To the Graduate Council:

I am submitting herewith a thesis written by Wei-Ping Huang entitled "On Quaternion Update Algorithm." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

J. C. Hung, Major Professor

We have read this thesis and recommend its acceptance:

Yueh-er Kuo

Marshall Pace

Accepted for the Council:

Vice Chancellor
Graduate Studies and Research

Thesis
79
.H782
cop. 2

ON QUATERNION UPDATE ALGORITHM

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Wei-Ping Huang

August 1979

1397058

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to his major professor and advisor, Dr. J. C. Hung, for his suggesting this area of study, and under whose supervision and direction the work was conducted.

The author would also like to thank Dr. Y. E. Kuo for her directing in mathematics.

Appreciation is extended to Mr. Bill Devan for his carefully correcting the structure and grammar on this thesis. Also, special thanks to Miss Mai-Nei Yu for her cautiously rewriting the original paper.

Deep appreciation is presented to my family, especially my wife, for their constant encouragement during the period of the study.

ABSTRACT

Space vehicle attitude control system performance using a strapdown inertial measurement unit is often limited by the computational burden of the digital computer. The application of quaternion to digital attitude transformation can reduce computation time. This thesis presents two update algorithms using quaternions. These algorithms have a simple expression, and do not involve any trigonometric functions. These algorithms are compared by means of digital simulation test. The first algorithm, developed from the algebra of quaternion, yields higher accuracy than the second algorithm; the second algorithm, derived from the quaternion differential equations, produce a less deviation of unity norm compared to the first.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
II.	QUATERNIONS AS ROTATIONAL OPERATOR	4
	Quaternion Algebra	5
	Quaternions as a Rotational Operator	8
	Quaternion Difference Equation	14
	Quaternion Differential Equations	16
III.	QUATERNION ATTITUDE ALGORITHM	19
	Approach I	19
	Approach II	24
IV.	SIMULATION TEST OF QUATERNION UPDATE ALGORITHMS	44
	Analysis of the Algorithms	44
	Norm Checking Method	48
	Comparison of Computed Input Angles	54
	Comparison of Quaternion Accuracy	66
V.	CONCLUSIONS	89
REFERENCES		90
APPENDIXES		93
I.	DERIVATION OF THE RATE OF CHANGE OF DIRECTION COSINE	94
II.	DERIVATION OF ANGULAR VELOCITY AND ANGULAR ACCELERATION IN TERMS OF ANGULAR INCREMENTS	97
III.	DERIVATION OF EULER ANGLE TRANSFORMATION	99
IV.	SUM OF UPDATE QUATERNION FOR ALGORITHM I	103
V.	SUM OF UPDATE QUATERNION FOR ALGORITHM II	105
VI.	EULER ANGULAR INCREMENTS	107
VII.	UPDATE QUATERNION FOR ALGORITHM I	111
VIII.	UPDATE QUATERNION FOR ALGORITHM II	113
VITA		115

LIST OF TABLES

TABLE		PAGE
1.	Comparison of Algorithm I and II	45
2.	Difference between Algorithm I and Algorithm II	46
3.	Result of Sum of Update Quaternion Elements for Algorithm I, Using $\Delta\theta_{x1}$ = 0.001°, $\Delta\theta_{x2}$ = 0.002°, $\Delta\theta_{y1}$ = $\Delta\theta_{y2}$ = 0.0015°, $\Delta\theta_{z1}$ = $\Delta\theta_{z2}$ = 0.0015° (Time Used: 3.19 sec in IBM 360) . . .	51
4.	Result of Sum of Update Quaternion Elements for Algorithm II, Using $\Delta\theta_{x1}$ = 0.001°, $\Delta\theta_{x2}$ = 0.002°, $\Delta\theta_{y1}$ = $\Delta\theta_{y2}$ = 0.0015°, $\Delta\theta_{z1}$ = $\Delta\theta_{z2}$ = 0.0015° (Time Used: 1.93 sec in IBM 360) . . .	52
5.	Comparison of Algorithms I and II in Special Cases	55
6.	Result of Input and Euler Angular Increments Using $\Delta\theta_{x1}$ = $\Delta\theta_{x2}$ = 0.01°, $\Delta\theta_{y1}$ = $\Delta\theta_{y2}$ = $\Delta\theta_{z1}$ $\Delta\theta_{z2}$ = 0°, and n Equals to 1 to 40	62
7.	Result of Input and Euler Angular Increments Using $\Delta\theta_{x1}$ = $\Delta\theta_{x2}$ = 0.02°, $\Delta\theta_{y1}$ = $\Delta\theta_{y2}$ = $\Delta\theta_{z1}$ $\Delta\theta_{z2}$ = 0°, and n Equals to 1 to 20	63
8.	Result of Input and Euler Angular Increments Using $\Delta\theta_{x1}$ = $\Delta\theta_{x2}$ = 0.04°, $\Delta\theta_{y1}$ = $\Delta\theta_{y2}$ = $\Delta\theta_{z1}$ $\Delta\theta_{z2}$ = 0°, and n Equals to 1 to 10	64
9.	Result of Update Quaternion Elements of Algorithm I for Set 1 as Input Angular Increments, and n Equals to 1 to 10000 (Time Used: 24.09 sec in IBM 360)	69
10.	Result of Update Quaternion Elements of Algorithm II for Set 1 as Input Angular Increments, and n Equals to 1 to 10000 (Time Used: 18.45 sec in IBM 360)	70
11.	Result of Update Quaternion Elements of Algorithm I for Set 2 as Input Angular Increments, and n Equals to 1 to 10 (Time Used: 3.26 sec in IBM 360)	71
12.	Result of Update Quaternion Elements of Algorithm II for Set 2 as Input Angular Increments, and n Equals to 1 to 10 (Time Used: 3.01 sec in IBM 360)	72

TABLE PAGE

13. Result of Update Quaternion Elements of Algorithm I for Set 3 as Input Angular Increments, and n Equals to 1 to 10000 77

14. Result of Update Quaternion Elements of Algorithm II for Set 3 as Input Angular Increments, and n Equals to 1 to 10000 78

15. Result of Update Quaternion Elements of Algorithm I for Set 4 as Input Angular Increments, and n Equals to 1 to 10 79

16. Result of Update Quaternion Elements of Algorithm II for Set 4 as Input Angular Increments, and n Equals to 1 to 10 80

17. Result of Update Quaternion Elements of Algorithm I for Set 5 as Input Angular Increments, and n Equals to 1 to 10000 81

18. Result of Update Quaternion Elements of Algorithm II for Set 5 as Input Angular Increments, and n Equals to 1 to 10000 82

19. Result of Update Quaternion Elements of Algorithm I for Set 6 as Input Angular Increments, and n Equals to 1 to 10 83

20. Result of Update Quaternion Elements of Algorithm II for Set 6 as Input Angular Increments, and n Equals to 1 to 10 84

LIST OF FIGURES

FIGURE		PAGE
1.	Rotation of a Position Vector about a Unit Vector $\hat{e}$	9
2.	Flow Chart of Computation of Sum of Square of Update Quaternion Elements	49
3.	Comparison of Unity Norm, N(Q), of Algorithms I and II for $\Delta\theta_{x1} = 0.001°$ $\Delta\theta_{x2} = 0.002°$ $\Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1} = \Delta\theta_{z2} = 0.0015°$	53
4.	Block Diagram of Computation of Euler Angular Increments	56
5.	Flow Chart of Comparison of Computed Input and Euler Angular Increments	59
6.	Comparison of Error for Different Angular Increments	65
7.	Block Diagram of Comparison of Quaternion Accuracy	68
8.	Comparison of Update Quaternion Elements for Set 1 as References and Set 2 as Input Incremental Angles of Algorithm I and Algorithm II. The Scale of Curve I and II have been Enlarged	74
9.	Comparison of Update Quaternion Elements for Set 3 as References and Set 4 as Input Incremental Angles of Algorithm I and Algorithm II. The Scale of Curve I and II have been Enlarged	85
10.	Comparison of Update Quaternion Elements for Set 5 as References and Set 6 as Input Incremental Angles of Algorithm I and Algorithm II. The Scale of Curve I and II have been Enlarged	87
11.	Transformation of Rotating of the Coordinate Axis	100

FIGURE PAGE

10.2. Comparison of Q_1 for Set 5 (Ref.) and Set 6 87

10.3. Comparison of Q_2 for Set 5 (Ref.) and Set 6 88

10.4. Comparison of Q_3 for Set 5 (Ref.) and Set 6 88

A-III-1 100

CHAPTER I

INTRODUCTION

The attitude of a rigid body in space can be defined uniquely by three independent parameters, since a rigid body has only three degrees of freedom. A larger number of parameters can be used for the same purpose, but additional relations will exist.

Attitude transformation is required in a strapdown inertial measurement unit. Several different attitude transformation algorithms are available for this purpose. They are (1) the direction cosine algorithm, (2) the Euler angle algorithm, (3) Euler's four parameter algorithms, (4) the quaternion algorithm, (5) the Cayler-Klein algorithm, and (6) the vector representation algorithm. These algorithms are described in References [1] and [2]. The application of quaternions to digital attitude transformation can reduce computation time by more than 40% over the equivalent direction cosine matrix solution [5].

The quaternion, containing four parameters, was first devised by Hamilton in 1843. The advantage of the quaternion lies in its ability to define the rotational relationship between two coordinate systems using only four numbers instead of the nine elements of a direction cosine matrix; this technique requires less computer time and gives less trancation error when both algorithms are operated at the same sampling rate [4].

Hamilton developed quaternions as an extension of complex numbers. These quaternions constitute a four dimensional vector space over the field of real numbers with a basis of four special vectors denoted by 1, $\hat{i}$, $\hat{j}$, and $\hat{k}$. The unit real number 1 represents one direction [9] [15].

This thesis presents two quaternion update algorithms. A brief review of quaternion algebra is made in Chapter II. Then, a rotational transformation of quaternion is derived in Section II.2, so that the quaternion attitude algorithm can be developed. The quaternion difference and differential equations are derived in Sections II.3 and II.4.

Chapter III includes the derivation of two types of quaternion attitude algorithms. The first algorithm is developed from the algebra of quaternion; the second is derived by solving the quaternion differential equation using Taylor series. The advantage of these algorithms is that they require only addition, subtraction, multiplication, division, and square root (only in algorithm I) computational operations; they do not involve any trigonometric functions which will increase computation burden.

Comparisons of these quaternion update algorithms are performed in Chapter IV by means of digital simulation tests. These tests include

1. Test of normality of quaternion.
2. Comparison of the generated error using different input angular increments.

3. Comparison of the precision of updated quaternions.

CHAPTER II

QUATERNIONS AS ROTATIONAL OPERATOR

Quaternion representation is developed from the extension of complex numbers. A complex number can be represented as a 2-dimensional vector. A complex number Z, with real part q_0 and imaginary part q_1, can be represented in the standard form $Z = q_0 + q_1\hat{i}$, or in Euler form, $e^{\hat{i}\theta} = \cos\theta + \hat{i}\sin\theta$ $(0 \leq \theta \leq 2\pi)$. Viewing the Euler form of complex numbers as an operator, this quantity rotates any vector Z by an angle θ radians and does not change its length. Consequently, when any vector $Z = e^{i\theta}$ is multiplied by the operator $W = ce^{i\psi}$, Z is rotated by an angle ψ and increased in length by a factor c (not equal to zero), i.e., $Z_1 = ce^{i(\theta + \psi)}$.

Hamilton derived a method called quaternion to manipulate the representation of vector. These quaternions constitute a four dimensional vector space over the field of real numbers, with a basis of four special vectors denoted by $\hat{e}$, $\hat{i}$, $\hat{j}$, $\hat{k}$. The elements of quaternion are ordered pairs of complex numbers, i.e., they can be represented as follows.

$$Q = (q_0 + q_1\hat{i}) + (q_2 + q_3\hat{i})\,\hat{j} = Z_1 + Z_2\hat{j}$$

where Z_1 and Z_2 behave like ordinary complex numbers.

I. QUATERNION ALGEBRA [3]

A quaternion Q is defined as

$$Q = q_0 + \hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3 = q_0 + \underline{v} \ , \tag{1}$$

where q_0, q_1, q_2 and q_3 are scalars and

$$\underline{v} = \hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3 \tag{2}$$

is a vector with $\hat{i}$, $\hat{j}$ and $\hat{k}$ being three orthogonal unit vectors.

"Addition" for quaternion is defined as

$$Q + Q' = (q_0 + q_0') + (\underline{v} + \underline{v}') \ ,$$

$$Q + Q' = (q_0 + q_0') + \hat{i}(q_1 + q_1') + \hat{j}(q_2 + q_2') + \hat{k}(q_3 + q_3') \ .$$

It is obvious that addition is associative and commutative.

"Equality" is defined as $Q = Q'$ if and only if $q_0 = q_0'$ and $\underline{v} = \underline{v}'$.

"Multiplication" is defined by using the distributive law on the elements as in ordinary algebra but with order preserved, and by defining the products of unit vectors as

$$\left.\begin{aligned} \hat{i}^2 = \hat{j}^2 = \hat{k}^2 &= -1 \\ \hat{i}\hat{j} = -\hat{j}\hat{i} &= \hat{k} \\ \hat{j}\hat{k} = -\hat{k}\hat{j} &= \hat{i} \\ \hat{k}\hat{i} = -\hat{i}\hat{k} &= \hat{j} \end{aligned}\right\} \ . \tag{3}$$

Now, we can write the "product"

$$Q'Q = (q_0' + \underline{v}')(q_0 + \underline{v}) ,$$

$$Q'Q = q_0'q_0 + q_0\underline{v}' + q_0'\underline{v} + \underline{v}'\underline{v} .$$

But

$$\underline{v}'\underline{v} = (\hat{i}q_1' + \hat{j}q_2' + \hat{k}q_3')(\hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3) ,$$

$$\underline{v}'\underline{v} = -(q_1'q_1 + q_2'q_2 + q_3'q_3) + \hat{i}(q_2'q_3 - q_3'q_2)$$

$$+ \hat{j}(q_3'q_1 - q_1'q_3) + \hat{k}(q_1'q_2 - q_2'q_1) ,$$

$$\underline{v}'v = -(\underline{v}' \cdot \underline{v}) + (\underline{v}' \times \underline{v}) ,$$

where "$\cdot$" and "$\times$" denote, respectively, the dot and cross products in the sense of vector analysis. Note that $\underline{v}' \cdot \underline{v}$ is a scalar and $\underline{v}' \times \underline{v}$ is a vector. Therefore,

$$Q'Q = (q_0'q_0 - \underline{v}' \cdot \underline{v}) + (q_0\underline{v}' + q_0'\underline{v} + \underline{v}' \times \underline{v}) . \qquad (4)$$

The product of quaternion is defined differently from both of the dot and cross products associated with pure vectors. The product is not commutative due to the term $\underline{v}' \times \underline{v}$ in Eq. (4). Quaternion multiplication is distributive and associative, i.e.,

$$Q(Q' + Q'') = QQ' + QQ'' ,$$

and

$$Q(Q'Q'') = (QQ')Q'' .$$

The "conjugate" of Q* of Q is defined as

$$Q^* = q_0 - \underline{v} \ . \tag{5}$$

The conjugate of the product is the product of conjugate in reverse order, i.e.,

$$(Q'Q)^* = [(q_0'q_0 - \underline{v}' \cdot \underline{v}) + (q_0\underline{v}' + q_0'\underline{v} + \underline{v}' \times \underline{v})]^* \ ,$$

$$(Q'Q)^* = (q_0'q_0 - \underline{v}' \cdot \underline{v}) - (q_0\underline{v}' + q_0'\underline{v} + \underline{v}' \times \underline{v}) \ ,$$

$$(Q'Q)^* = Q^*Q'^* \ . \tag{6}$$

The "norm" of a quaternion is defined as a scalar, i.e.,

$$N(Q) = q_0^2 + q_1^2 + q_2^2 + q_3^2 \ . \tag{7}$$

A unit norm is defined as N(Q) = 1. Also, by Eq. (4), one may obtain

$$QQ^* = Q^*Q = q_0^2 + \underline{v} \cdot \underline{v} + (q_0\underline{v} - q_0\underline{v} - \underline{v} \times \underline{v}) \ ,$$

$$QQ^* = Q^*Q = q_0^2 + q_1^2 + q_2^2 + q_3^2 \ .$$

Hence, one may represent the norm as

$$N(Q) = QQ^* = Q^*Q \ . \tag{8}$$

The norm of the product is the product of the norm, i.e.,

$$N(Q'Q) = (Q'Q)(Q'Q)^* \ ,$$

$$N(Q'Q) = Q'QQ^*Q'^* \ ,$$

$$N(Q'Q) = Q'N(Q)Q'^* \ ,$$

$$N(Q'Q) = N(Q)N(Q') \ . \tag{9}$$

The "inverse" of a quaternion Q is denoted by Q^{-1} and is defined through the relationship

$$Q^{-1}Q = QQ^{-1} = 1 \ . \tag{10}$$

since

$$N(Q) = QQ^* = Q^*Q \ ,$$

therefore, Q^{-1} may also define as

$$Q^{-1} = \frac{Q^*}{N(Q)} \ . \tag{11}$$

For a unit quaternion, N(Q) = 1, the inverse of a quaternion Q is equal to the conjugate, i.e.,

$$Q^{-1} = Q^* \ .$$

II. QUATERNIONS AS A ROTATIONAL OPERATOR [1][2][3]

The attitude of a rigid body in space is usually described by the three Euler angles that are defined by three successive rotations about three different axes. However, by a known theorem due to Euler, it is always possible to combine a number of finite rotations of a rigid body into a single rotation about some axis.

Referring to Figure 1, a position vector $\underline{r}'$ is related to another position vector $\underline{r}$ through a rotation of angle θ about a unit vector e that is stationary in space. The notation $\overline{AB}$ will be used to denote a

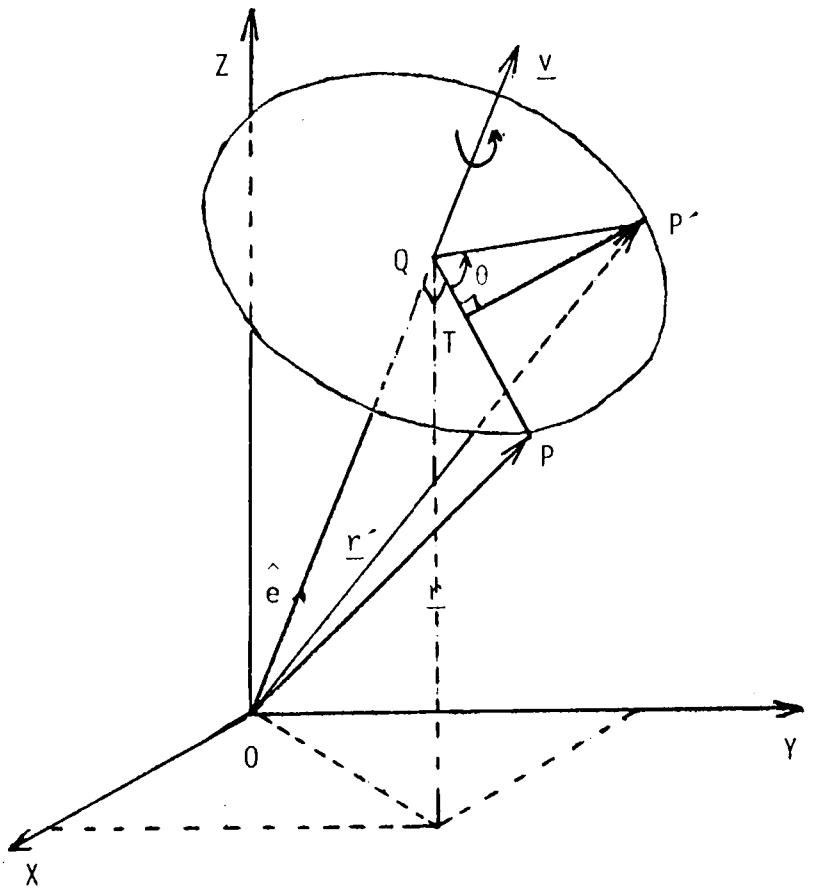


Figure 1. Rotation of a position vector about a unit vector ê.

vector which begins at point A and ends at point B through the section. One may see that the following geometrical relationships exist.

$$\overrightarrow{QT} = \overrightarrow{QP}\cos\theta = (\underline{r} - \overrightarrow{OQ})\cos\theta \ ,$$

$$\overrightarrow{TP'} = (\hat{e} \times \underline{r})\sin\theta \ ,$$

$$\underline{r'} = \overrightarrow{OQ} + \overrightarrow{QT} + \overrightarrow{TP'} \ ,$$

$$\underline{r'} = \underline{r}\cos\theta + \overrightarrow{OQ}(1 - \cos\theta) + (\hat{e} \times \underline{r})\sin\theta \ . \tag{12}$$

Defining the quaternion components in terms of $\hat{e}$ and θ, it follows that

$$q_0 = \cos\frac{\theta}{2} \tag{13}$$

$$\underline{v} = \hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3 = \hat{e}\sin\frac{\theta}{2} \ . \tag{14}$$

since

$$\cos\theta = \cos^2\frac{\theta}{2} - \sin^2\frac{\theta}{2} = q_0^2 - \underline{v}\cdot\underline{v} \ , \tag{15}$$

$$\sin\theta = 2\sin\frac{\theta}{2}\cos\frac{\theta}{2} = 2|v|q_0 \ , \tag{16}$$

$$1 - \cos\theta = 2\underline{v}\cdot\underline{v} \ , \tag{17}$$

one may insert Eqs. (15), (16) and (17) into (12) yields

$$\underline{r'} = \underline{r}(q_0^2 - \underline{v}\cdot\underline{v}) + 2\underline{v}(\underline{v}\cdot\underline{r}) + 2q_0(\underline{v}\times\underline{r}) \ . \tag{18}$$

Using the identity

$$\underline{v}(\underline{v}\cdot\underline{r}) - (\underline{v}\cdot\underline{v})\underline{r} = \underline{v}\times(\underline{v}\times\underline{r}) \ , \tag{19}$$

Eq. (18) becomes

$$r' = r + 2q_0(\underline{v} \times \underline{r}) + 2\underline{v} \times (\underline{v} \times \underline{r}) \ . \tag{20}$$

Equation (20) describes a rotational transformation between two vectors $\underline{r}$ and $\underline{r}'$ in terms of quaternion. Let us express this quaternion in matrix notation. Defining

$$\underline{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \ , \qquad \underline{r}' = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \ ,$$

$$P = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \ , \tag{21}$$

then, one may obtain the following relationships

$$\underline{v} \times \underline{r} = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = p\underline{r} \ ,$$

and

$$\underline{v} \times (\underline{v} \times \underline{r}) = p^2\underline{r} = \begin{bmatrix} -q_2^2 - q_3^2 & q_1q_2 & q_1q_3 \\ q_1q_2 & -q_1^2 - q_3^2 & q_2q_3 \\ q_1q_3 & q_2q_3 & -q_1^2 - q_2^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \ .$$

Let $\underline{a}$ and $\underline{b}$ represent two vectors and I be the unit matrix, then Eq. (20) can take the form

$$\underline{a} = [I + 2q_0p + 2p^2]\underline{b} \ .$$

Defining the transformation matrix as

$$[M] = I + 2q_0p + 2p^2 \ , \tag{22}$$

M matrix becomes

$$[M] = \begin{bmatrix} 1-2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1-2(q_3^2 + q_1^2) & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1-2(q_1^2 + q_2^2) \end{bmatrix} \ . \tag{23}$$

Using the relationship of unit norm

$$N(Q) = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \ ,$$

the M matrix may also be written as

$$[M] = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_2^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} . \tag{24}$$

Therefore, Eq. (20) becomes

$$\underline{r}' = (I + 2q_0p + p^2)\underline{r} \ .$$

Inserting Eq. (22) into this relationship, one may have

$$\underline{r}' = [M]\underline{r} \ , \tag{25}$$

which is the first form of the rotational transformation using quaternion.

A second form of rotational transformation using quaternion is given by

$$\underline{r}' = Q\underline{r}Q^* \ . \tag{26}$$

Let us show that (25) and (26) are equivalent.

$$Q\underline{r}Q^* = (q_0 + \underline{v})\underline{r}(q_0 - \underline{v}) \ ,$$

$$Q\underline{r}Q^* = (q_0 + \underline{v})[\underline{r} \cdot \underline{v} + (q_0\underline{r} - \underline{r} \times \underline{v})] \ ,$$

$$Q\underline{r}Q^* = q_0(\underline{r} \cdot \underline{v}) + q_0(q_0\underline{r} - \underline{r} \times \underline{v}) + \underline{v}(\underline{r} \cdot \underline{v}) - \underline{v} \cdot (q_0\underline{r} - \underline{r} \times \underline{v}) + \underline{v} \times (q_0\underline{r} - \underline{r} \times \underline{v}) \ ,$$

$$Q\underline{r}Q^* = q_0^2\underline{r} + 2q_0(\underline{v} \times \underline{r}) + \underline{v}(\underline{r} \cdot \underline{v}) - \underline{v} \times (\underline{r} \times \underline{v}) \ .$$

Adding and subtracting $(\underline{v} \cdot \underline{v})\underline{r}$, one may obtain

$$Q\underline{r}Q^* = (q_0^2 + \underline{v} \cdot \underline{v})\underline{r} + 2q_0(\underline{v} \times \underline{r}) + \underline{v} \times (\underline{v} \times \underline{r}) + [\underline{v}(\underline{r} \cdot \underline{v}) - (\underline{v} \cdot \underline{v})\underline{r}] \ .$$

Using identity (19) and

$$q_0^2 + \underline{v} \cdot \underline{v} = \cos^2 \frac{\theta}{2} + \sin^2 \frac{\theta}{2} = 1 \ ,$$

Eq. (26) can be written as

$$\underline{r}' = \underline{r} + 2q_0(\underline{v} \times \underline{r}) + 2\underline{v} \times (\underline{v} \times \underline{r}) \tag{27}$$

This equation is identical to Eq. (20). Therefore, Eqs. (26) and (25) are equivalent.

III. QUATERNION DIFFERENCE EQUATION

Let Q_1 and Q_2 be quaternions associated with two successive rotations. Then, using Eq. (26),

$$\underline{r}_2 = Q_1 \underline{r}_1 Q_1^* ,$$

$$\underline{r}_3 = Q_2 \underline{r}_2 Q_2^* ,$$

$$\underline{r}_3 = (Q_2 Q_1) \underline{r}_1 (Q_2 Q_1)^* ,$$

the quaternion associated with the resultant rotation is therefore given by

$$Q_3 = Q_2 Q_1 \tag{28}$$

Quaternions can be used to express an attitude algorithm used in strapdown inertial measurement unit (IMU). In using a strapdown platform, the attitude of a space vehicle is measured along a coordinate frame fixed to the vehicle body. However, it is commonly desired to determine the attitude of the space vehicle with respect to a different coordinate frame, such as an inertial coordinate frame or an earth fixed coordinate frame. Therefore, a transformation of attitude information from the vehicle body frame to a desired coordinate frame is needed [2].

Let quaternion Q_n represent the vehicle attitude at time n with respect to a chosen reference and let Q_n' be the updating quaternion. By Eq. (28), the updated quaternion is given by

$$Q_{n+1} = Q_n' Q_n . \tag{29}$$

Denoting Q_n' and Q_n as follows

$$Q_n' = q_0' + \hat{i}q_1' + \hat{j}q_2' + \hat{k}q_3' ,$$

$$Q_n = q_0 + \hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3 ,$$

then Eq. (29) becomes

$$\begin{aligned} Q_{n+1} = {} & (q_0q_0' - q_1q_1' - q_2q_2' - q_3q_3') \\ & + \hat{i}(q_0q_1' + q_1q_0' - q_2q_3' + q_3q_2') \\ & + \hat{j}(q_0q_2' + q_1q_3' + q_2q_0' - q_3q_1') \\ & + \hat{k}(q_0q_3' - q_1q_2' + q_2q_1' + q_3q_0') . \end{aligned}$$

In matrix notation, the quaternion updating equation is

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & -q_3' & q_2' \\ q_2' & q_3' & q_0' & -q_1' \\ q_3' & -q_2' & q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_n . \qquad (30)$$

It is interesting to note from Eq. (14) that quaternion components are related to three direction cosines n_1, n_2 and n_3 of $\hat{e}$ and the magnitude θ of the rotation angle

$$\left.\begin{aligned} q_0' &= \cos\frac{\theta}{2} \\ q_1' &= n_1 \sin\frac{\theta}{2} \\ q_2' &= n_2 \sin\frac{\theta}{2} \\ q_3' &= n_3 \sin\frac{\theta}{2} \end{aligned}\right\} . \qquad (31)$$

Following these relationships, we shall derive a quaternion attitude algorithm in Chapter III.

IV. QUATERNION DIFFERENTIAL EQUATIONS

Quaternion differential equations will be derived in this section. One can see that the rates of change of direction cosines are given by the well known formula (shown in Appendix I)

$$\dot{[M]} = [M] \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} , \tag{32}$$

where [M] is the direction cosine matrix. The direction cosine matrix in terms of quaternion parameters is shown in Eq. (23), i.e.,

$$[M] = \begin{bmatrix} 1-2(q_2^2 + q_3^2) & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & 1-2(q_1^2 + q_3^2) & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_0q_2 & 2q_2q_3 + 2q_0q_1 & 1-2(q_1^2 + q_2^2) \end{bmatrix} . \tag{23}$$

Equation (32) shows the following relationships

$$\left.\begin{aligned} \dot{M}_{11} &= M_{12}\omega_z - M_{13}\omega_y \\ \dot{M}_{22} &= -M_{21}\omega_z + M_{23}\omega_x \\ \dot{M}_{33} &= M_{31}\omega_y - M_{32}\omega_x \end{aligned}\right\} . \tag{33}$$

Substituting Eq. (23) into Eq. (33) yields

$$-4q_2\dot{q}_2 - 4q_3\dot{q}_3 = 2(q_1q_2 - q_0q_3)\omega_z - 2(q_1q_3 + q_0q_2)\omega_y \ , \tag{34}$$

$$-4q_1\dot{q}_1 - 4q_3\dot{q}_3 = 2(q_1q_2 - q_0q_3)\omega_z + 2(q_2q_3 - q_0q_2)\omega_x \ , \tag{35}$$

$$-4q_1\dot{q}_1 - 4q_2\dot{q}_2 = 2(q_1q_3 - q_0q_2)\omega_y - 2(q_2q_3 + q_0q_1)\omega_x \ . \tag{36}$$

Solving Eqs. (34) to (36) and using relationship

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \ , \tag{37}$$

the following equations are obtained

$$\dot{q}_0 = \frac{dq_0}{dt} = -\frac{\omega_x}{2} q_1 - \frac{\omega_y}{2} q_2 - \frac{\omega_z}{2} q_3 \ , \tag{38}$$

$$\dot{q}_1 = \frac{dq_1}{dt} = \frac{\omega_x}{2} q_0 + \frac{\omega_z}{2} q_2 - \frac{\omega_y}{2} q_3 \ , \tag{39}$$

$$\dot{q}_2 = \frac{dq_2}{dt} = \frac{\omega_y}{2} q_0 - \frac{\omega_z}{2} q_1 + \frac{\omega_x}{2} q_3 \ , \tag{40}$$

$$\dot{q}_3 = \frac{dq_3}{dt} = \frac{\omega_z}{2} q_0 - \frac{\omega_y}{2} q_1 - \frac{\omega_x}{2} q_2 \ . \tag{41}$$

Combining Eqs. (38) to (41) in matrix form yields the quaternion differential equations

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} 0 & \frac{-\omega_x}{2} & \frac{-\omega_y}{2} & \frac{-\omega_z}{2} \\ \frac{\omega_x}{2} & 0 & \frac{\omega_z}{2} & \frac{-\omega_y}{2} \\ \frac{\omega_y}{2} & \frac{-\omega_z}{2} & 0 & \frac{\omega_x}{2} \\ \frac{\omega_z}{2} & \frac{\omega_y}{2} & \frac{-\omega_x}{2} & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \ . \tag{42}$$

We shall also derive a quaternion attitude algorithm by solving these differential equations.

CHAPTER III

QUATERNION ATTITUDE ALGORITHM

Quaternion can be used to express attitude in a space vehicle control system; for example, it can be used in strapdown inertial measurement unit (IMU) [4], [5]. In a strapdown system, gyros measure the instantaneous angular rates of IMU about its own axes. Assume that ω_x, ω_y, ω_z are the angular rate with respect to inertial space along the vehicle's body coordinates and $\Delta\theta_x$, $\Delta\theta_y$ and $\Delta\theta_z$ are the components of vehicle's incremental attitude change which are sensed by strapdown gyros along vehicle's body coordinates. On the assumption, two update algorithms using quaternion were derived in this chapter. One of these was developed from the algebra of quaternion and its difference equations and the other from quaternion differential equations.

I. APPROACH I[3]

The quaternion difference equation shown in Eq. (30) is repeated below

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & -q_3' & q_2' \\ q_2' & q_3' & q_0' & -q_1' \\ q_3' & -q_2' & q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_n . \tag{30}$$

It, together with Eq. (25), constitutes an attitude algorithm. Elements of the updating quaternion in Eq. (30) are given by

$$\left.\begin{aligned} q_0' &= \cos\frac{\Delta\theta}{2} \\ q_1' &= n_1 \sin\frac{\Delta\theta}{2} \\ q_2' &= n_2 \sin\frac{\Delta\theta}{2} \\ q_3' &= n_3 \sin\frac{\Delta\theta}{2} \end{aligned}\right\} . \tag{43}$$

where

$$n_1 = \frac{\Delta\theta_x}{\Delta\theta} , \quad n_2 = \frac{\Delta\theta_y}{\Delta\theta} , \quad n_3 = \frac{\Delta\theta_z}{\Delta\theta} ,$$

and

$$\Delta\theta = \sqrt{\Delta\theta_x^{\,2} + \Delta\theta_y^{\,2} + \Delta\theta_z^{\,2}} .$$

If one denotes the quaternions associated with angular increments $\Delta\theta_1$ and $\Delta\theta_2$ by Q_1' and Q_2', then

$$Q_n' = Q_2'Q_1' = q_{0_n}' + \hat{i}q_{1_n}' + \hat{j}q_{2_n}' + \hat{k}q_{3_n}' . \tag{44}$$

Let the elements of Q_1' and Q_2' be $(q_{01}, q_{11}, q_{21}, q_{31})$ and $(q_{02}, q_{12}, q_{22}, q_{32})$, respectively. Then Eq. (44) becomes

$$Q_n' = (q_{02} + \hat{i}q_{12} + \hat{j}q_{22} + \hat{k}q_{32})(q_{01} + \hat{i}q_{11} + \hat{j}q_{21} + \hat{k}q_{31}) ,$$

$$Q_n' = (q_{01}q_{02} - q_{11}q_{12} - q_{21}q_{22} - q_{31}q_{32})$$
$$+ \hat{i}(q_{01}q_{12} + q_{11}q_{02} + q_{22}q_{31} - q_{32}q_{21})$$
$$+ \hat{j}(q_{01}q_{22} + q_{21}q_{02} + q_{11}q_{32} - q_{12}q_{31})$$
$$+ \hat{k}(q_{01}q_{32} + q_{31}q_{02} + q_{12}q_{21} - q_{22}q_{11}) ,$$

therefore,

$$q_{0_n}' = q_{01}q_{02} - q_{11}q_{12} - q_{21}q_{22} - q_{31}q_{32} , \tag{45}$$
$$q_{1_n}' = q_{01}q_{12} + q_{11}q_{02} + q_{22}q_{31} - q_{32}q_{21} , \tag{46}$$
$$q_{2_n}' = q_{01}q_{22} + q_{21}q_{02} + q_{11}q_{32} - q_{31}q_{12} , \tag{47}$$
$$q_{3_n}' = q_{01}q_{32} + q_{31}q_{02} + q_{12}q_{21} - q_{22}q_{11} . \tag{48}$$

Using the third order Taylor series approximations

$$\cos \frac{\Delta\theta}{2} \approx 1 - \frac{\Delta\theta^2}{8} ,$$

and

$$\sin \frac{\Delta\theta}{2} \approx \frac{\Delta\theta}{2} - \frac{\Delta\theta^3}{48} ,$$

the two sets of four quaternion elements become

$$\left.\begin{aligned} q_{01} &\approx 1 - \frac{\Delta\theta_1{}^2}{8} \\ q_{11} &\approx \Delta\theta_{x1}\left(\frac{1}{2} - \frac{\Delta\theta_1{}^2}{48}\right) \\ q_{21} &\approx \Delta\theta_{y1}\left(\frac{1}{2} - \frac{\Delta\theta_1{}^2}{48}\right) \\ q_{31} &\approx \Delta\theta_{z1}\left(\frac{1}{2} - \frac{\Delta\theta_1{}^2}{48}\right) \end{aligned}\right\} , \tag{49}$$

$$\left.\begin{aligned} q_{02} &\approx 1 - \frac{\Delta\theta_2{}^2}{8} \\ q_{12} &\approx \Delta\theta_{x2}\left(\frac{1}{2} - \frac{\Delta\theta_2{}^2}{48}\right) \\ q_{22} &\approx \Delta\theta_{y2}\left(\frac{1}{2} - \frac{\Delta\theta_2{}^2}{48}\right) \\ q_{32} &\approx \Delta\theta_{z2}\left(\frac{1}{2} - \frac{\Delta\theta_2{}^2}{48}\right) \end{aligned}\right\} . \tag{50}$$

Substituting Eqs. (49) and (50) into Eq. (45) yields

$$\begin{aligned} q_0{}' = 1 &- \frac{\Delta\theta_1{}^2}{8} - \frac{\Delta\theta_2{}^2}{8} + \frac{\Delta\theta_1{}^2\Delta\theta_2{}^2}{64} \\ &- \left[\Delta\theta_{x1}\Delta\theta_{x2}\left(\frac{1}{4} - \frac{\Delta\theta_1{}^2}{96} - \frac{\Delta\theta_2{}^2}{96} + \frac{\Delta\theta_1{}^2\Delta\theta_2{}^2}{2304}\right)\right] \\ &- \left[\Delta\theta_{y1}\Delta\theta_{y2}\left(\frac{1}{4} - \frac{\Delta\theta_1{}^2}{96} - \frac{\Delta\theta_2{}^2}{96} + \frac{\Delta\theta_1{}^2\Delta\theta_2{}^2}{2304}\right)\right] \\ &- \left[\Delta\theta_{z1}\Delta\theta_{z2}\left(\frac{1}{4} - \frac{\Delta\theta_1{}^2}{96} - \frac{\Delta\theta_2{}^2}{96} + \frac{\Delta\theta_1{}^2\Delta\theta_2{}^2}{2304}\right)\right] . \end{aligned} \tag{51}$$

Omitting terms higher than second order, Eq. (51) becomes

$$q_0{}' = 1 - \frac{1}{8}\left[\Delta\theta_1{}^2 + \Delta\theta_2{}^2 + 2\Delta\theta_{x1}\Delta\theta_{x2} + 2\Delta\theta_{y1}\theta_{y2} + 2\Delta\theta_{z1}\Delta\theta_{z2}\right] . \tag{52}$$

Letting

$$\left.\begin{aligned} \Delta\theta_1{}^2 &= \Delta\theta_{x1}{}^2 + \Delta\theta_{y1}{}^2 + \Delta\theta_{z1}{}^2 \\ \Delta\theta_2{}^2 &= \Delta\theta_{x2}{}^2 + \Delta\theta_{y2}{}^2 + \Delta\theta_{z2}{}^2 \end{aligned}\right\} , \tag{53}$$

Eq. (52) becomes

$$q_0' = 1 - \frac{1}{8}\left[(\Delta\theta_{x1} + \Delta\theta_{x2})^2 + (\Delta\theta_{y1} + \Delta\theta_{y2})^2 + (\Delta\theta_{z1} + \Delta\theta_{z2})^2\right] . \tag{54}$$

Similarly, substituting Eqs. (49) and (50) into Eqs. (46), (47) and (48), and omitting terms higher than third order, one obtains

$$q_1' = \frac{1}{2}(\Delta\theta_{x1} + \Delta\theta_{x2}) + \frac{1}{4}(\Delta\theta_{y2}\Delta\theta_{z1} - \Delta\theta_{y1}\Delta\theta_{z2}) - \frac{1}{48}\left[\Delta\theta_{x2}(3\Delta\theta_1^2 + \Delta\theta_2^2) + \Delta\theta_{x1}(\Delta\theta_1^2 + 3\Delta\theta_2^2)\right] , \tag{55}$$

$$q_2' = \frac{1}{2}(\Delta\theta_{y1} + \Delta\theta_{y2}) + \frac{1}{4}(\Delta\theta_{x1}\Delta\theta_{z2} - \Delta\theta_{z1}\Delta\theta_{x2}) - \frac{1}{48}\left[\Delta\theta_{y2}(3\Delta\theta_1^2 + \Delta\theta_2^2) + \Delta\theta_{y1}(3\Delta\theta_2^2 + \Delta\theta_1^2)\right] , \tag{56}$$

$$q_3' = \frac{1}{2}(\Delta\theta_{z1} + \Delta\theta_{z2}) + \frac{1}{4}(\Delta\theta_{y1}\Delta\theta_{x2} - \Delta\theta_{x1}\Delta\theta_{y2}) - \frac{1}{48}\left[\Delta\theta_{z2}(3\Delta\theta_1^2 + \Delta\theta_2^2) + \Delta\theta_{z1}(3\Delta\theta_2^2 + \Delta\theta_1^2)\right] . \tag{57}$$

As a result, the matrix difference equation of Approach I is

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & -q_3' & q_2' \\ q_2' & q_3' & q_0' & -q_1' \\ q_3' & -q_0' & q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_n , \tag{31}$$

where

$$q_0' = 1 - \frac{1}{8}\left[(\Delta\theta_{xn} + \Delta\theta_{x2n})^2 + (\Delta\theta_{y1n} + \Delta\theta_{y2n})^2 + (\Delta\theta_{z1n} + \Delta\theta_{z2n})^2\right] , \tag{54}$$

$$q_1' = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) + \frac{1}{4}(\Delta\theta_{y2n}\Delta\theta_{z1n} - \Delta\theta_{z2n}\Delta\theta_{y1n}) - \frac{1}{48}\left[\Delta\theta_{x2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \Delta\theta_{x1n}(3\Delta\theta_{2n}^2 + \Delta\theta_{1n}^2)\right] , \tag{55}$$

$$q_2' = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) + \frac{1}{4}(\Delta\theta_{x1n}\Delta\theta_{z2n} - \Delta\theta_{x2n}\Delta\theta_{z1n}) - \frac{1}{48}\left[\Delta\theta_{y2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \Delta\theta_{y1n}(3\Delta\theta_{2n}^2 + \Delta\theta_{1n}^2)\right] , \tag{56}$$

$$q_3' = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) + \frac{1}{4}(\Delta\theta_{x2n}\Delta\theta_{y1n} - \Delta\theta_{y2n}\Delta\theta_{x1n}) - \frac{1}{48}\left[\Delta\theta_{z2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \Delta\theta_{z1n}(3\Delta\theta_{2n}^2 + \Delta\theta_{1n}^2)\right] , \tag{57}$$

for

$$\Delta\theta_{1n}^2 = \Delta\theta_{x1n}^2 + \Delta\theta_{y1n}^2 + \Delta\theta_{z1n}^2 ,$$

and

$$\Delta\theta_{2n}^2 = \Delta\theta_{x2n}^2 + \Delta\theta_{y2n}^2 + \Delta\theta_{z2n}^2 .$$

II. APPROACH II

The quaternion differential equations derived in section II.4 also constitute an attitude algorithm. Derive the algorithm using the Taylor series expansion,

$$q_{n+1} = q_n + \left(\frac{dq}{dt}\right)_n \Delta T + \left(\frac{d^2q}{dt^2}\right)\frac{\Delta T^2}{2!} + \left(\frac{d^3q}{dt^3}\right)\frac{\Delta T^3}{3!}$$
$$+ \left(\frac{d^4q}{dt^4}\right)\frac{\Delta T^4}{4!} + \left(\frac{d^5q}{dt^5}\right)\frac{\Delta T^5}{5!} + \left(\frac{d^6q}{dt^6}\right)\frac{\Delta T^6}{6!} + \ldots , \tag{58}$$

and the quaternion differential equations

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} 0 & -\frac{\omega_x}{2} & -\frac{\omega_y}{2} & -\frac{\omega_z}{2} \\ \frac{\omega_x}{2} & 0 & \frac{\omega_z}{2} & -\frac{\omega_y}{2} \\ \frac{\omega_y}{2} & -\frac{\omega_z}{2} & 0 & \frac{\omega_x}{2} \\ \frac{\omega_z}{2} & \frac{\omega_y}{2} & -\frac{\omega_x}{2} & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} . \tag{42}$$

Four kinds of Taylor series approximations were derived as follows.

2.1. First Order Taylor Series Algorithm

In this section, the quaternion differential equation shown in Eq. (42) is solved by first order Taylor series expansion. The first order Taylor series expansion is given by taking the first two terms of Eq. (58)

$$q_{0n+1} = q_{0n} + \left(\frac{dq_0}{dt}\right)_n \Delta T , \tag{59}$$

$$q_{1n+1} = q_{1n} + \left(\frac{dq_1}{dt}\right)_n \Delta T , \tag{60}$$

$$q_{2n+1} = q_{2n} + \left(\frac{dq_2}{dt}\right)_n \Delta T , \tag{61}$$

$$q_{3n+1} = q_{3n} + \left(\frac{dq_3}{dt}\right)_n \Delta T . \tag{62}$$

Substituting Eqs. (38) to (41) into Eqs. (59) to (62) respectively, one obtains

$$q_{0n+1} = q_{0n} - \frac{\omega_{xn}\Delta T}{2} q_{1n} - \frac{\omega_{yn}\Delta T}{2} q_{2n} - \frac{\omega_{zn}\Delta T}{2} q_{3n} , \tag{63}$$

$$q_{1n+1} = \frac{\omega_{xn}\Delta T}{2} q_{0n} + q_{1n} + \frac{\omega_{zn}\Delta T}{2} q_{2n} - \frac{\omega_{yn}\Delta T}{2} q_{3n} , \tag{64}$$

$$q_{2n+1} = \frac{\omega_{yn}\Delta T}{2} q_{0n} - \frac{\omega_{zn}\Delta T}{2} q_{1n} + q_{2n} + \frac{\omega_{xn}\Delta T}{2} q_{3n} , \tag{65}$$

$$q_{3n+1} = \frac{\omega_{zn}\Delta T}{2} q_{0n} + \frac{\omega_{yn}\Delta T}{2} q_{1n} - \frac{\omega_{xn}\Delta T}{2} q_{2n} + q_{3n} . \tag{66}$$

Combining Eqs. (63) to (66) in matrix form yields

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \Delta T \begin{bmatrix} \frac{1}{\Delta T} & -\frac{\omega_{xn}}{2} & -\frac{\omega_{yn}}{2} & -\frac{\omega_{zn}}{2} \\ \frac{\omega_{xn}}{2} & \frac{1}{\Delta T} & \frac{\omega_{zn}}{2} & -\frac{\omega_{yn}}{2} \\ \frac{\omega_{yn}}{2} & -\frac{\omega_{zn}}{2} & \frac{1}{\Delta T} & \frac{\omega_{xn}}{2} \\ \frac{\omega_{zn}}{2} & \frac{\omega_{yn}}{2} & -\frac{\omega_{xn}}{2} & \frac{1}{\Delta T} \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n'} . \tag{67}$$

Let

$$q_0' = 1 , \tag{68}$$

$$q_1' = \frac{\omega_{xn}}{2} \Delta T , \tag{69}$$

$$q_2' = \frac{\omega_{yn}}{2} \Delta T , \tag{70}$$

$$q_3' = \frac{\omega_{zn}}{2} \Delta T \ , \tag{71}$$

then Eq. (67) becomes

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & q_3' & -q_2' \\ q_2' & -q_3' & q_0' & q_1' \\ q_3' & q_2' & -q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_n . \tag{72}$$

The relationship between angular velocity and angular increment is shown in Appendix II, the results are shown below

$$\left.\begin{aligned} \omega_{xn} &= \frac{3\Delta\theta_{x1n} - \Delta\theta_{x2n}}{\Delta T} \\ \omega_{yn} &= \frac{3\Delta\theta_{y1n} - \Delta\theta_{y2n}}{\Delta T} \\ \omega_{zn} &= \frac{3\Delta\theta_{z1n} - \Delta\theta_{z2n}}{\Delta T} \end{aligned}\right\} . \tag{73}$$

Substituting Eq. (73) into Eqs. (68) to (71), the quaternion updating elements in Eq. (72) become

$$q_0' = 1 \ , \tag{74}$$

$$q_1' = \frac{1}{2} (3\Delta\theta_{x1n} - \Delta\theta_{x2n}) \ , \tag{75}$$

$$q_2' = \frac{1}{2} (3\Delta\theta_{y1n} - \Delta\theta_{y2n}) \ , \tag{76}$$

$$q_3' = \frac{1}{2} (3\Delta\theta_{z1n} - \Delta\theta_{z2n}) \ . \tag{77}$$

2.2. Second Order Taylor Series Algorithm

In order to solve the quaternion differential equations second order Taylor series one may take the first three terms of Eq. (58), i.e.,

$$q_{0n+1} = q_{0n} + \left(\frac{dq_0}{dt}\right)_n \Delta T + \left(\frac{d^2q}{dt^2}\right)_n \frac{\Delta T^2}{2!} , \tag{78}$$

$$q_{1n+1} = q_{1n} + \left(\frac{dq_1}{dt}\right)_n \Delta T + \left(\frac{d^2q}{dt^2}\right)_n \frac{\Delta T^2}{2!} , \tag{79}$$

$$q_{2n+1} = q_{2n} + \left(\frac{dq_2}{dt}\right)_n \Delta T + \left(\frac{d^2q}{dt^2}\right)_n \frac{\Delta T^2}{2!} , \tag{80}$$

$$q_{3n+1} = q_{3n} + \left(\frac{dq_3}{dt}\right)_n \Delta T + \left(\frac{d^2q}{dt^2}\right)_n \frac{\Delta T^2}{2!} . \tag{81}$$

The derivatives of Eqs. (38) to (41) are

$$\left(\frac{d^2q_0}{dt^2}\right)_n = \frac{-q_{0n}}{4}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - \frac{q_{1n}}{2}\dot{\omega}_{xn} - \frac{q_{2n}}{2}\dot{\omega}_{yn} - \frac{q_{3n}}{2}\dot{\omega}_{zn} , \tag{82}$$

$$\left(\frac{d^2q_1}{dt^2}\right)_n = \frac{q_{0n}}{2}\dot{\omega}_{xn} - \frac{q_{1n}}{4}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) + \frac{q_{2n}}{2}\dot{\omega}_{zn} - \frac{q_{3n}}{2}\dot{\omega}_{yn} , \tag{83}$$

$$\left(\frac{d^2q_2}{dt^2}\right)_n = \frac{q_{0n}}{2}\dot{\omega}_{yn} - \frac{q_{1n}}{2}\dot{\omega}_{zn} - \frac{q_{2n}}{4}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) + \frac{q_{3n}}{2}\dot{\omega}_{yn} , \tag{84}$$

$$\left(\frac{d^2q_3}{dt^2}\right)_n = \frac{q_{0n}}{2}\dot{\omega}_{zn} + \frac{q_{1n}}{2}\dot{\omega}_{yn} - \frac{q_{2n}}{2}\dot{\omega}_{xn} - \frac{q_{3n}}{4}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) . \tag{85}$$

Substituting Eqs. (38) to (41) and Eqs. (83) to (86) into Eqs. (79) to (82) respectively, one obtains

$$q_{0n+1} = q_{0n}\left[1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\right] - q_{1n}\left[\frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4}\right] - q_{2n}\left[\frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4}\right] - q_{3n}\left[\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4}\right], \tag{86}$$

$$q_{1n+1} = q_{0n}\left[\frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4}\right] + q_{1n}\left[1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\Delta T^2\right] + q_{2n}\left[\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4}\right] - q_{3n}\left[\frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4}\right], \tag{87}$$

$$q_{2n+1} = q_{0n}\left[\frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4}\right] - q_{1n}\left[\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4}\right] + q_{2n}\left[1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\,\Delta T^2\right] + q_{3n}\left[\frac{\omega_{xn}\Delta T}{2} + \frac{\omega_{xn}\Delta T^2}{4}\right], \tag{88}$$

$$q_{3n+1} = q_{0n}\left[\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4}\right] + q_{1n}\left[\frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4}\right] - q_{2n}\left[\frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4}\right] + q_{3n}\left[1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\Delta T^2\right] . \tag{89}$$

Letting

$$q_0' = 1 - \frac{\Delta T^2}{8}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) , \tag{90}$$

$$q_1' = \frac{\omega_{xn}}{2}\Delta T + \frac{\dot{\omega}_{xn}}{4}\Delta T^2 , \tag{91}$$

$$q_2' = \frac{\omega_{yn}}{2}\Delta T + \frac{\dot{\omega}_{yn}}{4}\Delta T^2 , \tag{92}$$

$$q_3' = \frac{\omega_{zn}}{2}\Delta T + \frac{\dot{\omega}_{zn}}{4}\Delta T^2 , \tag{93}$$

Eqs. (87) to (90) become

$$q_{0n+1} = q_{0n}q_0' - q_{1n}q_1' - q_{2n}q_2' - q_{3n}q_3' , \tag{94}$$

$$q_{1n+1} = q_{0n}q_1' + q_{1n}q_0' + q_{2n}q_3' - q_{3n}q_2' , \tag{95}$$

$$q_{2n+1} = q_{0n}q_2' - q_{1n}q_3' + q_{2n}q_0' + q_{3n}q_1' , \tag{96}$$

$$q_{3n+1} = q_{0n}q_3' + q_{1n}q_2' - q_{2n}q_1' + q_{3n}q_0' . \tag{97}$$

Combining Eqs. (95) to (98) in matrix form

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & q_3' & -q_2' \\ q_2' & -q_3' & q_0' & q_1' \\ q_3' & q_2' & -q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_n , \tag{98}$$

one may see that Eq. (98) is the same as Eq. (72).

The relationships between angular acceleration and angular increment are also shown in Appendix II, the results are

$$\left.\begin{aligned} \dot{\omega}_{xn} &= \frac{4(\Delta\theta_{x2n} - \Delta\theta_{x1n})}{\Delta T^2} \\ \dot{\omega}_{yn} &= \frac{4(\Delta\theta_{y2n} - \Delta\theta_{y1n})}{\Delta T^2} \\ \dot{\omega}_{zn} &= \frac{4(\Delta\theta_{z2n} - \Delta\theta_{z1n})}{\Delta T^2} \end{aligned}\right\} . \tag{99}$$

Substituting Eqs. (73) and (99) into Eqs. (91) to (94), one obtains the second order Taylor series algorithm, i.e.,

$$q_0' = 1 - \frac{1}{8}\left[(3\Delta\theta_{x1n} - \Delta\theta_{x2n})^2 + (3\Delta\theta_{y1n} - \Delta\theta_{y2n})^2 + (3\Delta\theta_{z1n} - \Delta\theta_{z2n})^2\right] , \tag{100}$$

$$q_1' = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) , \tag{101}$$

$$q_2' = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) , \tag{102}$$

$$q_3' = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) . \tag{103}$$

2.3. Third Order Taylor Series Algorithm

To derive third order Taylor series algorithm, one may take the first four terms of Eq. (58)

$$q_{0n+1} = q_{0n} + \left(\frac{dq_0}{dt}\right)_n \Delta T + \left(\frac{d^2q_0}{dt^2}\right)_n \frac{\Delta T^2}{2!} + \left(\frac{d^3q_0}{dt^3}\right)_n \frac{\Delta T^3}{3!} \tag{104}$$

The derivative of Eq. (82) is as follows

$$\left(\frac{d^3q_0}{dt^3}\right)_n = q_{0n}\left[\frac{-3}{4}\left(\omega_{xn}\dot{\omega}_{xn} + \omega_{yn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{zn}\right)\right]$$
$$+ q_{1n}\left[\frac{1}{8}\omega_{xn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - \frac{1}{4}\left(\omega_{yn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{yn}\right)\right]$$
$$+ q_{2n}\left[\frac{1}{8}\omega_{yn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) + \frac{1}{4}\left(\omega_{xn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{xn}\right)\right]$$
$$+ q_{3n}\left[\frac{1}{8}\omega_{zn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - \frac{1}{4}\left(\omega_{xn}\dot{\omega}_{yn} - \omega_{yn}\dot{\omega}_{xn}\right)\right] . \tag{105}$$

Substituting Eqs. (38), (82) and (105) into Eq. (104) yields

$$q_{0n+1} = q_{0n}\left\{1 - \frac{\Delta T^2}{8}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - \frac{\Delta T^3}{8}\left(\omega_{xn}\dot{\omega}_{xn} + \omega_{yn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{zn}\right)\right\} - q_{1n}\left\{\frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{xn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - 2\left(\omega_{yn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{yn}\right)\right]\right\} - q_{2n}\left\{\frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{yn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) + 2\left(\omega_{xn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{xn}\right)\right]\right\} - q_{3n}\left\{\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{zn}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - 2\left(\omega_{xn}\dot{\omega}_{yn} - \omega_{yn}\dot{\omega}_{xn}\right)\right]\right\} . \tag{106}$$

Letting

$$q_0' = 1 - \frac{\Delta T^2}{8}\left(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2\right) - \frac{\Delta T^3}{8}\left(\omega_{xn}\dot{\omega}_{xn} + \omega_{yn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{zn}\right) , \tag{107}$$

$$q_1' = \frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{xn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - 2(\omega_{yn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{yn})\right] , \tag{108}$$

$$q_2' = \frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{yn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) + 2(\omega_{xn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{xn})\right] , \tag{109}$$

$$q_3' = \frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{zn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - 2(\omega_{xn}\dot{\omega}_{yn} - \omega_{yn}\dot{\omega}_{xn})\right] , \tag{110}$$

Eq. (106) becomes

$$q_{0n+1} = q_{0n}q_0' - q_{1n}q_1' - q_{2n}q_2' - q_{3n}q_3' . \tag{111}$$

Similarly, as shown before, one may obtain the following relationships

$$q_{1n+1} = q_{0n}q_1' + q_{1n}q_0' + q_{2n}q_3' - q_{3n}q_2' , \tag{112}$$

$$q_{2n+1} = q_{01}q_2' - q_{1n}q_3' + q_{2n}q_0' + q_{3n}q_1' , \tag{113}$$

and

$$q_{3n+1} = q_0q_3' + q_{1n}q_2' - q_{2n}q_1' + q_{3n}q_0' . \tag{114}$$

Substituting Eqs. (73) and (99) into Eqs. (107) to (110) yields

$$q_0' = 1 - \frac{1}{8}\left[-3(\Delta\theta_{x1n}^2 + \Delta\theta_{x2n}^2 + \Delta\theta_{y1n}^2 + \Delta\theta_{y2n}^2 + \Delta\theta_{z1n}^2 + \Delta\theta_{z2n}^2) + 10(\Delta\theta_{x1n}\Delta\theta_{x2n} + \Delta\theta_{y1n}\Delta\theta_{y2n} + \Delta\theta_{z1n}\Delta\theta_{z2n})\right] , \tag{115}$$

$$q_1' = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) + \frac{1}{3}(\Delta\theta_{y1n}\Delta\theta_{z2n} - \Delta\theta_{y2n}\Delta\theta_{z1n})$$
$$- \frac{1}{48}(3\Delta\theta_{x1n} - \Delta\theta_{x2n})[(3\Delta\theta_{x1n} - \Delta\theta_{x2n})^2 + (3\Delta\theta_{y1n} - \Delta\theta_{y2n})^2$$
$$+ (3\Delta\theta_{z1n} - \Delta\theta_{z2n})^2], \tag{116}$$

$$q_2' = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) + \frac{1}{3}(\Delta\theta_{x2n}\Delta\theta_{z1n} - \Delta\theta_{x1n}\Delta\theta_{z2n})$$
$$- \frac{1}{48}(3\Delta\theta_{y1n} - \Delta\theta_{y2n})[(3\Delta\theta_{x1n} - \Delta\theta_{x2n})^2 + (3\Delta\theta_{y1n} - \Delta\theta_{y2n})^2$$
$$+ (3\Delta\theta_{z1n} - \Delta\theta_{z2n})^2], \tag{117}$$

and

$$q_3' = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) + \frac{1}{3}(\Delta\theta_{x1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n})$$
$$- \frac{1}{48}(3\Delta\theta_{z1n} - \Delta\theta_{z2n})[(3\Delta\theta_{x1n} - \Delta\theta_{x2n})^2 + (3\Delta\theta_{y1n} - \Delta\theta_{y2n})^2$$
$$+ (3\Delta\theta_{z1n} - \Delta\theta_{z2n})^2]. \tag{118}$$

2.4. Sixth Order Taylor Series Algorithm

According to approximations derived in sections III.2.1, III.2.2, and III.2.3, one may derive a third form of the Taylor series approximation of sixth order. Taking Eq. (58) to the sixth order derivative yields

$$q_{0n+1} = q_{0n} + \left(\frac{dq_0}{dt}\right)_n \Delta T + \left(\frac{d^2q_0}{dt^2}\right)_n \frac{\Delta T^2}{2!} + \left(\frac{d^3q_0}{dt^3}\right)_n \frac{\Delta T^3}{3!}$$
$$+ \left(\frac{d^4q_0}{dt^4}\right)_n \frac{\Delta T^4}{4!} + \left(\frac{d^5q_0}{dt^5}\right)_n \frac{\Delta T^5}{5!} + \left(\frac{d^6q_0}{dt^6}\right)_n \frac{\Delta T^6}{6!}. \tag{119}$$

Taking the derivative of Eq. (105) and neglecting those terms higher than third order and terms containing derivatives of ω higher than first order (we assume that the angular acceleration $\dot{\omega}$ is constant over the whole process). After rearrangement, the following equation is acquired.

$$\left(\frac{d^4 q_0}{dt^4}\right)_n = q_0 \left[-\frac{3}{4}\left(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2\right)\right] + q_1 \left[\frac{1}{4}\left(3\omega_{xn}^2\dot{\omega}_{xn} + 2\omega_{xn}\omega_{yn}\dot{\omega}_{yn} + \omega_{yn}^2\dot{\omega}_{xn} + 2\omega_{xn}\omega_{zn}\dot{\omega}_{zn} + \omega_{zn}^2\dot{\omega}_{xn}\right)\right] + q_2 \left[\frac{1}{4}\left(2\omega_{xn}\omega_{yn}\dot{\omega}_{xn} + \omega_{xn}^2\dot{\omega}_{yn} + 3\omega_{yn}^2\dot{\omega}_{yn} + \omega_{xn}^2\dot{\omega}_{yn} + 3\omega_{yn}^2\dot{\omega}_{yn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{zn} + \omega_{zn}^2\dot{\omega}_{yn}\right)\right] + q_3 \left[\frac{1}{4}\left(2\omega_{xn}\omega_{zn}\dot{\omega}_{xn} + \omega_{xn}^2\dot{\omega}_{zn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{yn} + \omega_{yn}^2\dot{\omega}_{zn} + 3\omega_{zn}^2\dot{\omega}_{zn}\right)\right] . \tag{120}$$

Similarly, differentiating Eq. (120) and neglecting those terms mentioned previously,

$$\left(\frac{d^5 q_0}{dt^5}\right)_n = q_1 \left[\frac{15}{8}\omega_{xn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{xn}\dot{\omega}_{yn}^2 + \frac{7}{8}\omega_{xn}\dot{\omega}_{zn}^2 + \omega_{yn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{xn}\dot{\omega}_{zn}\right] + q_2 \left[\frac{15}{8}\omega_{yn}\dot{\omega}_{yn} + \frac{7}{8}\omega_{yn}\dot{\omega}_{xn} + \frac{7}{8}\omega_{yn}\dot{\omega}_{zn} + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{yn}\dot{\omega}_{zn}\right] + q_3 \left[\frac{15}{8}\omega_{zn}\dot{\omega}_{zn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{yn}^2 + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{zn} + \omega_{yn}\dot{\omega}_{yn}\dot{\omega}_{zn}\right] , \tag{121}$$

and

$$\left(\frac{d^6 q_0}{dt^6}\right)_n = q_1 \left[\frac{15}{8}\dot{\omega}_{xn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right] + q_2 \left[\frac{15}{8}\dot{\omega}_{yn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right] + q_3 \left[\frac{15}{8}\dot{\omega}_{zn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right] . \tag{122}$$

Substituting Eqs. (38), (82), (105), (120), (121) and (122) into Eq. (119), one may obtain

$$\begin{aligned} q_{0n+1} = q_{0n} &\left\{1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - \frac{\Delta T^3}{8}(\omega_{xn}\dot{\omega}_{xn} + \omega_{yn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{zn}) - \frac{\Delta T^4}{32}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\right\} \\ -q_{1n} &\left\{\frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4} - \frac{\Delta T^3}{48}[\omega_{xn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - 2(\omega_{yn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{yn})] - \frac{\Delta T^4}{96}[3\omega_{xn}^2\dot{\omega}_{xn} + 2\omega_{xn}\omega_{yn}\dot{\omega}_{yn} \right. \\ &\left. + \omega_{yn}^2\dot{\omega}_{xn} + 2\omega_{xn}\omega_{zn}\dot{\omega}_{zn} + \omega_{zn}^2\dot{\omega}_{xn}] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{xn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{xn}\dot{\omega}_{yn}^2 + \frac{7}{8}\omega_{xn}\dot{\omega}_{zn}^2 + \omega_{yn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{xn}\dot{\omega}_{zn}\right] \right. \\ &\left. - \frac{\Delta T^6}{384}[\dot{\omega}_{xn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)]\right\} \\ -q_{2n} &\left\{\frac{\omega_{yn}\Delta T}{2} + \frac{\omega_{yn}\Delta T^2}{4} - \frac{\Delta T^3}{48}[\omega_{yn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) + 2(\omega_{xn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{xn})] - \frac{\Delta T^4}{96}[3\omega_{yn}^2\dot{\omega}_{yn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{zn} \right. \\ &\left. + \omega_{zn}^2\dot{\omega}_{yn} + 2\omega_{xn}\omega_{yn}\dot{\omega}_{xn} + \omega_{xn}^2\dot{\omega}_{yn}] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{yn}\dot{\omega}_{yn}^2\right.\right. \end{aligned}$$

$$+ \frac{7}{8}\omega_{yn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{yn}\dot{\omega}_{zn}^2 + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{yn}\dot{\omega}_{zn}\Big]$$

$$- \frac{\Delta T^6}{384}\left[\dot{\omega}_{yn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right]\Bigg\}$$

$$-q_{3n}\Bigg\{\frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{zn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2)\right.$$

$$\left. - 2(\omega_{xn}\dot{\omega}_{yn} - \omega_{yn}\dot{\omega}_{xn})\right] - \frac{\Delta T^4}{96}\left[3\omega_{zn}^2\dot{\omega}_{zn} + 2(\omega_{xn}\omega_{zn}\dot{\omega}_{xn}\right.$$

$$\left. + \omega_{xn}^2\dot{\omega}_{zn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{yn} + \omega_{yn}^2\dot{\omega}_{zn}\right] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{zn}\dot{\omega}_{zn}^2\right.$$

$$\left. + \frac{7}{8}\omega_{zn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{yn}^2 + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{zn} + \omega_{yn}\dot{\omega}_{yn}\dot{\omega}_{zn}\right]$$

$$- \frac{\Delta T^6}{384}\left[\dot{\omega}_{zn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right]\Bigg\} . \qquad (123)$$

Now, letting

$$q_0' = 1 - \frac{\Delta T^2}{8}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - \frac{\Delta T^3}{8}(\omega_{xn}\dot{\omega}_{xn} + \omega_{yn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{zn})$$

$$- \frac{\Delta T^4}{32}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) , \qquad (124)$$

$$q_{10}' = \frac{\omega_{xn}\Delta T}{2} + \frac{\dot{\omega}_{xn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{xn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - 2(\omega_{yn}\dot{\omega}_{zn}\right.$$

$$\left. - \omega_{zn}\dot{\omega}_{yn})\right] - \frac{\Delta T^4}{96}\left[3\omega_{xn}^2\dot{\omega}_{xn} + 2\omega_{xn}\omega_{yn}\dot{\omega}_{yn} + \omega_{yn}^2\dot{\omega}_{xn}\right.$$

$$\left. + 2\omega_{xn}\omega_{zn}\dot{\omega}_{zn} + \omega_{zn}^2\dot{\omega}_{xn}\right] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{xn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{xn}\dot{\omega}_{yn}^2\right.$$

$$\left. + \frac{7}{8}\omega_{xn}\dot{\omega}_{zn}^2 + \omega_{yn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{xn}\dot{\omega}_{zn}\right] - \frac{\Delta T^6}{384}\left[\dot{\omega}_{xn}(\dot{\omega}_{xn}^2\right.$$

$$\left. + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right] \qquad (125)$$

$$q_{2\acute{0}} = \frac{\omega_{yn}\Delta T}{2} + \frac{\dot{\omega}_{yn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{yn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{yn}^2) + 2(\omega_{xn}\dot{\omega}_{zn} - \omega_{zn}\dot{\omega}_{xn})\right] - \frac{\Delta T^4}{96}\left[3\omega_{yn}^2\dot{\omega}_{yn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{zn} + \omega_{zn}^2\dot{\omega}_{yn} + 2\omega_{xn}\omega_{yn}\dot{\omega}_{xn} + \omega_{xn}^2\dot{\omega}_{yn}\right] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{yn}\dot{\omega}_{yn} + \frac{7}{8}\omega_{yn}\dot{\omega}_{zn} + \frac{7}{8}\omega_{yn}\dot{\omega}_{zn}^2 + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{yn} + \omega_{zn}\dot{\omega}_{yn}\dot{\omega}_{zn}\right] - \frac{\Delta T^6}{384}\left[\dot{\omega}_{yn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right], \tag{126}$$

and

$$q_{3\acute{0}} = \frac{\omega_{zn}\Delta T}{2} + \frac{\dot{\omega}_{zn}\Delta T^2}{4} - \frac{\Delta T^3}{48}\left[\omega_{zn}(\omega_{xn}^2 + \omega_{yn}^2 + \omega_{zn}^2) - 2(\omega_{xn}\dot{\omega}_{yn} - \omega_{yn}\dot{\omega}_{xn})\right] - \frac{\Delta T^4}{96}\left[3\omega_{zn}^2\dot{\omega}_{zn} + 2\omega_{xn}\omega_{zn}\dot{\omega}_{xn} + \omega_{xn}^2\dot{\omega}_{zn} + 2\omega_{yn}\omega_{zn}\dot{\omega}_{yn} + \omega_{yn}^2\dot{\omega}_{zn}\right] - \frac{\Delta T^5}{120}\left[\frac{15}{8}\omega_{zn}\dot{\omega}_{zn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{xn}^2 + \frac{7}{8}\omega_{zn}\dot{\omega}_{yn}^2 + \omega_{xn}\dot{\omega}_{xn}\dot{\omega}_{zn} + \omega_{yn}\dot{\omega}_{yn}\dot{\omega}_{zn}\right] - \frac{\Delta T^6}{384}\left[\dot{\omega}_{zn}(\dot{\omega}_{xn}^2 + \dot{\omega}_{yn}^2 + \dot{\omega}_{zn}^2)\right]. \tag{127}$$

Equation (123) becomes

$$q_{0n+1} = q_{0n}q_{0}^{\prime} - q_{1n}q_{1}^{\prime} - q_{2n}q_{2}^{\prime} - q_{3n}q_{3}^{\prime}. \tag{128}$$

Similarly, it can be shown that

$$q_{1n+1} = q_{0n}q_{1}^{\prime} + q_{1n}q_{0}^{\prime} + q_{2n}q_{3}^{\prime} - q_{3n}q_{2}^{\prime}, \tag{129}$$

$$q_{2n+1} = q_{0n}q_{2}^{\prime} - q_{1n}q_{3}^{\prime} + q_{2n}q_{0}^{\prime} + q_{3n}q_{1}^{\prime}, \tag{130}$$

$$q_{3n+1} = q_{0n}q_3' + q_{1n}q_2' - q_{2n}q_1' + q_{3n}q_0' , \tag{131}$$

where q_0', q_1', q_2', q_3' are as shown in Eqs. (124) to (127) respectively. Substituting Eqs. (73) and (74) into Eqs. (124) to (127) yields

$$q_0' = 1 - \frac{1}{8}\left[(\Delta\theta_{x1n} + \Delta\theta_{x2n})^2 + (\Delta\theta_{y1n} + \Delta\theta_{y2n})^2 + (\Delta\theta_{z1n} + \Delta\theta_{z2n})^2\right] , \tag{132}$$

$$\begin{aligned} q_{10}' = {} & \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) + \frac{1}{3}(\Delta\theta_{y1n}\Delta\theta_{z2n} - \Delta\theta_{y2n}\Delta\theta_{z1n}) \\ & - \frac{1}{48}(\Delta\theta_{x1n}^3 + 3\Delta\theta_{x1n}^2 + 3\Delta\theta_{x1n}\Delta\theta_{x2n}^2 + \Delta\theta_{x2n}^3 \\ & \quad + \Delta\theta_{x1n}\Delta\theta_{y1n}^2 + \Delta\theta_{x2n}\Delta\theta_{y2n}^2 + \Delta\theta_{x2n}\Delta\theta_{z2n}^2 + \Delta\theta_{x1n}\Delta\theta_{z1n}^2) \\ & + \frac{1}{40}(\Delta\theta_{x1n}\Delta\theta_{y1n}\Delta\theta_{y2n} + \Delta\theta_{x2n}\Delta\theta_{y1n}\Delta\theta_{y2n} + \Delta\theta_{x1n}\Delta\theta_{z1n}\Delta\theta_{z2n} \\ & \quad + \Delta\theta_{x2n}\Delta\theta_{z1n}\Delta\theta_{z2n}) \\ & - \frac{7}{80}(\Delta\theta_{x1n}\Delta\theta_{y2n}^2 + \Delta\theta_{x2n}\Delta\theta_{y1n}^2 + \Delta\theta_{x1n}\Delta\theta_{z2n}^2 + \Delta\theta_{x2n}\Delta\theta_{z1n}^2) , \end{aligned} \tag{133}$$

$$\begin{aligned} q_{20}' = {} & \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) - \frac{1}{3}(\Delta\theta_{x1n}\Delta\theta_{z2n} - \Delta\theta_{x2n}\Delta\theta_{z1n}) \\ & - \frac{1}{48}(\Delta\theta_{y1n}^3 + 3\Delta\theta_{y1n}^2\Delta\theta_{y2n} + 3\Delta\theta_{y1n}\Delta\theta_{y2n}^2 + \Delta\theta_{y2n}^3 \\ & \quad + \Delta\theta_{y1n}\Delta\theta_{z1n}^2 + \Delta\theta_{y2n}\Delta\theta_{z2n}^2 + \Delta\theta_{x2n}^2\Delta\theta_{y2n} + \Delta\theta_{x1n}^2\Delta\theta_{y1n}) \\ & + \frac{1}{40}(\Delta\theta_{y1n}\Delta\theta_{z1n}\Delta\theta_{z2n} + \Delta\theta_{y2n}\Delta\theta_{z1n} + \Delta\theta_{z2n}) \\ & \quad + \Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{y1n} + \Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{y2n}) \end{aligned}$$

$$- \frac{7}{80} (\Delta\theta_{y1n}\Delta\theta_{z2n}^2 + \Delta\theta_{y2n}\Delta\theta_{z1n}^2 + \Delta\theta_{x2n}^2\Delta\theta_{y1n} + \Delta\theta_{x1n}^2\Delta\theta_{y2n}) , \tag{134}$$

and

$$\begin{aligned} q_{30}' = {} & \frac{1}{2} (\Delta\theta_{z1n} + \Delta\theta_{z2n}) + \frac{1}{3} (\Delta\theta_{x1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n}) \\ & - \frac{1}{48} (\Delta\theta_{z1n}^3 + 3\Delta\theta_{z1n}^2\Delta\theta_{z2n} + 3\Delta\theta_{z1n}\Delta\theta_{z2n}^2 + \Delta\theta_{z2n}^3 \\ & \quad + \Delta\theta_{x2n}^2\Delta\theta_{z2n} + \Delta\theta_{x1n}^2\Delta\theta_{z1n} + \Delta\theta_{y2n}^2\Delta\theta_{z2n} + \Delta\theta_{y1n}^2\Delta\theta_{z1n}) \\ & + \frac{1}{40} (\Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z1n} + \Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z2n} + \Delta\theta_{y1n}\Delta\theta_{y2n}\Delta\theta_{z1n} \\ & \quad + \Delta\theta_{y1n}\Delta\theta_{y2n}\Delta\theta_{z2n}) \\ & - \frac{7}{80} (\Delta\theta_{x2n}^2\Delta\theta_{z1n} + \Delta\theta_{x1n}^2\Delta\theta_{z2n} + \Delta\theta_{y2n}^2\Delta\theta_{z1n} + \Delta\theta_{y1n}^2\Delta\theta_{z2n}) \end{aligned} \tag{135}$$

In order to reduce the algorithm to a simpler form, it is possible to neglect some third-order terms with negligible effect on computer accuracy. Let

$$\begin{aligned} q_{11}' = {} & \frac{1}{15} (\Delta\theta_{x1n}\Delta\theta_{y2n}^2 + \Delta\theta_{x2n}\Delta\theta_{y1n}^2 + \Delta\theta_{x1n}\Delta\theta_{z2n}^2 + \Delta\theta_{x2n}\Delta\theta_{z1n}^2 \\ & - \Delta\theta_{x1n}\Delta\theta_{y1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n}\Delta\theta_{y2n} - \Delta\theta_{x1n}\Delta\theta_{z1n}\Delta\theta_{z2n} \\ & - \Delta\theta_{x2n}\Delta\theta_{z1n}\Delta\theta_{z2n}) , \end{aligned} \tag{136}$$

$$\begin{aligned} q_{21}' = {} & \frac{1}{15} (\Delta\theta_{y1n}\Delta\theta_{z2n}^2 + \Delta\theta_{y2n}\Delta\theta_{z1n}^2 + \Delta\theta_{x1n}\Delta\theta_{z2n}^2 + \Delta\theta_{x2n}\Delta\theta_{z1n}^2 \\ & - \Delta\theta_{x1n}\Delta\theta_{y1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n}\Delta\theta_{y2n} - \Delta\theta_{x1n}\Delta\theta_{z1n}\Delta\theta_{z2n} \\ & - \Delta\theta_{x2n}\Delta\theta_{z1n}\Delta\theta_{z2n}) , \end{aligned} \tag{137}$$

and

$$q_{31}' = \frac{1}{15}\left(\Delta\theta_{x2n}{}^{2}\Delta\theta_{z1n} + \Delta\theta_{x1n}{}^{2}\Delta\theta_{z2n} + \Delta\theta_{y2n}{}^{2}\Delta\theta_{z1n} + \Delta\theta_{y1n}{}^{2}\Delta\theta_{z2n}\right.$$
$$- \Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z1n} - \Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z2n} - \Delta\theta_{y1n}\Delta\theta_{y2n}$$
$$\left. - \Delta\theta_{y1n}\Delta\theta_{y2n}\Delta\theta_{z2n}\right) . \qquad (138)$$

Adding Eqs. (136), (137) and (138) into Eqs. (133), (134) and (135) respectively, one may obtain

$$q_1' = \frac{1}{2}\left(\Delta\theta_{x1n} + \Delta\theta_{x2n}\right) + \frac{1}{3}\left(\Delta\theta_{y1n}\Delta\theta_{z2n} - \Delta\theta_{y2n}\Delta\theta_{z1n}\right)$$
$$- \frac{1}{48}\left(\Delta\theta_{x1n}{}^{3} + 3\Delta\theta_{x1n}{}^{2}\Delta\theta_{x2n} + 3\Delta\theta_{x1n}\Delta\theta_{x2n}{}^{2} + \Delta\theta_{x2n}{}^{3}\right.$$
$$+ \Delta\theta_{x2n}\Delta\theta_{y2n}{}^{2} + 2\Delta\theta_{x1n}\Delta\theta_{y1n}\Delta\theta_{y2n} + \Delta\theta_{x1n}\Delta\theta_{y2n}{}^{2}$$
$$+ \Delta\theta_{x2n}\Delta\theta_{y1n}{}^{2} + 2\Delta\theta_{x2n}\Delta\theta_{y1n}\Delta\theta_{y2n} + 2\Delta\theta_{x1n}\Delta\theta_{z1n}\Delta\theta_{z2n}$$
$$+ \Delta\theta_{x1n}\Delta\theta_{z2n}{}^{2} + \Delta\theta_{x2n}\Delta\theta_{z1n}{}^{2} + 2\Delta\theta_{x2n}\Delta\theta_{z1n}\Delta\theta_{z2n}$$
$$\left. + \Delta\theta_{x2n}\Delta\theta_{z2n}{}^{2} + \Delta\theta_{x1n}\Delta\theta_{z1n}{}^{2}\right) , \qquad (139)$$

$$q_2' = \frac{1}{2}\left(\Delta\theta_{y1n} + \Delta\theta_{y2n}\right) + \frac{1}{3}\left(\Delta\theta_{x2n}\Delta\theta_{z1n} - \Delta\theta_{x1n}\Delta\theta_{z2n}\right)$$
$$- \frac{1}{48}\left(\Delta\theta_{y1n}{}^{3} + 3\Delta\theta_{y1n}{}^{2}\Delta\theta_{y2n} + 3\Delta\theta_{y1n}\Delta\theta_{y2n}{}^{2} + \Delta\theta_{y2n}{}^{3}\right.$$
$$+ \Delta\theta_{y1n}\Delta\theta_{z1n}{}^{2} + \Delta\theta_{y2n}\Delta\theta_{z2n}{}^{2} + 2\Delta\theta_{y1n}\Delta\theta_{z1n}\Delta\theta_{z2n}$$
$$+ \Delta\theta_{y1n}\Delta\theta_{z2n}{}^{2} + \Delta\theta_{y2n}\Delta\theta_{z1n}{}^{2} + 2\Delta\theta_{y2n}\Delta\theta_{z1n}\Delta\theta_{z2n}$$
$$+ 2\Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{y1n} + \Delta\theta_{x2n}{}^{2}\Delta\theta_{y1n} + \Delta\theta_{x1n}{}^{2}\Delta\theta_{y2n}$$
$$\left. + 2\Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{y2n} + \Delta\theta_{x2n}{}^{2}\Delta\theta_{y2n} + \Delta\theta_{x1n}\Delta\theta_{y1n}\right) , \qquad (140)$$

$$q_3' = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) + \frac{1}{3}(\Delta\theta_{x1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n})$$
$$- \frac{1}{48}(\Delta\theta_{z1n}^3 + 3\Delta\theta_{z1n}^2\Delta\theta_{z2n} + 3\Delta\theta_{z1n}\Delta\theta_{z2n}^2 + \Delta\theta_{z2n}^3$$
$$+ \Delta\theta_{x1n}^2\Delta\theta_{z1n} + \Delta\theta_{x2n}^2\Delta\theta_{z2n} + 2\Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z1n}$$
$$+ \Delta\theta_{x2n}^2\Delta\theta_{z1n} + \Delta\theta_{x1n}^2\Delta\theta_{z2n} + 2\Delta\theta_{x1n}\Delta\theta_{x2n}\Delta\theta_{z2n}$$
$$+ 2\Delta\theta_{y1n}\Delta\theta_{y2n}\Delta\theta_{z1n} + \Delta\theta_{y2n}^2\Delta\theta_{z1n} + \Delta\theta_{y1n}^2\Delta\theta_{z2n}$$
$$+ 2\Delta\theta_{y1n}\Delta\theta_{y2n}\Delta\theta_{z2n} + \Delta\theta_{y2n}^2\Delta\theta_{z2n} + \Delta\theta_{y1n}^2\Delta\theta_{z1n}) \quad . \tag{141}$$

Now, letting

$$\alpha_1 = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) \quad , \tag{142}$$

$$\alpha_2 = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) \quad , \tag{143}$$

$$\alpha_3 = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) \quad , \tag{144}$$

$$\beta = \frac{1}{2}(\alpha_1^2 + \alpha_2^2 + \alpha_3^2) \quad , \tag{145}$$

one can acquire the following form

$$q_0' = 1 - \beta \quad , \tag{146}$$

$$q_1' = \alpha_1 + \frac{1}{3}[(\Delta\theta_{y1n}\Delta\theta_{z2n} - \Delta\theta_{y2n}\Delta\theta_{z1n}) - \alpha_1\beta] \quad , \tag{147}$$

$$q_2' = \alpha_2 + \frac{1}{3}[(\Delta\theta_{x2n}\Delta\theta_{z1n} - \Delta\theta_{x1n}\Delta\theta_{z2n}) - \alpha_2\beta] \quad , \tag{148}$$

$$q_3' = \alpha_3 + \frac{1}{3}[(\Delta\theta_{x1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n}) - \alpha_3\beta] \quad . \tag{149}$$

where α_1, α_2, α_3, β are shown in Eqs. (142) to (145).

As a result, the difference equations of Approach II are

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}_{n+1} = \begin{bmatrix} q_0' & -q_1' & -q_2' & -q_3' \\ q_1' & q_0' & q_3' & -q_2' \\ q_2' & -q_3' & q_0' & q_1' \\ q_3' & q_2' & -q_1' & q_0' \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} . \tag{99}$$

In sixth order form q_0', q_1', q_2' and q_3' are shown in Eqs. (146) to (149) respectively.

2.5. Higher Order Taylor Series Approximation

Higher order Taylor series approximations will not be considered, since there are no terms lower than third order when the derivative of the Taylor series expansion in Eq. (58) is higher than sixth order.

CHAPTER IV

SIMULATION TEST OF QUATERNION UPDATE ALGORITHMS

Simulation tests were conducted to compare the precision of the two algorithms which are listed in Table 1. These tests are

1. Calculating the deviation of the unit Norm of quaternion.
2. Comparing the errors between input angular increments ($\Delta\theta$) and angular increments obtained by Euler angle transformation ($\Delta\theta e$).
3. Comparing the accuracy of the updating Quaternion.

The following quaternion is assumed as the initial position for this simulation

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

In order to obtain a higher precision result, double precision was used in all the computations.

I. ANALYSIS OF THE ALGORITHMS

The two algorithms shown in Table 1 will be compared for different angular increments. Table 2 shows the difference between quaternion elements for Algorithm I and Algorithm II. Three different cases will be discussed.

Table 1

Comparison of Algorithm I and II

Algorithm I	Algorithm II
$q_0' = 1 - \frac{1}{8}[(\Delta\theta_{x1n} + \Delta\theta_{x2n})^2 + (\Delta\theta_{y1n} + \Delta\theta_{y2n})^2 + (\Delta\theta_{z1n} + \Delta\theta_{z2n})^2]$	$q_0' = 1 - \beta$
$q_1' = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n}) + \frac{1}{4}(\Delta\theta_{y2n}\Delta\theta_{z1n} - \Delta\theta_{z2n}\Delta\theta_{y1n}) - \frac{1}{48}[\Delta\theta_{x2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \theta_{x1n}(3\Delta\theta_{2n}^2 + \theta_{1n}^2)]$	$q_1' = \alpha_1 + \frac{1}{3}[(\Delta\theta_{y1n}\Delta\theta_{z2n} - \Delta\theta_{y2n}\Delta\theta_{z1n} - \alpha_1\beta)]$
$q_2' = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n}) + \frac{1}{4}(\Delta\theta_{x1n}\Delta\theta_{z2n} - \Delta\theta_{x2n}\Delta\theta_{z1n})^2 - \frac{1}{48}[\Delta\theta_{y2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \Delta\theta_{y1n}(3\Delta\theta_{2n}^2 + \Delta\theta_{1n}^2)]$	$q_2' = \alpha_2 + \frac{1}{3}[(\Delta\theta_{x2n}\Delta\theta_{z1n} - \Delta\theta_{x1n}\Delta\theta_{z2n} - \alpha_2\beta)]$
$q_3' = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n}) + \frac{1}{4}(\Delta\theta_{x2n}\Delta\theta_{y1n} - \Delta\theta_{y2n}\Delta\theta_{x1n}) - \frac{1}{48}[\Delta\theta_{z2n}(3\Delta\theta_{1n}^2 + \Delta\theta_{2n}^2) + \Delta\theta_{z1n}(3\Delta\theta_{2n}^2 + \Delta\theta_{1n}^2)]$	$q_3' = \alpha_3 + \frac{1}{3}[(\Delta\theta_{x1n}\Delta\theta_{y2n} - \Delta\theta_{x2n}\Delta\theta_{y1n} - \alpha_3\beta)]$
where	where
$\Delta\theta_{1n}^2 = \Delta\theta_{x1n}^2 + \Delta\theta_{y1n}^2 + \Delta\theta_{z1n}^2$	$\alpha_1 = \frac{1}{2}(\Delta\theta_{x1n} + \Delta\theta_{x2n})$
$\Delta\theta_{2n}^2 = \Delta\theta_{x2n}^2 + \Delta\theta_{y2n}^2 + \Delta\theta_{z2n}^2$	$\alpha_2 = \frac{1}{2}(\Delta\theta_{y1n} + \Delta\theta_{y2n})$
	$\alpha_3 = \frac{1}{2}(\Delta\theta_{z1n} + \Delta\theta_{z2n})$
	$\beta = \frac{1}{2}(\alpha_1^2 + \alpha_2^2 + \alpha_3^2)$

Table 2

Difference between Algorithm I and Algorithm II

	Q' (Algorithm II) minus Q' (Algorithm I)
q_0'	Zero
q_1'	$\frac{7}{12}\Delta\theta_{y1}\Delta\theta_{z2} - \frac{7}{12}\Delta\theta_{y2}\Delta\theta_{z1} + \frac{1}{48}(2\Delta\theta_{x1}\Delta\theta_{y2}^2 + 2\Delta\theta_{x2}\Delta\theta_{y1}^2 + 2\Delta\theta_{x1}\Delta\theta_{z2}^2 + 2\Delta\theta_{x2}\Delta\theta_{z1}^2 - 2\Delta\theta_{x1}\Delta\theta_{y1}\Delta\theta_{y2} - 2\Delta\theta_{x2}\Delta\theta_{y1}\Delta\theta_{y2} - 2\Delta\theta_{x1}\Delta\theta_{z1}\Delta\theta_{z2} - 2\Delta\theta_{x2}\Delta\theta_{z1}\Delta\theta_{z2})$
q_2'	$\frac{7}{12}\Delta\theta_{x2}\Delta\theta_{z1} - \frac{7}{12}\Delta\theta_{x1}\Delta\theta_{z2} + \frac{1}{48}(2\Delta\theta_{x2}^2\Delta\theta_{y1} + 2\Delta\theta_{x1}^2\Delta\theta_{y2} + 2\Delta\theta_{y1}\Delta_{z2}^2 + 2\Delta\theta_{y2}\Delta\theta_{z1}^2 - 2\Delta\theta_{y1}\Delta\theta_{z1}\Delta\theta_{z2} - 2\Delta\theta_{y2}\Delta\theta_{z1}\Delta\theta_{z2} - 2\Delta\theta_{x1}\Delta\theta_{x2}\Delta\theta_{y1} - 2\Delta\theta_{x1}\Delta\theta_{x2}\Delta\theta_{y2})$
q_3'	$\frac{7}{12}\Delta\theta_{x1}\Delta\theta_{y2} - \frac{7}{12}\Delta\theta_{x2}\Delta\theta_{y1} + \frac{1}{48}(2\Delta\theta_{x2}^2\Delta\theta_{z1} + 2\Delta\theta_{x1}^2\Delta\theta_{z2} + 2\Delta\theta_{y2}^2\Delta\theta_{z1} + 2\Delta\theta_{y1}^2\Delta\theta_{z2} - 2\Delta\theta_{x1}\Delta\theta_{x2}\Delta\theta_{z1} - 2\Delta\theta_{x1}\Delta\theta_{x2}\Delta\theta_{z2} - 2\Delta\theta_{y1}\Delta\theta_{y2}\Delta\theta_{z1} - 2\Delta\theta_{y1}\Delta\theta_{y2}\Delta\theta_{z2})$

Case I.

If all the angular increments are equal, i.e.,

$$\Delta\theta_{x1} = \Delta\theta_{x2} = \Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1} = \Delta\theta_{z2} \ ;$$

then the differences are all zero. The two algorithms give the same result.

Case II.

If the angular increments are:

$$\Delta\theta_{x1} = \Delta\theta_{y1} = \Delta\theta_{z1} \ ,$$

$$\Delta\theta_{x2} = \Delta\theta_{y2} = \Delta\theta_{z2} = r \cdot \Delta\theta_{x1} \ ,$$

where r is a real number, then the difference (D) will also be zero, since

$$D = \frac{1}{48} (2r^2\Delta\theta_{x1}^3 + 2r\Delta\theta_{x1}^3 + 2r^2\Delta\theta_{x1}^3 + 2r\Delta\theta_{x1}^3$$

$$- 2r\Delta\theta_{x1}^3 - 2r^2\Delta\theta_{x1}^3 - 2r\Delta\theta_{x1}^3 - 2r\Delta\theta_{x1}^3) \ ,$$

$$D = 0 \ .$$

The two algorithms, again, give the same result.

Case III.

For the angular inputs other than those of Case I and Case II, the difference value will not be zero. The comparison will be shown by the following simulation.

II. NORM CHECKING METHOD

An exact set of quaternion satisfies the following condition

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \ .$$

Because of inexact computations, the computed quaternion set will gradually depart from this ideal relationship. In this method, the amount of deviation from this relationship is used as a measure for the quality of quaternion update algorithm.

The steps of this simulation are as follows:

1. Set the initial position.
2. Compute the update quaternion.
3. Store the update quaternion elements for use next update.
4. Give a set of angular increments.
5. Use the updated quaternion elements and the given angular increments to obtain another set of updated quaternion elements.
6. These steps are repeated.

Figure 2 shows the flow chart for this process.

Many different angular increments have been tried. Typical results are shown in Tables 3 and 4. Figure 3 shows the results of these tables. One finds that the deviation of Algorithm II is less than Algorithm I. The results for other angular increments are always the same as the typical example. Therefore, one may conclude that Algorithm II preserves unity norm better than Algorithm I.

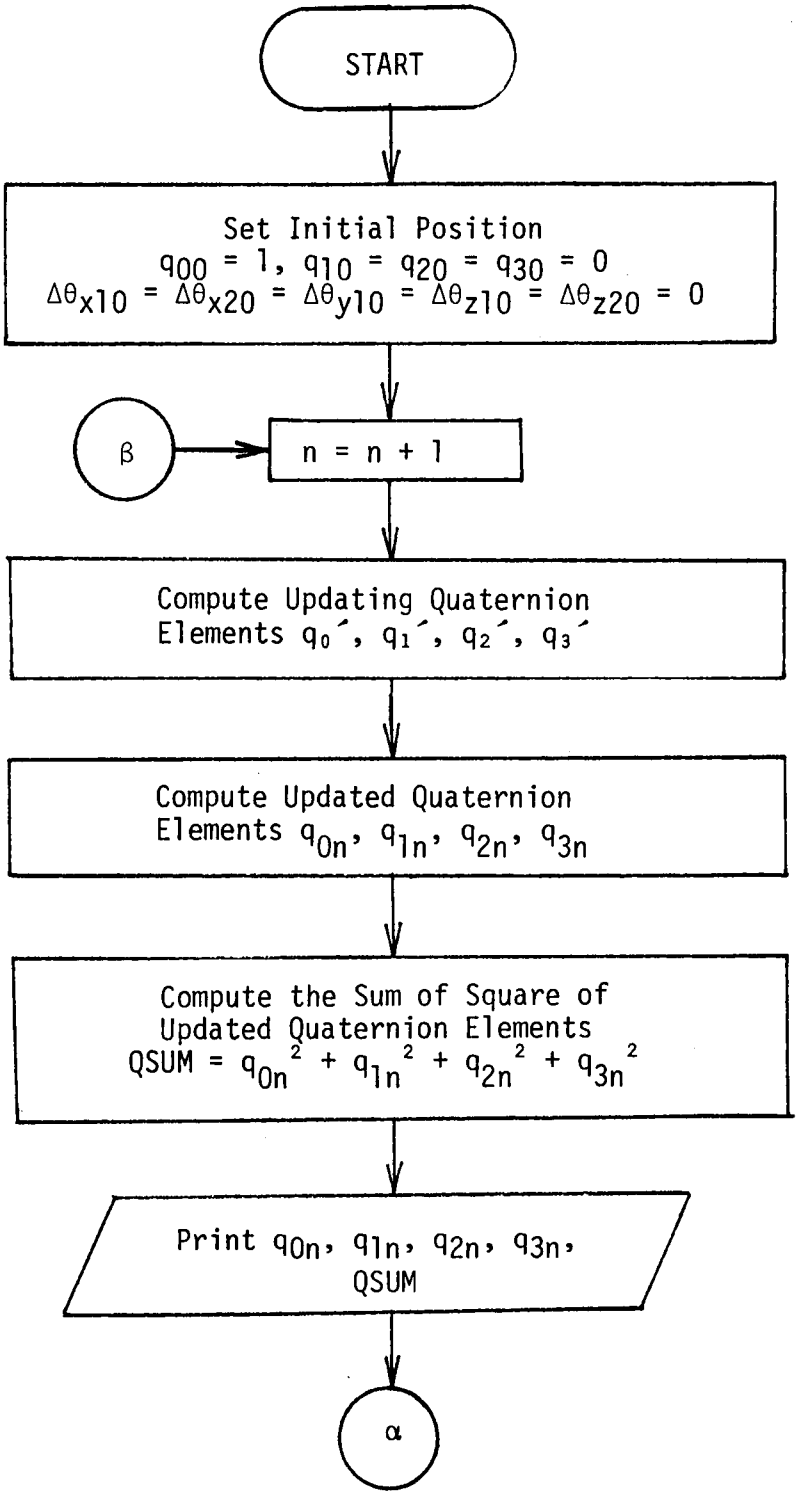


Figure 2. Flow Chart of Computation of Sum of Square of Update Quaternion Elements.

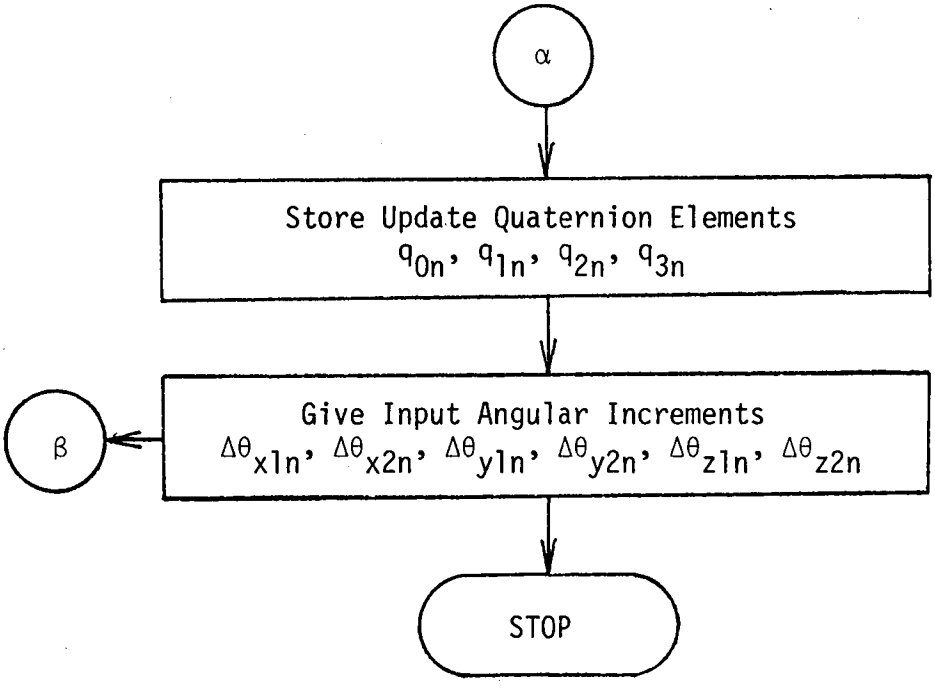


Figure 2 (Continued)

Table 3

Result of Sum of Update Quaternion Elements for Algorithm I, Using $\Delta\theta_{x1} = 0.001°$, $\Delta\theta_{x2} = 0.002°$, $\Delta\theta_{y1} = \Delta\theta_{y2} = 0.0015°$, $\Delta\theta_{z1} = \Delta\theta_{z2} = 0.0015°$
(Time Used: 3.19 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	$Q_0^2 + Q_1^2 + Q_2^2 + Q_3^2$
0.0	0.10000000000000000D 01
0.3000D-02	0.99999999996109300D 00
0.6000D-02	0.99999999992218700D 00
0.9000D-02	0.99999999988328100D 00
0.1200D-01	0.99999999984437400D 00
0.1500D-01	0.99999999980546700D 00
0.1800D-01	0.99999999976656100D 00
0.2100D-01	0.99999999972765400D 00
0.2400D-01	0.99999999968874700D 00
0.2700D-01	0.99999999964984000D 00
0.3000D-01	0.99999999961093300D 00

Table 4

Result of Sum of Update Quaternion Elements for Algorithm II, Using $\Delta\theta_{x1} = 0.001°$, $\Delta\theta_{x2} = 0.002°$, $\Delta\theta_{y1} = \Delta\theta_{y2} = 0.0015°$, $\Delta\theta_{z1} = \Delta\theta_{z2} = 0.0015°$ (Time Used: 1.93 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	$Q_0^2 + Q_1^2 + Q_2^2 + Q_3^2$
0.0	0.10000000000000000D 01
0.3000D-02	0.99999999996703100D 00
0.6000D-02	0.99999999993406200D 00
0.9000D-02	0.99999999990109300D 00
0.1200D-01	0.99999999986812400D 00
0.1500D-01	0.99999999983515500D 00
0.1800D-01	0.99999999980218500D 00
0.2100D-01	0.99999999976921600D 00
0.2400D-01	0.99999999973624700D 00
0.2700D-01	0.99999999970327800D 00
0.3000D-01	0.99999999967030900D 00

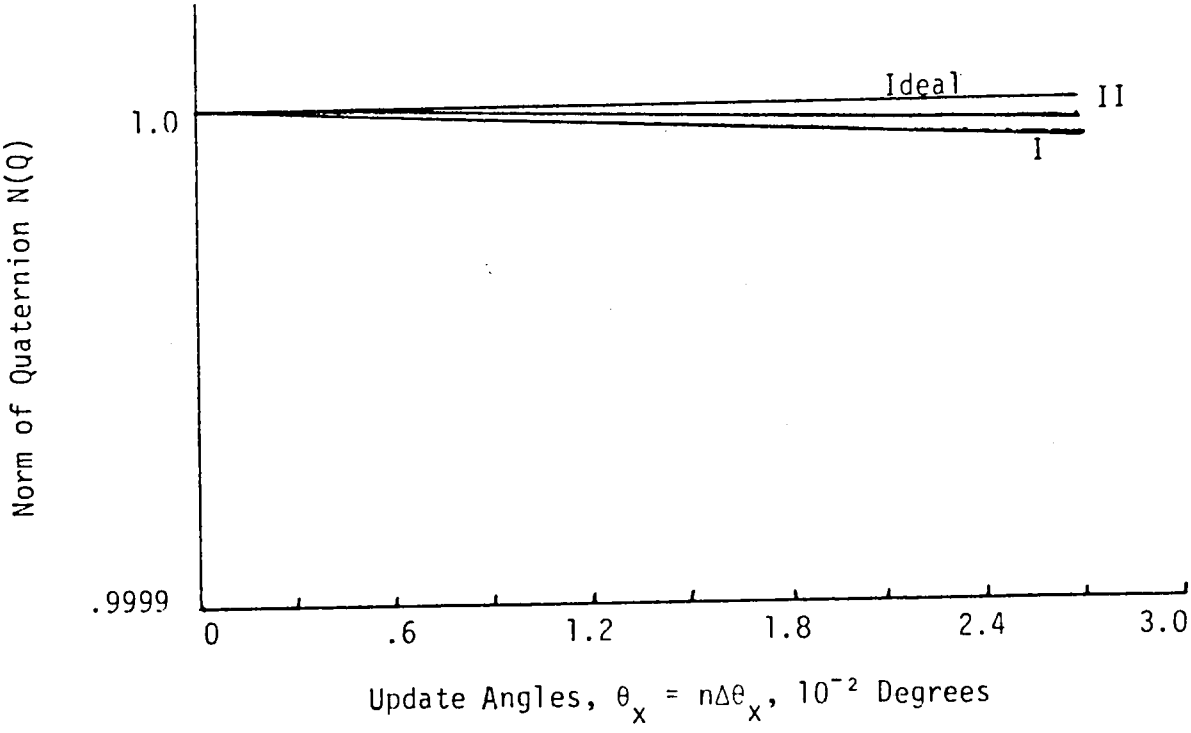


Figure 3. Comparison of unity norm, N(Q), of Algorithms I and II for

$\Delta\theta_{x1} = 0.001°$ $\Delta\theta_{x2} = 0.002°$

$\Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1} = \Delta\theta_{z2} = 0.0015°$

III. COMPARISON OF COMPUTED INPUT ANGLES

The errors generated by different input angular increments will be compared in this section. These errors were computed by the difference between input angular increments and Euler angular increments.

In the case where the three axes of the coordinate system are rotated simultaneously, the updating angles of each axis are not the successive summation of angular increments. But if we assume that only one axis is rotated in a coordinate system, the updating angles are equal to the successive summation of the angular increments. One axis will be rotated in this simulation. Under the single axis rotation condition, Table 5 shows that the formula for Algorithm II is the same as for Algorithm I. Therefore, one may use either one of the two Algorithms for the simulation test.

Figure 4 shows the block diagram of this simulation. The transformation of two coordinate systems, S_1 and S_2, can be expressed by the Euler angular transformation matrix or the quaternion transformation matrix. In quaternion form, this is shown in Eq. (25)

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = [M] \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} , \tag{25}$$

where

$$[M] = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_1q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix} . \tag{23}$$

Table 5

Comparison of Algorithms I and II in Special Cases

Case	Algorithm I	Algorithm II
$\Delta\theta_{y1}$, $\Delta\theta_{y2}$, $\Delta\theta_{z1}$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{x1} + \Delta\theta_{x2})^2$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{x1} + \Delta\theta_{x2})^2$
$\Delta\theta_{z2}$ equal to 0	$q_1' = \frac{1}{2}(\Delta\theta_{x1} + \Delta\theta_{x2})$	$q_1' = \frac{1}{2}(\Delta\theta_{x1} + \Delta\theta_{x2})$
	$q_2' = 0$	$q_2' = 0$
	$q_3' = 0$	$q_3' = 0$
$\Delta\theta_{x1}$, $\Delta\theta_{x2}$, $\Delta\theta_{z1}$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{y1} + \Delta\theta_{y2})^2$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{y1} + \Delta\theta_{y2})^2$
$\Delta\theta_{z2}$ equal to 0	$q_1' = 0$	$q_1' = 0$
	$q_2' = \frac{1}{2}(\Delta\theta_{y1} + \Delta\theta_{y2})$	$q_2' = \frac{1}{2}(\Delta\theta_{y1} + \Delta\theta_{y2})$
	$q_3' = 0$	$q_3' = 0$
$\Delta\theta_{x1}$, $\Delta\theta_{x2}$, $\Delta\theta_{y1}$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{z1} + \Delta\theta_{z2})^2$	$q_0' = 1 - \frac{1}{8}(\Delta\theta_{z1} + \Delta\theta_{z2})^2$
$\Delta\theta_{y2}$ equal to 0	$q_1' = 0$	$q_1' = 0$
	$q_2' = 0$	$q_2' = 0$
	$q_3' = \frac{1}{2}(\Delta\theta_{z1} + \Delta\theta_{z2})^2$	$q_3' = \frac{1}{2}(\Delta\theta_{z1} + \Delta\theta_{z2})^2$

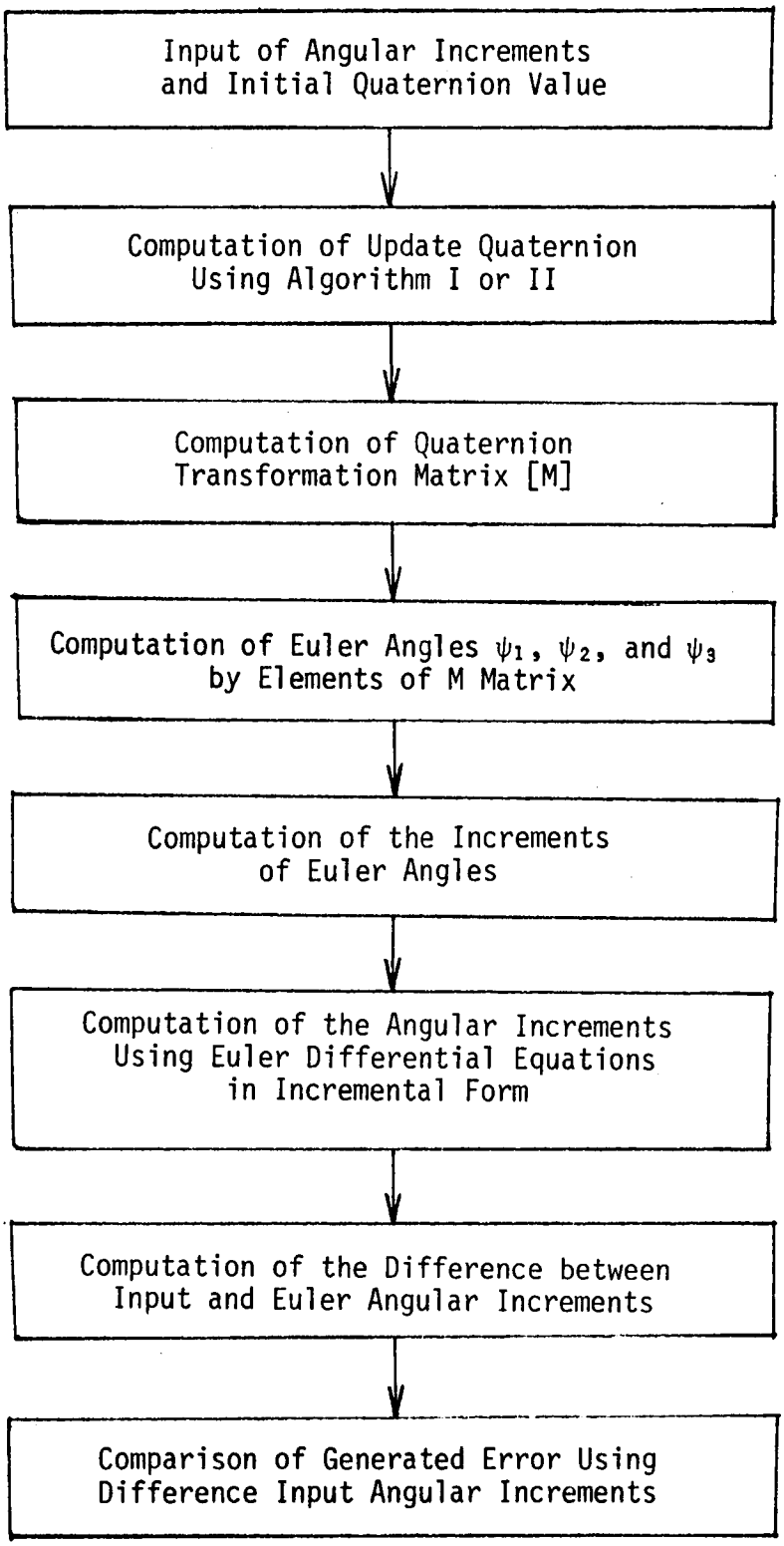


Figure 4. Block Diagram of Computation of Euler Angular Increments.

In Euler form this is shown in Appendix III as

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = [B] \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} , \qquad \text{(A-III-1)}$$

where

$$[B] = \begin{bmatrix} C_2C_3 & -C_1C_2S_3 + S_1S_2 & S_1C_2S_2 + C_1S_2 \\ S_3 & C_1C_3 & -S_1C_3 \\ -S_2C_3 & C_1S_2C_3 + S_1C_2 & -S_1S_2S_3 + C_1C_2 \end{bmatrix} . \qquad (150)$$

In matrix B

$$\left.\begin{aligned} C_1 &= \cos\psi_1 \\ C_2 &= \cos\psi_2 \\ C_3 &= \cos\psi_3 \end{aligned}\right\} \text{, and } \left.\begin{aligned} S_1 &= \sin\psi_1 \\ S_2 &= \sin\psi_2 \\ S_3 &= \sin\psi_3 \end{aligned}\right\} ,$$

where ψ_1, ψ_2, and ψ_3 are roll, pitch and Yaw angles respectively.

One may see that the corresponding elements of matrices M and B are equal. Therefore, Euler angles ψ_1, ψ_2, ψ_3 can be computed by the update quaternion; the relationships are directly derived from matrices M and B as

$$\left.\begin{aligned} \psi_1 &= -\tan^{-1}\frac{M_{23}}{M_{22}} \\ \psi_2 &= -\tan^{-1}\frac{M_{31}}{M_{11}} \\ \psi_3 &= \tan^{-1}\frac{M_{21}}{\sqrt{M_{22}^2 + M_{23}^2}} \end{aligned}\right\} . \qquad (151)$$

The increments of Euler angles can be defined as follows

$$\left.\begin{aligned}\Delta\psi_1 &= \psi_{1n+1} - \psi_{1n}\\ \Delta\psi_2 &= \psi_{2n+1} - \psi_{2n}\\ \Delta\psi_3 &= \psi_{3n+1} - \psi_{3n}\end{aligned}\right\} . \tag{152}$$

The Euler differential equations in incremental form are also shown in Appendix III as

$$\begin{bmatrix}\Delta\theta_{xe}\\ \Delta\theta_{ye}\\ \Delta\theta_{ze}\end{bmatrix} = \begin{bmatrix}\cos\psi_2\cos\psi_3 & 0 & \sin\psi_2\\ \sin\psi_3 & 1 & 0\\ -\sin\psi_2\sin\psi_3 & 0 & \cos\psi_2\end{bmatrix} \begin{bmatrix}\Delta\psi_1\\ \Delta\psi_2\\ \Delta\psi_3\end{bmatrix} . \tag{153}$$

where $\Delta\theta_{xe}$, $\Delta\theta_{ye}$ and $\Delta\theta_{ze}$ are Euler angular increments. Substituting Eqs. (151) and (152) into (153), one can obtain the values of Euler angular increments. Theoretically, one may find that Euler angular increments are equal to input angular increments. The flow chart of this simulation is shown in Figure 5.

Assume the x coordinate axis rotates. If the input angular increments are $\Delta\theta_{x1} = 0.01°$, $\Delta\theta_{x2}$ $0.01°$, and $\Delta\theta_{y1} = \Delta\theta_{y1} = \Delta\theta_{z1} = \Delta\theta_{z2} = 0$, the results of Euler angular increments for this simulation are shown in Table 6. The generated errors are also shown in this table. The other two sets of input angular increments ($\Delta\theta_{x1} = 0.02°$, $\Delta\theta_{x2} = 0.02°$ and $\Delta\theta_{x1} = 0.04°$, $\Delta\theta_{x2} = 0.04°$) are also given for this simulation; the results are shown in Table 7 and Table 8. Figure 6

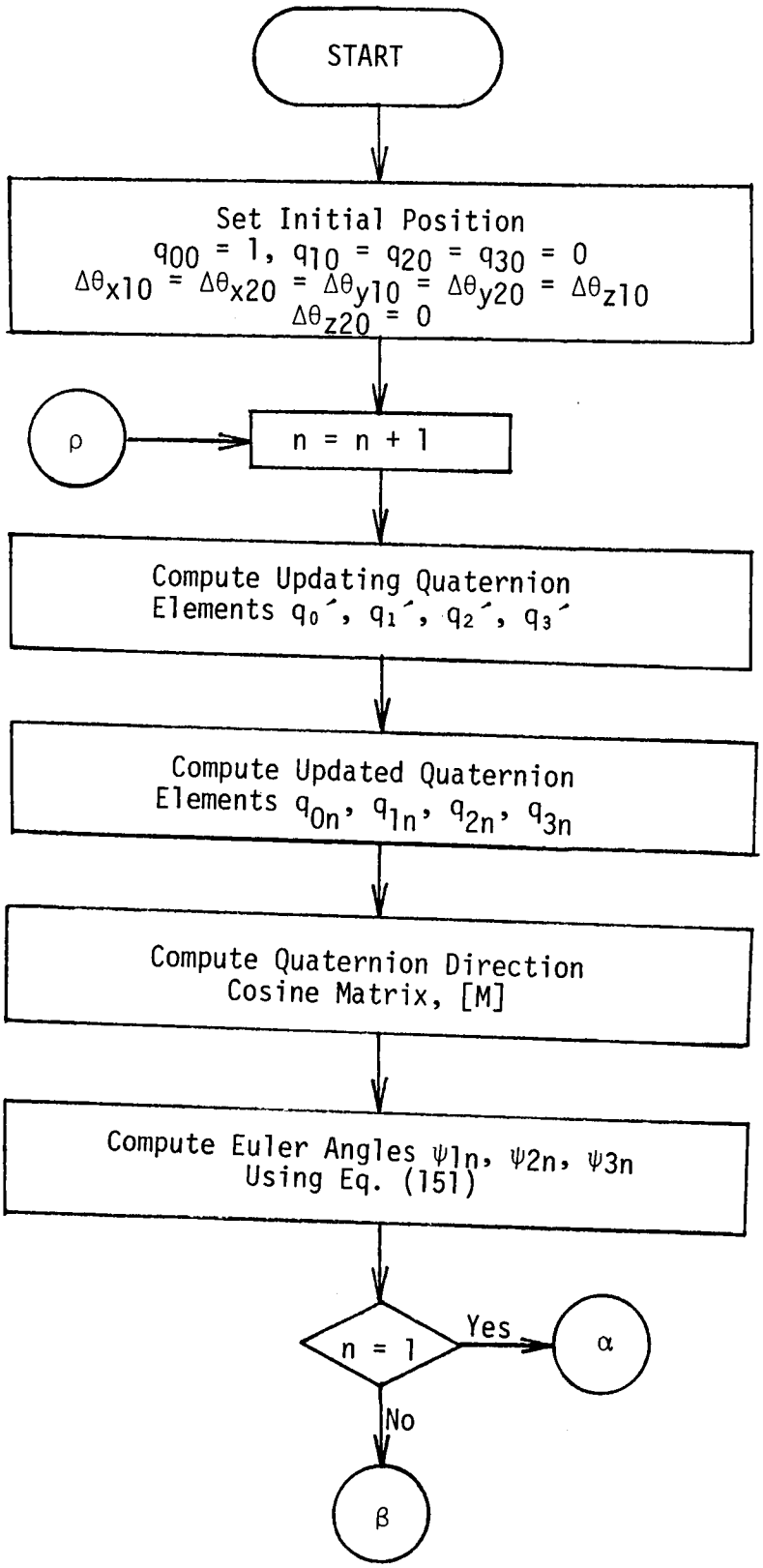


Figure 5. Flow Chart of Comparison of Computed Input and Euler Angular Increments.

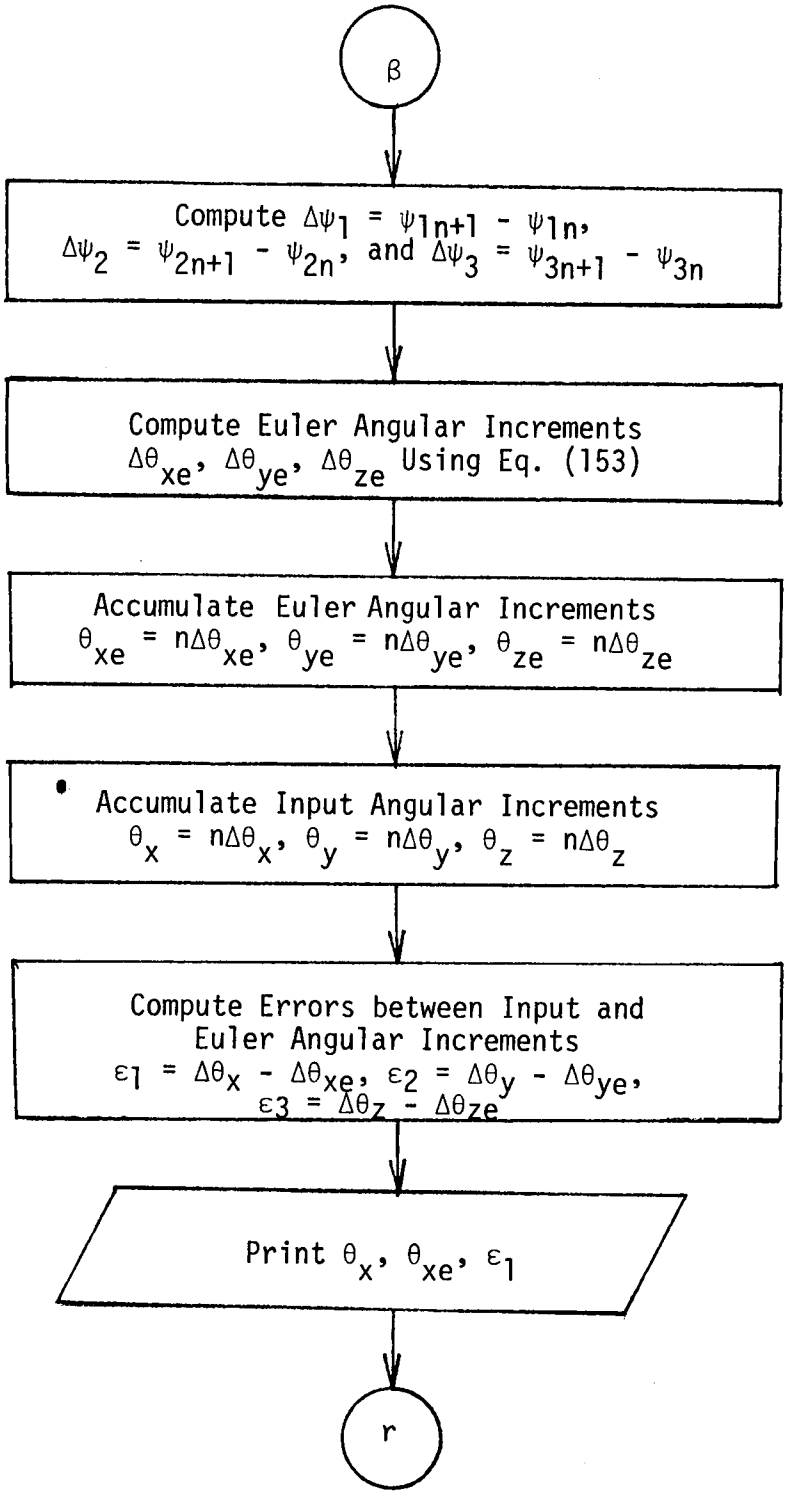


Figure 5 (continued)

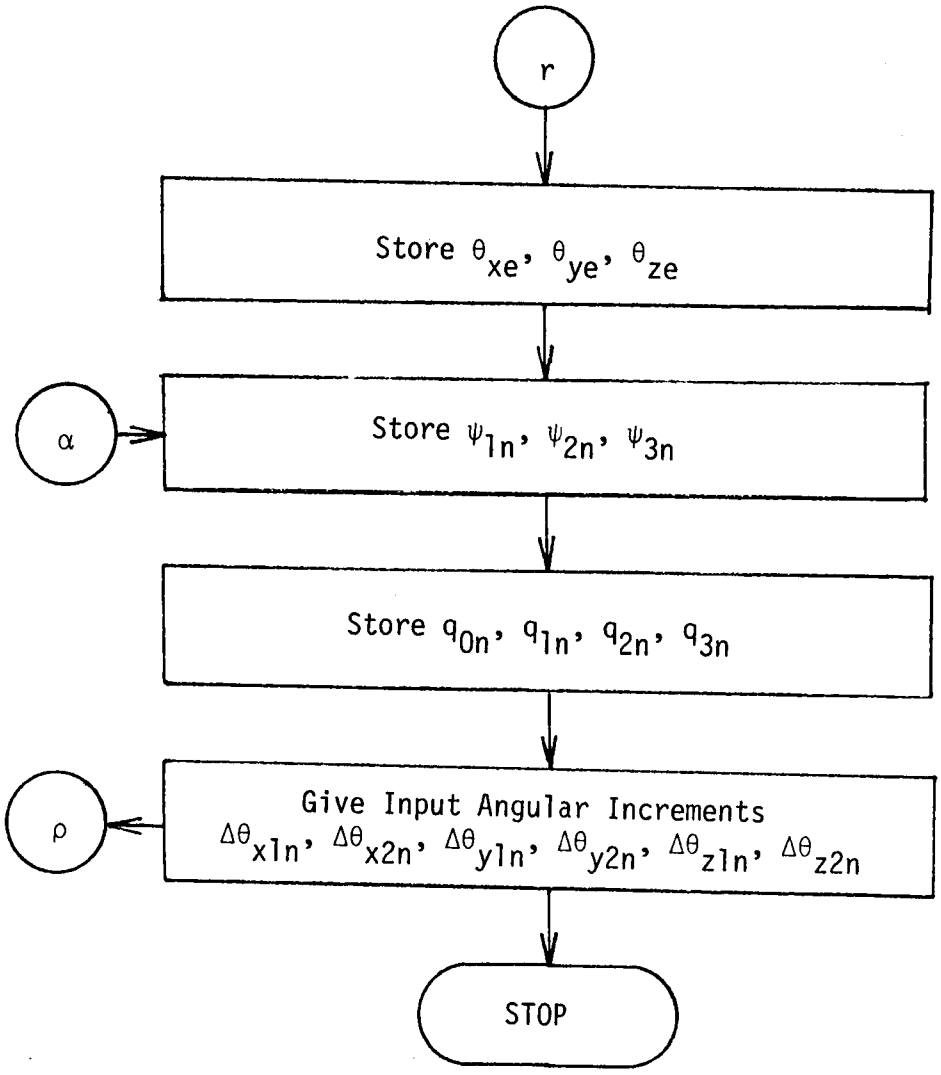


Figure 5 (Continued)

Table 6

Result of Input and Euler Angular Increments Using
$\Delta\theta_{x1} = \Delta\theta_{x2} = 0.01°$, $\Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1}$
$\Delta\theta_{z2} = 0°$, and n Equals to 1 to 40

Update Angle $\theta_x = n\Delta\theta_x$	Euler Update Angle $\theta_{xe} = n\Delta\theta_{xe}$	Error = $\theta_x - \theta_{xe}$
0.8000D-01	0.79999999999999000D-01	0.41633363423443700D-16
0.1600D 00	0.15999999999999800D 00	0.16653345369377350OD-15
0.2400D 00	0.23999999999999500D 00	0.38857805861880480OD-15
0.3200D 00	0.31999999999999400D 00	0.55511151231257830OD-15
0.4000D 00	0.39999999999999000D 00	0.88817841970012520OD-15
0.4800D 00	0.47999999999998500D 00	0.13600232051658160OD-14
0.5600D 00	0.55999999999997700D 00	0.21926904736346840OD-14
0.6400D 00	0.63999999999996800D 00	0.30669911055269950OD-14
0.7200D 00	0.71999999999995900D 00	0.39551695252271200OD-14
0.8000D 00	0.79999999999995200D 00	0.46213033400022140OD-14

Table 7

Result of Input and Euler Angular Increments Using
$\Delta\theta_{x1} = \Delta\theta_{x2} = 0.02°$, $\Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1}$
$\Delta\theta_{z2} = 0°$, and n Equals to 1 to 20

Update Angle $\theta_x = n\Delta\theta_x$	Euler Update Angle $\theta_{xe} = n\Delta\theta_{xe}$	Error = $\theta_x - \theta_{xe}$
0.8000D-01	0.79999999999998000D-01	0.18041124150158790OD-15
0.1600D 00	0.15999999999999200D 00	0.77715611723760960OD-15
0.2400D 00	0.23999999999998100D 00	0.18041124150158790OD-14
0.3200D 00	0.31999999999996800D 00	0.31641356201816960OD-14
0.4000D 00	0.39999999999994900D 00	0.49682480351975760OD-14
0.4800D 00	0.47999999999992700D 00	0.72164496600635170OD-14
0.5600D 00	0.55999999999989800D 00	0.10075273948473290OD-13
0.6400D 00	0.63999999999986600D 00	0.13308798507694060OD-13
0.7200D 00	0.71999999999983100D 00	0.16708856520608610OD-13
0.8000D 00	0.79999999999979100D 00	0.20691781621451350OD-13

Table 8

Result of Input and Euler Angular Increments Using $\Delta\theta_{x1} = \Delta\theta_{x2} = 0.04°$, $\Delta\theta_{y1} = \Delta\theta_{y2} = \Delta\theta_{z1}$ $\Delta\theta_{z2} = 0°$, and n Equals to 1 to 10

Update Angle $\theta_x = n\Delta\theta_x$	Euler Update Angle $\theta_{xe} = n\Delta\theta_{xe}$	Error = $\theta_x - \theta_{xe}$
0.8000D-01	0.79999999999990000D-01	0.95756735873919750OD-15
0.1600D 00	0.15999999999994900D 00	0.50792703376600910OD-14
0.2400D 00	0.23999999999987500D 00	0.12434497875801750OD-13
0.3200D 00	0.31999999999977100D 00	0.22828960943854780OD-13
0.4000D 00	0.39999999999963400D 00	0.36554093085783270OD-13
0.4800D 00	0.47999999999946500D 00	0.53457238635701280OD-13
0.5600D 00	0.55999999999926400D 00	0.73580030957032240OD-13
0.6400D 00	0.63999999999903100D 00	0.96797569959505830OD-13
0.7200D 00	0.71999999999876800D 00	0.12309597785531420OD-12
0.8000D 00	0.79999999999847300D 00	0.15257239915911210OD-12

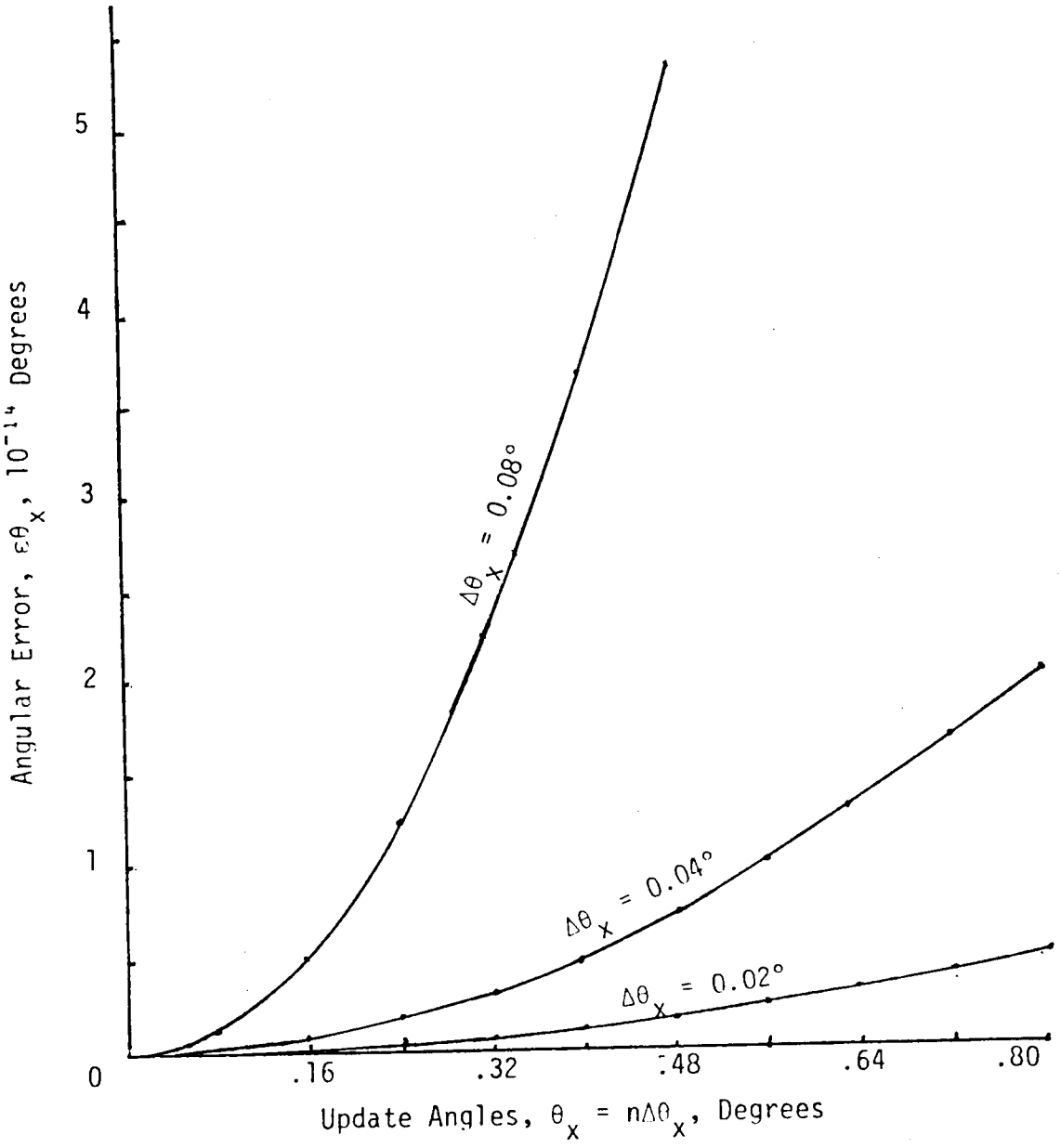


Figure 6. Comparison of error for different angular increments.

shows the comparison of angular errors (ε) for these three input angular increments. One may find from Figure 6 that the curve of $\Delta\theta_x = 0.08°$ has the highest angular error, and $\Delta\theta_x = 0.02°$ has the lowest angular error. This result shows that lower angular increments have less error than higher angular increments. This is only a typical example. Many different angular increments have been tried. The results follow this typical example.

Therefore, in a single axis rotating coordinate system, one may have a result that when the larger angular increments are used, the higher angular errors will be generated.

IV. COMPARISON OF QUATERNION ACCURACY

The above method showed that the error was decreased when small angular increments were used. This fact can be used to compare the accuracy of update quaternions. If it is assumed that a set of angular increments are used to compute the update quaternion elements, one may test the accuracy of these elements by using smaller angular increments as references. Using this point of view, the author compared the accuracy of update quaternion elements for Algorithm I and Algorithm II. For convenience, the following notations were used in this section

1. $\Delta\theta_i$: Input angular increment.
2. Q_{s1}: A set of update quaternion elements obtained by Algorithm I.
3. Q_{s2}: A set of update quaternion elements obtained by Algorithm II.

4. Q_{f1}: A set of reference of update quaternion elements obtained by Algorithm I.

5. Q_{f2}: A set of reference of update quaternion elements obtained by Algorithm II.

Figure 7 shows the block diagram of this simulation process.

Two sets of update quaternion elements, Q_{s1} and Q_{s2}, can be obtained by a set of input angular increments $\Delta\theta_i$. Using a set of smaller $\Delta\theta_i$ to obtain two sets of Q_{f1} and Q_{f2} as references, one may compare the accuracy of Q_{s1} and Q_{s2}. The flow chart of this simulation is the same as that shown in Figure 2.

One may see that the references are established only in the condition that Q_{f1} and Q_{f2} must have the same valid number of digits, and the value of Q_{s1} and Q_{s2} to be compared must have smaller valid numbers of digits than the references. For example, Table 9 and Table 10 show the results of Q_{f1} and Q_{f2} using the following angular increments and rotating n = 10000 times,

Set 1:

$$\Delta\theta_{x1} = 0.000001° \qquad \Delta\theta_{x2} = 0.000002°$$
$$\Delta\theta_{y1} = 0.0000015° \qquad \Delta\theta_{y2} = 0.0000015°$$
$$\Delta\theta_{z1} = 0.0000015° \qquad \Delta\theta_{z2} = 0.0000015°$$

The results of Q_{f1} and Q_{f2} show that the valid number of digits are up to 14 digits for Q_0, 13 digits for Q_1, 6 digits for Q_2, and 6 digits for Q_3. Table 11 and Table 12 show the results of this simulation using the following angular increments (Set 2) and rotating n = 10 times.

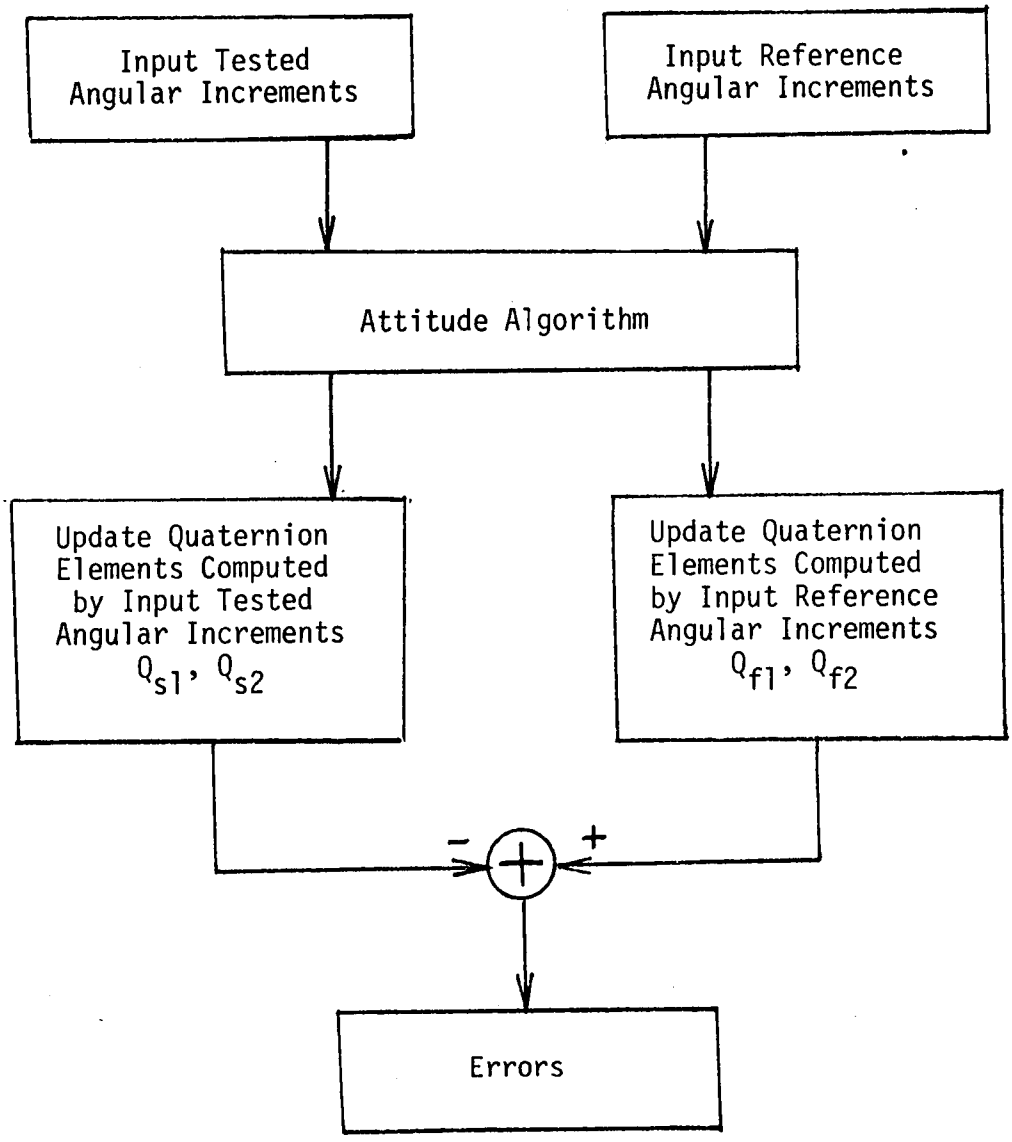


Figure 7. Block Diagram of Comparison of Quaternion Accuracy.

Table 9

Result of Update Quaternion Elements of Algorithm I for Set 1
as Input Angular Increments, and n Equals to 1 to 10000
(Time Used: 24.09 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.3000D-02	Q0= 0.99999662500188210OD 00	Q1= 0.14999983125004730OD-02
	Q2= 0.14999979375008550OD-02	Q3= 0.14999986875000310OD-02
0.6000D-02	Q0= 0.99998650003033900OD 00	Q1= 0.29999865000179960OD-02
	Q2= 0.29999857500212950OD-02	Q3= 0.29999872500145880OD-02
0.9000D-02	Q0= 0.99996962515371700D 00	Q1= 0.44999544376373940OD-02
	Q2= 0.44999533126488300OD-02	Q3= 0.44999555626262520OD-02
0.1200D-01	Q0= 0.99994600048592200OD 00	Q1= 0.59998920005804460OD-02
	Q2= 0.59998905006078520OD-02	Q3= 0.59998935005544390OD-02
0.1500D-01	Q0= 0.99991562618642100OD 00	Q1= 0.74997890642752460OD-02
	Q2= 0.74997871893287460OD-02	Q3= 0.74997909392242520OD-02
0.1800D-01	Q0= 0.99987850246023960OD 00	Q1= 0.89996355044223200OD-02
	Q2= 0.89996332545145840OD-02	Q3= 0.89996377543336760OD-02
0.2100D-01	Q0= 0.99983462955796280OD 00	Q1= 0.10499421197063910OD-01
	Q2= 0.10499418572210110OD-01	Q3= 0.10499423821922430OD-01
0.2400D-01	Q0= 0.99978400777573280OD 00	Q1= 0.11999136018652250OD-01
	Q2= 0.11999133018870100OD-01	Q3= 0.11999139018440230OD-01
0.2700D-01	Q0= 0.99972663745524620OD 00	Q1= 0.13498769846118000OD-01
	Q2= 0.13498766471427740OD-01	Q3= 0.13498773220815180OD-01
0.3000D-01	Q0= 0.99966251898375270OD 00	Q1= 0.14998312556938500OD-01
	Q2= 0.14998308807362940OD-01	Q3= 0.14998316306522120OD-01

Table 10

Result of Update Quaternion Elements of Algorithm II for Set 1
as Input Angular Increments, and n Equals to 1 to 10000
(Time Used: 18.45 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.100000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.3000D-02	Q0= 0.999996625001881600D 00	Q1= 0.149999831250047300D-02
	Q2= 0.149999881249992900D-02	Q3= 0.149999781250107800D-02
0.6000D-02	Q0= 0.999986500030338300D 00	Q1= 0.299998650001799500D-02
	Q2= 0.299998750001353800D-02	Q3= 0.299998550002258900D-02
0.9000D-02	Q0= 0.999969625153716500D 00	Q1= 0.449995443763739100D-02
	Q2= 0.449995593762240400D-02	Q3= 0.449995293765301500D-02
0.1200D-01	Q0= 0.999946000485921100D 00	Q1= 0.599989200058044000D-02
	Q2= 0.599989400054503500D-02	Q3= 0.599989000061769500D-02
0.1500D-01	Q0= 0.999915626186420100D 00	Q1= 0.749978906427523400D-02
	Q2= 0.749979156420590200D-02	Q3= 0.749978656434762100D-02
0.1800D-01	Q0= 0.999878502460238400D 00	Q1= 0.899963550442230500D-02
	Q2= 0.899963850430217300D-02	Q3= 0.899963250454668200D-02
0.2100D-01	Q0= 0.999834629557961800D 00	Q1= 0.104994211970638800D-01
	Q2= 0.104994246968726800D-01	Q3= 0.104994176972605000D-01
0.2400D-01	Q0= 0.999784007775731900D 00	Q1= 0.119991360186522100D-01
	Q2= 0.119991400183663300D-01	Q3= 0.119991320189447100D-01
0.2700D-01	Q0= 0.999726637455245500D 00	Q1= 0.134987698461179500D-01
	Q2= 0.134987743457104000D-01	Q3= 0.134987653465333300D-01
0.3000D-01	Q0= 0.999662518983752300D 00	Q1= 0.149983125569384400D-01
	Q2= 0.149983175563788900D-01	Q3= 0.149983075575070700D-01

Table 11

Result of Update Quaternion Elements of Algorithm I for Set 2
as Input Angular Increments, and n Equals to 1 to 10
(Time Used: 3.26 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.3000D-02	Q0= 0.99999662500000000D 00	Q1= 0.14999983125000000D-02
	Q2= 0.14996232500000000D-02	Q3= 0.15003732500000000D-02
0.6000D-02	Q0= 0.99998650002667190OD 00	Q1= 0.29999865000113900OD-02
	Q2= 0.29992363775430620OD-02	Q3= 0.30007363724805620OD-02
0.9000D-02	Q0= 0.99996962514835910OD 00	Q1= 0.44999544376197400OD-02
	Q2= 0.44988292602464090OD-02	Q3= 0.45010792399964460OD-02
0.1200D-01	Q0= 0.99994600047896720OD 00	Q1= 0.59998920005473090OD-02
	Q2= 0.59983917758639190OD-02	Q3= 0.60013917252391360OD-02
0.1500D-01	Q0= 0.99991562617796250OD 00	Q1= 0.74997890642213800OD-02
	Q2= 0.74979138023544400D-02	Q3= 0.75016637011051920OD-02
0.1800D-01	Q0= 0.99987850245037160OD 00	Q1= 0.89996355043426110OD-02
	Q2= 0.89973852179501410OD-02	Q3= 0.90018850407646140OD-02
0.2100D-01	Q0= 0.99983462954677940OD 00	Q1= 0.10499421196953370OD-01
	Q2= 0.10496795901224790OD-01	Q3= 0.10502045617729200OD-01
0.2400D-01	Q0= 0.99978400776332790OD 00	Q1= 0.11999136018506060OD-01
	Q2= 0.11996135731162110OD-01	Q3= 0.12002135305920880OD-01
0.2700D-01	Q0= 0.99972663744171390OD 00	Q1= 0.13498769845931470OD-01
	Q2= 0.13495394587224080OD-01	Q3= 0.13502143979740110OD-01
0.3000D-01	Q0= 0.99966251869187000D 00	Q1= 0.14998312556707110OD-01
	Q2= 0.14994562349419260OD-01	Q3= 0.15002061514134180OD-01

Table 12

Result of Update Quaternion Elements of Algorithm II for Set 2
as Input Angular Increments, and n Equals to 1 to 10
(Time Used: 3.01 sec in IBM 360)

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.3000D-02	Q0= 0.99999662500000000D 00	Q1= 0.14999983125000000D-02
	Q2= 0.15004983125000000D-02	Q3= 0.14994983125000000D-02
0.6000D-02	Q0= 0.99998650002607810OD 00	Q1= 0.29999865000113900OD-02
	Q2= 0.30009864966363900OD-02	Q3= 0.29989865033863900OD-02
0.9000D-02	Q0= 0.99996962514657900D 00	Q1= 0.44999544376188490OD-02
	Q2= 0.45014544241188740OD-02	Q3= 0.44984544511188260OD-02
0.1200D-01	Q0= 0.99994600047504700D 00	Q1= 0.59998920005437450OD-02
	Q2= 0.60018919667938890OD-02	Q3= 0.59978920342936030OD-02
0.1500D-01	Q0= 0.99991562617202510OD 00	Q1= 0.74997890642124730OD-02
	Q2= 0.75022889967129690OD-02	Q3= 0.74972891317119800OD-02
0.1800D-01	Q0= 0.99987850244146560OD 00	Q1= 0.89996355043247980OD-02
	Q2= 0.90026353862011080OD-02	Q3= 0.89966356224484910OD-02
0.2100D-01	Q0= 0.99983462953431100D 00	Q1= 0.10499421196922190OD-01
	Q2= 0.10502921007925120OD-01	Q3= 0.10495921385919270OD-01
0.2400D-01	Q0= 0.99978400774670380OD 00	Q1= 0.11999136018456180OD-01
	Q2= 0.12003135734962020OD-01	Q3= 0.11995136301950350OD-01
0.2700D-01	Q0= 0.99972663742034050OD 00	Q1= 0.13498769845856660OD-01
	Q2= 0.13503269440867330OD-01	Q3= 0.13494270250846000OD-01
0.3000D-01	Q0= 0.99966251894247070OD 00	Q1= 0.14998312556600240OD-01
	Q2= 0.15003311999734800D-01	Q3= 0.14993313113457000OD-01

Set 2:

$\Delta\theta_{x1} = 0.001°$ $\Delta\theta_{x2} = 0.002°$

$\Delta\theta_{y1} = 0.0015°$ $\Delta\theta_{y2} = 0.0015°$

$\Delta\theta_{z1} = 0.0015°$ $\Delta\theta_{z2} = 0.0015°$.

The results of Q_{s1} and Q_{s2} show that the valid number of digits are 11 digits for Q_0, 11 digits for Q_1, 3 digits for Q_2, and 3 digits for Q_3.

Since the precision of Q_{f1} and Q_{f2} computed using set 1 as input angular increments is higher than Q_{s1} and Q_{s2} computed using set 2, either Q_{f1} or Q_{f2} can be taken as references for Q_{s1} and Q_{s2}. The comparisons of the elements of Q_{s1} and Q_{s2} are shown in Figure 8. They show that the curves of update quaternion elements of Algorithm I are closer to the reference than Algorithm II.

One may also obtain Q_{f1}, Q_{f2}, Q_{s1}, and Q_{s2} using the following sets of input.

Set 3:

$\Delta\theta_{x1} = 0.0000002°$ $\Delta\theta_{x2} = 0.0000008°$

$\Delta\theta_{y1} = 0.0000003°$ $\Delta\theta_{y2} = 0.0000007°$

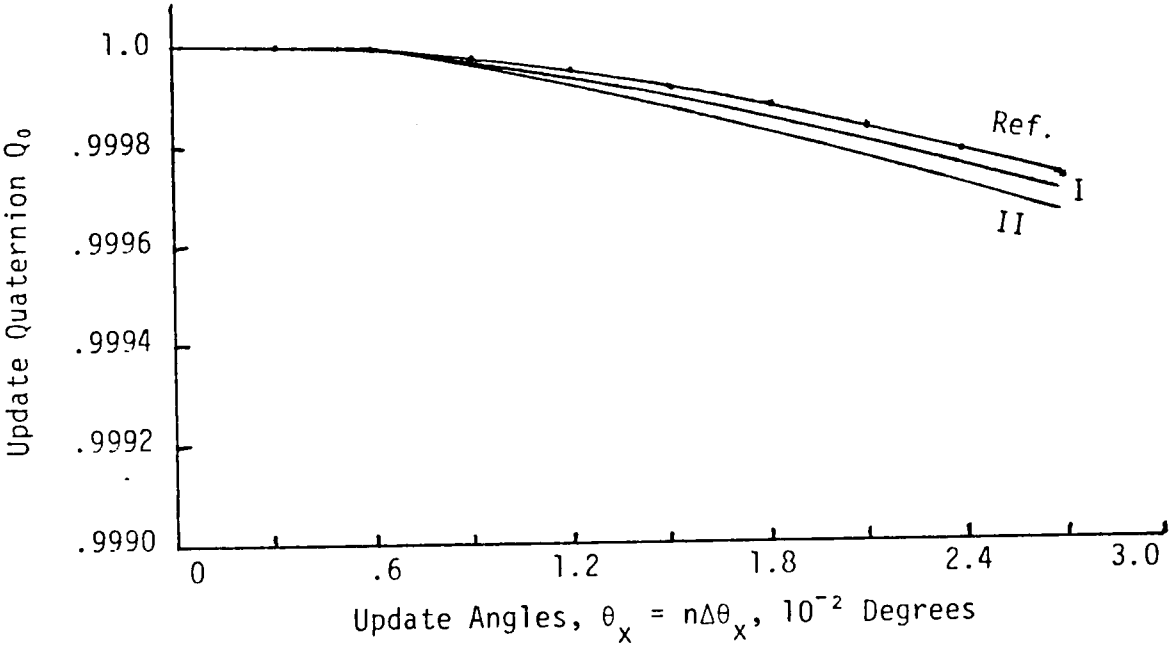
$\Delta\theta_{z1} = 0.0000006°$ $\Delta\theta_{z2} = 0.0000004°$

Set 4:

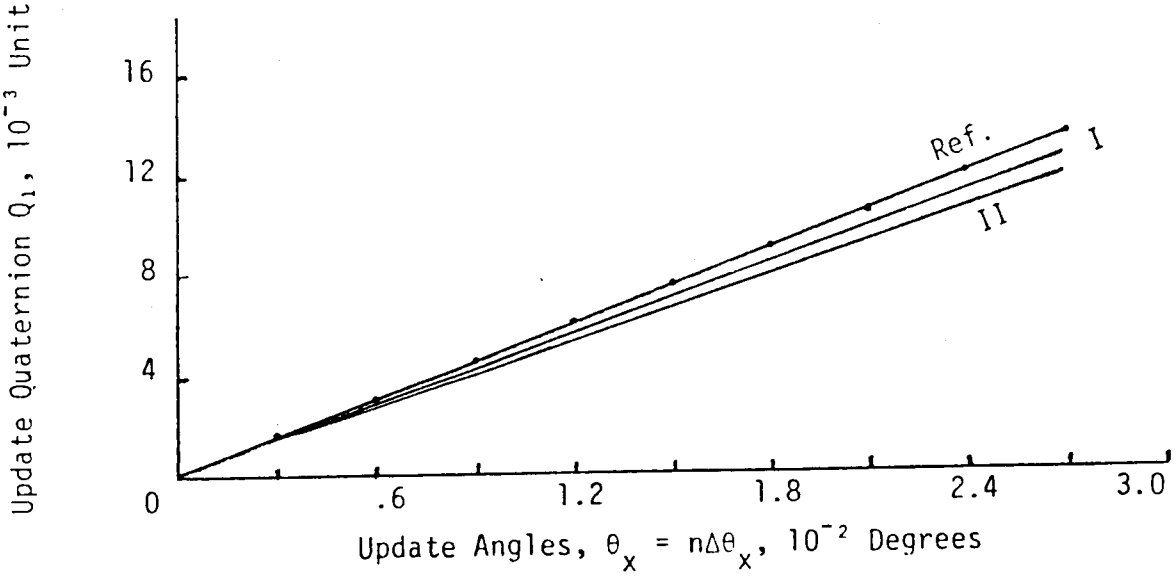
$\Delta\theta_{x1} = 0.0002°$ $\Delta\theta_{x2} = 0.0008°$

$\Delta\theta_{y1} = 0.0003°$ $\Delta\theta_{y2} = 0.0007°$

$\Delta\theta_{z1} = 0.0006°$ $\Delta\theta_{z2} = 0.0004°$

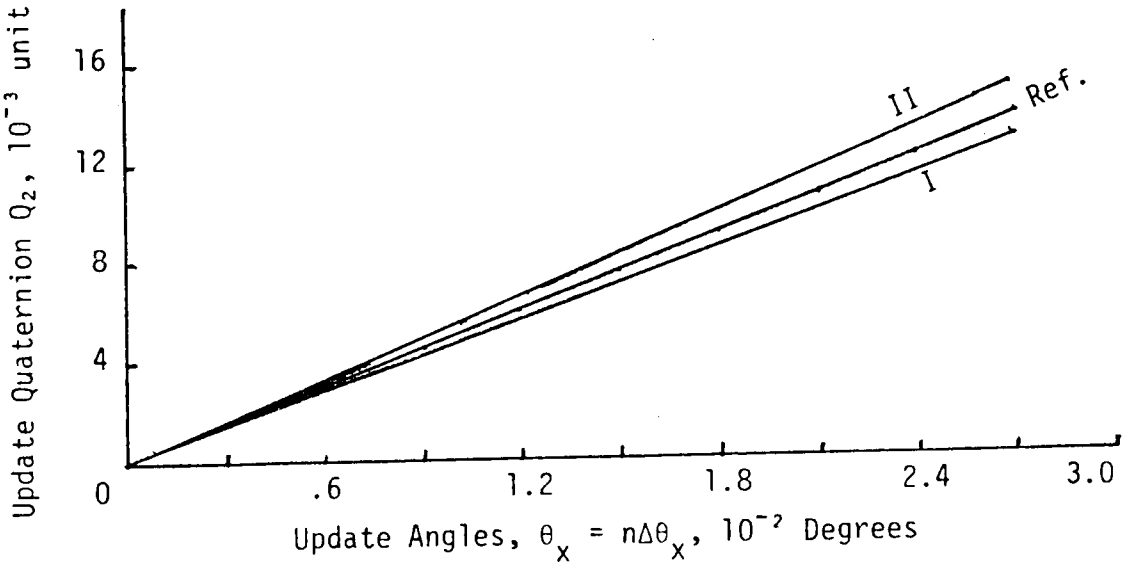


(a) Comparison of Q_0 for Set 1 (Ref.) and Set 2.

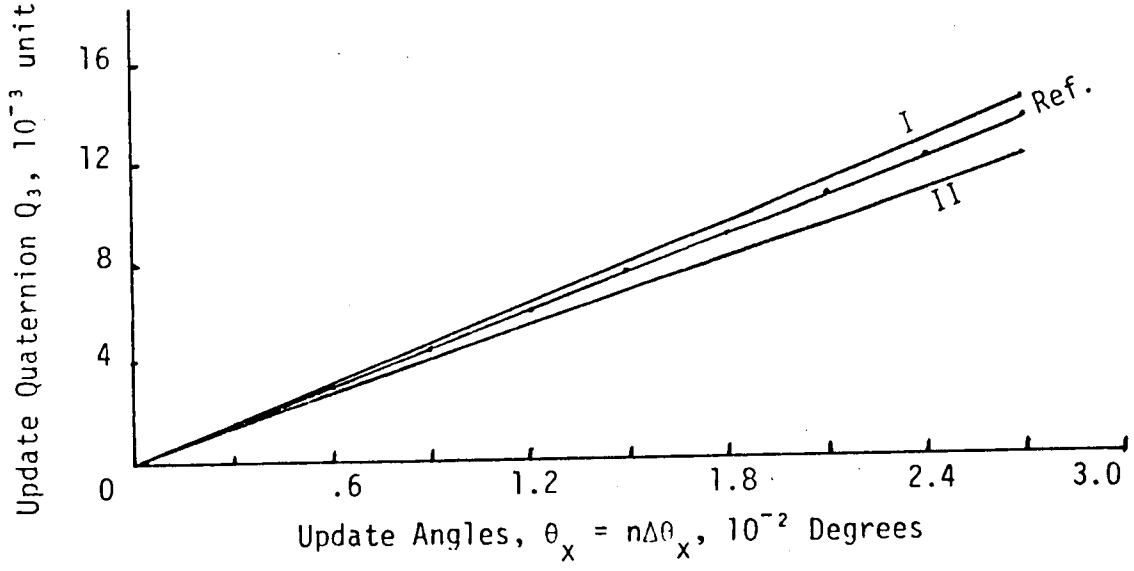


(b) Comparison of Q_1 for Set 1 (Ref.) and Set 2.

Figure 8. Comparison of update quaternion elements for Set 1 as references and Set 2 as input incremental angles of Algorithm I and Algorithm II. The scale of Curve I and II have been enlarged.



(c) Comparison of Q_2 for Set 1 (Ref.) and Set 2.



(d) Comparison of Q_3 for Set 1 (Ref.) and Set 2.

Figure 8 (continued)

Set 5:

$\Delta\theta_{x1} = 0.00001°$ $\Delta\theta_{x2} = 0°$

$\Delta\theta_{y1} = 0.00002°$ $\Delta\theta_{y2} = 0.00008°$

$\Delta\theta_{z1} = 0.00003°$ $\Delta\theta_{z2} = 0.00007°$

Set 6:

$\Delta\theta_{x1} = 0.01°$ $\Delta\theta_{x2} = 0°$

$\Delta\theta_{y1} = 0.02°$ $\Delta\theta_{y2} = 0.08°$

$\Delta\theta_{z1} = 0.03°$ $\Delta\theta_{z2} = 0.07°$

The results of update quaternion using these input angular increments are shown in Tables 13 to 20. Figures 9 and 10 show the comparison results. One may also see that the precision of the elements of Q_{s1} is higher than the elements of Q_{s2}.

Therefore, in this simulation, one may conclude that Algorithm I yields higher accuracy than Algorithm II.

Table 13

Result of Update Quaternion Elements of Algorithm I for Set 3
as Input Angular Increments, and n Equals to 1 to 10000

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-02	Q0= 0.99999962500000138000D 00	Q1= 0.50000001249993380000D-03
	Q2= 0.49999983749996720000D-03	Q3= 0.49999996249995420000D-03
0.2000D-02	Q0= 0.99999850000035520000D 00	Q1= 0.99999964999983370000D-03
	Q2= 0.99999930000004210000D-03	Q3= 0.99999954999993480000D-03
0.3000D-02	Q0= 0.99999662500186770000D 00	Q1= 0.14999985375000410000D-02
	Q2= 0.14999980125006880000D-02	Q3= 0.14999983875002980000D-02
0.4000D-02	Q0= 0.99999400000595760000D 00	Q1= 0.19999963000014140000D-02
	Q2= 0.19999956000028930000D-02	Q3= 0.19999961000019400000D-02
0.5000D-02	Q0= 0.99999062501459310000D 00	Q1= 0.24999925625056540000D-02
	Q2= 0.24999916875084910000D-02	Q3= 0.24999923125066000000D-02
0.6000D-02	Q0= 0.99998650003030530000D 00	Q1= 0.29999869500155860000D-02
	Q2= 0.29999859000204380000D-02	Q3= 0.29999866500171380000D-02
0.7000D-02	Q0= 0.99998162505618790000D 00	Q1= 0.34999790875354430000D-02
	Q2= 0.34999778625430960000D-02	Q3= 0.34999787375378270000D-02
0.8000D-02	Q0= 0.99997600009589570000D 00	Q1= 0.39999686000708430000D-02
	Q2= 0.39999672000822990000D-02	Q3= 0.39999682000744660000D-02
0.9000D-02	Q0= 0.99996962515364880000D 00	Q1= 0.44999551126287020000D-02
	Q2= 0.44999535376453450000D-02	Q3= 0.44999546626344760000D-02
0.1000D-01	Q0= 0.99996250023422840000D 00	Q1= 0.49999382502204370000D-02
	Q2= 0.49999365002434490000D-02	Q3= 0.49999377502287080000D-02

Table 14

Result of Update Quaternion Elements of Algorithm II for Set 3
as Input Angular Increments, and n Equals to 1 to 10000

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.100000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-02	Q0= 0.999999625000013500D 00	Q1= 0.499999837499957100D-03
	Q2= 0.500000708332754 00D-03	Q3= 0.499999904166643500D-03
0.2000D-02	Q0= 0.999998500000354600D 00	Q1= 0.999999300000011400D-03
	Q2= 0.999999766666482200D-03	Q3= 0.999999433333356700D-03
0.3000D-02	Q0= 0.999996625001867100D 00	Q1= 0.149999801250063500D-02
	Q2= 0.149999871249991000D-02	Q3= 0.149999821250054000D-02
0.4000D-02	Q0= 0.999994000005956900D 00	Q1= 0.199999560000281800D-02
	Q2= 0.199999653333437000D-02	Q3= 0.199999586666913300D-02
0.5000D-02	Q0= 0.999990625014592400D 00	Q1= 0.249999168750839300D-02
	Q2= 0.249999285417152400D-02	Q3= 0.249999202084091700D-02
0.6000D-02	Q0= 0.999986500030304700D 00	Q1= 0.299998590002031600D-02
	Q2= 0.299998730001415100D-02	Q3= 0.299998630001880100D-02
0.7000D-02	Q0= 0.999981625056187100D 00	Q1= 0.349997786254295300D-02
	Q2= 0.349997949586644100D-02	Q3= 0.349997832920709700D-02
0.8000D-02	Q0= 0.999976000095895600D 00	Q1= 0.399996720008205800D-02
	Q2= 0.399996906673403700D-02	Q3= 0.399996773341165300D-02
0.9000D-02	Q0= 0.999969625153648800D 00	Q1= 0.449995353764467400D-02
	Q2= 0.449995563762403600D-02	Q3= 0.449995413763992700D-02
0.1000D-01	Q0= 0.999962500234228300D 00	Q1= 0.499993650024232200D-02
	Q2= 0.499993883354750300D-02	Q3= 0.499993716690280700D-02

Table 15

Result of Update Quaternion Elements of Algorithm I for Set 4
as Input Angular Increments, and n Equals to 1 to 10

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-02	Q0= 0.99999625000000000D 00	Q1= 0.50007493583333320D-03
	Q2= 0.49989993249999990D-03	Q3= 0.50002492249999980D-03
0.2000D-02	Q0= 0.99999850000033500D 00	Q1= 0.10001494966104640D-02
	Q2= 0.99979949007505040D-03	Q3= 0.10000494699813080D-02
0.3000D-02	Q0= 0.99999662500184430D 00	Q1= 0.15002233072754990D-02
	Q2= 0.14996982978005090D-02	Q3= 0.15000732674255390D-02
0.4000D-02	Q0= 0.99999400000593870D 00	Q1= 0.20002959277310600D-02
	Q2= 0.19995959807522970D-02	Q3= 0.20000959398148730D-02
0.5000D-02	Q0= 0.99999062501458530D 00	Q1= 0.25003671780487960D-02
	Q2= 0.24994921640071780D-02	Q3= 0.25001171121323290D-02
0.6000D-02	Q0= 0.99998650003031550D 00	Q1= 0.30004364880492080D-02
	Q2= 0.29993864726430400D-02	Q3= 0.30001364093620560D-02
0.7000D-02	Q0= 0.99998162505622900D 00	Q1= 0.35005035477223850D-02
	Q2= 0.34992785317391700D-02	Q3= 0.35001534564896070D-02
0.8000D-02	Q0= 0.99997600009596380D 00	Q1= 0.40005679820180560D-02
	Q2= 0.39991679663765740D-02	Q3= 0.40001678785022220D-02
0.9000D-02	Q0= 0.99996962515375700D 00	Q1= 0.45006294158879230D-02
	Q2= 0.44990544016381780D-02	Q3= 0.45001793003891110D-02
0.1000D-01	Q0= 0.99996250023438360D 00	Q1= 0.50006874742859370D-02
	Q2= 0.49989374626091900D-02	Q3= 0.50001873471417350D-02

Table 16

Result of Update Quaternion Elements of Algorithm II for Set 4
as Input Angular Increments, and n Equals to 1 to 10

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-02	Q0= 0.99999962500000000D 00	Q1= 0.49989993749999900D-03
	Q2= 0.50013327083333200D-03	Q3= 0.49996660416666600D-03
0.2000D-02	Q0= 0.99999850000029920D 00	Q1= 0.99979950007504660D-03
	Q2= 0.10002661665671300D-02	Q3= 0.99993283335837980D-03
0.3000D-02	Q0= 0.99999662500174140D 00	Q1= 0.14996983128004770D-02
	Q2= 0.15003983121004770D-02	Q3= 0.14998983126004770D-02
0.4000D-02	Q0= 0.99999400000573290D 00	Q1= 0.19995960007521910D-02
	Q2= 0.20005293323355250D-02	Q3= 0.19998626669188570D-02
0.5000D-02	Q0= 0.99999062501424230D 00	Q1= 0.24994921890069310D-02
	Q2= 0.25006588521736000D-02	Q3= 0.24998255213402650D-02
0.6000D-02	Q0= 0.99998650002980090D 00	Q1= 0.29993865026425660D-02
	Q2= 0.30007864965175730D-02	Q3= 0.29997865008925680D-02
0.7000D-02	Q0= 0.99998162505550250D 00	Q1= 0.34992785667383680D-02
	Q2= 0.35009118902717180D-02	Q3= 0.34997452306050400D-02
0.8000D-02	Q0= 0.99997600009500340D 00	Q1= 0.39991680063752990D-02
	Q2= 0.40010346583419980D-02	Q3= 0.39997013355086420D-02
0.9000D-02	Q0= 0.99996962515252210D 00	Q1= 0.44990544466362880D-02
	Q2= 0.45011544256363480D-02	Q3= 0.44996544406363060D-02
0.1000D-01	Q0= 0.99996250023283990D 00	Q1= 0.49989375126065120D-02
	Q2= 0.50012708170649500D-02	Q3= 0.49996041710232110D-02

Table 17

Result of Update Quaternion Elements of Algorithm I for Set 5
as Input Angular Increments, and n Equals to 1 to 10000

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-01	Q0= 0.99748855193315900OD 00	Q1= 0.49960633394968330OD-02
	Q2= 0.49958310374640260OD-01	Q3= 0.49957935684502380OD-01
0.2000D-01	Q0= 0.98996682247547240OD 00	Q1= 0.99670319717610870OD-02
	Q2= 0.99665685345241400OD-01	Q3= 0.99664937846999900OD-01
0.3000D-01	Q0= 0.97747259249275270OD 00	Q1= 0.14887937237670760OD-01
	Q2= 0.14887244993026040OD 00	Q3= 0.14887133337852230OD 00
0.4000D-01	Q0= 0.96006861920447260OD 00	Q1= 0.19734061941191770OD-01
	Q2= 0.19733144366211920OD 00	Q3= 0.19732996366520830OD 00
0.5000D-01	Q0= 0.93784232096077210OD 00	Q1= 0.24481064501287730OD-01
	Q2= 0.24479926204856220OD 00	Q3= 0.24479742604034920OD 00
0.6000D-01	Q0= 0.91090533814916420OD 00	Q1= 0.29105101217153160OD-01
	Q2= 0.29103747916814470OD 00	Q3= 0.29103529637070810OD 00
0.7000D-01	Q0= 0.87939297243646550OD 00	Q1= 0.33582946032646210OD-01
	Q2= 0.33581384525887060OD 00	Q3= 0.33581132663617500OD 00
0.8000D-01	Q0= 0.84346350716257520OD 00	Q1= 0.37892107198356120OD-01
	Q2= 0.37890345328462490OD 00	Q3= 0.37890061148745050OD 00
0.9000D-01	Q0= 0.80329741229970130OD 00	Q1= 0.42010940245324480OD-01
	Q2= 0.42008986861985450OD 00	Q3= 0.42008671792225420OD 00
0.1000D 00	Q0= 0.75909643797043330OD 00	Q1= 0.45918756702964610OD-01
	Q2= 0.45916621617821520OD 00	Q3= 0.45916277240581480OD 00

Table 18

Result of Update Quaternion Elements of Algorithm II for Set 5
as Input Angular Increments, and n Equals to 1 to 10000

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-01	Q0= 0.99748855193301840OD 00	Q1= 0.49954804977457110OD-02
	Q2= 0.49957902381865250OD-01	Q3= 0.49958401963221290OD-01
0.2000D-01	Q0= 0.98996682247491190OD 00	Q1= 0.99658692158108080OD-02
	Q2= 0.99664871408982300OD-01	Q3= 0.99665868062353700OD-01
0.3000D-01	Q0= 0.97747259249149740OD 00	Q1= 0.14886200407920270OD-01
	Q2= 0.14887123413880630OD 00	Q3= 0.14887272285810860OD 00
0.4000D-01	Q0= 0.96006861920225420OD 00	Q1= 0.19731759761551660OD-01
	Q2= 0.19732983212222500D 00	Q3= 0.19733180542977300D 00
0.5000D-01	Q0= 0.93784232095733230OD 00	Q1= 0.24478208535360980OD-01
	Q2= 0.24479726285476540OD 00	Q3= 0.24479971083884560OD 00
0.6000D-01	Q0= 0.91090533814425750OD 00	Q1= 0.29101705810152520OD-01
	Q2= 0.29103510236212110D 00	Q3= 0.29103801272676180OD 00
0.7000D-01	Q0= 0.87939297242986060OD 00	Q1= 0.33579028239339500OD-01
	Q2= 0.33581110277898890OD 00	Q3= 0.33581446090572970OD 00
0.8000D-01	Q0= 0.84346350715405810OD 00	Q1= 0.37887686697402150OD-01
	Q2= 0.37890035890598210OD 00	Q3= 0.37890414792730140OD 00
0.9000D-01	Q0= 0.80329741228907790OD 00	Q1= 0.42006039240429100OD-01
	Q2= 0.42008643788508720OD 00	Q3= 0.42009063876912450OD 00
0.1000D 00	Q0= 0.75909643795753150OD 00	Q1= 0.45913399811353860OD-01
	Q2= 0.45916246631942940OD 00	Q3= 0.45916705796558090OD 00

Table 19

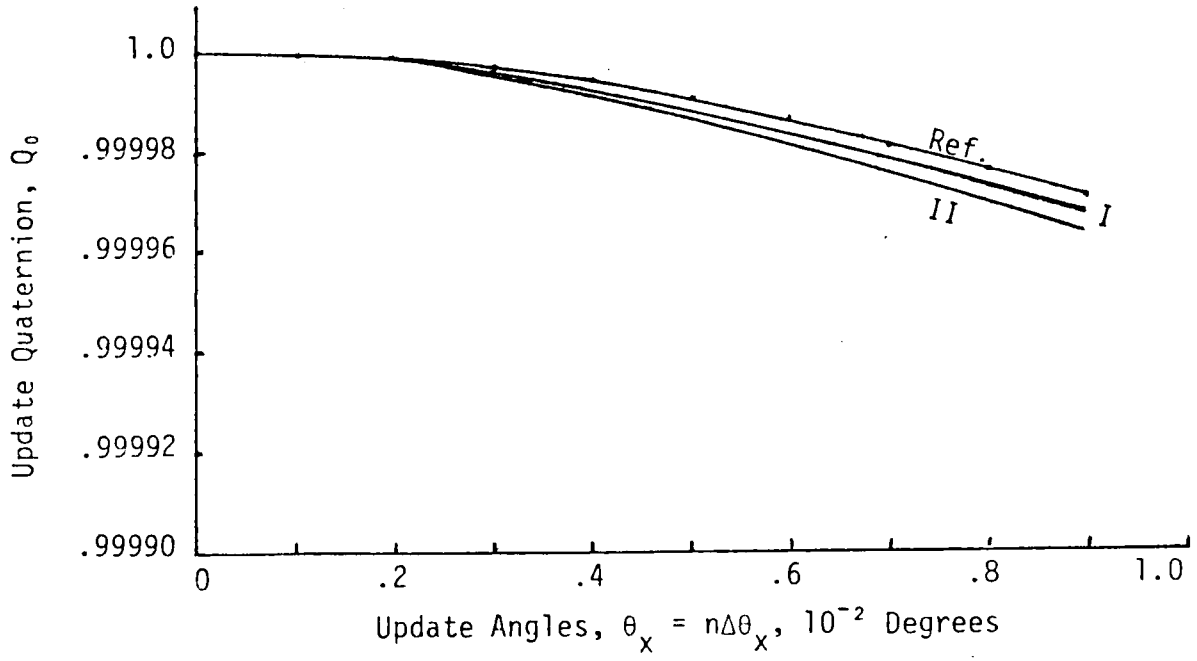
Result of Update Quaternion Elements of Algorithm I for Set 6
as Input Angular Increments, and n Equals to 1 to 10

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-01	Q0= 0.99748750000000000D 00	Q1= 0.52426458333333100D-02
	Q2= 0.50134458333333200D-01	Q3= 0.49755333333333200D-01
0.2000D-01	Q0= 0.98996477021342840OD 00	Q1= 0.10458947371354160OD-01
	Q2= 0.10001699101354160OD 00	Q3= 0.99260646116666700OD-01
0.3000D-01	Q0= 0.97746960696839910OD 00	Q1= 0.15622703943789850OD-01
	Q2= 0.14939704584735310OD 00	Q3= 0.14826728087352400D 00
0.4000D-01	Q0= 0.96006478232470830OD 00	Q1= 0.20707978862313930OD-01
	Q2= 0.19802659505231850OD 00	Q3= 0.19652908544824460OD 00
0.5000D-01	Q0= 0.93783772872530330OD 00	Q1= 0.25689229696207070OD-01
	Q2= 0.24566138105900830OD 00	Q3= 0.24380364938721590OD 00
0.6000D-01	Q0= 0.91090009978289650OD 00	Q1= 0.30541436567439850OD-01
	Q2= 0.29206214337330430OD 00	Q3= 0.28985352152343420OD 00
0.7000D-01	Q0= 0.87938720940809580OD 00	Q1= 0.35240227820773880OD-01
	Q2= 0.33699582033647360OD 00	Q3= 0.33444740264867120OD 00
0.8000D-01	Q0= 0.84345735209648900OD 00	Q1= 0.39762002437664170OD-01
	Q2= 0.38023671974681460OD 00	Q3= 0.37736130728268600OD 00
0.9000D-01	Q0= 0.80329100778903930OD 00	Q1= 0.44084048579102560OD-01
	Q2= 0.42156765246307580OD 00	Q3= 0.41837968870414970OD 00
0.1000D 00	Q0= 0.75908993529990720OD 00	Q1= 0.48184657661986900OD-01
	Q2= 0.46078102329580330OD 00	Q3= 0.45729652159249340OD 00

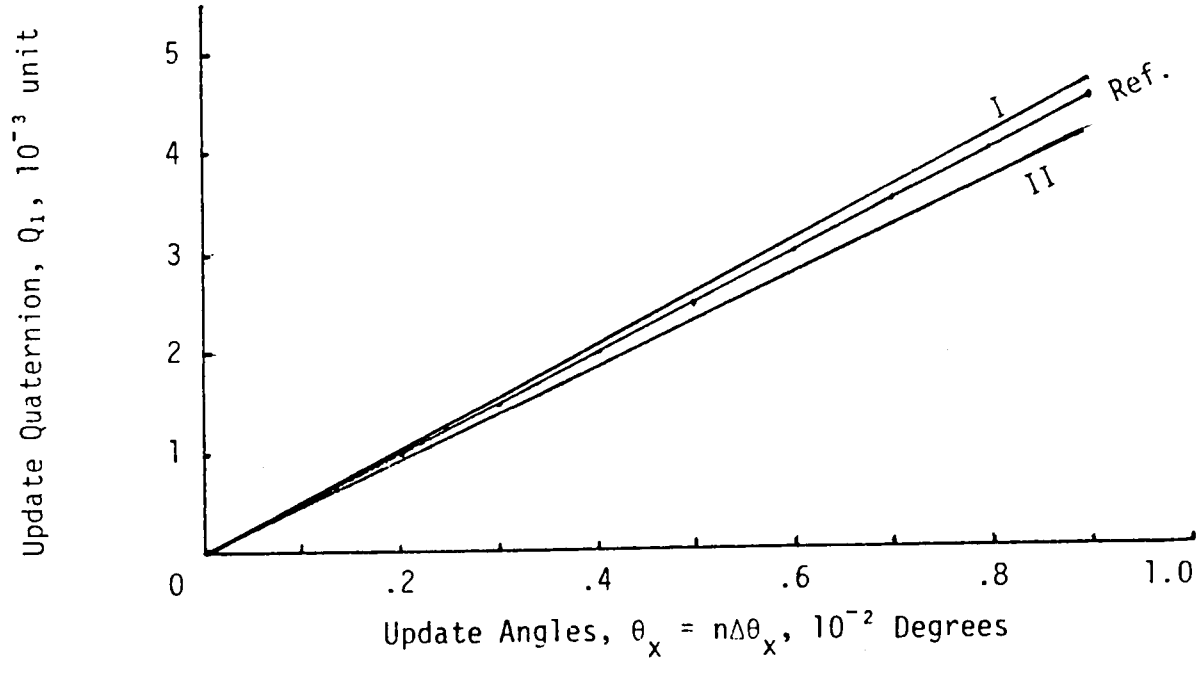
Table 20

Result of Update Quaternion Elements of Algorithm II for Set 6
as Input Angular Increments, and n Equals to 1 to 10

Update Angle $\theta_x = n\Delta\theta_x$	Update Quaternion Elements $q_{0n}, q_{1n}, q_{2n}, q_{3n}$	
0.0	Q0= 0.10000000000000000D 01	Q1= 0.0
	Q2= 0.0	Q3= 0.0
0.1000D-01	Q0= 0.99748750000000000D 00	Q1= 0.46624791666666500D-02
	Q2= 0.49724791666666500D-01	Q3= 0.50224791666666500D-01
0.2000D-01	Q0= 0.98996448934001690OD 00	Q1= 0.93015293755208290OD-02
	Q2= 0.99199716255208300D-01	Q3= 0.10019720375520830OD 00
0.3000D-01	Q0= 0.97746876646524800OD 00	Q1= 0.13893848090252460OD-01
	Q2= 0.14817625495794500D 00	Q3= 0.14966621851527370OD 00
0.4000D-01	Q0= 0.96006310836004020OD 00	Q1= 0.18416367556637290OD-01
	Q2= 0.19640839289052370OD 00	Q3= 0.19838334729764480OD 00
0.5000D-01	Q0= 0.93783495516400080OD 00	Q1= 0.22846370674565160OD-01
	Q2= 0.24365385485344170OD 00	Q3= 0.24610387874179490OD 00
0.6000D-01	Q0= 0.91089597090231430OD 00	Q1= 0.27161605108469230OD-01
	Q2= 0.28967532230637000OD 00	Q3= 0.29258810798728580OD 00
0.7000D-01	Q0= 0.87938148253700640OD 00	Q1= 0.31340395062966850OD-01
	Q2= 0.33424162544219880OD 00	Q3= 0.33760254073454250OD 00
0.8000D-01	Q0= 0.84344980015710480OD 00	Q1= 0.35361750162252380OD-01
	Q2= 0.37712890437295240OD 00	Q3= 0.38092106610689260OD 00
0.9000D-01	Q0= 0.80328142172269510OD 00	Q1= 0.39205470886331380OD-01
	Q2= 0.41812173359481900OD 00	Q3= 0.42232609242248120OD 00
0.1000D 00	Q0= 0.75907812635776560OD 00	Q1= 0.42852250034481900OD-01
	Q2= 0.45701420408402680OD 00	Q3= 0.46160964017099590OD 00

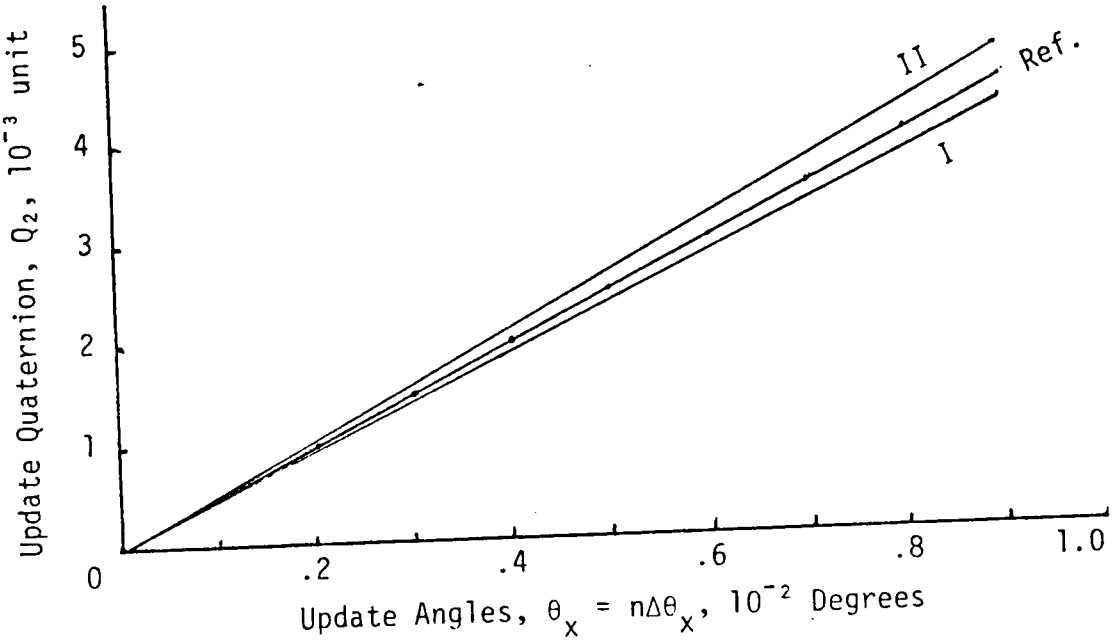


(a) Comparison of Q_0 for Set 1 (Ref.) and Set 2.

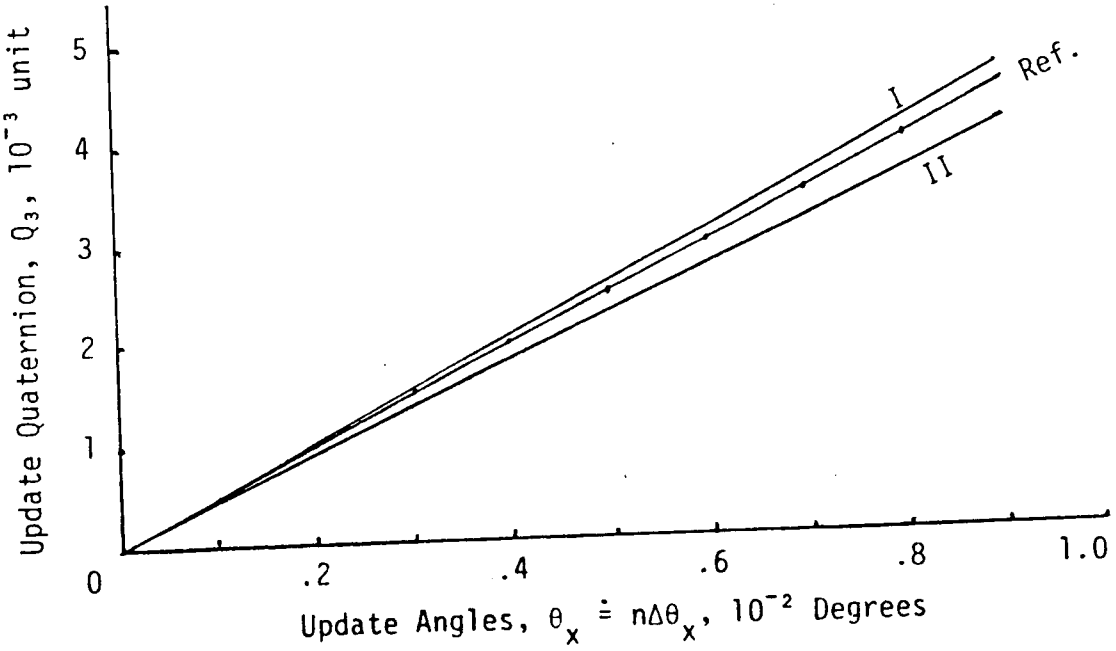


(b) Comparison of Q_1 for Set 1 (Ref.) and Set 2.

Figure 9. Comparison of update quaternion elements for Set 3 as references and Set 4 as input incremental angles of Algorithm I and Algorithm II. The scale of Curve I and II have been enlarged.

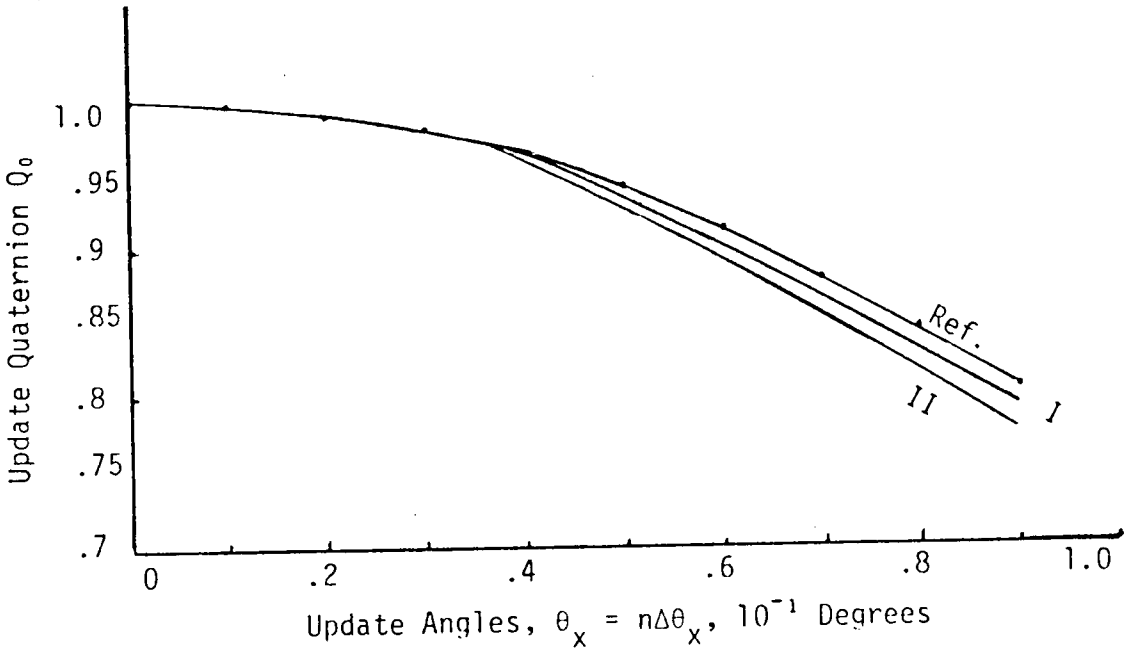


(c) Comparison of Q_2 for Set 1 (Ref.) and Set 2.

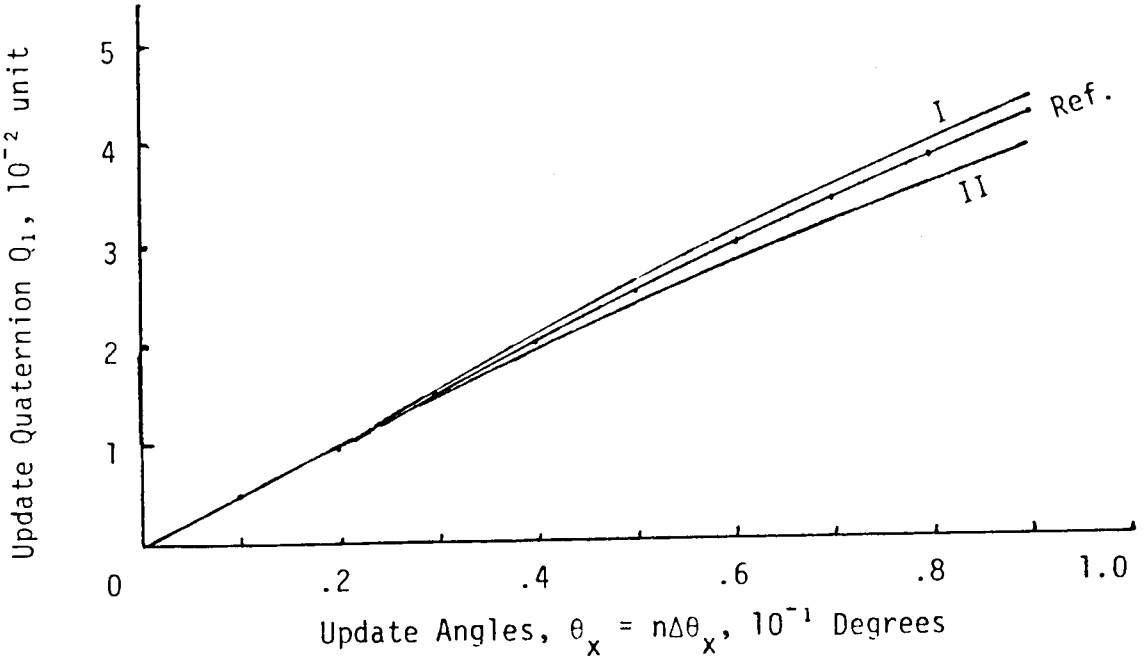


(d) Comparison of Q_3 for Set 1 (Ref.) and Set 2.

Figure 9 (continued)

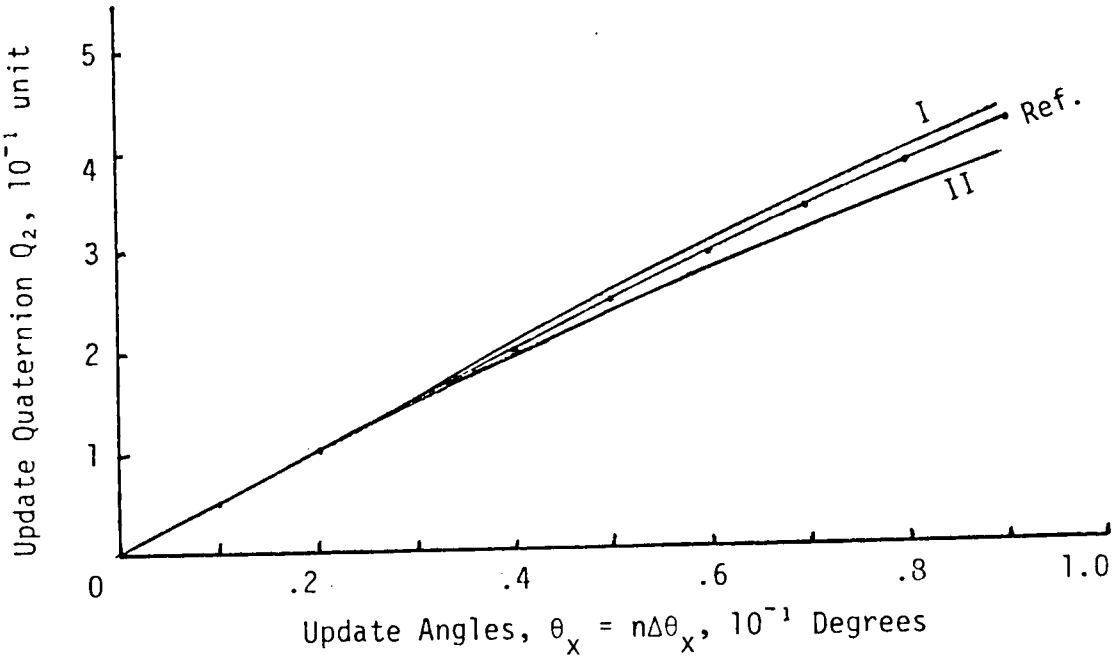


(a) Comparison of Q_0 for Set 1 (Ref.) and Set 2.

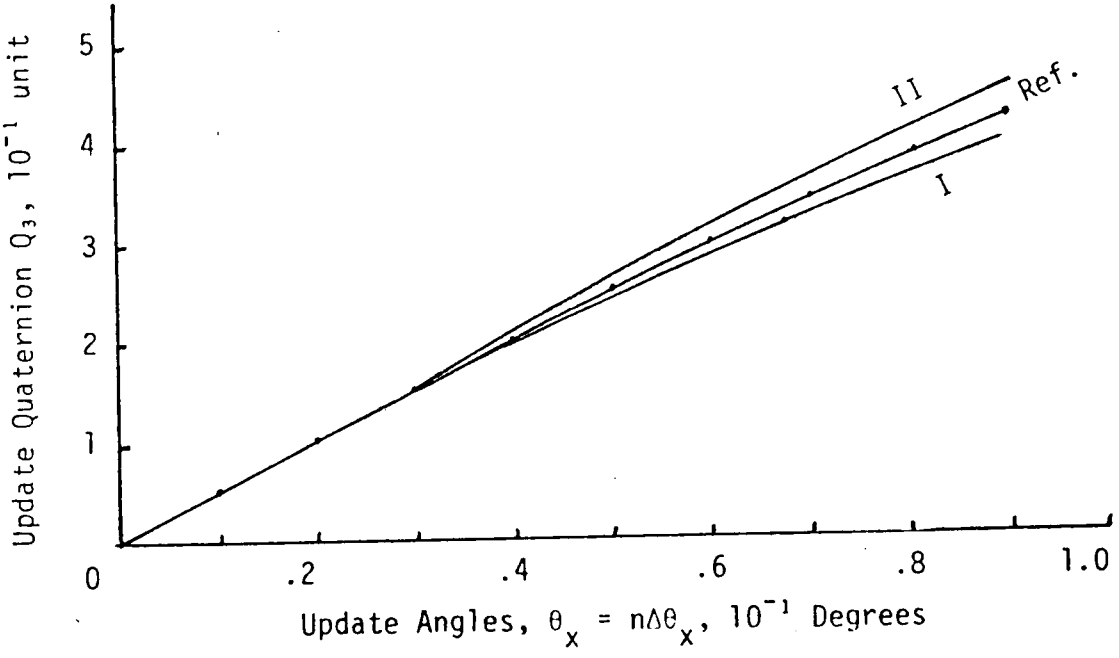


(b) Comparison of Q_1 for Set 1 (Ref.) and Set 2.

Figure 10. Comparison of update quaternion elements for Set 5 as references and Set 6 as input incremental angles of Algorithm I and Algorithm II. The scale of Curve I and II have been enlarged.



(c) Comparison of Q_2 for Set 1 (Ref.) and Set 2.



(d) Comparison of Q_3 for Set 1 (Ref.) and Set 2.

Figure 10 (continued)

CHAPTER V

CONCLUSIONS

Two quaternion update algorithms were investigated. One of these is developed from the algebra of quaternion and the other from solving the quaternion differential equations by Taylor series up to sixth order.

Three digital simulation tests were performed. By means of these tests, one may compare the two algorithms as follows.

Algorithm I: This algorithm requires more computation time, since square root operation is needed. Although the preservation of unity norm results in more deviation, the accuracy is higher than in Algorithm II.

Algorithm II: This algorithm requires less computation time due to having a simple expression and using only addition, subtraction, multiplication, and division as the computational operations. The preservation of unity norm is better than in Algorithm I, but this algorithm yields lower accuracy.

REFERENCES

REFERENCES

[1] J. C. Hung, "A Study of Strapdown Platform Technology," Scientific Report S-23, Department of Electrical Engineering, The University of Tennessee, Knoxville, Tennessee, April 1972.

[2] Y. Yeh and J. C. Hung, "DDA Realization of Attitude Algorithms," Proceedings of Fifth Hawaii International Conference of System Science, January 11-13, 1972.

[3] J. C. Hung, "Use of Quaternion in Attitude Algorithms" An unpublished note, Electrical Engineering Department, The University of Tennessee, February 1, 1977.

[4] James C. Wilcox, "A New Algorithm for Strapped-Down Inertial Navigation." Transactions on Aerospace and Electronic Systems, IEEE, September 1967.

[5] B. P. Ickes, "A New Method for Performing Digital Control System Attitude Computation Using Quaternions," AIAA Journal, Vol. 8, No. 1, January 1970.

[6] Alexander Macfarlane, "Vector Analysis and Quaternions," John Wiley and Sons, Inc., New York, 1906.

[7] Authur S. Hathaway, "A Primer of Quaternions," Macmillan and Co., New York, 1896.

[8] Phillip M. Morse and Herman Fesbach, "Methods of Theoretical Physics," McGraw-Hill, New York, 1953.

[9] William Rowan Hamilton, "Lecture on Quaternions," Macmillan and Co., Cambridge, 1853.

[10] P. Kelland and P. G. Tait, "Introduction to Quaternions," Macmillan and Co., London, 1873.

[11] P. G. Tait, "An Elementary treatise on Quaternions," Deighton, Bell and Co., Cambridge, October 1873.

[12] A. S. Hardy, "Elements of Quaternions," Ginn Heath and Co., Boston, 1881.

[13] T. G. Room and P. B. Kirkpatrick, "Mini-quaternion Geometry," Cambridge University Press, Cambridge, 1971.

[14] Peter A. Stark, "Introduction to Numerical Methods," Macmillan Publishing Co., Inc., New York, 1970.

[15] Garrett Birkhoff and Saunders MacLane, "A Survey of Modern Algebra," Macmillan Company, New York, 1958.

[16] R. W. Hamming, "Numerical Methods for Scientists and Engineers," McGraw-Hill Book Company, New York, 1973.

APPENDIXES

APPENDIX I

DERIVATION OF THE RATE OF CHANGE OF DIRECTION COSINE [2]

If $\hat{e}_x$, $\hat{e}_y$, $\hat{e}_z$ denote the unit vector in the system s_1 and $\hat{e}_x'$, $\hat{e}_y'$, $\hat{e}_z'$ denote the unit vector in the system s_2, then:

$$\begin{bmatrix} \hat{e}_x' \\ \hat{e}_y' \\ \hat{e}_z' \end{bmatrix} = \begin{bmatrix} M_{11} & M_{21} & M_{31} \\ M_{12} & M_{22} & M_{32} \\ M_{13} & M_{23} & M_{33} \end{bmatrix} \begin{bmatrix} \hat{e}_x \\ \hat{e}_y \\ \hat{e}_z \end{bmatrix} , \tag{A-I-1}$$

where the M_{ij}'s represent the direction cosine between the two systems. The rate of change of any vector caused by a rotation of the system s_2 is

$$\dot{\hat{e}}_x' = \hat{\Omega} \times \hat{e}_x' , \tag{A-I-2}$$

$$\dot{\hat{e}}_y' = \hat{\Omega} \times \hat{e}_y' , \tag{A-I-3}$$

$$\dot{\hat{e}}_z' = \hat{\Omega} \times \hat{e}_z' , \tag{A-I-4}$$

where

$$\hat{\Omega} = \omega_x \hat{e}_x' + \omega_y \hat{e}_y' + \omega_z \hat{e}_z' . \tag{A-I-5}$$

For a right-handed system

$$\dot{\hat{e}}_{x'} = \omega_z \hat{e}_{y'} - \omega_y \hat{e}_{z'}$$

$$= \hat{e}_x(\omega_z M_{12} - \omega_y M_{13}) + \hat{e}_y(\omega_z M_{22} - \omega_y M_{23}) + \hat{e}_z(\omega_z M_{32} - \omega_y M_{33}) \ , \tag{A-I-6}$$

$$\dot{\hat{e}}_{y'} = \omega_x \hat{e}_{z'} - \omega_z \hat{e}_{x'}$$

$$= \hat{e}_x(\omega_x M_{13} - \omega_z M_{11}) + \hat{e}_y(\omega_x M_{23} - \omega_z M_{21}) + \hat{e}_z(\omega_x M_{33} - \omega_z M_{31}) \ , \tag{A-I-7}$$

$$\dot{\hat{e}}_{z'} = \omega_y \hat{e}_{x'} - \omega_x \hat{e}_{y'}$$

$$= \hat{e}_x(\omega_y M_{11} - \omega_x M_{12}) + \hat{e}_y(\omega_y M_{21} - \omega_x M_{22}) + \hat{e}_z(\omega_y M_{31} - \omega_x M_{32}) \ . \tag{A-I-8}$$

Combining Eqs. (A-I-6), (A-I-7), and (A-I-8) in matrix form yields

$$\begin{bmatrix} \dot{\hat{e}}_{x'} \\ \dot{\hat{e}}_{y'} \\ \dot{\hat{e}}_{z'} \end{bmatrix} = \begin{bmatrix} \omega_z M_{12} - \omega_y M_{13} & \omega_z M_{22} - \omega_y M_{23} & \omega_z M_{32} - \omega_y M_{33} \\ \omega_x M_{13} - \omega_z M_{11} & \omega_x M_{23} - \omega_z M_{21} & \omega_x M_{33} - \omega_z M_{31} \\ \omega_y M_{11} - \omega_x M_{12} & \omega_y M_{21} - \omega_x M_{22} & \omega_y M_{31} - \omega_x M_{32} \end{bmatrix} \begin{bmatrix} \hat{e}_x \\ \hat{e}_y \\ \hat{e}_z \end{bmatrix} \ . \tag{A-I-9}$$

From (A-I-1), one may obtain

$$\begin{bmatrix} \dot{\hat{e}}_x \\ \dot{\hat{e}}_y \\ \dot{\hat{e}}_z \end{bmatrix} = \begin{bmatrix} \dot{M}_{11} & \dot{M}_{21} & \dot{M}_{31} \\ \dot{M}_{12} & \dot{M}_{22} & \dot{M}_{32} \\ \dot{M}_{13} & \dot{M}_{23} & \dot{M}_{33} \end{bmatrix} \begin{bmatrix} \hat{e}_x \\ \hat{e}_y \\ \hat{e}_z \end{bmatrix} . \qquad \text{(A-I-10)}$$

Combining (A-I-9) and (A-I-10) yields

$$\begin{bmatrix} \dot{M}_{11} & \dot{M}_{21} & \dot{M}_{31} \\ \dot{M}_{12} & \dot{M}_{22} & \dot{M}_{32} \\ \dot{M}_{13} & \dot{M}_{23} & \dot{M}_{33} \end{bmatrix} = \begin{bmatrix} \omega_z M_{12} - \omega_y M_{13} & \omega_z M_{22} - \omega_y M_{23} & \omega_z M_{32} - \omega_y M_{33} \\ \omega_x M_{13} - \omega_z M_{11} & \omega_x M_{23} - \omega_z M_{21} & \omega_x M_{33} - \omega_z M_{31} \\ \omega_y M_{11} - \omega_x M_{12} & \omega_y M_{21} - \omega_x M_{22} & \omega_y M_{31} - \omega_x M_{32} \end{bmatrix} .$$

(A-I-11)

Rearranging Eq. (A-I-11) results in

$$\begin{bmatrix} \dot{M}_{11} & \dot{M}_{12} & \dot{M}_{13} \\ \dot{M}_{21} & \dot{M}_{22} & \dot{M}_{23} \\ \dot{M}_{31} & \dot{M}_{32} & \dot{M}_{33} \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} .$$

(A-I-12)

APPENDIX II

DERIVATION OF ANGULAR VELOCITY AND ANGULAR ACCELERATION IN TERMS OF ANGULAR INCREMENTS

Let an angular increment $\Delta\theta_{i1}$ occurring over time $\Delta T/2$ and an angular increment $\Delta\theta_{i2}$ occurring over the time interval from $\Delta T/2$ to ΔT be given. Also, assume that the angular acceleration $\dot{\omega}_i$ is constant over the interval, i.e.,

$$\dot{\omega}_i = C \ , \tag{A-II-1}$$

then,

$$\omega_i = ct + d \ , \tag{A-II-2}$$

and

$$\theta_i = \frac{1}{2} ct^2 + dt + e \ . \tag{A-II-3}$$

Let the initial position of θ be zero, i.e., at $t = 0$, $\theta_i = 0$. Then, Eq. (A-II-3) yields $e = 0$, and Eq. (A-II-2) and Eq. (A-II-1) become

$$\omega_i = \omega_{in} = d \ ,$$

$$\dot{\omega}_i = \dot{\omega}_{in} = c \ .$$

Now at $t = \Delta T/2$, $\theta = \Delta\theta_{i1}$

$$\Delta\theta_{i1} = \frac{1}{2} c \left(\frac{\Delta T^2}{4}\right) + d \left(\frac{\Delta T}{2}\right) . \qquad \text{(A-II-4)}$$

At $t = \Delta T$, $\theta = \Delta\theta_{i1} + \Delta\theta_{i2}$

$$\Delta\theta_{i1} + \Delta\theta_{i2} = \frac{1}{2} c\Delta T^2 + d\Delta T . \qquad \text{(A-II-5)}$$

Solving Eqs. (A-II-4) and (A-II-5) yields

$$d = \omega_{in} = \frac{3\Delta\theta_{i1} - \Delta\theta_{i2}}{\Delta T} , \qquad \text{(A-II-6)}$$

and

$$c = \dot{\omega}_{in} = \frac{4(\Delta\theta_{i2} - \Delta\theta_{i1})}{\Delta T^2} . \qquad \text{(A-II-7)}$$

APPENDIX III

DERIVATION OF EULER ANGLE TRANSFORMATION [1][2][8]

One way to represent the Euler angle transformation matrix of two coordinate systems (X, Y, Z) and (X′, Y′, Z′) in right-handed rotations is shown in Figure 11 [8].

First, rotate along X axis by an angle ψ_1.

Next, rotate along the new Z axis by an angle ψ_3.

Finally, rotate along the new Y axis by an angle ψ_2.

The overall transformation is given by

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = [B] \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} , \qquad \text{(A-III-1)}$$

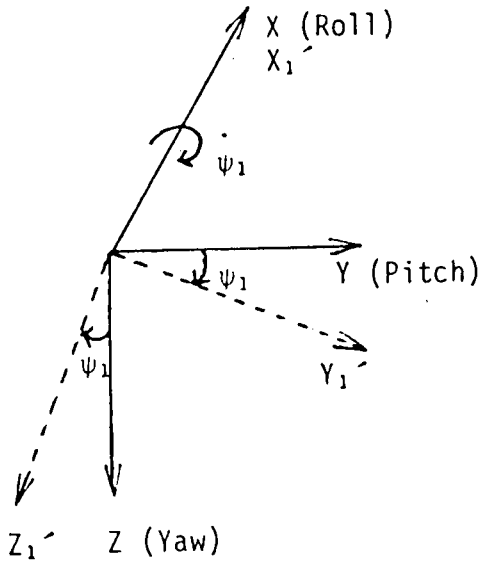
where

$$B = B\psi_2 B\psi_3 B\psi_1$$

$$= \begin{bmatrix} \cos\psi_2 & 0 & \sin\psi_2 \\ 0 & 1 & 0 \\ -\sin\psi_2 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} \cos\psi_3 & -\sin\psi_3 & 0 \\ \sin\psi_3 & \cos\psi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi_1 & -\sin\psi_1 \\ 0 & \sin\psi_1 & \cos\psi_1 \end{bmatrix}$$

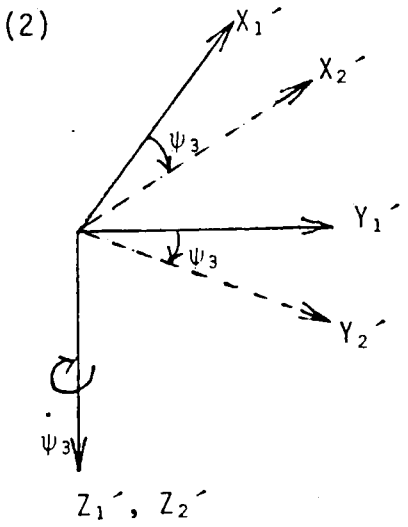
$$= \begin{bmatrix} \cos\psi_2\cos\psi_3 & -\cos\psi_1\cos\psi_2\sin\psi_3 + \sin\psi_1\sin\psi_2 & \sin\psi_1\cos\psi_2\sin\psi_3 + \cos\psi_1\sin\psi_2 \\ \sin\psi_3 & \cos\psi_1\cos\psi_3 & -\sin\psi_1\cos\psi_3 \\ -\sin\psi_2\cos\psi_3 & \cos\psi_1\sin\psi_2\sin\psi_3 + \sin\psi_1\cos\psi_2 & -\sin\psi_1\sin\psi_2\sin\psi_3 + \cos\psi_1\cos\psi_2 \end{bmatrix} . \qquad \text{(A-III-2)}$$

(1)



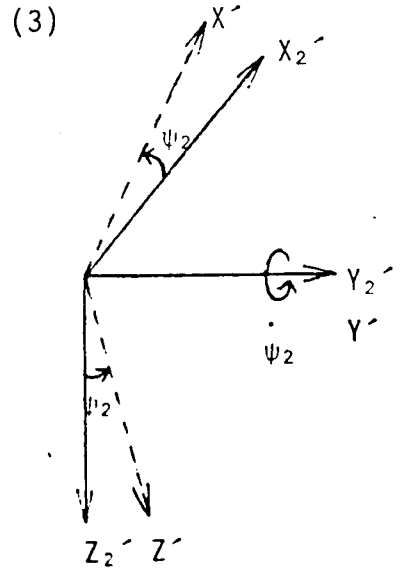
$$\begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi_1 & -\sin\psi_1 \\ 0 & \sin\psi_1 & \cos\psi_1 \end{bmatrix}}_{B\psi_1} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

(2)



$$\begin{bmatrix} X_2' \\ Y_2' \\ Z_2' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\psi_3 & -\sin\psi_3 & 0 \\ \sin\psi_3 & \cos\psi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{B\psi_3} \begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \end{bmatrix}$$

(3)



$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\psi_2 & 0 & \sin\psi_2 \\ 0 & 1 & 0 \\ -\sin\psi_2 & 0 & \cos\psi_2 \end{bmatrix}}_{B\psi_2} \begin{bmatrix} X_2' \\ Y_2' \\ Z_2' \end{bmatrix}$$

Figure 11. Transformation of rotating of the coordinate axis.

If the X′Y′Z′-frame is rotating with respect to the XYZ-frame, then the Euler angle ψ_1, ψ_2, and ψ_3 change with time. The rotation can be described as a function of their rates, namely, $\dot{\psi}_1$, $\dot{\psi}_2$, and $\dot{\psi}_3$.

Let $\hat{\omega}$ be a vector representing the angular velocity of the X′Y′Z′-frame with respect to the XYZ-frame. Then

$$\hat{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = B\psi_2 B\psi_3 \begin{bmatrix} \dot{\psi} \\ 0 \\ 0 \end{bmatrix} + B\psi_2 \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\psi}_2 \\ 0 \end{bmatrix}$$

$$\hat{\omega} = \begin{bmatrix} \cos\psi_2 & 0 & \sin\psi_2 \\ 0 & 1 & 0 \\ -\sin\psi_2 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} \cos\psi_3 & -\sin\psi_3 & 0 \\ \sin\psi_3 & \cos\psi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\psi}_1 \\ 0 \\ 0 \end{bmatrix}$$

$$+ \begin{bmatrix} \cos\psi_2 & 0 & \sin\psi_2 \\ 0 & 1 & 0 \\ -\sin\psi_2 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\psi}_2 \\ 0 \end{bmatrix} ,$$

so

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \cos\psi_2\cos\psi_3 & 0 & \sin\psi_2 \\ \sin\psi_3 & 1 & 0 \\ -\sin\psi_2\cos\psi_3 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} \dot{\psi}_1 \\ \dot{\psi}_2 \\ \dot{\psi}_3 \end{bmatrix} . \qquad \text{(A-III-3)}$$

In incremental form, one may multiply Δt to both sides of Eq. (A-III-3),

$$\begin{bmatrix} \omega_x\Delta t \\ \omega_y\Delta t \\ \omega_z\Delta t \end{bmatrix} = \begin{bmatrix} \cos\psi_2\cos\psi_3 & 0 & \sin\psi_2 \\ \sin\psi_3 & 1 & 0 \\ -\sin\psi_2\cos\psi_3 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} \Delta\psi_1 \\ \Delta\psi_2 \\ \Delta\psi_3 \end{bmatrix} ,$$

therefore,

$$\begin{bmatrix} \Delta\theta_x \\ \Delta\theta_y \\ \Delta\theta_z \end{bmatrix} = \begin{bmatrix} \cos\psi_2\cos\psi_3 & 0 & \sin\psi_2 \\ \sin\psi_3 & 1 & 0 \\ -\sin\psi_2\cos\psi_3 & 0 & \cos\psi_2 \end{bmatrix} \begin{bmatrix} \Delta\psi_1 \\ \Delta\psi_2 \\ \Delta\psi_3 \end{bmatrix} . \qquad \text{(A-III-4)}$$

APPENDIX IV

SUM OF SQUARE OF UPDATE QUATERNION FOR ALGORITHM I

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C       COMPUTATION OF THE SUM OF SQUARE OF UPDATE                    C
C       QUATERNION ELEMENTS FOR ALGORITHM I                           C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      DIMENSION TH1(3),TH2(3),QOLD(4),QNEW(4)
      DOUBLE PRECISION TH1,TH2,THE1,THE2,QOLD,QNEW,THSUM,
     *                 Q0P,Q1P,Q2P,Q3P,QSUM,U,V,W,X,Y,Z,
     *                 A1,A2,A3,B
C
C       INITIALIZATION
      QOLD(1)=1.D0
      QOLD(2)=0.D0
      QOLD(3)=0.D0
      QOLD(4)=0.D0
      DO 1 I=1,3
      TH1(I)=0.D0
      TH2(I)=0.D0
    1 CONTINUE
C
C       READING OF INPUT ANGULAR INCREMENTS
      READ(5,7) U,V,W
    7 FORMAT(3E16.9)
      READ(5,8) X,Y,Z
    8 FORMAT(3E16.9)
C
C       COMPUTATION OF APPROACH 1 ALGORITHM
      DO 10 J=1,11
      THE1=DSQRT(TH1(1)**2+TH1(2)**2+TH1(3)**2)
      THE2=DSQRT(TH2(1)**2+TH2(2)**2+TH2(3)**2)
      A1=(TH1(1)+TH2(1))/2
      A2=(TH1(2)+TH2(2))/2
      A3=(TH1(3)+TH2(3))/2
      B=(A1**2+A2**2+A3**2)/2
      Q0P=1-B
      Q1P=(TH1(1)+TH2(1))/2+(TH2(2)*TH1(3)-TH2(3)*TH1(2))/4
     *    -(TH2(1)*(3*THE1**2+THE2**2)+TH1(1)*(3*THE2**2
     *    +THE1**2))/48
      Q2P=(TH1(2)+TH2(2))/2+(TH1(1)*TH2(3)-TH2(1)*TH1(3))/4
     *    -(TH2(2)*(3*THE1**2+THE2**2)+TH1(2)*(3*THE2**2
     *    +THE1**2))/48
```

```
      Q3P=(TH1(3)+TH2(3))/2+(TH2(1)*TH1(2)-TH2(2)*TH1(1))/4
     *     -(TH2(3)*(3*THE1**2+THE2**2)+TH1(3)*(3*THE2**2
     *     +THE1**2))/48
C
C     CALCULATION OF QUATERNION DIFFERENCE EQUATION
      QNEW(1)=Q0P*QOLD(1)-Q1P*QOLD(2)-Q2P*QOLD(3)-Q3P*QOLD(4)
      QNEW(2)=Q1P*QOLD(1)+Q0P*QOLD(2)-Q3P*QOLD(3)+Q2P*QOLD(4)
      QNEW(3)=Q2P*QOLD(1)+Q3P*QOLD(2)+Q0P*QOLD(3)-Q1P*QOLD(4)
      QNEW(4)=Q3P*QOLD(1)-Q2P*QOLD(2)+Q1P*QOLD(3)+Q0P*QOLD(4)
C
C     COMPUTATION OF THE SUM OF SQUARE OF UPDATED
C     QUATERNION ELEMENTS
      QSUM=QNEW(1)**2+QNEW(2)**2+QNEW(3)**2+QNEW(4)**2
C
C     ACCUMULATION OF INPUT INCREMENTAL ANGLES
      THSUM=(J-1)*(TH1(1)+TH2(1))
C
C     PRINTING OF OUTPUT DATA
      WRITE(6,30) THSUM,QSUM
   30 FORMAT(/,30X,E11.4,7X,E25.18)
C
C     GIVING OF INPUT ANGULAR INCREMENTS
      TH1(1)=U
      TH2(1)=V
      TH1(2)=W
      TH2(2)=X
      TH1(3)=Y
      TH2(3)=Z
C
C     STORAGE OF UPDATING QUATERNION
      QOLD(1)=QNEW(1)
      QOLD(2)=QNEW(2)
      QOLD(3)=QNEW(3)
      QOLD(4)=QNEW(4)
   10 CONTINUE
      STOP
      END
```

APPENDIX V

SUM OF SQUARE OF UPDATE QUATERNION FOR ALGORITHM II

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C       COMPUTATION OF THE SUM OF SQUARE OF UPDATE                     C
C       QUATERNION ELEMENTS FOR ALGORITHM II                           C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
        DIMENSION TH1(3),TH2(3),QOLD(4),QNEW(4)
        DOUBLE PRECISION TH1,TH2,A1,A2,A3,B,QOLD,QNEW,
     *                   Q0P,Q1P,Q2P,Q3P,QSUM,THSUM,
     *                   U,V,W,X,Y,Z
C
C       INITIALIZATION
        QOLD(1)=1.D0
        QOLD(2)=0.D0
        QOLD(3)=0.D0
        QOLD(4)=0.D0
        DO 1 I=1,3
        TH1(I)=0.D0
        TH2(I)=0.D0
    1   CONTINUE
C
C       READING OF INPUT ANGULAR INCREMENTS
        READ(5,7) U,V,W
    7   FORMAT(3E16.9)
        READ(5,8) X,Y,Z
    8   FORMAT(3E16.9)
C
C       COMPUTATION OF APPROACH 2 ALGORITHM
        DO 10 J=1,11
        A1=(TH1(1)+TH2(1))/2
        A2=(TH1(2)+TH2(2))/2
        A3=(TH1(3)+TH2(3))/2
        B=(A1**2+A2**2+A3**2)/2
        Q0P=1-B
        Q1P=A1+((TH1(2)*TH2(3)-TH2(2)*TH1(3))-A1*B)/3
        Q2P=A2+((TH2(1)*TH1(3)-TH1(1)*TH2(3))-A2*B)/3
        Q3P=A3+((TH1(1)*TH2(2)-TH2(1)*TH1(2))-A3*B)/3
C
C       COMPUTATION OF QUATERNION DIFFERENCE EQUATION
        QNEW(1)=Q0P*QOLD(1)-Q1P*QOLD(2)-Q2P*QOLD(3)-Q3P*QOLD(4)
        QNEW(2)=Q1P*QOLD(1)+Q0P*QOLD(2)+Q3P*QOLD(3)-Q2P*QOLD(4)
        QNEW(3)=Q2P*QOLD(1)-Q3P*QOLD(2)+Q0P*QOLD(3)+Q1P*QOLD(4)
        QNEW(4)=Q3P*QOLD(1)+Q2P*QOLD(2)-Q1P*QOLD(3)+Q0P*QOLD(4)
```

```
C
C       COMPUTATION OF THE SUM OF SQUARE OF UPDATED
C       QUATERNION ELEMENTS
        QSUM=QNEW(1)**2+QNEW(2)**2+QNEW(3)**2+QNEW(4)**2
C
C       ACCUMULATION OF INPUT INCREMENTAL ANGLES
        THSUM=(J-1)*(TH1(1)+TH2(1))
C
C       PRINTING OF OUTPUT DATA
        WRITE(6,30) THSUM,QSUM
   30   FORMAT(/,30X,E11.4,7X,E25.18)
C
C       GIVING OF INPUT ANGULAR INCREMENTS
        TH1(1)=U
        TH2(1)=V
        TH1(2)=W
        TH2(2)=X
        TH1(3)=Y
        TH2(3)=Z
C
C       STORAGE OF UPDATING QUATERNION
        QOLD(1)=QNEW(1)
        QOLD(2)=QNEW(2)
        QOLD(3)=QNEW(3)
        QOLD(4)=QNEW(4)
   10   CONTINUE
        STOP
        END
```

APPENDIX VI

EULER ANGULAR INCREMENTS

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C       COMPUTATION OF EULER ANGULAR INCREMENTS                       C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      DIMENSION TH1(3),TH2(3),QOLD(4),QNEW(4),EM(3,3),A(3,3),
     *          B(3,2),TH3(3,1),TH4(3,1),WKAREA(9)
      DOUBLE PRECISION EM,A,TH4,B,QOLD,QNEW,Q0P,Q1P,Q2P,Q3P,
     *                 TH1,TH2,TH3,EM11,EM21,EM31,EM22,EM23,
     *                 TH321,TH331,THE1,THE2,TH51,TH52,TH5,
     *                 P2,B21,B3,B31,ERROR1,ERROR2,ERROR3,
     *                 TOLD11,TOLD21,TOLD31,C2,C3,S2,S3,RTOD,
     *                 B11OLD,B21OLD,B31OLD,W,X,Y,Z,U,V,SQ,
     *                 TH6,TH7,TH311,B1,B11,DTOR
C
C       INITIALIZATION
      RTOD=180.D0/3.14159265358979323846D0
      DTOR=1.0D0/RTOD
      DO 1 I=1,3
      TH1(I)=0.D0
      TH2(I)=0.D0
    1 CONTINUE
      QOLD(1)=1.D0
      QOLD(2)=0.D0
      QOLD(3)=0.D0
      QOLD(4)=0.D0
      B11OLD=0.D0
      B21OLD=0.D0
      B31OLD=0.D0
C
C       READING OF INPUT ANGULAR INCREMENTS
      READ(5,2) U,V,W
    2 FORMAT(3E16.9)
      READ(5,3) X,Y,Z
    3 FORMAT(3E16.9)
C
C       COMPUTATION OF APPROCH 1 ALGORITHM
      DO 10 J=1,41
      TH1(1)=TH1(1)*DTOR
      TH2(1)=TH2(1)*DTOR
      TH1(2)=TH1(2)*DTOR
      TH2(2)=TH2(2)*DTOR
      TH1(3)=TH1(3)*DTOR
      TH2(3)=TH2(3)*DTOR
```

```
      THE1=DSQRT(TH1(1)**2+TH1(2)**2+TH1(3)**2)
      THE2=DSQRT(TH2(1)**2+TH2(2)**2+TH2(3)**2)
      Q0P=1-((TH1(1)+TH2(1))**2+(TH1(2)+TH2(2))**2
     *     +(TH1(3)+TH2(3))**2)/8
      Q1P=(TH1(1)+TH2(1))/2+(TH2(2)*TH1(3)-TH2(3)*TH1(2))/4
     *     -(TH2(1)*(3*THE1**2+THE2**2)+TH1(1)*(3*THE2**2
     *     +THE1**2))/48
      Q2P=(TH1(2)+TH2(2))/2+(TH1(1)*TH2(3)-TH2(1)*TH1(3))/4
     *     -(TH2(2)*(3*THE1**2+THE2**2)+TH1(2)*(3*THE2**2
     *     +THE1**2))/48
      Q3P=(TH1(3)+TH2(3))/2+(TH2(1)*TH1(2)-TH2(2)*TH1(1))/4
     *     -(TH2(3)*(3*THE1**2+THE2**2)+TH1(3)*(3*THE2**2
     *     +THE1**2))/48
C
C     COMPUTATION OF QUATERNION DIFFERENCE EQUATION
      QNEW(1)=Q0P*QOLD(1)-Q1P*QOLD(2)-Q2P*QOLD(3)-Q3P*QOLD(4)
      QNEW(2)=Q1P*QOLD(1)+Q0P*QOLD(2)-Q3P*QOLD(3)+Q2P*QOLD(4)
      QNEW(3)=Q2P*QOLD(1)+Q3P*QOLD(2)+Q0P*QOLD(3)-Q1P*QOLD(4)
      QNEW(4)=Q3P*QOLD(1)-Q2P*QOLD(2)+Q1P*QOLD(3)+Q0P*QOLD(4)
C
C     GENERATION OF QUATERNION DIRECTION COSINE MATRIX
      EM(1,1)=1-2*(QNEW(3)**2+QNEW(4)**2)
      EM(1,2)=2*(QNEW(2)*QNEW(3)-QNEW(1)*QNEW(4))
      EM(1,3)=2*(QNEW(2)*QNEW(4)+QNEW(1)*QNEW(3))
      EM(2,1)=2*(QNEW(2)*QNEW(3)+QNEW(1)*QNEW(4))
      EM(2,2)=1-2*(QNEW(4)**2+QNEW(2)**2)
      EM(2,3)=2*(QNEW(3)*QNEW(4)-QNEW(1)*QNEW(2))
      EM(3,1)=2*(QNEW(2)*QNEW(4)-QNEW(1)*QNEW(3))
      EM(3,2)=2*(QNEW(3)*QNEW(4)+QNEW(1)*QNEW(2))
      EM(3,3)=1-2*(QNEW(2)**2+QNEW(3)**2)
C
C     COMPUTATION OF ROLL,PITCH,AND YAW ANGLES
      EM22=EM(2,2)
      EM23=EM(2,3)
      EM11=EM(1,1)
      EM31=EM(3,1)
      EM21=EM(2,1)
      TH3(1,1)=(-DATAN(EM23/EM22))
      TH3(2,1)=(-DATAN(EM31/EM11))
      TH311=TH3(1,1)
      X1=DCOS(TH311)
      TH3(3,1)=(DATAN(X1*EM21/EM22))
C
C     COMPUTATION OF THE DIFFERENCE BETWEEN UPDATED
C     AND UPDATING EULER ANGLES
      TH321=TH3(2,1)
      TH331=TH3(3,1)
```

```
      IF(J .EQ. 1) GO TO 50
      TH4(1,1)=TH311-TOLD11
      TH4(2,1)=TH321-TOLD21
      TH4(3,1)=TH331-TOLD31
C
C     COMPUTATION OF EULER INCREMENTAL ANGLES
      C2=DCOS(TH321)
      C3=DCOS(TH331)
      S2=DSIN(TH321)
      S3=DSIN(TH331)
      A(1,1)=C2*C3
      A(1,2)=0.D0
      A(1,3)=S2
      A(2,1)=S3
      A(2,2)=1.D0
      A(2,3)=0.D0
      A(3,1)=-S2*C3
      A(3,2)=0.D0
      A(3,3)=C2
C
C     COMPUTATION OF MULTIPLICATION TWO MATRIX IN IMSL
C     SUBROUTINE
      CALL VMULFF(A,TH4,3,3,1,3,3,B,3,IER)
C
C     ACCUMULATION OF EULER INCREMENTAL ANGLES
      B1=B(1,1)
      B11=(B1)*RTOD
      B2=B(2,1)
      B21=(B2)*RTOD
      B3=B(3,1)
      B31=(B3)*RTOD
      B11=B11+B11OLD
      B21=B21+B21OLD
      B31=B31+B31OLD
C
C     ACCUMULATION OF INPUT INCREMENTAL ANGLES
      TH5=(J-1)*(TH1(1)+TH2(1))*RTOD
      TH6=(J-1)*(TH1(2)+TH2(2))*RTOD
      TH7=(J-1)*(TH1(3)+TH2(3))*RTOD
C
C     COMPUTATION OF ERROR BETWEEN INPUT AND EULER
C     INCREMENTAL ANGLES
      ERROR1=TH5-B11
      ERROR2=TH6-B21
      ERROR3=TH7-B31
```

```
C
C       PRINTING OF OUTPUT DATA
        DO 20 I=1,11
        JJ=4*I+1
        IF(J .EQ. JJ) GO TO 30
   20   CONTINUE
        GO TO 45
   30   WRITE(6,40) TH5,B11,FRROR1
   40   FORMAT(/,20X,E11.4,5X,E25.18,5X,E25.18)
C
C       STORAGE OF ACCUMULATIVE EULER INCREMENTAL ANGLES
   45   B11OLD=B11
        B21OLD=B21
        B31OLD=B31
C
C       STORAGE OF UPDATING EULER ANGLES
   50   TOLD11=TH311
        TOLD21=TH321
        TOLD31=TH331
C
C       STORAGE OF UPDATING QUOTERNION
        QOLD(1)=QNEW(1)
        QOLD(2)=QNEW(2)
        QOLD(3)=QNEW(3)
        QOLD(4)=QNEW(4)
C
C       GIVING OF INPUT INCREMENTAL ANGLES
        TH1(1)=U
        TH2(1)=V
        TH1(2)=W
        TH2(2)=X
        TH1(3)=Y
        TH2(3)=Z
   10   CONTINUE
        STOP
        END
```

APPENDIX VII

UPDATE QUATERNION FOR ALGORITHM I

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C       COMPUTATION OF UPDATE QUATERNION OF ALGORITHM I               C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
        DIMENSION TH1(3),TH2(3),QOLD(4),QNEW(4)
        DOUBLE PRECISION TH1,TH2,THE1,THE2,QOLD,QNEW,THSUM,
     *                   Q0P,Q1P,Q2P,Q3P,U,V,W,X,Y,Z
C
C       INITIALIZATION
        QOLD(1)=1.D0
        QOLD(2)=0.D0
        QOLD(3)=0.D0
        QOLD(4)=0.D0
        DO 1 I=1,3
        TH1(I)=0.D0
        TH2(I)=0.D0
    1   CONTINUE
C
C       READING OF INPUT ANGULAR INCREMENTS
        READ(5,2) U,V,W
    2   FORMAT(3E16.9)
        READ(5,3) X,Y,Z
    3   FORMAT(3E16.9)
C
C       COMPUTATION OF APPROACH 1 ALGORITHM
        DO 10 J=1,10001
        THE1=DSQRT(TH1(1)**2+TH1(2)**2+TH1(3)**2)
        THE2=DSQRT(TH2(1)**2+TH2(2)**2+TH2(3)**2)
        Q0P=1-((TH1(1)+TH2(1))**2+(TH1(2)+TH2(2))**2
     *      +(TH1(3)+TH2(3))**2)/8
        Q1P=(TH1(1)+TH2(1))/2+(TH2(2)*TH1(3)-TH2(3)*TH1(2))/4
     *      -(TH2(1)*(3*THE1**2+THE2**2)+TH1(1)*(3*THE2**2
     *      +THE1**2))/48
        Q2P=(TH1(2)+TH2(2))/2+(TH1(1)*TH2(3)-TH2(1)*TH1(3))/4
     *      -(TH2(2)*(3*THE1**2+THE2**2)+TH1(2)*(3*THE2**2
     *      +THE1**2))/48
        Q3P=(TH1(3)+TH2(3))/2+(TH2(1)*TH1(2)-TH2(2)*TH1(1))/4
     *      -(TH2(3)*(3*THE1**2+THE2**2)+TH1(3)*(3*THE2**2
     *      +THE1**2))/48
```

```
C
C       CALCULATION OF QUATERNION DIFFERENCE EQUATION
        QNEW(1)=Q0P*QOLD(1)-Q1P*QOLD(2)-Q2P*QOLD(3)-Q3P*QOLD(4)
        QNEW(2)=Q1P*QOLD(1)+Q0P*QOLD(2)-Q3P*QOLD(3)+Q2P*QOLD(4)
        QNEW(3)=Q2P*QOLD(1)+Q3P*QOLD(2)+Q0P*QOLD(3)-Q1P*QOLD(4)
        QNEW(4)=Q3P*QOLD(1)-Q2P*QOLD(2)+Q1P*QOLD(3)+Q0P*QOLD(4)
C
C       ACCUMULATION OF INPUT INCREMENTAL ANGLES
        THSUM=(J-1)*(TH1(1)+TH2(1))
C
C       PRINTING OF OUTPUT DATA
        IF(J .EQ. 1) GO TO 30
        DO 20 I=1,50
        JJ=1000*I+1
        IF(J .EQ. JJ) GO TO 30
   20   CONTINUE
        GO TO 70
   30   WRITE(6,40) THSUM,QNEW(1),QNEW(2)
   40   FORMAT(20X,E11.4,5X,3HQ0=,E25.18,5X,3HQ1=,F25.18)
        IF(J .EQ. 1) GO TO 50
        IF(J .EQ. JJ) GO TO 50
   50   WRITE(6,60) QNEW(3),QNEW(4)
   60   FORMAT(36X,3HQ2=,E25.18,5X,3HQ3=,F25.18)
C
C       GIVING OF INPUT INCREMENTAL ANGLES
   70   TH1(1)=U
        TH2(1)=V
        TH1(2)=W
        TH2(2)=X
        TH1(3)=Y
        TH2(3)=Z
C
C       STORAGE OF UPDATING QUATERNION
        QOLD(1)=QNEW(1)
        QOLD(2)=QNEW(2)
        QOLD(3)=QNEW(3)
        QOLD(4)=QNEW(4)
   10   CONTINUE
        STOP
        END
```

APPENDIX VIII

UPDATE QUATERNION FOR ALGORITHM II

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C       COMPUTATION OF UPDATE QUATERNION OF ALGORITHM II              C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
        DIMENSION TH1(3),TH2(3),QOLD(4),QNEW(4)
        DOUBLE PRECISION TH1,TH2,A1,A2,A3,B,QOLD,QNEW,THSUM,
     *                   Q0P,Q1P,Q2P,Q3P,U,V,W,X,Y,Z
C
C       INITIALIZATION
        QOLD(1)=1.D0
        QOLD(2)=0.D0
        QOLD(3)=0.D0
        QOLD(4)=0.D0
        DO 1 I=1,3
        TH1(I)=0.D0
        TH2(I)=0.D0
    1   CONTINUE
C
C       READING OF INPUT ANGULAR INCREMENTS
        READ(5,2) U,V,W
    2   FORMAT(3E16.9)
        READ(5,3) X,Y,Z
    3   FORMAT(3E16.9)
C
C       CALCULATION OF APPROACH 2 ALGORITHM
        DO 10 J=1,10001
        A1=(TH1(1)+TH2(1))/2
        A2=(TH1(2)+TH2(2))/2
        A3=(TH1(3)+TH2(3))/2
        B=(A1**2+A2**2+A3**2)/2
        Q0P=1-B
        Q1P=A1+((TH1(2)*TH2(3)-TH2(2)*TH1(3))-A1*B)/3
        Q2P=A2+((TH2(1)*TH1(3)-TH1(1)*TH2(3))-A2*B)/3
        Q3P=A3+((TH1(1)*TH2(2)-TH2(1)*TH1(2))-A3*B)/3
C
C       COMPUTATION OF QUATERNION DIFFERENCE EQUATION
        QNEW(1)=Q0P*QOLD(1)-Q1P*QOLD(2)-Q2P*QOLD(3)-Q3P*QOLD(4)
        QNEW(2)=Q1P*QOLD(1)+Q0P*QOLD(2)+Q3P*QOLD(3)-Q2P*QOLD(4)
        QNEW(3)=Q2P*QOLD(1)-Q3P*QOLD(2)+Q0P*QOLD(3)+Q1P*QOLD(4)
        QNEW(4)=Q3P*QOLD(1)+Q2P*QOLD(2)-Q1P*QOLD(3)+Q0P*QOLD(4)
```

```
C
C       ACCUMULATION OF INPUT INCREMENTAL ANGLES
        THSUM=(J-1)*(TH1(1)+TH2(1))
C
C       PRINTING OF OUTPUT DATA
        IF(J .EQ. 1) GO TO 30
        DO 20 I=1,50
        JJ=1000*I+1
        IF(J .EQ. JJ) GO TO 30
   20   CONTINUE
        GO TO 70
   30   WRITE(6,40) THSUM,QNEW(1),QNEW(2)
   40   FORMAT(20X,E11.4,5X,3HQ0=,E25.18,5X,3HQ1=,E25.18)
        IF(J .EQ. 1) GO TO 50
        IF(J .EQ. JJ) GO TO 50
   50   WRITE(6,60) QNEW(3),QNEW(4)
   60   FORMAT(36X,3HQ2=,E25.18,5X,3HQ3=,E25.18)
C
C       GIVING OF INPUT INCREMENTAL ANGLES
   70   TH1(1)=U
        TH2(1)=V
        TH1(2)=W
        TH2(2)=X
        TH1(3)=Y
        TH2(3)=Z
C
C       STORAGE OF UPDATING QUATERNION
        QOLD(1)=QNEW(1)
        QOLD(2)=QNEW(2)
        QOLD(3)=QNEW(3)
        QOLD(4)=QNEW(4)
   10   CONTINUE
        STOP
        END
```

VITA

Wei-Ping Huang was born in Indonesia on August 9, 1947. In June 1958, his family moved to their own lovely country, Taiwan, Republic of China. He attended elementary and high school in Taipei, Taiwan. In September 1968, he entered Chung-Cheng Institute of Technology. In August 1972, he received a Bachelor of Science degree in electrical engineering and was commissioned as a teaching assistant in CCIT. Two years later, he entered the Institute of Nuclear Energy Research as an Engineer.

In June 1977, he received a government scholarship and came to the University of Tennessee as a graduate student to work his Master's degree in the department of Electrical Engineering. In August 1979, he received a Master of Science degree in Electrical Engineering from the University of Tennessee.