JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 23, No. 1, January–February 2000

Two New Classes of Strapdown Navigation Algorithms

Yury A. Litmanovich, Vladimir M. Lesyuchevsky, and Valery Z. Gusinsky
Central Scientific and Research Institute "Elektropribor," 197046, St. Petersburg, Russia

Two alternative approaches are presented for deriving strapdown navigation algorithms as opposed to the conventional approach that was used until now and was recently summarized in the paper by Savage (Savage, P. G., "Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 208–221). A key point of the two approaches is the use of additional gyro/accelerometer output signals that are the increments of the angular-rate/specific-force multiple integrals over the iteration interval. This results in two new families of specific-force transformation algorithms that accurately account for the sculling effects and attenuate the pseudosculling errors that can arise because of the high-frequency gyro/accelerometer behavior.

I. Introduction

DURING the past 20–25 years researchers have accumulated great expertise in developing the strapdown inertial navigation system (INS) discrete attitude and navigation algorithms. There are commonly accepted general approaches for deriving strapdown algorithms. Recently, these algorithms were summarized in two fundamental papers by Savage,[1,2] one of the recognized pioneers and experts in this field. These papers properly mark the end of a long period of algorithm development. Moreover, the computational load, which earlier was a key criteria for the algorithm's performance, is no longer as important for the modern-day computers with progressively increasing throughput.

We present two alternative approaches for deriving strapdown navigation algorithms. We realize that there should be a good reason to come up with this paper at a moment when the question seems to be closed. In our opinion, the conventional approach has two limitations, which are discussed.

The traditional procedure for deriving strapdown algorithms is based on the assumption that gyro/accelerometer output signals, which can be used as input parameters for the algorithms, are the increments of the angular-rate/specific-force one-time integrals over the sampling interval. This assumption is not questioned because gyros and accelerometers with incremental outputs are used in practically all INSs examined to date. It should be noted that this statement

Yury A. Litmanovich received his M.S. degree in Electrical Engineering from Leningrad Electrical-Engineering Institute in 1976 and his Ph.D. degree in Technical Sciences from the Central Scientific and Research Institute Elektropribor (Leningrad) in 1987. Since 1976 he has been employed at Elektropribor, where he is currently a Principal Research Scientist. He is a member of the Russian Academy for Navigation and Motion Control.

Vladimir M. Lesyuchevsky received his M.S. degree in Electrical Engineering from the Leningrad Institute of Precise Mechanics and Optics in 1970 and his Ph.D. degree in Technical Sciences from the Central Scientific and Research Institute Elektropribor (Leningrad) in 1978. In 1970 he joined Elektropribor, where he is currently a Department Head. Dr. Lesyuchevsky is a member of the Russian Academy for Navigation and Motion Control.

Valery Z. Gusinsky received his M.S. degree in Electrical Engineering from the Leningrad Institute of Precise Mechanics and Optics in 1963 and his Ph.D. and Sc.D. degrees in Technical Sciences from the Central Scientific and Research Institute Elektropribor (Leningrad) in 1969 and 1989, respectively. In 1963 he joined Elektropribor, where he is now a Division Head, Professor. Dr. Gusinsky is a member of the Russian Academy for Navigation and Motion Control Board and the Editorial Board of the journal *Gyroscopy and Navigation*.

Received 10 August 1998; revision received 11 April 1999; accepted for publication 23 April 1999.

is in principle a limiting one for deriving the algorithms. One should recognize that one-time integration is properly a computational operation. Consequently, we may say that in any INS with incremental sensors there is already a separation of the total computational process into two phases: 1) the primary processing of the angular-rate/specific-force components, which amounts to high-frequency integration over the sampling interval, and 2) the subsequent computations according to the INS algorithms with the computational cycle equal to (or greater than) the sampling interval. Hence, theoretically one may set up a problem of deriving the strapdown algorithms without any restrictions being imposed on the primary processing algorithms. This removal of restrictions on the primary algorithms leads to the possibility of using increments of the angular-rate/specific-force multiple integrals (rather than only one-time integrals) as inputs to the INS algorithms. These additional gyro/accelerometer output signals can be easily generated during signal primary processing. In fact, in the process of the gyro/accelerometer quantum summing, which is performed in sensor electronics, the current state of a counter is updated at a high frequency; consequently the integration of these quantities can be performed at the same or higher frequency.

Another proposition of the conventional approach that we critically review refers to the manner of treating the high-frequency gyro/accelerometer instrument errors, which can result in pseudo-coning/sculling errors.[1] All conventional algorithms are derived to account accurately for vibration-induced sculling effects (with the angular vibrations resulting from linear vibration because of the sensor assembly mount imbalance). As it is impossible to correct the real and the pseudo-coning/sculling effects at the same time if their frequency bands overlap, special measures need to be taken to separate vibration and instrument error spectra, and the algorithm's repetition rate is selected to be higher than the dominant vibration frequencies, but lower than the instrument error frequencies. In doing so, it is supposed that the pseudo-coning/sculling errors will be somehow attenuated. The real attenuation properties of the conventional algorithms are often ignored; they are not expressly structured to account for the instrument errors.

This paper presents two different approaches for deriving strapdown navigation algorithms that would accurately compensate for the vibration-induced sculling effects and, at the same time, attenuate the pseudosculling errors. The first approach is based on one for deriving smoothing strapdown attitude algorithms, which was recently developed by Litmanovich.[3] A key point of this approach is in deriving the angular-rate/specific-force polynomial model's coefficients (which form the basis for any strapdown navigation algorithm) as the least-squares method (LSM) estimates that provide a good smoothing property of the algorithms. For this reason we designated the so-derived new class of strapdown transformation algorithms as smoothing algorithms. These algorithms use the increments of the angular-rate and the specific-force multiple integrals over the iteration interval and the transformation matrix values at the start of the iteration interval. The second approach amounts to one that was developed in the mid-1980s at Elektropribor (St. Petersburg, Russia) for the isolator-free strapdown systems. This approach results in a class of strapdown transformation algorithms that we named acceleration-invariant algorithms for the reason that the general formulas that form the basis for these algorithms are derived without any restrictions imposed on the acceleration frequency band relative to the iteration rate. These algorithms use the increments of the specific-force multiple integrals over the iteration interval and the transformation matrix current and past iteration cycle values. With the acceleration-invariant transformation algorithms, the pseudosculling errors will be attenuated, provided that the gyro instrument errors are effectively smoothed over while performing the attitude computations. For this purpose the recently derived smoothing attitude algorithms[3] can be used with the increments of the angular-rate multiple integrals over the iteration interval being invoked as an additional input to the algorithm.

The algorithm design is performed in view of the INS rapid alignment requirements, which are typical for many applications in which the external navigation data, the master INS output data or zero-velocity information at the stops, are used as the reference data. (This problem is very relevant for the INS/global positioning system (GPS) integrated systems that are of prime interest nowadays.) In doing so, comparatively high-frequency (higher than the Schuler frequency) INS errors are to be rigidly bounded. The differences of the linear position components, generated in the INS and those obtained integrating the external velocity data, are to be used as the measurements for the leveling filters. The idea of using the position measurements instead of the velocity ones was proposed in the early 1970s and now is commonly accepted.

The paper content is organized so that in the navigation algorithm design we move from the end to the beginning, i.e., from the specific-force integration in the navigation frame to the specific-force transformation from the body-fixed to the navigation frame. In Sec. II we analytically examine the performance of both true and conventional integration algorithms under the rapid alignment requirements in conditions of linear vibration. In Sec. III the conventional and the two new procedures for deriving the transformation algorithms are presented and illustrated by examples and a comparison of the derived algorithms is made. The possibility of generating additional accelerometer outputs is justified in Sec. IV. In Sec. V the analytical study of the algorithms under consideration is performed with an emphasis on comparison of the conventional and the proposed algorithm performances under the sculling and the high-frequency accelerometer instrument errors. To confirm the analytically derived estimates, the simulation of the algorithms was performed, and its results are presented and discussed in Sec. VI. Concluding remarks are provided in Sec. VII.

Before coming to the subject we make a few remarks.

1) With the detailed paper by Savage[2] in hand, we concentrated on the principal issues of the total strapdown navigation algorithm only. So, for all the points left beyond the scope of this paper (the initial continuous navigation equations, gravity and Coriolis term compensation, etc.) see Ref. 2. We also do not present nomenclature and coordinate frames for they are basically the same as those given in Ref. 2.

2) The careful reader will find much in this paper that is in common with the most recent Savage paper[2] when the position calculation algorithms are discussed in Sec. II. We did not restrict ourselves to the reference to the cited paper mainly for completeness of the presentation. One reason more is that, in contrast to Savage, we examine the conventional integration algorithm's errors under the rapid alignment requirements that seem to be more realistic for many users. It should be mentioned that the utility of using the true velocity integration algorithm instead of the conventional ones had been justified by the authors in the mid-1980s and widely published in 1994.[4]

3) We consider it our duty to mention the investigations on strapdown algorithm design performed in the former USSR during this period in parallel with the work of American scientists. The most significant theoretical work in this field had been carried out by Branetz and Shmyglevsky (Moscow, Russia) and Panov, Lebedev, and Tkachenko (Kiev, Ukraine) and was recently summarized in monographs. We do not refer to these publications in our paper simply for the reason that these authors develop the same principal approach as Savage does (which we call conventional), but these monographs are written in Russian, which makes them unavailable for the majority of the readers.

II. Specific-Force Integration in the Navigation Frame

According to the conventional discrete specific-force integration algorithm, which is used in many INSs, the velocity components expressed in the navigation frame are calculated by summation of the appropriate velocity increments over the navigation iteration interval,

$$\boldsymbol{V}_n(k+1) = \boldsymbol{V}_n(k) + \delta\boldsymbol{V}_n(k, k+1) \tag{1}$$

where

$$\delta\boldsymbol{V}_n(k, k+1) = \int_{t_k}^{t_{k+1}} \boldsymbol{a}_n(t)\, \mathrm{d}t \tag{2}$$

and the position components, by numerical integration of the so-obtained discrete velocity values, are

$$\boldsymbol{S}_n(k+1) = \boldsymbol{S}_n(k) + \boldsymbol{V}_n(k) \cdot \Delta T \tag{3}$$

where $\boldsymbol{a}_n$, $\boldsymbol{V}_n$, and $\boldsymbol{S}_n$ are the vectors of specific force and its one-time (linear velocity) and two-time (linear position) integrals, respectively, expressed in the navigation frame; $t_k = k \cdot \Delta T$ ($k = 0, 1, 2, \ldots$); ΔT is the navigation iteration interval; and k is the iteration index.

Equation (3) represents a rectangular integration scheme, although a trapezoid or higher-order integration scheme can be used. The important point is that the position components will be determined with iteration-interval-dependent errors, whatever numerical integration method is used. In this case a restriction is imposed to the iteration rate: It must be higher than the frequency of the most high-frequency significant component in the velocity spectrum.

The navigation iteration rate is typically chosen based on the vehicle's dynamics. It turns out that in this case a high rapid alignment accuracy cannot be provided under the linear vibration with the traditional position calculation algorithm of the type of Eq. (3). To illustrate this statement, let us consider a harmonic linear vibration acting along one navigation frame axis of an amplitude (in acceleration) a_V and a circular frequency ω_V. In this case, when the vibration frequency is close to a frequency that is divisible by the navigation iteration rate f ($f = 1/\Delta T$), the velocity and the position outputs will be seen as the harmonics of a low aliased frequency $\tilde{\omega}_V$ and amplitudes $V_0 = a_V/\omega_V$, $S_0 = a_V/\omega_V\tilde{\omega}_V$, respectively. For the worst case from the rapid alignment viewpoint, when the period of the aliased vibration harmonics $\tilde{T}_V$ is greater than the INS alignment time interval, the leveling error β can be evaluated with the polynomial expansion for the INS outputs as $\beta = S_0\tilde{\omega}_V^2$. For $a_V = 2$ g, $\omega_V = 2\pi\ 50\ \mathrm{s}^{-1}$, $f = 50$ Hz, and $\tilde{T}_V = 200$ s it gives $\beta \approx 40$ arcsec, irrespective of whether velocity or position components are used as the measurements for the leveling filter. This rudimentary analysis clearly shows that, with the conventional velocity integration algorithm, errors can arise that may well be intolerable in high-performance systems. The only way to provide high rapid alignment accuracy with this algorithm is to select a navigation iteration rate that is higher than the vibration frequencies. Note that the example presented in Savage's paper (Ref. 2, Sec. VI) when the aliased (sampled) velocity of the vibration is exactly a constant is not disastrous from the INS alignment viewpoint.

Another form of the discrete position calculation algorithm can be trivially derived from the continuous one. It is as follows:

$$\boldsymbol{S}_n(k+1) = \boldsymbol{S}_n(k) + \boldsymbol{V}_n(k) \cdot \Delta T + \delta\boldsymbol{S}_n(k, k+1) \tag{4}$$

where

$$\delta\boldsymbol{S}_n(k, k+1) = \int_{t_k}^{t_{k+1}} \int_{t_k}^{t} \boldsymbol{a}_n(\tau)\, \mathrm{d}\tau\, \mathrm{d}t \tag{5}$$

If there are no errors in the generation of the quantities $\delta\boldsymbol{V}_n$ and $\delta\boldsymbol{S}_n$, algorithm (4) permits the calculation of position components without numerical integration errors in the discrete time points t_k at any magnitude of iteration interval ΔT. Therein lies the fundamental difference between this algorithm and the conventional one [of the type given by Eq. (3)]. In the preceding example the linear position associated with the vibration will be correctly generated without amplitude amplification, so the leveling error appears to be attenuated by a factor of $\omega_V/\tilde{\omega}_V \approx 10{,}000$ in the case in which the position components are used as the measurements for the leveling filter. Therefore, with this true velocity integration algorithm, it becomes possible to provide high rapid alignment accuracy with a low iteration rate selected based on the vehicle's dynamics.

According to the derived algorithm the processes of both one-time and second-time specific-force integration are separated into two stages: a primary continuous (quasi-continuous) integration over the navigation iteration interval and a secondary integration with the iteration interval over the system operation time. A possibility of mechanizing algorithm (4) and its accuracy completely depends on the possibility and the accuracy of generating the quantities $\delta\boldsymbol{S}_n$ which can be called the partial position increments [in contrast to the total position increments that are defined by the last two terms in Eq. (4)].

Going to the strapdown INS to implement the true discrete algorithms (1) and (4), we calculate the following integrals:

$$\delta\boldsymbol{V}_n(k, k+1) = \int_{t_k}^{t_{k+1}} C(t)\boldsymbol{a}(t)\, \mathrm{d}t \tag{6}$$

$$\delta\boldsymbol{S}_n(k, k+1) = \int_{t_k}^{t_{k+1}} \int_{t_k}^{t} C(\tau)\boldsymbol{a}(\tau)\, \mathrm{d}\tau\, \mathrm{d}t \tag{7}$$

where $\boldsymbol{a}$ is the specific-force vector, expressed in the body-fixed frame coordinates, and C is the transformation matrix from the body axes to the navigation axes.

Equations (6) and (7) represent the extended formulation of the specific-force transformation problem. It should be noted that, with regard to vibration, the integral of Eq. (7) is to be calculated only in the case in which a two-speed computation scheme is used.[2] With a single-speed navigation computation, the integral of Eq. (6) is calculated at a high iteration rate (higher than the vibration frequency band), so the conventional velocity integration algorithm of the type of Eq. (3) can be used but at the same high iteration rate.

III. Strapdown Specific-Force Transformation Algorithms

In this section we present the procedures for deriving the conventional and one of the proposed new algorithms—smoothing algorithms—taking a linearly ramping model for the angular-rate/specific-force components as an example. The algorithms for the constant-signal models are presented as well. In doing so we restrict our consideration to the calculation of the velocity increments [Eq. (6)] for the reason that the algorithms for the position increments [Eq. (7)] can be derived similarly. In regard to the acceleration invariant algorithms, we first derive the general solution for the specific-force transformation problem (which forms the basis of the algorithms of this type) and then specify two particular algorithms of this class based on the linear and the square-law models for the transformation matrix (which is equivalent to the constant and the linear models for the angular-rate components, respectively). The general formula for $\delta\boldsymbol{S}_n$ is derived because it is not trivial.

A. Conventional Algorithms

All conventional strapdown specific-force transformation (sculling) algorithms can be divided into two groups: one that uses only the accelerometer outputs as the input signals for the algorithm and another that uses both gyro and accelerometer outputs. Among the first group are a well-known half-sum algorithm with accelerometer outputs that are transformed by means of the average of the attitude matrix computed at the start and the end of the iteration interval and the centered algorithms[5] that use the attitude matrix computed in the center of the iteration interval and the current and the past sample accelerometer outputs. The general procedure for deriving the algorithms of this group is not published most likely owing to its clarity: The algorithms can be derived directly from Eq. (6) by use of the polynomial expansion for the acceleration and the attitude matrix with respect to the start or the center point of the iteration interval. In doing so, the specific-force derivatives are expressed by means of the accelerometer current and past data samples whereas the transformation matrix derivatives are calculated by means of computed current and past iteration cycle matrix C values by use of one numerical method or other. The procedure for deriving transformation algorithms of the second group is likely to be the basic one and was summarized in details in the latest paper by Savage.[2] It should be noted that the procedure for deriving the algorithms of the first group can be transformed to the one for the second group of algorithms by calculation of the matrix C derivatives by use of the analytical relationships between these parameters and the angular-rate derivatives (not by the numerical differentiation). Below we briefly present the conventional procedure as applied to a single-speed computation format. To be specific, we restrict our

consideration to the case in which the sensor samples are taken within the current navigation iteration interval.

According to the general procedure, a solution for the integral of Eq. (6) is sought by use of a first-order approximation for the transformation matrix expressed by the attitude increments over the iteration interval as follows:

$$\delta \boldsymbol{V}_n(k, k+1) = C(k)[\Delta \boldsymbol{V}(k, k+1) + \delta \boldsymbol{V}_s(k, k+1)] \quad (8)$$

where

$$\delta \boldsymbol{V}_s(k, k+1) = \int_{t_k}^{t_{k+1}} \Delta\boldsymbol{\Theta}(t_k, t) \times \boldsymbol{a}(t)\, \mathrm{d}t \quad (9)$$

$$\Delta \boldsymbol{V}(k, k+1) = \int_{t_k}^{t_{k+1}} \boldsymbol{a}(t)\, \mathrm{d}t, \qquad \Delta\boldsymbol{\Theta}(t_k, t) = \int_{t_k}^{t} \omega(t)\, \mathrm{d}t \quad (10)$$

The sculling correction term $\delta \boldsymbol{V}_s$ is expressed by the angular-rate/specific-force polynomial model's coefficients, which can be determined from the gyro/accelerometer outputs $\Delta\boldsymbol{\Theta}$ and $\Delta \boldsymbol{V}$.

By this approach, the number of coefficients to be determined is equal to the number of sensor samples available over the iteration interval ΔT. For example, for the linearly ramping model for the angular-rate ω and specific-force $\boldsymbol{a}$ vectors over the time interval (t_k, t_{k+1}),

$$\omega(t_k + \tau) = A_\omega + B_\omega \tau, \qquad \boldsymbol{a}(t_k + \tau) = A_{\boldsymbol{a}} + B_{\boldsymbol{a}} \tau$$

where

$$A_\omega = \omega(t_k), \qquad A_{\boldsymbol{a}} = \boldsymbol{a}(t_k), \qquad B_\omega = \dot{\omega}(t_k), \qquad B_{\boldsymbol{a}} = \dot{\boldsymbol{a}}(t_k)$$

we have to have two gyro/accelerometer samples over the iteration interval. In this case we can derive the solution for the coefficients A and B by solving identical sets of two linear inhomogeneous algebraic equations written for $\Delta\boldsymbol{\Theta}$ and $\Delta \boldsymbol{V}$. Such a solution for the case of two equally spaced over the interval ΔT samples can be easily derived, as follows:

$$\hat{A}_\omega = (1/\Delta T)(3\Delta\boldsymbol{\Theta}_1 - \Delta\boldsymbol{\Theta}_2), \qquad \hat{A}_{\boldsymbol{a}} = (1/\Delta T)(3\Delta \boldsymbol{V}_1 - \Delta \boldsymbol{V}_2)$$

$$\hat{B}_\omega = (4/\Delta T^2)(\Delta\boldsymbol{\Theta}_2 - \Delta\boldsymbol{\Theta}_1), \qquad \hat{B}_{\boldsymbol{a}} = (4/\Delta T^2)(\Delta \boldsymbol{V}_2 - \Delta \boldsymbol{V}_1) \quad (11)$$

where $\hat{A}_\omega$, $\hat{B}_\omega$, $\hat{A}_{\boldsymbol{a}}$, and $\hat{B}_{\boldsymbol{a}}$ are the appropriate coefficient's estimates and $\Delta\boldsymbol{\Theta}_1$, $\Delta \boldsymbol{V}_1$, $\Delta\boldsymbol{\Theta}_2$, and $\Delta \boldsymbol{V}_2$ are the gyro/accelerometer outputs at time points $t_k + \Delta T/2$ and $t_k + \Delta T$, respectively.

Expressing the functions in Eq. (9) by means of the angular-rate/specific-force model's coefficients, after integration we have

$$\delta \boldsymbol{V}_s = (A_\omega \times A_{\boldsymbol{a}})(\Delta T^2/2) + (A_\omega \times B_{\boldsymbol{a}})(\Delta T^3/3) + (B_\omega \times A_{\boldsymbol{a}})(\Delta T^3/6) + (B_\omega \times B_{\boldsymbol{a}})(\Delta T^4/8) \quad (12)$$

Here and below, the index of the iteration interval is omitted except when it is necessary for proper understanding.

Substituting expressions (11) into Eq. (12) for the sculling correction algorithm finally gives

$$\delta \boldsymbol{V}_s = \tfrac{1}{2}(\Delta\boldsymbol{\Theta} \times \Delta \boldsymbol{V}) + \tfrac{2}{3}[(\Delta\boldsymbol{\Theta}_1 \times \Delta \boldsymbol{V}_2) - (\Delta\boldsymbol{\Theta}_2 \times \Delta \boldsymbol{V}_1)] \quad (13)$$

where $\Delta\boldsymbol{\Theta} = \Delta\boldsymbol{\Theta}_1 + \Delta\boldsymbol{\Theta}_2$ and $\Delta \boldsymbol{V} = \Delta \boldsymbol{V}_1 + \Delta \boldsymbol{V}_2$.

For the constant angular-rate/specific-force models the algorithm is defined by the first term of Eq. (13) and uses the gyro/accelerometer samples over the iteration interval,[2] namely,

$$\delta \boldsymbol{V}_s = \tfrac{1}{2}(\Delta\boldsymbol{\Theta} \times \Delta \boldsymbol{V}) \quad (14)$$

If the square-law or higher polynomial models are taken for the angular rate/specific force over the iteration interval, the algorithms derived according to the general procedure can be optimized for the pure sculling input by a special tuning of the algorithm's coefficients in the same manner as is done with the attitude algorithms for the coning input.[6–8]

B. Smoothing Algorithms

By the basic conventional procedure, the angular-rate/specific-force model's coefficients are calculated from the gyro/accelerometer outputs in a deterministic way. A statistical approach to this problem seems to be more appropriate because of the presence of the additive high-frequency components, which are to be smoothed over. Such an approach, recently developed by Litmanovich, was applied to the strapdown attitude algorithms[3] and can be used for the specific-force transformation problem as well. It has been shown that when the increments of the signal multiple integrals (of multiplicity from 1 to n) over the iteration interval are used, the estimates of the signal n-order polynomial model coefficients can be derived by the LSM. To do this it is sufficient to solve a set of n linear inhomogeneous algebraic equations written for all involved multiple integrals, which are expressed by means of the signal model coefficients to be determined. The so-derived estimates approach the LSM estimates as the number of signal measurements over the integration interval N grows (for $N = 100$ the difference between the true and the approximate LSM estimates did not exceed 1%). Thus this simple method can be used for large N (as it usually is).

Again taking the linearly ramping polynomial model for ω and $\boldsymbol{a}$ over the iteration interval as an example now gives [as opposed to Eqs. (11)]

$$\begin{aligned} \hat{A}_\omega &= (6/\Delta T^2)[\Delta\boldsymbol{\Theta}^2 - \Delta\boldsymbol{\Theta}^1(\Delta T/3)] \\ \hat{A}_{\boldsymbol{a}} &= (6/\Delta T^2)[\Delta \boldsymbol{V}^2 - \Delta \boldsymbol{V}^1(\Delta T/3)] \\ \hat{B}_\omega &= (12/\Delta T^3)[\Delta\boldsymbol{\Theta}^1(\Delta T/2) - \Delta\boldsymbol{\Theta}^2] \\ \hat{B}_{\boldsymbol{a}} &= (12/\Delta T^3)[\Delta \boldsymbol{V}^1(\Delta T/2) - \Delta \boldsymbol{V}^2] \end{aligned} \quad (15)$$

where

$$\Delta\boldsymbol{\Theta}^r = \underbrace{\int_{t_k}^{t_{k+1}} \cdots \int_{t_k}^{t_{r-1}}}_{r} \omega(\tau_r)\, \mathrm{d}\tau_r \cdots \mathrm{d}\tau_1$$

$$\Delta \boldsymbol{V}^r = \underbrace{\int_{t_k}^{t_{k+1}} \cdots \int_{t_k}^{t_{r-1}}}_{r} \boldsymbol{a}(\tau_r)\, \mathrm{d}\tau_r \cdots \mathrm{d}\tau_1 \quad (16)$$

$$\Delta\boldsymbol{\Theta}^1 = \Delta\boldsymbol{\Theta}, \qquad \Delta \boldsymbol{V}^1 = \Delta \boldsymbol{V}$$

The superscripts in $\Delta\boldsymbol{\Theta}$ and $\Delta \boldsymbol{V}$ denote the order of the integral multiplicity.

We can obtain the algorithm for the sculling correction term calculation by substituting expressions (15) into Eq. (12):

$$\delta \boldsymbol{V}_s = \tfrac{1}{2}(\Delta\boldsymbol{\Theta}^1 \times \Delta \boldsymbol{V}^1) - (1/\Delta T)[(\Delta\boldsymbol{\Theta}^1 \times \Delta \boldsymbol{V}^2) - (\Delta\boldsymbol{\Theta}^2 \times \Delta \boldsymbol{V}^1)] \quad (17)$$

Under the constant angular-rate/specific-force conditions, the smoothing algorithm takes the form of conventional algorithm (14).

It is clear that, when the suggested approach is applied with the higher-order angular-rate/specific-force polynomial models, a family of strapdown transformation algorithms can be derived with signal multiple integrals of increasing multiplicity. Reasoning from the essence of the procedure for deriving these algorithms, we can safely assume that the smoothing property of these algorithms under the high-frequency inertial sensor instrument errors will be much better than that of the corresponding conventional algorithms.

C. Acceleration-Invariant Algorithms

All available transformation algorithms of the first group (which do not use the gyro outputs) were derived with the polynomial expansions for both the attitude matrix and the body-fixed specific-force components. It turns out that the general solution for the specific-force transformation problem can be easily derived with no restrictions imposed on the specific-force frequency content. To do this it will suffice to perform the integration in formulas (6) and (7) by using the polynomial expansion for the attitude parameters only. In

fact, let us represent the matrix C as a Taylor series expansion over the iteration interval ΔT:

$$C(t_k + \tau) = C(k) + \dot{C}(k) \cdot \tau + \ddot{C}(k) \cdot (\tau^2/2) + \cdots \qquad (18)$$

where $0 < \tau \leq \Delta T$ and $C(k)$, $\dot{C}(k)$, $\ddot{C}(k)$, etc., are the values of matrix C and its first, its second, etc., derivatives calculated at the time point t_k.

Substituting this relation into formulas (6) and (7) and integrating by parts results in (see Appendix)

$$\begin{aligned}
\delta \boldsymbol{V}_n(k, k+1) &= C(k+1) \int_{t_k}^{t_{k+1}} \boldsymbol{a}(t)\, \mathrm{d}t \\
&\quad - \dot{C}(k+1) \int_{t_k}^{t_{k+1}} \int_{t_k}^{t} \boldsymbol{a}(\tau)\, \mathrm{d}\tau\, \mathrm{d}t + \cdots \\
&= \sum_{r=1}^{R \to \infty} (-1)^{r-1} C^{(r-1)}(k+1) \Delta \boldsymbol{V}^r(k, k+1) \\
\delta \boldsymbol{S}_n(k, k+1) &= C(k+1) \int_{t_k}^{t_{k+1}} \int_{t_k}^{t} \boldsymbol{a}(\tau)\, \mathrm{d}\tau\, \mathrm{d}t \qquad (19) \\
&\quad - 2\dot{C}(k+1) \int_{t_k}^{t_{k+1}} \int_{t_k}^{\tau_1} \int_{t_k}^{\tau_2} \boldsymbol{a}(\tau_2)\, \mathrm{d}\tau_2\, \mathrm{d}\tau_1\, \mathrm{d}t + \cdots \\
&= \sum_{r=1}^{R \to \infty} r(-1)^{r-1} C^{(r-1)}(k+1) \\
&\quad \times \Delta \boldsymbol{V}^{r+1}(k, k+1)
\end{aligned}$$

The derived general formulas specify a class of strapdown transformation algorithms with specific-force multiple integrals as the input parameters, which possess a new fundamental property compared with the conventional ones—they are acceleration invariant, i.e., they hold true for the acceleration input of unrestricted frequency content. It is clear that, to use the algorithms of this class, the iteration rate should be selected to be higher than the angular motion frequency band. Thus, if the high-frequency gyro instrument errors are expected to be noticeable, the smoothing attitude algorithms should be used.

When truncation is set in general formulas (19), the particular quadrature formulas of differing accuracy can be derived. Matrix C derivatives can be calculated in two ways (as was mentioned for the conventional algorithms of the first group): by the numerical differentiation of the computed matrix values and by using the analytical relationships between the required parameters and the angular-rate derivatives. We chose the first one for developing the particular algorithms of this class.

We restrict our consideration to two transformation algorithms when two and three terms are held in the quadrature formulas (i.e., for two values of the parameter R: 2; 3) and the appropriate numerical methods for the calculation of matrix C derivatives are used. (For $R = 1$ we have the simplest transformation algorithm that does not account for the body-frame rotation.) These algorithms are

$R = 2$:

$$\begin{aligned}
\delta \boldsymbol{V}_n(k, k+1) &= C(k+1) \Delta \boldsymbol{V}^1(k, k+1) \\
&\quad - \dot{C}(k+1) \Delta \boldsymbol{V}^2(k, k+1) \qquad (20) \\
\dot{C}(k+1) &= \frac{C(k+1) - C(k)}{\Delta T}
\end{aligned}$$

$R = 3$:

$$\begin{aligned}
\delta \boldsymbol{V}_n(k, k+1) &= C(k+1) \Delta \boldsymbol{V}^1(k, k+1) \\
&\quad - \dot{C}(k+1) \Delta \boldsymbol{V}^2(k, k+1) + \ddot{C}(k+1) \Delta \boldsymbol{V}^3(k, k+1) \\
\dot{C}(k+1) &= \frac{3C(k+1) - 4C(k) + C(k-1)}{2\Delta T} \qquad (21) \\
\ddot{C}(k+1) &= \frac{C(k+1) - 2C(k) + C(k-1)}{\Delta T^2}
\end{aligned}$$

D. Comparison of the Algorithms

As follows from the comparison of derived algorithms (13) and (17), the smoothing algorithms are identical in form to the appropriate conventional ones but use the angular-rate/specific-force multiple integrals over the iteration interval instead of the same number of the conventional gyro/accelerometer samples. The algorithms are similar in structure because the procedure for deriving the smoothing algorithms mostly retraces the procedure for the conventional algorithms. In the conventional algorithms, the higher accuracy is provided by involving a greater number of gyro samples (over the iteration interval and/or from the past intervals). In the new algorithms, the accuracy is provided by use of the multiple integrals of higher multiplicity with the sampling rate equal to the iteration rate.

The acceleration-invariant algorithms have a unique structure, which, however, has computational complexity similar to that of the corresponding conventional or smoothing algorithms. It follows from the comparison of two new types of algorithms—smoothing and acceleration invariant—that the latter use the specific-force multiple integrals of one more multiplicity than the appropriate smoothing algorithms. It should be noted that the proposed algorithms can be transformed to the appropriate conventional ones by use of the relationships between the conventional and the additional sensor outputs for a given signal polynomial model. In particular, it is easy to verify that, for the constant angular-rate/specific-force conditions, the algorithm of Eqs. (20) transforms to both the half-sum algorithm and algorithm (14).

It should be mentioned that when the new algorithms are implemented, the gyro/accelerometer instrument error model parameters should be accounted for in both conventional and additional (multiple integral) outputs. Another point is that the higher the order of the integral multiplicity, the greater the word length needed.

Note that the conventional and the smoothing algorithms use the attitude matrix values at the start of the iteration interval only, whereas the acceleration-invariant algorithms use those at the end of the iteration interval and from the previous iteration cycles. For this reason, with the acceleration-invariant algorithms, the navigation computations should follow the attitude computations, whereas for the conventional and the smoothing algorithms they can be executed in the opposite order.

IV. Specific-Force Multiple Integrals

Let us examine the possibility of the specific-force multiple integral generation, taking a single-axis pulse-rebalance pendulous accelerometer as an example. The accelerometer output signal is generated by a counter that counts the number of pulses over the sampling interval with the pulse value ΔV_0 corresponding to an elementary velocity increment. The increments of the specific-force one-time integral are generated by multiplication of the accumulated number of pulses n^1 by the pulse value: namely

$$\Delta V^1 = \int_{t_k}^{t_k + \Delta T} a(t)\, \mathrm{d}t = n^1 \Delta V_0$$

To generate the increments of the specific-force multiple integrals over the sampling interval, we need to take only the current counter readings with a fixed frequency divisible by the sampling frequency (in our example, the pulse-rebalance frequency) and to perform the summing of these quantities as follows:

$$n^2 = \sum_{i=1}^{N} n_i^1, \qquad n^3 = \sum_{i=1}^{N} n_i^2, \qquad \ldots, \qquad n^{r+1} = \sum_{i=1}^{N} n_i^r$$

where $N = \Delta T / \Delta t$ and Δt is a pulse-rebalance cycle.

Then for the specific-force multiple integrals we have

$$\Delta V^{r+1} = n^{r+1} \Delta t^r \Delta V_0, \qquad r = 0, 1, 2, \ldots$$

To generate the additional accelerometer outputs, either the series adders are to be built into the accelerometer electronics, as shown in Fig. 1, or a digital signal processor should be used. In our opinion in either case this is not difficult.

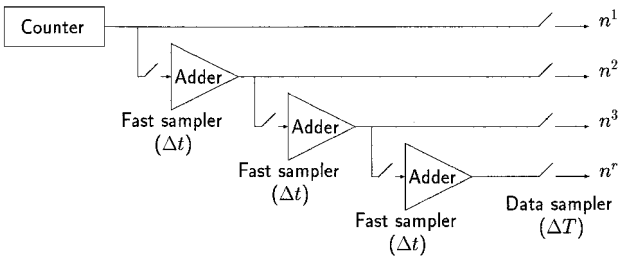

Fig. 1 Diagram for specific-force multiple integral generation.

The possibility of angular-rate multiple integral generation for both the angular rate and the angular-rate integrating sensors was similarly illustrated previously in Ref. 3.

V. Analytical Evaluation of the Algorithm Errors

In this analytical study we restrict our consideration to the acceleration error components in the navigation frame as the major error source in the navigation computations. If true discrete velocity integration algorithm (4) is used, the additional velocity errors that are due to the errors in position increment [Eq. (7)] calculation can be evaluated in a manner similar to that used for the acceleration errors examined below. For each of three types of the algorithms under examination we first derive the algorithm's error expressions for the general case of angular/linear motion (benign environment) and then specify them for the sculling input. In doing so an oscillatory angular motion of an amplitude θ along one navigation-frame axis and an oscillatory linear acceleration of an amplitude a along the orthogonal navigation-frame axis of the common circular frequency ω_0 is considered. The rectified acceleration error along the third navigation axis is evaluated for two cases of sculling—quadrature and in phase.[5] To examine the problem of the smoothing algorithm's optimization under the sculling input, we additionally derive the algorithm for the square-law angular-rate/specific-force models and evaluate its accuracy. We do not present the algorithm itself for it can be easily derived according to the preceding procedure presented in this paper. We present the appropriate results for the conventional algorithms according to a recent paper.[6] For the acceleration-invariant algorithm we additionally examine the algorithm's error under the accelerometer high-frequency instrumental errors for the case in which the gyro high-frequency instrument errors are assumed to be effectively smoothed over in the attitude computations.

A. Conventional Algorithms

To evaluate the algorithm's error we should take into account that the total sculling correction calculation error $\Delta \boldsymbol{W}_s$ is a sum of two components, both of which are due to the truncation of the angular-rate/specific-force polynomial representation (as well as with the rotation vector error in the coning correction calculation[8]). The first term $\Delta \boldsymbol{W}_{s1}$ is caused by the discarding of the higher terms in the expression for the sculling correction term [Eq. (12)], and the second term $\Delta \boldsymbol{W}_{s2}$ is caused by the impact of the higher-order signal derivatives on the estimates of chosen polynomial model coefficients.

Holding the most significant terms for the error components averaged over the iteration interval for the algorithm under consideration [Eq. (13)], we can derive:

$$\Delta \boldsymbol{W}_{s1} = -(\Delta T^4/10)\left[\tfrac{1}{3}(C_\omega \times B_a) + \tfrac{1}{2}(B_\omega \times C_a) + \tfrac{1}{12}(D_\omega \times A_a) + \tfrac{1}{3}(A_\omega \times D_a)\right]$$

$$\Delta \boldsymbol{W}_{s2} = (\Delta T^4/576)[20(C_\omega \times B_a) + 28(B_\omega \times C_a) + 5(D_\omega \times \boldsymbol{a}_a) + 19(A_\omega \times D_a)]$$

where C_ω, D_ω, and C_a, D_a are the second and the third angular-rate/specific-force derivatives in the time point t_k, respectively.

The total algorithm (13) acceleration error is given by

$$\Delta \boldsymbol{W}_s = -(\Delta T^4/720)\left[(C_\omega \times B_a) - (B_\omega \times C_a) + \tfrac{1}{4}(D_\omega \times A_a) - \tfrac{1}{4}(A_\omega \times D_a)\right] \tag{22}$$

Under the case of quadrature sculling, the algorithm's error is virtually zero [thanks to the first term of algorithm (13) (Ref. 5)]. Specified for in-phase sculling, expression (22) takes the form

$$\Delta \boldsymbol{W}_s = \frac{\theta a}{2}\frac{3(\omega_0 \Delta T)^4}{1440} \tag{23}$$

For algorithm (14) the following can be derived:

$$\Delta \boldsymbol{W}_s = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^2}{6} \tag{24}$$

The optimal sculling correction algorithm that uses three gyro/accelerometer samples equally spaced over the iteration interval can be derived by optimization of the general algorithm based on the square-law angular-rate/specific-force models. Based on the formulations in Ref. 6, the acceleration error of such an algorithm is as follows:

For sculling,

$$\Delta \boldsymbol{W}_s = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^6}{102{,}060}$$

For a benign environment,

$$\Delta \boldsymbol{W}_s = (\Delta T^3/120)[(A_\omega \times C_a) - (C_\omega \times A_a)]$$

Therefore the optimal sculling algorithm demonstrates the degradation of its performance in a benign environment, even compared with algorithm (13).

B. Smoothing Algorithms

The total acceleration error for the algorithm under consideration [Eq. (17)] can be derived in a manner similar to that for the conventional algorithm discussed in the preceding subsection. It is defined by the following most-significant terms:

$$\Delta \boldsymbol{W}_s = -(\Delta T^4/720)[(C_\omega \times B_a) - (B_\omega \times C_a)] \tag{25}$$

Under the case of quadrature sculling the algorithm's error is virtually zero as well. Specified for the in-phase sculling, expression (25) gives

$$\Delta \boldsymbol{W}_s = \frac{\theta a}{2}\frac{4(\omega_0 \Delta T)^4}{1440} \tag{26}$$

Based on the general procedure, the algorithm was derived for the square-law angular-rate/specific-force models that use the increments of three angular-rate/specific-force multiple integrals over the iteration interval. For the acceleration error of such an algorithm, the following were obtained:

1) For a benign environment,

$$\Delta \boldsymbol{W}_s = -(\Delta T^6/700)[(D_\omega \times C_a) - (C_\omega \times D_a)] \tag{27}$$

2) For sculling,

$$\Delta \boldsymbol{W}_s = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^6}{100{,}800} \tag{28}$$

In the derivation of expression (27) it is found that two sculling algorithm error components (one is caused by discarding the higher terms in the expression for the sculling correction term and the other is due to the impact of the higher-order signal derivatives on the estimates of chosen polynomial model coefficients) of the fourth order in ΔT compensate each other, as with the smoothing attitude algorithms.[3] For this reason the accuracy of the appropriate conventional algorithm optimized for the pure sculling input is provided in a nonsculling environment as well, as is clearly seen from the comparison of expressions (27) and (28). It is reasonable to suppose that this unique property is peculiar to all the algorithms of this family.

C. Acceleration-Invariant Algorithms

According to the procedure for deriving the algorithms of this class, the algorithm's accuracy is dictated by the order of the quadrature formula [the number of terms retained in matrix C expansion (18)] and the errors in its realization (the errors in calculation of matrix C derivatives). We denote the average of these acceleration error components over the iteration interval as $\Delta \boldsymbol{W}_1$ and $\Delta \boldsymbol{W}_2$, respectively. We can derive the expressions for the algorithm errors in a benign environment by expressing $\Delta \boldsymbol{V}$ by means of the polynomial expansion for the acceleration components and by using the expressions for the errors in calculating the matrix C derivatives. For the algorithm of Eqs. (21) it yields

$$\begin{gathered}\delta\dot{C}(k+1) = -C^{(3)}(k)(2\Delta T^2/3) - C^{(4)}(k)(\Delta T^3/6)\\ \delta\ddot{C}(k+1) = -C^{(3)}(k)\Delta T - C^{(4)}(k)(5\Delta T^2/12)\end{gathered} \tag{29}$$

$$\Delta \boldsymbol{W}_1 = C^{(3)} A_a(\Delta T^3/24) + C^{(3)} B_a(\Delta T^4/120) + C^{(4)} A_a(\Delta T^4/30)$$

$$\Delta \boldsymbol{W}_2 = C^{(3)} B_a(\Delta T^4/72) - C^{(4)} A_a(\Delta T^4/36)$$

In the case of quadrature sculling, the algorithm's error is determined by the first term of $\Delta \boldsymbol{W}_1$:

$$\Delta \boldsymbol{W} = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^3}{24} \tag{30}$$

For in-phase sculling according to Eqs. (29),

$$\Delta \boldsymbol{W} = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^4}{60} \tag{31}$$

For the algorithm of Eqs. (20) it can be derived that, in the case of quadrature sculling, the algorithm's error is virtually zero. For in-phase sculling it gives

$$\Delta \boldsymbol{W} = \frac{\theta a}{2}\frac{(\omega_0 \Delta T)^2}{12} \tag{32}$$

To derive the expressions for the algorithm error with respect to the high-frequency acceleration instrument error component $\boldsymbol{a}_h$ we represent the quantities $\Delta \boldsymbol{V}^R(t_k, t)$ as

$$\Delta \boldsymbol{V}^R(t_k, t) = \boldsymbol{V}^R(t) - \sum_{r=1}^{R} \boldsymbol{V}^R(t_k)\frac{(t-t_k)^{R-r}}{(R-r)!}$$

where

$$\boldsymbol{V}^R(t) = \boldsymbol{V}^R(0) + \underbrace{\int_0^t \cdots \int_0^{t_R}}_{R} \boldsymbol{a}_h(t_R)\, \mathrm{d}t_R \cdots \mathrm{d}t_1$$

The quantities $\boldsymbol{V}^R(t_k)$ can be considered as the values of the functions $\tilde{\boldsymbol{V}}^R(t)$, which are equal to the functions $\boldsymbol{V}^R(t)$ sampled at time interval ΔT.

As with the analysis of the navigation algorithm errors under vibration, here we restrict our consideration to the case in which the acceleration error frequency f_e is close to a frequency divisible by the iteration rate $f = 1/\Delta T$. In this case the functions $\tilde{\boldsymbol{V}}^R(t)$ can be approximately considered as the harmonics of low frequency $\tilde{f}_e$, which is equal to the aliased acceleration noise frequency:

$$\tilde{f}_e = f_e - nf$$

where n is an integer. Again, averaging the estimates over the time intervals (t_k, t_{k+1}), for the algorithm of Eqs. (21) we approximately have

$$\begin{gathered}\Delta \boldsymbol{W}_{1h} = -C^{(3)}[\tilde{\boldsymbol{V}}^3 + \tilde{\boldsymbol{V}}^2(\Delta T/2) + \tilde{\boldsymbol{V}}^1(\Delta T^2/6)]\\ \Delta \boldsymbol{W}_{2h} = C^{(3)}[\tilde{\boldsymbol{V}}^2 - \tilde{\boldsymbol{V}}^1(\Delta T/6)]\Delta T\end{gathered} \tag{33}$$

Our prime interest is in conditions under which the low-frequency errors (in the limiting case, the constants) arise. It is seen from the analysis of relationships (33) that these errors occur when the aliased acceleration error frequency $\tilde{f}_e$ is close to the angular motion frequency. Let there be an oscillatory angular motion of an amplitude θ and circular frequency ω_0 and an oscillatory acceleration instrument error of an amplitude in acceleration a_e and a circular frequency $\omega_e = 2\pi f_e$, which satisfies the following relationship:

$$2\pi \tilde{f}_e = \omega_0 \tag{34}$$

To evaluate the performance of the acceleration-invariant algorithms under these conditions, the approximate expressions for the rectified acceleration errors for three algorithms of this class (for $R = 1, 2, 3$) for the worst case of phase relationship are summarized in Table 1. The error expressions for the algorithm of Eqs. (21) were derived according to formulas (33); those for the algorithm of Eqs. (20) and the simplest transformation algorithm can be derived in the same way and are presented in Ref. 9.

Therefore, the greater the order of the acceleration-invariant algorithm, the lower the pseudosculling errors.

Table 1 Acceleration-invariant algorithm errors

Algorithm's order, R	Algorithm's error
1	$(\theta a_e/2)(\omega_0/\omega_e)$
2	$(\theta a_e/2)(\omega_0/\omega_e)^2$
3	$(\theta a_e/2)(\omega_0/\omega_e)^3$

Table 2 Algorithm sculling response (constant ω/a models)

Algorithm type	Conventional	Smoothing	Acceleration invariant
Algorithm equation	(14)	(14)	(20)
Algorithm error	$\frac{1}{6}(\omega_0 \Delta T)^2$	$\frac{1}{6}(\omega_0 \Delta T)^2$	$\frac{1}{12}(\omega_0 \Delta T)^2$

Table 3 Algorithm sculling response (linearly ramping ω/a models)

Algorithm type	Conventional	Smoothing	Acceleration invariant
Algorithm equation	(13)	(17)	(21)
Algorithm error	$3/1440(\omega_0 \Delta T)^4$	$4/1440(\omega_0 \Delta T)^4$	$24/1440(\omega_0 \Delta T)^4$

D. Comparison of the Algorithm Performances

As a result of the analytical study, we summarized the sculling responses of the conventional and the proposed algorithms for the case of in-phase sculling based on derived error expressions (23), (24), (26), (31), and (32). In Tables 2 and 3 the relative sculling error (which is the ratio of the acceleration error to the sculling acceleration $\theta a/2$) is presented for two algorithm groups based on the constant and linearly ramping angular-rate/specific-force models, respectively.

As Tables 2 and 3 show, a conclusion can be made that the new derived algorithms are of the same order of accuracy as the corresponding conventional ones for in-phase sculling. When examined in detail, the acceleration-invariant algorithms are slightly better than the others in the first group and slightly worse than those in the second one. The conventional and the smoothing algorithms are of nearly the same accuracy. Note that the error of the acceleration-invariant algorithm in the case of quadrature sculling is not necessarily zero for this class as a whole [this is the case for the algorithm of the second group; see Eq. (30)]. To complete the comparison we remind the reader that in a nonsculling benign environment, the smoothing algorithms demonstrate a noticeable improvement over the corresponding conventional ones.

VI. Simulation

In the first stage of the simulation the relative sculling error as a function of the relative frequency (which is the ratio of the sculling frequency $\omega_0/2\pi$ to the iteration rate $f = 1/\Delta T$) was determined for in-phase sculling. In the simulation, we generated the gyro/accelerometer outputs following the exact analytical expressions; therefore numerical integration errors were avoided. The

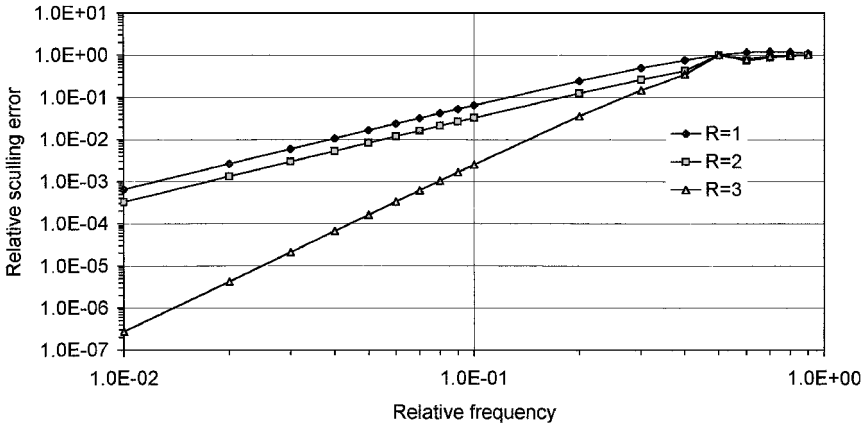

Fig. 2 Sculling response of acceleration-invariant algorithms (log–log plot).

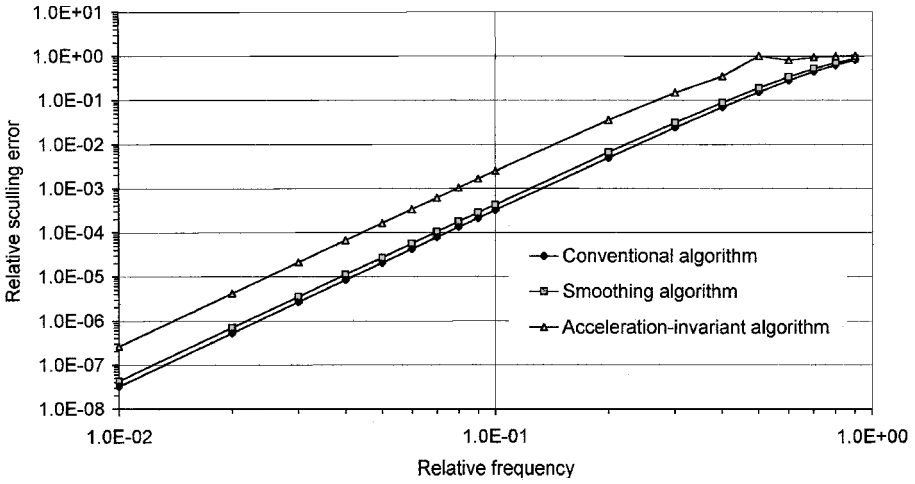

Fig. 3 Algorithm sculling response (log–log plot).

algorithm acceleration error was extracted from the INS position output by the square-law LSM estimator. The estimation interval was taken to be large enough to smooth over the oscillatory error components effectively and to provide high numerical calculation accuracy. The simulation results for three algorithms of the acceleration-invarianttype and those for the corresponding conventional and new algorithms of two types [Eqs. (13), (17), and (21)] are shown in Figs. 2 and 3, respectively. In the simulation, the sculling frequency varied in a wide range (from 10 to 1000 Hz), with a fixed computation rate of 1000 Hz. These results are in close agreement with those derived analytically.

To examine the pseudosculling errors of the proposed and the conventional algorithms, the simulation was performed with high-frequency gyro and accelerometer noise components. In the simulation, the orthogonal angular-rate and specific-force noise components were specified as the harmonics of the same frequency f_e and amplitude ω_e, a_e, respectively, but with a 90-deg phase shift. In this case an acceleration error arises, which is similar to the sculling error. If the iteration rate is higher than the harmonic frequency, the pseudosculling error can be evaluated as

$$\Delta \boldsymbol{W}_{\text{ps}} = \frac{\omega_e a_e}{2(2\pi f_e)} \tag{35}$$

As the iteration rate becomes lower than the noise frequency, the sensor noise components are smoothed over by the algorithm, so that the error diminishes as the noise frequency increases. The relative pseudosculling error [which is the ratio of the algorithm error to its maximum value, Eq. (35)] as a function of the relative frequency (which is the ratio of the noise frequency to the iteration rate) is compared for three algorithms in Fig. 4. In the simulation, the noise frequency varied from 1300 to 50,300 Hz. In doing so, the aliased noise frequency $\tilde{f}_e$ was taken to be equal to 300 Hz. The computational rate was taken to be 1000 Hz, and the instrument noise parameters were specified as follows: $\omega_e = 1$ deg/h, $a_e = 0.2$ g. Figure 4 shows a definite improvement of the smoothing algorithm over the conventional one. The acceleration-invariant algorithm demonstrates a significantly greater improvement that grows with the relative frequency.

In the next stage, the comparative simulation of the algorithms was performed under the angular motion in combination with the high-frequency accelerometer noise. The frequency of the noise harmonics was taken in such a way that the aliased noise frequency was equal to the angular motion frequency. An additional acceleration error arises that in this case is normalized by the maximum pseudosculling error value,

$$\Delta \boldsymbol{W}_{\text{ps}} = \theta a_e / 2 \tag{36}$$

In the simulation, the noise frequency varied from 1100 to 100,100 Hz. The computational rate was taken to be 500 Hz, and the angular motion parameters were specified as follows: $\theta = 0.001$ rad, $\omega_0 = 2\pi \cdot 100\,\text{s}^{-1}$. The plots of the algorithm relative error as a function of the relative vibration frequency are shown in Figs. 5 and 6 for three acceleration-invariant algorithms and for the corresponding conventional and new algorithms of two types, respectively. In this case as well, the simulation results are in close agreement with the analytical estimates derived for the acceleration-invariant algorithm errors with the accelerometer noise input. It is clearly seen from a comparison of the results presented in Figs. 5 and 6 that, under these conditions, the conventional algorithm is of equal accuracy as the simplest first-order algorithm and ranks far below both

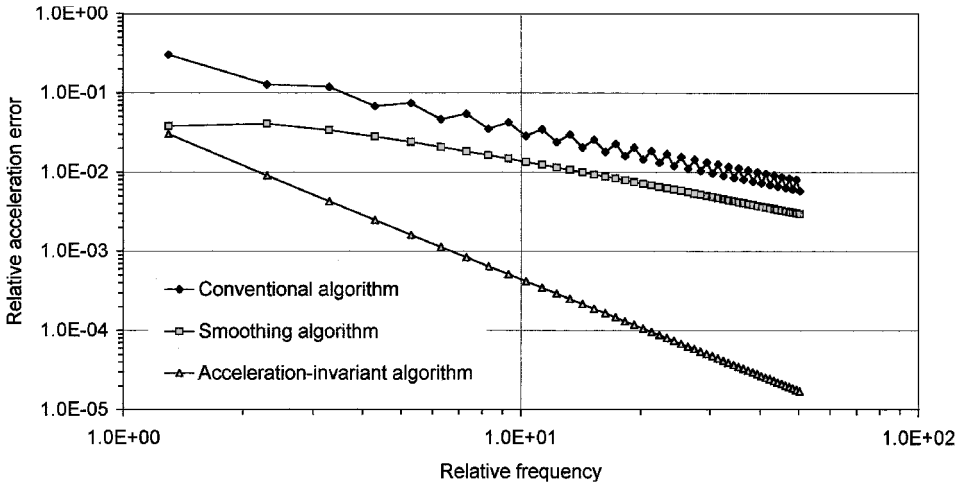

Fig. 4 Algorithm errors with the gyro/accelerometer noise (log–log plot).

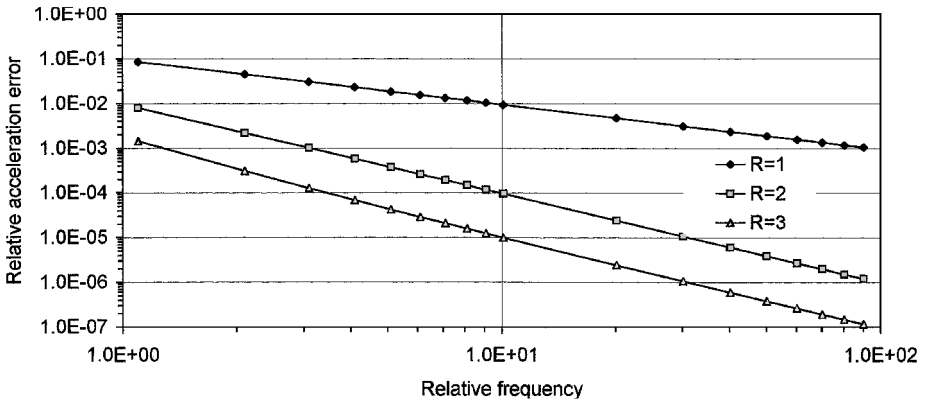

Fig. 5 Acceleration-invariant algorithm errors with angular motion combined with accelerometer noise (log–log plot).

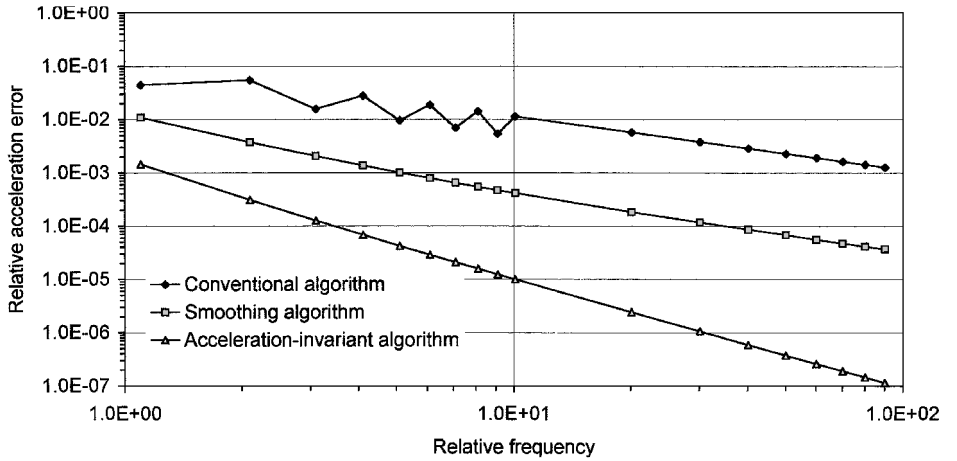

Fig. 6 Algorithm errors with angular motion combined with accelerometer noise (log–log plot).

new algorithms. Among these, the acceleration-invariantalgorithm demonstrates a noticeable improvement over the smoothing transformation algorithm. The accuracy improvement increases with relative vibration frequency.

VII. Conclusion

The approaches for deriving two new classes of strapdown navigation algorithms were presented and examined. One is based on deriving the angular-rate/specific-force polynomial model's coefficients (which forms the basis for any algorithm) as the LSM estimates; for this reason we designated the so-derived algorithms as smoothing. The other has its basis in new general formulas derived without restrictions imposed on the acceleration frequency band relative to the iteration rate; therefore, we called these algorithms acceleration invariant. The goal of this design was to derive algorithms that would accurately account for the vibration-induced sculling effects and attenuate the pseudosculling errors that can arise from the high-frequency gyro/accelerometer instrument noise. A key feature of both approaches is the use of the additional gyro/accelerometer output signals, which are the increments

of the angular-rate/specific-force multiple integrals over the iteration interval.

The accuracy equivalence of the corresponding conventional and proposed algorithms under sculling motion was demonstrated. A unique property of the error autocompensation typical for the preceding derived smoothing attitude algorithms was proven to be valid for the smoothing transformation algorithms as well. This results in an accuracy improvement of the derived algorithms over the corresponding optimal ones of three orders in a nonsculling benign environment. Thus, with the smoothing algorithms, the accuracy of the corresponding conventional algorithms optimized for the pure sculling input is also provided under the nonsculling conditions.

A notable improvement of the new algorithms over the conventional ones was manifested under conditions of high-frequency instrument errors. Among the proposed algorithms, the acceleration-invariant algorithms were shown to be even more effective in attenuating the pseudosculling errors than the corresponding smoothing algorithms, with an accuracy improvement increasing with the relative instrument error frequency.

The smoothing algorithms are identical in form to the corresponding conventional ones, but use several sequential angular-rate/specific-force multiple integrals instead of the same number of conventional gyro/accelerometer samples. The acceleration-invariant algorithms have a unique structure, which has a computational complexity similar to that of the corresponding conventional or smoothing algorithms. When comparing corresponding smoothing and acceleration-invariant algorithms, we note that the latter use the specific-force multiple integrals of one more multiplicity than the former. The price we must pay to implement the new algorithms is the generation of additional sensor outputs, to correct them for the instrument error model parameters, and to provide the increased word length they require. All these problems can be easily solved if the performance improvement the proposed algorithms provide is valuable for a particular strapdown design.

Appendix: Derivation of the General Quadrature Formulas

Substituting Eq. (18) into Eq. (6) for the current values of the velocity increments over the interval $(t_k, t_k + \Delta T)$ yields

$$\delta \boldsymbol{V}_n(t_k, t_k + \tau) = C(k) \int_0^\tau \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau + \dot{C}(k) \int_0^\tau \tau \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau + \ddot{C}_k \int_0^\tau \frac{\tau^2}{2} \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau + \cdots \tag{A1}$$

where $0 \le \tau \le \Delta T$.

With the introduced designations of Eqs. (16), expressions for the integrals in Eq. (A1) can be transformed by integration by parts, as follows:

$$\begin{aligned} &\int_0^\tau \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau = \Delta \boldsymbol{V}^1(t_k, t_k + \tau) \\ &\int_0^\tau \tau \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau = \tau \Delta \boldsymbol{V}^1(t_k, t_k + \tau) - \Delta \boldsymbol{V}^2(t_k, t_k + \tau) \\ &\int_0^\tau \frac{\tau^2}{2} \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau = \frac{\tau^2}{2} \Delta \boldsymbol{V}^1(t_k, t_k + \tau) \\ &\quad - \tau \Delta \boldsymbol{V}^2(t_k, t_k + \tau) + \Delta \boldsymbol{V}^3(t_k, t_k + \tau) \\ &\vdots \end{aligned} \tag{A2}$$

Substituting Eqs. (A2) into Eq. (A1), we obtain

$$\begin{aligned} \delta \boldsymbol{V}_n(t_k, t_k + \tau) &= [C(k) + \dot{C}(k)\tau + \cdots] \Delta \boldsymbol{V}^1(t_k, t_k + \tau) \\ &\quad - [\dot{C}(k) + \ddot{C}(k)\tau + \cdots] \Delta \boldsymbol{V}^2(t_k, t_k + \tau) \\ &\quad + \left[\ddot{C}(k) + C^{(3)}(k)\tau + \cdots\right] \Delta \boldsymbol{V}^3(t_k, t_k + \tau) - \cdots \end{aligned} \tag{A3}$$

In view of Eq. (6) and the polynomial expansion for matrix C and its derivatives, Eq. (A3) can be rewritten as

$$\int_0^\tau C(t_k + \tau) \boldsymbol{a}(t_k + \tau)\, \mathrm{d}\tau = C(t_k + \tau) \Delta \boldsymbol{V}^1(t_k, t_k + \tau) - \dot{C}(t_k + \tau) \Delta \boldsymbol{V}^2(t_k, t_k + \tau) + \ddot{C}(t_k + \tau) \Delta \boldsymbol{V}^3(t_k, t_k + \tau) - \cdots \tag{A4}$$

For $\tau = \Delta T$, Eq. (A4) transforms into the first equation of Eqs. (19).

To derive the second equation of Eqs. (19), let us represent the quantity under examination in the following form:

$$\delta \boldsymbol{S}_n(k, k + 1) = \int_0^{\Delta T} \delta \boldsymbol{V}_n(t_k, t_k + \tau)\, \mathrm{d}\tau \tag{A5}$$

Substituting Eq. (A4) into Eq. (A5) yields

$$\delta \boldsymbol{S}_n(k, k + 1) = \int_0^{\Delta T} C(t_k + \tau) \Delta \boldsymbol{V}^1(t_k, t_k + \tau)\, \mathrm{d}\tau - \int_0^{\Delta T} \dot{C}(t_k + \tau) \Delta \boldsymbol{V}^2(t_k, t_k + \tau)\, \mathrm{d}\tau + \int_0^{\Delta T} \ddot{C}(t_k + \tau) \Delta \boldsymbol{V}^3(t_k, t_k + \tau)\, \mathrm{d}\tau - \cdots \tag{A6}$$

As the structures of the polynomial expansion for matrix C and its derivatives are identical, each integral in Eq. (A6) can be represented in a form similar to that of Eq. (A4), namely,

$$\begin{aligned} &\int_0^\tau C \Delta \boldsymbol{V}^1\, \mathrm{d}\tau = C \Delta \boldsymbol{V}^2 - \dot{C} \Delta \boldsymbol{V}^3 + \ddot{C} \Delta \boldsymbol{V}^4 - \cdots \\ &\int_0^\tau \dot{C} \Delta \boldsymbol{V}^2\, \mathrm{d}\tau = \dot{C} \Delta \boldsymbol{V}^3 - \ddot{C} \Delta \boldsymbol{V}^4 + C^{(3)} \Delta \boldsymbol{V}^5 - \cdots \\ &\int_0^\tau \ddot{C} \Delta \boldsymbol{V}^3\, \mathrm{d}\tau = \ddot{C} \Delta \boldsymbol{V}^4 - C^{(3)} \Delta \boldsymbol{V}^5 + C^{(4)} \Delta \boldsymbol{V}^6 - \cdots \\ &\vdots \end{aligned} \tag{A7}$$

In view of expressions Eq. (A7), Eq. (A6) can be transformed to the form of the second equation of Eqs. (19).

Acknowledgments

This study was partially sponsored by the Russian Foundation for Basic Research under Grant 97-01-01134. The part of this work concerned with the specific-force integration and the acceleration-invariant transformation algorithms was presented at the first (25–26 May 1994) and the fifth (25–27 May 1998) St. Petersburg International Conferences on Integrated Navigation Systems, St. Petersburg, Russia.

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., "Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 208–221.

[3] Litmanovich, Yu. A., "Use of Angular Rate Multiple Integrals as Input Signals for Strapdown Attitude Algorithms," *Proceedings of Symposium Gyro Technology 1997*, Institut A fur Mechanik, Universität Stuttgart, Stuttgart, Germany, 1997, pp. 20.0–20.9.

[4] Lesyuchevsky, V. M., and Litmanovich, Yu. A., "Specific Force Transformation and Navigation Discrete Algorithms of Inertial Navigation Systems,"

Proceedings of the 1st St. Petersburg International Conference on Gyroscopic Technology, Central Scientific and Research Institute "Elektropribor," St. Petersburg, Russia, Post-conference edition, 1994, pp. 207–217.

[5]Mark, J. G., and Tazartes, D. A., "On Sculling Algorithms," *Proceedings of the 3rd St. Petersburg International Conference on Integrated Navigation Systems*, Central Scientific and Research Institute "Elektropribor," St. Petersburg, Russia, 1996, Pt. 2, pp. 22–26.

[6]Ignagni, M. B., "Duality of Optimal Strapdown Sculling and Coning Compensation Algorithms," *Navigation: Journal of the Institute of Navigation*, Vol. 45, No. 2, Summer 1998, pp. 85–95.

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

[8]Gusinsky, V. Z., Lesyuchevsky, V. M., Litmanovich, Yu. A., Musoff, H., and Schmidt, G. T., "New Procedure for Deriving Optimized Strapdown Attitude Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 4, 1997, pp. 673–680.

[9]Litmanovich, Yu. A., Lesyuchevsky, V. M., and Gusinsky, V. Z., "Study of Strapdown Transformation Algorithms with Specific Force Multiple Integrals as Input Signals," *Proceedings of the 5th St. Petersburg International Conference on Integrated Navigation Systems*, Central Scientific and Research Institute "Elektropribor," St. Petersburg, Russia, 1998, pp. 45–55.