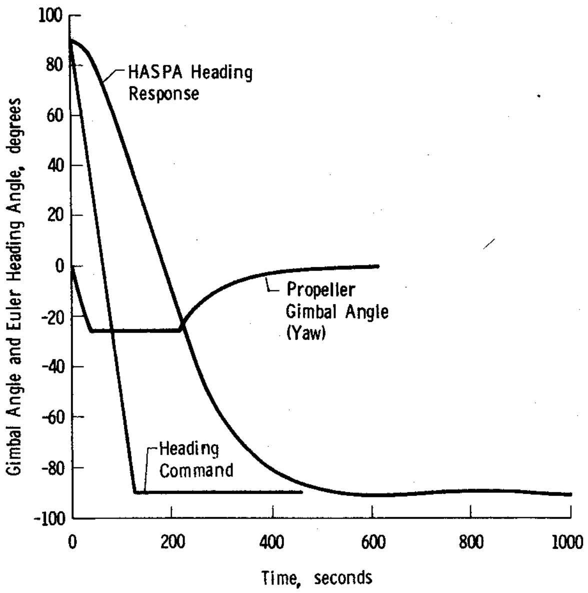

Fig. 3 HASPA nonlinear simulation response to 180-deg heading command – aero-acceleration equations (Table 1).

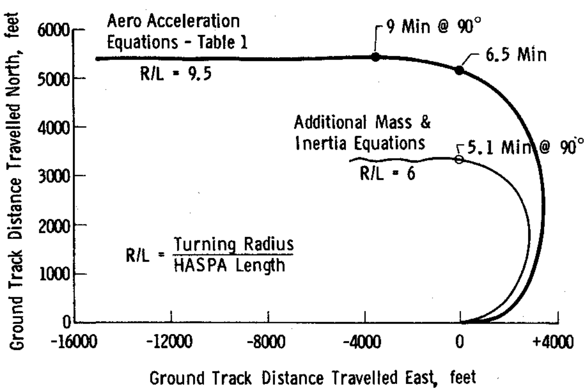

Fig. 4 HASPA nonlinear simulation response to 180-deg heading command.

equation given in Eq. (1) above. It is now believed that the equations for HASPA and similar LTA vehicles should be based on treating the apparent masses and inertias as aerodynamic acceleration force and moment derivatives as in Table 1 and Eq. (2).

Differences in the HASPA dynamics (airframe plus autopilot) as predicted by the two different sets of equations were evaluated by deriving linear transfer functions from the equations of Table 1 and from the earlier equations. From these transfer functions the Nichols plot of Fig. 1, was obtained wherein increasing stability margin is indicated by increasing distance of the frequency response locus from the point (−180 deg, 0 dB). Stability margins using the equations of Table 1 are clearly improved over the margins predicted with the additional mass and inertia form of the equations.

A 180-deg turn was simulated with both forms of the equations, including all control and aerodynamic nonlinearities. With the additional mass and inertia equations, the results of Fig. 2 were obtained. The turn command was applied at a rate of 1.4 deg/s and the resulting gimbal response [linear law: $\delta = (\psi_{\text{ref}} - \psi_b) + 75(\dot{\psi}_{\text{ref}} - \dot{\psi}_b)$] and heading angle are shown. The heading angle exhibits limit cycles of approximately ±3-deg amplitude at a frequency of 0.024 rad/s approximately 320 s after starting to turn. Figure 3 shows the same maneuver using the equations of Table 1; the turn is completed in approximately 540 s. Limit cycling is barely perceptible after the turn is complete.

Another difference between the dynamic predictions is indicated in Fig. 4, which shows the dirigible ground track in response to the same input turn command as in Figs. 2 and 3. The response from the Table 1 equations is more sluggish than the response shown by the additional mass and inertia equations ($R/L = 9.5$ and $R/L = 6.0$, respectively).

The increased stability predicted by the equations of Table 1 makes the dynamic system less sensitive to large variations which may be encountered in estimating the aerodynamic data. Also, the improved stability margins are beneficial because they minimize attitude limit cycling.

Conclusions

The form of the equations of motion for airships and the manner in which the apparent additional mass and inertias are treated can be reliably based on the proven standard equations of motion for submarines. The apparent masses and inertias are really aerodynamic force and moment acceleration derivatives and should be treated as shown in Table 1. For HASPA, this treatment of the equations predicts greater stability margins than was forecast with the equations[1] formerly used.

References

[1]Hookway, R.O. and Pretty, J.R., "HASPA Flight Control Concepts," AIAA Paper No. 75-942, AIAA Lighter Than Air Technology Conference, Snowmass, Colo., July 15-17, 1975.

[2]Clark, J.N., Jr., "Derivation and Application of Equations of Motion for Buoyant and Partially Buoyant Air Vehicles," U.S. Navy, Naval Air Development Center, Warminster, Pa., Tech. Memo. No. VT-TM-1716, Feb. 1976.

[3]Scales, S.H. and McComas, C.B., "HASPA Demonstration Program – Final Report," Martin Marietta Corporation, Denver Division, Denver, Colo., Sept. 1977.

[4]Lindgren, A.G., Cretella, D.B., and Bessacini, A.F., "Dynamics and Control of Submerged Vehicles," *ISA Transactions,* Vol. 6, No. 4, 1967, pp. 335-346.

[5]Gertler, M. and Hagen, G. R., "Standard Equations of Motion for Submarine Simulation," Naval Ship Research and Development Center, Bethesda, Md., Rept. 2510, June 1967.

General Expression for a Three-Angle Rotation Matrix

James M. Wilkes*

White Sands Missile Range, N. Mex.

THE solution on a digital computer of the equations of motion of a system of rigid bodies requires frequent computations of the orthogonal matrices relating various orthonormal frames introduced in the formulation of the problem. The usual method is to relate any two such frames (both assumed to be right-handed) by a sequence of, at most, three single-axis rotations, so that the complete transformation matrix R is a matrix product of the form

$$R = R_i(a) R_j(b) R_k(c) \tag{1}$$

where $R_n(x)$ denotes a rotation about axis n ($n = 1,2,$ or 3) through an angle x. A simple and efficient method for the generation of matrices of the form of Eq. (1) is thus of considerable interest.

Two fairly recent papers[1,2] have presented useful algorithms for the solution of the stated rotation matrix

Received April 3, 1978.

Index categories: Analytical and Numerical Methods; LV/M Simulation; Spacecraft Simulation.

*Physicist, Simulation Group (TE-PC).

generation problem. The procedure described in Ref. 1, though simple and quite general, has the slight disadvantage (as far as efficiency is concerned) of requiring three calls to its FORTRAN subroutine in order to generate a single matrix R of the form of Eq. (1), whereas the method presented in Ref. 2, though very efficient for the transformation of the components of a vector from one frame to another by a matrix of the form of Eq. (1), does not actually produce the elements of the complete transformation matrix R.

In this Note we derive a general expression for R, from which its nine elements can be read off as functions of the rotation angles a, b, and c and the rotation axes i, j, and k. Our use of the phrase "general expression for R" must be slightly qualified, however, as follows. It is easily seen that there are 27 distinct types of matrices of the form of Eq. (1), since any one of the three single-axis rotation matrices can describe a rotation about any one of three coordinate axes. However, types for which any two consecutive single-axis rotations (or all three of them) occur about the same axis, reduce the matrix R to one of the three forms:

$$R=R_i(a)R_k(b)R_k(c)\equiv R_i(a)R_k(b+c), i\neq k \tag{2}$$

$$R=R_j(a)R_j(b)R_k(c)\equiv R_j(a+b)R_k(c), j\neq k \tag{3}$$

$$R=R_k(a)R_k(b)R_k(c)\equiv R_k(a+b+c) \tag{4}$$

In these cases, the original three-rotation sequence of Eq. (1) degenerates to either a two-rotation sequence of type (2) or (3), or a single-rotation sequence of type (4), where we have used the well-known fact that successive rotations about a single axis are equivalent to one rotation about that axis through the sum of the angles. In deriving our expression for R, we specifically exclude matrices of types (2), (3), and (4) [of which there are 15 total; six of type (2), six of type (3), and three of type (4)], i.e., we disallow the possibilities $i=j$, $j=k$, and $i=j=k$. The 12 remaining types (all of which describe three-rotation sequences) are contained in our general expression for R, and separate into two classes, one class for which $i=k$, and one class for which $i\neq k$. We then indicate how the resulting algorithms can be used to compute the degenerate forms (2), (3), and (4) as well.

Our derivation uses the fact that the set of all 3×3 matrices constitutes a vector space of dimension $n=9$. The standard basis of this vector space is a set of nine matrices E_{ij}, $i,j=1,2,3$, with the elements of matrix E_{ij} defined by:

$$E_{ij}(k,l)\equiv\delta_{ik}\delta_{jl} \tag{5}$$

where δ_{ij} is the Kronecker delta symbol ($\delta_{ij}=1$ if $i=j$, and $\delta_{ij}=0$ if $i\neq j$). That is, E_{ij} is the matrix having the entry 1 in its ith row and jth column, and the entry 0 elsewhere. As examples,

$$E_{23}=\begin{bmatrix}0&0&0\\0&0&1\\0&0&0\end{bmatrix}$$

and the identity matrix may be written as

$$I=E_{11}+E_{22}+E_{33}=\sum_\mu E_{\mu\mu} \tag{6}$$

where all summations are understood to be from 1 to 3. An arbitrary 3×3 matrix A can always be written as a linear combination of the basis matrices E_{ij}, viz,

$$A=\sum_\mu\sum_\nu A(\mu,\nu)E_{\mu\nu} \tag{7}$$

where $A(\mu,\nu)$ is the entry in the μth row and νth column of A. It should be noted that in our notation a subscript labels a particular matrix, while the elements of a matrix are labeled by indicating the row and column in parentheses, i.e., $R_n(x)$ is the value at x of the matrix R_n, while $R_n\ (i,j)(x)$ is the value of the ij element of R_n at x.

Using Eq. (5), we deduce the following important multiplication rule for any two basis matrices:

$$(E_{ij}E_{kl})(m,p)=\sum_\mu E_{ij}(m,\mu)E_{kl}(\mu,p)$$

$$=\sum_\mu(\delta_{im}\delta_{j\mu})(\delta_{k\mu}\delta_{lp})$$

$$=\delta_{im}\delta_{jk}\delta_{lp}=\delta_{jk}(\delta_{im}\delta_{lp})=\delta_{jk}E_{il}(m,p)$$

By the definition of equality of two matrices A and B, i.e., $A=B$ if and only if $A(m,p)=B(m,p)$ for all m and p, it follows from the last equation that

$$E_{ij}E_{kl}=\delta_{jk}E_{il} \tag{8}$$

At this point we note that a single-axis rotation matrix $R_n(x)$, $n=1,2$, or 3, can be written as the following linear combination of the basis matrices:

$$R_n(x)=\cos x I+(1-\cos x)E_{nn}+\sin x\sum_\mu\sum_\nu\epsilon_{n\mu\nu}E_{\mu\nu} \tag{9}$$

where ϵ_{ijk} is the completely skew-symmetric permutation symbol defined by:

$$\epsilon_{ijk}=\begin{cases}+1 \text{ if } ijk \text{ is } 123, 312, \text{ or } 231\\ 0 \text{ if any two indices are the same}\\ -1 \text{ if } ijk \text{ is } 132, 213, \text{ or } 321\end{cases} \tag{10}$$

$$=(i-j)(j-k)(k-i)/2 \tag{11}$$

For example, if $n=1$, then by Eqs. (9), (10), and (6),

$$R_1(x)=\cos xI+(1-\cos x)E_{11}+\sin x\sum_\mu\sum_\nu\epsilon_{1\mu\nu}E_{\mu\nu}$$

$$=E_{11}+\cos x(E_{22}+E_{33})+\sin x(\epsilon_{123}E_{23}+\epsilon_{132}E_{32})$$

$$=E_{11}+\cos x(E_{22}+E_{33})+\sin x(E_{23}-E_{32})$$

$$=\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}+\cos x\begin{bmatrix}0&0&0\\0&1&0\\0&0&1\end{bmatrix}+\sin x\begin{bmatrix}0&0&0\\0&0&1\\0&-1&0\end{bmatrix}$$

$$=\begin{bmatrix}1&0&0\\0&\cos x&\sin x\\0&-\sin x&\cos x\end{bmatrix}$$

which is the familiar form.

The derivation of the desired expression for R is now a straightforward, but very tedious, algebraic exercise. Our procedure is to write R, defined by Eq. (1), as a linear combination of the basis matrices. The matrix element $R(i,j)$ is then, by Eq. (7), the coefficient of E_{ij} in the resulting expansion.

We begin by substituting the general expression (9) for $R_i(a)$, $R_j(b)$, and $R_k(c)$ into Eq. (1), to obtain

$$R=\left[\cos a I+(1-\cos a)E_{ii}+\sin a\sum_{\alpha}\sum_{\beta}\epsilon_{i\alpha\beta}E_{\alpha\beta}\right]$$
$$\left[\cos b I+(1-\cos b)E_{jj}+\sin b\sum_{\lambda}\sum_{\sigma}\epsilon_{j\lambda\sigma}E_{\lambda\sigma}\right]$$
$$\left[\cos c I+(1-\cos c)E_{kk}+\sin c\sum_{\mu}\sum_{\nu}\epsilon_{k\mu\nu}E_{\mu\nu}\right]$$

Performing the indicated matrix multiplications, one obtains an expression for R containing 27 terms. This expression simplifies somewhat when we disallow the possibilities $i=j$ and $j=k$, as previously discussed (so that $\delta_{ij}=0$, and $\delta_{jk}=0$), and use repeatedly the well-known identity[3]

$$\sum_{\mu}\epsilon_{mn\mu}\epsilon_{pq\mu}=\delta_{mp}\delta_{nq}-\delta_{mq}\delta_{np} \tag{12}$$

The resulting expression for R *is*

$$\begin{aligned}
R=&\cos a\cos b\cos c I+(1-\cos a)\cos b\cos c E_{ii}\\
&+\cos a(1-\cos b)\cos c E_{jj}+\sin a\sin b\cos c E_{ji}\\
&-\sin a(1-\cos b)\cos c\,\epsilon_{ijl}E_{lj}-(1-\cos a)\sin b\cos c\,\epsilon_{ijl}E_{il}\\
&+\sin a\cos b\cos c\,\epsilon_{ijl}(E_{jl}-E_{lj})+\cos a\sin b\cos c\,\epsilon_{ijl}(E_{li}-E_{il})\\
&+\cos a\cos b(1-\cos c)E_{kk}+(1-\cos a)\cos b(1-\cos c)\delta_{ik}E_{ik}\\
&+\sin a\sin b(1-\cos c)\delta_{ik}E_{jk}\\
&-(1-\cos a)\sin b(1-\cos c)\epsilon_{ijk}E_{ik}\\
&+\sin a\cos b(1-\cos c)\epsilon_{ilk}E_{lk}+\cos a\sin b(1-\cos c)\epsilon_{jlk}E_{lk}\\
&+\cos a\cos b\sin c\,\epsilon_{jkl}(E_{lj}-E_{jl})+(1-\cos a)\cos b\sin c\,\epsilon_{kil}E_{il}\\
&-\cos a(1-\cos b)\sin c\,\epsilon_{jkl}E_{jl}+\sin a\sin b\sin c\,\epsilon_{kil}E_{jl}\\
&+\sin a(1-\cos b)\sin c\,\epsilon_{ijl}\sum_{\mu}\sum_{\nu}\epsilon_{jk\nu}E_{\mu\nu}+\cos a\sin b\sin c E_{kj}\\
&+(1-\cos a)\sin b\sin c\,\delta_{ik}E_{ij}+\sin a\cos b\sin c(E_{ki}-\delta_{ik}I)
\end{aligned} \tag{13}$$

The general expression Eq. (13), is now specialized by considering the two possibilities $i=k$, and $i\neq k$, as separate cases:

1) If $i\neq k$, then $\delta_{ik}=0$, and $l=k$ in permutation symbols of the form ϵ_{ijl}, $l=j$ in symbols of the form ϵ_{ilk}, and $l=i$ in symbols of the form ϵ_{jkl}. Since i, j, and k are all different, we may write the identity matrix as $I=E_{ii}+E_{jj}+E_{kk}$. The matrix R in Eq. (13) then simplifies, in this case, to:

$$\begin{aligned}
R=&\cos b\cos c E_{ii}+\epsilon_{ijk}\cos b\sin c E_{ij}-\epsilon_{ijk}\sin b E_{ik}\\
&+(-\epsilon_{ijk}\cos a\sin c+\sin a\sin b\cos c)E_{ji}\\
&+(\cos a\cos c+\epsilon_{ijk}\sin a\sin b\sin c)E_{jj}+\epsilon_{ijk}\sin a\cos b E_{jk}\\
&+(\sin a\sin c+\epsilon_{ijk}\cos a\sin b\cos c)E_{ki}\\
&+(-\epsilon_{ijk}\sin a\cos c+\cos a\sin b\sin c)E_{kj}+\cos a\cos b E_{kk}
\end{aligned} \tag{14}$$

2) If $i=k$, then $\epsilon_{ijk}=0$, and $l=6-(i+j)$ is the integer from the set {1,2,3} different from i and j. In this case i, j, and l are all different, so that $I=E_{ii}+E_{jj}+E_{ll}$. The rotation matrix, Eq. (13), then simplifies to:

$$\begin{aligned}
R=&\cos b E_{ii}+\sin b\sin c E_{ij}-\epsilon_{ijl}\sin b\cos c E_{il}\\
&+\sin a\sin b E_{ji}+(\cos a\cos c-\sin a\cos b\sin c)E_{jj}\\
&+\epsilon_{ijl}(\cos a\sin c+\sin a\cos b\cos c)E_{jl}+\epsilon_{ijl}\cos a\sin b E_{li}\\
&-\epsilon_{ijl}(\sin a\cos c+\cos a\cos b\sin c)E_{lj}\\
&+(-\sin a\sin c+\cos a\cos b\cos c)E_{ll}
\end{aligned} \tag{15}$$

The elements of the two classes of three-rotation sequence matrices are simply the coefficients of the basis matrices in Eqs. (14) and (15). The expressions for these elements can be programmed as a simple FORTRAN subroutine, the inputs to which are the three rotation axes, specified by i, j, and k, and the three angles of rotation a, b, and c. The computation of the rotation matrix elements then requires only a single call to this subroutine.

These algorithms can also be used to calculate the elements of the degenerate matrices defined by Eqs. (2-4). We illustrate this procedure by considering a matrix of type (2). In this case, one simply replaces the angle c by $(b+c)$ and the angle b by 0, and assigns to the index j in the subroutine call the value $j=6-(i+k)$. For example, if $k=1$ and $i=3$, then $j=2$ (the value from the set {1,2,3} different from i and k).

As a final remark, we wish to emphasize the usefulness of the analytical expression, Eq. (11), for computing the value of the permutation symbol as a function of its indices. Our version of a FORTRAN subroutine implementing the preceding algorithms is available upon request.

References

[1] Cupit, C.R., "Rotation Matrix Generation," *Simulation*, Vol. 15, Oct. 1970, pp. 145-147.

[2] Ohkami, Y., "Computer Algorithms for Computation of Kinematical Relations for Three Attitude Angle Systems," *AIAA Journal*, Vol. 14, Aug. 1976, pp. 1136-1137.

[3] Jeffreys, H. and Jeffreys, B.S., *Methods of Mathematical Physics*, 3rd ed., Cambridge, 1962, p. 73.

Numerical Derivatives for Parameter Optimization

David G. Hull*
University of Texas at Austin, Austin, Texas
and
Walton E. Williamson†
Sandia Laboratories, Albuquerque, N.M.

Introduction

THE accuracy with which a minimal point can be computed by a parameter optimization method using numerical derivatives depends on the accuracy with which the derivatives can be computed. Numerican derivatives are computed by differencing the performance index at a known point and a perturbed point. Hence, the problem is to predict the size of the perturbation so that the derivative has the most accuracy. If the perturbation is too large, the derivative is corrupted by truncation error, and if it is too small, by roundoff error. Hence, derivative accuracy is achieved by balancing the effects of these two error sources. In this paper a procedure for determining the perturbation is proposed for both first-order and second-order methods taking advantage

Received April 10, 1978; revision received Oct. 6, 1978.

Index category: Analytical and Numerical Methods.

*Professor, Dept. of Aerospace Engineering and Engineering Mechanics. Member AIAA

†Member of Technical Staff, Aerodynamics Dept. Member AIAA.