JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 29, No. 1, January–February 2006

Fast Star-Pattern Recognition Using Planar Triangles

Craig L. Cole*
Orbital Sciences Corporation, Dulles, Virginia 20166
and
John L. Crassidis†
University at Buffalo, State University of New York, Amherst, New York 14260-4400

A new method for star identification using planar triangles is developed and compared to a standard angle method approach. The angle method creates angles between stars within the field of view of a star tracker and matches them to angles in a catalog to determine which stars they are. The new method creates triangles from the stars within the field of view of a star tracker, determines their area and polar moment, then matches the triangles to a catalog of triangles in a way that is faster and more successful than the angle method. Simulation results are provided that show the performance of the new method as well as its robustness with respect to false stars.

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

Introduction

Star Identification Methods

A VERY IMPORTANT device on any spacecraft is that which determines the attitude of a spacecraft. There are many different methods by which attitude can be determined, but one of the mostly widely used is the star tracker. Other methods for determining attitude exist, including sun sensors and magnetometers, but these cannot report attitude with the precision of a star tracker. Often, on high-budget missions, these other sensors will complement or back up a star tracker. On missions with tight attitude knowledge requirements, the primary means to determine attitude is the star tracker. Before an attitude can be determined, the stars within the field of view (FOV) of a star tracker must be identified. If it can identify at least two stars, then the attitude of the spacecraft can be determined.

The technology behind star trackers has changed much over the years. Evolving from gimballed to direct head, the latest star trackers now use charge-coupled devices (CCDs) for imaging, which offer high accuracy.[1] Still, they are not perfect devices because measurement errors will always exist. Even the smallest amount of error in the image measurement of the stars makes identifying the stars very challenging. As a result, there are many different methods for identifying the stars within the FOV. One well-known method involves measuring the angular separation between stars in the FOV and then matching them to angles within a catalog of angles between stars.[2] This is called the "angle method" and has been the basis for many star-recognition algorithms. It does not rely on other sensors for assistance and can be used in a "lost-in-space" condition. Star magnitude can also be used to help in identification, but this is not simple when CCDs are used. For instance, the sensitivity of a CCD to different spectra of light must be understood and accounted for.[3]

Other methods use a priori information to help the star tracker determine the stars within its FOV.[2,4–9] Methods based on knowing the attitude of the spacecraft at an earlier point in time would not be suitable for lost-in-space conditions. Other methods are completely attitude independent. For example, Mortari et al.[10] match star quadruples by ordering angular observations to form an index. The method works well, but no statistical analysis is given on the errors in the quadruple measurements. Even neural networks are being tested for their potential to recognize stars.[11] Another approach uses a grid over an image, and centers and aligns a grid according to two stars in the FOV.[12] A binary grid is created by marking each grid bin as either containing a star or not containing a star. A probabilistic model is used to adapt the matching function according to the density of stars in the FOV. The approach is conceptually quite simple, which provides very accurate star identification to both positional and magnitude noise. However, at least 10 stars are required in the FOV and a minimum of 7 stars need to be matched for proper identification. The approach presented in this paper can work using a much smaller number of stars and, unlike the methods of Refs. 2 and 4–9, requires no a priori information. Furthermore, unlike in Ref. 10, a rigorous probabilistic analysis is shown on the new observations derived herein.

The method presented here is called the "planar triangle method." It creates planar triangles from the stars in the FOV and compares them to a catalog of triangles to find a match. It too has its roots in the angle method, which is used as a basis for comparative studies with the planar triangle method. To make the testing of both methods realistic, a typical star tracker has been chosen as a model. The Ball CT-601 star tracker has an 8×8–deg FOV and can see down to magnitude 6.0 stars.[4,13] The testing model created here assumes an 8-deg circular FOV for simplicity, but the star-pattern-recognition algorithms tested can be applied to square FOVs.

Design Theory

The importance of star trackers on spacecraft is difficult to overestimate. Errors in attitude knowledge can result in damage to or the complete loss of a spacecraft. An ideal star tracker would be able to report attitude instantly without chance of error in lost-in-space conditions and without aid from other sensors. In addition, it would require little in the way of computer resources such as storage and CPU speed. These are certainly conflicting constraints, and decisions have to be made when designing the star tracker as to what is most important.

For the methods presented here, rate of success and speed are given the highest priorities in the design of the algorithms. The ability of a star tracker to report attitude quickly is desirable in many situations, for example, to point toward the oncoming sun for power requirements before the sun is even available. The idea behind the planar triangle method is that by using more than one property to recognize a pattern of stars (area and polar moment), it is more likely to reach the correct solution and do it using fewer stars. By comparison, the angle method can only use one property, the angle itself, for pattern matching and as a result will not approach a solution as quickly.

Received 8 September 2004; revision received 15 July 2005; accepted for publication 18 July 2005. Copyright © 2005 by Craig L. Cole and John L. Crassidis. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/06 $10.00 in correspondence with the CCC.

*ACS Engineer, Space Systems Group; Cole.Craig@orbital.com. Member AIAA.

†Associate Professor, Department of Mechanical & Aerospace Engineering; johnc@eng.buffalo.edu. Associate Fellow AIAA.

The penalty for success and speed is storage. The database used with the planar triangle method is quite large when compared to that required for the angle method. It is only large by spacecraft standards, however; the database would easily fit onto a keychain-sized flash drive. The computer hardware used in spacecraft is designed to withstand the harsh environment of space and as a result lags years behind the current state of the art in processor speed and storage. The technology continues to evolve, however, and tasks that are too demanding of spacecraft hardware today will likely be practical in the not-too-distant future.

Angle Method

A well-known identification method involves matching the angle of separation between pairs of stars within the FOV to a catalog containing all of the angles between stars in the celestial sphere that can fit within the FOV of a star tracker.[2] To create the catalog of angles, a data structure called a spherical quad-tree is used.[14] A quad-tree structure is typically used to store objects located in two-dimensional space in such a way that objects can be found within a certain area without examining each and every object. The spherical quad-tree used here to catalog the angles enables cataloging angles in such a way that only each neighboring star within a certain distance has to be examined to see if the angle between them is small enough to fit within the star tracker FOV. It greatly reduces the total number of pairs that needs to be examined and greatly reduces the time required to create the catalog. For the star tracker modeled here, the catalog created contains 106,308 angles and occupies 12 MB of memory.

Once the angles are cataloged, they are sorted by angle. To make finding all the angles that lie within a given range fast, a technique called the "k-vector" approach is used.[15] If the angle of each pair of stars is plotted against its location in the catalog, a line can be drawn connecting the first and last pair of stars. The equation of this line can be used in association with the generated k-vector to locate where in the catalog a particular pair of stars with a given angle is located. This greatly reduces the computational burden because the star-pattern search algorithm requires a search of the pairs of stars only within a measurement uncertainty region, not a search of the entire catalog.

Angle Between Two Stars

The angle between the vectors pointing to the stars is given by $\theta = \cos^{-1}(\boldsymbol{r}_1 \cdot \boldsymbol{r}_2)$ where $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are unit vectors pointing to each star. The vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are given in inertial space. However, only body measurements are known. The angle θ is the same whether inertial vectors or body vectors are used. The problem is that the angle measured between the stars within the FOV of the star tracker will contain a certain amount of measurement error, which cannot be ignored. If the measurement follows a Gaussian distribution, standard deviation can be determined and used to establish a range within which the true measurement is likely to lie. For instance, if the range is chosen to be the measurement angle ± 3 times the standard deviation of the measurement noise, 3σ, then the true measurement is expected to be within this range 99.7% of the time.

Standard Deviation of Angle Measurement

The measurement error made by a typical modern-day star tracker follows a nearly zero-mean Gaussian white-noise process.[16] What will be needed for the angle method is the standard deviation of the angle between two stars when each star measurement possesses the error described. The standard deviation of the attitude-independent measurement, involving the dot product of two star vectors, can be used to provide a bound on the expected errors. We begin with the standard coordinate transformation equation: $\boldsymbol{b}_i = A\boldsymbol{r}_i$, where $\boldsymbol{r}_i$ is the direction of the star in the earth-centered inertial (ECI) coordinate system, A is the direction cosine matrix, which is orthogonal and proper, and $\boldsymbol{b}_i$ is the direction of the star in the star-tracker body-coordinate system. When measurement errors exist, Shuster[17] has shown that nearly all of the probability of the errors is concentrated on a very small area about the direction of $A\boldsymbol{r}_i$, so that the sphere containing that point can be approximated by a tangent plane, characterized by

$$\tilde{\boldsymbol{b}}_i = A\boldsymbol{r}_i + \boldsymbol{v}_i, \qquad \boldsymbol{v}_i^T A\boldsymbol{r}_i = 0 \tag{1}$$

where $\tilde{\boldsymbol{b}}_i$ denotes the ith measurement and the sensor error $\boldsymbol{v}_i$ is approximately Gaussian, which satisfies

$$E\{\boldsymbol{v}_i\} = \boldsymbol{0} \tag{2a}$$

$$R_i \equiv E\left\{\boldsymbol{v}_i \boldsymbol{v}_i^T\right\} = \sigma_i^2\left[I - (A\boldsymbol{r}_i)(A\boldsymbol{r}_i)^T\right] \tag{2b}$$

where σ_i^2 is the variance and E denotes expectation. Further details on this model are given in Ref. 18.

Taking the dot product of two body observations gives $\boldsymbol{b}_1^T\boldsymbol{b}_2 = \boldsymbol{r}_1^T A^T A\boldsymbol{r}_2 = \boldsymbol{r}_1^T\boldsymbol{r}_2$. This shows that the dot product is an attitude-invariant measurement. Consider two body measurements, denoted by $\tilde{\boldsymbol{b}}_1$ and $\tilde{\boldsymbol{b}}_2$, with noise:

$$\tilde{\boldsymbol{b}}_i = A\boldsymbol{r}_i + \boldsymbol{v}_i, \qquad i = 1, 2 \tag{3}$$

where $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are assumed uncorrelated. Define the following effective measurement:

$$\begin{aligned} y &\equiv \tilde{\boldsymbol{b}}_1^T\tilde{\boldsymbol{b}}_2 \\ &= \boldsymbol{r}_1^T\boldsymbol{r}_2 + \boldsymbol{r}_1^T A^T\boldsymbol{v}_2 + \boldsymbol{r}_2^T A^T\boldsymbol{v}_1 + \boldsymbol{v}_1^T\boldsymbol{v}_2 \end{aligned} \tag{4}$$

Because $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are uncorrelated, $E\{y\} = \boldsymbol{r}_1^T\boldsymbol{r}_2$. Define the following variable:

$$\begin{aligned} p &\equiv y - E\{y\} \\ &= \boldsymbol{r}_1^T A^T\boldsymbol{v}_2 + \boldsymbol{r}_2^T A^T\boldsymbol{v}_1 + \boldsymbol{v}_1^T\boldsymbol{v}_2 \end{aligned} \tag{5}$$

Then taking $E\{p^2\}$ yields the following variance for the dot-product measurement error:

$$\begin{aligned} \sigma_p^2 &\equiv E\{p^2\} \\ &= \boldsymbol{r}_1^T A^T R_2 A\boldsymbol{r}_1 + \boldsymbol{r}_2^T A^T R_1 A\boldsymbol{r}_2 + \mathrm{Trace}(R_1 R_2) \\ &= \mathrm{Trace}\left(A\boldsymbol{r}_1\boldsymbol{r}_1^T A^T R_2\right) + \mathrm{Trace}\left(A\boldsymbol{r}_2\boldsymbol{r}_2^T A^T R_1\right) + \mathrm{Trace}(R_1 R_2) \end{aligned} \tag{6}$$

where $E\{\boldsymbol{v}_1\boldsymbol{v}_1^T\} \equiv R_1$ and $E\{\boldsymbol{v}_2\boldsymbol{v}_2^T\} \equiv R_2$, each given by Eq. (2b). Note that the effective measurement noise in the dot product contains both Gaussian and χ^2 components. But because the noise is small compared to the unit vector observation, then using the same analogy shown in Ref. 19, the effective noise can essentially be treated as purely Gaussian. Because we do not know the attitude, $A\boldsymbol{r}_1$ and $A\boldsymbol{r}_2$ are replaced by $\tilde{\boldsymbol{b}}_1$ and $\tilde{\boldsymbol{b}}_2$, respectively. Typically, errors introduced by this substitution are second-order in nature,[17] so the variance is completely independent of the attitude matrix A.

The chosen $\beta\sigma_p$ bound (for some scalar β) has a large effect on the results of both the angle and planar triangle methods. If the correct solution lies outside the bounds of the measurement error, neither method will find the solution or, worse, arrive at an incorrect solution. If the $\beta\sigma_p$ bound is set high, it is less likely an incorrect solution will be reached, because it is more likely the correct solution lies within the $\beta\sigma$ bound. However, if the $\beta\sigma_p$ bound is set too high, too many possible solutions will exist for each angle and planar triangle in the FOV and a single solution may not be reached. Both methods are demonstrated using a $3\sigma_p$ bound, meaning the probability that the correct angle will be within the range of the measured angle will be 99.7%.

Angle Pivot

If there exists more than one possible solution to a measured angle, one method by which the correct solution can be determined is based on "pivoting." After all the possible solutions to the first angle are determined, a second angle within the FOV is selected such that it shares one star in common with the first angle. Once all

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

the potential solutions to the second angle are found; the stars that make up both angles are examined. Because the solution to both angles must have a star in common, any angles on either list that do not have a star in common with at least one angle on the other list are rejected. If the number of possible solutions for each measured angle is not reduced to one after the elimination, another pivot is made; a third angle is chosen such that it has at least one star in common with the second angle, and the possible solutions for the second and third angles that do not share a common star are rejected. The pivoting process continues until a single solution is reached or the star tracker runs out of angles to which it can pivot. If it runs out of angles before obtaining a single solution, then the result is inconclusive.

Planar Triangle Method

Instead of measuring the angle between pairs of stars, the planar triangle method creates planar triangles of combinations of three stars. The idea is that more information can be obtained from a planar triangle than an angle, which will enable a star tracker to determine the identity of stars more quickly and use fewer stars overall than the angle method. In the algorithm presented here, the area and polar moment of triangles are used to determine what triangle is being examined by the star tracker. One drawback of the this method is that it will be impossible to identify stars with less than three stars in the FOV, whereas the angle method needs only two stars. However, after including the measurement error present in the star tracker, usually more than two stars are required to arrive at a solution using the angle method. The planar triangle method, in fact, will require fewer pivots than the angle method and will more likely yield a solution using fewer stars overall.

Although the angle catalog can be created using more straightforward methods than the spherical quad-tree, to create the planar triangle catalog without it would have been difficult. Having to examine stars in combinations of three instead of two greatly increases the total number of objects to examine. The catalog suiting the star tracker modeled here contains 662,779 planar triangles. For each of these triangles, the area and polar moment must be calculated, making the catalog 167 MB in size, more than ten times larger than the equivalent angle catalog.

Like the angle method, the triangles must be sorted by area and polar moment so that the k-vector approach can be used to locate planar triangles quickly by their area or polar moment. So that two separate catalogs are not necessary, two linked-list data structures are sorted instead of the planar triangles themselves. In addition, plotting each triangle area against position in the sorted list is not linear. To make the k-vector approach as fast as possible, a parabola is fit between the first and last points (more details can be found in Ref. 14).

Planar Triangle Area

Given three unit vectors pointing toward three stars, denoted by $\tilde{\boldsymbol{b}}_1, \tilde{\boldsymbol{b}}_2$, and $\tilde{\boldsymbol{b}}_3$, the area of a planar triangle can be found using Heron's formula:

$$\mathcal{A} = \sqrt{s(s-a)(s-b)(s-c)} \tag{7}$$

where

$$s = \tfrac{1}{2}(a+b+c) \tag{8a}$$

$$a = \|\tilde{\boldsymbol{b}}_1 - \tilde{\boldsymbol{b}}_2\| \tag{8b}$$

$$b = \|\tilde{\boldsymbol{b}}_2 - \tilde{\boldsymbol{b}}_3\| \tag{8c}$$

$$c = \|\tilde{\boldsymbol{b}}_1 - \tilde{\boldsymbol{b}}_3\| \tag{8d}$$

The equation is given for the star-tracker frame but can also be used in the ECI frame. To obtain a bound for the measurement error, the standard deviation of the calculated area must be calculated.

Because the planar area is a nonlinear function of $\tilde{\boldsymbol{b}}_1, \tilde{\boldsymbol{b}}_2$, and $\tilde{\boldsymbol{b}}_3$, a linearization approach must be used to determine its variance. To compute this quantity the following 1×9 partial derivative matrix is evaluated:

$$H = \begin{bmatrix} \boldsymbol{h}_1^T & \boldsymbol{h}_2^T & \boldsymbol{h}_3^T \end{bmatrix} \tag{9}$$

where

$$\boldsymbol{h}_1^T \equiv \frac{\partial \mathcal{A}}{\partial a}\frac{\partial a}{\partial \boldsymbol{b}_1} + \frac{\partial \mathcal{A}}{\partial c}\frac{\partial c}{\partial \boldsymbol{b}_1} \tag{10a}$$

$$\boldsymbol{h}_2^T \equiv \frac{\partial \mathcal{A}}{\partial a}\frac{\partial a}{\partial \boldsymbol{b}_2} + \frac{\partial \mathcal{A}}{\partial b}\frac{\partial b}{\partial \boldsymbol{b}_2} \tag{10b}$$

$$\boldsymbol{h}_3^T \equiv \frac{\partial \mathcal{A}}{\partial b}\frac{\partial b}{\partial \boldsymbol{b}_3} + \frac{\partial \mathcal{A}}{\partial c}\frac{\partial c}{\partial \boldsymbol{b}_3} \tag{10c}$$

The partials with respect to a, b, and c are given by

$$\frac{\partial \mathcal{A}}{\partial a} = \frac{u_1 - u_2 + u_3 + u_4}{4\mathcal{A}} \tag{11a}$$

$$\frac{\partial \mathcal{A}}{\partial b} = \frac{u_1 + u_2 - u_3 + u_4}{4\mathcal{A}} \tag{11b}$$

$$\frac{\partial \mathcal{A}}{\partial c} = \frac{u_1 + u_2 + u_3 - u_4}{4\mathcal{A}} \tag{11c}$$

where

$$u_1 = (s-a)(s-b)(s-c) \tag{12a}$$

$$u_2 = s(s-b)(s-c) \tag{12b}$$

$$u_3 = s(s-a)(s-c) \tag{12c}$$

$$u_4 = s(s-a)(s-b) \tag{12d}$$

The partials with respect to $\boldsymbol{b}_1, \boldsymbol{b}_2$, and $\boldsymbol{b}_3$ are given by

$$\frac{\partial a}{\partial \boldsymbol{b}_1} = \frac{(\boldsymbol{b}_1 - \boldsymbol{b}_2)^T}{a}, \qquad \frac{\partial a}{\partial \boldsymbol{b}_2} = -\frac{\partial a}{\partial \boldsymbol{b}_1} \tag{13a}$$

$$\frac{\partial b}{\partial \boldsymbol{b}_2} = \frac{(\boldsymbol{b}_2 - \boldsymbol{b}_3)^T}{b}, \qquad \frac{\partial b}{\partial \boldsymbol{b}_3} = -\frac{\partial b}{\partial \boldsymbol{b}_2} \tag{13b}$$

$$\frac{\partial c}{\partial \boldsymbol{b}_1} = \frac{(\boldsymbol{b}_1 - \boldsymbol{b}_3)^T}{c}, \qquad \frac{\partial c}{\partial \boldsymbol{b}_3} = -\frac{\partial c}{\partial \boldsymbol{b}_1} \tag{13c}$$

The variance of the area, denoted by $\sigma_{\mathcal{A}}^2$, is given by

$$\sigma_{\mathcal{A}}^2 = HRH^T \tag{14}$$

where

$$R \equiv \begin{bmatrix} R_1 & 0_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & R_2 & 0_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} & R_3 \end{bmatrix} \tag{15}$$

where $0_{3\times 3}$ denotes a 3×3 matrix of zeros and R_1, R_2, and R_3 are given by Eq. (2b). Note that the matrices H and R are evaluated at the respective true values; however, replacing the true values with the measured ones leads to second-order errors that are negligible for typical star-tracker noise levels. Because the standard deviation $\sigma_{\mathcal{A}}$ is derived analytically, the bounds over which the true area is likely to exist can be determined precisely, no matter the shape or size of the planar triangle.

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

Planar Triangle Polar Moment

The polar moment makes a good counterpart to area, because it is possible for two planar triangles that have the same area to have very different second moments. In addition, two planar triangles that have the same polar moments may have very different areas. When it comes time to match planar triangles seen within the FOV of the star tracker to planar triangles in the catalog, use of these two approaches will rapidly reduce the number of possible solutions. The polar moment for a planar triangle is given by

$$\mathcal{J} = \mathcal{A}(a^2 + b^2 + c^2)/36 \tag{16}$$

As with the area, the variance of the polar moment can also be derived in closed form. To compute this quantity the following 1×9 partial derivative matrix is evaluated:

$$\bar{H} = \begin{bmatrix} \bar{\boldsymbol{h}}_1^T & \bar{\boldsymbol{h}}_2^T & \bar{\boldsymbol{h}}_3^T \end{bmatrix} \tag{17}$$

where

$$\bar{\boldsymbol{h}}_1^T \equiv \frac{\partial \mathcal{J}}{\partial a}\frac{\partial a}{\partial \boldsymbol{b}_1} + \frac{\partial \mathcal{J}}{\partial c}\frac{\partial c}{\partial \boldsymbol{b}_1} + \frac{\partial \mathcal{J}}{\partial \mathcal{A}}\boldsymbol{h}_1^T \tag{18a}$$

$$\bar{\boldsymbol{h}}_2^T \equiv \frac{\partial \mathcal{J}}{\partial a}\frac{\partial a}{\partial \boldsymbol{b}_2} + \frac{\partial \mathcal{J}}{\partial b}\frac{\partial b}{\partial \boldsymbol{b}_2} + \frac{\partial \mathcal{J}}{\partial \mathcal{A}}\boldsymbol{h}_2^T \tag{18b}$$

$$\bar{\boldsymbol{h}}_3^T \equiv \frac{\partial \mathcal{J}}{\partial b}\frac{\partial b}{\partial \boldsymbol{b}_3} + \frac{\partial \mathcal{J}}{\partial c}\frac{\partial c}{\partial \boldsymbol{b}_3} + \frac{\partial \mathcal{J}}{\partial \mathcal{A}}\boldsymbol{h}_3^T \tag{18c}$$

with

$$\frac{\partial \mathcal{J}}{\partial a} = \frac{\mathcal{A}a}{18}, \qquad \frac{\partial \mathcal{J}}{\partial a} = \frac{\mathcal{A}b}{18}, \qquad \frac{\partial \mathcal{J}}{\partial a} = \frac{\mathcal{A}c}{18} \tag{19a}$$

$$\frac{\partial \mathcal{J}}{\partial \mathcal{A}} = \frac{(a^2 + b^2 + c^2)}{36} \tag{19b}$$

All other quantities in Eq. (18) are given from the area-variance calculations. The variance of the polar moment, denoted by $\sigma_{\mathcal{J}}^2$, is given by

$$\sigma_{\mathcal{J}}^2 = \bar{H}R\bar{H}^T \tag{20}$$

As with the area variance, the true values are replaced with their respective measured ones.

Planar Triangle Pivot

The method for matching planar triangles is similar to the method for matching angles. A planar triangle is made from three stars in the FOV and its area and polar moment are calculated. A range over which the true area and polar moment exist are calculated using the standard deviations for each. Triangles that have an area and polar moment that fit within the bounds calculated for the triangle in the FOV are sought. Ideally, only one possible solution exists, but this typically does not happen. When more than one solution exists, a pivot similar to the angle method is made. Another planar triangle is made from the stars in the FOV such that there are two stars in common with the first triangle, as shown in Fig. 1. A list of possible solutions is made and then the solutions between the first planar triangle and second planar triangle are compared. Any solution in each list that does not have two stars in common with at least one solution in the other lists is discarded. After the comparison is made, if more than one solution exists, another pivot is made. Pivoting continues until either a single solution is found or there are no more triangles to which to pivot. Pivoting such that only one star is shared between the first and second planar triangles can be done but would be less effective. The number of triangles that are likely to share one star is greater than two, so the solution would require a greater number of pivots.

Testing each possible solution to the first triangle against each possible solution to the pivot triangle can be time consuming, because there can be hundreds of solutions per triangle in the FOV. To reduce the number of combinations to test, three binary search trees

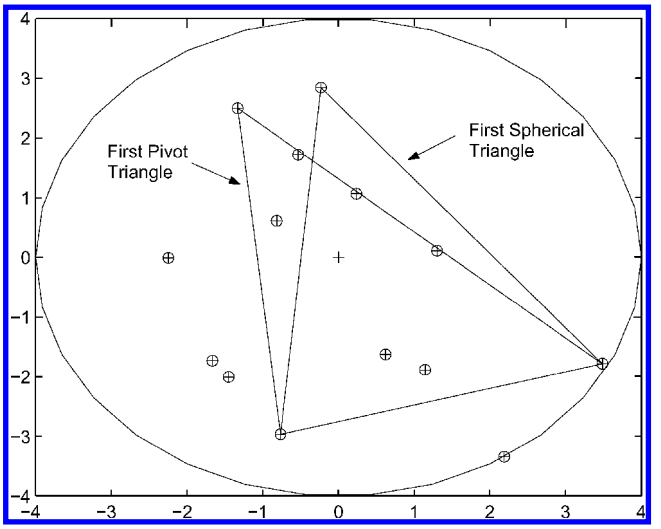

Fig. 1 Pivoting stars within field of view of the star tracker.

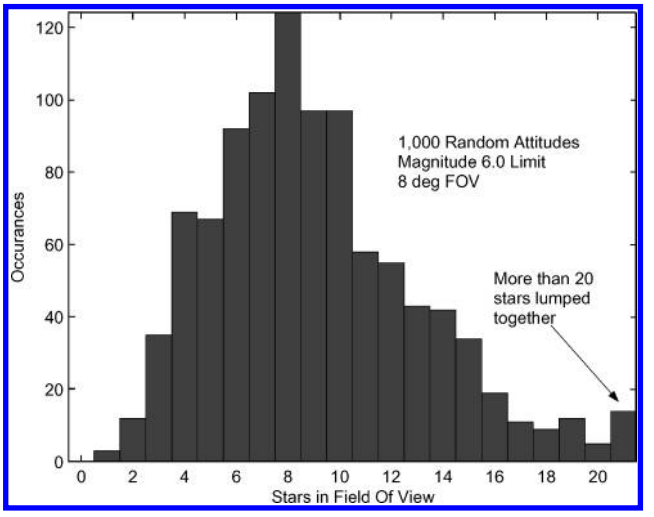

Fig. 2 Distribution of the number of stars in the field of view.

are created.[14] The first tree contains star number 1 of each of the first triangle's possible solutions. The second tree contains star number 2, and the third tree contains star number 3. One of the pivot triangle stars is checked to see if it exists in any of the three binary trees, and if it does, it tests the other two stars to see if it matches any of the stars of the triangle it matched in the binary tree. If not, a second star is tested against the binary trees, similar to the first. If still nothing matches, that triangle can be excluded as a solution. This approach greatly reduces the number of overall combinations that need to be tested, which greatly improves the speed of the algorithm.

Comparison of Methods

To test the angle and planar triangle methods, a random star-tracker attitude is generated 1000 times using the approach shown in Ref. 20. Figure 2 shows the distribution of the number of stars in the FOV. Each method is tested to see if it can positively identify at least one angle or triangle in the FOV. The star-pattern-recognition algorithm does not attempt to determine final attitude; it only identifies the stars that make up the angles or triangles. If all of the stars determined by the star-identification algorithm actually are in the FOV, it is considered a correct result. If any of the stars output by the algorithm are not in the FOV, the result is considered a failure. If the algorithm cannot yield a single solution to the angles or triangles in the FOV, it is recorded as an inconclusive result. Under ideal conditions, a star tracker will only see stars within its FOV; however, it is entirely possible that a piece of space debris enters the FOV of a star tracker, which may be misinterpreted as a star. Both the angle

method and the planar triangle method are tested to see how such a "false star" affects the ability of each algorithm to reach a solution.

Each time the star-pattern-recognition algorithm is tested, the simulated star tracker must be set in a random attitude and the stars within the FOV determined. The locations of the stars then have to be mapped to the star-tracker body frame to demonstrate that the properties of the triangles cataloged are not dependent on the attitude in which the triangles properties have been calculated. Given the body axis in terms of the tracker coordinate frame, the location of the stars in the ECI coordinate frame can be converted to the frame of the star tracker using $\boldsymbol{b}_i = A\boldsymbol{r}_i$ with a random attitude. The coordinates of the stars are so far still true, which would not be reported by a real star tracker. For a typical tracker with a 512×512 pixel array and 8×8–deg FOV, the pixel subtense is about 60 arcsec or 270 μrad. Typical errors are about 1/10th of a pixel, so a measurement error with a standard deviation, denoted by σ_i in Eq. (2b), of 21 μrad is added to the true body vectors.

If a false star is added to the FOV, then another random frame is created such that one axis is collinear with the boresight of the tracker. Either of the two remaining axes is perpendicular to the boresight, and so a vector is created along one of the axes with a random arc length up to half the FOV of the tracker. The direction of the false star in ECI coordinates is then the boresight vector plus the random vector unitized. The vector is then converted to the frame of the star tracker and noise added as if it were another star. This false star is then added to the list of stars seen in the FOV of the star tracker and used in either the angle or planar triangle method for solution.

Testing Conditions

The CPU times stated are based on hardware using an Apple Powerbook G4, 500 MHz, with 512 MB of RAM. All simulations are done using MATLAB® version 6.5.0.1915, release 13. All graphics output from the routines are turned off and care is taken to ensure there are no other significant processes running in the background during testing. Each method is reduced to a single subroutine called from the random attitude-testing routine. The time is measured from the line of code directly before calling the routine to the time it returns from it. The MATLAB variable "CPUTIME" is used. Both the angle and planar triangle method star-pattern-recognition algorithms are run with only one limit: if the number of possible solutions to any particular angle or triangle exceeds 10,000, then the angle or triangle is rejected. A typical angle will have only a few hundred possible solutions, and a triangle many hundred solutions. Angles or triangles that have far more possible solutions take a large amount of time to process and do not contribute to a successful run, so they are removed. Although the absolute time measurements reported here cannot be directly compared to actual star trackers, because the hardware is so completely different, it does make relative comparison of the angle method and planar triangle method possible.

Angle Method Results

In Fig. 3, it can be seen that the angle method is successful in identifying at least one angle in the FOV 59% of the time and cannot positively identify any of the angles 40% of the time, even when allowed unlimited pivots. Less than 1% of the attitudes tested contained less than two stars within the FOV, for which the angle method could not be tested and less than 1% of the attitudes give incorrectly identified results. Because every possible triangle has been cataloged, the only way an incorrect result can occur is if the correct solution lies outside the 3σ bound placed on the angle-recognition algorithm. Because the probability of the angle being within a 3σ bound is 99.7%, the errors encountered are not unexpected. In Fig. 4, it can be seen that the angle method rarely works if there are fewer than four stars in the FOV. This means that at least two pivots are required before any stars can be identified. It is interesting to note that the angle method is most successful when there are four stars in the FOV. If more than two pivots are required, the rate of success drops.

The number of pivots required by the angle method is shown in Fig. 5. The average number of pivots is 5.669, with a standard

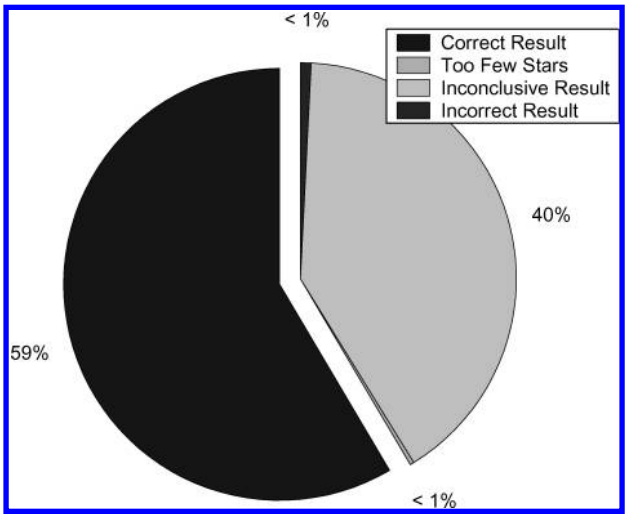

Fig. 3 Overall results for angle method without pivot limit.

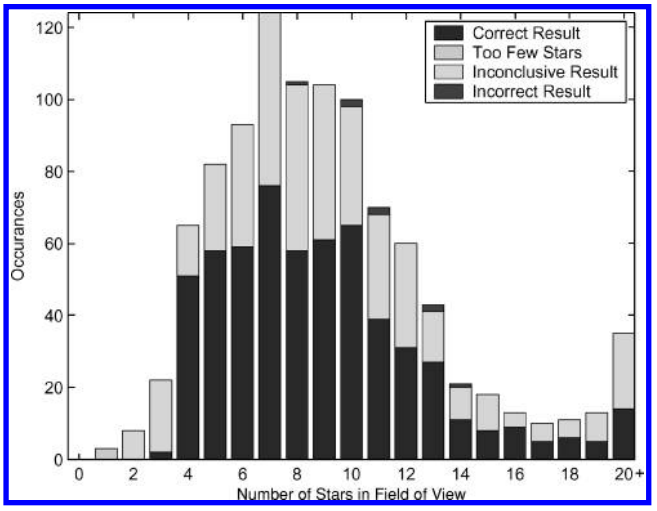

Fig. 4 Distribution of results for angle method without pivot limit.

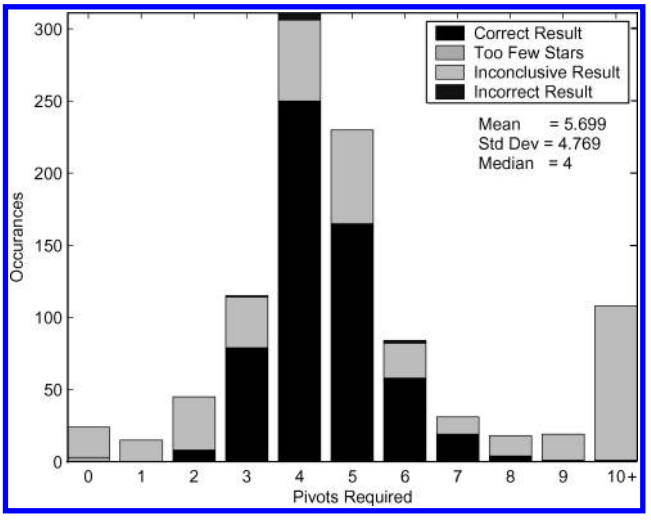

Fig. 5 Pivots required for angle method without pivot limit.

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

deviation of 4.769. Because the angle method requires two stars before pivoting, the average number of stars required is the average number of pivots required plus two. In this case, the angle method requires an average of 7.669 stars before yielding a result, positive or negative.

It is important to note that virtually any attitude that required nine or more pivots would ultimately not be solved. The angle method attempts pivots from largest angle to smallest angle, because the largest angles tend to have the smallest number of possible solutions. After a certain number of pivots, the number of possible solutions for an angle becomes so great that pivoting no longer reduces the possible number of solutions. In some cases, the number of solutions can begin to rise, worsening the problem. Because a lot of time is spent by the angle-method algorithm determining the order in which to pivot, limiting the number of pivots to nine significantly improves the time required by the angle method without significantly affecting the success rate of the method.

The CPU time required to execute the angle method is shown in Fig. 6. The average time required is 7.203 s with a standard deviation of 3.787 s. Note that this chart shows a short spike at the left side, which results from requiring at least three stars in the FOV before a pivot can be made. Because no pivots can be made in those situations, little CPU time is required. Another spike exists at the right side of the chart. All the attitudes requiring more than 10 s are grouped together to make the chart more compact.

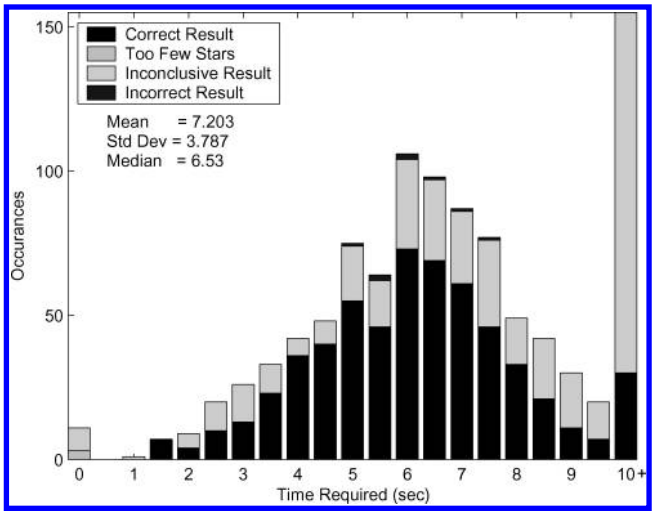

Fig. 6 CPU time required for angle method without pivot limit.

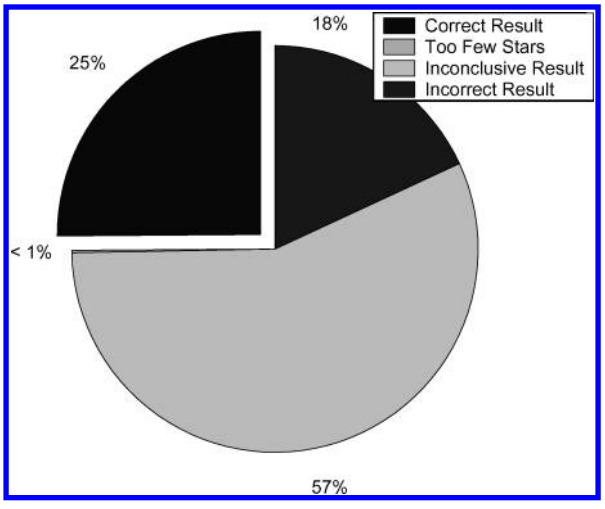

Fig. 7 Angle method results with a false star in the FOV.

With the nine-pivot limit, the overall success rate climbs slightly to 61%. Each test uses different attitudes, so it is not unreasonable to see some variation in the results after only 1000 attitudes. It does demonstrate, however, that limiting the pivots does not significantly affect the success rate of the method. More interesting is that the average time required per attitude drops to 6.217 s, a 13.7% improvement. The standard deviation also drops to 2.142 s. Because the number of pivots is limited, the average number of pivots required drops to 4.67, and therefore the number of average number of stars required drops to 6.67, a 15% improvement.

If a false star is added to the FOV, the angle method has a much more difficult time reaching a solution, as shown in Fig. 7. The overall success rate drops to 25% and 18% incorrect results. Essentially, if any of the stars used by the angle method include the false star, the angle method cannot reach the correct result and will either approach the wrong solution or no solution at all. We should note that the results of the angle method shown here do not seem to match the current state of the art provided by several vendor specifications. There are many possibilities for this disparity. For example, numerous other checks can be created to increase the results of the angle method, such as using relative brightness information.[21] The results shown here are strictly based on only matching angles with no other checks. The angle method results are next compared to results using planar triangles.

Planar Triangle Method Results

The overall results for the planar triangle method are shown in Fig. 8. This method produces a successful result 95% of the time but is not able to positively identify any planar triangles 4% of the time. Less than 1% percent of the attitudes contained less than three stars in the FOV, showing that this does not occur often. In addition, less than 1% of the attitudes are incorrectly identified, which is to be expected because of the 3σ bound placed on the triangle search.

The overall result as a function of the number of stars present in the FOV is shown in Fig. 9. This method cannot be used when there are fewer than three stars in the FOV and is never able to determine any of the triangles in the FOV with fewer than four stars, meaning that at least one pivot is always required. Once there are four stars in the FOV, the planar triangle method almost always succeeds in finding the correct solution. The number of pivots required by the planar triangle method is shown in Fig. 10. The mean number of pivots required is 1.262 and more than two pivots are rarely required. Because the planar triangle method requires three stars before a pivot can occur, the average number of stars required by this method is

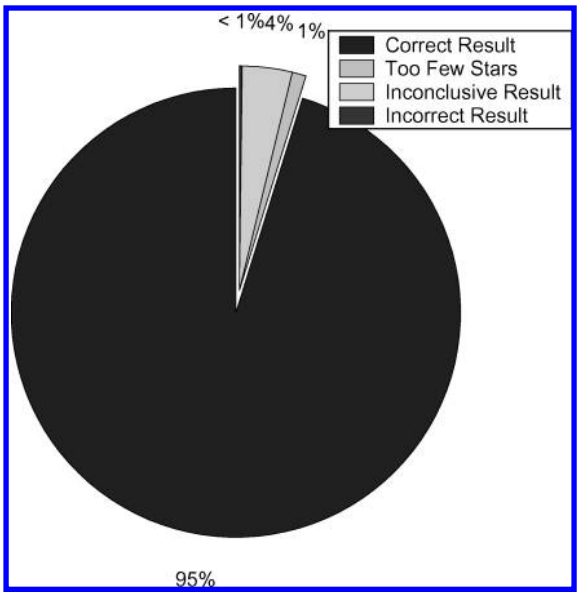

Fig. 8 Overall results for planar triangle method without pivot limit.

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

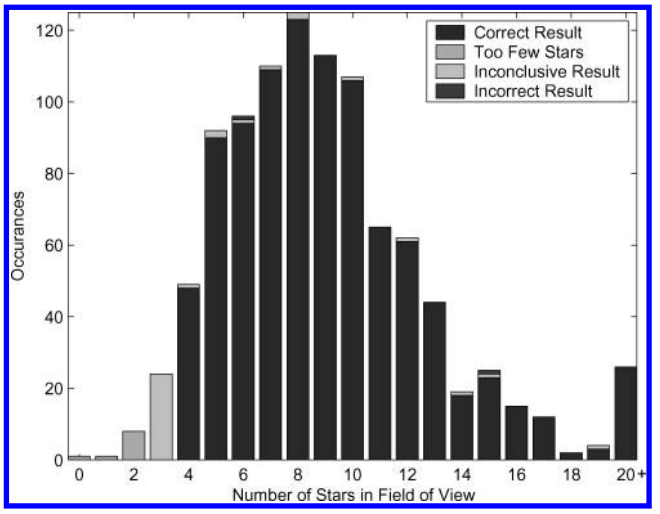

Fig. 9 Distribution of results for planar triangle method without pivot limit.

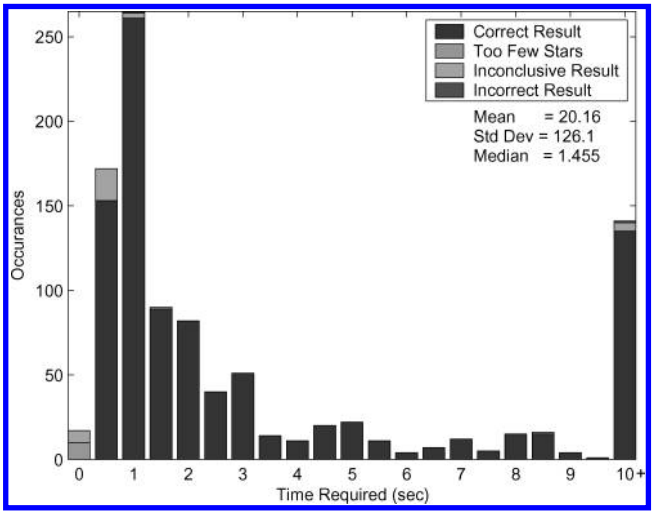

Fig. 11 CPU time required for planar triangle method without pivot limit.

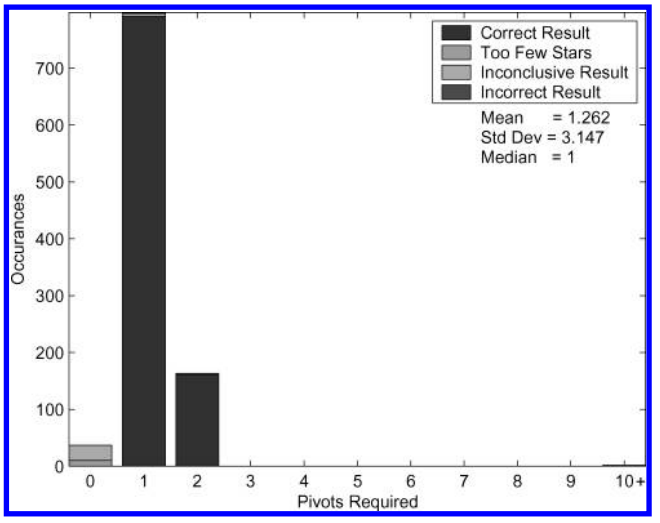

Fig. 10 Pivots required for planar triangle method without pivot limit.

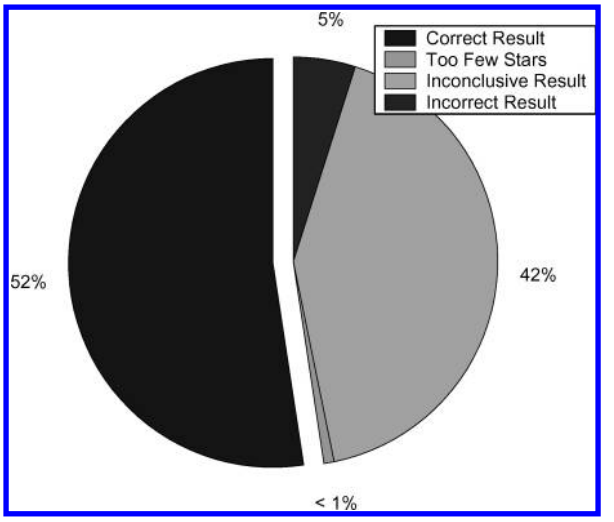

Fig. 12 Planar triangle method results with a false star in the FOV.

4.262. Because only a small number of attitudes require more than two pivots, the time efficiency of the method might be improved by limiting the number of pivots required to two.

The CPU time required for the planar triangle method is shown in Fig. 11. Solutions that take 10 s or more are grouped together into the 10-s slot to save space. The average time required is 20.16 s, yet the median is 1.455 s. This happens because some of the 150 attitudes that took more than 10 s took extraordinary amounts of time, as long as 400 s. These attitudes generally contained 20 or more stars in the FOV, which creates conditions where thousands of triangles in the FOV have to be studied and ordered for pivoting. Without pivot limits, the time required can be surprising.

As used in the angle method, by limiting the number of pivots to an appropriate number, the time required by the method can be improved without significantly changing the rate of success. Because the planar triangle method rarely needs more than two pivots, the number of pivots can be limited to two. The resulting graphs look very much like the results without pivot limits and are not included here. With this limit the overall success rate drops slightly to 94%. The average time drops significantly to 1.022 s, a 98% improvement. The median moves to 0.7531 s. Because the number of pivots is limited, the average number of pivots drops to 1.097, meaning only an average of 4.097 stars are required.

If a false star is added to the FOV, the planar triangle method, like the angle method, has a much more difficult time reaching a solution, as shown in Fig. 12. The overall success rate drops to 52%, and 5% of the results are incorrect. Like the angle method, if a false star happens to be used by the method, the planar triangle method cannot reach the correct result and will either approach the wrong solution or no solution at all.

Summary of Results

A summary of the results between the angle method and planar triangle method, both with pivot limits, is shown in Table 1. Being able to match the stars in the celestial sphere to those seen in the FOV of a star tracker by using more than one property can result in a very fast recognition algorithm with a high success rate. For a star tracker with an 8-deg FOV, sensitive to magnitude 6.0 stars, the planar triangle method is almost twice as likely to be successful in identifying stars than the angle method. With appropriate pivot limits placed on both methods to minimize the response times without significantly sacrificing the rate of successful results, the planar triangle method achieves a solution five times faster than the angle method. Just as important, the planar triangle method achieves greater success using fewer stars than the angle method. The planar triangle method, on average, examines only 4.097 stars vs the angle method's 6.671, which is a 39% improvement.

Both methods have similar amounts of incorrect results when there is no false star in the FOV, because of the 3σ bound on the angle and planar triangle searches. With such bounds, it is expected that 0.3% of the attitudes will not be determined correctly, because in those situations the correct solution to an angle or triangle lies outside the bounds of the search. If a false star is added to the FOV of

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

Table 1 Angle method with pivot limits vs planar triangle method with pivot limits

Item	Angle	Triangle
Minimum stars required	2	3
Min. stars req'd for success	3	4
overall success rate	61%	94%
Inconclusive result	37%	5%
Incorrect result	<1%	<1%
Average no. pivots required	4.67	1.097
Average no. stars required	6.671	4.097
Average CPU time required	6.217 s	1.022 s
False star correct result	25%	52%
False star inconclusive result	57%	42%
False star incorrect result	18%	5%

the star tracker, success rates of both methods suffer, but the planar triangle method suffers less. It is able to correctly determine the stars in the FOV almost twice as often as the angle method and incorrectly identifies the stars in the FOV about 15% as often. This is due to the fact that fewer stars are required to determine the stars in the FOV and so the planar triangle method is less likely to start with or pivot to a triangle that would include the false star. It is interesting to note that the planar triangle method has almost the same overall success with a false star in the FOV as the angle method does without it.

Determining the order for pivoting requires a large portion of the CPU time, particularly for the planar triangle method, because combinations of three stars have to be examined as opposed to combinations of two stars required by the angle method. Because the number of pivots for the planar triangle method can be limited to three without greatly affecting the results, the average CPU time required dramatically drops. The angle method can only be limited to nine pivots before impacting the results and so cannot benefit as much. The price for speed is certainly storage. Requiring 167 MB, the planar triangle catalog, including the k-vector array, is more than ten times larger than that required for the angle method. Methods for data storage aboard spacecraft are constantly improving and perhaps in the not-too-distant future (if not already), this size database will not be a significant issue.

Conclusions

A new approach for star-pattern recognition has been developed using planar triangles that incorporates area and polar moments to effectively identify stars. The approach presented here tackles the worst-case scenario for a star tracker: no a priori attitude information and no other sensor information to assist the identification process. Simulation results indicated that the planar triangle method is almost twice as likely to be successful in identifying stars and achieves greater success using fewer stars than the angle method. Furthermore, the planar triangle method is less sensitive to false-star problems than the angle method.

Acknowledgments

The authors acknowledge the significant discussions on the subject matter with John L. Junkins and Daniele Mortari from Texas A&M University. The authors also acknowledge the reviewers for many helpful comments and suggestions.

References

[1]Ju, G., and junkins, J. L., "Overview of Star Tracker Technology and Its Trends in Research and Development," *Advances in the Astronautical Sciences, The John L. Junkins Astrodynamics Symposium*, Vol. 115, Univelt Inc. Publishers, San Diego, CA, 2003, pp. 461–478.

[2]Gottlieb, D. M., "Star Identification Techniques," *Spacecraft Attitude Determination and Control*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1978, pp. 259–266.

[3]Ketchum, E. A., and Tolson, R. H., "Onboard Star Identification Without A Priori Attitude Information," *Journal of Guidance, Control, and Dynamics,* Vol. 18, No. 2, March–April 1995, pp. 242–246.

[4]Kosik, J. C., "Star Pattern Identification Aboard an Inertially Stabilized Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 2, March–April 1991, pp. 230–235.

[5]Gambardella, P., "Algorithms for Autonomous Star Identification," NASA T.R TM-84789, Oct. 1980.

[6]Junkins, J. L., and White, C. C., and Turner, J. D., "Star Pattern Recognition for Real Time Attitude Determination," *Journal of Astronautical Sciences*, Vol. 25, No. 3, Nov. 1997, pp. 251–270.

[7]Junkins, J. L., and Strikwerda, T. E., "Autonomous Attitude Estimation via Star Sensing and Pattern Recognition," *Proceedings of the Flight Mechanics and Estimation Theory Symposium*, NASA, Greenbelt, MD, 1979, pp. 127–147.

[8]Strikwerda, T. E., Junkins, J. L., and Turner, J. D., "Real-Time Spacecraft Attitude Determination by Star Pattern Recognition: Further Results," AIAA Paper 79-0254, Jan. 1979.

[9]Sheela, B. V., Shekhar, C., Padmanabhan, P., and Chandrasekhar, M. G., "New Star Identification Technique for Attitude Control," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 2, March–April 1991, pp. 477–480.

[10]Mortari, D., Junkins, J. L., and Samaan, M. A., "Lost-in-Space Pyramid Algorithm for Robust Star Pattern Recognition," *Advances in the Astronautical Sciences, Guidance, and Control*, Vol. 107, Univelt Publishers Inc., San Diego, CA, 2001, pp. 49–68.

[11]Williams, K. E., Strikwerda, T. E., Fisher, H. L., Strohbehn, K., and Edwards, T. G., "Design Study: Parallel Architectures for Autonomous Star Pattern Identification and Tracking," American Astronomical Society, AAS Paper 93-102, Feb. 1993.

[12]Padgett, C., and Kreutz-Delgado, K., "A Grid Algorithm for Autonomous Star Identification," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, No. 1, Jan. 1997, pp. 202–213.

[13]Ball Aerospace Systems Group, "Specification Sheet for Ball Aerospace CT-601 Star Tracker," Electro-Optics Cryogenics Division Boulder, CO, 2001.

[14]Cole, C. L., *Fast Star Pattern Recognition Using Spherical Triangles*, Mastre's Thesis, State University of New York at Buffalo, NY, Jan. 2004.

[15]Mortari, D., "A Fast On-Board Autonomous Attitude Determination System Based on a New Star-ID Technique for a Wide FOV Star Tracker," *Advances in the Astronautical Sciences, Sixth Annual AIAA/AAS Space Flight Mechanics Meeting*, Vol. 93, Pt. 2, Univelt Publishers Inc., San Diego, CA, 1996, pp. 893–903.

[16]Crassidis, J. L., Markley, F., Kyle, A., and Kull, K., "Attitude Determination Improvements for GOES," *Proceedings of the Flight Mechanics/Estimation Theory Symposium*, NASA, Greenbelt, MD, 1996, pp. 151–165.

[17]Shuster, M. D., "Maximum Likelihood Estimation of Spacecraft Attitude," *The Journal of the Astronautical Sciences*, Vol. 37, No. 1, Jan.–March 1989, pp. 79–88.

[18]Shuster, M. D., 'Kalman Filtering of Spacecraft Attitude and the QUEST Model," *Journal of the Astronautical Sciences*, Vol. 38, No. 3, July–Sept. 1990, pp. 377–393.

[19]Alonso, R., and Shuster, M. D., "TWOSTEP: A Fast Robust Algorithm for Attitude-Independent Magnetometer-Bias Determination," *Journal of the Astronautical Sciences*, Vol. 50, No. 4, Oct.–Dec. 2002, pp. 433–451.

[20]Shuster, M. D., "Uniform Attitude Probability Distributions," *Journal of the Astronautical Sciences*, Vol. 51, No. 4, Oct.–Dec. 2003, pp. 451–475.

[21]Juang, J.-N., Kim, H.-Y., and Junkins, J. L., "An Efficient and Robust Singular Value Method for Star Pattern Recognition and Attitude Determination," *Advances in the Astronautical Sciences, The John L. Junkins Astrodynamics Symposium*, Vol. 115, Univelt Publishers Inc., San Diego, CA, 2003, pp. 491–504.

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314

This article has been cited by:

1. Geoffrey R. McVittie. 2012. Color star tracking I: star measurement. *Optical Engineering* **51**:8, 084402. [CrossRef]
2. Jae-Jun Kim, Jack Tappe, Albert Jordan, Brij AgrawalStar Tracker Attitude Estimation for an Indoor Ground-Based Spacecraft Simulator . [Citation] [PDF] [PDF Plus]
3. David D. Needelman, James P. Alstad, Peter C. Lai, Haytham M. Elmasri. 2010. Fast Access and Low Memory Star Pair Catalog for Star Pattern Identification. *Journal of Guidance, Control, and Dynamics* **33**:5, 1396-1403. [Citation] [PDF] [PDF Plus]
4. Wei Quan, Jiancheng Fang. 2010. A Star Recognition Method Based on the Adaptive Ant Colony Algorithm for Star Sensors. *Sensors* **10**:3, 1955-1966. [CrossRef]

Downloaded by University of Toronto on August 11, 2014 | http://arc.aiaa.org | DOI: 10.2514/1.13314