JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 18, No. 2, March–April 1995

Study of Strapdown Navigation Attitude Algorithms

Howard Musoff* and James H. Murphy†
Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts 02139

Typically coning motion has been used as the input to test the effectiveness of attitude algorithms. Jacobian elliptic functions are also shown to meet the criteria for a test input. Simulations using several of these inputs are performed on Miller's algorithm. Analytical evaluation of a Jacobian elliptic function input would be prohibitive. To alleviate this problem Richardson extrapolation is used to numerically evaluate the algorithms for each test input. Additionally, a method for improving all of the attitude algorithms using interpolation is presented.

I. Introduction

DEVELOPMENTS of attitude algorithms for strapdown inertial navigators have been predicated on using classical coning motion as a test input.[1,2] These algorithms are designed to reduce attitude errors in the presence of this specific motion. In this paper, we examine the justification for using coning motion and develop more general test inputs for checking the performance of these algorithms. Furthermore, a procedure borrowed from numerical analysis is used in conjunction with our new test inputs for actually computing their performance.[3]

Finally, a method developed by one of the authors based on interpolation of gyroscope outputs is shown to be very useful in further enhancing the performance of any of the attitude algorithms examined by us.[4]

II. Use of Test Inputs

All the algorithmic developments alluded to in the Introduction are for the solution of the rotation vector differential equation[1]

$$\dot{\boldsymbol{\Phi}} = \boldsymbol{\omega} + \tfrac{1}{2}\boldsymbol{\Phi} \times \boldsymbol{\omega} + \tfrac{1}{12}\boldsymbol{\Phi} \times (\boldsymbol{\Phi} \times \boldsymbol{\omega}) \tag{1}$$

where $\boldsymbol{\Phi}$ is the rotation vector and $\boldsymbol{\omega}$ is the body angular rate vector.

Classical coning motion used to develop algorithms for the solution of Eq. (1) is given by the following body rate vector[1]:

$$\boldsymbol{\omega} = \begin{bmatrix} \dfrac{wa^2}{2} \\ -aw \sin wt \\ aw \cos wt \end{bmatrix} \tag{2}$$

where w is the coning frequency, and a is the coning half-angle (the cone made by the x axis has an angle of $2a$). Vector $\boldsymbol{\omega}$ has been simplified without loss of generality by assuming that a is very small (i.e., $a \cong \sin a$).

A heuristic argument will be used to show how more general versions of Eq. (2) can be developed. A first-order solution to Eq. (1) is given by

$$\boldsymbol{\Phi} = \int_0^t \boldsymbol{\omega}\, \mathrm{d}t \tag{3}$$

Substituting Eq. (3) into Eq. (1) results in

$$\dot{\boldsymbol{\Phi}} = \boldsymbol{\omega} + \tfrac{1}{2}\boldsymbol{\Phi}_1 \times \boldsymbol{\omega} + \tfrac{1}{12}\boldsymbol{\Phi}_1 \times (\boldsymbol{\Phi}_1 \times \boldsymbol{\omega}) \tag{4}$$

The accompanying rotation vector rate error for the first-order solution is given by the vector cross product terms in Eq. (4), because the rate for the first-order solution should be $\boldsymbol{\omega}$. These terms are a maximum when $\boldsymbol{\omega}$ is orthogonal to $\boldsymbol{\Phi}_1$, that is, when

$$\boldsymbol{\omega} \perp \int_0^t \boldsymbol{\omega}\, \mathrm{d}t \tag{5}$$

This condition is not quite met with the classical coning motion because of the zero frequency term in the first component of Eq. (2). However, it is the sinusoidal components in Eq. (2) that are actually used to test the algorithms in Refs. 1 and 2 and for these, with the zero frequency term ignored, our condition (5) does hold.

However, instead of accepting classical coning motion as fulfilling the worst case requirement on $\boldsymbol{\omega}$ and its integral (to the extent of ignoring the constant rate terms), we look for a more general motion which also meets the requirement. The condition (5) can also be represented by the vector dot product

$$\boldsymbol{\omega} \cdot \int_0^t \boldsymbol{\omega}\, \mathrm{d}t = 0 \tag{6}$$

or

$$\theta_x \dot{\theta}_x + \theta_y \dot{\theta}_y + \theta_z \dot{\theta}_z = 0 \tag{7}$$

with

$$\boldsymbol{\theta} = \left(\int_0^t \omega_x\, \mathrm{d}t, \int_0^t \omega_y\, \mathrm{d}t, \int_0^t \omega_z\, \mathrm{d}t \right)^T \tag{8}$$

An integral of Eq. (7) is given by

$$\theta_x^2 + \theta_y^2 + \theta_z^2 = \kappa^2 \tag{9}$$

where κ is a constant set by the particular test motion chosen. Equation (9) is not only satisfied by combinations of sinusoids but also by Jacobian elliptic functions.[5]

The solution of Euler's equations of motion for an asymmetric, torque free rigid body can be expressed in terms of Jacobian elliptic functions. The body rates W_x, W_y, and W_z are given by

$$W_x = a_X cn(wt, m) \tag{10}$$

$$W_y = a_y sn(wt, m) \tag{11}$$

$$W_z = a_z dn(wt, m) \tag{12}$$

where m is a small parameter dependent on the asymmetry in the rigid body moments of inertia about the principal axes and cn, sn,

Received June 29, 1993; presented as Paper 93-3891 at the AIAA Guidance, Navigation, and Control Conference, Monterey, CA, Aug. 9–11, 1993; revision received June 22, 1994; accepted for publication June 27, 1994.

*Principal Member Technical Staff, Performance Analysis Group. Member AIAA.

†Member Technical Staff, Sapient Corp., 1 Memorial Drive, Cambridge, MA 02142.

and dn are the Jacobian elliptic functions. They satisfy the following formulas[5]:

$$cn^2(wt, m) + sn^2(wt, m) = 1 \tag{13}$$

$$dn^2(wt, m) + msn^2(wt, m) = 1 \tag{14}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} sn(wt, m) = wcn(wt, m)dn(wt, m) \tag{15}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} cn(wt, m) = -wsn(wt, m)dn(wt, m) \tag{16}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} dn(wt, m) = -mwsn(wt, m)cn(wt, m) \tag{17}$$

Because cn, sn, and dn have the relationships given by Eqs. (13) and (14) we can use them for the purpose of generating our test body rates. They are a generalization of the sinusoidal motion case and as m approaches zero (in the perfectly symmetrical case) cn becomes cos, sn becomes sin, and dn becomes a constant.

It should be noted that for the body rates given by Eqs. (10–12) the amplitudes

$$a_x \neq a_y \tag{18}$$

even though they would be equal for classical sinusoidal coning motion. The case of almost equal amplitudes is approached, however, if for the three principal axis moments of inertia I_1, I_2, and I_3 of the rigid body we have, for example,[5]

$$I_3 > I_1, I_2 \tag{19}$$

and

$$I_1 \neq I_2 \qquad \text{but} \qquad I_1 \cong I_2 \tag{20}$$

which does characterize many space vehicles.

III. Procedures for Computing Algorithm Performance

Analytic computation of the correct rotation vector response to our more general coning motion body rates can get quite involved. A convenient way around this for the purpose of computing the performance of specific algorithms is an application of Richardson extrapolation technique used in numerical analysis.[3]

Using this technique results in knowledge of how the algorithm error varies with the size of the algorithm iteration interval. Analytical derivations for this were employed in Ref. 1 and 2. Therefore, we can compare our results directly with those of the references.

If for the component of the rotation vector being investigated we record its value for three different algorithm iteration intervals at a single frequency of the coning test input, then to first order the relationships of these outputs is

$$\phi_1 - \phi_2 = k(\Delta t_1)^r - k(\Delta t_2)^r \tag{21}$$

$$\phi_1 - \phi_3 = k(\Delta t_1)^r - k(\Delta t_3)^r \tag{22}$$

where ϕ_j is the output of the algorithm having an iteration interval of Δt_j, Δt_j is the size of the iteration interval, k is the constant coefficient for the algorithm being investigated, and r is the iteration interval exponent for the algorithm being investigated.

Here $k(\Delta t_j)^r$ is assumed to be the error in the algorithm. Its use is justified by the polynomial form of the algorithm under consideration. Experimental results subsequently presented will also confirm its applicability.

Subtracting ϕ_2 from ϕ_1 and ϕ_3 from ϕ_1 gets rid of the common outputs of the algorithm at the different iteration rates and leaves only the difference of the errors.

Equations (21) and (22) are a pair of equations in two unknowns r and k. An accurate approximate solution of Eqs. (21) and (22) is easy to obtain by choosing one of the iteration intervals to be very small.

Say

$$\Delta t_1 \ll \Delta t_2, \Delta t_3 \tag{23}$$

so that we can ignore $k(\Delta t_j)^r$. Thus we have

$$\phi_1 - \phi_2 \cong -k(\Delta t_2)^r \tag{24}$$

$$\phi_1 - \phi_3 \cong -k(\Delta t_3)^r \tag{25}$$

Equations (24) and (25) are solved for r and k which leads to

$$r \cong \ln\left(\frac{\phi_1 - \phi_2}{\phi_1 - \phi_3}\right) \Bigg/ \ln\left(\frac{\Delta t_2}{\Delta t_3}\right) \tag{26}$$

$$k \cong -(\phi_1 - \phi_2)/(\Delta t_2)^r \tag{27}$$

We already have a benchmark to check out this technique because we know k and r for the drift error of several algorithms. A similar analysis can be made for investigating the effects of algorithm word length.

IV. Simulations

The first step of the simulations is to show that Richardson extrapolation can be used for the analysis of the attitude algorithms as described in Sec. III. A test of this method is to check the results computed by Richardson extrapolation with the results as published.[1] For our simulations we have chosen sampling rates of 300 Hz, 150 Hz, and 30,000 Hz. Classical coning is used as the test input with the parameter a, the coning angle 1 deg, and with the coning frequency ω 10 Hz. One attribute of using Richardson extrapolation is that the exponent and the magnitude can be calculated from a single coning frequency simulation, when typically several frequencies must be simulated and then plotted to observe whether the slope of the curve vs coning frequency matches analytically determined values. Figure 1a presents the drift of the Miller's algorithm with the coefficients from the solution to the Taylor series solution and whereas Fig. 1b shows the results with the coefficients optimized for the classical coning input. These plots are generated by using the 30,000 Hz quaternion as if it were exact and then

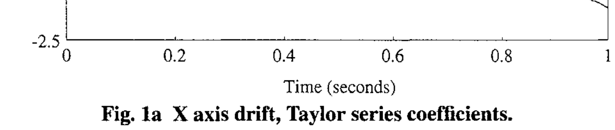

Fig. 1a X axis drift, Taylor series coefficients.

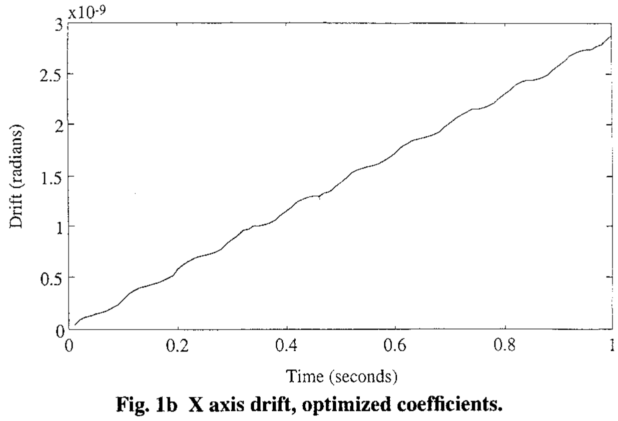

Fig. 1b X axis drift, optimized coefficients.

Table 1 Verification of Miller's results

	Exponent r	Magnitude k
Published Miller[1]		
with Taylor coef.	4	4.361×10^{-7}
Calculated	3.97	4.477×10^{-7}
Published Miller[1]		
with optimized coef.	6	5.769×10^{-9}
Calculated	6.04	5.755×10^{-9}

Table 2 Performance for classical coning vs Jacobian elliptic inputs

	r, Taylor coefficients	r, Optimized coefficients
Classical coning		
(drift on x axis)	3.97	6.04
Jacobian elliptic	4.01	4.01
(drift on x, y, z axes, respectively)	4.00	4.00
	3.96	4.28

comparing it with the 300 Hz or 150 Hz quaternion to derive the error quaternion.

$$\tilde{q}_{300} = q_{30,000} \cdot q^*_{300} \tag{28}$$

$$\tilde{q}_{150} = q_{30,000} \cdot q^*_{150} \tag{29}$$

where $\tilde{q}_n$ is the error quaternion for the n-Hz data, and q^*_n is the conjugate quaternion.

For classical coning the drift is present in the second element of the quaternion.

$$\text{drift} = k(\Delta t)^r \tag{30}$$

whereas the published results for the Taylor series coefficients[1] are

$$\text{drift}_T = \frac{a^2 w^5 (\Delta t)^4}{12,960} \tag{31}$$

and when the coefficients are optimized for classical coning the drift is

$$\text{drift}_O = \frac{a^2 w^7 (\Delta t)^6}{204,120} \tag{32}$$

The comparison of the computed and published results are presented in Table 1, where a is 1 deg, w is 10 Hz, and Δt is 1/300 s.

Clearly using Richardson extrapolation provides a numerical means of comparing algorithms without the need to derive the analytical solution for coning motion. Table 1 has rows for published solutions for Miller's algorithm with the coefficients developed using the Taylor series expansion for the solution to Eq. (1) assuming the input angles approximate a cubic. The optimized coefficients were developed by optimizing for the classical coning Eq. (2).[1]

Table 2 shows the results using the Jacobian elliptic function example. Here, $a_x = 1.1$ deg, $a_y = 0.9$ deg, $a_z = 1.0$ deg, $w = 10$ Hz, and $m = 0.7$ [refer to Eqs. (10–12)]. For the particular general coning input the optimized coefficients perform better than the Taylor series coefficients. The Jacobian elliptic function example shows the same performance for both sets of coefficients. The optimization as based on classical coning does not work for the Jacobian elliptic function.

V. Interpolation

As can be seen from previous sections, the performance of an attitude algorithm depends, among other factors, on the size Δt of the iteration interval. All of the algorithms examined would have improved performance if Δt could be reduced. This can obviously be done by using a higher algorithm iteration rate. More data from the gyros would be required in a given interval of time. This may not be possible if we are processing data on the ground using telemetered data because of telemetry constraints. We may also want to improve the performance of a system, whose gyro output data rates cannot be modified, through postflight ground processing.

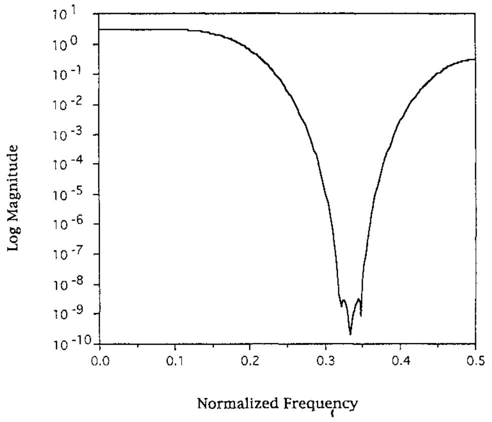

Fig. 2a Lagrange interpolator logarithmic magnitude response.

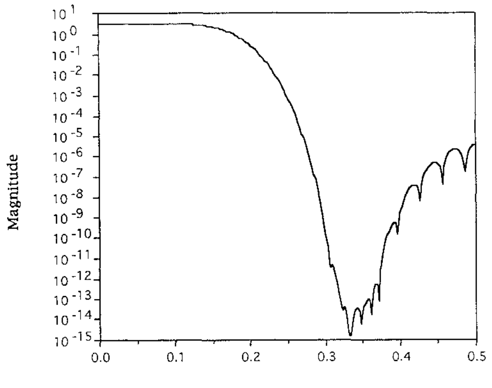

Fig. 2b Interpolating filter response.

These limitations on the amount of actual data can be overcome by interpolating between available gyro samples.[4] Doing this effectively reduces Δt and, therefore, the resulting coning error. A beneficial side effect is that any high-frequency noise (such as caused by quantization) in the original data is attenuated by the interpolating filter.

Increments $\Delta\theta$, of the gyro output are added together to form an angle θ which is then sampled at a higher rate than the original $\Delta\theta$ to form smaller increments corresponding to the desired reduced iteration interval.

The sampling between the original data points is performed by a Lagrangian interpolating filter.[3] This filter has an extremely flat passband at low frequencies and passes low-order derivatives of the gyro data unchanged. This high fidelity to both amplitude and low-order derivatives is required for the proper operation of the attitude algorithm. The low-order derivatives are required because the attitude algorithms are based on Taylor series expansions involving these derivatives.

For an example of the Lagrangian interpolation filter response we have chosen to interpolate by a factor of three. The resulting normalized response is shown in Fig. 2a. The low-pass filter amplitude shown is what we want. However, the amplitude at the Nyquist frequency (0.5) is much too high. Therefore, the Lagrangian filter is convolved with a Wilkinson filter[4] which also has excellent passband characteristics in addition to good attenuation of the higher frequencies. The response of these cascaded filters is shown in Fig. 2b.

By interpolating the original delta theta coning input and the computing the attitude at the higher rate we obtain performance similar to delta thetas sampled at the higher rate and the corresponding attitude calculated at that rate. Figures 3 and 4 compare attitude drift errors as a function of coning frequency for three different attitude computations without and with quantization, respectively. The first attitude computation is performed using 100-Hz $\Delta\theta$ information

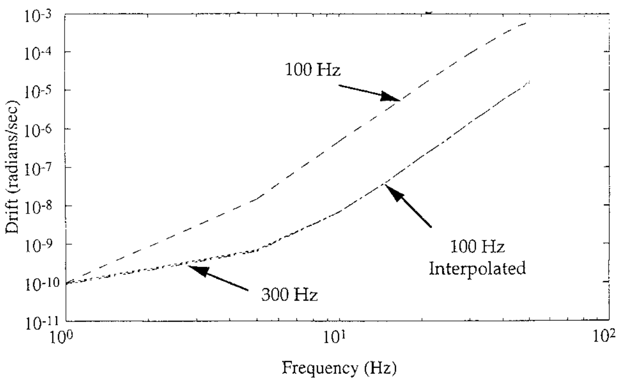

Fig. 3 Interpolated drift—Miller's algorithm.

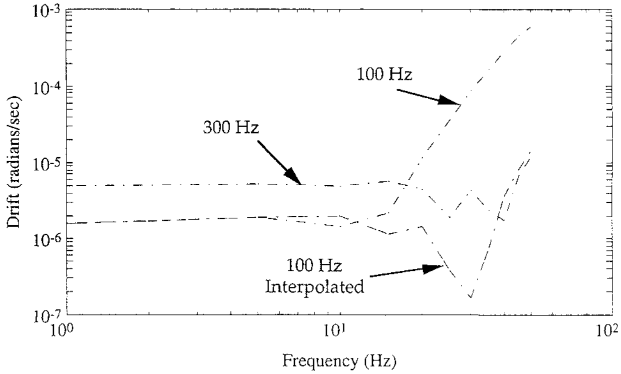

Fig. 4 Interpolated drift—Miller's algorithm with quantized data.

and an attitude update rate of 100 Hz. The next drift error shown for 100-Hz $\Delta\theta$ interpolated to a rate of 300 Hz and the attitude is updated at 300 Hz. The final error plotted is for 300-Hz $\Delta\theta$ with an attitude update rate of 300 Hz.

Figure 3 indicates that the results using interpolated $\Delta\theta$ are as good as the results computed with 300-Hz $\Delta\theta$. Figure 4 shows the drift error when the $\Delta\theta$ have been quantized to 1 μrad. The 300-Hz $\Delta\theta$ run shows worse performance at lower frequencies than the other runs. The 100-Hz $\Delta\theta$ and 100-Hz attitude update run has a higher drift error at the higher coning frequencies. The interpolated data run combines the low-frequency performance of the 100-Hz data with the low drift error of the 300-Hz attitude update at higher frequencies.

VI. Conclusions

We have shown that optimizing an attitude algorithm using classical coning as a test input is not valid when the input is composed of Jacobian elliptic functions that are representative of rigid body satellite angular motion. This raises the issue of how to optimize strapdown attitude algorithms in the presence of nonclassical coning inputs. The algorithm response to classical coning may be too narrow a criterion for some cases. Also, regardless of the algorithm type, significant improvement in algorithm performance can be obtained with the data interpolation technique disclosed here given the right circumstances.

References

[1] Miller, R. B., "A New Strapdown Attitude Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 6, No. 4, 1983, pp. 287–291.

[2] Lee, J. G., Yoon, Y. J., Mark, J. G., and Tazartes, D. A., "Extension of Strapdown Attitude Algorithm for High-Frequency Base Motion," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 4, 1990, pp. 738–743.

[3] Hildebrand, F. B., *Introduction to Numerical Analysis*, McGraw–Hill, New York, 1974, p. 100.

[4] Murphy, J. H., "Lagrangian Interpolation of Input Data to Reduce Coning Error of Attitude Algorithms for Navigation," M.S. Thesis CSDL-T-1011, Dept. of Electrical Engineering, Tufts Univ., Medford, MA, June 1989.

[5] Wertz, J. R. (ed.), *Spacecraft Attitude Determination and Control*, Kluwer Academic, Boston, MA, 1988, pp. 526–528.