOV) of a star tracker sensor attached to the spacecraft. These sensors stand out because its precision in calculating the spacecraft attitude, is on the order of arc seconds.

This sensor determines its attitude comparing the stars coordinates observed by its FOV with the stars coordinates presented in a catalogue stored in the sensor’s on-board memory. This comparison is performed by star pattern algorithms which search the catalogue for similar patterns to the FOV observation.

Basically, these sensors are divided in two classes, where in the first class they are called non-autonomous and are not able to determine alone the spacecraft attitude and in the second class these sensors are able to determine alone the spacecraft attitude and are called autonomous sensors. The second class of star tracker sensors is the focus of this work.

The autonomous star tracker sensor essentially works in two operation modes. In the first one, the sensor determines the spacecraft attitude without any previous information. This is called the acquisition mode. The second one is called actualization mode and in this mode the sensor just atualize the spacecraft attitude obtained by the acquisition mode.

The simulations performed in this work are focused in the acquisition mode, where the star pattern algorithm indentifies the stars observed by the sensor FOV in the catalogue without any previous information. There are many approaches to solve this problem and recently evolutionary algorithms have been also used (Paladugu et. al, 2006) (McClintock, S. et al, 1998).

The evolutionary algorithm GEO, used in this work, is a new evolutionary algorithm that has been used successfully in several optimization problems, and here it is used to perform identification of the stars observed by sensor FOV 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.

A frequent approach used for solves this problem, called Triangle Algorithm, is implemented for performance comparison with GEO approach.

The Triangle Algorithm executes the identification by comparing the triangle geometry shaped by three stars presented in the sensor FOV with similar geometries presented in the catalogue. All simulations in this work use a real star catalogue with 1577 stars arranged in the celestial sphere.

2. STAR TRACKER SENSOR

The star tracker sensor is an attitude determination sensor that is being largely used in space missions because of its accuracy in calculating the spacecraft attitude.

These sensors, in a more general way, can be divided in two classes (Fialho et. al, 2005): in the first class these sensors are called not autonomous because they just inform to the control system of the spacecraft the coordinates of the stars observed by its FOV and the identification of the stars on catalogue and the spacecraft attitude determination is not make by the sensor. In the second class these sensors are able to identify the stars observed by their FOV and calculate their attitude independently.

This work has as its focus the second class, where the sensors are able to identify the stars observed by their FOV and determine the spacecraft attitude. In the fig.1a an autonomous star tracker sensor is shown (Fialho, 2007) and in the fig.1b is shown an exploded view of MultMission space Platform with two star tracker sensors which has been developed in Brazilian Institute for Space Research - INPE.

Figure 1a. Star Tracker Sensor

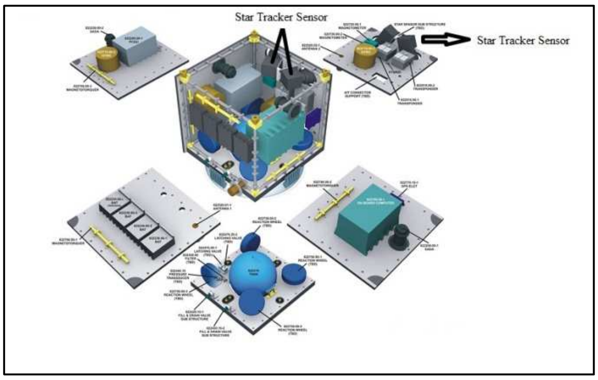

Figure 1b. Exploded view of PMM with two star tracker sensors

2.1. Autonomous Star Tarcker Operation

This kind of sensors calculates the spacecraft attitude based in observations of the stars made by its FOV. The stars which are observed by its FOV are mapped in an electronic matrix which send those information's for a star pattern algorithm. These algorithms realize the comparison of the pattern of the stars observed with the ones stored in the catalogue in the sensor's onboard memory. This catalogue contains the stars coordinates mapped in the inertial system.

After the identification of the stars observed by the FOV sensor in the catalogue, the stars coordinates mapped in the electronic matrix and the ones identified mapped in the inertial system are sent for an attitude determination algorithm that performs the calculation of the attitude.

The attitude calculated is then sent to the attitude and orbit control system.

The operation procedure of the autonomous star tracker sensor can be described in a concise form in four steps (Fialho et. al, 2005):

- First step: Image acquisition of the stars made by 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 be better observed in the fig. 2.

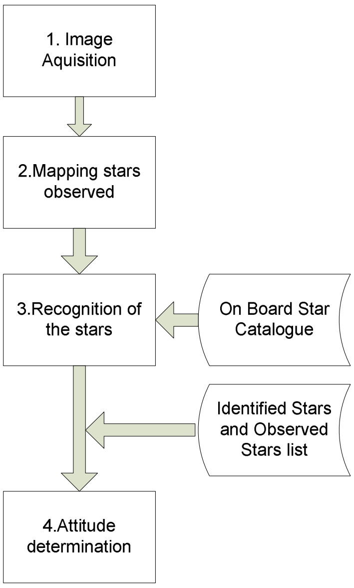

Figure 2. Autonomous Star Tracker Operation

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 called Generalized Extremal Optimization (GEO) and the results are compared with a Triangle Algorithm approach.

3. GENERALIZED EXTREMAL OPTMIZATION (GEO)

The Generalized Extremal Optimization algorithm (GEO), is a new evolutionary algorithm proposed by De Sousa (2002) and De Sousa et al. (2003), this algorithm is easy to implementation and can be applied in any 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 fig. 3(De Sousa, 2002).

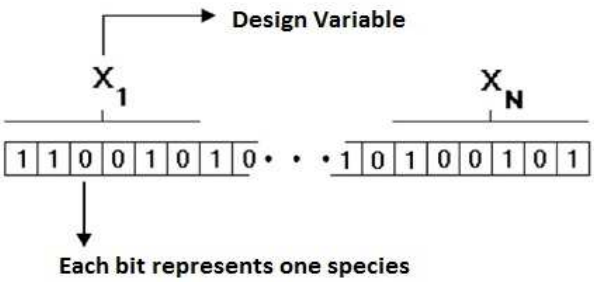

Figure 3. Design Variable encoded in a binary string

The number (m) of bits necessary for each design variable (X_i) to have a precision (p) can be calculated by the eq. (1) (De Sousa, 2002).

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

Where the X_{i_sup} and the X_{i_inf} are respectively the upper and lower limits of design variables.

The values of the design variables encoded in real number can be calculated by eq. (2) (De Sousa, 2002).

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

Where X_{i_b10} is an integer number obtained from the transformation of the design variable X_i in a decimal representation.

The implementation of the GEO algorithm is presented in fig. 4 (Lopes, 2008).

The parameter τ showed in fig. 4, is an adjustable parameter which value must be chosen depending on the optimization problem being tackled. This parameter permits set the determinism degree of the search, it has been observed that the optimal values for this parameter are between [0,75 – 3,0].

The canonical GEO algorithm is shown in fig. 4. Other implementations derived from it have been developed (Galski, 2006) (Lopes, 2008) in order to improve its performance.

This work uses the GEO_{real} to identify the stars observed by FOV sensor, the GEO_{real} was chosen after tests done with canonical GEO, GEO_{var} and GEO_{real} to solve this problem and the GEO_{real} achieved better results because its codification is continues and does not need a precision stipulated as is done in GEO and GEO_{var}.

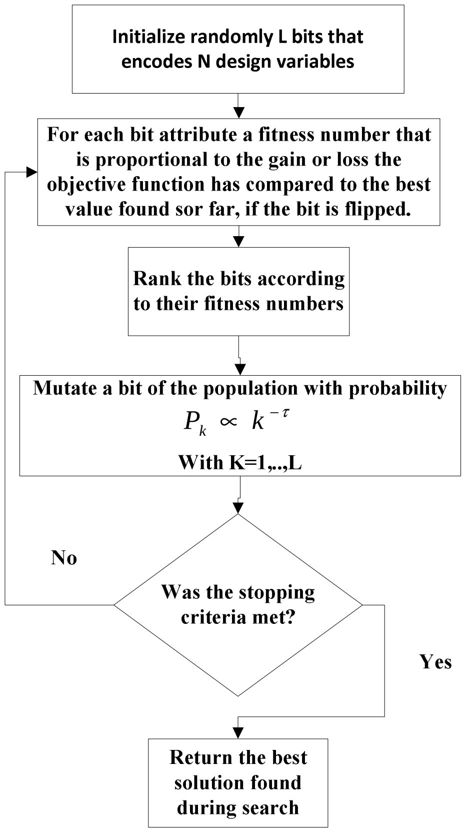

Figure 4. Canonical GEO flowchart

3.1 GEO_{real}

With the purpose of the use a real codification for the representation of the design variables, Lopes (2008) developed two versions of the GEO using real codification.

This codification avoids the necessity to stipulate a precision for the design variables and the mutation in the design variables is made now using the eq. 3 (Lopes, 2008).

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

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

Lopes also proposed the GEO_{real2} that differs from GEO_{real1} in the fact of the GEO_{real2} realizes P changes in design variables with P different standard deviations. These standard deviations are calculated using the eq. 4 (Lopes, 2008).

$$\sigma_{i+1} = \frac{\sigma_i}{2 \cdot i} \quad (4)$$

With i = 1 to P.

Another modification done in the GEO_{real2} is in the fact that now the mutation occurs in all design variables similarly to what occurs in GEO_{var}.

The flowchart of the implementation of GEO_{real2} is showed in fig. 5 (Lopes, 2008).

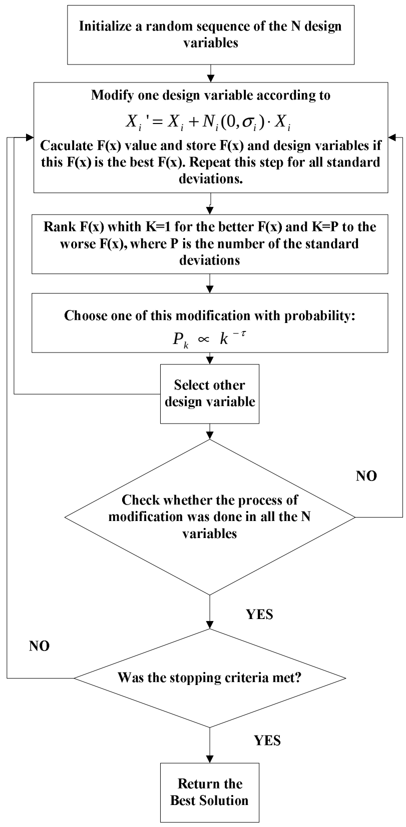

Figure 5. GEO_{real2} flowchart

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

4. STAR TRACKER SENSOR SIMULATION MODEL

For the star tracker sensor simulation, were set some basic parameters of the sensor, these parameters are shown in Tab. 1.

Table 1. Star Tracker Sensor Parameters

Field of View (°)	25,5x25,5
Visual Magnitude	Between 0 and 5
Number of Rastreable Stars	16
Output	Triaxial Attitude

The star tracker sensor model consists basically in simulate the FOV of the sensor. To do this, it is used a real catalogue with 1577 stars arranged in a celestial sphere as shown in fig. 6.

In this catalogue the coordinates (X,Y,Z) of the 1577 stars are organized in a matrix V_{3x1577}.

It is also made a simulation of the attitude matrix A_{3x3}. With the matrices A_{3x3} and V_{3x1577} is possible to mount a matrix U_{3x1577} containing the coordinates (X,Y,Z) of the 1577 stars mapped in the FOV sensor system using the eq. (5).

$$U_{3x1577} = A_{3x3} \cdot V_{3x1577} \quad (5)$$

With the matrix U_{3x1577} is possible to mount the matrix $U_{vis(3xn)}$ containing the (n) stars visible by the FOV sensor. To do this the stars of the matrix U_{3x1577} shall comply with the conditions (6) and (7).

$$U_{1xm} \leq U_{3xm} \cdot tg\left(12{,}75^{o}\right) \tag{6}$$

$$U_{2xm} \leq U_{3xm} \cdot tg\left(12{,}75^{o}\right) \tag{7}$$

Where the index m = 1 to 1577.

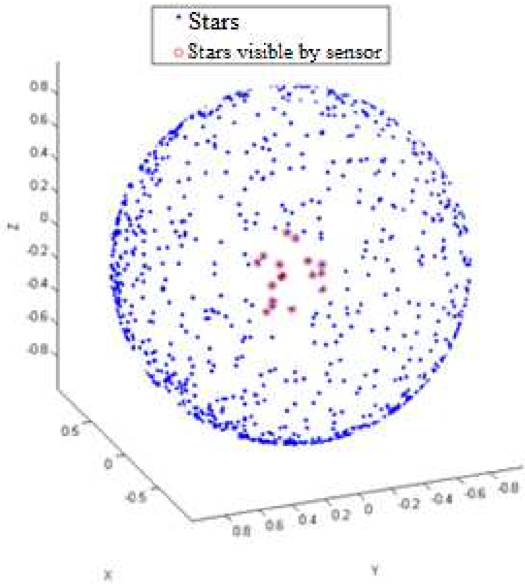

Figure 6. Stars in celestial sphere and FOV sensor view

In the matrix U_{vis}, in order to simulate intrinsic sensor errors, is included to the stars coordinates observed by the sensor FOV a Gaussian noise with mean zero and standard deviation 0,001. The same is done with the magnitude of the stars observed by the FOV sensor using a standard deviation of the 0,1. These values are compatible with commercial sensors.

To determine the attitude is necessary that the stars present in the FOV sensor are identified by the star pattern algorithm. With the stars observed by the FOV sensors and the stars identified by the algorithm in the catalogue, is possible to calculates the spacecraft attitude using an attitude determination algorithm. In this work is used singular value decomposition (SVD) (Markley, 1988) to determine the attitude.

5. STAR PATTERN IDENTIFICATION USING GEO

After the FOV simulation, the stars coordinates observed by sensor are passed on to the star pattern algorithm. The first step of the star pattern identification algorithm using GEO is to transform it in an optimization problem. Here is used the approach proposed by Paladugu et al (2006) to create the objective function. For this, is to calculated the angular distance between the stars mapped in FOV and the FOV center and these angular distances are sorted in increasing order.

The second step is to create a vector (D_a) with the angular distances of the sixteen stars which are closest the FOV center and contained in a circle entered in FOV as illustrated in fig. 7.

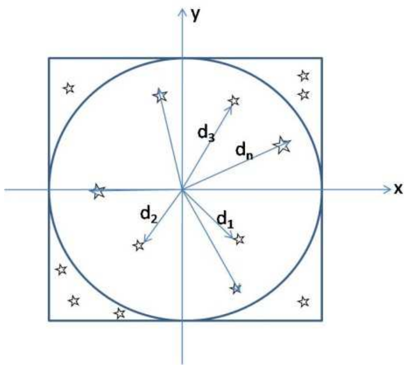

Figure 7. FOV and angular distances (d_i)

The vector (D_a) is shown in eq. (8).

$$D_a = [d_1, d_2, ..., d_n] \tag{8}$$

In the third step, the GEO algorithm creates randomly the design variables which are the right ascension (α) and declination (δ) coordinates of the center of candidate FOV. With these coordinates is created a candidate FOV which generates a new vector (D_i), these coordinates are modified using Gaussian perturbation in accordance with GEO_{real2} criteria. The vector (D_i) is shown in eq. (9) and is used to determine how close to the solution through cost function shown in eq. (10) (Paladugu et. al, 2006).

$$D_i = [d_{i1}, d_{i2}, ..., d_{in}] \tag{9}$$

Minimize:

$$C_i = \sum_{k=1}^{n} \left| D_a[d_k] - D_i[d_k] \right| \tag{10}$$

The algorithm GEO searches in the design space until the cost function (10) reaches a minimum stipulated and return the best FOV found. The minimization of the objective function (10) is made with constrains in the design variables right ascension (α) and declination (δ) as shown in the eq. (11) and eq. (12).

Subject to:

$$0^o \le \alpha \le 360^o \tag{11}$$

$$-90^o \le \delta \le 90^o \tag{12}$$

It is interesting to note that the design space defined by eq. (10) is quite complex and depends on the attitude of the spacecraft as shown in fig. 8.

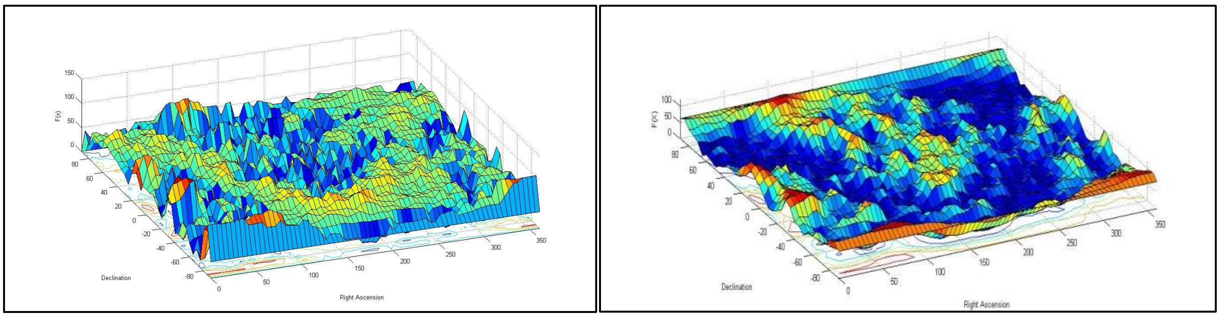

Figure 8. Design Space of Cost Function (10) for two random different attitudes

A much simplier design space can be obtained if the design variables are treated in a discreet form. To do this, a modification was proposed to make the design space discrete. The modification occurs by the fact that now the design variables are the coordinates of the star closer to the FOV center.

For this, the algorithm now stores the coordinates of the star closer to the FOV center and their magnitude and calculates the angular distance between the stars present in sensor FOV and this star closer to the center FOV as shown in fig. 9. With these angular separations the algorithm creates a vector (D_a) equal to the ones shown in eq. (8).

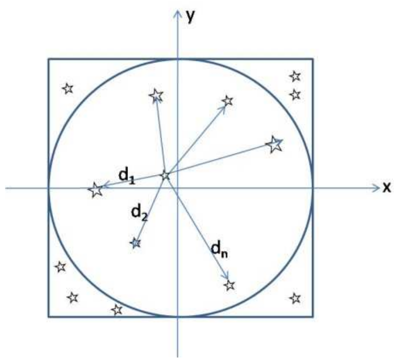

Figure 9. FOV and angular distances (d_i)

With the magnitude of the star closer to the center of FOV the algorithm creates a table with index (1 to n) containing stars with similar magnitude to this star closer to the center of FOV.

Then the algorithm generates a random number between 1 to n which is now the design variable, where (n) is the number of stars in the table, with this random number, the algorithm takes the star with this index number in the table and all the stars of the catalog with angular separation between the star selected from the table smaller than a circle of radius inscribed in the FOV as shown in fig. 9 and creates the vector (D_i) similar to exhibited in eq. (9). The algorithm then modifies this random number with Gaussian perturbation in accordance with GEO_{real2} criteria generating new index and new vectors (D_i) for calculate the cost function shown in eq. (10) until this function reaches a minimum stipulated and returns the stars found.

With this modification the design space remain simpler as shown in fig. 10 where it is illustrated the discreet design space for two random attitudes.

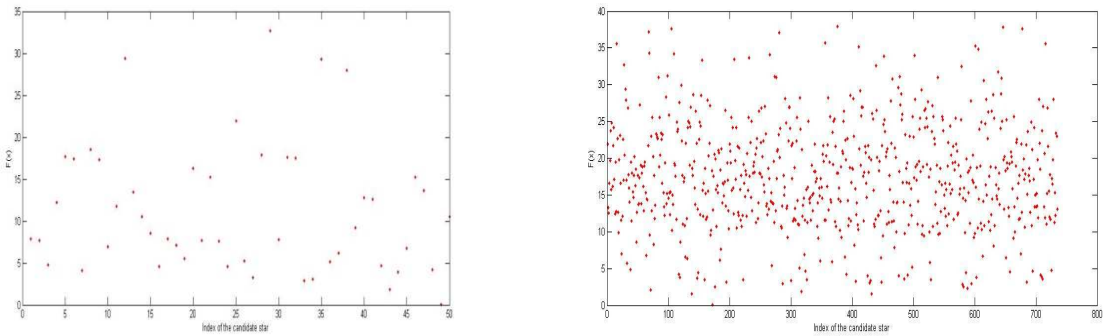

Figure 10. Discreet design space using eq. (10) for two random attitudes

6. TRIANGLE ALGORITHM

In this algorithm are selected in the FOV sensor, the three more brilliant stars wich are called alpha, beta and gama. The angular distances between them are calculated and stored.

The algorithm then creates three lists of the candidate stars, where in list one are the stars with similar brightness to the star alpha, in the list two are the stars with similar brightness to the star beta and in the list three are the stars with similar brightness to the star gama.

With these three lists are calculated the angular distances between the stars candidates until it is found three stars which have angular distances between them silmilar to the three stars selected in FOV sensor as illustrated in fig. 11.

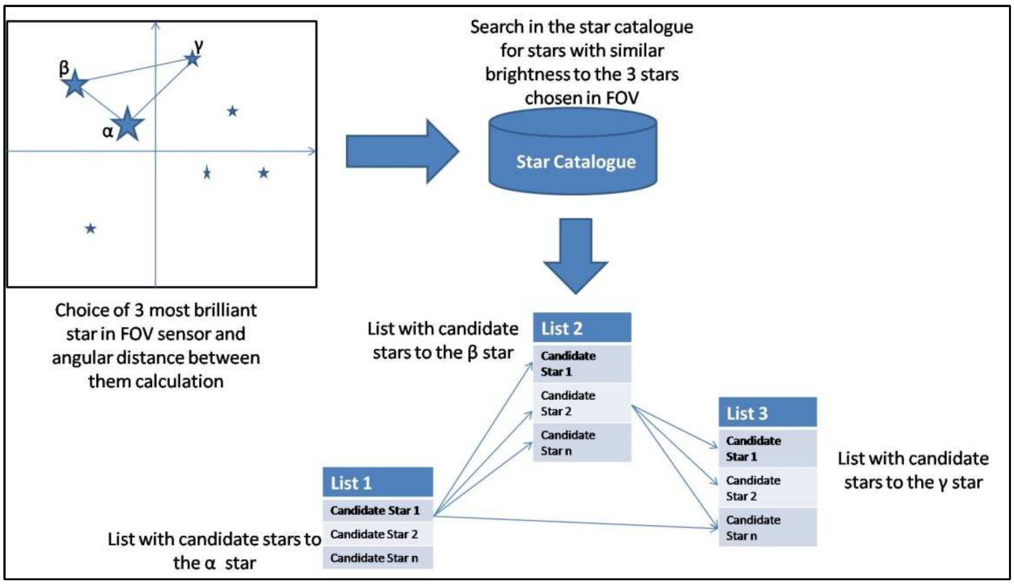

Figure 11. Triangle Algorithm

With these three identified stars is calculated an stimate to the attitude, and with this stimate attitude are confirmed the others stars present in FOV sensor by direct match (Wertz, 1978) and with the others identified stars present in FOV is calculated a new stimate attitude with more precision using SVD (Markley, 1988). The implementation of this algorithm is similar to the algorithm shown in Fialho (2007).

7. RESULTS

The tab. 2 show the results for 50 executions of the algorithms explicated in this work using random attitudes. The results show that the three algorithms are able to identified the stars presents in FOV sensor and consequently calculate correctly the attitude, however the triangle algorithm, even after the change in the design space done in the algorithm using GEO, has proved to be more faster and simple to implementations.

In both algorithms with GEO are used the value 2,75 for the parameter τ, in the GEO with continuos space are used eight standard deviations for change the design variables and five standard deviations in the design variable of GEO with discreet space.

Table 2. Results

	Algorithm Using GEO (Continuous Space)	**Algorithm Using GEO (Discreet Space)**	**Triangle Algorithm**
Mean Time for each execution (s)	34,94	2,91	0,1
Not identified	0	1	0
Standard deviation in X (°)	$2,5 . 10^{-4}$	$2,21 . 10^{-4}$	$3,77 . 10^{-4}$
Standard deviation in Y (°)	$2,39 . 10^{-4}$	$2,45 . 10^{-4}$	$2,89 . 10^{-4}$
Standard deviation in Z (°)	0.002	0,0013	0,0017

The results demonstrate that the proposed algorithm gets to accomplish the identification with success, making the sensor to calculate the space vehicle attitude correctly, however, after making a comparison with the results of a classic star pattern algorithm implementation, called triangle algorithm, it was noticed that the algorithm using GEO with continuos design space was slower, however, considering that this problem is the "lost in space" problem, i.e., the sensor was operating in the acquisition mode, the mean time for each execution shown in tab. 2 to the algorithm using GEO with continuous space is an acceptable time.

An investigation on the design space of the cost function used was accomplished according to the display on fig. 8 and it was verified that it was very complex, generating a difficulty in the search of the results. A solution proposed to improve the search time of the solution was to use a discreet space as shown on the item 5 of this work and in fig. 10.

8. CONCLUSIONS

The analysis of the results shown that the three algorithms are able to identified the stars presents in FOV sensor and consequently calculate correctly the spacecraft attitude, however, the triangle algorithm demonstrated to be faster in the search and simpler of being implemented, leading us to conclude that, from the standpoint of search time, treat this problem as an optimization problem using the objective function (10) appears to be less effective.

In the future work shall be done a study of the efficiency of these algorithms from the point of view of the failures tolerance.

9. ACKNOWLEDGEMENTS

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

10. REFERENCES

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.

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.

Fialho, M. A. A., Estudo comparativo entre dois algoritmos de identificação de estrelas para um sensor de estrelas autônomo de campo largo. 2007, Tese de mestrado – Curso de Engenharia Eletrônica e Computação. Área de Dispositivos e Sistemas Eletrônicos – Instituto Tecnológico de Aeronáutica, São José dos Campos.

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.

GALSKI, R. L. Desenvolvimento de versoes aprimoradas, hibridas, paralela e multiobjetivo do metodo da otimizacao extrema generalizada e sua aplicacao no projeto de sistemas espaciais Sao Jose dos Campos, SP: INPE, 2006

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.

McClintock, S., Lunney, T., and Hashim, A. 1998. A genetic algorithm environment for star pattern recognition. J. Intell. Fuzzy Syst. 6, 1 (Jan. 1998), 3-16.

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.

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.

Wertz, J. R., (1978), “Spacecraft Attitude Determination and Control”, London, D. Reidel.

11. RESPONSIBILITY NOTICE

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