

A precise quaternion-based navigation algorithm for simulating signals of accelerometers and gyroscopes with low sample frequencies

Accelerometers and gyroscopes based on MEMS technology are promising for tracing motion in medicine, sport activities, human-machine interaction, robotics and many other areas due to the fact that they are self-containing and have a range of other advantages. Three orthogonally placed accelerometers and gyroscopes are combined into a single module fitted with a controller for processing the signals from inertial sensors. However, the same module may be suitable for one application and inapplicable for another, since the accuracy of tracking a motion trajectory depends not only on the error characteristics of the inertial sensors but also on the trajectory itself. Simulation may help decide whether an inertial measurement unit is a reasonable choice for a specific application or not. The idea is to allow the user to preset a desirable motion trajectory and error characteristics of the inertial sensors specified by their manufacturer. Then software simulates signals of real accelerometers and gyroscopes and computes a set of potential trajectories upon these signals. Upon the discrepancies between the prescribed and synthesized trajectories one can judge on applicability of the inertial sensors with the preset error characteristics for a specific task, without implementing a real device. The software should be based on well-known navigation equations, expressed via direction cosine matrices or quaternions. However, the equations are only valid for infinitesimal rotation angles. Their usage leads to cumulating errors in computation of some trajectories due to the fact that low-cost accelerometers and gyroscopes available on the market offer limited sample frequencies. The work reveals the problem related to usage of the above-mentioned equations, both analytically and by numerical experiments. Examples of trajectories irreproducible at low frequencies are shown. The work analyzes the reasons why some trajectories are irreproducible and shows that the reasons can scarcely be eliminated in case of rotation matrices. We have proposed amended equations universal for any trajectory and any sample frequency.

Keywords: navigation equation; MEMS accelerometer; MEMS gyroscope; Poisson equation; quaternion; signal synthesis; trajectory.

Introduction. Inertial measurement units are widely used in a broad range of applications including human-machine interaction [1], sport [2], user authentication [3], robotics and telemedicine [4]. Inertial measurement units produce inevitably faulty readings [5]. In order to be able to use such erroneous readings different filters have been designed. All filters have their drawbacks, thus data fusion and filtering area is still being intensively elaborated [6, 7]. The same inertial measurement unit may be sufficient for tracking one kind of motion and completely inapplicable for another. Particularly, fast movements are irreproducible if the inertial sensor supports low sample frequencies [8, 9]. In order to decide on applicability of a specific inertial measurement unit for tracking a specific motion trajectory one can try and see what happens, i.e., take a real hardware module, move it, calculate the motion trajectory and evaluate the results by comparison of the real motion and computed one. Such comparison might be difficult without additional reference motion capture systems alike Vicon. However, this straightforward method is definitely cumbersome, time-consuming and expensive. Simulation would help to solve the problem. A software tool which allows prescribing a motion scenario and modelling measurement errors will enable the researchers and engineers to predict whether an inertial navigation system with the prescribed error characteristics can be used in a specific motion capture application or not. In this way, simulation of output signals from accelerometers and gyroscopes is of crucial importance, since it enables the researchers and engineers to assess the performance of a navigation system prior to its actual implementation and thus save costs and efforts. Moreover, filters are better to be tested on simulated signals and then subjected to more complicated tests with real equipment only after their efficacy has been proven on simulated data.

Related work. The first step that the user of the mentioned software tool is supposed to take is to preset a motion trajectory (Bézier curves or B-splines can be used with this purpose), velocities and rotation angles. Then the user predefines a set of measurement error characteristics. The latter may be found out in the manual of an inertial measurement unit. The tool simulates possible trajectories that one could obtain if the inertial measurement unit were used for tracking the trajectory. Taking into account the fact that all inertial sensors suffer from stochastic errors, there can be multiple possible outcomes. Finally, the software evaluates how much the outcomes deviate from the originally prescribed trajectory. Depending on the level of uncertainty demonstrated by the simulation results, the user can judge about applicability of the inertial measurement unit. Logically, if there are no measurement errors, the prescribed trajectory should be reproduced exactly. If this

requirement is not met, it becomes difficult to single out the influence of each gyroscope or accelerometer error characteristic on the discrepancies between the preset and calculated trajectories.

A triaxial accelerometer measures the acceleration a body experiences due to both non-gravitational and gravitational forces:

$$\bar{a}_i = \bar{a} - \bar{g}, \quad (1)$$

where \bar{a} is the so-called true acceleration, i.e., acceleration with respect to the inertial reference system; \bar{g} is the gravity vector expressed in the inertial reference frame. We assume that this vector is $[0 \ 0 \ g]$, where g is approximately $9,81 \text{ m/s}^2$. The vector is written with the minus sign because an accelerometer reacts to the force that prevents it from free falling, i.e. the force that balances gravity. Particularly, if the accelerometer lies still on a table, it will measure the components of the vector $-[0 \ 0 \ g]$ expressed in its body frame. If the accelerometer moves in a sharp, uneven manner, it will sense both components of acceleration (due to motion and gravity), measured with respect to the inertial frame but expressed in its body frame. A triaxial accelerometer is comprised of three one-axis accelerometers whose sensitivity axes are mutually orthogonal. The subscript «i» means «inertial», the subscript «b» denotes «body». The measurements are expressed in the device body frame.

If the true acceleration and initial velocity V_0 are known, then the velocity of the body can be computed as:

$$\bar{v}(t) = v_0 + \int_{t_0}^t \bar{a} dt. \quad (2)$$

If the initial position is known, then the whole trajectory can be determined by integration of the velocity:

$$[x(t) \ y(t) \ z(t)] = [x_0 \ y_0 \ z_0] + \int_{t_0}^t \bar{v} dt. \quad (3)$$

Obviously, formulas (2) and (3) can only be applied if the acceleration is expressed in the same frame. The most convenient way to keep measurements from the accelerometer aligned is to transform the acceleration from the body frame to the inertial frame.

Presentation of the same (acceleration) vector in different frames can be performed in several ways. Using direction cosine matrices, one applies the following formula:

$$\bar{a}_i = C_b^I \bar{a}_b, \quad (4)$$

where \bar{a}_i is the total acceleration vector in the inertial frame, \bar{a}_b is the same vector in the accelerometer body frame, and C_b^I is the direction cosine matrix that relates the body frame to the inertial frame:

$$C_b^I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix} \begin{pmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (5)$$

The true acceleration then would be:

$$\bar{a} = C_b^I \bar{a}_b + \bar{g}, \quad (6)$$

Matrix C_b^I describes 3 rotations, performed one after another around axes z , y and x . Angles of rotations, ψ about axis z , θ about axis y and ϕ about axis x are called yaw (heading), pitch and roll (bank), correspondingly and collectively referred to as Euler or Tait-Bryan angles. When using the direction cosine matrices, it is worth bearing in mind that the order of rotations really matters. For instance, rotations around z , y and x in general do not yield the same result as rotations around x , y and z . Moreover, when the pitch is around $\pm\pi/2$, gimbal lock is observed, i.e., one degree of freedom will be lost.

Matrix C_b^I is orthogonal and the following formula is correct:

$$C_b^I = (C_b^I)^{-1} = (C_b^I)^T, \quad (7)$$

An alternative form of rotation representation is using quaternions. Quaternions are less intuitive than direction cosine matrices and much less intuitive than Euler angles. However, they use only four parameters instead of nine, in contrast to direction cosine matrices. According to the Euler theorem, three consecutive rotations around three different axes can be replaced by a single rotation around a single axis. The latter is defined by the eigenvector of the corresponding direction cosine matrix. The single rotation by some angle around the single axis is described by a quaternion:

$$q = [q_1 \ q_2 \ q_3 \ q_4] = q_1 + q_2 \bar{i} + q_3 \bar{j} + q_4 \bar{k} = \left[\cos \frac{|\Phi|}{2} \quad \sin \frac{|\Phi|}{2} \frac{\Phi_x}{|\Phi|} \quad \sin \frac{|\Phi|}{2} \frac{\Phi_y}{|\Phi|} \quad \sin \frac{|\Phi|}{2} \frac{\Phi_z}{|\Phi|} \right], \quad (8)$$

where $[\Phi_x \ \Phi_y \ \Phi_z] = |\Phi|[\cos\alpha \ \cos\beta \ \cos\gamma]$ is the rotation vector directed along the axis of rotation; α, β, γ are the angles between the axis of rotation and the coordinate frame, and

$$|\Phi| = \sqrt{\Phi_x^2 + \Phi_y^2 + \Phi_z^2}. \quad (9)$$

Both direction cosine matrices and quaternions can represent mutual frame orientation as well as rotations. Thus, in order to use the readings of an accelerometer, one needs to know the orientation (attitude) of the body frame with respect to the inertial frame. Attitude information is commonly provided by a gyroscope. A triaxial rate gyroscope measures the components of the angular velocity vector $[\omega_x \ \omega_y \ \omega_z]$

If the initial orientation of the body frame with respect to the inertial frame is known, then its current orientation at any time moment can be computed using the Poisson equation:

$$\dot{C}_b^I = C_b^I \bar{\omega}_b, \quad (10)$$

where $\bar{\omega}_b$

$$\bar{\omega}_b = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}. \quad (11)$$

Formula (11) has the following background. The rotation matrix derivative is expressed as

$$\dot{C}_b^I = \frac{C_b^I(t + \Delta t) - C_b^I(t)}{\Delta t}. \quad (12)$$

The rotation matrix $C_b^I(t + \Delta t)$ can be written as a product of the rotation matrix of the previous time moment, $C_b^I(t)$, and the matrix of small rotation, A . The latter is supposed to be equal to:

$$A = \begin{pmatrix} 1 & -\Delta\psi & \Delta\theta \\ \Delta\psi & 1 & -\Delta\phi \\ -\Delta\theta & \Delta\phi & 1 \end{pmatrix} = I + \begin{pmatrix} 0 & -\Delta\psi & \Delta\theta \\ \Delta\psi & 0 & -\Delta\phi \\ -\Delta\theta & \Delta\phi & 0 \end{pmatrix}, \quad (13)$$

where $\Delta\psi, \Delta\theta, \Delta\phi$ are small rotations around axes z, y and x , correspondingly. Matrix A is a simplified form of a rotation matrix made under assumption that for infinitesimal rotation angles $\sin\Delta\psi = \Delta\psi, \sin\Delta\theta = \Delta\theta, \sin\Delta\phi = \Delta\phi, \cos\Delta\psi = \cos\Delta\theta = \cos\Delta\phi = 1$. Division of small angles $\Delta\psi, \Delta\theta, \Delta\phi$ by Δt yields the components of the angular velocity vector.

A quaternion analog of the Poisson equation is:

$$\dot{q} = \frac{1}{2} q \otimes \omega, \quad (14)$$

where q is the current quaternion, \dot{q} is the derivative, ω is the quaternion $\omega = [0 \ \omega_x \ \omega_y \ \omega_z]$ and \otimes denotes the quaternion product.

For small rotation one can use the following formula:

$$q_{k+1} = q_k + \frac{1}{2} q_k \otimes \omega \Delta t. \quad (15)$$

As in the case of direction cosine matrices, for small rotations $\sin \frac{|\Phi|}{2} = \frac{|\Phi|}{2}$ and $\cos \frac{|\Phi|}{2} = 1$, thus a quaternion of small rotation is determined as:

$$q = \left[1 \quad \frac{1}{2} \omega_x \Delta t \quad \frac{1}{2} \omega_y \Delta t \quad \frac{1}{2} \omega_z \Delta t \right]. \quad (16)$$

Formulas (1) – (16) are widely covered in the scientific literature [10, 11]. Existing software for simulation of signals from inertial sensors [12] relies on them without disclosing subtle details. Mechanisms of direction cosine matrices and quaternions are often used interchangeably and implicitly supposed to provide the same accuracy.

If the signals are noise-free and all the initial conditions are set correctly, one expects to restore the originally defined trajectory exactly upon the simulated signals from inertial sensors. However, in practice this expectation may not be met when the body rotates along more than one axis. Our numerical experiments show that: 1) the trajectory itself influences the discrepancies between its original and restored versions; 2) some trajectories are almost irreproducible, at least at reasonable sample frequencies; 3) the discrepancies depend on the sample frequency: the higher the sample frequency, the better; however, starting with some threshold sample frequency its further increase shows no or only insignificant improvement; 4) the extent to which the trajectory

will be restored depends on the mathematical background: quaternions generally demonstrate better accuracy than direction cosine matrices.

The aim of the work is to explain the discrepancies between the preset and re-calculated trajectories, compensate for them, detect the dependency of the discrepancies on the trajectory itself and the sample frequency and form the recommendations on choosing between direction cosine matrices and quaternions.

Mathematical background, numerical experiments and algorithm description. The above-stated observations on discrepancies are illustrated by Figure 1–2.

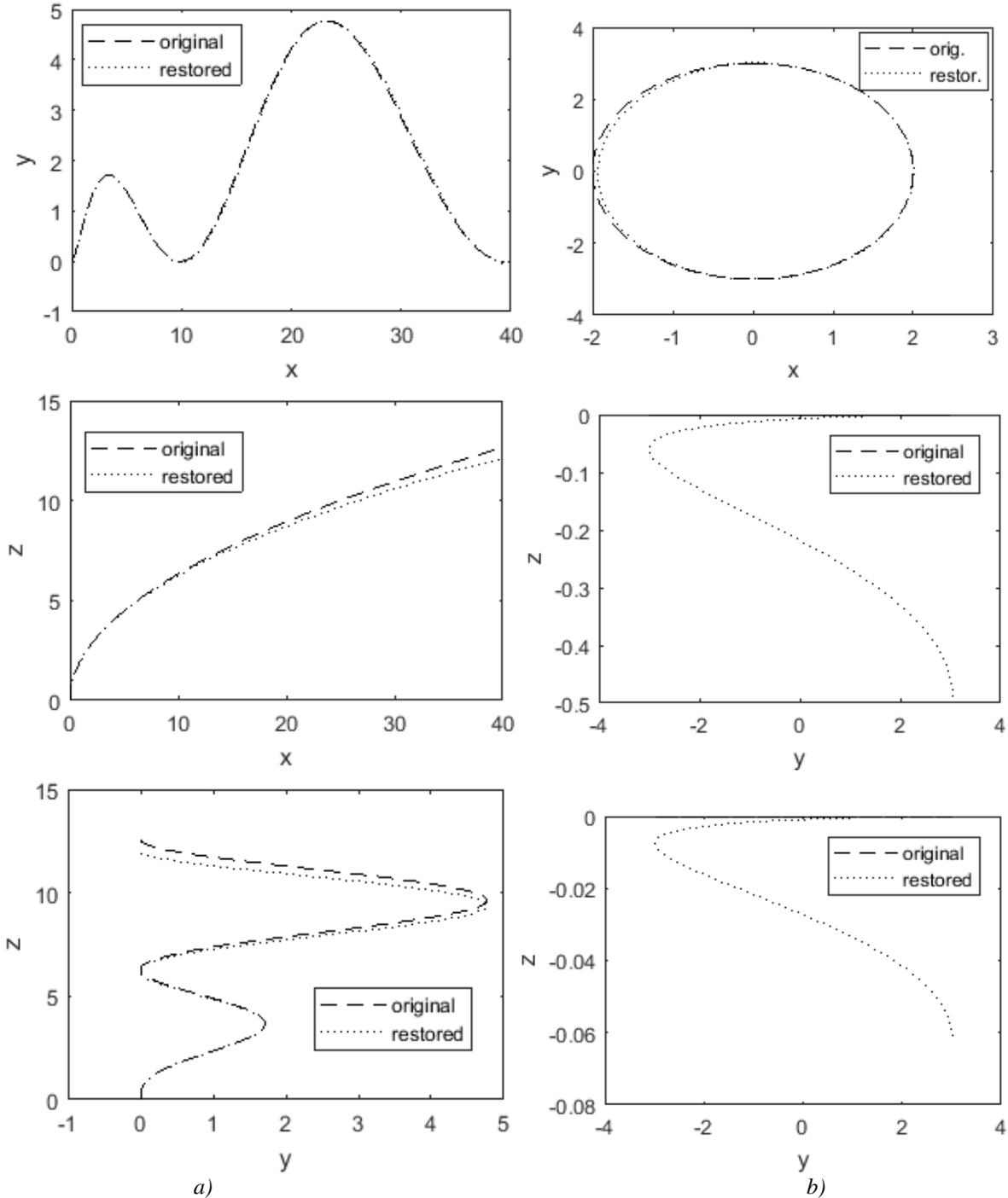


Fig. 1. The original and restored trajectories: a) a curve $x = t^2$, $y = t \sin^2 t$, $z = 2t$, sampled at 100 Hz; both signal simulation and trajectory restoring were performed using direction cosine matrices, with simultaneous rotation around all the axes; b) a curve $x = 2 \sin t$, $y = 3 \cos t$, $z = 0$, with rotation around all the axes; both signal simulation and trajectory restoring were performed using direction cosine matrices; the upper y-z plot was obtained at 100 Hz, the lower one – at 800 Hz

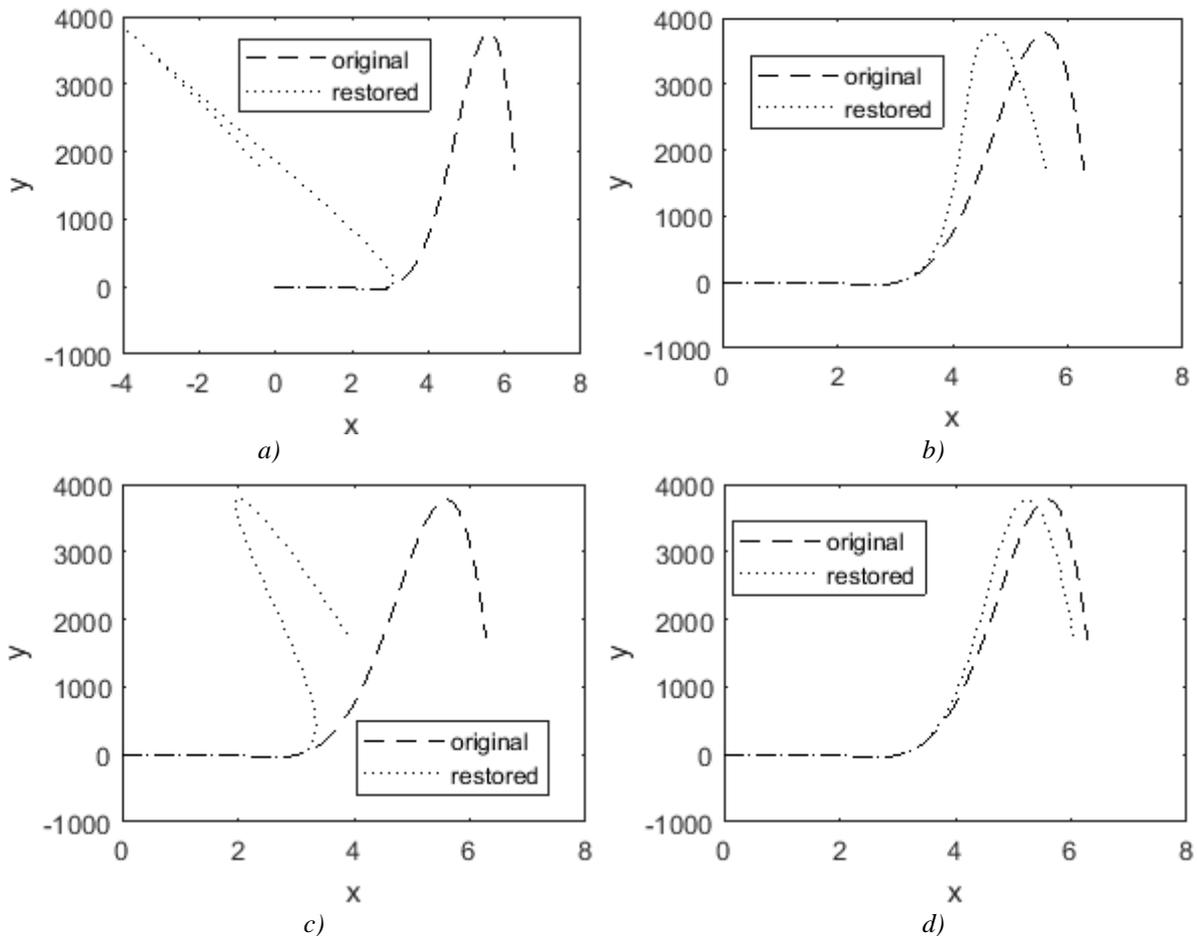


Fig. 2. The original and restored trajectory $y = 0,025t^6 - 0,835t^5 \sin t + 0,425t^3 \cos t + 1,25t^2 + 0,25t$, $x = t$, $z = (1/2\pi)t$ with the following sample frequency and rotation representation method used: a) 100 Hz, direction cosine matrices; b) 1000 Hz, direction cosine matrices; c) 100 Hz, quaternions; d) 1000 Hz, quaternions

A choice of a high-degree polynomial may seem a bit far-fetched. Nevertheless, the code should be universal and workable for any trajectory the user thinks fit to preset. It is worth bearing in mind that the sample frequency of real accelerometers and gyroscopes is limited to several options and usually does not exceed 1000 Hz. For instance, the maximum sample frequency of BNO055 sensor used in Yocto-3D-V2 is 100 Hz. There is no point in using higher frequencies in simulation, because the model would disagree with the real parameters of the sensors. As can be seen from Figure 2, quaternions provide somewhat better accuracy than rotation matrices.

Upon formulas (1) – (16) one applies the following algorithm for simulation of the output signals of «ideal» accelerometers and gyroscopes (Part A):

1. Prescribe a translational motion scenario. The results of this step are 7 one-dimensional arrays: T , which represents the motion time period, X , Y , and Z , which define the x -, y - and z -coordinates of the body at each time moment t , and V_x , V_y , V_z which indicate the motion velocity along the three axes. In the simplest case V_x velocity can be calculated as $(x_{k+1} - x_k)/dt$, where $dt = 1/f_{sample}$. Other velocity vector components may be computed in a similar way. The simplest way of setting the x -, y - and z -coordinates of the trajectory is to define analytical dependencies $x(t)$, $y(t)$, and $z(t)$ and perform their time-quantization;

2. Using V_x , V_y , V_z calculate A_x , A_y , A_z which are the acceleration vector components expressed in the inertial frame. The acceleration along the x -axis is computed as $A_{x_k} = (V_{x_{k+1}} - V_{x_k})/dt$, and the two other components are found in a similar way;

3. Preset a rotational motion scenario. In practice, the heading and velocity vectors correlate. However, for simplicity reasons we assume that the body attitude changes in an arbitrary way, regardless the translational motion. Obviously, the sample frequency for rotational motion should be the same as for translational one, because one needs to know the attitude of the body frame related to the current accelerometer readings;

4. At each step calculate the attitude of the body frame with respect to the inertial frame;

5. Obtain the angular velocity vector;

5.1. In case if direction cosine matrices are used, the angular velocity vector can be computed as follows.

First, one needs to define the rotation matrix derivative as:

$$\dot{C}_b^I = \frac{C_{b_{k+1}}^I - C_{b_k}^I}{dt}, \quad (17)$$

where the old and new rotation matrices, indexed (k + 1) and (k) correspondingly, are obtained using the preset values of the Euler angles. Then an angular velocity matrix, Ω_{orig} , is computed:

$$\Omega_{orig} = (C_{b_k}^I)^T \dot{C}_b^I, \quad (18)$$

The subscript «orig» is meant to underline that the matrix has been calculated in Part A, during signals simulation. Elements $-\Omega_{orig}(2,3)$, $\Omega_{orig}(1,3)$ and $-\Omega_{orig}(1,2)$ represent the components ω_x , ω_y , ω_z of the angular velocity vector, respectively;

5.2. In case if quaternions are used, the angular velocity vector can be calculated as

$$[\omega_x \quad \omega_y \quad \omega_z] = \left[\frac{2q_{S.R.2}}{dt} \quad \frac{2q_{S.R.3}}{dt} \quad \frac{2q_{S.R.4}}{dt} \right], \quad (19)$$

where $q_{S.R.}$ is a quaternion of small rotation:

$$q_{S.R.} = (q_{k+1} \otimes q_k^* / \|q_k\|)^*. \quad (20)$$

Quaternions q_{k+1} and q_k describe the current and previous attitude of the body frame with respect to the inertial frame. All the angular velocity vector components are stored in three arrays G_x , G_y , G_z . In this way the signals of a triaxial gyroscope are simulated;

6. Having figured out the current attitude of the body, express the acceleration vector components in the body frame. With direction cosine matrices, one writes:

$$[a_{bx}, a_{by}, a_{bz}] = (C_{b_{k+1}}^I)^T (a_{ix} \quad a_{iy} \quad a_{iz} - g)^T. \quad (21)$$

In the case of quaternions, one obtains:

$$[0 \quad a_{bx} \quad a_{by} \quad a_{bz}] = (q_{k+1} \otimes [0, a_{ix}, a_{iy}, a_{iz} - g]) \otimes q_{k+1}^*. \quad (22)$$

In this way, the signals of a triaxial accelerometer are simulated.

In order to restore the original trajectories, defined at Part A. Step 1 one should take the steps (Part B):

1. Input simulated signals A_x , A_y , A_z , G_x , G_y , G_z , initial velocities V_{x_0} , V_{y_0} and V_{z_0} , initial coordinates x_0 , y_0 and z_0 and initial Euler angles that define the orientation of the body frame with respect to the inertial frame;

2. Calculate the current body attitude using direction cosine matrices:

$$C_{b_{k+1}}^I = C_{b_k}^I + C_{b_k}^I \Omega_{restored} dt, \quad (23)$$

where

$$\Omega_{restored} = \begin{pmatrix} 0 & -G_{z_k} & G_{y_k} \\ G_{z_k} & 0 & -G_{x_k} \\ -G_{y_k} & G_{x_k} & 0 \end{pmatrix}, \quad (24)$$

is the matrix composed of the simulated signals of a gyroscope at each step, and this fact is reflected in the subscript «restored». Alternatively, quaternions may be used:

$$\Delta q = \left[1 \quad \frac{1}{2} G_{x_k} \cdot dt \quad \frac{1}{2} G_{y_k} \cdot dt \quad \frac{1}{2} G_{z_k} \cdot dt \right], \quad (25)$$

$$q_{k+1} = \Delta q^* \otimes q_k. \quad (26)$$

Part A and Part B may rely on different mathematical background. Particularly, signals can be simulated using direction cosine matrices in Part A and processed using quaternions in Part B, and vice versa.

3. Represent the simulated accelerometer signals in the inertial frame and compensate for gravity using

$$[a_{ix} \quad a_{iy} \quad a_{iz}] = C_{b_{k+1}}^I [A_{x_k} \quad A_{y_k} \quad A_{z_k}]^T + [0 \quad 0 \quad g] \quad (27)$$

or

$$[0 \quad a_{ix} \quad a_{iy} \quad a_{iz}] = (q_{k+1}^* \otimes [0 \quad A_{x_k} \quad A_{y_k} \quad A_{z_k}]) \otimes q_{k+1} + [0 \quad 0 \quad 0 \quad g]. \quad (28)$$

4. Using (2), obtain the velocity vector;

5. Using (3), obtain the coordinates x , y , z .

Our numerical experiments have shown that matrix Ω_{orig} is never strictly skew-symmetrical. Firstly, it has small but still no-zero values on its principal diagonal. Secondly, the absolute values of its off-diagonal values differ, and the difference depends on how quickly the body rotates around each axis. For instance, if the body moves L degrees around x -axis, N degrees around y -axis and M degrees around z -axis and $L < M < N$, then the difference between $|\Omega_{orig}(1,3)|$ and $|\Omega_{orig}(3,1)|$ will be the greatest, followed by $(|\Omega_{orig}(1,2)| - |\Omega_{orig}(2,1)|)$ and then $(|\Omega_{orig}(2,3)| - |\Omega_{orig}(3,2)|)$. A triaxial gyroscope provides only three signal values at a time, thus one has no choice but to compose a matrix $\Omega_{restored}$ of form (24) using the only three available values. Because matrices Ω_{orig} and $\Omega_{restored}$ differ, so do the trajectories. The greater the difference between Ω_{orig} and $\Omega_{restored}$, the less the restored trajectory resembles the original one.

Let us deduce analytical expressions for matrix Ω_{orig} and estimate how the sample frequency influences its deflection from being perfectly skew-symmetrical. We denote ψ_k, θ_k, ϕ_k the yaw, pitch and roll angles describing the body attitude at time moment t , $\psi_{k+1}, \theta_{k+1}, \phi_{k+1}$ – the corresponding angles at time moment $(t + dt)$; $\delta\psi = \psi_{k+1} - \psi_k$, $\delta\theta = \theta_{k+1} - \theta_k$, $\delta\phi = \phi_{k+1} - \phi_k$. The analytical expression for Ω_{orig} is

$$\Omega_{orig} = \left(\left(C_{b_k}^I \right)^T C_{b_{k+1}}^I - I_3 \right) / dt, \quad (29)$$

where I_3 is a 3×3 identity matrix, $C_{b_k}^I$ can be expressed using (5) as:

$$C_{b_k}^I = \begin{pmatrix} \cos\psi_k \cos\theta_k & \sin\psi_k \cos\theta_k & -\sin\theta_k \\ -\sin\psi_k \cos\phi_k + \cos\psi_k \sin\theta_k \sin\phi_k & \cos\psi_k \cos\phi_k + \sin\psi_k \sin\theta_k \sin\phi_k & \cos\theta_k \sin\phi_k \\ \sin\psi_k \sin\phi_k + \cos\psi_k \sin\theta_k \cos\phi_k & -\cos\psi_k \sin\phi_k + \sin\psi_k \sin\theta_k \cos\phi_k & \cos\theta_k \cos\phi_k \end{pmatrix}. \quad (30)$$

Matrix $C_{b_{k+1}}^I$ looks similarly, with angles $\psi_{k+1}, \theta_{k+1}, \phi_{k+1}$ instead, and is omitted for brevity.

Multiplication of $\left(C_{b_k}^I \right)^T C_{b_{k+1}}^I$ yields a new matrix, C , which elements are presented by rather cumbersome trigonometrical expressions. Since we are interested in the influence of the sample frequency on matrix Ω_{orig} regardless any particular angles $\psi_k, \theta_k, \phi_k, \psi_{k+1}, \theta_{k+1}, \phi_{k+1}$, we applied trigonometric formulas to introduce sines and cosines of $\delta\psi, \delta\theta, \delta\phi$ whenever it is possible. Thus, we obtained

$$\Omega_{orig}(1,1) = \frac{1}{dt} \begin{bmatrix} \cos\delta\theta \cdot \cos\delta\psi + \sin\psi_k \sin\psi_{k+1} (\cos\delta\phi - \cos\delta\theta) - \\ \sin\psi_k \cos\psi_{k+1} \sin\theta_{k+1} \sin\delta\phi + \cos\psi_k \sin\psi_{k+1} \sin\theta_k \sin\delta\phi + \\ \cos\psi_k \cos\psi_{k+1} \sin\theta_k \sin\theta_{k+1} (\cos\delta\phi - 1) - 1 \end{bmatrix}, \quad (31)$$

$$\Omega_{orig}(2,2) = \frac{1}{dt} \begin{bmatrix} \cos\psi_k \cos\psi_{k+1} \cos\delta\phi + \\ \sin\delta\phi (\cos\psi_k \sin\psi_{k+1} \sin\theta_{k+1} - \sin\psi_k \cos\psi_{k+1} \sin\theta_k) + \\ \sin\psi_k \sin\psi_{k+1} (\cos\delta\theta \cos\delta\phi + \cos\theta_k \cos\theta_{k+1} (1 - \cos\delta\phi)) - 1 \end{bmatrix}, \quad (32)$$

and

$$\Omega_{orig}(3,3) = (\cos\delta\theta \cdot \cos\delta\phi - \sin\theta_{k+1} \sin\theta_k (\cos\delta\phi - 1) - 1) / dt. \quad (33)$$

If one assumes $\cos\delta\theta = 1$ and $\cos\delta\phi = 1$, then $\Omega_{orig}(3,3) = 0$. However, in practice such an assumption causes slight errors whose cumulative effect leads to large discrepancies. Figure 3 shows the differences $(1 - \cos\alpha)$ and $(\alpha - \sin\alpha)$ for α running the values through 0 to 5° (0,0872 rad). As one can see the assumption $\cos\alpha = 1$ contributes much greater errors to calculations than the assumption $\sin\alpha = \alpha$.

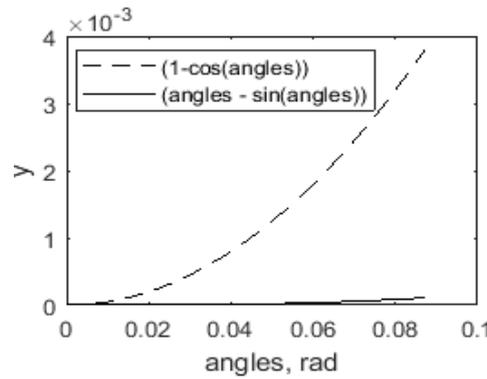


Fig. 3. Cosines and sines of small angles

Taking into account this fact we replace $\sin \theta_{k+1}$ with θ_{k+1} and $\sin \theta_k$ with θ_k and simplify $\Omega_{orig}(3,3)$ to $\Omega_{orig}(3,3) = (\cos \delta\theta \cdot \cos \delta\phi - (\theta_k^2 + \theta_k \delta\theta)(\cos \delta\phi - 1) - 1) / dt$. For instance, for $\delta\theta = 0,1^\circ$, $\delta\phi = 0,1^\circ$ and $\theta = 30^\circ$, which is a quite realistic scenario, at the sample frequency 100 Hz, $\Omega_{orig}(3,3) = -2,63e^{-4} \neq 0$.

Similar reasoning is valid for $\Omega_{orig}(1,1)$ and $\Omega_{orig}(2,2)$. It explains why one observes perfect coincidence of trajectories in x-y planes when there are no rotations around axes x and y and angles ϕ and θ are equal to 0.

Thus, formulas (13), (16), (24) and (25) can be applied for infinitesimal rotations, but this is not the case with accelerometers and gyroscopes whose signals can be sampled with a limited frequency.

Since matrix representation of Ω_{orig} contains angles $\psi_k, \theta_k, \phi_k, \psi_{k+1}, \theta_{k+1}, \phi_{k+1}$ even after simplification and, moreover, direction cosine matrices are rather redundant, compensation for errors may become extremely difficult if not impossible. On the contrary, quaternions provide much less complicated dependencies between angles of rotations and thus are more convenient to deal with. Taking into account the demonstrated effect of misapplied assumptions $\cos \alpha = 1, \sin \alpha = \alpha$, we suggest the following corrections for quaternions of small rotations both in Parts A and B of the signal simulation algorithm:

1. Calculate the quaternion defined by formula (20), at each Part A. Step 5.2;

2. In Part A, compute $\Phi = 2 \arccos(q_1)$. If $\Phi = 0$, then apply the classic textbook formula (19) for simulation of the angular velocity vector. Otherwise amend the formulas as follows:

$$\omega_x = \frac{q_{S.R.}(2) \cdot |\Phi|}{\sin \frac{|\Phi|}{2} dt}; \omega_y = \frac{q_{S.R.}(3) \cdot |\Phi|}{\sin \frac{|\Phi|}{2} dt}; \omega_z = \frac{q_{S.R.}(4) \cdot |\Phi|}{\sin \frac{|\Phi|}{2} dt}; \tag{34}$$

3. In Part B, compose the quaternion upon the angular velocity vector components as follows. Compute

$$\Phi = dt \sqrt{G_x^2 + G_y^2 + G_z^2}. \tag{35}$$

If $\Phi = 0$, then apply the classic textbook formula (25). Otherwise compose the following quaternion:

$$\Delta q = \begin{bmatrix} \cos \frac{|\Phi|}{2} & G_x \sin \frac{|\Phi|}{2} \frac{dt}{|\Phi|} & G_y \sin \frac{|\Phi|}{2} \frac{dt}{|\Phi|} & G_z \sin \frac{|\Phi|}{2} \frac{dt}{|\Phi|} \end{bmatrix}. \tag{36}$$

The proposed corrections allowed us to restore any trajectory exactly as it has been set by the user. The modified formulas (19), (34), (25) and (36) have been intensively verified on the same set of trajectories, which had been used for studying the discrepancies between the trajectories built during simulation.

Conclusions. We have considered the problem of using direction cosine matrices and quaternions for simulation of signals from triaxial accelerometers and gyroscopes. Upon the analytical reasoning corroborated by numerical experiments, we can conclude that: 1) sample frequencies supported by real accelerometers and gyroscopes (up to 1000 Hz) are not sufficient to consider rotation angles infinitesimal, and for this reason classic navigation equations widely proposed in the scientific literature provide principally erroneous results, which has been demonstrated on analytical expressions for items of the direction cosine matrices; 2) the direction cosine matrices become non-skew-symmetrical, however this effect is difficult to take into account and compensate for because the underlying expressions contain Euler angles that change from step to step; 3) in contrast to the direction cosine matrices, quaternions of small rotation can be easily corrected; the correction to the navigation equations we have proposed allows us to eliminate completely any discrepancies between the originally simulated trajectory and the trajectory restored from simulated signals of accelerometers and gyroscopes.

References:

1. El-Gohary, M. (2013), «Joint angle tracking with inertial sensors», Ph.D. Thesis of dissertation, Portland State University, Portland, 143 p.
2. Bulling, A., Blanke, U. and Schiele, B. (2014), «A tutorial on human activity recognition using body-worn inertial sensors», *ACM Computing Surveys*, Vol. 46 (3), pp. 1–33, doi: 10.1145/2499621.
3. Ngo, T., Makihara, Y., Nagahara, H., Mukaigawa, Y. and Yagi, Y. (2015), «Similar gait action recognition using an inertial sensor», *Pattern Recognition*, Vol. 48 (4), pp. 1289–1301, doi: 10.1016/j.patcog.2014.10.012.
4. Chen, C., Jafari, R. and Kehtarnavaz, N. (2015), «A survey of depth and inertial sensor fusion for human action recognition», *Multimedia Tools and Applications*, Vol. 76 (3), pp. 4405–4425, doi: 10.1007/s11042-015-3177-1.
5. Daroogheha, S., Lasky, T. and Ravani, B. (2018), «Position measurement under uncertainty using magnetic field sensing», *IEEE Transactions on Magnetics*, Vol. 54 (12), pp. 1–8, doi: 10.1109/tmag.2018.2873158.
6. Shmaliy, Y., Zhao, S. and Ahn, C. (2019), «Optimal and unbiased filtering with colored process noise using state differencing», *IEEE Signal Processing Letters*, Vol. 26, Issue 4, pp. 548–551.
7. Lin, X., Jiao, Y. and Zhao, D. (2018), «An improved Gaussian filter for dynamic positioning ships with colored noises and random measurements loss», *IEEE Access*, Vol. 6, pp. 6620–6629.
8. Marusenkova, T. (2020), «An algorithm for detecting the minimum allowable accelerometer sample rate for tracing translational motion along a Bézier curve», *Tehnichna inzhenerija*, Vol. 1 (85), pp. 147–154, doi: 10.26642/ten-2020-1(85)-147-154.
9. Fedasyuk, D. and Marusenkova, T. (2020), «An algorithm for detecting the minimal sample frequency for tracking a preset motion scenario», *International Journal of Intelligent Systems and Applications*, Vol. 12, No 4, pp. 1–12, doi: 10.5815/ijisa.2020.04.01.
10. Titterton, D. and Weston, J. (2004), *Strapdown Inertial Navigation Technology*, 2nd ed., The American Institute of Aeronautics and Astronautics, 576 p.
11. Britting, K.P. (2010), *Inertial Navigation System Analysis*, Artech House, Boston, 249 p.
12. Woodman, O. (2007), *An introduction to inertial navigation*, University of Cambridge Computer Laboratory, Cambridge, pp. 21–37.

Marusenkova Tetiana Anatoliivna – Ph.D. in Engineering, Associate Professor of Lviv Polytechnic National University.

Scientific interests:

- inertia sensors;
- mathematical modelling;
- software engineering.

<http://orcid.org/0000-0003-4508-5725>.

E-mail: tetyana.marus@gmail.com.

The article was sent to the editorial board on 01.09.2020.