SSC03-XI-2

Star Pattern Recognition for Attitude Determination Using Genetic Algorithms

Lalitha Paladugu
Measurement and Controls Engineering Research Center
College of Engineering
Idaho State University
Pocatello, ID 83209
paladugu_lalitha@yahoo.com

Brian G. Williams
Measurement and Controls Engineering Research Center
College of Engineering
Idaho State University
Pocatello, ID 83209
(208) 282-4129
willbria@isu.edu

Marco P. Schoen
Measurement and Controls Engineering Research Center
College of Engineering
Idaho State University
Pocatello, ID 83209
(208) 282-4377
schomarc@isu.edu

Abstract

This paper presents a study using Genetic Algorithms (GA) to solve the star pattern recognition problem associated with star tracker attitude determination systems. Characteristics of the stars that are visible within the Field of View (FOV) of an imager are defined with regard to relative distances and angles. The proposed GA minimizes the discrepancy between the characteristics of the stars inside the actual FOV and a candidate FOV selected from the star map in order to determine the inertial coordinates of the FOV bore sight. The proposed algorithm has the capability of determining the rotational angle between the spacecraft's coordinate system and that of a standardized star map. Simulations indicate that the GA approach is highly suited for this type of problem.

Introduction

Attitude, as applied to spacecraft, implies the rotational motion about its center of mass. Almost all spacecraft have attitude requirements, for example the pointing of antennas and cameras, or the orientation of solar panels.[1] Attitude control involves a two-step process: measuring the attitude and then being capable of controlling the attitude. To determine the attitude, various methods may be employed, for example sun sensors, earth

horizon sensors, magnetometers, and star sensors. Each technique has its advantages and disadvantages that will not be discussed in this paper. Star sensors (also called star trackers) are the selected method of attitude measurement for the work presented in this paper. The operation of a star tracker can be summarized as directing some image-forming device towards the stars. If the image can be matched to a reference, the direction of the imaging device is known. If the axis of the imaging device is known with respect to the spacecraft, then the attitude of the spacecraft is also known.[2] Controlling or changing the attitude can be accomplished with thrusters, momentum wheels, torque rods, to name a few, and is not discussed in this paper.

An attitude measurement sensor is typically required to operate in three different modes: (1) "lost in space", (2) maintain a predetermined attitude, and (3) track (or rotate) from one attitude to another. This paper focuses on the "lost in space" mode in which it is desired to determine from celestial observations the attitude (or in other terms, where the spacecraft is pointing). An intelligent control theory, Genetic Algorithms (GA), is used as a pattern recognition tool to match imaged star patterns against those found in a star map. Genetic Algorithms are evolutionary algorithms that rely on Darwin's concept of "survival of the fittest" to determine the optimum solution, in this case, the closest match to the star map.

This paper presents a background of Genetic Algorithms and discusses their implementation for star pattern recognition. Simulations were performed with a randomly generated star map; the GA routine would converge to a candidate location on the map and then reset the convergence criteria to allow a second level GA convergence to an even tighter location (x, y coordinates). Additionally, once convergence is achieved, the angle of rotation between the image and the star map can be determined and implies the rotation angle of the spacecraft with respect to the star patterns along the plane of the stars.

Theory of Genetic Algorithms

Optimization algorithms traditionally involve the computation of gradients and the application of the Weierstrass Theorem to determine the existence of a global minimum. If constraints are involved, Lagrangian multipliers are used along with the Kuhn-Tucker Theorem. Nonetheless, finding the global optimum point is not guaranteed. A different approach to the optimization problem, called Genetic Algorithms[3], has become more and more popular in recent years. This new algorithm does not involve any derivatives and is basically a numerical approach to the problem. Genetic Algorithms are evolutionary algorithms that simulate Darwin's survival of the fittest principle. These algorithms involve some amount of randomness in their procession through each step, which in turn ensures that with sufficient number of iterations (known as generations) the global optimum point will be found. In addition to the ability to find the global optimum point, a set of candidate solutions are available. The general outline for such an algorithm is given in Figure 1.

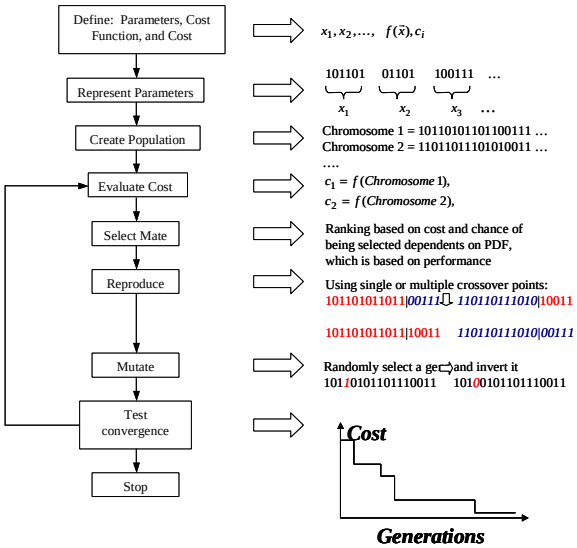

Figure 1. Outline of a Genetic Algorithm using the binary format.

First the initial population of candidate solutions is randomly generated and represented as chromosomes in the form of genes. This can be done in continuous numbers or in a binary format. Figure 1 reflects the binary representation, which is easier to be visualized. The continuous number approach is very similar to the binary approach. The encoding of the parameters into binary format is a simple conversion from a floating-point format to binary numbers or gray code number. Each parameter represents one gene and the set of parameters to describe the problem constitutes one chromosome. The generated chromosomes are evaluated based on a cost or objective function (this function is also often referred to as the fitness function) and ranked in terms of its fitness. The evaluation of each chromosome requires a translation into real numbers for each gene so that it can be computed with the given objective function. A subset of the next generation of candidate solutions is selected based on their performance with the objective function. A mating process generates the remaining sets of the new generation, where the best performing candidate solutions comprise the subset of the parents. The selection of the parents is done randomly based on the probability density function, which can be formulated based on the chromosome's performance. The mating process involves a low number of so called crossover points, where the chromosome of each parent is divided and the resulting parts recombined with other parts of the other parent chromosome. For example, for a single crossover point the parents will generate two offspring, for two crossover points, the parents will generate three offspring, etc. In addition to the mating process, a mutation rate is also imbedded in the generation of the new population. For binary representation, the mutation is given by changing the binary bits for an arbitrary small percentage of the entire collection of zeros and ones. The mutation enables the search for the optimum solution to overcome local minimums and 'jump' over constraint boundaries in the search space to locate the global minimum/optimum. This process of selection, mating, and mutation is repeated a number of times until the best performing candidate solution converges to some stationary value. Besides the capability of overcoming local minima, some additional advantages of a genetic algorithm are the ease with which large numbers of parameters can be handled, the fact that they do not require the traditional approach of taking derivatives, the fact that they result in a set of optimum candidate solutions rather than a single candidate solution, that a fairly complex system with numerous constraints can be solved, and that they work well with experimental data as well as simulated data.

Application of Genetic Algorithms to Star Pattern Recognition

A star sensor initiates its attitude determination routine by taking an image of a star pattern, bounded by the field of view (FOV) and centered along the bore-sight. It is

assumed that the FOV size is known as well as the positioning of the imaging system's bore-sight with respect to the axis of the spacecraft. Since the distance from the observer to the stars is so large, it appears as if the three-dimensional location of the stars is projected onto a two-dimensional surface, known as the celestial sphere. As such, the geometry relationships between stars can be treated as if they are all on the same plane. Unfolding the celestial sphere results in a flat surface, often represented as a rectangular map, with the north celestial pole aligned with the north rotational axis of the earth.

In solving the attitude determination problem, parameters must be computed and then compared between the imaged FOV and the candidate FOV's derived from the GA and the star map. The parameters used for this effort are distances to stars (**R**) and the angle of rotation (**θ**) from a reference to the **R** vector. The distance to each star in the field of view is referenced from the center of the FOV (bore-sight). The angle is referenced from the x-axis (x_i in Figure 2) and measured counter-clockwise to the **R** vector. The magnitude of the **R** vectors for all stars in the FOV is combined into a distance vector (**D**). **R** values are numbered sequentially, starting with the star closest to the bore-sight (smallest **R** magnitude), proceeding to the next closest, and so forth. The distance vector for the FOV imaged stars is given by

$$\mathbf{D}_a = [R_1, R_2, ..., R_n] \qquad (1)$$

where n is the number of stars visible in the FOV whose parameters are used in the GA routine. The subscript "a" refers to the actual (or imaged) values. Angles are numbered according to the sequence above for the **R** values. For example, the closest star is numbered R_1 with a corresponding angle of θ_1. The θ values are combined, in order of their numbering, into an angle vector ($\boldsymbol{\theta}_a$) and is given by

$$\boldsymbol{\theta}_a = [\theta_1, \theta_2, ..., \theta_n] \qquad (2)$$

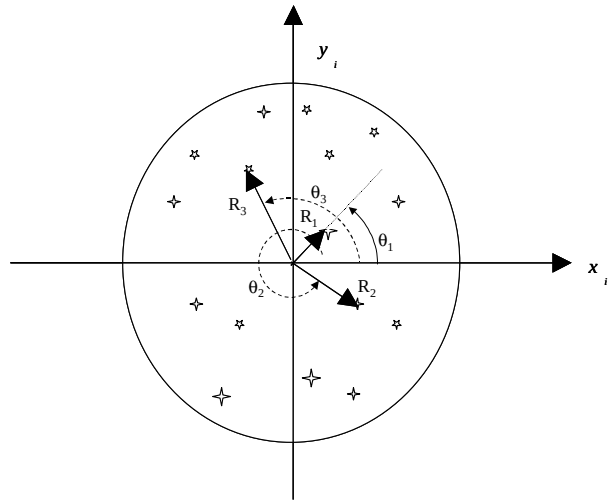

Figure 2. Schematic showing the method for calculating the distance R and angle θ.

Each chromosome generated with the GA corresponds to a candidate (x,y) location of the imager's bore-sight on the star map (x_i, y_i coordinate system in Figure 3, for the ith chromosome). Based on the method described previously, a distance vector and angle vector ($\mathbf{D}_i$ and $\boldsymbol{\theta}_i$, respectively) is computed for each chromosome. A cost function (C_i) for the ith chromosome is defined as the sum of the differences of R_k between the actual distance vector and the chromosome's distance vector:

$$C_i = \sum_k^n \left| \mathbf{D}_a[R_k] - \mathbf{D}_i[R_k] \right| \qquad (3)$$

The top chromosome is then defined as the chromosome with the minimum cost function for a given generation. Once the top chromosome's cost function is less than a predetermined convergence value (δ), then the

routine is completed and the program moves on to the next GA routine.

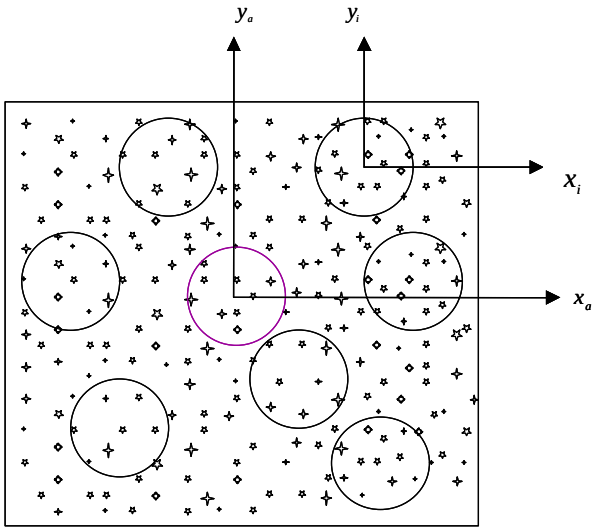

Figure 3. Star map with chromosomes (candidate x, y locations) for the GA approach.

The concept of spiral genetic algorithm (SGA) is incorporated which decreases the search area of the subsequent GA's that is proportional to the minimal cost function (δ) of the previous GA and thus decreasing the cost to provide better convergence. The next-level GA routine bounds its search, centered around the top chromosome from the previous GA routine, extending out some value Δ on all sides. The limiting value, Δ, is determined from the size of the original FOV as well as the previous GA's target minimal cost. A new minimal cost provides a convergence criterion for the new GA routine. This process continues (spirals) until a solution is found that approximates the observed location with an error that approaches zero (see Figure 4).

Once a candidate position is found, the angle of rotation (ϕ) of the spacecraft with respect to the star map can be computed using still another GA routine (see Figure 5). If the candidate position matches that imaged along the FOV's bore-sight, then there should be a distinct constant angular value, when added/subtracted from the FOV-imaged angles, that produces a perfect match to a position on the star map. If a distinct single rotation angle cannot be determined, then the process restarts with the "lost in space" GA. The equation of cost function for the angle of rotation will be

$$C_i = \sum_k^n \left| \boldsymbol{\theta}_a[\theta_k] - \boldsymbol{\theta}_i[\theta_k] \right| \qquad (4)$$

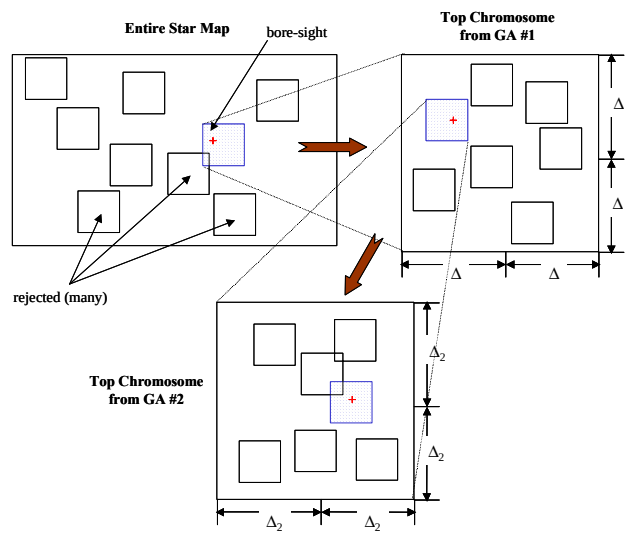

Figure 4. Concept picture of the Spiral Genetic Algorithm approach.

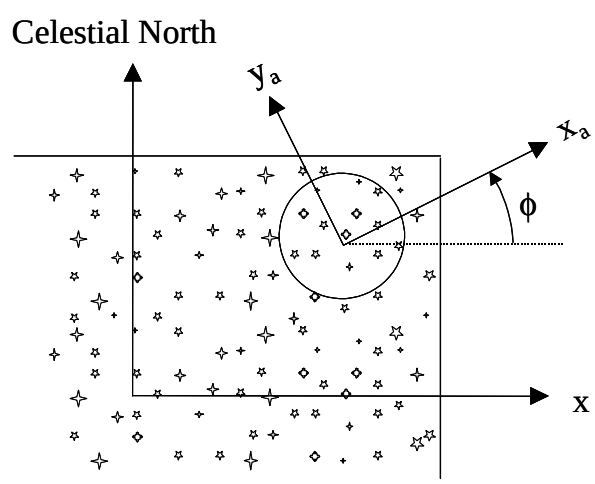

Figure 5. Picture showing the rotation angle ϕ with respect to the celestial sphere coordinates.

Simulation Results

Simulations are carried out using two GA's for finding the location using distance and one GA for calculating the angle of rotation (ϕ) after the convergence is met using the distance values. For the first GA using distances, a simulated star map was generated for 1000 randomly placed stars. The dimensions of the overall star map are 300 units by 300 units (-150 to +150) and 25 by 25 units for each individual FOV. The number of stars used in the FOV (typically m = 15 to 20) for a particular simulation is kept constant. If the FOV has fewer stars, then zeros are added to the beginning of the distance vector for a total of m stars (actual plus added). If the FOV has more stars, then only the m closest stars to the center are included. A total of 300 initial locations for candidate FOVs were selected randomly, of which 150 were retained as steady state population for each generation. Of the 150 candidate FOVs, 80 are selected based on their performance measured by the cost function to survive into the new generation. 80 new FOVs were generated using the pairing, mating, and mutation algorithm described in the Theory section. The mutation rate was set to 6%. None of these parameters were optimized, which could potentially increase the efficiency of the proposed algorithm dramatically. Such an optimization is based on the distribution of the stars in the star map and its resulting sensitivities based on the size of the candidate FOV and the proposed fitness function (cost function). If the cost function does not converge to the expected value, a new set of chromosomes is taken and the algorithm is run from the beginning with the new chromosomes.

After the cost is converged to a value which is less than or equal to the cost required in the first GA, a second GA that has a total of 100 initial locations of candidate FOVs were selected randomly in the region around the point where the first GA is converged with some error Δ around the point of convergence. From these 100 initial locations 50 were retained as steady state population for each generation, of which 30 are selected based on their performance measured by the cost function. The mutation rate remained 6%.

For the GA to find the angles, a total of 120 initial locations of the angles (degrees) were selected randomly, of which 60 were retained and 30 are selected based on their performance, the mutation rate being the same.

Multiple simulations using the GA approach were conducted in order to statistically describe the potential accuracy of the proposed algorithm. Results were recorded of simulation runs that took about 600 generations or less for conversion of the minimum cost. The conversion was determined when the top chromosomes cost was less than 0.001882 for situations when the stars in the FOV varied from 15 to 20. Since no optimization of the selected parameters for the GA is incorporated in the present study, the determination of the accuracy potential is unaffected by the above mentioned selection criteria. The actual (imaged) location of the bore-sight was selected as the origin of the star map (x = 0, y = 0). Simulations resulted in a deviation not more than 0.0012% in the *x*-coordinate and 0.000062% in the *y*-coordinate locations (x_{mean} = -0.0018544 with a standard deviation of x_{std} = 0.0020563 and y_{mean} = -0.00009245 with a standard deviation of y_{std} = 0.000657 units).

As an example, consider the results of one such simulation. The location of the FOV's bore-sight was x = 0, y = 0 with an angle of rotation of 60° (1.0472 radians). The

algorithm used 20 stars located within the FOV.

Step 1: The first GA requires a minimum cost of less than 3.0 using the distance values (see Figure 6). The top chromosome of the first GA is x = -0.0363 and y = 0.2010. The resulting cost function is 2.1825.

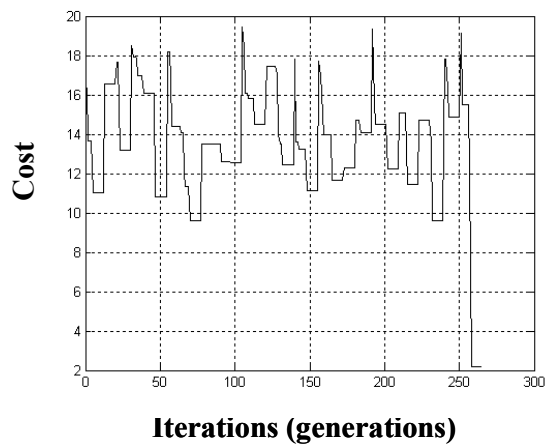

Figure 6. Plot of cost function versus number of iterations for the first GA.

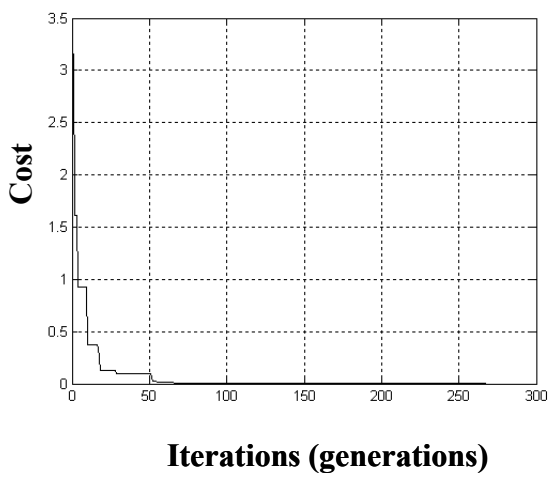

Figure 7. Plot of cost function versus number of iterations for the second GA.

Step 2: A second GA requires a minimum cost of less than 1.0 using the distance values (see Figure 7). The search area for potential candidate locations (chromosomes) is decreased to a square whose center coincides with the location of the top chromosome from the previous GA (x = -0.0363 and y = 0.2010). The size of the search box is equal to a +/-Δ = 2.40075 units; this corresponds to a high-end x value of 2.36445 units and a low-end x value of –2.43705 units. For the y, the high-end value is 2.60175 units and the low-end value is –2.19975 units. From this second GA, a top chromosome of x = –0.0059 and y = –0.0031 is found, with a cost function of 0.0065.

Step 3: Since convergence to the x, y value was achieved, another GA for finding the rotational angle, ϕ, was used. The range of chromosomes considered varies from 0° to 360°. The top chromosome for this GA is 60.0001° (corresponding to 1.0472 radians), which is approximately the same as the actual rotation angle. The minimum cost for this GA is 0.0003712 (see Figure 8).

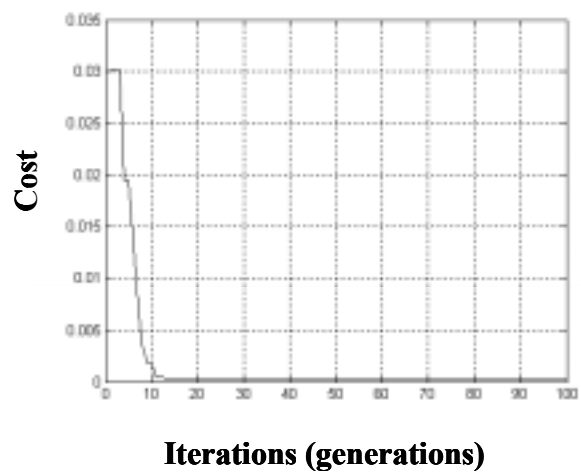

Figure 8. Plot of cost function versus number of iterations for the rotation angle GA.

Discussion of Simulation Results

It is assumed that the 300-unit wide star map represents a full 360° view. For the simulation example described previously, this

results in an inaccuracy of 0.00708° for the x direction and 0.00372° for the y. This simulation example is almost a worse case scenario; both x_{mean} and y_{mean} are closer to zero than x and y from the example. Using the mean values, the inaccuracy of two GA routines is 0.00223° for the x direction and 0.000111° for the y. As a comparison, the EMS Technologies CALTRAC™ Star Tracker has a noise angle of ± 0.005° in the pitch/yaw direction[4]. To increase the accuracy of the SGA technique, additional GA could be used with successively decreasing search areas and cost functions.

The application of GA for star pattern recognition can still be improved by optimizing the parameters so that the position is more accurately known. The numbers of stars in the star catalog is directly proportional to the cost function. As the number of stars increases, the cost function increases. With a greater number of stars there is a better possibility of finding the exact, distinct position of the image. Implementing the present algorithm on a real star catalog will be done in later work. For the present work, x and y coordinates were used. For an actual star catalog, the x,y values would need to be converted into Right Ascension and Declination. Besides this minor transformation of coordinates, no additional work would need to be done to convert to the typical stellar catalog values.

During simulations, it was noted that the time needed for the algorithm to converge is considerable. Several reasons for this that are inherent to the system used are a single, PC-technology microprocessor as well as running all code in MatLab (usually slower than running pure complied code). One method to increase computational speeds for the lost in space operation would be to involve parallel processors. Currently, a single processor (the desk-top computer processor) calculates the parameters of each chromosome (candidate FOV) in a sequential form, and then sequentially compares them to the actual (imaged) parameters. Multiple parallel processors would allow each individual processor to compute the candidate's parameters and cost function; if ten parallel processors are used, theoretically this would decrease the required computational period by ten times. A single processor could then sort the cost values to determine the top chromosome for the iteration. The advantage of the parallel processing system would be simultaneous computation of redundant multiple calculations. Disadvantages could include an increase in hardware complexity, cost, and power consumption.

Another option is to use the brightness (or magnitude rating) of a star as a pre-computation filter. In choosing GA-selected candidate positions, a comparison is first made between the magnitude of the star closest to the bore-sight and the star closest to the GA-selected position. If the candidate's magnitude is not within a given value (for example, ± 1) of the star nearest the bore-sight, then there is no need to compute the distance vector, $\mathbf{D_i}$, and hence a cost function. Distance vectors and cost functions are ***only*** computed when the candidate star's magnitude is within the required magnitude of the bore-sight star. The ± 1 in magnitude accounts for any variations between star brightness measured with photometric techniques in the star tracker and those recorded in star catalogs.

Conclusions

The efforts presented in this paper successfully demonstrate the ability of spiraling Genetic Algorithms to perform the pattern-matching computations associated with the "lost in space" mode of a star sensor, using a randomly generated star map.

Multiple simulation results indicate an angular accuracy of 0.00223° for the x direction and 0.000111° for the y, which is within the current yaw/pitch accuracies of commercial systems. To further increase the accuracy of the Genetic Algorithm approach, successive (spiraling) routines can be added with decreasing search pattern size and convergence criteria (cost function). Additionally, the Genetic Algorithm approach demonstrated the ability to successfully determine the "roll" rotation angle of the imaged pattern with respect to the celestial sphere's poles.

Future efforts will concentrate on the use of an actual star catalog with a Right Ascension and Declination coordinate system. Efforts will also be made to optimize the Genetic Algorithm parameters as well as decrease the required computational time.

Acknowledgements

This work was partially funded by an ISU infrastructure grant. The support is greatly appreciated.

References

1. Wiesel, William E., Spaceflight Dynamics, McGraw-Hill Book Company, New York, 1989.

2. Liebe, C. C., "Pattern Recognition of Star Constellations for Spacecraft Applications," IEEE Aerospace and Electronic Systems Magazine, vol. 8, No. 1, Jan 1993.

3. Haupt R.L., and Haupt S.E., Practical Genetic Algorithms, John Wiley and Sons, Inc., New York, 1998.

4. EMS Technologies, Inc., "EMS Space & Technology – Commercial Space Products – Optical: CALTRAC™ Star Tracker," available from http://www.emsstg.com/science/optical.asp, accessed 10 June 2003.