JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 21, No. 2, March–April 1998

Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms

Paul G. Savage*
Strapdown Associates, Inc., Maple Plain, Minnesota 55359

This series of two papers (Parts 1 and 2) provides a rigorous comprehensive approach to the design of the principal software algorithms utilized in modern-day strapdown inertial navigation systems: integration of angular rate into attitude, acceleration transformation/integration into velocity, and integration of velocity into position. The algorithms are structured utilizing the two-speed updating approach originally developed for attitude updating; an analytically exact equation is used at moderate speed to update the integration parameter (attitude, velocity, or position) with input provided from a high-speed algorithm measuring rectified dynamic motion within the parameter update time interval [coning for attitude updating, sculling for velocity updating, and scrolling (writer's terminology) for high-resolution position updating]. The algorithm design approach accounts for angular rate/specific force acceleration inputs from the strapdown system inertial sensors, as well as rotation of the navigation frame used for attitude referencing and velocity integration. The Part 1 paper (Savage, P. G., "Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms," ***Journal of Guidance, Control, and Dynamics*****, Vol. 21, No. 1, 1998, pp. 19–28) defined the overall design requirement for the strapdown inertial navigation integration function and developed the attitude updating algorithms. This paper, Part 2, deals with design of the acceleration transformation/velocity integration and position integration algorithms. Although Parts 1 and 2 often cover basic concepts, the material presented is intended for use by the practitioner who is already familiar with inertial navigation fundamentals.**

Nomenclature

A, A_1, A_2 = arbitrary coordinate frames

$\boldsymbol{a}_{\mathrm{SF}}$ = specific force defined as the acceleration relative to nonrotating inertial space produced by applied nongravitational forces, measured by accelerometers

$C_{A_2}^{A_1}$ = direction cosine matrix that transforms a vector from its A_2 frame projection form to its A_1 frame projection form

I = identity matrix

$\boldsymbol{V}^A$ = column matrix with elements equal to the projection of vector $\boldsymbol{V}$ on frame A axes

$(\boldsymbol{V}^A \times)$ = skew symmetric (or cross product) form of $\boldsymbol{V}^A$ represented by the square matrix

$$\begin{bmatrix} 0 & -V_{ZA} & V_{YA} \\ V_{ZA} & 0 & -V_{XA} \\ -V_{YA} & V_{XA} & 0 \end{bmatrix}$$

where V_{XA}, V_{YA}, V_{ZA} are the components of $\boldsymbol{V}^A$; matrix product of $(\boldsymbol{V}^A \times)$ with another A frame vector equals the cross product of $\boldsymbol{V}^A$ with the vector in the A frame

$\boldsymbol{\omega}_{A_1 A_2}$ = angular rate of coordinate frame A_2 relative to coordinate frame A_1; when A_1 is the inertial I frame, $\boldsymbol{\omega}_{A_1 A_2}$ is the angular rate measured by angular rate sensors mounted on frame A_2

I. Introduction

A STRAPDOWN inertial navigation system (INS) is typically composed of an orthogonal three-axis set of inertial angular rate sensors and accelerometers providing data to the INS computer. The inertial sensors are directly mounted (strapdown) to the INS chassis structure in contrast with original INS technology that utilized an active multiaxis gimbal isolation mounting assembly to isolate the sensors from rotation. The principal software functions executed in the strapdown INS computer are the integration of sensed angular rate into attitude, transformation of accelerometer sensed specific force acceleration into a navigation coordinate frame, addition of software modeled gravity to the transformed specific force to calculate total acceleration, and double integration of total acceleration into velocity and position. The key element in the INS software design process is the development of repetitive digital algorithms that will flawlessly execute the attitude, velocity, and position digital integration functions in the presence of dynamic angular rate/specific force acceleration inputs.

As discussed in Part 1 (Ref. 1), most modern-day strapdown INSs utilize attitude updating algorithms based on a two-speed approach[2–4]: a higher-order updating algorithm is processed at moderate repetition rate using inputs from a high-speed algorithm. The moderate-speed routine can be represented by an exact closed-form attitude updating operation.[3,4] The high-speed algorithm is designed to accurately account for multiaxis high-frequency angular motion between moderate speed algorithm updates that can rectify into systematic attitude change (traditionally denoted as coning). Originally conceived as a simple first-order algorithm,[2] today's high-speed attitude algorithms have taken advantage of increased throughput capabilities in modern-day computers and become higher order for improved accuracy (Refs. 1; 5–7; and 8, Chap. 7). While the attitude updating function has been evolving to its current form, very little parallel work has been published on the development of the companion strapdown INS algorithms for specific force acceleration transformation/velocity integration and position integration, the subject of this paper.

The specific force transformation algorithm processes the inertial sensor data to calculate an integrated specific force increment in navigation coordinates over the velocity algorithm update time interval. The velocity is updated by adding the navigation frame specific force increment (plus an increment for gravity and coordinate frame rotation effects) to the previous velocity value. A key function of the transformation algorithm is to accurately account for attitude rotation (hence, rotation of the strapdown accelerometers) during the velocity update time period. In some applications, this has been achieved using a centering algorithm[9] in which attitude data for the specific force transformation is updated at the center of the velocity update time interval (thereby introducing a staggered

Received July 7, 1997; revision received Oct. 9, 1997; accepted for publication Oct. 9, 1997.

*President. Member AIAA.

attitude update/velocity update software architecture). The transformation operation then consists of integrating the accelerometer specific force output over the velocity update interval and transforming the integrated specific force increment to the navigation frame using attitude data at the center of the velocity update time interval. A variation of the latter approach updates the attitude at twice the velocity update rate so that the attitude solution between velocity updates is available for specific force increment transformation. Another variation calculates the attitude used for specific force transformation as the average of the computed attitude at the start and end of the velocity update time interval. A two-speed approach can also be used for specific force transformation/velocity integration in a dynamic environment that parallels the two-speed attitude integration approach (Refs. 5 and 8, Sec. 7.2). A high-speed algorithm is designed to account for high-frequency angular and linear oscillations that can rectify into systematic velocity buildup (traditionally denoted as sculling), and a moderate-speed algorithm executes the specific force transformation based on inputs from the high-speed algorithm.

In general, the specific force transformation/velocity integration algorithms have lacked the analytical sophistication of the attitude integration algorithms, being typically limited to first-order accuracy under maneuvering conditions. Virtually no specialized work has been reported for the inertial navigation position integration function. From the writer's understanding, modern-day strapdown INSs typically generate position as a simple trapezoidal integration of velocity at an update rate equal to or lower than the velocity update frequency. For applications requiring precise position change data in a dynamic environment, such a rudimentary approach to position integration may prove inadequate.

This paper provides a comprehensive process for the design of strapdown inertial navigation specific force transformation, velocity integration, and position integration algorithms. The material presented is a condensed version of Ref. 8, Secs. 7.2 and 7.3 (an expansion of material in Ref. 5), emphasizing a more rigorous analytical formulation and the use of exact closed-form equations where possible for ease in computer software documentation/validation. The velocity and position algorithms presented are structured using a two-speed computation format; the moderate-speed algorithm, e.g., 50–200 Hz, is designed to be exact under constant angular rate/specific force acceleration conditions during the moderate-speed update interval; the moderate-speed algorithm is fed by a high-speed computation algorithm, e.g., 1–4 kHz, that accounts for dynamic variations from constant angular rate/specific force [sculling for the velocity algorithm and scrolling (writer's terminology) for the position algorithm]. Included is a rigorous treatment of navigation coordinate frame rotation during the integration update time periods.

This paper is organized as follows. Section II defines the coordinate frames utilized. Section III utilizes the Part 1 (Ref. 1) attitude algorithm derivation as a model to formulate two-speed specific force acceleration transformation/velocity integration algorithms. Section IV then uses Sec. III as a framework for the development of position updating algorithms in two forms: a traditional form based on trapezoidal integration and a two-speed high-resolution form. A tabular reference summary of the derived algorithms is presented in Sec. V. Section VI provides a general discussion of the process followed in selecting algorithms for a particular application and establishing their execution rates. Concluding remarks are provided in Sec. VII.

Finally, it is important to recognize that, whereas the original intent of the two-speed approach was to overcome throughput limitations of early computer technology (1965–1975), that limitation is rapidly becoming insignificant with continuing rapid advances in modern high-speed computers. This provides the motivation to eventually return to a simpler single-speed algorithm structure whereby all computations are executed at a repetition rate that is sufficiently high to accurately account for multiaxis high-frequency angular rate and specific force acceleration rectification effects. The two-speed structure presented in this paper and in Part 1 (Ref. 1) is compatible with compression into such a single-speed format as explained in the particular sections where the algorithms are formulated.

II. Coordinate Frames

A coordinate frame is an analytical abstraction defined by three consecutively numbered (or lettered) unit vectors that are mutually perpendicular to one another in the right-hand sense. It can be visualized as a set of three perpendicular lines (axes) passing through a common point (origin) with the unit vectors emanating from the origin along the axes. In this paper, the physical locations of the coordinate frame origins are arbitrary. A vector's components (or projections) in a particular coordinate frame equal the dot product of the vector with the coordinate frame unit vectors. The vectors used in this paper are classified as free vectors and, hence, have no preferred location in coordinate frames in which they are analytically described.

The coordinate frames are defined as follows.

1) The E frame is the Earth-fixed coordinate frame used for position location definition. It is typically defined with one axis parallel to the Earth polar axis and with the other axes fixed to the Earth and parallel to the equatorial plane.

2) The N frame is the navigation coordinate frame having its Z axis parallel to the upward vertical at the local Earth surface referenced position location. It is used for integrating acceleration into velocity and for defining the angular orientation of the local vertical in the E frame.

3) The L frame is the locally level coordinate frame parallel to the N frame but with the Z axis parallel to the downward vertical and X and Y along N frame Y and X axes. It is used as the reference for describing the strapdown sensor coordinate frame orientation.

4) The B frame is the strapdown inertial sensor coordinate frame (body frame) with axes parallel to nominal right-handed orthogonal sensor input axes.

5) The I frame is the nonrotating inertial coordinate frame used as a reference for angular rate measurements. Particular orientations selected for the I frame are discussed in the sections where its orientation is pertinent to analytical operations.

III. Velocity Update Algorithms

In this section we develop algorithms for integrating the Ref. 1, Eq. (20), velocity rate equation using Ref. 1, Eqs. (16) and (18), for the specific force transformation term and using angular rates from Ref. 1, Eqs. (14) and (15), in the Coriolis acceleration term (angular rate products with velocity):

$$\dot{\boldsymbol{v}}^N = C_L^N C_B^L \boldsymbol{a}_{\mathrm{SF}}^B + \boldsymbol{g}_P^N - \left(\boldsymbol{\omega}_{EN}^N + 2\boldsymbol{\omega}_{IE}^N\right) \times \boldsymbol{v}^N \tag{1}$$

$$\boldsymbol{\omega}_{IE}^N = \left(C_N^E\right)^T \boldsymbol{\omega}_{IE}^E \tag{2}$$

$$\boldsymbol{\omega}_{EN}^N = F_C\left(\boldsymbol{u}_{ZN}^N \times \boldsymbol{v}^N\right) + \rho_{ZN}\boldsymbol{u}_{ZN}^N \tag{3}$$

where $\boldsymbol{v}$ is the velocity relative to the Earth defined analytically as the time derivative in the E frame of the position vector from Earth's center to the INS, and $\boldsymbol{g}_P$ is plumb-bob gravity (or gravity) that, for a stationary INS, lies along the line of a plumb bob. F_C is a curvature matrix (3×3) that is a function of position having elements $3,i$ and $i,3$ equal to zero and the remaining elements symmetrical about the diagonal. For a spherical Earth model, the remaining elements of F_C are zero off the diagonal and equal the reciprocal of the radial distance from the Earth's center to the INS on the diagonal. For an oblate Earth model, the remaining F_C terms represent the local curvature on the Earth's surface projected to the INS altitude (see Ref. 8, Sec. 5.3, for closed-form expression). ρ_{ZN} is the vertical component of $\boldsymbol{\omega}_{EN}^N$. The value selected for ρ_{ZN} depends on the type of N frame utilized, e.g., wander azimuth or free azimuth designed to assure that $\boldsymbol{\omega}_{EN}^N$ is nonsingular for all Earth locations (see Ref. 8, Sec. 4.6, and Ref. 10, pp. 88–89). $\boldsymbol{u}_{ZN}$ is a unit vector upward along the geodetic vertical (the Z axis of the N frame).

Equation (1) uses direction cosine matrix transformed specific force rather than the alternative Ref. 1, Eq. (17), quaternion transformation approach, e.g., for situations where the B frame attitude is computed in the form of an attitude quaternion. The velocity integration algorithm based on quaternion specific force transformation can be developed by extension of the results presented here.

The digital velocity integration algorithm is formulated directly from Eq. (1) as

$$v_m^N = v_{m-1}^N + C_L^N \Delta v_{\mathrm{SF}_m}^L + \Delta v_{G/\mathrm{Cor}_m}^N \tag{4}$$

$$\Delta v_{G/\mathrm{Cor}_m}^N = \int_{t_{m-1}}^{t_m} \left[g_P^N - \left(\omega_{EN}^N + 2\omega_{IE}^N \right) \times v^N \right] \mathrm{d}t \tag{5}$$

$$\Delta v_{\mathrm{SF}_m}^L = \int_{t_{m-1}}^{t_m} C_B^L a_{\mathrm{SF}}^B \, \mathrm{d}t \tag{6}$$

where m is the digital velocity integration algorithm update rate computer cycle index.

If vertical channel gravity/divergence stabilization is to be incorporated, an additional update operation would be included in Eq. (4) representing the vertical velocity control function (Ref. 8, Sec. 4.4.1, and Ref. 10, pp. 102–103).

Digital algorithms are formulated next for the gravity/Coriolis velocity increment $\Delta v_{G/\mathrm{Cor}_m}^N$ in Eq. (5) and the integrated transformed specific force increment $\Delta v_{\mathrm{SF}_m}^L$ in Eq. (6).

A. Gravity/Coriolis Velocity Increment

The g_P^N term in Eq. (5) is a function of position location with very small horizontal components. Because the position varies smoothly over a digital algorithm m cycle with limited magnitude change (particularly in altitude), g_P^N in Eq. (5) can be approximated by its average value across the m cycle. Because the Eq. (5) Coriolis term is small (due to the small size of the angular rates) and because velocity varies smoothly over an m cycle, the Coriolis contributors can also be approximated by their average value over the m cycle. The latter rationale forms the basis for the following algorithm for $\Delta v_{G/\mathrm{Cor}_m}^N$ in Eq. (5) using Eq. (3) for ω_{EN}^N:

$$\Delta v_{G/\mathrm{Cor}_m}^N \approx \Bigg\{ g_{P_{m-\frac{1}{2}}}^N - \bigg[2\omega_{IE_{m-\frac{1}{2}}}^N + \rho_{ZN_{m-\frac{1}{2}}} u_{ZN}^N + F_{C_{m-\frac{1}{2}}} \left(u_{ZN}^N \times v_{m-\frac{1}{2}}^N \right) \bigg] \times v_{m-\frac{1}{2}}^N \Bigg\} T_m \tag{7}$$

where $m - \frac{1}{2}$ designates the parameter value midway between t_{m-1} and t_m, and T_m is the velocity integration algorithm update period $t_m - t_{m-1}$.

The ω_{IE}^N term in Eq. (7) is evaluated with Eq. (2), and g_P^N is calculated from Ref. 1, Eq. (19). Because $\Delta v_{G/\mathrm{Cor}_m}^N$ is used in Eq. (4) to update v^N from its $m-1$ to m cycle value, $v_{m-1/2}^N$ is not explicitly available for Eq. (7) and must be approximated based on extrapolation from past values. An example is the linear extrapolation algorithm

$$v_{m-\frac{1}{2}}^N \approx v_{m-1}^N + \tfrac{1}{2}\left[v_{m-1}^N - v_{m-2}^N \right] = \tfrac{3}{2} v_{m-1}^N - \tfrac{1}{2} v_{m-2}^N \tag{8}$$

The g_P^N, ω_{IE}^N, ρ_{ZN}, and F_C parameters in Eq. (7) are functions of position, which (from Sec. IV.A) is updated following the velocity update, possibly at a slower n cycle repetition rate, e.g., five times slower. Therefore, the designated $m - \frac{1}{2}$ value for these parameters is not explicitly available and must also be approximated based on extrapolation from past values. For example, for linear extrapolation

$$(\,)_{m-\frac{1}{2}} \approx (\,)_{n-1} + \frac{\left(r - \frac{1}{2}\right)}{j} [(\,)_{n-1} - (\,)_{n-2}] \tag{9}$$

where

n = computer cycle index for position updates
j = number of m cycles in each n cycle
r = number of m cycles since last n cycle, i.e., since t_{n-1}

B. Integrated Transformed Specific Force Increment

A digital algorithm for integrated transformed specific force increment equation (6) must account for rotation of the local level L frame and the strapdown sensor body B frame during the t_{m-1} to t_m computer cycle period. Adopting the same notation used in Ref. 1, Sec. IV.A, to describe discrete orientations of the L and B frames relative to inertial space I at computer update time instants, Eq. (6) can be expanded using the Ref. 1, Eq. (3), chain rule as follows:

$$\Delta v_{\mathrm{SF}_m}^L = \int_{t_{m-1}}^{t_m} C_{L_{I(n-1)}}^{L_{I(m)}} C_{B_{I(m-1)}}^{L_{I(n-1)}} C_{B_{(t)}}^{B_{I(m-1)}} a_{\mathrm{SF}}^B \, \mathrm{d}t \tag{10}$$

or, on further expansion,

$$\Delta v_{\mathrm{SF}_m}^{L_{I(n-1)}} = C_{B_{I(m-1)}}^{L_{I(n-1)}} \Delta v_{\mathrm{SF}_m}^{B_{I(m-1)}} \tag{11}$$

$$\Delta v_{\mathrm{SF}_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} C_{B_{(t)}}^{B_{I(m-1)}} a_{\mathrm{SF}}^B \, \mathrm{d}t \tag{12}$$

$$\Delta v_{\mathrm{SF}_m}^L = C_{L_{I(n-1)}}^{L_{I(m)}} \Delta v_{\mathrm{SF}_m}^{L_{I(n-1)}} = \Delta v_{\mathrm{SF}_m}^{L_{I(n-1)}} + \left(C_{L_{I(n-1)}}^{L_{I(m)}} - \mathrm{I} \right) \Delta v_{\mathrm{SF}_m}^{L_{I(n-1)}} \tag{13}$$

Equations (11–13) allow for the general case whereby the C_B^L matrix is updated for L frame rotation at a cycle rate (index n) that may differ from (be slower than) the C_B^L update rate for B frame rotation (index m). For example, in the interest of minimizing computer throughput requirements, the software architecture might have the n cycle L frame update rate set five times slower than the m cycle B frame update rate. Equations (11–13) are also valid, however, if we choose to update C_B^L at equal rates for B and L frame motion, i.e., $n = m$. Note that, for $n \neq m$, Eq. (13) still requires an L frame orientation evaluation at the B frame m cycle update time (for $L_{I(m)}$ in the $C_{L_{I(n-1)}}^{L_{I(m)}}$ matrix). Note also that the form of Eq. (11) is based on the use of C_B^L at the preceding B frame m cycle, i.e., $B_{I(m-1)}$ in the $C_{B_{I(m-1)}}^{L_{I(n-1)}}$ matrix. This implies that C_B^L will be updated for B frame rotation following the Eq. (11) transformation operation. It remains to define algorithms for $C_{L_{I(n-1)}}^{L_{I(m)}}$ in Eq. (13) to account for local level frame rotation during specific force transformation and for the $\Delta v_{\mathrm{SF}_m}^{B_{I(m-1)}}$ body frame integrated specific force increment term in Eq. (12).

1. *Correction for Local Level Frame Rotation During Specific Force Transformation*

Because of the slow angular rate of the L frame relative to inertial space, $C_{L_{I(n-1)}}^{L_{I(m)}}$ in Eq. (13) is very close to the identity matrix I. For many applications, $(C_{L_{I(n-1)}}^{L_{I(m)}} - \mathrm{I})$ in Eq. (13) can, therefore, be totally ignored as negligible compared to other acceleration error sources. For high-accuracy applications where $(C_{L_{I(n-1)}}^{L_{I(m)}} - \mathrm{I})$ is to be included, a first-order form of the Ref. 1 Eqs. (49) and (50) usually suffices, whereby

$$C_{L_{I(n-1)}}^{L_{I(m)}} \approx \mathrm{I} - (\zeta_{n-1,m} \times) \tag{14}$$

$$\zeta_{n-1,m} = \int_{t_{n-1}}^{t_m} \omega_{IL}^L \, \mathrm{d}t \tag{15}$$

We then approximate ω_{IL}^L in Eq. (15) using Eq. (3) in Ref. 1, Eq. (13), and the assumption of slowly changing contributors as in Sec. III.A,

$$\omega_{IL}^L = C_N^L \left(\omega_{IE}^N + \omega_{EN}^N \right) \approx C_N^L \left[\omega_{IE_{n-1,m}}^N + \rho_{ZN_{n-1,m}} u_{ZN}^N + F_{C_{n-1,m}} \left(u_{ZN}^N \times v^N \right) \right] \tag{16}$$

where the subscript $n-1$, m indicates the value for the parameter midway between times t_{n-1} and t_m.

Substituting Eq. (16) into Eq. (15) yields

$$\zeta_{n-1,m} \approx C_N^L \left[\omega_{IE_{n-1,m}}^N r T_m + \rho_{ZN_{n-1,m}} u_{ZN}^N r T_m + F_{C_{n-1,m}} \left(u_{ZN}^N \times \Delta R_{n-1,m}^N \right) \right] \tag{17}$$

$$\Delta R_{n-1,m}^N \equiv \int_{t_{n-1}}^{t_m} v^N \, \mathrm{d}t \tag{18}$$

The ω^N_{IE} term in Eq. (17) is evaluated with Eq. (2). As in Sec. III.A, $()_{n-1,m}$ in Eq. (17) must be approximated based on past value extrapolation; e.g.,

$$()_{n-1,m} \approx \left(()_{n-1} + \tfrac{1}{2}(r/j)[()_{n-1} - ()_{n-2}]\right) \tag{19}$$

Because Eq. (17) is used to update $\boldsymbol{v}^N$ in Eqs. (4), (13), and (14), current values of $\boldsymbol{v}^N$ are not available for evaluating $\Delta \boldsymbol{R}^N_{n-1,m}$ in Eq. (18). Hence, past value extrapolation must be employed, such as in Sec. III.A:

$$\Delta \boldsymbol{R}^N_{n-1,m} = \frac{T_m}{2}\left(3\boldsymbol{v}^N_{m-1} - \boldsymbol{v}^N_{m-2}\right) \qquad \text{for} \qquad r = 1$$

$$\Delta \boldsymbol{R}^N_{n-1,m} = \frac{T_m}{2}\left[3\boldsymbol{v}^N_{m-1} - \boldsymbol{v}^N_{m-2} + \sum_{i=m+1-r}^{m-1}\left(\boldsymbol{v}^N_i + \boldsymbol{v}^N_{i-1}\right)\right] \qquad \text{for} \qquad r > 1 \tag{20}$$

2. Body Frame Integrated Specific Force Increment

The $\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m}$ integral term in Eqs. (11) and (12) is calculated using a high-speed digital repetition algorithm similar to the type employed in Ref. 1, Eqs. (35) and (36), for attitude updating. The derivation of the algorithm is initially based on first-order approximations for $C^{B_{I(m-1)}}_{B}$. The first-order solution is divided into two parts for application of the two-speed algorithm approach: a portion that can be calculated at the m cycle rate which measures the effect of constant B frame angular rate and specific force, and a high-speed portion within the m cycle, which measures dynamic variations in B frame angular rate/specific force. The first-order m cycle portion is then expanded to be analytically exact under constant angular rate/specific force.

Following the development approach in Ref. 1, Sec. IV.A.1, the $C^{B_{I(m-1)}}_{B(t)}$ term in the Eq. (12) $\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m}$ integrand is expressed as

$$C^{B_{I(m-1)}}_{B(t)} = \mathrm{I} + \frac{\sin\phi(t)}{\phi(t)}\left(\boldsymbol{\phi}(t)\times\right) + \frac{1-\cos\phi(t)}{\phi(t)^2}\left(\boldsymbol{\phi}(t)\times\right)^2 \tag{21}$$

where $\boldsymbol{\phi}(t)$ is the rotation vector defining the general orientation of frame B relative to frame $B_{I(m-1)}$ for time t greater than t_{m-1}. Reference 1 Eqs. (32) and (33) show that $\boldsymbol{\phi}(t)$ in Eq. (21) can be approximated by

$$\boldsymbol{\phi}(t) \approx \boldsymbol{\alpha}(t) \tag{22}$$

$$\boldsymbol{\alpha}(t) = \int_{t_{m-1}}^{t} \boldsymbol{\omega}^B_{IB}\,\mathrm{d}\tau \tag{23}$$

where τ is an integration time parameter. A first-order approximation for Eq. (21) that is consistent with Eq. (22) neglects $(\boldsymbol{\phi}(t)\times)^2$ and approximates $\sin\phi(t)/\phi(t)$ by unity [assuming that the m cycle rate is selected fast enough to maintain $\phi(t)$ at a reasonably small value, e.g., less than 0.05 rad]. With Eq. (22), Eq. (21) reduces to

$$C^{B_{I(m-1)}}_{B(t)} \approx \mathrm{I} + \left(\boldsymbol{\alpha}(t)\times\right) \tag{24}$$

Substituting Eq. (24) into Eq. (12) then yields to first order

$$\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m} = \int_{t_{m-1}}^{t_m}\left[\mathrm{I} + \left(\boldsymbol{\alpha}(t)\times\right)\right]\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}t$$

$$= \int_{t_{m-1}}^{t_m}\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}t + \int_{t_{m-1}}^{t_m}\left(\boldsymbol{\alpha}(t)\times\right)\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}t \tag{25}$$

or, including Eq. (23),

$$\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m} = \boldsymbol{\upsilon}_m + \int_{t_{m-1}}^{t_m}\left(\boldsymbol{\alpha}(t)\times\boldsymbol{a}^B_{\mathrm{SF}}\right)\mathrm{d}t$$

$$\boldsymbol{\alpha}(t) = \int_{t_{m-1}}^{t}\boldsymbol{\omega}^B_{IB}\,\mathrm{d}\tau, \qquad \boldsymbol{\upsilon}(t) = \int_{t_{m-1}}^{t}\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}\tau, \qquad \boldsymbol{\upsilon}_m = \boldsymbol{\upsilon}(t_m) \tag{26}$$

Equations (26) define a method for calculating $\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m}$ in Eq. (11). It is instructive to analyze these equations under constant B frame angular rate $\boldsymbol{\omega}^B_{IB}$ and specific force $\boldsymbol{a}^B_{\mathrm{SF}}$ for which

$$\boldsymbol{\alpha}(t) = (t - t_{m-1})\boldsymbol{\omega}^B_{IB}, \qquad \boldsymbol{\upsilon}(t) = (t - t_{m-1})\,\boldsymbol{a}^B_{\mathrm{SF}}$$

$$\boldsymbol{\omega}^B_{IB}, \boldsymbol{a}^B_{\mathrm{SF}} = \text{const} \tag{27}$$

Substituting $\boldsymbol{\alpha}(t)$ from Eq. (27) into the Eq. (26) $\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m}$ expression yields for constant B frame angular rate and specific force

$$\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m} = \boldsymbol{\upsilon}_m + \int_{t_{m-1}}^{t_m}\left((t - t_{m-1})\boldsymbol{\omega}^B_{IB}\times\boldsymbol{a}^B_{\mathrm{SF}}\right)\mathrm{d}t$$

$$= \boldsymbol{\upsilon}_m + \left(\boldsymbol{\omega}^B_{IB}\times\boldsymbol{a}^B_{\mathrm{SF}}\right)\int_{t_{m-1}}^{t_m}(t - t_{m-1})\,\mathrm{d}t$$

$$= \boldsymbol{\upsilon}_m + \left(\boldsymbol{\omega}^B_{IB}\times\boldsymbol{a}^B_{\mathrm{SF}}\right)\frac{1}{2}(t_m - t_{m-1})^2$$

$$= \boldsymbol{\upsilon}_m + \frac{1}{2}\left(\boldsymbol{\omega}^B_{IB}(t_m - t_{m-1})\right)\times\left(\boldsymbol{a}^B_{\mathrm{SF}}(t_m - t_{m-1})\right) \tag{28}$$

or, with Eqs. (26) and (27) for constant B frame angular rate and specific force,

$$\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m} = \boldsymbol{\upsilon}_m + \frac{1}{2}\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m$$

$$\boldsymbol{\alpha}(t) = \int_{t_{m-1}}^{t}\boldsymbol{\omega}^B_{IB}\,\mathrm{d}\tau, \qquad \boldsymbol{\alpha}_m = \boldsymbol{\alpha}(t_m) \tag{29}$$

$$\boldsymbol{\upsilon}(t) = \int_{t_{m-1}}^{t}\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}\tau, \qquad \boldsymbol{\upsilon}_m = \boldsymbol{\upsilon}(t_m)$$

Comparing Eq. (26) for the general case with Eq. (29) for the constant angular rate/specific force condition, we see that the difference is the replacement of the integral term with $\frac{1}{2}\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m$.

For situations where constant angular rate/specific force is a reasonable approximation over the t_{m-1} to t_m time interval, Eq. (29) is preferred over Eq. (26) because the integral term (and its attendant high-speed algorithm) is replaced by $\frac{1}{2}\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m$, which is evaluated once each m cycle.

A fundamental limitation in Eq. (26) or Eq. (29) is the first-order approximation that underlies their development, i.e., Eq. (24) for $C^{B_{I(m-1)}}_{B(t)}$ that was used in the Eq. (12) $\Delta \boldsymbol{v}^{B_{I(m-1)}}_{\mathrm{SF}_m}$ expression. It would be desirable if the Eq. (24) approximation could be applied only to the high-frequency content of $C^{B_{I(m-1)}}_{B(t)}$ with the low-frequency content retaining the full Eq. (21) form. Such an algorithm can be synthesized by first noting that

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{\alpha}(t)\times\boldsymbol{\upsilon}(t)\right) = \boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) + \dot{\boldsymbol{\alpha}}(t)\times\boldsymbol{\upsilon}(t)$$

$$= \boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) - \boldsymbol{\upsilon}(t)\times\dot{\boldsymbol{\alpha}}(t) \tag{30}$$

with $\boldsymbol{\alpha}(t)$ and $\boldsymbol{\upsilon}(t)$ as defined in Eq. (26). Upon rearrangement, Eq. (30) becomes

$$\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{\alpha}(t)\times\boldsymbol{\upsilon}(t)\right) + \boldsymbol{\upsilon}(t)\times\dot{\boldsymbol{\alpha}}(t) \tag{31}$$

Trivially,

$$\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) = \tfrac{1}{2}\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) + \tfrac{1}{2}\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) \tag{32}$$

We now substitute Eq. (31) for one of the terms on the right in Eq. (32) to obtain

$$\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{\alpha}(t)\times\boldsymbol{\upsilon}(t)\right) + \frac{1}{2}\left(\boldsymbol{\alpha}(t)\times\dot{\boldsymbol{\upsilon}}(t) + \boldsymbol{\upsilon}(t)\times\dot{\boldsymbol{\alpha}}(t)\right) \tag{33}$$

From Eq. (26) we know that

$$\dot{\boldsymbol{\alpha}}(t) = \boldsymbol{\omega}^B_{IB}, \qquad \dot{\boldsymbol{\upsilon}}(t) = \boldsymbol{a}^B_{\mathrm{SF}} \tag{34}$$

whereby Eq. (33) assumes the form

$$\alpha(t) \times a_{SF}^B = \frac{d}{dt}\Big(\alpha(t) \times v(t)\Big) + \frac{1}{2}\Big(\alpha(t) \times a_{SF}^B + v(t) \times \omega_{IB}^B\Big) \quad (35)$$

Equation (35) is an alternate for the integrand in the Eq. (26) $\Delta v_{SF_m}^{B_{I(m-1)}}$ expression. Substitution of Eq. (35) for the integrand then yields the following equivalent form:

$$\Delta v_{SF_m}^{B_{I(m-1)}} = v_m + \frac{1}{2}(\alpha_m \times v_m) + \frac{1}{2}\int_{t_{m-1}}^{t_m} \Big(\alpha(t) \times a_{SF}^B + v(t) \times \omega_{IB}^B\Big)\,dt \quad (36)$$

If we now compare $\Delta v_{SF_m}^{B_{I(m-1)}}$ in Eqs. (36) and (29) under constant angular rate/specific force conditions, we see that they are equivalent except for the integral term in Eq. (36). It is easily verified by substitution of Eq. (27) that the integral term in Eq. (36) vanishes for constant B frame angular rate/specific force. We conclude that the integral term in Eq. (36) represents the integrated contribution of the high-frequency content in the Eq. (12) $\Delta v_{SF_m}^{B_{I(m-1)}}$ integrand; the remaining terms, i.e., $v_m + \frac{1}{2}(\alpha_m \times v_m)$, represent the low-frequency content.

The integral term in Eq. (36), denoted as sculling, measures the rectification of combined dynamic angular rate/specific force into a net constant contribution to $\Delta v_{SF_m}^{B_{I(m-1)}}$. The rectification is a maximum under classical sculling motion defined as sinusoidal angular rate/specific force in which the angular rate about one B frame axis is at the same frequency and in phase with the specific force along another B frame axis (with rectified constant specific force then produced along the average third axis direction). This is the same principle used by mariners to propel a boat in the forward direction using a single oar operated with an undulating motion (also denoted as sculling, the original use of the term). Note that the $\Delta v_{SF_m}^{B_{I(m-1)}}$ integral term in Eq. (26) has also been denoted as sculling even though it contains large contributions under constant angular rate/specific force, i.e., nonsculling conditions. The $\frac{1}{2}(\alpha_m \times v_m)$ term in Eq. (36) is identified here as velocity rotation compensation. The velocity notation has been adopted to denote that this rotation compensation term feeds the velocity rate equation (in contrast with a position rotation compensation term to be discussed in Sec. IV that feeds the position rate equation). With these definitions, a comparison between Eqs. (26) and (36) identifies the integral term in Eq. (26) as representing the composite of sculling and velocity rotation compensation effects. Using the latter terminology, Eq. (36) is rewritten as

$$\Delta v_{SF_m}^{B_{I(m-1)}} = v_m + \Delta v_{rot_m} + \Delta v_{scul_m} \quad (37)$$

$$\Delta v_{scul}(t) = \frac{1}{2}\int_{t_{m-1}}^{t} \Big(\alpha(\tau) \times a_{SF}^B + v(\tau) \times \omega_{IB}^B\Big)\,d\tau$$

$$\Delta v_{scul_m} = \Delta v_{scul}(t_m)$$

$$\alpha(\tau) = \int_{t_{m-1}}^{\tau} \omega_{IB}^B\,dt, \qquad \alpha_m = \alpha(t_m) \quad (38)$$

$$v(\tau) = \int_{t_{m-1}}^{\tau} a_{SF}^B\,dt, \qquad v_m = v(t_m)$$

$$\Delta v_{rot_m} = \tfrac{1}{2}(\alpha_m \times v_m) \quad (39)$$

where Δv_{rot_m} is the velocity rotation compensation term and Δv_{scul_m} is the sculling term. Alternatively, beginning from the Eq. (26) version,

$$\Delta v_{SF_m}^{B_{I(m-1)}} = v_m + \Delta v_{rot/scul_m} \quad (40)$$

$$\Delta v_{rot/scul}(t) = \int_{t_{m-1}}^{t} \Big(\alpha(\tau) \times a_{SF}^B\Big)\,d\tau$$

$$\Delta v_{rot/scul_m} = \Delta v_{rot/scul}(t_m) \quad (41)$$

with $\alpha(\tau)$ and v_m from sculling Eq. (38) and where $\Delta v_{rot/scul_m}$ is the composite sculling and velocity rotation compensation term.

Equations (37–39) are completely equivalent to Eqs. (40) and (41); both equation sets exhibit only first-order accuracy. However, Eq. (37) is now in a form that enables us to substitute an expanded expression for the Eq. (39) velocity rotation compensation term that makes Eq. (37) exact under constant rate/specific force conditions. This is an important extension because general motion is typically dominated by low-frequency angular rate and specific force components that may have large amplitudes under extreme maneuvers (where second-order algorithm errors may not be negligible). The extension to exactness is not possible for Eqs. (40) and (41) because the rotation compensation effect is imbedded within the integral, which includes the first-order sculling term. The following subsections derive an exact Δv_{rot_m} velocity rotation compensation algorithm for Eq. (37) in addition to digital integration algorithms for the Eq. (38) integral terms. Using the same procedure, a digital integration algorithm can also be developed for $\Delta v_{rot/scul_m}$ in Eqs. (40) and (41), as shown in Ref. 8, Sec. 7.2.2.2.2.

Exact velocity rotation compensation. The exact velocity rotation compensation algorithm is defined as the algorithm that, when substituted for Δv_{rot_m} in Eq. (37), provides an exact solution for $\Delta v_{SF_m}^{B_{I(m-1)}}$ in Eq. (12) under constant B frame angular rate/specific force conditions. The exact velocity rotation compensation algorithm is derived from Eq. (12) using Eq. (21) for $C_{B_{(t)}}^{B_{I(m-1)}}$ under constant angular rate/specific force. We first consider the more general condition where only the direction of the angular rate vector is constant, i.e., a nonconing environment in which the angular rate vector is not rotating. From Eq. (23), for a nonconing angular rate condition,

$$\alpha(t) = \alpha(t)u_\omega, \qquad \alpha(t) = \int_{t_{m-1}}^{t} \omega\,d\tau, \qquad \frac{\alpha(t)}{\alpha(t)} = u_\omega \quad (42)$$

where ω is the magnitude of ω_{IB}^B, and u_ω is a unit vector along ω_{IB}^B that is considered constant in the B frame.

As discussed in Ref. 1, Sec. IV.A.1, for the case where ω_{IB}^B is not rotating, $\phi(t)$ is equal to $\alpha(t)$ (the integral of ω_{IB}^B). Under this restriction, Eq. (21) with Eq. (42) for $\phi(t)$ substituted in Eq. (12) gives for the nonconing angular rate condition

$$\Delta v_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} \Big[\mathrm{I} + \sin\alpha(t)(u_\omega\times) + \big(1-\cos\alpha(t)\big)(u_\omega\times)^2\Big]a_{SF}^B\,dt \quad (43)$$

For nonconing angular rate and constant B frame specific force, Eq. (43) can be expanded to

$$\Delta v_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} a_{SF}^B\,dt + \Big(u_\omega \times a_{SF}^B\Big)\int_{t_{m-1}}^{t_m} \sin\alpha(t)\,dt + \Big[u_\omega \times \Big(u_\omega \times a_{SF}^B\Big)\Big]\int_{t_{m-1}}^{t_m} \Big(1-\cos\alpha(t)\Big)\,dt \quad (44)$$

Section III.B.2 nomenclature is now applied with the nonconing rate/constant specific force assumption and appropriate Eq. (42) relationships,

$$v_m = \int_{t_{m-1}}^{t_m} a_{SF}^B\,dt = a_{SF}^B(t_m - t_{m-1}) = a_{SF}^B T_m, \qquad a_{SF}^B = \frac{v_m}{T_m}$$

$$\frac{\alpha(t)}{\alpha(t)} = u_\omega, \qquad u_\omega = \frac{\alpha_m}{\alpha_m} \quad (45)$$

and $\alpha_m = \alpha(t_m)$ is the magnitude of $\boldsymbol{\alpha}(t_m)$. Substituting Eqs. (45) into Eq. (44) then yields for nonconing angular rate and constant specific force

$$\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}} = \boldsymbol{v}_m + \frac{\boldsymbol{\alpha}_m \times \boldsymbol{v}_m}{\alpha_m T_m} \int_{t_{m-1}}^{t_m} \sin\alpha(t)\,\mathrm{d}t + \frac{\boldsymbol{\alpha}_m \times (\boldsymbol{\alpha}_m \times \boldsymbol{v}_m)}{\alpha_m^2 T_m} \int_{t_{m-1}}^{t_m} \big(1-\cos\alpha(t)\big)\,\mathrm{d}t \tag{46}$$

To evaluate the integral terms in Eq. (46), we now adopt the constant angular rate condition whereby $\boldsymbol{\omega}$ in Eq. (42) is constant. Then,

$$\alpha(t) = \omega(t - t_{m-1}), \qquad \omega = \mathrm{const} \tag{47}$$

Applying Eq. (47) in Eq. (46) with Eq. (45) for α_m allows the integral terms to be evaluated for constant B frame angular rate as

$$\begin{aligned} \int_{t_{m-1}}^{t_m} \sin\alpha(t)\,\mathrm{d}t &= \frac{T_m}{\alpha_m}(1-\cos\alpha_m) \\ \int_{t_{m-1}}^{t_m} \big(1-\cos\alpha(t)\big)\,\mathrm{d}t &= T_m\left(1 - \frac{\sin\alpha_m}{\alpha_m}\right) \end{aligned} \tag{48}$$

Substitution in Eq. (46) then yields the desired form for the exact $\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$ solution under constant B frame angular rate and specific force

$$\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}} = \boldsymbol{v}_m + \frac{(1-\cos\alpha_m)}{\alpha_m^2}\boldsymbol{\alpha}_m \times \boldsymbol{v}_m + \frac{1}{\alpha_m^2}\left(1 - \frac{\sin\alpha_m}{\alpha_m}\right)\boldsymbol{\alpha}_m \times (\boldsymbol{\alpha}_m \times \boldsymbol{v}_m) \tag{49}$$

Equation (49) constitutes an exact solution for $\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$ under constant angular rate/specific force. We are now in a position to compare Eq. (49) with Eq. (37) under the same conditions to identify the exact velocity rotation compensation term. Under constant rate/specific force conditions, the sculling term in Eq. (37) vanishes (see discussion in Sec. II.B.2), and $\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$ is given by

$$\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}} = \boldsymbol{v}_m + \Delta \boldsymbol{v}_{\mathrm{rot}_m} \tag{50}$$

If we compare Eqs. (49) and (50) it should be clear from its definition that the exact velocity rotation compensation term $\Delta \boldsymbol{v}_{\mathrm{rot}_m}$ is

$$\Delta \boldsymbol{v}_{\mathrm{rot}_m} = \frac{(1-\cos\alpha_m)}{\alpha_m^2}\boldsymbol{\alpha}_m \times \boldsymbol{v}_m + \frac{1}{\alpha_m^2}\left(1 - \frac{\sin\alpha_m}{\alpha_m}\right)\boldsymbol{\alpha}_m \times (\boldsymbol{\alpha}_m \times \boldsymbol{v}_m) \tag{51}$$

The trigonometric coefficients in Eq. (51) can be calculated from the Taylor series formulas

$$\begin{aligned} \frac{(1-\cos\alpha_m)}{\alpha_m^2} &= \frac{1}{2!} - \frac{\alpha_m^2}{4!} + \frac{\alpha_m^4}{6!} - \cdots \\ \frac{1}{\alpha_m^2}\left(1 - \frac{\sin\alpha_m}{\alpha_m}\right) &= \frac{1}{3!} - \frac{\alpha_m^2}{5!} + \frac{\alpha_m^4}{7!} - \cdots \end{aligned} \tag{52}$$

Equation (51) with Eqs. (52) constitute an alternative algorithm for the $\Delta \boldsymbol{v}_{\mathrm{rot}_m}$ velocity rotation compensation term in Eq. (39) that will generate an exact solution for $\Delta \boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$ in Eq. (37) under constant B frame angular rate/specific force conditions. In contrast, the $\Delta \boldsymbol{v}_{\mathrm{rot}}$ algorithm in Eq. (39) is accurate to only first order. Note that, to first order in $\boldsymbol{\alpha}_m$, Eq. (51) with Eq. (52) reduces to the Eq. (39) $\Delta \boldsymbol{v}_{\mathrm{rot}_m}$ form (as it should).

Integrated specific force and sculling increments. In this subsection we develop digital algorithms for calculating the $\boldsymbol{v}_m$ and $\Delta \boldsymbol{v}_{\mathrm{scul}_m}$ integral terms in Eq. (37) and (38) [the $\boldsymbol{\alpha}_m$ term for these equations is provided from the attitude algorithm in Ref. 1, Eqs. (41)]. A similar procedure can be used to develop an algorithm for $\Delta \boldsymbol{v}_{\mathrm{rot/scul}_m}$ in Eqs. (40) and (41). Following the identical procedure used in Ref. 1, Sec. IV.A.1, for the coning algorithm, we develop the $\Delta \boldsymbol{v}_{\mathrm{scul}_m}$ sculling algorithm by considering $\Delta \boldsymbol{v}_{\mathrm{scul}_m}$ to be the value at $t = t_m$ of the general function $\Delta \boldsymbol{v}_{\mathrm{scul}}(t)$ [as in Eq. (38)]. Let us consider the Eq. (38) $\Delta \boldsymbol{v}_{\mathrm{scul}}(t)$ integration as being divided into portions up to and after a general time t_{l-1} within the t_{m-1} to t_m interval so that

$$\begin{aligned} \Delta \boldsymbol{v}_{\mathrm{scul}}(t) &= \Delta \boldsymbol{v}_{\mathrm{scul}_{l-1}} + \delta \boldsymbol{v}_{\mathrm{scul}}(t) \\ \delta \boldsymbol{v}_{\mathrm{scul}}(t) &= \frac{1}{2}\int_{t_{l-1}}^{t} \big(\boldsymbol{\alpha}(\tau) \times \boldsymbol{a}_{\mathrm{SF}}^B + \boldsymbol{v}(\tau) \times \boldsymbol{\omega}_{IB}^B\big)\,\mathrm{d}\tau \end{aligned} \tag{53}$$

We now define the next l cycle time point t_l within the t_{m-1} to t_m interval so that Eqs. (53) at t_l with $\boldsymbol{\alpha}(\tau)$ and $\boldsymbol{v}(\tau)$ from Eq. (38), including initial conditions, become

$$\begin{aligned} &\boldsymbol{\alpha}(\tau) = \boldsymbol{\alpha}_{l-1} + \Delta\boldsymbol{\alpha}(\tau) \\ &\Delta\boldsymbol{\alpha}(\tau) = \int_{t_{l-1}}^{\tau} \boldsymbol{\omega}_{IB}^B\,\mathrm{d}t, \qquad \Delta\boldsymbol{\alpha}_l = \Delta\boldsymbol{\alpha}(t_l) \\ &\boldsymbol{\alpha}_l = \boldsymbol{\alpha}_{l-1} + \Delta\boldsymbol{\alpha}_l, \qquad \boldsymbol{\alpha}_m = \boldsymbol{\alpha}_l(t_l = t_m) \\ &\boldsymbol{\alpha}_l = 0 \quad \text{at} \quad \tau = t_{m-1} \end{aligned} \tag{54}$$

$$\begin{aligned} &\boldsymbol{v}(\tau) = \boldsymbol{v}_{l-1} + \Delta\boldsymbol{v}(\tau) \\ &\Delta\boldsymbol{v}(\tau) = \int_{t_{l-1}}^{\tau} \boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}t, \qquad \Delta\boldsymbol{v}_l = \Delta\boldsymbol{v}(t_l) \\ &\boldsymbol{v}_l = \boldsymbol{v}_{l-1} + \Delta\boldsymbol{v}_l, \qquad \boldsymbol{v}_m = \boldsymbol{v}_l(t_l = t_m) \\ &\boldsymbol{v}_l = 0 \quad \text{at} \quad \tau = t_{m-1} \end{aligned} \tag{55}$$

$$\begin{aligned} &\Delta \boldsymbol{v}_{\mathrm{scul}_l} = \Delta \boldsymbol{v}_{\mathrm{scul}_{l-1}} + \delta \boldsymbol{v}_{\mathrm{scul}_l} \\ &\delta \boldsymbol{v}_{\mathrm{scul}}(t) = \frac{1}{2}\int_{t_{l-1}}^{t} \big(\boldsymbol{\alpha}(t) \times \boldsymbol{a}_{\mathrm{SF}}^B + \boldsymbol{v}(t) \times \boldsymbol{\omega}_{IB}^B\big)\,\mathrm{d}t \\ &\delta \boldsymbol{v}_{\mathrm{scul}_l} = \delta \boldsymbol{v}_{\mathrm{scul}}(t_l) \\ &\Delta \boldsymbol{v}_{\mathrm{scul}_m} = \Delta \boldsymbol{v}_{\mathrm{scul}_l}(t_l = t_m), \qquad \Delta \boldsymbol{v}_{\mathrm{scul}_l} = 0 \quad \text{at} \quad t = t_{m-1} \end{aligned} \tag{56}$$

where l is the high-speed computer cycle index. Equations (54–56) constitute the construct of a digital recursive algorithm at the l computer cycle rate for calculating the $\Delta \boldsymbol{v}_{\mathrm{scul}_m}$ sculling term and $\boldsymbol{v}_m$ as a summation of changes in $\Delta \boldsymbol{v}_{\mathrm{scul}}$ and $\boldsymbol{v}$ over the t_{m-1} to t_m interval. It remains to determine a digital equivalent for the $\delta \boldsymbol{v}_{\mathrm{scul}_l}$ integral term in Eq. (56). We begin by substitution of $\boldsymbol{\alpha}(t)$ and the definitions for $\Delta\boldsymbol{\alpha}_l$ and $\Delta\boldsymbol{v}_l$ from Eq. (54) into $\delta \boldsymbol{v}_{\mathrm{scul}}$:

$$\delta \boldsymbol{v}_{\mathrm{scul}_l} = \frac{1}{2}(\boldsymbol{\alpha}_{l-1} \times \Delta\boldsymbol{v}_l + \boldsymbol{v}_{l-1} \times \Delta\boldsymbol{\alpha}_l) + \frac{1}{2}\int_{t_{l-1}}^{t_l} \big(\Delta\boldsymbol{\alpha}(t) \times \boldsymbol{a}_{\mathrm{SF}}^B + \Delta\boldsymbol{v}(t) \times \boldsymbol{\omega}_{IB}^B\big)\,\mathrm{d}t \tag{57}$$

Development of a digital algorithm for the integral term in sculling Eq. (57) is based on an assumed form for the B frame angular rate/specific force history during the t_{l-1} to t_l time interval. Unlike the coning algorithm, very little published work exists for selecting angular rate/specific force time histories for application to sculling algorithm design. In principle, the approaches used for the coning algorithm can also be applied for sculling, including optimization for sculling-type motion (see discussion in Ref. 1, Sec. IV.A.1). For this paper, we provide an example based on general linearly changing angular rate/specific force over the t_{l-1} to t_l time interval:

$$\boldsymbol{\omega}_{IB}^B \approx \boldsymbol{A} + \boldsymbol{B}(t - t_{l-1}), \qquad \boldsymbol{a}_{\mathrm{SF}}^B \approx \boldsymbol{C} + \boldsymbol{D}(t - t_{l-1}) \tag{58}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ are constant vectors.

An algorithm for the integral term in Eq. (57) can be developed by first substituting Eq. (58) for $\boldsymbol{\omega}_{IB}^B$ and $\boldsymbol{a}_{\mathrm{SF}}^B$ in Eq. (57) and then calculating the Eq. (57) integral term analytically over the t_{l-1} to t_l time interval. The intermediate result is an equation for the Eq. (57)

integral term as a function of the $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ constant vectors. A set of $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ constants is then calculated for each t_{l-1} to t_l time interval using successive measurements of integrated angular rate and specific force increments from the inertial sensors. Two successive measurements would be required to uniquely determine the four constant vectors $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ for the Eq. (58) linearly ramping model. (A parabolic model would be characterized by six constant vectors and require three successive sensor measurements for determination, etc.) The result is then substituted for $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ in the intermediate result (defined earlier) to derive the algorithm equivalent to the Eq. (57) integral term over the t_{l-1} to t_l time interval. If the successive sensor increments are sampled at the l cycle rate, measurements would be taken at t_{l-1} and t_l, spanning t_{l-2} to t_{l-1} and t_{l-1} to t_l (or t_{l-2} to t_l, overall). Alternatively,[6,7,11] the sensor samples can be taken within the t_{l-1} to t_l time interval, two samples per l cycle for the Eq. (58) linearly ramping model, three for a parabolic model, etc. For sensor samples taken at the l cycle rate, the results of the latter procedure (detailed in Ref. 8, Sec. 7.2.2.2.2) show that for the Eq. (58) linearly ramping model, the algorithm equivalent to Eqs. (54–57) is given by

$$\Delta\boldsymbol{\alpha}_l,\ \boldsymbol{\alpha}_l = \text{integrated angular rate sensor outputs from Ref. 1, Eqs. (46)} \tag{59}$$

$$\Delta\boldsymbol{v}_l = \int_{t_{l-1}}^{t_l} \mathrm{d}\boldsymbol{v}$$
$$\boldsymbol{v}_l = \boldsymbol{v}_{l-1} + \Delta\boldsymbol{v}_l, \quad \boldsymbol{v}_m = \boldsymbol{v}_l(t_l = t_m), \quad \boldsymbol{v}_l = 0 \text{ at } t = t_{m-1} \tag{60}$$

$$\delta\boldsymbol{v}_{\text{scul}_l} = \tfrac{1}{2}\left[\left(\boldsymbol{\alpha}_{l-1} + \tfrac{1}{6}\Delta\boldsymbol{\alpha}_{l-1}\right)\times\Delta\boldsymbol{v}_l + \left(\boldsymbol{v}_{l-1} + \tfrac{1}{6}\Delta\boldsymbol{v}_{l-1}\right)\times\Delta\boldsymbol{\alpha}_l\right]$$
$$\Delta\boldsymbol{v}_{\text{scul}_l} = \Delta\boldsymbol{v}_{\text{scul}_{l-1}} + \delta\boldsymbol{v}_{\text{scul}_l}, \qquad \Delta\boldsymbol{v}_{\text{scul}_m} = \Delta\boldsymbol{v}_{\text{scul}_l}(t_l = t_m)$$
$$\Delta\boldsymbol{v}_{\text{scul}_l} = 0 \quad \text{at} \quad t = t_{m-1} \tag{61}$$

where

$\Delta\boldsymbol{v}_l$ = summation of integrated specific force output increments from accelerometers

$\mathrm{d}\boldsymbol{v}$ = differential integrated specific force increment, i.e., analytical representation of pulse output from strapdown accelerometers, $\boldsymbol{a}_{\text{SF}}^B\,\mathrm{d}t$

Equation (61) for $\Delta\boldsymbol{v}_{\text{scul}_m}$ has been classified as a second-order algorithm because it includes current and past l cycle $\Delta\boldsymbol{\alpha}$, $\Delta\boldsymbol{v}$ products. The l, $l-1$ cycle $\Delta\boldsymbol{\alpha}$, $\Delta\boldsymbol{v}$ product terms in $\delta\boldsymbol{v}_{\text{scul}_l}$, i.e., the $\frac{1}{6}$ terms, stem from the approximation of linearly ramping angular rate and specific force in the t_{l-2} to t_l time interval. If the angular rate and specific force terms were approximated as parabolically varying functions of time, a third-order algorithm would result, containing l, $l-1$, and $l-2$ cycle $\Delta\boldsymbol{\alpha}$, $\Delta\boldsymbol{v}$ products. If the angular rate and specific force were approximated as constants over t_{l-1} to t_l, the $\frac{1}{6}$ terms in Eq. (61) would vanish, resulting in a first-order algorithm for $\Delta\boldsymbol{v}_{\text{scul}_m}$. Finally, if angular rate and specific force are slowly varying, we can approximate $\Delta\boldsymbol{v}_{\text{scul}_m}$ as being equal to zero. Alternatively (and more accurately), we can set the l cycle rate equal to the m cycle rate, which equates $\Delta\boldsymbol{v}_{\text{scul}_m}$ to $\delta\boldsymbol{v}_{\text{scul}_l}$ in Eq. (61) calculated once at time t_m [and noting from the initial condition definitions in Eq. (60) and Ref. 1, Eqs. (46), that $\boldsymbol{\alpha}_{l-1}$ and $\boldsymbol{v}_{l-1}$ would be zero]. Note that setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high-speed, higher-order algorithm with a simpler software architecture than the two-speed approach but requiring more throughput. Continuing advances in the speed of modern-day computers may make this the preferred approach for the future.

IV. Position Update Algorithms

In this section we develop digital integration algorithms for calculating position relative to the Earth in the form of altitude h above the Earth's surface and the C_N^E direction cosine matrix defining the angular orientation between the local level N frame and the Earth-fixed E frame (from which latitude/longitude can be extracted). Two algorithm forms are developed: a typical form based on trapezoidal integration of velocity and a high-resolution form that accounts for dynamic attitude and velocity changes within the position update period. The high-resolution algorithm is modeled after the Sec. III two-speed velocity update approach.

Both the typical and high-resolution forms can be represented by the continuous differential equation form of Ref. 1, Eqs. (21) and (22), repeated here as

$$\dot{h} = \boldsymbol{u}_{ZN}^N \cdot \boldsymbol{v}^N \tag{62}$$

$$\dot{C}_N^E = C_N^E\left(\boldsymbol{\omega}_{EN}^N\times\right) \tag{63}$$

where h is altitude above the Earth's surface. The typical and high-resolution forms derive from a general updating formulation for h and C_N^E. The following sections formulate the general position updating process and then derive computational approaches for typical and high-resolution position updating.

A. Position Updating in General

The general altitude h updating algorithm is formulated as the integral of Eq. (62) over a position update cycle n:

$$h_n = h_{n-1} + \Delta h_n \tag{64}$$

$$\Delta h_n = \int_{t_{n-1}}^{t_n} \boldsymbol{u}_{ZN}^N \cdot \boldsymbol{v}^N\,\mathrm{d}t \tag{65}$$

Allowing for the higher-speed digital computation loop, i.e., the m loop for attitude and velocity integration, Eq. (65) can be written as

$$\Delta h_n = \boldsymbol{u}_{ZN}^N \cdot \sum_{m=1}^{j} \Delta\boldsymbol{R}_m^N \tag{66}$$

$$\Delta\boldsymbol{R}_m^N \equiv \int_{t_{m-1}}^{t_m} \boldsymbol{v}^N\,\mathrm{d}t \tag{67}$$

If vertical channel gravity/divergence stabilization is to be incorporated, an additional operation would be included in Eq. (64) representing the altitude control function (see Ref. 8, Sec. 4.4.1, and Ref. 10, pp. 102–103).

The general updating algorithm for the C_N^E direction cosine matrix is designed to achieve the same numerical result at the update times as would the formal continuous integration of the Eq. (63) $\dot{C}_N^E$ expression at the same time instant. The algorithm is developed by envisioning the local level navigation N frame orientation history in the digital updating world [produced in Eq. (63) by $\boldsymbol{\omega}_{EN}^N$] as being constructed of successive discrete orientations relative to the Earth (E frame) at each update time instant. The general updating algorithm for C_N^E is then constructed as follows using the Ref. 1, Eq. (3), direction cosine matrix product chain rule:

$$C_{N_{E(n)}}^E = C_{N_{E(n-1)}}^E C_{N_{E(n)}}^{N_{E(n-1)}} \tag{68}$$

where

$N_{E(n)}$ = discrete orientation of the N frame in rotating Earth frame space (E) at computer update time t_n

$C_{N_{E(n-1)}}^E$ = C_N^E relating the N frame at time t_{n-1} to the E frame

$C_{N_{E(n)}}^E$ = C_N^E relating the N frame at time t_n to the E frame

$C_{N_{E(n)}}^{N_{E(n-1)}}$ = direction cosine matrix that accounts for N frame rotation relative to the Earth (E) from its orientation at time t_{n-1} to its orientation at time t_n

The $C_{N_{E(n)}}^{N_{E(n-1)}}$ matrix in Eq. (68) is defined formally as

$$C_{N_{E(n)}}^{N_{E(n-1)}} = \mathrm{I} + \int_{t_{n-1}}^{t_n} \dot{C}_{N_{(t)}}^{N_{E(n-1)}}\,\mathrm{d}t \tag{69}$$

with $N_{(t)}$ in Eq. (69) representing the N frame attitude at an arbitrary time in the interval t_{n-1} to t_n.

Following the same development procedure as for $C_{B_{I(m)}}^{B_{I(m-1)}}$ in Ref. 1, Sec. IV.A.1, the $C_{N_{E(n)}}^{N_{E(n-1)}}$ matrix can also be expressed in terms of the rotation vector defining the frame $N_{E(n)}$ attitude relative to frame $N_{E(n-1)}$. Applying Ref. 1, Eq. (4), with Taylor series expansion for the coefficient terms obtains

$$C_{N_{E(n-1)}}^{N_{E(n)}} = \mathrm{I} + \frac{\sin\xi_n}{\xi_n}(\boldsymbol{\xi}_n\times) + \frac{(1-\cos\xi_n)}{\xi_n^2}(\boldsymbol{\xi}_n\times)(\boldsymbol{\xi}_n\times)$$
$$\frac{\sin\xi_n}{\xi_n} = 1 - \frac{\xi_n^2}{3!} + \frac{\xi_n^4}{5!} - \cdots \tag{70}$$
$$\frac{(1-\cos\xi_n)}{\xi_n^2} = \frac{1}{2!} - \frac{\xi_n^2}{4!} + \frac{\xi_n^4}{6!} - \cdots$$

where $\boldsymbol{\xi}_n$ is the rotation vector defining the frame $N_{E(n)}$ attitude at time t_n relative to the frame $N_{E(n-1)}$ attitude at time t_{n-1}.

The angular rate of the N frame relative to the Earth $\boldsymbol{\omega}_{EN}^N$ is small and typically no larger than one or two Earth rates. As such, because the t_{n-1} to t_n update cycle is relatively short, $\boldsymbol{\xi}_n$ will be very small in magnitude. Because $\boldsymbol{\omega}_{EN}^N$ is small and slowly changing over a typical t_{n-1} to t_n update cycle (due to small changes in velocity and position over this time period) the N frame rate vector $\boldsymbol{\omega}_{EN}^N$ can be approximated as nonrotating. The result is that $\boldsymbol{\xi}_n$ for Eq. (70) can be calculated as the integral of the simplified form of the Ref. 1, Eq. (10), rotation vector rate expression whereby the cross-product terms are neglected:

$$\boldsymbol{\xi}_n \approx \int_{t_{n-1}}^{t_n} \boldsymbol{\omega}_{EN}^N\,\mathrm{d}t \tag{71}$$

A discrete digital algorithm for the Eq. (71) $\boldsymbol{\xi}_n$ integral can be constructed by first approximating Eq. (3) for $\boldsymbol{\omega}_{EN}^N$ as

$$\boldsymbol{\omega}_{EN}^N \approx \rho_{ZN_{n-\frac{1}{2}}}\boldsymbol{u}_{ZN}^N + F_{C_{n-\frac{1}{2}}}\left(\boldsymbol{u}_{ZN}^N \times \boldsymbol{v}^N\right) \tag{72}$$

where $(\,)_{n-1/2}$ is the value for ( ) midway between times t_{n-1} and t_n. Using Eq. (72) in Eq. (71) and applying the Eq. (67) definition then obtains

$$\boldsymbol{\xi}_n \approx \rho_{ZN_{n-\frac{1}{2}}}\boldsymbol{u}_{ZN}^N T_n + F_{C_{n-\frac{1}{2}}}\left(\boldsymbol{u}_{ZN}^N \times \sum_{m=1}^{j} \Delta\boldsymbol{R}_m^N\right) \tag{73}$$

where T_n is the computer n cycle update period $t_n - t_{n-1}$. The $(\,)_{n-1/2}$ terms in Eq. (73) are all functions of position, which has not yet been updated. Hence, to calculate the $(\,)_{n-1/2}$ terms, an approximate extrapolation formula must be used based on previously computed values for the ( ) parameters. For example, a linear extrapolation formula using the last two computed values for ( ) would be

$$(\,)_{n-\frac{1}{2}} \approx (\,)_{n-1} + \tfrac{1}{2}[(\,)_{n-1} - (\,)_{n-2}] = \tfrac{3}{2}(\,)_{n-1} - \tfrac{1}{2}(\,)_{n-2} \tag{74}$$

The method for calculating the $\Delta\boldsymbol{R}_m^N$ term for Eqs. (66) and (73) from the Eq. (67) integral depends on whether typical trapezoidal integration is used for position updating or whether a more precise high-resolution integration approach is to be applied. Both are described in the following sections.

B. Typical Position Updating

Applying typical trapezoidal integration for the h and C_N^E updating process would utilize Eqs. (64), (66), (68), (70), (73), and (74) with a trapezoidal integration algorithm in Eq. (67) for $\Delta\boldsymbol{R}_m^N$:

$$\Delta\boldsymbol{R}_m^N \approx \tfrac{1}{2}\left(\boldsymbol{v}_m^N + \boldsymbol{v}_{m-1}^N\right)T_m \tag{75}$$

C. High-Resolution Position Updating

The high-resolution approach for implementing the h and C_N^E updating process utilizes Eqs. (64), (66), (68), (70), (73), and (74) with a high-speed digital integration algorithm in Eq. (67) for $\Delta\boldsymbol{R}_m^N$. The digital algorithm for $\Delta\boldsymbol{R}_m^N$ is developed by first expanding the Eq. (67) $\boldsymbol{v}^N$ integrand. Using the expression for $\boldsymbol{v}_m^N$ in Eq. (4) with Eq. (6), $\boldsymbol{v}^N$ can be defined as a continuous time function at a general time point since the last t_{m-1} update:

$$\boldsymbol{v}^N(t) = \boldsymbol{v}_{m-1}^N + C_L^N\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t) + \Delta\boldsymbol{v}_{G/\mathrm{Cor}m}^N\frac{(t-t_{m-1})}{T_m}$$
$$\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t) = \int_{t_{m-1}}^{t} C_B^L\,\boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}\tau \tag{76}$$

Equations (76) are based on the assumption that gravity/Coriolis term $\Delta\boldsymbol{v}_{G/\mathrm{Cor}m}^N$ can be approximated as the integral of a constant over t_{m-1} to t_m. With Eq. (76), $\Delta\boldsymbol{R}_m^N$ from Eq. (67) is given by

$$\Delta\boldsymbol{R}_m^N = \left(\boldsymbol{v}_{m-1}^N + \tfrac{1}{2}\Delta\boldsymbol{v}_{G/\mathrm{Cor}m}^N\right)T_m + C_L^N\Delta\boldsymbol{R}_{\mathrm{SF}m}^L$$
$$\Delta\boldsymbol{R}_{\mathrm{SF}m}^L = \int_{t_{m-1}}^{t_m}\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t)\,\mathrm{d}t, \qquad \Delta\boldsymbol{v}_{\mathrm{SF}}^L(t) = \int_{t_{m-1}}^{t} C_B^L\,\boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}\tau \tag{77}$$

where $\Delta\boldsymbol{R}_{\mathrm{SF}_m}^L$ is the L frame coordinate portion of $\Delta\boldsymbol{R}_m^N$ produced by specific force.

Equations (11), (13), and (36) show that $\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t)$ in Eq. (77) can be approximated to first order (in body rotation angle) by

$$\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t) = C_{L(n-1)}^{L(t)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t) = C_{L(m-1)}^{L(t)}C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$$
$$= \left(C_{L(m-1)}^{L(t)} - \mathrm{I}\right)C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t) + C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$$
$$= \left(C_{L(m-1)}^{L(m)} - \mathrm{I}\right)\frac{(t-t_{m-1})}{T_m}C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}m}^{L(n-1)}\frac{(t-t_{m-1})}{T_m}$$
$$+ C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$$
$$= \left(C_{L(m-1)}^{L(m)} - \mathrm{I}\right)C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}m}^{L(n-1)}\frac{(t-t_{m-1})^2}{T_m^2}$$
$$+ C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$$
$$= \left(C_{L(n-1)}^{L(m)} - C_{L(n-1)}^{L(m-1)}\right)\Delta\boldsymbol{v}_{\mathrm{SF}m}^{L(n-1)}\frac{(t-t_{m-1})^2}{T_m^2} \tag{78}$$
$$+ C_{L(n-1)}^{L(m-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$$
$$\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t) = C_{B(m-1)}^{L(n-1)}\Delta\boldsymbol{v}_{\mathrm{SF}}^{B(m-1)}(t)$$
$$\Delta\boldsymbol{v}_{\mathrm{SF}}^{B(m-1)}(t) = \boldsymbol{\upsilon}(t) + \frac{1}{2}\left(\boldsymbol{\alpha}(t)\times\boldsymbol{\upsilon}(t)\right)$$
$$+ \frac{1}{2}\int_{t_{m-1}}^{t}\left(\boldsymbol{\alpha}(\tau)\times\boldsymbol{a}_{\mathrm{SF}}^B + \boldsymbol{\upsilon}(\tau)\times\omega_{IB}^B\right)\mathrm{d}\tau$$
$$\boldsymbol{\alpha}(\tau) = \int_{t_{m-1}}^{\tau}\omega_{IB}^B\,\mathrm{d}t, \qquad \boldsymbol{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau}\boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}t$$

where

$C_{B(m-1)}^{L(n-1)}$ = C_B^L matrix updated for B frame motion at time t_{m-1} and for L frame motion at time t_{n-1}

$C_{L(n-1)}^{L(m)}, C_{L(n-1)}^{L(m-1)}$ = current and past m cycle values for the direction cosine matrix relating frame L at times t_{n-1} and t_m as calculated from Eq. (14)

In Eq. (78), the I notation in subscripts and superscripts, used for clarity in Eqs. (11) and (13), has been dropped for simplicity. In addition, $(C_{L(m-1)}^{L(t)} - \mathrm{I})$ and $\Delta\boldsymbol{v}_{\mathrm{SF}}^{L(n-1)}(t)$ in the first part of the $\Delta\boldsymbol{v}_{\mathrm{SF}}^L(t)$ expression have been approximated to be linearly ramping in time over t_{m-1} to t_m. Note also, as in Sec. III.B.1, that the $C_{L(n-1)}^L$ terms in Eq. (78) can be approximated by the identity matrix for all but very high-precision applications. Based on Eq. (78) and including

Eq. (14), the $\Delta \boldsymbol{R}^L_{\mathrm{SF}_m}$ term in Eq. (77) can be defined by the equivalent forms

$$\Delta \boldsymbol{R}^L_{\mathrm{SF}_m} = -\frac{1}{3}[(\boldsymbol{\zeta}_{n-1,m} - \boldsymbol{\zeta}_{n-1,m-1})\times]\Delta \boldsymbol{v}^{L(n-1)}_{\mathrm{SF}_m} T_m + C^{L(m-1)}_{L(n-1)} C^{L(n-1)}_{B(m-1)} \Delta \boldsymbol{R}^B_{\mathrm{SF}_m}$$

$$\Delta \boldsymbol{R}^B_{\mathrm{SF}_m} = \int_{t_{m-1}}^{t_m} \Delta \boldsymbol{v}^{B(m-1)}_{\mathrm{SF}}(t)\,\mathrm{d}t = \int_{t_{m-1}}^{t_m} \left[\boldsymbol{\upsilon}(t) + \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big) + \Delta \boldsymbol{v}_{\mathrm{scul}}(t)\right]\mathrm{d}t \tag{79}$$

with $\Delta \boldsymbol{v}_{\mathrm{scul}}(t)$, $\boldsymbol{\alpha}(t)$, and $\boldsymbol{\upsilon}(t)$ from sculling Eq. (38).

Following a similar development path as used in Sec. III.B.2 for the body frame integrated specific force increment, the $\int \frac{1}{2}(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t))\,\mathrm{d}t$ term in the Eq. (79) $\Delta \boldsymbol{R}^B_{\mathrm{SF}_m}$ expression can be revised into a nonintegral term plus an integral term that vanishes under constant angular rate/specific force, both being of first-order accuracy. The nonintegral term will then be extended into a more accurate form that is exact under constant angular rate/specific force conditions. We begin by using classical integration by parts substitution [as in Sec. III.B.2 leading to Eq. (35)] to show that the $\int \frac{1}{2}(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t))\,\mathrm{d}t$ term in Eq. (79) has the following equivalent forms:

$$r_0 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big)\,\mathrm{d}t$$

$$r_1 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big)\,\mathrm{d}t = \frac{1}{2}\big(\boldsymbol{S}_{\alpha_m} \times \boldsymbol{\upsilon}_m\big) - \frac{1}{2}\int_{t_{m-1}}^{t_m} \big(\boldsymbol{S}_\alpha(t) \times \boldsymbol{a}^B_{\mathrm{SF}}\big)\,\mathrm{d}t$$

$$r_2 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big)\,\mathrm{d}t = \frac{1}{2}\big(\boldsymbol{\alpha}_m \times \boldsymbol{S}_{\upsilon_m}\big) + \frac{1}{2}\int_{t_{m-1}}^{t_m} \big(\boldsymbol{S}_\upsilon(t) \times \boldsymbol{\omega}^B_{IB}\big)\,\mathrm{d}t \tag{80}$$

$$\boldsymbol{S}_\alpha(t) = \int_{t_{m-1}}^{t} \boldsymbol{\alpha}(\tau)\,\mathrm{d}t, \qquad \boldsymbol{S}_\upsilon(t) = \int_{t_{m-1}}^{t} \boldsymbol{\upsilon}(\tau)\,\mathrm{d}\tau$$

$$\boldsymbol{\alpha}_m = \boldsymbol{\alpha}(t_m), \qquad \boldsymbol{\upsilon}_m = \boldsymbol{\upsilon}(t_m)$$

$$\boldsymbol{S}_{\alpha_m} = \boldsymbol{S}_\alpha(t_m), \qquad \boldsymbol{S}_{\upsilon_m} = \boldsymbol{S}_\upsilon(t_m)$$

where $\boldsymbol{S}_\alpha$ and $\boldsymbol{S}_\upsilon$ are time integrals of $\boldsymbol{\alpha}$ and $\boldsymbol{\upsilon}$.

Because r_1 and r_2 are analytically equivalent to the original integral form r_0, we can write

$$\int_{t_{m-1}}^{t_m} \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big)\,\mathrm{d}t = \frac{1}{3}(r_0 + r_1 + r_2) \tag{81}$$

Substituting for r_0, r_1 and r_2 from Eq. (80) into Eq. (81) and combining terms then yields

$$\int_{t_{m-1}}^{t_m} \frac{1}{2}\big(\boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big)\,\mathrm{d}t = \frac{1}{6}\big(\boldsymbol{S}_{\alpha_m} \times \boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m \times \boldsymbol{S}_{\upsilon_m}\big) - \frac{1}{6}\int_{t_{m-1}}^{t_m} \big[\boldsymbol{S}_\alpha(t) \times \boldsymbol{a}^B_{\mathrm{SF}} - \boldsymbol{S}_\upsilon(t) \times \boldsymbol{\omega}^B_{IB} - \boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big]\,\mathrm{d}t \tag{82}$$

We now substitute Eq. (82) with the Eq. (80) definitions into Eqs. (77) and (79) to obtain the desired form for calculating $\Delta \boldsymbol{R}^N_m$:

$$\Delta \boldsymbol{R}^N_m = \big(\boldsymbol{v}^N_{m-1} + \tfrac{1}{2}\Delta \boldsymbol{v}^N_{G/\mathrm{Cor}_m}\big)T_m + C^N_L \Delta \boldsymbol{R}^L_{\mathrm{SF}_m}$$

$$\Delta \boldsymbol{R}^L_{\mathrm{SF}_m} = -\tfrac{1}{3}[(\boldsymbol{\zeta}_{n-1,m} - \boldsymbol{\zeta}_{n-1,m-1})\times]\Delta \boldsymbol{v}^{L(n-1)}_{\mathrm{SF}_m} T_m + C^{L(m-1)}_{L(n-1)} C^{L(n-1)}_{B(m-1)} \Delta \boldsymbol{R}^B_{\mathrm{SF}_m} \tag{83}$$

$$\Delta \boldsymbol{R}^B_{\mathrm{SF}_m} = \boldsymbol{S}_{\upsilon_m} + \Delta \boldsymbol{R}_{\mathrm{rot}_m} + \Delta \boldsymbol{R}_{\mathrm{scrl}_m} \tag{84}$$

$$\Delta \boldsymbol{R}_{\mathrm{scrl}_m} = \frac{1}{6}\int_{t_{m-1}}^{t_m} \big[6\Delta \boldsymbol{v}_{\mathrm{scul}}(t) - \boldsymbol{S}_\alpha(t) \times \boldsymbol{a}^B_{\mathrm{SF}} + \boldsymbol{S}_\upsilon(t) \times \boldsymbol{\omega}^B_{IB} + \boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\big]\,\mathrm{d}t \tag{85}$$

with $\Delta \boldsymbol{v}_{\mathrm{scul}}(t)$, $\boldsymbol{\alpha}(t)$, $\boldsymbol{\upsilon}(t)$, $\boldsymbol{\alpha}_m$, and $\boldsymbol{\upsilon}_m$ from sculling equation (38) and

$$\boldsymbol{S}_\alpha(t) = \int_{t_{m-1}}^{t} \boldsymbol{\alpha}(\tau)\,\mathrm{d}\tau, \qquad \boldsymbol{S}_{\alpha m} = \boldsymbol{S}_\alpha(t_m)$$

$$\boldsymbol{S}_\upsilon(t) = \int_{t_{m-1}}^{t} \boldsymbol{\upsilon}(\tau)\,\mathrm{d}\tau, \qquad \boldsymbol{S}_{\upsilon_m} = \boldsymbol{S}_\upsilon(t_m) \tag{86}$$

$$\Delta \boldsymbol{R}_{\mathrm{rot}_m} = \tfrac{1}{6}\big(\boldsymbol{S}_{\alpha_m} \times \boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m \times \boldsymbol{S}_{\upsilon_m}\big)$$

where $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ is position rotation compensation analogous to the velocity rotation compensation term in Eqs. (37) and (39), and $\Delta \boldsymbol{R}_{\mathrm{scrl}_m}$ is the scrolling term analogous to the sculling term in Eqs. (37) and (38). The term *scrolling* was coined by the writer merely to have a name for the term and also to have one that sounds like sculling but for position integration (change in the position vector $\boldsymbol{R}$ stressing the R sound). The complex mathematical derivations and associated algorithms that accompany scrolling may be a more appropriate reason for the name.

A key characteristic of Eq. (84) is that the $\Delta \boldsymbol{R}_{\mathrm{scrl}_m}$ scrolling term from Eq. (85) is identically zero under constant body axis angular rate and specific force conditions. This can be readily verified from Eq. (85) by substituting a constant angular rate and specific force vector for the ω^B_{IB} and $\boldsymbol{a}^B_{\mathrm{SF}}$ terms and carrying out the indicated operations analytically. As such, $\Delta \boldsymbol{R}_{\mathrm{scrl}_m}$ will only produce an output under the presence of dynamic body axis angular rate/specific force components. This is an important characteristic because, for most real dynamic environments, the magnitude of high-frequency angular rate/specific force is small so that first-order approximations accurately apply (first order in integrated body angular rate/specific force over the t_{m-1} to t_m time interval). We conclude that the analytical form for $\Delta \boldsymbol{R}_{\mathrm{scrl}_m}$ will also yield a reasonably accurate solution under situations where the low-frequency body angular rate and specific force components are large.

The Eq. (86) first-order version of position rotation compensation $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ can have noticeable second-order error under extreme maneuvers. The form of Eq. (84) that has $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ separate from other terms allows us to expand $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ to a more accurate form that is exact under constant angular rate/specific force (as in the first subsection of Sec. III.B.2 for the velocity rotation compensation term). The following sections develop algorithms for the exact position rotation compensation term in Eq. (84) and for the scrolling and other integral terms in Eq. (85).

1. *Exact Position Rotation Compensation*

An improved accuracy version of $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ for Eq. (84) is developed by specifying the solution to be exact under constant body angular rate/specific force but to first order, to also equal $\Delta \boldsymbol{R}_{\mathrm{rot}_m}$ in Eq. (86) under general angular rate/specific force conditions. The derivation begins by returning to the basic definition for $\Delta \boldsymbol{R}^B_{\mathrm{SF}_m}$ in Eq. (79):

$$\Delta \boldsymbol{R}^B_{\mathrm{SF}_m} = \int_{t_{m-1}}^{t_m} \Delta \boldsymbol{v}^{B(m-1)}_{\mathrm{SF}}(t)\,\mathrm{d}t \tag{87}$$

As with Eqs. (42) and (44), we now use the exact definition for $\Delta \boldsymbol{v}^{B(m-1)}_{\mathrm{SF}}(t)$ in Eq. (87) based on constant B frame specific force and nonconing angular rate

$$\Delta \boldsymbol{R}^B_{\mathrm{SF}_m} = \int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t} \boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}\tau\,\mathrm{d}t + \big(\boldsymbol{u}_\omega \times \boldsymbol{a}^B_{\mathrm{SF}}\big)\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t} \sin\alpha(\tau)\,\mathrm{d}\tau\,\mathrm{d}t + \big[\boldsymbol{u}_\omega \times \big(\boldsymbol{u}_\omega \times \boldsymbol{a}^B_{\mathrm{SF}}\big)\big]\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t} \big(1 - \cos\alpha(\tau)\big)\,\mathrm{d}\tau\,\mathrm{d}t \tag{88}$$

For constant B frame angular rate (vector direction and magnitude), as in Eq. (47), the integral terms in Eq. (88) can be evaluated for constant B frame angular rate as

$$\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\sin\alpha(\tau)\,\mathrm{d}\tau\,\mathrm{d}t = \frac{1}{\omega^2}(\alpha_m - \sin\alpha_m)$$
$$\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\big(1-\cos\alpha(\tau)\big)\,\mathrm{d}\tau\,\mathrm{d}t = \frac{1}{\omega^2}\left(\frac{1}{2}\alpha_m^2 - (1-\cos\alpha_m)\right) \tag{89}$$

From Eq. (47) we can also write

$$\omega = \alpha_m / T_m \tag{90}$$

so that Eq. (89) for constant B frame angular rate becomes

$$\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\sin\alpha(\tau)\,\mathrm{d}\tau\,\mathrm{d}t = \frac{T_m^2}{\alpha_m}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right)$$
$$\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\big(1-\cos\alpha(\tau)\big)\,\mathrm{d}\tau\,\mathrm{d}t = T_m^2\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right) \tag{91}$$

Applying Eq. (91) in Eq. (88) then obtains for constant B frame angular rate and specific force:

$$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B = \int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}\tau\,\mathrm{d}t + \big(\boldsymbol{u}_\omega\times\boldsymbol{a}_{\mathrm{SF}}^B\big)\frac{T_m^2}{\alpha_m}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right) + \big[\boldsymbol{u}_\omega\times\big(\boldsymbol{u}_\omega\times\boldsymbol{a}_{\mathrm{SF}}^B\big)\big]T_m^2\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right) \tag{92}$$

Equation (92) can be further refined by substitution of $\boldsymbol{S}_{\upsilon_m}$ as defined in Eq. (85) for the double integral, application of the Eq. (45) definitions for appropriate terms, and factorization:

$$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B = \boldsymbol{S}_{\upsilon_m} + \left[\frac{1}{\alpha_m^2}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right)\mathrm{I} + \frac{1}{\alpha_m^2}\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right)(\boldsymbol{\alpha}_m\times)\right](\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m)T_m \tag{93}$$

The $(\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m)T_m$ term in Eq. (93) can be expressed in an alternative form through the following development. Using appropriate definitions from Eq. (85), we find for constant B frame angular rate and specific force that

$$\boldsymbol{S}_{\upsilon_m} = \int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\boldsymbol{a}_{\mathrm{SF}}^B\,\mathrm{d}\tau\,\mathrm{d}t = \boldsymbol{a}_{\mathrm{SF}}^B\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\mathrm{d}\tau\,\mathrm{d}t = \frac{1}{2}\boldsymbol{a}_{\mathrm{SF}}^B T_m^2 = \frac{1}{2}\boldsymbol{\upsilon}_m T_m$$
$$\boldsymbol{S}_{\alpha_m} = \int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\boldsymbol{\omega}_{IB}^B\,\mathrm{d}\tau\,\mathrm{d}t = \boldsymbol{\omega}_{IB}^B\int_{t_{m-1}}^{t_m}\int_{t_{m-1}}^{t}\mathrm{d}\tau\,\mathrm{d}t = \frac{1}{2}\boldsymbol{\omega}_{IB}^B T_m^2 = \frac{1}{2}\boldsymbol{\alpha}_m T_m \tag{94}$$

From Eq. (94) we then can show that $(\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m)T_m$ under the Eq. (93) constant B frame angular rate specific force condition is equivalently

$$(\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m)T_m = \boldsymbol{S}_{\alpha_m}\times\boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m + \boldsymbol{S}_{\upsilon_m} \tag{95}$$

We then substitute Eq. (95) for $(\boldsymbol{\alpha}_m\times\boldsymbol{\upsilon}_m)T_m$ in Eq. (93) to obtain for constant B frame angular rate and specific force

$$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B = \boldsymbol{S}_{\upsilon_m} + \left[\frac{1}{\alpha_m^2}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right)\mathrm{I} + \frac{1}{\alpha_m^2}\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right)(\boldsymbol{\alpha}_m\times)\right]\big(\boldsymbol{S}_{\alpha_m}\times\boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m\times\boldsymbol{S}_{\upsilon_m}\big) \tag{96}$$

Equation (96) is now in a form for defining the exact position rotation compensation term by comparison with Eq. (84) for $\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B$. Under the conditions of constant B frame angular rate and specific force, the $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ term in Eq. (84) is zero, and $\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B$ with Eq. (86) reduces to

$$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B = \boldsymbol{S}_{\upsilon_m} + \Delta\boldsymbol{R}_{\mathrm{rot}_m}, \qquad \Delta\boldsymbol{R}_{\mathrm{rot}_m} = \tfrac{1}{6}\big(\boldsymbol{S}_{\alpha_m}\times\boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m\times\boldsymbol{S}_{\upsilon_m}\big) \tag{97}$$

We also note that by applying Taylor series expansion to the trigonometric terms [as shown subsequently in Eq. (99)] Eq. (96) to first order is given by

$$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B = \boldsymbol{S}_{\upsilon_m} + \tfrac{1}{6}\big(\boldsymbol{S}_{\alpha_m}\times\boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m\times\boldsymbol{S}_{\upsilon_m}\big) \tag{98}$$

Finally, we compare Eq. (96) and its Eq. (98) first-order version with Eq. (97) to deduce the sought-after exact position rotation compensation algorithm. Including trigonometric expansion formulas, the result is

$$\Delta\boldsymbol{R}_{\mathrm{rot}_m} = \left[\frac{1}{\alpha_m^2}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right)\mathrm{I} + \frac{1}{\alpha_m^2}\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right)(\boldsymbol{\alpha}_m\times)\right]\big(\boldsymbol{S}_{\alpha_m}\times\boldsymbol{\upsilon}_m + \boldsymbol{\alpha}_m\times\boldsymbol{S}_{\upsilon_m}\big)$$
$$\frac{1}{\alpha_m^2}\left(1-\frac{\sin\alpha_m}{\alpha_m}\right) = \frac{1}{3!} - \frac{\alpha_m^2}{5!} + \frac{\alpha_m^4}{7!} - \cdots$$
$$\frac{1}{\alpha_m^2}\left(\frac{1}{2}-\frac{(1-\cos\alpha_m)}{\alpha_m^2}\right) = \frac{1}{4!} - \frac{\alpha_m^2}{6!} + \frac{\alpha_m^4}{8!} - \cdots \tag{99}$$

Equations (99) can be utilized in Eq. (84) in place of $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ from Eq. (86) to obtain the equivalent higher-order equation for $\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B$ that is exact under constant body angular rate/specific force conditions.

2. *Scrolling and Other Integral Term Increments*

The computer algorithms used to implement the integration operations in Eq. (85) are executed at high computer repetition rate, i.e., the sculling l cycle rate, within the position update m cycle. The $\boldsymbol{\upsilon}(t)$, $\boldsymbol{\upsilon}_m$, $\boldsymbol{\alpha}(t)$, and $\boldsymbol{\alpha}_m$ integral terms in Eq. (85) are provided by Eqs. (54) and (55). The remaining integral terms in Eq. (85) can be rewritten to reflect the high-speed computing cycle as follows:

$$\boldsymbol{S}_\alpha(t) = \boldsymbol{S}_{\alpha_{l-1}} + \Delta\boldsymbol{S}_\alpha(t)$$
$$\Delta\boldsymbol{S}_\alpha(t) = \int_{t_{l-1}}^{t}\boldsymbol{\alpha}(\tau)\,\mathrm{d}\tau, \qquad \Delta\boldsymbol{S}_{\alpha_l} = \Delta\boldsymbol{S}_\alpha(t_l)$$
$$\boldsymbol{S}_{\alpha_l} = \boldsymbol{S}_{\alpha_{l-1}} + \Delta\boldsymbol{S}_{\alpha_l}, \qquad \boldsymbol{S}_{\alpha_m} = \boldsymbol{S}_{\alpha_l}(t_l = t_m)$$
$$\boldsymbol{S}_{\alpha_l} = 0 \quad \text{at} \quad t = t_{m-1}$$
$$\boldsymbol{S}_\upsilon(t) = \boldsymbol{S}_{\upsilon_{l-1}} + \Delta\boldsymbol{S}_\upsilon(t) \tag{100}$$
$$\Delta\boldsymbol{S}_\upsilon(t) = \int_{t_{l-1}}^{t}\boldsymbol{\upsilon}(\tau)\,\mathrm{d}\tau, \qquad \Delta\boldsymbol{S}_{\upsilon_l} = \Delta\boldsymbol{S}_\upsilon(t_l)$$
$$\boldsymbol{S}_{\upsilon_l} = \boldsymbol{S}_{\upsilon_{l-1}} + \Delta\boldsymbol{S}_{\upsilon_l}, \qquad \boldsymbol{S}_{\upsilon_m} = \boldsymbol{S}_{\upsilon_l}(t_l = t_m)$$
$$\boldsymbol{S}_{\upsilon_l} = 0 \quad \text{at} \quad t = t_{m-1}$$

$$\Delta \boldsymbol{R}_{\mathrm{scrl}_l} = \Delta \boldsymbol{R}_{\mathrm{scrl}_{l-1}} + \delta \boldsymbol{R}_{\mathrm{scrl}_l}, \qquad \Delta \boldsymbol{R}_{\mathrm{scrl}_m} = \Delta \boldsymbol{R}_{\mathrm{scrl}_l}(t = t_m)$$
$$\Delta \boldsymbol{R}_{\mathrm{scrl}_l} = 0 \quad \text{at} \quad t = t_{m-1}$$
$$\delta \boldsymbol{R}_{\mathrm{scrl}_l} = \frac{1}{6}\int_{t_{l-1}}^{t_l} \Big[6\,\Delta \boldsymbol{v}_{\mathrm{scul}}(t) - \boldsymbol{S}_\alpha(t) \times \boldsymbol{a}_{\mathrm{SF}}^B(t)$$
$$+ \boldsymbol{S}_\upsilon(t) \times \boldsymbol{\omega}_{IB}^B(t) + \boldsymbol{\alpha}(t) \times \boldsymbol{\upsilon}(t)\Big]\,\mathrm{d}t \tag{101}$$
$$\Delta \boldsymbol{v}_{\mathrm{scul}}(t) = \Delta \boldsymbol{v}_{\mathrm{scul}_{l-1}} + \delta \boldsymbol{v}_{\mathrm{scul}}(t)$$
$$\Delta \boldsymbol{v}_{\mathrm{scul}_l} = \Delta \boldsymbol{v}_{\mathrm{scul}}(t_l), \qquad \Delta \boldsymbol{v}_{\mathrm{scul}_l} = 0 \quad \text{at} \quad t = t_{m-1}$$

with $\delta \boldsymbol{v}_{\mathrm{scul}}(t)$ from sculling equation (56).

As in the second subsection of Sec. III.B.2 for the velocity sculling algorithm and other integral terms, algorithms can be designed for the integral terms in Eqs. (100) and (101) to be analytically exact under assumed forms of the angular rate and specific force profile within the l cycle. Coefficients for the angular rate/specific force profiles are then determined from sequential integrated angular rate/specific force increments taken at the l cycle rate (or, alternatively, at a higher-speed sensor sampling rate within the l cycle). As an example of the l cycle sensor sampling method, Ref. 8, Sec. 7.3.3.1.2, develops algorithms for the Eqs. (100) and (101) integral terms based on generalized linearly ramping angular rate and specific force conditions. The overall results are given by

$$\begin{aligned} \Delta\boldsymbol{\alpha}_l, \boldsymbol{\alpha}_l &= \text{integrated angular rate sensor outputs from Ref. 1, Eqs. (46)} \\ \Delta\boldsymbol{\alpha}_l, \boldsymbol{\upsilon}_l &= \text{integrated accelerometer outputs from algorithm Eqs. (61)} \end{aligned} \tag{102}$$

$$\Delta \boldsymbol{S}_{\alpha_l} = \boldsymbol{\alpha}_{l-1} T_l + (T_l/12)(5\,\Delta\boldsymbol{\alpha}_l + \Delta\boldsymbol{\alpha}_{l-1})$$
$$\boldsymbol{S}_{\alpha_l} = \boldsymbol{S}_{\alpha_{l-1}} + \Delta \boldsymbol{S}_{\alpha_l}, \qquad \boldsymbol{S}_{\alpha_m} = \boldsymbol{S}_{\alpha_l}(t_l = t_m) \tag{103}$$
$$\boldsymbol{S}_{\alpha_l} = 0 \quad \text{at} \quad t = t_{m-1}$$

$$\Delta \boldsymbol{S}_{\upsilon_l}^B = \boldsymbol{\upsilon}_{l-1}^B T_l + (T_l/12)\big(5\,\Delta\boldsymbol{\upsilon}_l^B + \Delta\boldsymbol{\upsilon}_{l-1}^B\big)$$
$$\boldsymbol{S}_{\upsilon_l} = \boldsymbol{S}_{\upsilon_{l-1}} + \Delta \boldsymbol{S}_{\upsilon_l}, \qquad \boldsymbol{S}_{\upsilon_m} = \boldsymbol{S}_{\upsilon_l}(t_l = t_m) \tag{104}$$
$$\boldsymbol{S}_{\upsilon_l} = 0 \quad \text{at} \quad t = t_{m-1}$$

$$\Delta \boldsymbol{R}_{\mathrm{scrl}_l} = \Delta \boldsymbol{R}_{\mathrm{scrl}_{l-1}} + \delta \boldsymbol{R}_{\mathrm{scrl}A_l} + \delta \boldsymbol{R}_{\mathrm{scrl}B_l}$$
$$\delta \boldsymbol{R}_{\mathrm{scrl}A_l} = \Delta \boldsymbol{v}_{\mathrm{scul}_{l-1}} T_l$$
$$+ \tfrac{1}{2}\big[\boldsymbol{\alpha}_{l-1} - \tfrac{1}{12}(\Delta\boldsymbol{\alpha}_l - \Delta\boldsymbol{\alpha}_{l-1})\big] \times \big(\Delta \boldsymbol{S}_{\upsilon_l} - \boldsymbol{\upsilon}_{l-1} T_l\big)$$
$$+ \tfrac{1}{2}\big[\boldsymbol{\upsilon}_{l-1} - \tfrac{1}{12}\Delta\boldsymbol{\upsilon}_l - \Delta\boldsymbol{\upsilon}_{l-1}\big] \times \big(\Delta \boldsymbol{S}_{\alpha_l} - \boldsymbol{\alpha}_{l-1} T_l\big)$$
$$\delta \boldsymbol{R}_{\mathrm{scrl}B_l} = \tfrac{1}{6}\big[\boldsymbol{S}_{\upsilon_{l-1}} + (T_l/24)(\Delta\boldsymbol{\upsilon}_l - \Delta\boldsymbol{\upsilon}_{l-1})\big] \times \Delta\boldsymbol{\alpha}_l \tag{105}$$
$$- \tfrac{1}{6}\big[\boldsymbol{S}_{\alpha_{l-1}} + (T_l/24)(\Delta\boldsymbol{\alpha}_l - \Delta\boldsymbol{\alpha}_{l-1})\big] \times \Delta\boldsymbol{\upsilon}_l$$
$$+ (T_l/6)\big[\alpha_{l-1} - \tfrac{1}{6}(\Delta\boldsymbol{\alpha}_l - \Delta\boldsymbol{\alpha}_{l-1})\big]$$
$$\times \big[\boldsymbol{\upsilon}_{l-1} - \tfrac{1}{6}(\Delta\boldsymbol{\upsilon}_l - \Delta\boldsymbol{\upsilon}_{l-1})\big]$$
$$- (T_l/2160)(\Delta\boldsymbol{\alpha}_l - \Delta\boldsymbol{\alpha}_{l-1}) \times (\Delta\boldsymbol{\upsilon}_l - \Delta\boldsymbol{\upsilon}_{l-1})$$
$$\Delta \boldsymbol{R}_{\mathrm{scrl}_m} = \Delta \boldsymbol{R}_{\mathrm{scrl}_l}(t_l = t_m), \qquad \Delta \boldsymbol{R}_{\mathrm{scrl}_l} = 0 \quad \text{at} \quad t = t_{m-1}$$

with $\Delta \boldsymbol{v}_{\mathrm{scul}_{l-1}}$ from the Eq. (61) sculling algorithm and where

$\delta \boldsymbol{R}_{\mathrm{scrl}A_l}$ = portion of $\delta \boldsymbol{R}_{\mathrm{scrl}_l}$ produced by the $\delta \boldsymbol{v}_{\mathrm{scul}}^B$ sculling term
$\delta \boldsymbol{R}_{\mathrm{scrl}B_l}$ = portion of $\delta \boldsymbol{R}_{\mathrm{scrl}_l}$ produced by all but the $\delta \boldsymbol{v}_{\mathrm{scul}}^B$ sculling term
T_l = high-speed computer update time interval t_l–t_{l-1}

Equations (105) can be classified as a second-order algorithm for $\delta \boldsymbol{R}_{\mathrm{scrl}_l}$ because they include current and past cycle $\Delta\boldsymbol{\alpha}_l$, $\Delta\boldsymbol{\upsilon}_l$ products. If the angular rate/specific force profile was approximated as constant over two successive l cycles, the $(\Delta\boldsymbol{\alpha}_l - \Delta\boldsymbol{\alpha}_{l-1})$ and $(\Delta\boldsymbol{\upsilon}_l - \Delta\boldsymbol{\upsilon}_{l-1})$ terms in Eq. (105) would vanish, resulting in a first-order $\delta \boldsymbol{R}_{\mathrm{scrl}_l}$ algorithm. Under conditions where the angular rate and specific force can be approximated as constant, i.e., slowly varying over an m cycle, $\Delta \boldsymbol{R}_{\mathrm{scrl}_l}$ in Eq. (105) is approximately zero and the $\Delta \boldsymbol{R}_{\mathrm{scrl}_l}$, $\delta \boldsymbol{R}_{\mathrm{scrl}A_l}$, $\delta \boldsymbol{R}_{\mathrm{scrl}B_l}$ calculations in Eq. (105) can be deleted. Alternatively (and more accurately), for slowly varying angular rate and specific force, one l cycle of Eq. (105) can be executed each m cycle, noting from the initial condition definitions that $\boldsymbol{\alpha}_{l-1}$, $\boldsymbol{\upsilon}_{l-1}$, $\boldsymbol{S}_{\alpha_{l-1}}$, and $\boldsymbol{S}_{\upsilon_{l-1}}$ are zero. As noted in the second subsection of Sec. III.B.2, setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high-speed, higher-order algorithm with a simpler software architecture than the two-speed approach but requiring more throughput. Continuing advances in the speed of modern-day computers may make this the preferred approach for the future.

V. Velocity/Position Integration Algorithm Summary

Table 1 is a summary of the algorithms described for the strapdown inertial navigation velocity/position integration function listed in the order that they would be executed in the navigation computer. Note in Table 1 that the normal speed attitude calculation follows the normal speed position calculation, in contrast to Table 1 of Ref. 1, which calculates attitude before position. Having the attitude follow the position calculation allows the high-resolution $\Delta \boldsymbol{R}_m^N$ from Eq. (77) to be used in Ref. 1, Eq. (53), rather than the less-accurate Ref. 1, Eq. (56), trapezoidal algorithm form of $\Delta \boldsymbol{R}_m^N$.

VI. Algorithm and Execution Rate Selection

Section VI of Part 1 (Ref. 1) discusses the general process of algorithm selection for a given application with required execution rates to achieve specified accuracy goals. A principal part of this process involves estimating the algorithm error under anticipated angular rate/specific force maneuvers/vibrations compared with specified error budget requirements. Evaluation of candidate algorithm error characteristics is generally performed using computerized time domain simulators that exercise the algorithms, in particular groupings, at their selected repetition rates. The simulators generate strapdown inertial sensor angular rate/specific force profiles for algorithm test input together with known navigation parameter solutions for algorithm output comparison, e.g., Ref. 8, Secs. 11.2.1–11.2.4.

For the two-speed velocity/position updating approach described, the repetition rate for the moderate speed (m cycle) algorithms would typically be selected based on maximum angular rate/specific force considerations to minimize power series truncation error in the moderate- and high-speed algorithms. The repetition rate for the high-speed (l cycle) algorithms would typically be selected based on the anticipated strapdown inertial sensor assembly vibration environment to accurately account for vibration induced sculling/scrolling effects.

For the velocity algorithms, simplified analytical error models can also be used to predict high-speed sculling algorithm error under selected sculling rates/amplitudes as a function of algorithm repetition rate (Refs. 5–7 and 8, Chap. 10). The sculling rates/amplitudes must be derived either from empirical data or, more commonly, from analytical models of the sensor assembly mount imbalance and its response to external input vibration at particular frequencies (Ref. 8, Chap. 10). Frequency-domain simulators can be used to evaluate high-speed sculling algorithm error under specified input vibration power spectral density profiles and sensor assembly mount imbalance as a function of algorithm repetition rate (Ref. 8, Chap. 10). For example, the sculling algorithm described by Eqs. (59–61) can be shown by such simulators to have an error of 0.044 μg when operated at a 2-kHz repetition rate under exposure to 7.6 g rms wideband random linear input vibration (flat 0.04 g^2/Hz density from 20 to 1000 Hz, then decreasing logarithmically to 0.01 g^2/Hz at 2000 Hz). The linear vibration generates a multiaxis 3.6 g/0.00038 rad rms specific force/angular oscillation of the sensor assembly with a corresponding rectified sculling acceleration of 1300 μg due to the

Table 1 Summary of strapdown INS velocity/position computation algorithms

Algorithm function	Input	Output	Equation number
High-speed calculations			
Integrated B frame angular rate increments	——	$\boldsymbol{\alpha}_l$, $\boldsymbol{\alpha}_m$	Ref. 1, Table 1
Integrated B frame specific force increments	$\Delta\boldsymbol{\upsilon}_l$	$\boldsymbol{\upsilon}_l$, $\boldsymbol{\upsilon}_m$	(55) or (60)
Sculling increment	$\Delta\boldsymbol{\alpha}_l$, $\boldsymbol{\alpha}_l$, $\Delta\boldsymbol{\upsilon}_l$, $\boldsymbol{\upsilon}_l$	$\Delta\boldsymbol{v}_{\mathrm{scul}_l}$, $\Delta\boldsymbol{v}_{\mathrm{scul}_m}$	(56) or (61)
Doubly integrated B frame angular rate and specific force increments (for high-resolution position algorithm)	$\Delta\boldsymbol{\alpha}_l$, $\boldsymbol{\alpha}_l$ $\Delta\boldsymbol{\upsilon}_l$, $\boldsymbol{\upsilon}_l$	$\Delta\boldsymbol{S}_{\alpha_l}$, $\boldsymbol{S}_{\alpha_l}$ $\boldsymbol{S}_{\alpha_m}$, $\Delta\boldsymbol{S}_{\upsilon_l}$ $\boldsymbol{S}_{\upsilon_l}$, $\boldsymbol{S}_{\upsilon_m}$	(100) or (103), (104)
Scrolling increment (for high-resolution position algorithm)	$\Delta\boldsymbol{\alpha}_l$, $\boldsymbol{\alpha}_l$, $\Delta\boldsymbol{S}_{\alpha_l}$, $\boldsymbol{S}_{\alpha_l}$ $\Delta\boldsymbol{\upsilon}_l$, $\boldsymbol{\upsilon}_l$, $\Delta\boldsymbol{S}_{\upsilon_l}$, $\boldsymbol{S}_{\upsilon_l}$, $\Delta\boldsymbol{v}_{\mathrm{scul}_l}$	$\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$	(101) or (105)
Normal-speed calculations for Earth-related parameters			
N frame plumb-bob gravity components	C_N^E, h	$\boldsymbol{g}_P^N$	Ref. 1, Eq. (19)
N frame Earth rate components	C_N^E	$\boldsymbol{\omega}_{IE}^N$	(2)
Vertical transport rate component	C_N^E	ρ_{ZN}	Ref. 8, Sec. 4.6
Curvature matrix	C_N^E, h	F_C	Ref. 8, Sec. 5.1.3
Normal-speed velocity calculations			
B frame velocity rotation compensation (exact formulation)	$\boldsymbol{\alpha}_m$, $\boldsymbol{\upsilon}_m$	$\Delta\boldsymbol{v}_{\mathrm{rot}_m}$	(51), (52)
B frame velocity rotation compensation (first-order approximation form)	$\boldsymbol{\alpha}_m$, $\boldsymbol{\upsilon}_m$	$\Delta\boldsymbol{v}_{\mathrm{rot}_m}$	(39)
B frame integrated specific force increment	$\boldsymbol{\upsilon}_m$, $\Delta\boldsymbol{v}_{\mathrm{rot}_m}$, $\Delta\boldsymbol{v}_{\mathrm{scul}_m}$	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$	(37)
L frame integrated specific force increment	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{B_{I(m-1)}}$, $C_{B_{I(m-1)}}^{L_{I(n-1)}}$	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{L_{I(n-1)}}$	(11)
L frame rotation vector (cycle $n-1$ to m)	$\boldsymbol{\omega}_{IE}^N$, ρ_{ZN}, F_C, $\boldsymbol{v}^N$	$\boldsymbol{\zeta}_{n-1,m}$	(17), (19), (20)
L frame rotation matrix (first-order form)	$\boldsymbol{\zeta}_{n-1,m}$	$C_{L_{I(n-1)}}^{L_{I(m)}}$	(14)
L frame rotation compensation	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{L_{I(n-1)}}$, $C_{L_{I(n-1)}}^{L_{I(m)}}$	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{L}$	(13)
Integrated Coriolis acceleration and plumb-bob gravity increment	$\boldsymbol{g}_P^N$, $\boldsymbol{\omega}_{IE}^N$ ρ_{ZN}, F_C, $\boldsymbol{v}^N$	$\Delta\boldsymbol{v}_{G/\mathrm{Cor}_m}^N$	(7), (8), (9)
N frame velocity update	$\Delta\boldsymbol{v}_{\mathrm{SF}_m}^L$, $\Delta\boldsymbol{v}_{G/\mathrm{Cor}_m}^N$, $\boldsymbol{v}_{m-1}^N$	$\boldsymbol{v}_m^N$	(4)
Normal-speed position calculations			
Position rotation compensation (high-resolution position algorithm, exact form)	$\boldsymbol{\alpha}_m$, $\boldsymbol{S}_{\alpha_m}$ $\boldsymbol{\upsilon}_m$, $\boldsymbol{S}_{\upsilon_m}$	$\Delta\boldsymbol{R}_{\mathrm{rot}_m}$	(99)
Position rotation compensation (high-resolution position algorithm, first-order accuracy form)	$\boldsymbol{\alpha}_m$, $\boldsymbol{S}_{\alpha_m}$ $\boldsymbol{\upsilon}_m$, $\boldsymbol{S}_{\upsilon_m}$	$\Delta\boldsymbol{R}_{\mathrm{rot}_m}$	(86)
Body frame position increment due to specific force (high-resolution position algorithm)	$\boldsymbol{S}_{\upsilon_m}$, $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$	$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B$	(84)
N frame position increment (high-resolution position algorithm)	$\Delta\boldsymbol{R}_{\mathrm{SF}_m}^B$, $\boldsymbol{v}_{m-1}^N$ $\Delta\boldsymbol{v}_{G/\mathrm{Cor}_m}^N$, $\Delta\boldsymbol{v}_{\mathrm{SF}_m}^{L(n-1)}$ $C_{B(m-1)}^{L(n-1)}$, $\boldsymbol{\zeta}_{n-1,m}$, $C_{L(n-1)}^{L(m-1)}$	$\Delta\boldsymbol{R}_m^N$	(83)
N frame position increment (trapezoidal position algorithm)	$\boldsymbol{v}_m^N$	$\Delta\boldsymbol{R}_m^N$	(75)
Altitude change	$\Delta\boldsymbol{R}_m^N$	Δh_n	(66)
Position rotation vector	ρ_{ZN}, F_C, $\Delta\boldsymbol{R}_m^N$	$\boldsymbol{\xi}_n$	(73)
Position rotation change matrix	$\boldsymbol{\xi}_n$	$C_{N_{E(n-1)}}^{N_{E(n)}}$	(70)
Altitude update	h_{n-1}, Δh_n	h_n	(64)
Position direction cosine matrix update	$C_{N_{E(n-1)}}^E$, $C_{N_{E(n)}}^{N_{E(n-1)}}$	$C_{N_{E(n)}}^E$	(68)
Normal-speed attitude calculations			
Attitude direction cosine matrix update	——	C_B^L	Ref. 1, Table 1

following typical sensor assembly mount characteristics selected as simulator input parameters: 50-Hz linear vibration mode undamped natural frequency, 0.125 linear vibration mode damping ratio, 71-Hz rotary vibration mode undamped natural frequency, 0.18 rotary vibration mode damping ratio, 5% sensor assembly mount/isolator spring/damping imbalance, and 1.4% sensor assembly center of mass offset from isolator/mount center of force (percent of distance between isolators).

The capabilities of modern-day computers and INS software technology make it reasonable to specify that the navigation algorithm error be no greater than 5% of the equivalent error produced by the INS inertial sensors (whose cost increases dramatically with accuracy demands). For an INS with a 40-μg accelerometer bias accuracy requirement (typical for an aircraft INS having 2–3 fps 1 σ velocity accuracy), the 0.044-μg sculling algorithm error is almost two orders of magnitude within the 5% allowance, providing a wide design margin for the algorithm 2-kHz repetition rate selection. For this case, a 1-kHz sculling algorithm rate would probably be more appropriate; however, 2 kHz might still be utilized for compatibility with the 2-kHz rate selected for the coning algorithm in Part 1 (Ref. 1) under the same conditions.

In the case of the positioning algorithms, the typical form presented in Sec. IV.B is usually adequate for almost all applications (to date). For the exceptional cases where very high-resolution position updating is required, the time interval for the accuracy requirement is usually restricted to brief periods during the application mission profile. Moreover, for some of these applications, postprocessing is acceptable using data recorded during the high-resolution time interval; hence, the complexity of the high-resolution algorithms would not be a real-time computer throughput issue. For example, for synthetic aperture radar (SAR) motion compensation, high-resolution position data are required for only brief intervals, e.g., 5–10 s, during SAR data acquisition, which may then be subsequently processed for SAR image formation. We also note that, in high-resolution applications, the Earth-referenced position of the INS chassis/mount is usually the required output, which equals the sum of Earth-referenced inertial sensor assembly position (calculated by the inertial navigation algorithms) plus vibration/specific force induced displacement of the sensor assembly relative to the INS chassis/mount (due to compliance of elastomeric isolators that interface the sensor assembly to the INS chassis). The latter displacement can be computed under dynamic maneuvers by quasistatic flexure modeling, i.e., displacement equals average specific force times the square of the sensor assembly/isolator undamped natural frequency, and by appropriate digital filtering of vibration-induced jitter (Ref. 8, Chap. 9). Note that, in principle, the displacement can also be measured directly using specially installed sensing devices.

As an example of the inertial navigation position integration algorithm selection process, let us consider a high-resolution application with an overall INS requirement for position error fluctuations to be significantly less than 1 cm during 5–10 s periods (not unusual for applications where the actual requirement is a function of error frequency content and not clearly known). Allowing design margin for error in the sensor assembly to chassis/mount flexure displacement calculation (described in the preceding paragraph), we budget the INS accuracy specification into a requirement for the position algorithm to have less than 0.01-cm dynamic position error fluctuation during 5–10 s. Let us further assume for this example that the basic position algorithm update rate has been selected to be 50 Hz and that the selected inertial velocity algorithm accuracy is compatible with high-resolution position updating requirements, e.g., includes high-rate sculling. Simplified pencil-and-paper analysis of the typical form equation (75) position algorithm (or other versions) can be used to assess its accuracy at 50 Hz using the high-resolution algorithm to represent the correct truth model. An analytical model for the high-resolution $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ increment truth model can be derived using Eq. (25) for $\Delta\boldsymbol{v}^B_{\mathrm{SF}}$ in Eq. (79):

$$\Delta\boldsymbol{R}^B_{\mathrm{SF}_m} = \int_{t_{m-1}}^{t_m} \Delta\boldsymbol{v}^B_{\mathrm{SF}}(\tau)\,\mathrm{d}t$$
$$\Delta\boldsymbol{v}^B_{\mathrm{SF}}(\tau) = \int_{t_{m-1}}^{\tau} \Big[\mathrm{I} + \big(\boldsymbol{\alpha}(t)\,\times\big)\Big]\boldsymbol{a}^B_{\mathrm{SF}}\,\mathrm{d}t \tag{106}$$

Neglecting the small L frame rotation effect, it can be shown that position updating based on the Eq. (75) typical algorithm is equivalent to Eq. (79) with $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ replaced by the typical algorithm equivalent $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ given by

$$\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m} = \tfrac{1}{2}\Delta\boldsymbol{v}^B_{\mathrm{SF}_m}T_m, \qquad \Delta\boldsymbol{v}^B_{\mathrm{SF}_m} = \Delta\boldsymbol{v}^B_{\mathrm{SF}}(t_m) \tag{107}$$

For the Eq. (58) linearly ramping specific force/angular rate model in Eqs. (106) and (107), the position increments for the truth model $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ and for the typical algorithm $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ become

$$\begin{aligned}\Delta\boldsymbol{R}^B_{\mathrm{SF}_m} &= \tfrac{1}{2}\boldsymbol{C}T_m^2 + \tfrac{1}{6}(\boldsymbol{D}+\boldsymbol{A}\times\boldsymbol{C})T_m^3\\ &\quad + \tfrac{1}{12}\big(\boldsymbol{A}\times\boldsymbol{D} + \tfrac{1}{2}\boldsymbol{B}\times\boldsymbol{C}\big)T_m^4 + \tfrac{1}{40}\boldsymbol{B}\times\boldsymbol{D}T_m^5\\ \Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m} &= \tfrac{1}{2}\boldsymbol{C}T_m^2 + \tfrac{1}{4}\big(\boldsymbol{D}+\boldsymbol{A}\times\boldsymbol{C}\big)T_m^3\\ &\quad + \tfrac{1}{6}\big(\boldsymbol{A}\times\boldsymbol{D} + \tfrac{1}{2}\boldsymbol{B}\times\boldsymbol{C}\big)T_m^4 + \tfrac{1}{16}\boldsymbol{B}\times\boldsymbol{D}T_m^5\end{aligned} \tag{108}$$

Comparing $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ with the $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ truth model in Eq. (108) allows the error in $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ to be assessed for selected maneuver values. Under a constant $\boldsymbol{C}$ specific force maneuver, $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ equals $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ and, hence, is error free. For $\boldsymbol{D} = 3$ g/s or for $\boldsymbol{C} = 3$ g with $\boldsymbol{A} = 1$ rad/s, the calculated error in $\Delta\boldsymbol{R}^B_{\mathrm{SF/typ}_m}$ (using T_m = 0.02 s for the 50-Hz update rate) is 0.00196 cm or 50 × 0.00196 = 0.098 cm in 1 s. Compared with the 0.01 cm in 5–10 s requirement, the 0.098 cm in 1 s figure would be considered unacceptable.

Position algorithm assessment under vibration can also be analytically estimated. For example, for the 3.6 g rms sensor assembly vibration (in the preceding sculling example), the associated velocity vibration is 11.2 cm/s rms centered around the sensor assembly 50-Hz mount resonance (which would be accurately measured by the hypothesized velocity algorithm). The aliasing error associated with sampling the vibrating velocity at 50 Hz for the Eq. (75) algorithm can produce a 11.2 × 0.02 = 0.22 cm error each position update. If the error is random per update, the total cumulative error in 1 s (50 updates) would be $0.22 \times \sqrt{50} = 1.6$ cm; if the error is systematic, the position error in 1 s would be 0.22 × 50 = 11.2 cm. In either case, the algorithm error greatly exceeds the 0.01 cm over 5–10 s requirement.

Based on such analyses, let us assume we have elected to use the Eqs. (83) and (84) high-resolution position algorithm to assure 5–10 s, 0.01-cm high-quality resolution. The next question is which terms in Eq. (84) are to be included. The $\boldsymbol{S}_{\upsilon m}$ term in Eq. (84) is the dominant term for integrating velocity into position and must be included. Under a 3-g constant specific force maneuver, $\boldsymbol{S}_{\upsilon m}$ from Eq. (85) equals 0.59 cm per 50-Hz position update cycle or 29.4 cm in 1 s. The next most important term is the $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ position rotation compensation term. Using Eq. (86) with Eq. (85) input, the magnitude of $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ under a constant 3 g/1 rad/s maneuver is 0.0039 cm per update cycle or 0.20-cm cumulative position change in 1 s. (Note, for a 3-g/s linearly ramping specific force, $\boldsymbol{S}_{\upsilon m}$ also equals 0.0039 cm per cycle and sums to 0.20 cm in 1 s.) For the 0.01 cm over 5–10 s requirement, the $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ term is, therefore, also needed. The question of whether to include the $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ term can be addressed by analyzing the magnitude of $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ under dynamic vibration motion using a rearranged version of Eq. (84):

$$\Delta\boldsymbol{R}_{\mathrm{scrl}_m} = \Delta\boldsymbol{R}^B_{\mathrm{SF}_m} - \boldsymbol{S}_{\upsilon m} - \Delta\boldsymbol{R}_{\mathrm{rot}_m} \tag{109}$$

Consider the 3.6-g rms vibration condition under 1-rad/s constant angular rate. For a 3.6-g rms pure sine wave, i.e., $3.6 \times \sqrt{2} = 5.1$-$g$ amplitude, at the 50-Hz isolator resonance frequency, the magnitudes of $\boldsymbol{S}_{\upsilon m}$ and $\boldsymbol{\upsilon}_m$ over 0.02 s are, from Eqs. (85), 0.32 cm and 0 cm/s, respectively. For the 1 rad/s rate over 0.02 s, α_m is 0.02 rad. Thus, from Eq. (86), $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ is (0.32 × 0.02)/6 = 0.0011 cm, which, if systematic, accumulates in 1 s to 0.0011 × 50 = 0.053 cm. If random from cycle to cycle, the error accumulation over 10 s would be $0.0011 \times \sqrt{(50 \times 10)} = 0.024$ cm. The true solution $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m}$ for this particular case can be demonstrated by analytical integration of Eq. (106) to be $\Delta\boldsymbol{R}^B_{\mathrm{SF}_m} = \boldsymbol{S}_{\upsilon m}$. Thus, from Eq. (109) and the latter $\Delta\boldsymbol{R}_{\mathrm{rot}_m}$ analyses, the cumulative magnitude of $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ is 0.053 cm/s (if systematic) and 0.024 cm over 10 s (if random). To meet the accuracy requirement of 0.01 cm over 5–10 s, we conclude that $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ will also be required. The final question is which particular terms in the Eq. (105) $\Delta\boldsymbol{R}_{\mathrm{scrl}_m}$ algorithm are needed. The answer can be obtained from similar individual analyses of each term in Eq. (105) to identify which are significant relative to the requirement. A simpler approach is to arbitrarily, but conservatively, use the full Eq. (105) form. The rationale might be that the savings in using a simplified version, e.g., without the second-order terms, is not worth the time and cost for justification, assuming computer throughput is not an issue. The latter approach has additional merit because it totally frees the system designer of concern for INS algorithm error during the development optimization process of the system using the INS.

The position algorithm selection process just described is fairly rudimentary, admittedly conservative, but sufficient if the outcome is the conservative approach of applying the full high-resolution algorithm, particularly if the accuracy requirement cannot be clearly defined. Had the choice been to use the typical algorithm or an alternate version thereof, a more sophisticated process would have been required to assure adequate performance over a more accurate and complete set of defined operating conditions. For example, complex maneuver/vibration profiles can be simulated and input to the trial algorithm, with its accuracy evaluated using the high-resolution algorithm (with the same input) as a reference. In this regard, the high-resolution algorithm can be viewed as a truth model for position algorithm evaluation but available for use if the trial algorithm is inadequate. An assessment of the need to include particular terms in the scrolling portion of the high-resolution algorithm can be made similarly by calculating the magnitude of each term under simulated input vs error allowances. (A term is needed if its magnitude exceeds the allowance.) The latter step can be augmented using analytical models for input conditions, similar to the approach described in the last example.

VII. Concluding Remarks

Reference 1 defined requirements for the strapdown INS integration algorithms in the form of continuous differential equations and developed the attitude integration algorithms. In Part 2, we have presented a comprehensive design process for development of the specific force transformation/velocity integration and position integration algorithms based on the two-speed updating approach described in Part 1 (Ref. 1) for attitude integration: use of an exact moderate-speed algorithm for the basic integration function fed by a high-speed algorithm to measure high-frequency rectification effects. The moderate-speed algorithms are analytically exact under constant angular rate/specific force; the high-speed algorithms account for deviations from constant angular rate/specific force (sculling for the velocity algorithm and scrolling for the position algorithm). Where computer throughput restrictions are not an issue, the two-speed structure can be compressed into a single high-speed format by operating the moderate-speed algorithm at the high-speed rate. A summary of the velocity/position integration algorithms developed herein is provided in Table 1 as a listing in the order they would be executed in the navigation computer. A similar table is provided in Part 1 (Ref. 1) for the attitude integration algorithms.

References

[1]Savage, P. G., "Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 1, 1998, pp. 19–28.

[2]Savage, P. G., "A New Second-Order Solution for Strapped-Down Attitude Computation," AIAA/JACC Guidance and Control Conf., Seattle, Washington, Aug. 1966.

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

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

[5]Savage, P. G., "Strapdown System Algorithms," *Advances in Strapdown Inertial Systems*, NATO AGARD Lecture Series No. 133, May 1984.

[6]Ignagni, M. B., "Optimal Strapdown Attitude Integration Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 2, 1990, pp. 363–369.

[7]Ignagni, M. B., "Efficient Class of Optimized Coning Compensation Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 2, 1996, pp. 424–429.

[8]Savage, P. G., *Strapdown Analytics*, Strapdown Associates, Inc., Maple Plain, MN (to be published).

[9]Mark, J. G., and Tazartes, D. A., "On Sculling Algorithms," 3rd St. Petersburg International Conf. on Integrated Navigation Systems, St. Petersburg, Russia, May 1996.

[10]Savage, P. G., *Introduction to Strapdown Inertial Navigation Systems*, 8th printing, Strapdown Associates, Inc., Maple Plain, MN, 1997.

[11]Miller, R., "A New Strapdown Attitude Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 6, No. 4, 1983, pp. 287–291.