A New Strapdown Attitude Algorithm

Robin B. Miller*
Aeronautical Research Laboratories, Defence Science and Technology Organisation, Melbourne, Australia

This paper develops the application to the strapdown attitude problem of the rotation vector concept, in which obtaining a solution for the rotation vector and updating the attitude quaternion are considered entirely separately. A new rotation vector algorithm is derived which takes three samples of gyro data per update, and offers greatly improved coning performance. This is obtained by making the assumption that gyro output varies according to a third-order time relationship, rather than second or first order as in other algorithms. Performance of the three sample algorithm is compared with well-known algorithms in a coning environment and in a random motion environment. The quaternion updating may be performed to whatever degree of accuracy is required: An economical modified fourth-order algorithm is proposed, and its comparative performance shown. This approach to the problem gives versatility in that computing resources use may to some extent be tailored according to the relative extents of coning and of fixed axis rotation in the expected environment.

Nomenclature

v = vector
θ = gyro output—a set of three samples, one from each axis
ϕ = rotation vector
ω = angular velocity
$\times$ = vector cross-product operator
$*$ = quaternion "multiplication" operator $[Q*q=(Q_0q_0 - Q\cdot q),\ (Q_0q+q_0Q+Q\times q)]$

The scalar part of a quaternion is written either as Q_0, q_0, or C; the vector part may be Q, q, or ϕS, according to the text. The quaternion may be written, e.g., $C,\phi S$, or as an array:

$$\begin{bmatrix} C \\ \phi_x S \\ \phi_y S \\ \phi_z S \end{bmatrix}$$

Introduction

The traditional approach to the problem of updating the quaternion or direction cosine matrix used as an attitude reference parameter in strapdown inertial navigation systems has been a solution of the differential equation of the parameter by, for example, Taylor series or Runge-Kutta techniques. The rotation vector concept was used by Bortz[1] and Jordan[2] as a means of obtaining a correction to be applied to the gyro outputs to account for noncommutativity effects. Gilmore[3] described an application where the gyro outputs were used to make incremental updates of the rotation vector in a "fast loop," with quaternion updating in a "slow loop"; this was based on the Bortz technique.

This paper will further develop the application of the rotation vector concept, and will show the versatility and relative conceptual simplicity of analysis which is available if obtaining a rotation vector solution and updating the quaternion are considered entirely separately. This analytical concept will be used to derive an algorithm which, taking three samples of gyro data per update, offers greatly improved coning performance at some cost per update in computer loading; alternatively, for a given coning performance, the computer loading may be considerably reduced. For example, the algorithm described by Gilmore was reported to have the same coning performance as the "traditional" third-order Taylor series algorithm of McKern,[4] with an approximately 50% reduction in computer loading (that is, 1420 multiplications per second instead of about 3000). The algorithm to be described offers this coning performance for a computer loading of about 600 to 800 multiplications per second (update rate of about 15 to 20 Hz).

The rotation vector ϕ for the interval is defined as a vector whose direction corresponds to the axis of, and whose length corresponds to the magnitude of, the "fixed axis" rotation which is equivalent to the total rotation undergone by the body during the interval. The first part of the analysis involves estimation of this vector from the gyro outputs obtained during the interval. For this purpose, certain assumptions are made concerning the behavior of the gyro output during the interval: e.g., that it varies linearly, or according to a square law, or, as in this case, a cube law.

The quaternion updating section of the analysis is merely the formation of the updating quaternion $q(h)$, which corresponds to the rotation which occurred in the interval, and using this to update the reference quaternion $Q(T)$. The $q(h)$ is given by the usual quaternion relation:

$$q(h)=C,\phi S$$

where $C=\cos(\frac{1}{2}\phi_0)$, $S=(1/\phi_0)\sin(\frac{1}{2}\phi_0)$, and $\phi_0=(\phi\cdot\phi)^{1/2}$, and the updated $Q(T+h)$ by quaternion "multiplication":

$$Q(T+h)=Q(T)*q(h)$$

Rotation Vector Estimation

The rotation vector equation is[1]

$$\dot{\phi}(t)=\omega(t)+\tfrac{1}{2}\phi(t)\times\omega(t)+A\phi(t)\times\{\phi(t)\times\omega(t)\} \qquad A\approx 1/12 \qquad (1)$$

for present purposes, the triple cross-product term will be assumed small enough to neglect. A Taylor series may be used to obtain a solution for this equation:

$$\phi(T+h)=\phi(T)+h\dot{\phi}(T)+(h^2/2!)\ddot{\phi}(T)+\ldots \qquad (2)$$

A gyro output variation of the form

$$\theta(t)=at+bt^2+ct^3 \qquad (3)$$

will be assumed, over the interval from T to $T+h$. By definition, $\theta(T)$ and $\phi(T)$ are zero, and $0<t<h$.

Received July 7, 1982; revision received Feb. 8, 1983.

*Research Scientist, Systems Operation Group.

Repeated differentiation of Eq. (3) with respect to time gives the relations at time $T(t=0)$:

$$\omega(T)=a \quad \dot{\omega}(T)=2b \quad \ddot{\omega}(T)=6c \quad \dddot{\omega}(T)=0 \tag{4}$$

Repeated differentiation of Eq. (1) with respect to time, and application of Eqs. (4) and the relation $\phi(T)=0$ gives the results

$$\frac{d^2}{dt^2}\{\phi(T)\}=2b \qquad \frac{d^3}{dt^3}\{\phi(T)\}=6c+(a\times b)$$

$$\frac{d^4}{dt^4}\{\phi(T)\}=6(a\times c) \qquad \frac{d^5}{dt^5}\{\phi(T)\}=12(b\times c) \tag{5}$$

Putting Eqs. (5) into Eq. (2) gives

$$\phi(T+h)=ah+bh^2+ch^3+\frac{1}{6}h^3(a\times b)$$
$$+\frac{1}{4}h^4(a\times c)+\frac{1}{10}h^5(b\times c) \tag{6}$$

(Henceforth, $\phi(T+h)$ will be written ϕ.)

The incremental gyro outputs at $t=⅓h$, $⅔h$, and h are θ_1, θ_2, and θ_3, respectively, with $\theta=\theta_1+\theta_2+\theta_3$. Putting these into Eq. (3), we obtain three equations which yield the values of a, b, and c.

When these values of a, b, and c are substituted into Eq. (6), it is found that

$$\phi=\theta+X(\theta_1\times\theta_3)+Y\theta_2\times(\theta_3-\theta_1) \tag{7}$$

where $X=0.4125$ and $Y=0.7125$.

It will now be shown that coning performance may be further improved by using the values $X=0.45$ and $Y=0.675$.

Application to Coning Motion

For the purposes of this exercise, we shall use a coning motion as described in the Appendix, in which are given the reference to body quaternion $Q(t)$, the true updating quaternion $q(h)$, the angular velocity $\omega(t)$, and the gyro output vector $\theta(h)$ for the period from time t to $(t+h)$. Gyro output vectors, taken at the three intervals as above, were used to calculate the estimated rotation vector ϕ for the period, which is also in the Appendix, using Eq. (7). The estimated updating quaternion $\hat{q}(h)$ is then given by the relation

$$\hat{q}(h)=C,\phi S$$

where $C=\cos(½\phi_0)$, $S=(1/\phi_0)\sin(½\phi_0)$, and $\phi_0=(\phi\cdot\phi)^{½}$.

If the error in $\hat{q}(h)$ is represented by $\tilde{q}(h)$, then

$$\tilde{q}(h)=q(h)*\hat{q}(h)^{-1}:$$

$$\tilde{q}(h)=\begin{bmatrix}\tilde{q}_0\\ \tilde{q}_1\\ \tilde{q}_2\\ \tilde{q}_3\end{bmatrix}=\begin{bmatrix}q_0\\ q_1\\ q_2\\ q_3\end{bmatrix}*\begin{bmatrix}C\\ -S\phi_x\\ -S\phi_y\\ -S\phi_z\end{bmatrix}$$
$$=\begin{bmatrix}q_0C-S\{-q_1\phi_x-q_2\phi_y-q_3\phi_z\}\\ q_1C-S\{q_0\phi_x-q_3\phi_y+q_2\phi_z\}\\ q_2C-S\{q_3\phi_x+q_0\phi_y-q_1\phi_z\}\\ q_3C-S\{-q_2\phi_x+q_1\phi_y+q_0\phi_z\}\end{bmatrix} \tag{8}$$

Inspection of $q(h)$ and ϕ shows that q_2, q_3, ϕ_y, and ϕ_z are all periodic with frequency equal to the coning frequency, and by inspection of Eq. (8), it can be seen that $\tilde{q}_2$ and $\tilde{q}_3$ are periodic; they contribute a reciprocating error. We are interested in the nonperiodic errors in $\hat{q}(h)$, because these act to produce a drift rate error in the updated quaternion $\hat{Q}(T+h)$. Such terms may be seen to occur in $\tilde{q}_1$. By inspection of the values of the terms q_2, q_3, ϕ_y, and ϕ_z, it may be seen that $q_3\phi_y-q_2\phi_z=0$; therefore

$$\tilde{q}_1=q_1C-Sq_0\phi_x$$

Similarly, the scalar component $\tilde{q}_0$, and thus the magnitude of the error rotation, are also found to be nonperiodic.

We are only interested in the first order of the error terms, so we may put $C=1$, and $S=½$; also, we can neglect the $\sin^2(½a)\sin^2(½\omega h)$ term in q_0, i.e., we may put $q_0=1$; therefore

$$\tilde{q}_1=q_1-½\phi_x$$

The error quaternion $\tilde{q}(h)$ may be written $\cos(½\phi_\epsilon)$, $e\sin(½\phi_\epsilon)$, where e is the unit vector defining the axis of the error rotation, and ϕ_ϵ is the magnitude of the error rotation.

Thus the vector part $\tilde{q}$ of $\tilde{q}(h)$ is $e\sin(½\phi_\epsilon)$, and if ϕ_ϵ is small, then, approximately, $e\phi_\epsilon=2\tilde{q}$.

In this example, the error drift rate appears on the x axis, so we may put $\phi_\epsilon=2\tilde{q}_1$. Therefore

$$\phi_\epsilon=2q_1-\phi_x$$

Inserting the values of q_1 and ϕ_x, obtained from the Appendix:

$$\phi_\epsilon=\frac{1}{2}a^2\left\{\omega h-\sin(\omega h)-16\sin\left(\frac{1}{3}\omega h\right)\sin^2\left(\frac{1}{6}\omega h\right)\times\left[X\cos\left(\frac{1}{3}\omega h\right)+Y\right]\right\}$$

i.e.,

$$\phi_\epsilon=\frac{1}{2}a^2\left\{\omega h+(2X-1)\sin(\omega h)+(2X-8Y)\sin\left(\frac{1}{3}\omega h\right)\right\}$$

If this is expanded by power series approximations for the sine terms, we get the following coefficients of (ωh):

$$(\omega h):\quad 0$$

$$(\omega h)^3:\quad \frac{-a^2(X+Y-9/8)}{864}$$

$$(\omega h)^5:\quad \frac{a^2(3X+Y-81/40)}{486}$$

$$(\omega h)^7:\quad \frac{-a^2(23X+3Y-729/56)}{131{,}220}$$

If the coefficients of $(\omega h)^3$ and $(\omega h)^5$ are set to zero, then we get $X=0.45$, $Y=0.675$, and

$$\phi_\epsilon=\frac{a^2(\omega h)^7}{204{,}120}$$

which corresponds to a drift rate of approximately $½a^2\omega^7h^6\times10^{-5}$.

In this scheme, we are analyzing the error in the updating quaternion $\hat{q}(h)$, whereas the quantity of real importance is the error in the updated quaternion $\hat{Q}(T+h)$. The errors in $\hat{q}(h)$ amount to a "drift" error about the coning body axis, together with reciprocating errors (at cone frequency), about the other body axes. If we wished to obtain the error in the updated quaternion $\hat{Q}(T+h)$, these errors in $\hat{q}(h)$ would have to be transformed to the reference axes. However, the angles of coning with which the analysis is concerned are small, so it is not unreasonable to approximate the actual errors [in $\hat{Q}(T+h)$] by those occurring with respect to the body axes.

Rotation Vectors in Other Algorithms

This method of analysis may be used to obtain some of the more familiar algorithms. If it is assumed that the gyro outputs follow a square law during the update interval, then c in Eq. (3) equals zero, and Eq. (6) becomes

$$\boldsymbol{\phi} = \boldsymbol{a}h + \boldsymbol{b}h^2 + (1/6)h^3(\boldsymbol{a}\times\boldsymbol{b}) \tag{9}$$

If two gyro samples $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are taken at the middle and end of the update interval, the new values of $\boldsymbol{a}$ and $\boldsymbol{b}$ may be found, and substituted into Eq. (9), giving

$$\boldsymbol{\phi} = \boldsymbol{\theta} + 2/3(\boldsymbol{\theta}_1 \times \boldsymbol{\theta}_2) \tag{10}$$

This is Jordan's[2] "preprocessor" algorithm.

If the gyro output $\boldsymbol{\theta}$, taken over the whole update interval, and the previous gyro output $\boldsymbol{\theta}'$, from the previous interval, are used, the resulting rotation vector is found to be

$$\boldsymbol{\phi} = \boldsymbol{\theta} + (1/12)(\boldsymbol{\theta}' \times \boldsymbol{\theta}) \tag{11}$$

This, in conjunction with third-order expansions for C and S in the updating quaternion, is equivalent to the third-order algorithm of McKern.[4]

If the gyro outputs are assumed linear over the update interval, the angular velocity is thus assumed constant and about a fixed axis; in this case, the rotation vector is simply

$$\boldsymbol{\phi} = \boldsymbol{\theta} \tag{12}$$

This, in conjunction with a first- or second-order expansion of C and S, gives the conventional first- or second-order algorithm.

The same analysis of errors in coning may also be applied to these algorithms, by calculating the corresponding values of ϕ_x for each, then obtaining ϕ_ϵ. The results of this exercise give the following small-angle coning drift rates: Two samples per update algorithm, Eq. (10), $a^2\omega^5h^4/960$; one sample plus previous sample algorithm, Eq. (11), $a^2\omega^5h^4/60$; and one sample only algorithm, Eq. (12), $a^2\omega^3h^2/12$.

Simulations were carried out to verify these results, for the case of 1-deg half-angle coning. Third-order series approximations were used in calculation of the updating quaternion. Drift errors appearing in the reference to body quaternion were found to be in close agreement with the above expressions. Figures 1 and 2 show the coning performance of the various algorithms.

Gyro Data Quantization Effects

A series of coning simulations was carried out for one-plus-previous sample, two sample, and three sample algorithms, with simulated quantization of gyro outputs. Update rates were 100 Hz, 50 Hz, and the range 15-50 Hz, respectively. The range of quantization used was from 0.3 to 30 arc seconds.

It was found that algorithm performance at low coning frequencies was limited by the quantization effect, and at higher frequencies, by the bandwidth effect as indicated in Fig. 1. The low-frequency quantization effect error was essentially an angular offset, varying in direction, of magnitude approximately equal to, or slightly less than, the

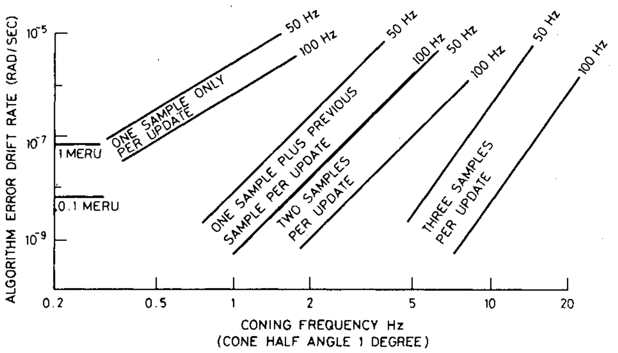

Fig. 1 Error drift rates of algorithms in coning (50- and 100-Hz update rates).

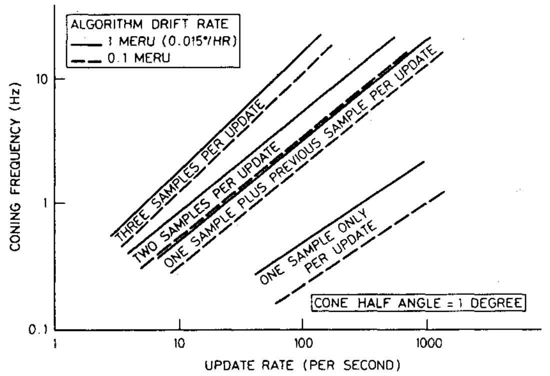

Fig. 2 Performance of algorithms in coning.

quantization magnitude. As the coning frequency was increased, an underlying drift rate became apparent, until at the higher frequencies, performance was essentially as indicated in Fig. 1. The smaller the quantization, the lower was the frequency (and magnitude) at which the drift rate could be detected.

Performance degradation caused by quantization effects was found to be substantially the same for all the algorithms tested.

Quaternion Updating

This is the calculation of the updating quaternion $q(h)$:

$$q(h) = C, \boldsymbol{\phi} S$$

where $C = \cos(\frac{1}{2}\phi_0)$, $S = (1/\phi_0)\sin(\frac{1}{2}\phi_0)$, $\phi_0 = (\boldsymbol{\phi}\cdot\boldsymbol{\phi})^{\frac{1}{2}}$, and the application of "quaternion multiplication" to give the updated reference quaternion:

$$Q(T+h) = Q(T) * q(h)$$

For a given value of $\boldsymbol{\phi}$, the above result is exact, subject to calculation of C and S. These can be obtained from the series expansions of sine and cosine. The errors arising from approximations in calculating C and S may be referred to as truncation errors—they arise from truncation of the sinz and cosine series. In the case of a true fixed axis rotation, the rotation vector is equal to the gyro output vector (all cross-product terms in the rotation vector solution are zero), and the primary source of algorithm error is the truncation error.

This was discussed by Wilcox,[5] who showed that the third-order approximation gave rise to errors of $(\phi_0^4/480)\phi_i$ per calculation; he also proposed a modified second-order approximation which gave errors $(\phi_0^4/720)\phi_i$ $(i = x,y,z)$.

Modified Fourth-Order Approximation

A modified fourth-order approximation has

$$C = 1 - \frac{\phi_0^2}{8} + \frac{\phi_0^4}{480} \quad \text{and} \quad S = \frac{1}{2} - \frac{\phi_0^2}{48}$$

giving drift errors per calculation of $(\phi_0^6/40{,}320)\phi_i$. This may be derived by a method similar to that used by Wilcox to obtain his modified second-order approximation: drift error per update $=2(S_N C_\infty - C_N S_\infty)\phi_i$ [Ref. 5, Eq. (49)]; the fourth-order approximation includes terms to ϕ_0^4 in C, and terms to ϕ_0^2 in S. If we put

$$C_4 = 1 - \frac{\phi_0^2}{2^2 2!} + k\frac{\phi_0^4}{2^4 4!} \quad \text{and} \quad S_4 = \frac{1}{2} - \frac{\phi_0^2}{2^3 3!}$$

then it is found that $k=0.8$ gives a zero coefficient of ϕ_0^4, and the drift errors per calculation are $(\phi_0^6/2^4 7!)2\phi_i$.

As in the modified second-order method, the reduction in drift error is accompanied by an increase in scale error; the scale error is given by $C_N^2 + \phi_0^2 S_N^2 - 1$, which in this case is equal to $\phi_0^4/480$. The scale error is removed periodically by normalization.

Figure 3 illustrates the tradeoff between update rates and input angular velocity for truncation error drift rates of 1 and 0.1 milli-earth-rate-unit (Meru), for various orders of approximation in C and S. For the usual 50 to 100 Hz updating, drift errors only become significant at high input rates, i.e., large rotation vector magnitudes; e.g., a 200-deg/s fixed axis rotation updated at 50 Hz (4 deg per update rotation vector magnitude) using third-order or modified second-order approximations would give an error drift rate of about 1 Meru. The potential high rate capability of the laser gyro might require a higher-order approximation if used in a high rate application.

The improved coning performance of the three-sample algorithm allows a lower iteration rate than is required for other algorithms in a given environment. This will tend to increase the rotation vector magnitude, and therefore the truncation error, per update. Thus, if low iteration rates are contemplated, consideration should also be given to increasing the order of approximation used for C and S.

Random Angular Vibration Environment

An evaluation of comparative algorithm performance was carried out for a simulated random angular vibration environment.

Sequences of zero mean, unit variance samples were passed through a second-order low-pass filter, to generate a power spectral density as shown in Fig. 4. These were scaled and applied as increments of rotation to the body axes, to simulate gyro outputs. This was arranged in such a way that the same simulated environment could be repeatedly applied for updating at rates of up to 800 Hz. The 800-Hz updating runs were used as baselines against which other runs at 50-Hz update rate were compared. At a 25-Hz rate, the "errors" in the reference to body quaternion, relative to the 800-Hz reference to body quaternion at the corresponding times, were calculated.

In these runs (of up to 60 s duration), it was not possible to distinguish any drift rate: The errors were angular offsets, having varying magnitude and direction. Magnitudes of the angular offsets, outputted at 25 Hz, were averaged over 6 s periods, with 23 of these periods for each algorithm. The averages and the ranges of offset over these 23 periods are shown in Fig. 5. It can be seen that the three sample algorithm has considerably less offset than the two sample algorithm, which has less offset than the single-plus-previous sample algorithm. The single sample only algorithm was not tested.

Computer Loading

Calculation of the three sample rotation vector requires 18 multiplications and 15 additions; the two sample calculation requires nine multiplications and six additions. Quaternion updating, using the modified second-order method for C and S, requires 20 multiplications (other than powers of 2) and 15 additions; using third order requires 23 multiplications and 16 additions. Higher orders for C and S require approximately two extra multiplications and one extra addition for each increase in order. Allowance must be made for normalization (eight multiplications and four additions) of the updated quaternion to be carried out at a low rate.

Consideration of these figures shows that the three sample algorithm has a computer loading approximately 30% greater than the two sample or the one-plus-previous sample algorithm for a given iteration rate.

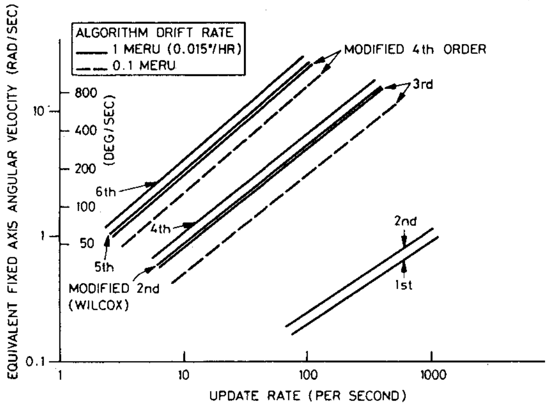

Fig. 3 Truncation errors for various approximation orders.

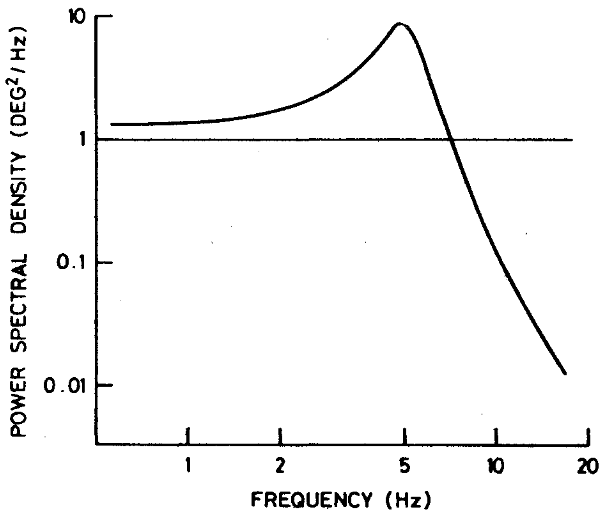

Fig. 4 Random vibration input to all axes.

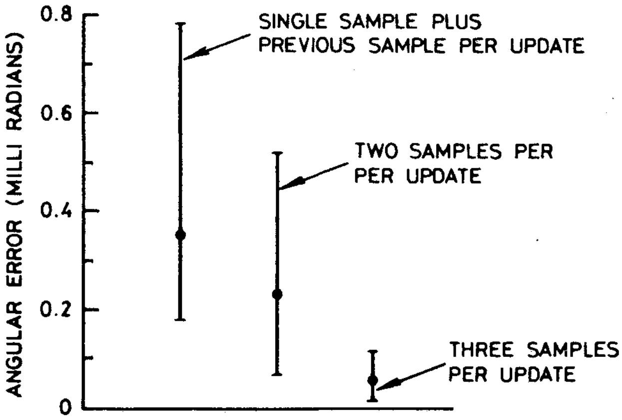

Fig. 5 Errors of algorithms in vibration test.

Conclusion

The technique of partitioning the attitude updating calculations into rotation vector estimation followed by quaternion multiplication has been used to derive a new attitude algorithm. This gives greatly improved performance in a coning environment and some improvement in performance in a random angular vibration environment, in comparison with current methods, at the expense of approximately 30% greater computer loading per update.

Alternatively, for comparable coning environment performance, this algorithm requires a lower iteration rate than current methods. Consequently the computer loading, in terms of multiplications per second, may be approximately halved.

In a purely fixed axis rotation environment, the three sample per update algorithm accuracy is the same as that of a two or a one sample algorithm. However, in any practical application, there will be some combination of fixed axis and coning rotation. The relative proportions of these components may influence the choice of algorithm.

The choice of order of approximations in the estimation of the updating quaternion should also be influenced by the expected environment: The larger is the magnitude of the rotation vector, the higher must be the order of approximation for a given accuracy. The use of higher orders is not very expensive of computation; however, for a case where the computation load is critical, a modified fourth-order approximation has been derived, which gives drift performance almost as good as fifth or sixth order.

Appendix: Coning Motion

Consider a body with body axes x, y, z, and a set of reference axes X, Y, Z, having a common origin O, at some time (t).

Let the relationship of reference to body axes be obtained by a rotation through an angle a about a line OL in the YZ and yz planes.

If OL is given an angular velocity ω about X, the axis x will describe a cone of half-angle a about X, and the axes y and z will experience oscillatory motions about Y and Z, respectively. The rotation vector representing the rotation from reference to body axes is of magnitude a, and direction along OL. If the (arbitrary) time origin is chosen with OL along the Y axis, then this rotation vector has components: $[0, a\cos(\omega t), a\sin(\omega t)]$. The reference to body quaternion $Q(t)$ is therefore

$$Q(t)=\begin{bmatrix}\cos(\frac{1}{2}a)\\ 0\\ \sin(\frac{1}{2}a)\cos(\omega t)\\ \sin(\frac{1}{2}a)\sin(\omega t)\end{bmatrix} \tag{A1}$$

Let $q(h)$ represent the rotation of the body from its position at time (t) to its position at time $(t+h)$; then

$$Q(t+h)=Q(t)*q(h)$$

i.e.,

$$q(h)=Q(t)^{-1}*Q(t+h)$$

this gives

$$q(h)=\begin{bmatrix}q_0\\ q_1\\ q_2\\ q_3\end{bmatrix}=\begin{bmatrix}1-2\sin^2(\frac{1}{2}a)\sin^2(\frac{1}{2}\omega h)\\ -\sin^2(\frac{1}{2}a)\sin(\omega h)\\ -\sin(a)\sin(\frac{1}{2}\omega h)\sin[\omega(t+\frac{1}{2}h)]\\ \sin(a)\sin(\frac{1}{2}\omega h)\cos[\omega(t+\frac{1}{2}h)]\end{bmatrix} \tag{A2}$$

The quaternion differential equation is given by

$$\dot{Q}(t)=\frac{1}{2}Q(t)*[\omega(t)]^B$$

therefore

$$[\omega(t)]^B=2Q^{-1}(t)*\dot{Q}(t)$$

where $[\ ]^B$ signifies body axes coordinates.

The result of performing the arithmetic is

$$[\omega(t)]^B=\begin{bmatrix}-2\omega\sin^2(\frac{1}{2}a)\\ -\omega\sin(a)\sin(\omega t)\\ \omega\sin(a)\cos(\omega t)\end{bmatrix} \tag{A3}$$

The output vector of a triad of rate integrating gyroscopes sensing body axis movements between time (t) and $(t+h)$ is

$$\theta(h)=\int_t^{t+h}[\omega(\tau)]^B d\tau$$

$$=\begin{bmatrix}-2\omega h\sin^2(\frac{1}{2}a)\\ -2\sin(a)\sin(\frac{1}{2}\omega h)\sin[\omega(t+\frac{1}{2}h)]\\ 2\sin(a)\sin(\frac{1}{2}\omega h)\cos[\omega(t+\frac{1}{2}h)]\end{bmatrix} \tag{A4}$$

When three gyro sample sets per update period are taken, the values may be obtained from Eq. (A4). They are given by

$$\begin{bmatrix}-E\\ -F\sin[\omega(t+\alpha h)]\\ F\cos[\omega(t+\alpha h)]\end{bmatrix}$$

where $E=\frac{2}{3}\omega h\sin^2(\frac{1}{2}a)$, $F=2\sin(a)\sin[(1/6)\omega h]$, and $\alpha=1/6$, $1/2$, $5/6$ for θ_1, θ_2, θ_3, respectively. If these values are inserted into Eq. (7), then, after manipulation, and assuming small a,

$$\phi=\begin{bmatrix}\phi_x\\ \phi_y\\ \phi_z\end{bmatrix}=\begin{bmatrix}-\frac{1}{2}a^2\left\{\omega h-16\sin^2\left(\frac{1}{6}\omega h\right)\sin\left(\frac{1}{3}\omega h\right)\times\left[X\cos\left(\frac{1}{3}\omega h\right)+Y\right]\right\}\\ -2a\left\{\sin\left(\frac{1}{2}\omega h\right)+G\right\}\sin\left[\omega\left(t+\frac{1}{2}h\right)\right]\\ 2a\left\{\sin\left(\frac{1}{2}\omega h\right)+G\right\}\cos\left[\omega\left(t+\frac{1}{2}h\right)\right]\end{bmatrix} \tag{A5}$$

where

$$G=\frac{1}{3}a^2(X+Y)\omega h\sin\left(\frac{1}{6}\omega h\right)\sin\left(\frac{1}{3}\omega h\right)$$

References

[1]Bortz, J.E., "A New Mathematical Formulation for Strapdown Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-7, No. 1, Jan. 1971, pp. 61-66.

[2]Jordan, J.W., "An Accurate Strapdown Direction Cosine Algorithm," NASA TN D-5384, Sept. 1969.

[3]Gilmore, J.P., "Modular Strapdown Guidance Unit with Embedded Microprocessors," *Journal of Guidance and Control*, Vol. 3, Jan. 1980, pp. 3-10.

[4]McKern, R.A., "A Study of Transformation Algorithms for Use in a Digital Computer," MIT, Cambridge, Mass., M.S. Thesis T493, Jan. 1968.

[5]Wilcox, J.C., "A New Algorithm for Strapped-Down Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-3, No. 5, Sept. 1967, pp. 796-802.