NAVIGATION: *Journal of The Institute of Navigation*
Vol. 37, No. 2, Summer 1990
Printed in U.S.A.

A Strapdown Inertial Algorithm Using an Earth-Fixed Cartesian Frame

M. WEI and K. P. SCHWARZ

The University of Calgary, Calgary, Alberta, Canada
Received August 1989
Revised April 1990

ABSTRACT

An algorithm for processing strapdown inertial data in an earth-fixed Cartesian frame is developed in this paper. It is compared with the standard algorithm that uses the local-level frame and the geographic coordinate system for the model formulation. A general formulation of the modeling equations for the two approaches is given, and the linearization of the equations and the formulation of the appropriate Kalman filter are outlined. The derivation of the reference gravity model for both frames is briefly discussed, and numerically efficient formulas for the model's computation are given. In the case of the Cartesian algorithm, this leads to new formulas for all three components of the gravity vector. Real data are used to compare the two algorithms and to show that the accuracy is the same in both cases, but that the Cartesian formulation is about 30 percent more efficient.

INTRODUCTION

In stable platform inertial systems, the coordinate frame for integration and filtering is essentially prescribed by the system mechanization. For strapdown inertial systems, this is not the case. The computational frame is in principle arbitrary, and its choice can be based on such criteria as speed of computation, operational ease, or required output coordinates. In navigation applications, the last criterion is usually chosen. Thus, the system of geographic coordinates is used, and computations are performed in the local-level frame or the closely related wander azimuth frame; see [1] for definitions.

This choice of the computational frame is not necessarily the best in terms of computational efficiency, even if results are needed in the geographical frame. If they are needed in a Cartesian frame, then the use of a different computational frame is not only more efficient, but also more appropriate. An application where efficiency can be gained by using a different frame is inertial surveying; see [2] for details. In this case, the vehicle in which the strapdown system is mounted stops about every 3 min to record a position. The intermediate results are not needed. The problem in this case is to find the coordinate frame in which the bulk of the computations can be performed most efficiently. The transformation of the results to the required output frame can be done as the last step because it takes very little time and has to be performed only every 3 min.

GPS/INS integration for very precise kinematic positioning applications, such as airborne photogrammetry and remote sensing, is a typical application where results are more conveniently expressed in a Cartesian frame. In this application, the spatial coordinates of the projective center of the photogrammetric bundle are determined by GPS/INS, and are then used to transform the image coordinates, measured in the aerial photographs, into spatial coordinates; see, for instance, [3] and [4]. Since the image coordinates are given in a Cartesian frame, the transformation is most easily performed if the spatial coordinates, derived from GPS/INS, are also given in a Cartesian frame. There is an additional advantage in using a Cartesian frame for INS computations in this specific case. GPS measurements are given in such a frame, and if they are directly used to update the INS-derived trajectory, the integration algorithm becomes much simpler. If needed, the transformation to geographic coordinates can always be done at the end.

It appears, therefore, that a comparison of strapdown algorithms in terms of efficiency and accuracy might be a useful exercise for some of the emerging applications, as well as for navigation in general. The paper thus compares strapdown computations in an earth-fixed Cartesian frame to the standard approach in the local-level or wander azimuth frames. Since the latter is very well documented (see, for instance, [5-7]), emphasis is on the derivation of the Cartesian algorithm, special attention being given to the reference gravity model. A numerical comparison of the two algorithms is based on a set of data recently collected in a road vehicle. The impact of these results on GPS/INS integration and general navigation is assessed.

MODELING EQUATIONS

The notation in the following sections closely follows that in [1]. Similarly, the definition of coordinate frames is the same as in [1], except for the z-axis of the local-level frame, which has been chosen as pointing upward to make it more consistent with the height definition. Subscripts and superscripts i, e, n, and b denote the operational inertial, the earth-fixed Cartesian, the local-level, and the body frames, respectively.

The measurements of an inertial strapdown unit come from two sensor triads: an accelerometer block, instrumenting three near-orthogonal axes; and a gyro assembly, instrumenting three axes, which are either aligned to the accelerometer axes or in a known relationship to them. The measurements are therefore three components of the specific force vector $\mathbf{f}^b$ and three components of the body rotation $\boldsymbol{\omega}_{ib}^b$. The body rates are measured as angular velocities with respect to the inertial frame. It is assumed that all measurements are given in body frame coordinates.

Newton's second law of motion in the gravitational field of the earth relates acceleration, specific force, and gravitation in the following way:

$$\ddot{\mathbf{r}}^i = \mathbf{f}^i + \mathbf{G}^i \tag{1}$$

where $\mathbf{r}^i$ is the position vector, $\mathbf{f}^i$ the specific force vector, and $\mathbf{G}^i$ the gravitational vector, all measured in the inertial frame.

For a rotating reference frame, equation (1) has to be transformed to take the effect of rotation into account. Expressing $\ddot{\mathbf{r}}^i$ in the earth-fixed frame leads to

$$\ddot{\mathbf{r}}^i = \mathbf{R}_e^i (\ddot{\mathbf{r}}^e + 2\,\boldsymbol{\Omega}_{ie}^e\,\dot{\mathbf{r}}^e + \boldsymbol{\Omega}_{ie}^e\,\boldsymbol{\Omega}_{ie}^e\,\mathbf{r}^e) \tag{2}$$

where $\mathbf{r}^e$ is the position vector from the center of mass of the earth; $\mathbf{R}_e^i$ is the transformation matrix from the e-frame to the i-frame; and $\boldsymbol{\Omega}_{ie}^e$ is the skew-symmetric matrix of the angular velocities of earth rotation, with $\boldsymbol{\Omega}\,\boldsymbol{\Omega}$ corresponding to $\boldsymbol{\omega} \times \boldsymbol{\omega}$ in vector notation (for more details, see [1]). The set of second-order differential equations (2) can be transformed into a set of first-order differential equations in $\dot{\mathbf{r}}^e$ and $\dot{\mathbf{v}}^e$. To take care of the changes in angular velocity, three first-order differential equations for change of the transformation matrix $\mathbf{R}_b^e$ are defined as $\dot{\mathbf{R}}_b^e$. Thus, the nonlinear mechanization equations in the earth-fixed frame are of the form

$$\begin{pmatrix} \dot{\mathbf{r}}^e \\ \dot{\mathbf{v}}^e \\ \dot{\mathbf{R}}_b^e \end{pmatrix} = \begin{pmatrix} \mathbf{v}^e \\ \mathbf{R}_b^e\,\mathbf{f}^b - 2\boldsymbol{\Omega}_{ie}^e\,\mathbf{v}^e + \mathbf{g}^e \\ \mathbf{R}_b^e\boldsymbol{\Omega}_{eb}^b \end{pmatrix} \tag{3}$$

where $\mathbf{r}^e$ is the position vector and $\mathbf{v}^e$ the velocity vector, both of them in the e-frame; $\mathbf{f}^b$ is the specific force vector in the b-frame; $\mathbf{g}^e$ is the gravity vector in the e-frame, $\mathbf{R}_b^e$ is the transformation matrix from the b-frame to the e-frame; and $\boldsymbol{\Omega}_{eb}^b$ is the skew-symmetric matrix of the angular velocity vector $\boldsymbol{\omega}_{eb}^b$ of the b-frame with respect to the e-frame, coordinated in the b-frame. For a derivation of these equations, see [8] and [9].

Figure 1 shows the algorithmic flowchart of the mechanization equation in the earth-fixed frame. In this algorithm, the integration of the raw data is done in two steps. In the first step, the body rates are integrated to obtain the transformation matrix from the b-frame to the e-frame. In the second step, the body accelerations are transformed into the computational frame, and then the velocities and coordinates of the INS are obtained by integrating the accelera-

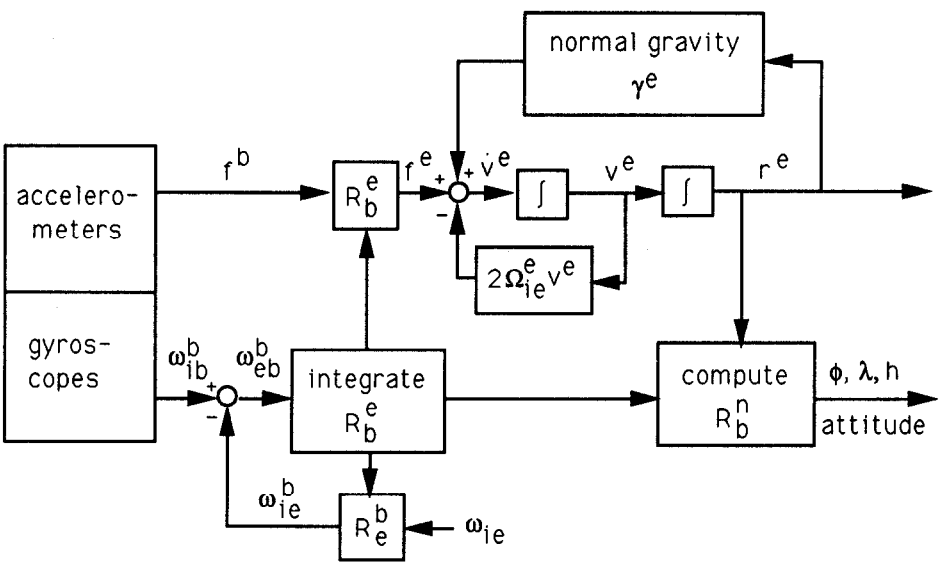

Fig. 1—Mechanization Equation in the Earth-Fixed Frame

tions. For the computation of the transformation matrix, the quaternion approach is used; see [10].

To obtain the transformation matrix, the earth rotation rate is removed from the measured rates before integration. $\mathbf{R}_b^e$ is then used to transform the specific force measurement into the e-frame. Similarly, the Coriolis acceleration and the gravity acceleration are removed from the transformed specific force measurements before integration. Equation (12) below is used for the normal gravity vector computation with feedback of the geocentric coordinates (x^e, y^e, z^e). All computations, with the exception of the gravity vector computation, are done at a 64 Hz rate. The gravity vector computation is done at a 1 Hz rate because the normal field changes rather smoothly. The algorithm outputs geocentric coordinates (x^e, y^e, z^e), earth-referenced velocities (v_x, v_y, v_z), and the transformation matrix $\mathbf{R}_b^e$ at a 64 Hz rate. Transformation to geographic coordinates (ϕ,λ,h), velocities (v_E, v_N, v_U), and Euler angles (roll, pitch, azimuth) is typically performed at a 1 Hz rate.

This algorithm has obvious advantages when results are needed in the e-frame because all outputs are directly obtained in that system. This algorithm is also simpler than the corresponding algorithm in the local-level frame because rotations due to ellipsoidal curvature do not have to be computed. It needs the input of the normal gravity vector in an earth-fixed Cartesian representation which is not the standard formulation. A detailed discussion of this problem is given in the next section.

Transforming the mechanization equation (3) into the local-level frame leads to the representation in geographic coordinates

$$\begin{pmatrix} \dot{\mathbf{r}}^n \\ \dot{\mathbf{v}}^n \\ \dot{\mathbf{R}}_b^n \end{pmatrix} = \begin{pmatrix} \mathbf{D}\mathbf{v}^n \\ \mathbf{R}_b^n\,\mathbf{f}^b - (2\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n)\,\mathbf{v}^n + \mathbf{g}^n \\ \mathbf{R}_b^n\,\boldsymbol{\Omega}_{nb}^b \end{pmatrix} \tag{4}$$

where $\mathbf{r}^n = (\phi,\lambda,h)$ are geographic coordinates, $\mathbf{v}^n = (\dot{\lambda}R_N\cos\phi, \dot{\phi}R_M, \dot{h})$ are the velocities in the local-level frame, R_M and R_N are the meridian and prime vertical radii of curvature, $\mathbf{R}_b^n$ is the transformation matrix from the b-frame to the n-frame, $\boldsymbol{\Omega}_{en}^n$ is the skew-symmetric matrix of the angular velocity vector $\boldsymbol{\omega}_{en}^n$ of the local-level frame, and $\boldsymbol{\Omega}_{nb}^b$ is the skew-symmetric matrix of the body rotation rate with respect to the local-level frame. All formulas in equation (4) refer to geographic coordinates.

Figure 2 shows the algorithmic flowchart of the mechanization equation in the local-level frame.

The basic structure of this algorithm is very similar to that of the previous one. Again, the two major steps can be clearly distinguished. In this case, the rotation rate of the n-frame due to earth rotation and body velocities is fed back to the integration loop for the Euler angles and is removed from the measured rates before integration. In the integration loop for accelerations, the Coriolis acceleration due to the rotation of the earth, the acceleration due to the body rotation with respect to the earth, and the gravity acceleration are removed from the measured specific force vector. Computation rates are as before. Equation (10) below is used for the gravity field computation. The outputs of the algorithm are geographic coodinates (ϕ,λ,h), earth-referenced velocities

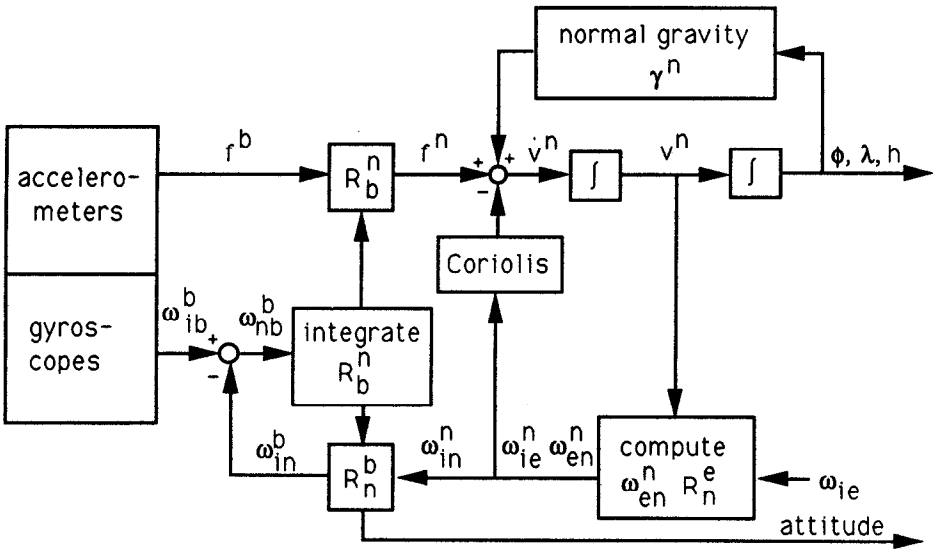

Fig. 2—Local-level Mechanization

(v_E,v_N,v_U), and attitude of the body with respect to the local-level frame (roll, pitch, azimuth) at a 64 Hz rate.

This algorithm is convenient when results and intermediate computations are required in curvilinear geographic coordinates. It also has a very simple gravity model and gives attitude directly with respect to the n-frame, i.e., as the familiar roll, pitch, and yaw. It is, however, not the most efficient algorithm because additional computations are required to eliminate the orientation changes of the local-level frame from the measurements.

Linearization of equation systems (3) and (4) about a reference trajectory leads to the standard linear dynamic model used in Kalman filtering:

$$\dot{\mathbf{x}} = \mathbf{F}\,\mathbf{x} + \mathbf{w} \tag{5}$$

where $\mathbf{x}$ is the state vector describing the trajectory perturbations, $\mathbf{F}$ is the dynamics matrix, and $\mathbf{w}$ is the driving noise. In the following computations, a state vector of the form

$$\mathbf{X}_{INS} = (\boldsymbol{\epsilon}, \delta\mathbf{r}, \delta\mathbf{v}, \mathbf{d}, \mathbf{b})^T \tag{6}$$

is used, where $\boldsymbol{\epsilon}$ contains the attitude errors, $\delta\mathbf{r}$ the position errors, $\delta\mathbf{v}$ the velocity errors, $\mathbf{d}$ the drift errors, and $\mathbf{b}$ the accelerometer biases. Each vector on the right-hand side has three components. The first three vectors are given in the e-frame or the n-frame, respectively, while the last two are given in the body frame.

The dynamics matrix corresponding to the above state vector is well known for the n-frame algorithm; see, for instance, [7]. For the e-frame, the dynamics matrix is given in Appendix A. Its derivation follows standard procedures. Implementation of the filter algorithm is straightforward in both cases.

The two algorithms outlined in Figures 1 and 2 are compared below. Before this can be done, the computation of the reference gravity vector must be briefly discussed.

NUMERICALLY EFFICIENT FORMULAS FOR THE COMPUTATION OF THE REFERENCE GRAVITY VECTOR

The gravity model used in standard inertial system mechanizations is based on an approximation of the actual gravity model of the earth, called the reference gravity model or the normal gravity model. It is derived from the gravitational potential of a rotationally symmetric ellipsoid to which the centrifugal potential due to earth rotation has been added; see [11] for details. The potential itself can be expressed by an infinite series of spherical harmonics. By differentiating the normal gravity potential with respect to the coordinates of interest, i.e., $\{\phi, \lambda, h\}$ in the case of the local-level frame and $\{x^e, y^e, z^e\}$ in the case of the Cartesian frame, the gravity vectors in the respective frames are obtained. In the first case, it can be shown that the north and east components of this vector are equal to zero if the ellipsoid chosen for the gravity vector computation coincides with the ellipsoid that defines the local-level frame. This means that the gravity vector computation can be reduced to the computation of one element, that orthogonal to the ellipsoid. In this case, a closed formula can be derived, the so-called formula of Somigliana, which expresses normal gravity γ as

$$\gamma = \frac{a\gamma_e \cos^2 \phi + b\gamma_p \sin^2 \phi}{(a^2 \cos^2 \phi + b^2 \sin^2 \phi)^{1/2}} \tag{7}$$

where a and b are the major and minor semi-axes of the ellipsoid; γ_e and γ_p are normal gravity at the equator and the pole, respectively; and ϕ is the geographic latitude. This formula is not well suited for high-speed computations. It is therefore rewritten as

$$\gamma = \gamma_e(1 + k \sin^2 \phi)(1 - e^2 \sin^2 \phi)^{-1/2} \tag{8}$$

where e is the first eccentricity of the ellipsoid, and

$$k = \frac{b\gamma_p}{a\gamma_e} - 1.$$

By expanding equation (8) into a power series with respect to e^2 and truncating it after the third term, a formula more convenient for numerical computations is obtained. It has the simple structure

$$\gamma = a_1(1 + a_2 \sin^2 \phi + a_3 \sin^4 \phi). \tag{9}$$

The coefficients a_1 to a_3 depend on the parameters of the ellipsoid, the mass and angular velocity of the earth. Details on the derivation are given in [12] and [13]. In [13], a further improvement in computational speed is achieved by expanding the sine-terms in equation (9) into series. The equation approximates normal gravity on the ellipsoid with an accuracy of 10^{-6} ms^{-2} (0.1 mGal). It is used as the standard model for local-level mechanization, and can be extended to the third dimension by adding terms that are linear and quadratic in height, i.e.,

$$\gamma = a_1(1 + a_2 \sin^2 \phi + a_3 \sin^4 \phi) + (a_4 + a_5 \sin^2 \phi)\, h + a_6 h^2 \tag{10}$$

where h is the height above the ellipsoid.

The derivation of formulas of similar accuracy and efficiency for the e-frame is obviously more involved because none of the vector elements will be equal

to zero, and no closed expressions can be obtained. The basic idea is, however, very similar. The infinite series for the gravity potential has to be differentiated with respect to the coordinates of the $\mathbf{r}^e$ vector. This is done in two steps. First, the series is differentiated with respect to the three curvilinear coordinates of the geocentric coordinate system (c) because this differentiation is much easier to perform. The resulting vector is then transformed into the Cartesian e-frame by premultiplying it by the matrix $\mathbf{R}_c^e$, which gives the transformation between the two frames. After some additional operations, the normal gravity vector in the e-frame is obtained as

$$\boldsymbol{\gamma}^e = \frac{1}{r}\begin{pmatrix}(\gamma_r - \gamma_\psi \tan\psi)x^e\\(\gamma_r - \gamma_\psi \tan\psi)y^e\\(\gamma_r + \gamma_\psi \cot\psi)z^e\end{pmatrix} \tag{11}$$

where

$$\tan\psi = \frac{z^e}{\{(x^e)^2 + (y^e)^2\}^{1/2}}.$$

For more details, see [1] or [14]. Note that in this formula, all vector elements are dependent on the components of the radius vector $\mathbf{r}^e$ and the normal gravity vector components γ_r and γ_ψ. The radius vector is directly obtained from the inertial computation (see Figure 1), while γ_r and γ_ψ are obtained by differentiating the infinite series for the normal gravity potential. The resulting formulas approximate the normal gravity vector with arbitrary accuracy. By truncating the series, approximations of a prescribed accuracy can be obtained. They have the general form

$$\boldsymbol{\gamma}^e = \frac{a_1}{r}\begin{pmatrix}\{c_1 + c_2t^2 + c_3t^4 + c_4t^6\}\,x^e\\\{c_1 + c_2t^2 + c_3t^4 + c_4t^6\}\,y^e\\\{d_1 + d_2t^2 + d_3t^4 + d_4t^6\}\,z^e\end{pmatrix} + \begin{pmatrix}\omega^2x^e\\\omega^2y^e\\0\end{pmatrix} \tag{12}$$

where

$$\begin{aligned} t &= \sin\psi \\ c_1 &= a_2 & \qquad d_1 &= a_2 + b_1 \\ c_2 &= a_3 - b_1 & \qquad d_2 &= c_2 + b_2 \\ c_3 &= a_4 - b_2 & \qquad d_3 &= c_3 + b_3 \\ c_4 &= a_5 - b_3 & \qquad d_4 &= c_4 \end{aligned}$$

and ω is the earth rotation rate.

The coefficients for these expansions are given in Appendix B, together with the numerical values for the parameters needed to compute them in some of the major internationally accepted geodetic reference systems. For details of the derivation, see [14].

The first term in equation (12) is the effect of normal gravitation, while the second term is due to centripetal acceleration. Thus, if a formula for normal gravitation is needed, as for instance for data processing in a space-stable system, the last term in formula (12) is simply omitted. It should also be noted that the two terms can be combined by replacing c by a modified coefficient $\tilde{c}_1$

$$\tilde{c}_1 = c_1 + \frac{\omega^2 r}{a_1},$$

and expressing γ^e simply by the series with modified coefficients. This shows that normal gravity and normal gravitation can be expressed by series of the same structure, and that they can be transformed into one another by simply changing one coefficient in each series.

Figure 3 shows the deviations between the rigorous value of γ, as computed by Somigliana's equation (7), and the approximation equation (12). The agreement is everywhere better than 0.5×10^{-6} m/s^2 (0.05 mGal) if three coefficients are used in each series. This corresponds to the expansion used in equation (9). If four coefficients are used, the agreement is everywhere better than 10^{-8} m/s^2 (0.001 mGal). The deviations are so small in this case that they do not show in Figure 3.

NUMERICAL TESTS

Testing of the algorithms was done in two steps. First, CPU times on a VAX 750 were determined for the three major parts of each algorithm—mechanization (integration), gravity computation, and filtering. For the Cartesian algorithm, the time needed for the transformation into the curvilinear system has also been added. The total time therefore reflects a filtered inertial system output in geographic coordinates at a 1 Hz rate. The data used were obtained from a Litton LTN 90/100 strapdown unit that outputs body rates and specific force measurements at a 64 Hz rate, after applying coning corrections at a much higher rate.

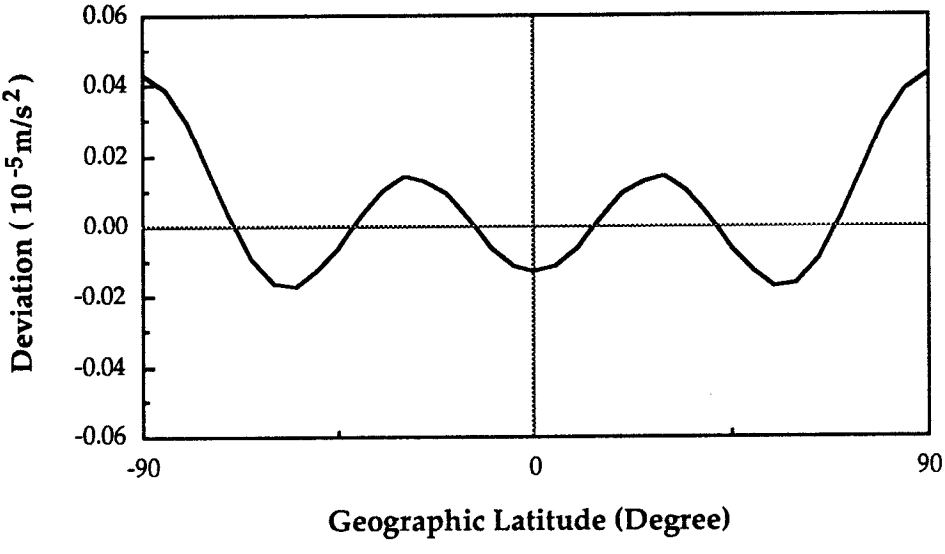

Fig. 3—Deviations Between the Rigorous Value for Normal Gravity γ and the Approximation (equations (7) and (12))

Table 1 shows the detailed comparison. Computation of the mechanization equations in the e-frame is about 40 percent faster than in the n-frame. Similarly, Kalman filter computations are about 10 percent faster in the e-frame. The normal gravity computation takes more time in the e-frame algorithm, but the total time for this component is so small that it does not affect the final balance very much. Overall, the e-frame algorithm is about 32 percent faster than the n-frame algorithm. In light of these results, integration and filtering in the e-frame seem to be preferable to that in the n-frame if an output rate between 1 and 5 Hz is acceptable.

Table 1—Computation Time for Main Software Modules

Modules	Computation Time (s)		Efficiency (%) e faster than n
	n-frame	e-frame	
Mechanization equation (64 Hz)	0.60	0.36	40%
Normal gravity computation (1 Hz)	0.00036	0.00046	−28%
Conversion to (ϕ,λ,h) (1 Hz)	/	0.0033	/
Kalman filter	0.21	0.19	10%
Total computation time	0.8104	0.5538	32%

In a second step, the correctness of the two algorithms was tested and their accuracy compared using a common data set. Measurements were collected for 40 min along a well-controlled traverse in the mountains west of Calgary. The operational procedure was the one typically used in inertial surveying, with stops for zero velocity updates (ZUPT) about every 3 min. In this case, all update points had been determined by GPS beforehand and had a relative accuracy of 5 cm (RMS). They were taken as ground truth to which the filtered results were compared. Since only the differences between the two algorithms were of interest and not their absolute accuracy, the coordinates at the first and sixth points were used for coordinate updates. The filtered coordinates, using zero velocity updates, are compared to the known coordinates at the control points.

Figure 4 shows the coordinate errors after filtering for both methods. In general, the agreement of the two algorithms is quite good. The deviations of both methods are well within the system noise. The differences to the control points show the same behavior for both algorithms and are of the expected order of magnitude for this class of system. In general, results from the e-frame computation are more accurate. However, the sample size is so small that the significance of this difference cannot be confirmed statistically.

Figures 5 to 7 show the agreement between the two algorithms in kinematic mode. The truth values were in this case determined by differential GPS using carrier phase measurements. Control values are available every 4 s. Their accuracy is estimated to be 10 cm (RMS) in each coordinate. Shown are differences between INS Kalman filter results and GPS results. Only ZUPT have been used for the INS at the control points; no GPS coordinate updates have been made. The filter predictions show the typical sawtooth pattern between updates and are in the expected range. The results of the two algorithms are very close. To test whether the overall accuracy is the same, the RMS values of the INS errors in {ϕ,λ,h} have been computed at 87 control values. They are

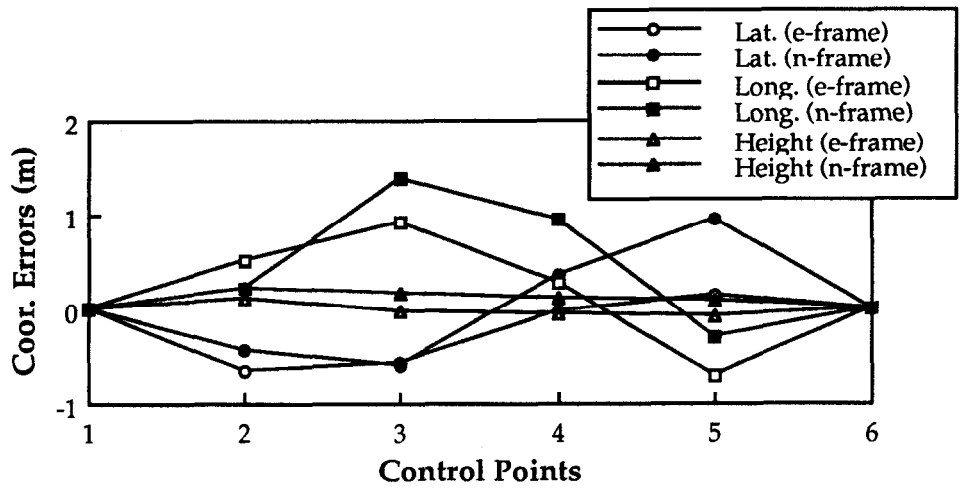

Fig. 4—Errors of Filtered Coordinates Using ZUPT at Points 2 to 5

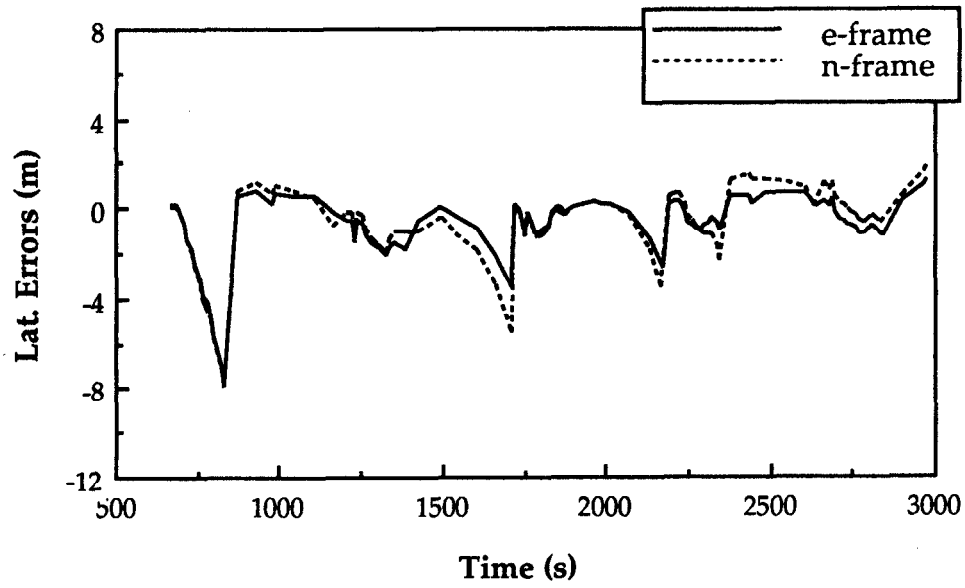

Fig. 5—Errors of Predicted INS Latitude Compared with GPS

{±1.682, ±1.573, and ±0.350 (m)} for the n-frame computation, and {±1.595, ±1.428, and ±0.314 (m)} for the e-frame computation. The difference in the RMS values is significant for ϕ and λ based on the accuracy of the GPS-derived control coordinates. This indicates that the Cartesian algorithm (e-frame) has better overall accuracy. This result has still to be confirmed for a longer data span (several hours) and for other dynamic environments.

CONCLUSIONS

Two algorithms for strapdown inertial data have been compared in terms of computational efficiency and accuracy. One is the standard local-level algo-

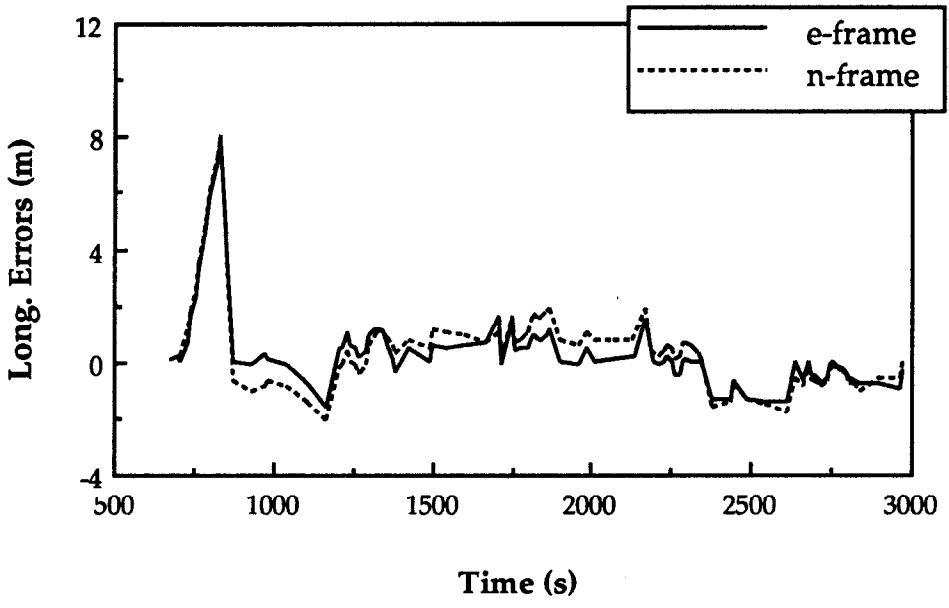

Fig. 6—Errors of Predicted INS Longitude Compared with GPS

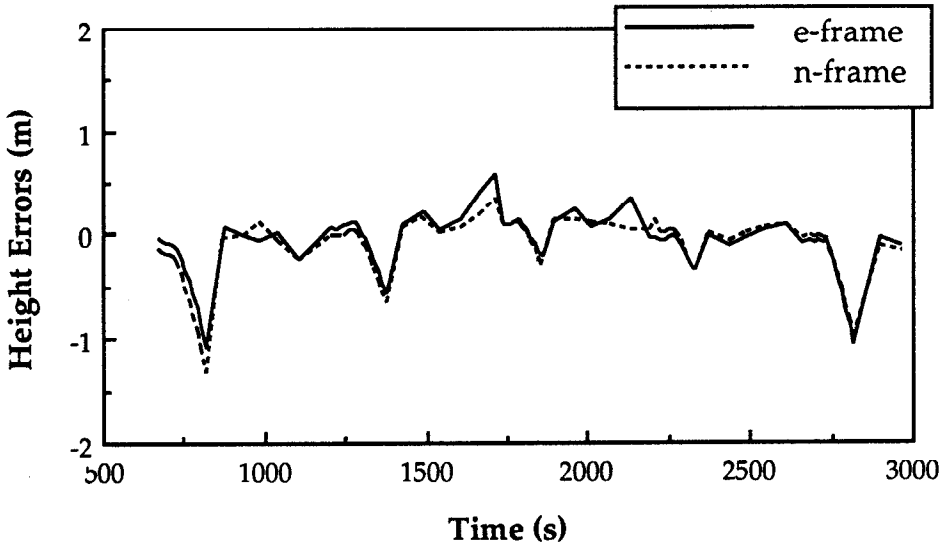

Fig. 7—Errors of Predicted INS Height Compared with GPS

rithm; the other is a proposed alternative for processing strapdown data in an earth-fixed Cartesian frame. An analysis of the algorithmic structure and of results of test computations leads to the following conclusions.

The proposed Cartesian algorithm is about 30 percent more efficient than the standard algorithm. It should therefore always be used if the output is required in three-dimensional Cartesian coordinates, as for instance in some of

the emerging precise positioning applications requiring GPS/INS integration. Even if the output is needed in geographic coordinates, the Cartesian algorithm is 30 percent more efficient as long as an output rate of 5 Hz is sufficient. The higher efficiency is due mainly to the elimination of the angular velocity computations, which rotate the measurements from the Cartesian to the local-level frame.

New equations are given for the fast computation of normal gravity in the Cartesian frame. Comparisons show that CPU times for the gravity computation are similar in both algorithms and do not affect the overall computation time in a major way.

The tests confirm that the theoretical equivalence of the two algorithms has been implemented with sufficient accuracy. Results of the two computations are very close considering the differences in computational procedure and filter formulation. The outputs show the same systematic trend with respect to known ground truth, which indicates that both algorithms apply the same mathematical operation to the measurements. A statistical analysis of the results indicates that the Cartesian algorithm gives slightly better results and that, from a statistical point of view, this difference is significant. This result has still to be confirmed for mission times of several hours.

ACKNOWLEDGMENTS

Partial financial support for this work has been obtained from a National Science and Engineering Research Council Operating Grant of the second author and from a research contract with Alberta Transport. Mr. H. Martell provided the program for integration and filtering in the local-level frame and assisted in running it. This help is gratefully acknowledged.

REFERENCES

1. Britting, K. R., *Inertial Navigation Systems Analysis,* Wiley-Interscience, New York, 1971.
2. Schwarz, K. P., *Inertial Surveying and Geodesy,* Rev. of Geoph. and Space Phys., 21, 1983, pp. 878–90.
3. Schwarz, K. P., Fraser, G. S., and Gustafson, P. C., *Aerotriangulation Without Ground Control,* International Archives of Photogrammetry and Remote Sensing, 25, Part A1, Rio de Janeiro, June 16–29, 1984.
4. Mader, G., *Dynamic Postioning Using GPS Carrier Phase Measurements,* Manuscripta Geodaetica, Vol. 11, No. 4, 1986, pp. 272–77.
5. Van Bronkhorst, A., *Strapdown System Algorithms,* AGARD Lecture Series No. 95, Neuilly Sur Seine, 1978.
6. Savage, P. G., *Strapdown System Algorithms,* AGARD Lecture Series No. 133, Neuilly Sur Seine, 1984.
7. Wong, R. V. C., *Development of a RLG Strapdown Inertial Survey System,* UCSE Reports No. 20027, Department of Surveying Engineering, The University of Calgary, Calgary, Canada, 1988.
8. Schwarz, K. P. and Wei, M., *A Framework for Modelling Kinematic Measurements in Gravity Field Applications,* Proc. AGU Chapman Conference on Progress in the Determination of the Earth's Gravity Field, Ft. Lauderdale, September 12–16, 1988.
9. Teunissen, P. G., *Strapdown INS,* Internal Report, Department of Surveying Engineering, The University of Calgary, Calgary, 1988.

10. Grubin, C., *Derivation of Quaternion Scheme via the Euler Axis and Angle. Engineering Notes,* Journal of Spacecraft, Vol. 7, No. 10, 1970, pp. 1261–63.
11. Heiskanen, W. A. and Moritz, H., *Physical Geodesy,* W. H. Freeman, San Francisco, 1967.
12. Mittermayer, E., *Numerical Formulas for the Geodetic Reference System 1967,* Bolletino di Geofisica, Vol. XI, 1969, pp. 96–107.
13. Nagy, D., *Direct Gravity Formula for the Geodetic Reference System 1967,* Bulletin Géodésique, Vol. 52, No. 2, 1978, pp. 159–64.
14. Schwarz, K. P. and Wei, M., *Efficient Numerical Formulas for the Computation of Normal Gravity in a Cartesian Frame,* submitted to Manuscripta Geodaetica, 1990.

APPENDIX A

The dynamics matrix of the 15-state Kalman filter in the earth-fixed frame corresponding to equation (5) is of the form

$$\begin{pmatrix}
0 & \omega_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{11} & R_{12} & R_{13} & 0 & 0 & 0 \\
-\omega_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{21} & R_{22} & R_{23} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{31} & R_{32} & R_{33} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & f_z & -f_y & N_{11} & N_{12} & N_{13} & 0 & 2\omega_e & 0 & 0 & 0 & 0 & R_{11} & R_{12} & R_{13} \\
-f_z & 0 & f_x & N_{21} & N_{22} & N_{23} & -2\omega_e & 0 & 0 & 0 & 0 & 0 & R_{21} & R_{22} & R_{23} \\
f_y & -f_x & 0 & N_{31} & N_{32} & N_{33} & 0 & 0 & 0 & 0 & 0 & 0 & R_{31} & R_{32} & R_{33} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\zeta & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\zeta & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\zeta & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\beta & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\beta & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\beta
\end{pmatrix}$$

where

ω_e is the earth rotation rate,
f_x, f_y, f_z are components of the specific force in the earth-fixed frame,
R_{ij} are elements of the transformation matrix $\mathbf{R}_b^e$,
N_{ij} are elements of the coefficient matrix $\mathbf{N}_G$ for the normal gravity error,
ζ is 1/(correlation time) of the first-order Gauss-Markov process describing the gyro drifts, and
β is 1/(correlation time) of the first-order Gauss-Markov process describing the accelerometer biases.

APPENDIX B

The coefficients a_j and b_j are:

$$a_1 = -\frac{GM}{r_2}$$

$$a_2 = 1 + \frac{3}{2}J_2\left(\frac{a}{r}\right)^2 - \frac{15}{8}J_4\left(\frac{a}{r}\right)^4 + \frac{35}{16}J_6\left(\frac{a}{r}\right)^6$$

$$a_3 = -\frac{9}{2}J_2\left(\frac{a}{r}\right)^2 + \frac{75}{4}J_4\left(\frac{a}{r}\right)^4 - \frac{735}{16}J_6\left(\frac{a}{r}\right)^6$$

$$a_4 = -\frac{175}{8}J_4\left(\frac{a}{r}\right)^4 + \frac{2205}{16}J_6\left(\frac{a}{r}\right)^6$$

$$a_5 = -\frac{1617}{16}J_6\left(\frac{a}{r}\right)^6$$

$$b_1 = 3J_2\left(\frac{a}{r}\right)^2 - \frac{15}{2}J_4\left(\frac{a}{r}\right)^4 + \frac{105}{8}J_6\left(\frac{a}{r}\right)^6$$

$$b_2 = \frac{35}{2}J_4\left(\frac{a}{r}\right)^4 - \frac{945}{12}J_6\left(\frac{a}{r}\right)^6$$

$$b_3 = \frac{693}{8}J_6\left(\frac{a}{r}\right)^6$$

The coefficients c_j and d_j are:

$$c_1 = 1 + \frac{3}{2}J_2\left(\frac{a}{r}\right)^2 - \frac{15}{8}J_4\left(\frac{a}{r}\right)^4 + \frac{35}{16}J_6\left(\frac{a}{r}\right)^6$$

$$c_2 = -\frac{15}{2}J_2\left(\frac{a}{r}\right)^2 + \frac{105}{4}J_4\left(\frac{a}{r}\right)^4 - \frac{945}{16}J_6\left(\frac{a}{r}\right)^6$$

$$c_3 = -\frac{315}{8}J_4\left(\frac{a}{r}\right)^4 + \frac{10395}{48}J_6\left(\frac{a}{r}\right)^6$$

$$c_4 = -\frac{9009}{48}J_6\left(\frac{a}{r}\right)^6$$

$$d_1 = 1 + \frac{9}{2}J_2\left(\frac{a}{r}\right)^2 - \frac{75}{8}J_4\left(\frac{a}{r}\right)^4 + \frac{245}{16}J_6\left(\frac{a}{r}\right)^6$$

$$d_2 = -\frac{15}{2} J_2 \left(\frac{a}{r}\right)^2 + \frac{175}{4} J_4 \left(\frac{a}{r}\right)^4 - \frac{6615}{48} J_6 \left(\frac{a}{r}\right)^6$$

$$d_3 = -\frac{315}{8} J_4 \left(\frac{a}{r}\right)^4 + \frac{14553}{48} J_6 \left(\frac{a}{r}\right)^6$$

$$d_4 = -\frac{9009}{48} J_6 \left(\frac{a}{r}\right)^6$$

Table B-1 gives the numerical values of the parameters needed to compute the coefficients c_j and d_j for the three reference systems officially adopted by the International Association of Geodesy. More details can be found in International Association of Geodesy: Geodetic Reference System 1967, Publ. Spec. No. 3 du Bulletin Géodésique, Paris, 1971; and The Geodesist's Handbook 1988, Bull. Géod., Vol. 62, No. 3, 1988.

Table B-1—Parameters to Compute Cj and dj

Parameter	International	GRS 67	GRS 80
a (m)	6378338.0	6378160.0	6378137.0
GM ($m^3\ s^{-2}$)	$3986329 \cdot 10^8$	$3986030 \cdot 10^8$	$3986005 \cdot 10^8$
J_2	0.0010920	0.0010827	0.00108263
J_4	$-2.43 \cdot 10^{-6}$	$-2.37126440 \cdot 10^{-6}$	$-2.37091222 \cdot 10^{-6}$
J_6	$6.31571 \cdot 10^{-9}$	$6.08516 \cdot 10^{-9}$	$6.08347 \cdot 10^{-9}$
w(s^{-1})	$7.29211511 \cdot 10^{-5}$	$7.2921151467 \cdot 10^{-5}$	$7.292115 \cdot 10^{-5}$