pitch angle program given by Eq. (4), where the slow control variable u is given by Eq. (17). Whenever the fundamental assumptions of the two time scale method are met, the fast time variable f and the slow time variable τ (or a) are completely independent of one another. As long as very many resolutions elapse the vehicle will arrive in the final orbit when $\tau = \tau_f$. The vehicle need not have the capability of solving the optimal control problem. Given the values of λ_i and τ_f for the desired transfer, the vehicle only needs access to the current values of a and f to calculate the control functions.

From an operational point of view, the separation of the two time variables has several advantages. Should a thruster fail, for example, the optimal solution *does not change.* Only the τ to t conversion changes as vehicle parameters are altered. Should the vehicle need to return to its original orbit after depositing a payload, the optimal return trajectory will be the simple τ-time reverse of the outbound transfer, as long as very many revolutions elapse on the return leg. In fact, the time transformation is not limited to the constant thrust vehicle we have assumed. *Any arbitrary throttle program $A(t)$ may be used in the time transformation and the same optimal τ-time trajectory results.* The only restriction is that $A(t)$ can vary only "slowly" over the period of one orbit.

Edelbaum, besides being the first to solve the short time scale problem, also addressed the long time scale problem and found a suboptimal solution. In order to find this solution, he assumed that the fast control variable $\vartheta(f)$ was a square wave for any value of u, and, more critically, he assumed that the vehicle *acceleration* (not its thrust) was constant. The first assumption may be defendable as differing only slightly from the optimal ϑ program, but modern attitude control systems are quite capable of executing the optimal $\vartheta(f)$ program. The restriction of Edelbaum's result to constant acceleration vehicles is much more severe. Our work has eliminated this restriction completely.

We have simulated several transfers numerically by using our control law in the full set of Lagrange planetary equations. The method of two time scales is, of course, exact only when an infinite number of revolutions elapse during the transfer, and our solution is exact only in this case. In all simulated transfers, the error in obtaining the desired final conditions a_f and i_f show a first-power law dependence on $1/N$, where N is the number of revolutions which elapse in the transfer. This strongly indicates that the problem of optimal *finite revolution* transfer should be solvable by a perturbation theory attack, starting from our infinite revolution solution. This is a topic of current research.

Conclusions

In this work we have detailed, for the first time, a closed-form solution to the low-thrust circle to circle orbit transfer problem. The solution is exact for an infinite number of revolutions during the transfer. The solution is simple enough to be implemented in the onboard control system of an orbit transfer vehicle. It can also tolerate changes in thrust programs over a very wide range, including loss of thrusters. Finally, this solution holds the promise of success of a perturbation theory approach to finite revolution orbit transfer.

Appendix

To implement our solution the user will need methods to easily generate values of the function φ^{-1}. The function $u = \varphi^{-1}(X)$ is best approximated as a rational function of the variable $Z = X^{-2}$ in the form

$$u = \varphi^{-1}(Z = X^{-2}) \approx \sum_{i=0}^{10} \alpha_i Z^i \Big/ \sum_{i=0}^{10} \beta_i Z^i \qquad \text{(A1)}$$

The variable Z eliminates a singularity in φ. The coefficients α_i and β_i listed in Table A1, yield values for φ^{-1} which are accurate to at least seven figures. This is a Chebyshev fit, not a Taylor series, and should not be truncated.

References

[1]Levin, E., "Low Thrust Transfer Between Circular Orbits," ASME Paper 59-AV-2, 1959.

[2]Shi, Y.Y. and Eckstein, M.C., "Ascent or Descent from Satellite Orbit by Low Thrust," *AIAA Journal,* Vol. 4, Dec. 1966, pp. 2203-2209.

[3]Eckstein, M.C. and Shi, Y.Y., "Low Thrust Elliptical Spiral Trajectories of a Satellite of Variable Mass," *AIAA Journal,* Vol. 5, Aug. 1967, pp. 1491-1494.

[4]Edelbaum, T.N., "Propulsion Requirements for Controllable Satellites," *American Rocket Society Journal,* Vol. 31, Aug. 1961, pp. 1079-1089.

[5]Danby, J.M.A., *Fundamentals of Celestial Mechanics,* MacMillan, New York, 1962, p. 242.

[6]Abramowitz, M. and Segun, I., *Handbook of Mathematical Functions,* Dover, New York, 1965, pp. 591-592.

[7]Bryson, A.E. and Ho., Y.C., *Applied Optimal Control,* Wiley, New York, 1975, p. 76.

[8]Alfano, S., *Low Thrust Orbit Transfer,* Master's Thesis Department of Aeronautics and Astronautics, Air Force Institute of Technology, Ohio.

The Number of Multiplications Required to Chain Coordinate Transformations

V.N. Dvornychenko*
South Pasadena, California

Introduction

MANY applications in guidance and control require transformations of coordinates and/or the rotation of vectors. When these computations are performed in the inner loop of a real-time guidance system, the computations can be quite demanding, even allowing for the great advances in hardware in recent years. In dealing with this problem, a number of investigators (for example, Ickes[1]) have rediscovered the eminent suitability of quaternions for this application.

Hamilton discovered his celebrated quaternions in 1843. Following the lead of a then recent discovery that complex numbers can be interpreted as rotations in the plane, Hamilton was able to interpret quaternions as, among other things, three-dimensional spatial rotations. The high hopes Hamilton had for the quaternions were not all realized (Ref. 2).† With Gibb's development of his vector analysis (c. 1901), the quaternions experienced a steady erosion in popularity, until rediscovered for guidance and control.

As shown by Ickes,[1] the straightforward multiplication of 3×3 matrices requires 27 multiplications and 18 additions, while straightforward multiplication of quaternions requires 16 and 12, respectively. The principal results of the present paper show that the previous numbers are not minimal: adaptations of Strassen-Winograd algorithms reduce the required number of multiplications to as few as ten. The achievable relief in computational requirements over the conventional quaternion algorithm is about 15%. Of course, the improvement over conventional matrix multiplication is much higher (approximately a factor of two).

Submitted July 7, 1983; revision received March 23, 1984.

*With Northrop EMD, Anaheim, Calif. Member AIAA.

†Written by a competent mathematician, this work contains valuable mathematical exposition besides the obligatory biographical material. Chapter VII deals with the quaternions: the motivation behind them, their genesis, Hamilton's very high expectations for them, and their somewhat disappointing fate and the reasons thereof.

Analytical Preliminaries

It is well known that the most general rigid-body displacement consists of a translation plus a rotation and that the rotation can be represented by a 3×3 orthogonal matrix having positive unity determinant. It is not difficult to show that such a matrix can be written in the following "rotation axis plus rotation angle" form:

$$B(\alpha,n)=\begin{bmatrix} \cos\alpha + n_1^2(1-\cos\alpha) & n_1n_2(1-\cos\alpha)+n_3\sin\alpha & n_1n_3(1-\cos\alpha)-n_2\sin\alpha \\ n_1n_2(1-\cos\alpha)-n_3\sin\alpha & \cos\alpha+n_2^2(1-\cos\alpha) & n_2n_3(1-\cos\alpha)+n_1\sin\alpha \\ n_1n_3(1-\cos\alpha)+n_2\sin\alpha & n_2n_3(1-\cos\alpha)-n_1\sin\alpha & \cos\alpha+n_3^2(1-\cos\alpha) \end{bmatrix} \tag{1}$$

where

$$n_1^2+n_2^2+n_3^2=1$$

The transition to quaternion form, i.e., the association

$$B\leftrightarrow Q=[q_1,q_2,q_3,q_4] \tag{2}$$

is effected via the following substitution in Eq. (1):

$$q_1=\cos\frac{\alpha}{2},\ q_2=-n_1\sin\frac{\alpha}{2},\ q_3=-n_2\sin\frac{\alpha}{2},\ q_4=-n_3\sin\frac{\alpha}{2} \tag{3a}$$

For which it will readily appear that

$$q_1^2+q_2^2+q_3^2+q_4^2=1 \tag{3b}$$

which states that Q has unit norm.

The resulting representation of B in terms of Q takes the form

$$B(Q)=\begin{bmatrix} 2(q_1^2+q_2^2)-1 & 2(q_2q_3-q_1q_4) & 2(q_2q_4+q_1q_3) \\ 2(q_2q_3+q_1q_4) & 2(q_1^2+q_3^2)-1 & 2(q_3q_4-q_1q_2) \\ 2(q_2q_4-q_1q_3) & 2(q_3q_4+q_1q_2) & 2(q_1^2+q_4^2)-1 \end{bmatrix} \tag{4}$$

The association symbolized by Eq. (2) would not be very meaningful were it to turn out that it is not preserved across multiplications. That it is preserved is demonstrated by Whittaker,[3] who ardently promoted quaternions in his classic text and who also provides much additional valuable material, such as the expressions for the quaternions in terms of Eulerian angles. (Some of this material is also presented in Ickes,[1] and in a somewhat disguised form‡ in Goldstein.[4])

We can summarize the association between orthogonal matrices and quaternions as follows:

If $\quad B_1=B(Q_1)\quad$ and $\quad B_2=B(Q_2)$

then $\quad B_3=B_2B_1=B(Q_3)\quad$ where $\quad Q_3=Q_2Q_1 \qquad (5)$

‡The fate of the quaternions is dramatically pointed out by noting that Goldstein does not mention the quaternions once, although there are extensive references to Hamilton in connection with the Hamilton-Jacobi theory. Furthermore, Section 4-5 provides a detailed discussion of the Cayley-Klein parameters, which can be shown to be equivalent (isomorphic) to the quaternions. In the second edition (1981), quaternions are grudgingly mentioned in four footnotes.

and quaternion multiplication is explicitly defined by

If

$$Q_1=[q_1,q_2,q_3,q_4]$$
$$Q_2=[q_1',q_2',q_3',q_4']$$
$$Q_3=[q_1'',q_2'',q_3'',q_4''] \tag{6a}$$

and if

$$Q_3=Q_2Q_1$$

then

$$\begin{aligned} q_1''&=q_1q_1'-q_2q_2'-q_3q_3'-q_4q_4' \\ q_2''&=q_1q_2'+q_2q_1'-q_3q_4'+q_4q_3' \\ q_3''&=q_1q_3'+q_2q_4'+q_3q_1'-q_4q_2' \\ q_4''&=q_1q_4'-q_2q_3'+q_3q_2'+q_4q_1' \end{aligned} \tag{6b}$$

Ickes[1] provides a number of interesting variants to the above in the form of a matrix times a vector, etc.

We now turn to the second tool required, the Strassen-Winograd algorithms for multiplication saving. This class of algorithms can be illustrated by the following simple example: Consider the two complex numbers

$$Z_1=a_1+b_1i \quad \text{and} \quad Z_2=a_2+b_2i \tag{7a}$$

Their product is, of course, given by

$$Z_1Z_2=(a_1a_2-b_1b_2)+(a_1b_2+b_1a_2)i \tag{7b}$$

from which it would appear that four real multiplications are required. Somewhat amazingly, it turns out that three are enough:

Let

$$u_1=a_1(a_2+b_2),\quad u_2=(a_1+b_1)b_2,\quad u_3=(b_1-a_1)a_2 \tag{8a}$$

We then have

$$a_1a_2-b_1b_2=u_1-u_2,\quad a_1b_2+b_1a_2=u_1+u_3 \tag{8b}$$

It is not clear to the author who first made the preceding observation, but the principal investigators in the domain of "multiplicative complexity" are V. Strassen (Germany), S. Winograd (USA), V. Pan (USSR), and A. Toom (USSR). Winograd[5] provides a thorough, if somewhat formidable, exposition.

In the next section we shall use the previous ideas to formulate more efficient quaternion multiplication rules and, by extension, more efficient coordinate transformation algorithms.

More Efficient Algorithms

The conventional "row-column" ("high school algorithm") applied to 3×3 matrices requires 27 multiplications and 18 additions. Equation (6) shows that the conventional quaternion multiplication algorithm requires only 16 and 12, respectively. The advantage of using quaternions in a long chain of coordinate transformations is apparent. Further improvements can be achieved by unconventional multiplication algorithms, as we proceed to show. Two such algorithms are presented. The first requires 11 multiplications and 19 additions, the second ten and 26. Since there is as yet no systematic method of deriving these algorithms, no derivations are presented. The author is not aware if the minimum number of multiplications has been established for quaternions (but see Winograd[5]). Indeed, the

Table 1 Figure-of-merit derivation

Algorithm	Multiplications	Additions	Computational units	FOM
Matrix mult. – HSA	27	18	99	1.01
Quat. mult. – HSA	16	12	60	1.67
Fast Q No. 1 [Alg. (1)]	11	19	52	1.92
Fast Q No. 2 [Alg. (2)]	10	26	56	1.75

principal reason for including the second algorithms is that it establishes that 11 multiplications are not minimal, although in most other respects it is inferior to Alg. (1). One shortcoming, for instance, is that Alg. (2) requires mutliplies/divides by two. Although these can usually be implemented as shifts (increment/decrement exponent for floating point), and so were not included in the operation count, their presence nevertheless tends to detract from Alg. (2). A final point: in computing the addition count, an intelligent grouping of terms was always assumed.

Algorithm 1 (11 multiplications, 19 additions; see Eq. (6) for notation, result is for QQ'):

$$u_1=(q_1+q_3)(q_1'+q_2'-q_3'+q_4')$$
$$u_2=(q_1+q_2+q_3+q_4)(q_2'+q_4')$$
$$u_3=(-q_1+q_2-q_3+q_4)(q_1'-q_3')$$
$$v_1=q_1q_4', \qquad v_5=q_1q_3'$$
$$v_2=q_2q_4', \qquad v_6=q_2q_3'$$
$$v_3=q_3q_1', \qquad v_7=q_3q_2'$$
$$v_4=q_4q_1', \qquad v_8=q_4q_2' \tag{9a}$$

then

$$q_1''=u_1-u_2+v_2-v_3+v_5+v_8$$
$$q_2''=u_1+u_3-v_1-v_4+v_6-v_7$$
$$q_3''=-v_2+v_3+v_5+v_8$$
$$q_4''=v_1+v_4+v_6-v_7 \tag{9b}$$

Algorithm 2 (10 multiplications, 26 additions):

$$u_1=(q_1-q_4)(q_1'+q_2'+q_3'-q_4')$$
$$u_2=(q_1+q_2+q_3-q_4)(q_2'+q_3')$$
$$u_3=(-q_1+q_2+q_3+q_4)(q_1'-q_4')$$
$$u_4=(q_1+q_4)(q_1'+q_2'-q_3'+q_4')$$
$$u_5=(q_1+q_2-q_3+q_4)(q_2'-q_3')$$
$$u_6=(-q_1+q_2-q_3-q_4)(q_1'+q_4')$$
$$v_1=q_4q_4', \qquad v_3=q_4q_2'$$
$$v_2=q_3q_4', \qquad v_4=q_3q_2' \tag{10a}$$

then

$$q_1''=\tfrac{1}{2}(u_1-u_2+u_4-u_5)-2v_1$$
$$q_2''=\tfrac{1}{2}(u_1+u_3+u_4+u_6)+2v_2$$
$$q_3''=\tfrac{1}{2}(u_1+u_3-u_4-u_6)+2v_3$$
$$q_4''=\tfrac{1}{2}(-u_1+u_2+u_4-u_5)-2v_4 \tag{10b}$$

A common assumption in real-time computer work is that a multiplication takes three times the time of an addition. Assuming this to be the case and defining a figure-of-merit by means of

FOM = 100/ "computational units"

we obtain Table 1 shown above left.

References

[1] Ickes, B.P., "A New Method for Performing Digital Control Attitude Computations Using Quaternions," *AIAA Journal,* Vol. 8, Jan. 1970, pp. 13-17.

[2] Hankins, T.L., *Sir William Rowan Hamilton,* The Johns Hopkins University Press, Baltimore, Md., 1980, pp. 281-326.

[3] Whittaker, E.T., *A Treatise on the Analytical Dynamics of Particles and Rigid Bodies,* Fourth ed., Cambridge Univeristy Press, New York, 1959.

[4] Goldstein, H., *Classical Mechanics,* Addison-Wesley, Reading, Mass., 1959.

[5] Winograd, S., "Some Bilinear Forms Whose Complexity Depends on the Field of Constants," *Mathematical Systems Theory 10,* 1977, pp. 169-180. Reprinted as part of anthology in *Number Theory in Signal Processing,* J.H. McClellan (ed.), Prentice-Hall, Englewood Cliffs, N.J., 1979, pp. 217-228.