Some Algorithms for Unmanned Aerial Vehicles Navigation

Anatol Azarievich Tunik
National Aviation University
91 PUBLICATIONS **222** CITATIONS
SEE PROFILE

Vladimir Borisovich Larin
180 PUBLICATIONS **1,386** CITATIONS
SEE PROFILE

Svitlana Ilnytska
National Aviation University
8 PUBLICATIONS **3** CITATIONS
SEE PROFILE

TECHNOLOGY & ENGINEERING / Aeronautics & Astronautics

This book presents simplified navigation algorithms suitable to onboard equipment of small UAVs and their flight missions.

The algorithms presented in this book were designed to achieve an acceptable trade-off between contradictive requirements to the software of small UAV navigation systems: sufficient accuracy and reliability in order to perform required flight missions on the one hand, and acceptable cost and simplicity of this software on the other hand. The core of modern navigation systems is integrated Strapdown Inertial Navigation System (SINS) and GPS, so in this book, the SINS algorithms and the algorithms of sensor fusion are described primarily. Inertial sensors (rate gyros and accelerometers) used in SINS are manufactured on the basis of the MEMS-technology. That is why they possess poor accuracy and need to be corrected with other sensors (GPS, magnetometers, and barometric altimeters). It is necessary to take into account that flight missions of small UAVs are characterized by small flight distances, small flight times, small flight speeds, etc. These properties of small UAV flight missions and properties of MEMS-sensors create a practical background for simplification of the SINS algorithms, simultaneously preserving their accuracy at acceptable levels. The navigation algorithms for gyro-free SINS are also considered. Increasing reliability of the UAV navigation systems requires a solution of the problems of the detection of the faulty sensors. These algorithms are described. Some practical aspects of the operation of navigation systems such as initial alignment, sensors calibration, and laboratory, ground, and flight testing of integrated SINS for small UAVs are also presented.

This book will be useful for a wide circle of researchers, engineers, and graduate students involved in modern UAV design and manufacturing.

Vladimir B. Larin, Doctor of Sci. (phys. & math.), Ph.D. (Eng.), professor, principal researcher of the Department of Processes Stability at the S. P. Timoshenko Institute of Mechanics of the National Academy of Sciences of Ukraine, laureate of the State Prize of the Ukrainian SSR in the field of science and technology.

He is a member of the National Committee of Ukraine on Theoretical and Applied Mechanics, Association of Automatic Control, and the American Mathematical Society.

Anatoly A. Tunik, Doctor of Sci. (Eng.), professor of the National Aviation University, Aerospace Control System Department, Kiev, Ukraine. He was Visiting Professor at the University of Alaska, Anchorage (USA, 1995), and Visiting Researcher in the Korean Aerospace Research Institute, Daejeon (S. Korea, 1997–2001). He is IEEE Life Senior Member and Member of International Academy of Navigation and Motion Control.

Svitlana I. Ilnytska, Ph.D. (Eng.), Senior Researcher at the "Aerospace Center" of the National Aviation University, Kiev, Ukraine. She was awarded a prize for young scientists in 2017 by the president of Ukraine for her work on the "Perspective integrated navigation of Unmanned Aviation Systems."

V1.0

outskirts press | outskirtspress.com

Algorithms for Unmanned Aerial Vehicle Navigation Systems

Vladimir Larin, Anatoly Tunik, Svitlana Ilnytska

outskirts press

Some algorithms for Unmanned Aerial Vehicles Navigation

V.B. Larin, A.A. Tunik, S.I. Ilnytska

Some algorithms for unmanned aerial vehicles navigation systems

V.B. Larin, A.A. Tunik, S.I. Ilnytska

CONTENT

Acknowledgements

Authors acknowledge Professor V.P. Kharchenko, Vice-Rector on Research of National Aviation University (NAU), for giving opportunity to work on this topic at NAU and his constant support, V.M. Kondratyuk, Director of Research and Training "Aerospace Center" of NAU, and other colleagues from the "Aerospace Center", namely, O.V. Kutsenko, Ye.V. Vyshnyakova, V.P. Trikoz, for their significant assistance in implementation of the presented algorithms in the hardware-software complexes, support in organization and performing of numerous experimental researches. Authors acknowledge Professor Victor V. Larin of Pace University in New York for his efforts to see this book published.

Eventually we wish to thank our families for their love and support.

Vladimir B. Larin
Anatoly A. Tunik
Svetlana I. Ilnitska

CHAPTER 1 – INTRODUCTION

Nowadays design and manufacturing of small Unmanned Aerial Vehicles (UAV) is the most dynamically developing branch of high technologies (Beard and McLain, 2012), (Austin, 2010). One of the key problems in this area is the creation of the airborne integrated navigation and flight control systems (Beard and McLain, 2012), (Siouris, 1993, 2007), (Grewal, Weill and Andrews, 2001). The basic requirements to these systems are contradictive: from one hand they must possess sufficient accuracy and reliability in order to perform complicated flight missions, and from the other hand they must be acceptably cheap and simple for conforming to the same properties of small UAV.

Taking in account the last remark and considering the Strapdown Inertial Navigation Systems (SINS), which are the main navigation core for small UAV, it is necessary to note that they use sufficiently cheap Micro-Electro-Mechanical Sensors (MEMS) as the basic components (accelerometers and rate gyros) of the Inertial Measurements Unit (IMU). However, these sensors possess low accuracy and need to be corrected with other sensors: GPS, magnetometers, and barometric altimeters, as well as with usage of corresponding algorithmic means for their data fusion (Beard and McLain, 2012), (Austin, 2010), (Siouris, 1993, 2007), (Grewal, Weill and Andrews, 2001). It is necessary also to note that the SINS software consists of sophisticated algorithms of rotational and translational mechanizations along with algorithms of the SINS integration with other aforementioned sensors (Siouris, 1993, 2007), (Grewal, Weill and Andrews, 2001). Some very precise SINS algorithms cannot be implemented in cheap onboard microprocessors due to their insufficient computational power. This circumstance creates very important problem, consisting in the development of the SINS software, which will possess sufficient accuracy of the navigation problem solution, and simultaneously sufficient low cost and simplicity for implementation in the small UAV hardware. The only way of finding acceptable compromise between these two mutually exclusive requirements is taking into account the basic properties of small UAV flight missions, notably: small flight distances, small flight times, small flight speeds etc. These circumstances create practical background for simplification of the SINS algorithms, simultaneously preserving their accuracy at the acceptable level, which is reasonable for these flight missions.

It should be noted that in the process of the SINS correction some sensors failures of the navigation system might be detected. In this connection very important problem of the faulty element detection arises.

It is also appreciable to note, that MEMS sensors are not always able to provide required range of the UAV angular rates measurements. This is the reason of the intensive development of the gyro-free accelerometer based SINS (Hajiyev and Guler, 2017), and some aspects of this problem are considered in this monograph.

Eventually it must be noted, that practical implementation of mentioned above algorithms is tightly connected with laboratory, ground and flight experiments. These experiments require certain test and telecommunication equipment, data processing software, and methods of experiment performing. These aspects are also the object of research work undertaken by authors.

Beyond the general characteristic of problems, which are discussed in this monograph, it is expedient to describe briefly its structure and contents of each its chapter.

In the 2-nd chapter the SINS software structure is considered, and the basic relations of the analytical mechanics, which are the mathematical background of this software, are given.

The definitive peculiarity of the rotational mechanization algorithms, which are described in this chapter, is the usage of procedures based on the analytical expressions for approximations of quaternions corresponding to the rigid body small turn. Several methods of these approximations allow variation of the accuracy of kinematic equations integrating, thus making a trade-off between complexity and accuracy of algorithms. The usage of the quadratic spline approximation for quasi-coordinates essentially simplifies the procedures of rotational and translational mechanizations and preserves the accuracy of integration.

The algorithms of rotational and translational mechanization for gyro-free accelerometer-based SINS are also considered in this chapter. Nowadays creation of such systems attracts considerable attention of researchers (Hajiyev and Guler, 2017). These algorithms are especially effective for UAV spinning respectively longitudinal axis with large angular rate. It is shown, that increasing of number of the redundant accelerometers above minimal value leads to essential increasing of the accuracy of the rotational and translational mechanization’s algorithms of gyro-free SINS.

The 3-rd chapter is devoted to the algorithms of SINS correction using external with respect to SINS sensors: GPS, magnetometer and barometric altimeter. Several variations of Kalman filtering algorithms are considered and special attention is paid for usage of the generalized Cholesky factors for improvement of the Kalman filtering procedure convergence.

Algorithms of the GPS, magnetometer, and altimeter data fusion are also proposed and analyzed along with the gyro’s bias compensation. The efficiency of the theoretical results achieved in this chapter is proved by mathematical modeling.

In the 4-th chapter the methods of SINS initial alignment and calibration of IMU sensors are described. Initial alignment procedures are very important for increasing the accuracy of integration procedures, which are key parts of the rotational and translational mechanization. As far as the accuracy of the MEMS sensors is reduced, and in particular the MEMS gyros can’t measure the Earth rotation rate, the simplest methods of initial alignment procedures based on the accelerometers and magnetometer readouts are considered. Numerical examples prove the efficiency of these methods. The calibration of the MEMS sensors used in the SINS has also significant importance for evaluation of the parameters of the IMU mathematical model, which is used in the SINS algorithms. In this chapter the simplest means for sensors calibration is proposed, when simplicity low cost of the motion table is balanced with algorithms of the experimental data processing. Results of experiments with real accelerometers and rate gyros are given.

The 5-th chapter is devoted to the solution of fault detection problem in system with redundant sensors. The structure of the fault detection system is based on the principle of analytical redundancy, proposed in (Desai, Deckert and Deyst, 1979), Deyst et.al, 1981), and some simple algorithms created on this principle. This software is augmented with several algorithms of various complexities. The fault detection procedure is based on monitoring of SINS outputs, including all algorithms of rotational and translational mechanization and errors correction. If some statistics obtained by monitoring algorithm exceeds certain prescribed value, then it indicates appearance of fault of some sensor. In this case the outputs of the redundant sensors have to be processed by one of three algorithms, which can identify faulty sensor. The choice from these 3 algorithms might be done by designer taking into account his/her preferences or some other considerations. Note, that in this chapter we consider only the fault detection problem. The reconfiguration problem is not represented here.

Eventually the 6-th chapter is devoted to the laboratory, ground and flight testing of integrated SINS for small UAV. Laboratory testing is based on the simplest mechanical rotating motion table and more sophisticated radio engineering methods of data acquisition and processing. Ground testing is based on the estimation of the accuracy of navigation problem solution in condition of walking with SINS equipment and its moving by automobile. Flight testing was made using small UAV controlled manually as well as automatically. Results of flight experiments were compared with the same results obtained by some COTS-available SINS. This comparison along with results of other testing proves ability of proposed algorithms to be applied for the small UAV navigation problem solution.

CHAPTER 2 – ALGORITHMS OF THE STRAP-DOWN INERTIAL NAVIGATION SYSTEMS FUNCTIONING

2.1. Structure of the Strap-down Inertial Navigation Systems Operation Algorithms

The basic feature of the strap-down inertial navigation system (SINS) lies in a fact, that the current values of the position, velocity and attitude defined by this system can be obtained by integrating the readouts of the UAV acceleration and angular rate sensors (accelerometers and rate gyros (RG)), which are rigidly mounted onboard (Titterton and Weston, 2004; Siouris, 1993, 2007; Grewall, Weill and Andrews, 2001). So the raw information is acquired in the UAV body frame but the final solution of the navigation problem must be obtained in some navigation frame. This feature determines the structure of the SINS operation algorithms, because it requires the attitude determination and the mapping of the current values of the accelerations readouts from the body frame to the navigation frame.

Therefore, depends on the choice of navigation frame, the solution of the navigation problem (implementation of navigation algorithms) will be different. *Navigation frame* is the system that is used for numerical integration of differential navigation equations (so called mechanization algorithms). It could be, for example, Earth Centered Inertial (ECI) Frame, Earth Centred Earth Fixed (ECEF) Frame or some Local-level topocentric frame, where the axes are directed North East Down (NED) or East North Up (ENU) (Lurie, 2002; Siouris, 1993, 2007; Titterton and Weston, 2004). The choice of frame depends on many considerations of system designers. For example, for a long-range aircraft the Earth Centered Earth Fixed (ECEF) coordinate system (Titterton and Weston, 2004) is the most often used for navigation in Earth's atmosphere. While the local-level topocentric frame is used as navigation frame in many practical applications of UAV, it is convenient to use such frame for distribution of vertical and horizontal velocity components. NED frame coincides with the axes of body frame (roll/pitch/yaw) when the vehicle is leveled in horizontal plane and is directed to the north. In body frame, direction of axis X (roll axis) usually coincides with that of longitudinal axis of the UAV, axis Y (pitch axis) with that of lateral one, and axis Z (yaw axis) is directed downwards. Such choice of axes is also called airplane axes (Lurie, 2002; Siouris, 1993, 2007). As is seen from Fig. 2.1, counterclockwise angles are positive.

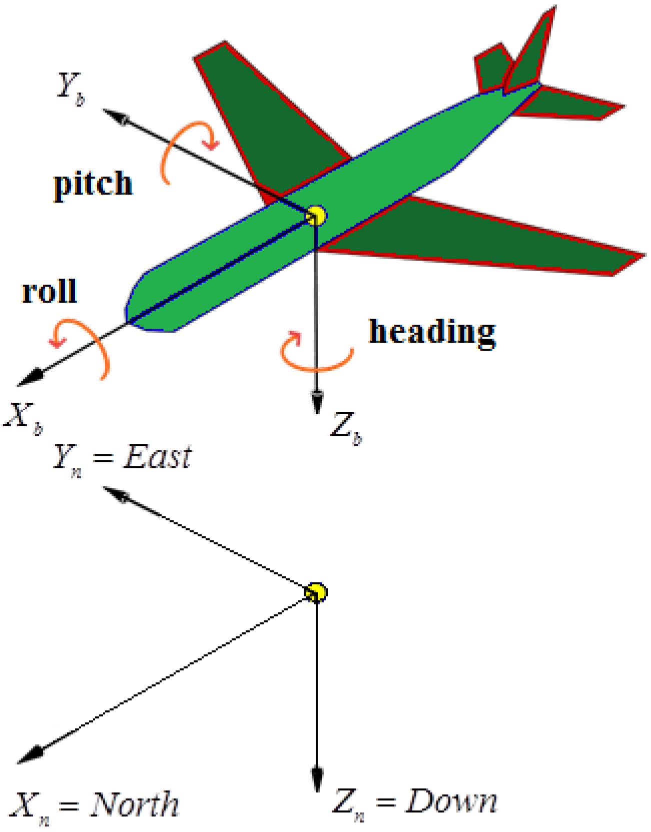


Figure 2.1 – Body and navigation frames

As far as this book is devoted to the navigation algorithms for small UAV, which are flying at much shorter distances, the local topocentric system is used as navigation frame taking into account essential simplification of the navigation problem solution in this case. We can conventionally distinguish the following main stages in the algorithm (Petovello 2003; Ilnytska 2012):

- Initial data input: attitude, position and velocity as initial conditions for solving kinematic differential equations;
- Error estimates: correction of the raw measurements, obtained by inertial sensors, with some other external measurements;
- Rotational Mechanization: integrating rotational kinematic equations to update current attitude values;
- Mapping the accelerations from BF into NF on the basis of current attitude;
- Translational Mechanization: integrating translational kinematic equations to calculate current position and velocity values.

Presenting these stages of solution of navigation equations as respective units, we obtain general SINS computation flow chart (Fig. 2.2), where rectangles denote main calculation units and rounded rectangles denote input/output data. Note, that in practice the rate gyros (RG) and accelerometers are united in the single inertial measurement unit (IMU).

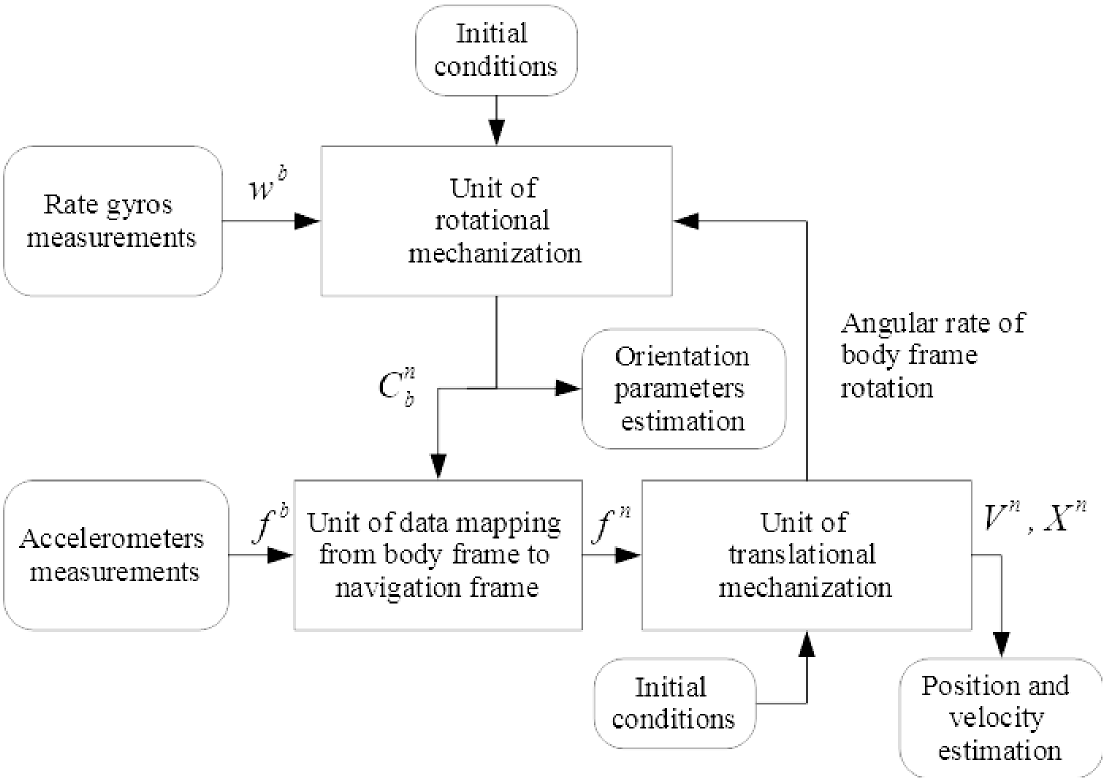

Initial conditions
Rate gyros measurements
w^b
Unit of rotational mechanization
Angular rate of body frame rotation
C_b^n
Orientation parameters estimation
Accelerometers measurements
f^b
Unit of data mapping from body frame to navigation frame
f^n
Unit of translational mechanization
V^n, X^n
Initial conditions
Position and velocity estimation

Figure 2.2 – Flow chart of SINS calculations

Correction of raw measurements of sensors is not represented by a separate unit in Fig. 2.2. Nevertheless, it is carried out either inside inertial measurement units or in onboard computers. Signals at the outputs of accelerometers and RG’s contain, along with useful signals, errors of scale factors, misalignment of sensitivity axes, zero biases, and noise components of errors.

The manufacturers of inertial sensors compensate high frequency noise components of measurement errors by including internal digital filters in the IMU. To correct other errors listed above, it is necessary either to know their numerical values, which can also be provided by the manufacturer in technical specifications, or to obtain them by calibration, which we will consider in more detail in Sec. 4.

Now let us analyze the algorithms for, which are included in the aforementioned basic stages of obtaining navigation solution of SINS. The mathematical background of these algorithms consists of some basic relations, which we consider below.

2.2. Basic Relations

Let us present the well-known relations that pertain to determining angular orientation of a rigid body (Lurie, 2002), (Onishchenko, 1983), (Larin, 2001), (Wittenburg, 1977). We will describe various techniques of the attitude determination. The relations presented in this section are the theoretical basis for all the SINS operation algorithms described below.

The Euler angles ψ, ϑ, φ (precession, nutation, and intrinsic rotation) define body's orientation, i.e., its transformation from the initial position defined by Oxyz axes of the fixed frame to the final position defined by $Ox'y'z'$ axes of moving frame (Fig. 2.3). This transformation can also be carried out by means of one turn by angle χ with respect to the axis whose direction is specified by angles α, β, γ. Therefore, four Rodriguez-Hamilton parameters (Lurie, 2002) $\lambda_0, \lambda_1, \lambda_2, \lambda_3$ (Euler's parameters (Wittenburg, 1977)) can characterize body's orientation:

$$\lambda_1 = \cos\alpha \sin\frac{\chi}{2}, \quad \lambda_2 = \cos\beta \sin\frac{\chi}{2}, \quad \lambda_3 = \cos\gamma \sin\frac{\chi}{2}, \quad \lambda_0 = \cos\frac{\chi}{2} \qquad (2.1)$$

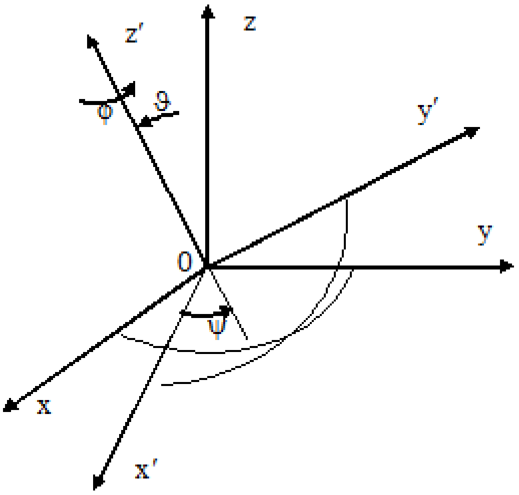


Fig. 2.3. Fixed and moving frames

Obviously: $\lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 = 1$. The Rodrigues-Hamilton parameters can be expressed in terms of Euler's angles as follows (Lurie, 2002):

$$\lambda_0 = \cos\frac{\vartheta}{2}\cos\frac{\varphi+\psi}{2}, \qquad \lambda_1 = \sin\frac{\vartheta}{2}\cos\frac{\psi-\varphi}{2},$$
$$\lambda_2 = \sin\frac{\vartheta}{2}\sin\frac{\psi-\varphi}{2}, \qquad \lambda_3 = \cos\frac{\vartheta}{2}\sin\frac{\varphi+\psi}{2}. \qquad (2.2)$$

Other interpretation of Euler's angles that is widely used on practice are so called "aircraft angles" (roll, pitch and heading) (Lurie, 2002). So, afterwards in practical examples the "aircraft" Euler angles are meant.

In terms of such Euler angles (roll, pitch and heading) the quaternion elements are given in (Siouris, 2007), relations (2.74a) – (2.74d) as the following:

$$\begin{aligned}
\lambda_0 &= \cos\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2}\\
\lambda_1 &= \sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2}\\
\lambda_2 &= \sin\frac{\theta}{2}\cos\frac{\psi}{2}\cos\frac{\phi}{2} - \sin\frac{\psi}{2}\sin\frac{\phi}{2}\cos\frac{\theta}{2}\\
\lambda_3 &= \sin\frac{\phi}{2}\cos\frac{\psi}{2}\cos\frac{\theta}{2} + \sin\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2}
\end{aligned} \tag{2.3}$$

The presented above relations (2.3) define uniquely the quaternion components for a chosen set of roll, pitch and heading angles (Euler angles). However, those relations are not very convenient for initialization due to potential inaccuracies, resulting from numerous multiplications and trigonometric functions evaluations.

We can describe orientation of a rigid body with respect to fixed frame Oxyz by direction cosine matrix (DCM) A of transformation of coordinates (the matrix of cosines of angles between axes of the fixed and moving frames), i.e., if m is some vector in the fixed frame and components of vector k are projections of this vector onto axes of the moving frame $Ox'y'z'$, then

$$k = Am. \tag{2.4}$$

This matrix has the following representation in terms of the Rodriguez-Hamilton parameters λ_0, λ_1, λ_2, λ_3 (Siouris, 1993, 2007), (Lurie, 2002):

$$A(\lambda) = \begin{bmatrix} \lambda_0^2+\lambda_1^2-\lambda_2^2-\lambda_3 & 2(\lambda_1\lambda_2+\lambda_0\lambda_3) & 2(\lambda_1\lambda_3-\lambda_0\lambda_2) \\ 2(\lambda_1\lambda_2-\lambda_0\lambda_3) & \lambda_0^2-\lambda_1^2+\lambda_2^2-\lambda_3^2 & 2(\lambda_2\lambda_3+\lambda_0\lambda_1) \\ 2(\lambda_1\lambda_3+\lambda_0\lambda_2) & 2(\lambda_2\lambda_3-\lambda_0\lambda_1) & \lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2 \end{bmatrix} \tag{2.5}$$

Note that if the trihedrons $Oxyz$ and $Ox'y'z'$ are close (the Euler angles are small), we can use approximate expression (for example (26) in (Wittenburg, 1977)) for matrix $\tilde{\boldsymbol{A}}$:

$$\tilde{\boldsymbol{A}} \cong \begin{bmatrix} 1 & \mu_3 & -\mu_2 \\ -\mu_3 & 1 & \mu_1 \\ \mu_2 & -\mu_1 & 1 \end{bmatrix}, \tag{2.6}$$

where μ_1, μ_2, μ_3 are small angles of rotation of trihedron $Oxyz$ about axes x, y, z, respectively.

Inverse relations also take place. For example, if $A = \lfloor a_{ij} \rfloor$, ij $= \overline{1,3}$ and$1 + a_{11} + a_{22} + a_{33} > 0$, then (see, for example, (Onishchenko, 1983), (Siouris, 1993, 2007) (Branets and Shmyglevsky, 1973), (Farrell, 2008):

$$\lambda_0 = \frac{1}{2}\sqrt{1+a_{11}+a_{22}+a_{33}}\,, \qquad \lambda_1 = \frac{a_{23}-a_{32}}{2\sqrt{1+a_{11}+a_{22}+a_{33}}}\,,$$

$$(2.7)\ \lambda_2 = \frac{a_{31}-a_{13}}{2\sqrt{1+a_{11}+a_{22}+a_{33}}}\,, \qquad \lambda_3 = \frac{a_{12}-a_{21}}{2\sqrt{1+a_{11}+a_{22}+a_{33}}}\,.$$

It should be noted that there are three more approaches to determine the elements of quaternion from rotation matrix, which are detailed in (Farrell, 2008), (Shuster, 1993).

The projections $\omega_1, \omega_2, \omega_3$ of body's angular velocity vector on BF axes through the Euler's angles from a classic mechanics could be presented as in (Lurie, 2002), equations (2.9.3):

$$\begin{aligned} \omega_1 &= \dot{\psi}\cdot\sin\vartheta\cdot\cos\,\varphi + \dot{\vartheta}\cdot\cos\,\varphi \\ \omega_2 &= \dot{\psi}\cdot\sin\vartheta\cdot\cos\,\varphi - \dot{\vartheta}\cdot\sin\varphi \\ \omega_3 &= \dot{\psi}\cdot\cos\vartheta + \dot{\varphi} \end{aligned} \qquad (2.8)$$

The projections $\omega_1, \omega_2, \omega_3$ of body's angular velocity vector on BF axes through the Euler's angles (roll, pitch, heading) are presented as in (Siouris, 2007), equations (2.20):

$$\begin{aligned} \omega_1 &= \dot{\phi} - \dot{\psi}\cdot\sin\theta \\ \omega_2 &= \dot{\psi}\cdot\cos\,\theta\cdot\sin\,\phi + \dot{\theta}\cdot\cos\,\phi \\ \omega_3 &= \dot{\psi}\cdot\cos\,\theta\cdot\cos\,\phi - \dot{\theta}\cdot\sin\,\phi \end{aligned} \qquad (2.9)$$

Given the rigid body position at the initial instant of time and measuring the projections of body's angular velocity vector $\boldsymbol{\omega} = [\omega_1, \omega_2, \omega_3]^T$ on BF axes, we can find the vector $\boldsymbol{\lambda} = [\lambda_0, \lambda_1, \lambda_2, \lambda_3]^T$ (quaternion) (Kantor and Solodovnikov, 1973) of the Rodriguez-Hamilton parameters as a result of integration of the kinematic equations:

$$\dot{\lambda} = \tfrac{1}{2}\Omega\lambda\,, \qquad (2.10)$$

$$\Omega = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix}$$

$$\|\lambda\|^2 = \lambda^T\lambda = 1\,.$$

Hereinafter, $\|\cdot\|$ denotes spectral matrix norm (Wittenburg, 1977) and the superscript T stands for transposition.

The conversion from the normalized quaternion to Euler angles (roll, pitch, and heading) can be written as follows (Titterton and Weston, 2004), (Siouris, 2007), (Grewal, Weill and Andrews, 2001), (Coopmans, Chao and Chen, 2009):

$$\begin{aligned}
\phi &= \arctan\left(\frac{2(q_2q_3+q_0q_1)}{q_0^2-q_1^2-q_2^2+q_3^2}\right);\\
\theta &= \arcsin\left(-2(q_1q_3-q_0q_2)\right);\\
\psi &= \arctan\left(\frac{2(q_1q_2+q_3q_0)}{q_0^2+q_1^2-q_2^2-q_3^2}\right).
\end{aligned} \tag{2.11}$$

The body (UAV) attitude with respect to the local-level topocentric NED frame can be described by means of rotation around axes roll (X), pitch (Y), and yaw (Z) of the body, beginning with the instant of time when these axes coincide with NED axes. As a result, we obtain the DCM $\boldsymbol{C}_b^n$, which can be used to find the Euler angles (roll, pitch, and yaw) as well as the corresponding elements of the quaternion (Titterton and Weston, 2004), (Siouris, 2007), (Grewal, Weill and Andrews, 2001):

$$\begin{aligned}
\mathbf{C}_{\mathrm{b}}^{\mathrm{n}} &= \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}\\
&= \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi+\sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi+\cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\phi\cos\psi+\sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi+\cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}
\end{aligned} \tag{2.12}$$

By examining matrix $\boldsymbol{C}_b^n$, the Euler angles (roll, pitch, heading) may be derived from it (Titterton and Weston, 2004), (Siouris, 2007):

$$\begin{aligned}
\phi &= \tan^{-1}\left(\frac{\boldsymbol{C}_{32}}{\boldsymbol{C}_{33}}\right);\\
\theta &= \tan^{-1}\left(\frac{-\boldsymbol{C}_{31}}{\sqrt{1-\boldsymbol{C}_{31}^2}}\right);\\
\psi &= \tan^{-1}\left(\frac{\boldsymbol{C}_{21}}{\boldsymbol{C}_{11}}\right),
\end{aligned} \tag{2.13}$$

where $\boldsymbol{C}_{ij}$, $1 \le (i,j) \le 3$ – are the (i,j)-th elements of matrix $\boldsymbol{C}_b^n$.

Concluding this part of item regarding rotational motion, it is necessary to note, that here we presented the basic relations for the classic Euler angles ψ, ϑ, φ (precession, nutation, and intrinsic rotation), which are used mostly in the analytical mechanics (Lurie, 2002). The same relations are presented for other interpretation of Euler angles, i.e. ones describing aircraft dynamics ϕ, θ, ψ (roll, pitch, and heading), which are more relevant in some specific areas of the SINS and UAV applications (Beard and McLain, 2012), (Lurie, 2002), (Onishchenko, 1983), (Coopmans, Chao and Chen, 2009), (Farrell, 2008), (Siouris, 1993, 2007). Actually, there is a difference in the Rodrigues-Hamilton parameters definition in terms of Euler angles. Here the relations (2.2) work for classic Euler angles, and relations (2.3) for the roll, pitch and heading (Euler) angles. For the mapping of body angular velocity to the Euler angles there are relations (2.8) for the first case and (2.9) for the second. All other relations for quaternions usage and processing, which are presented above and will be presented below, remain the same for both cases.

To describe translational motion (i.e. change of object position), we take Eqs. (30) from (Bronkhorst, 1978), as it was done in (Larin, 2001). It reflects the theorem about addition of accelerations in case of complex point's motion (the Coriolis theorem),

$$\begin{gathered}\frac{d\boldsymbol{V}}{dt} = \boldsymbol{a}_m + \bar{\boldsymbol{g}} - 2\Omega \times \boldsymbol{V}, \\ \tilde{\boldsymbol{a}} = \boldsymbol{a}_m - \bar{\boldsymbol{g}}, \quad \bar{\boldsymbol{g}} = \boldsymbol{g} - \Omega \times \Omega \times \boldsymbol{R},\end{gathered} \tag{2.14}$$

where $\tilde{\boldsymbol{a}}$ - specific force, $\boldsymbol{a}_m$ - accelerometers readouts, transformed to navigation frame, $\boldsymbol{g}$ is the gravity acceleration, $\bar{\boldsymbol{g}}$ is the result of subtraction of centripetal Earth's acceleration from gravity acceleration, $\boldsymbol{V}$ - ground speed, Ω is angular velocity of the Earth rotation, and $\boldsymbol{R}$ is position vector of a point in geocentric coordinate system.

2.3. Algorithm of Inertial Navigation Systems Including Rate Gyros and Accelerometers

2.3.1. Rotational Mechanization Algorithms

Thus, the described INS operation scheme involves integration of a system of differential equations. From the point of view of the realization of such INS, it is expedient to consider "sampling" of this system, i.e., to consider the case where sensors are readout not continuously but in regular time intervals Δt, i.e., with frequency $f = 1/\Delta t$. Respectively, the required navigation parameters (cosine matrix $A(\lambda)$, velocity v, and coordinates r) are calculated in time interval Δt. Since different "sampling" procedures can be used to find navigation parameters, we will dwell on each of them. We will begin with estimating quaternions at the discrete time instants $t_i, t_i - t_{i-1} = \Delta t$; $i = 1,2,3,\ldots k$ (Larin, 2001).

Let us suppose that quasi-coordinates (components of vector $\nabla\theta_i = \int_{t_i}^{t_i+\delta t} \omega dt$) are known on a small time interval Δt. We will use them to express $\delta\lambda(t_i)$, the solution of Eq. (2.10) on time interval Δt under the initial condition $\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T$ (or any other quaternion obtained as a result of initial alignment), i.e., calculate the quaternion corresponding to small rotational displacement of the rigid body in time Δt, and then determine the body’s orientation by sequential multiplication of the "elementary" quaternions $\delta\lambda(t_i)$

$$\lambda(t_i) = \lambda(t_{i-1})\delta\lambda(t_i)\,, \tag{2.15}$$

$$\delta\lambda(t_i) = \begin{bmatrix}\delta\lambda_0(t_i) & \delta\lambda_1(t_i) & \delta\lambda_2(t_i) & \delta\lambda_3(t_i)\end{bmatrix}^T .$$

In the matrix form, this procedure looks as follows:

$$\lambda(t_i) = \begin{bmatrix}\delta\lambda_0(t_i) & -\delta\lambda_1(t_i) & -\delta\lambda_2(t_i) & -\delta\lambda_3(t_i)\\ \delta\lambda_1(t_i) & \delta\lambda_0(t_i) & \delta\lambda_3(t_i) & -\delta\lambda_2(t_i)\\ \delta\lambda_2(t_i) & -\delta\lambda_3(t_i) & \delta\lambda_0(t_i) & \delta\lambda_1(t_i)\\ \delta\lambda_3(t_i) & \delta\lambda_2(t_i) & -\delta\lambda_1(t_i) & \delta\lambda_0(t_i)\end{bmatrix}\begin{bmatrix}\lambda_0(t_{i-1})\\ \lambda_1(t_{i-1})\\ \lambda_2(t_{i-1})\\ \lambda_3(t_{i-1})\end{bmatrix}. \tag{2.16}$$

In (Larin, 2001) quaternions $\delta\lambda(t_i)$ are expressed in terms of the vector of quasi-coordinates $\nabla\theta_i$, which ensure one quality of approximation or another depending on the expressions complexity. For example (Branets, and Shmyglevsky, 1973),

$$\delta\lambda^1 = \begin{bmatrix}1\\ 1/2\nabla\theta_i\end{bmatrix}, \qquad \delta\lambda^2 = \begin{bmatrix}1-1/8\|\nabla\theta_i\|^2\\ 1/2\nabla\theta_i\end{bmatrix}$$

$$\delta\lambda^3 = \begin{bmatrix}1-1/8\|\nabla\theta_i\|^2\\ 1/2(1-1/24\|\nabla\theta_i\|^2)\nabla\theta_i - 1/24(\nabla\theta_i \times \nabla\theta_{i-1})\end{bmatrix} \tag{2.17}$$

The first two quaternions $\delta\lambda^1$ and $\delta\lambda^2$ (corresponding to Euler's method and modified Euler's method) approximate the displacement of rigid body on time interval Δt as rotation about the axis collinear to vector $\nabla\theta_i$. The third-order approximation $(\delta\lambda^3)$ takes into account the non-commutativity of rotations.

Hereinafter, we will use the following approximation of quaternion $\delta\lambda(t_i)$ (relation (2.6) in (Larin, 2001)):

$$\delta\lambda(t_i) = \begin{bmatrix}1-\frac{1}{12}\|\nabla\theta_i\|^2\\ \frac{1}{2}\nabla\theta_i - \frac{1}{24}(\nabla\theta_i \times \nabla\theta_{i-1})\end{bmatrix}. \tag{2.18}$$

Likewise to (Larin, 2001), in order to calculate $\nabla\theta_i$ we can use quadratic spline approximation of the angular velocity vector $\omega(t)$. For example, if the values $\omega(t_{i-2}), \omega(t_{i-1}), \omega(t_i)$ are known, then

$$\nabla\theta_i = \frac{\Delta t}{12}\left(5 \cdot \omega(t_i) + 8 \cdot \omega(t_{i-1}) - \omega(t_{i-2})\right). \qquad (2.19)$$

Thus, for given RGs readouts we can use relations (2.15)–(2.19) to update the quaternion for rotational motion. Then we may find the direction cosine matrix (DCM) according to (2.5), and calculate the roll, pitch and heading values from the updated quaternion directly using (2.11), or from obtained DCM using (2.13) relation.

2.3.2. Translational Mechanization Algorithms

Obtained at the previous step parameters of orientation are used to transform accelerometer readouts by means of DCM. It should be noted that it is possible to do this operation using quaternion parameters directly, as it was done in (Kharchenko et.al, 2013). Then the value of the specific force $\tilde{\boldsymbol{a}}$ is calculated according to (2.14) relation.

The next step is the procedure of integration of Eq. (2.14), which will allow finding object's current coordinates and velocity.

As well as in (Bronkhorst, 1978), we consider the term $2\Omega \times V$ (Coriolis acceleration) as a small correction and do not take it into account in the example below. In this regard, if it is necessary to take the Coriolis acceleration into account, like in (Larin, 2001), when calculating it at the i th step, it is possible to use the value of velocity V corresponding to the instant of time t_{i-1}. This assumption allows replacing the procedure of integration of Eq. (2.14) with calculation of quadratures.

Thus, based on the readouts of accelerometers and DCM we can find $\tilde{a}$, the values of the right-hand side of Eq. (2.14). Given the values of $\tilde{a}$ at moments of times t_{i-2}, t_{i-1}, t_i , we can write the relations similar to (2.19) that define $v(t_i), r(t_i)$, i.e.,

$$v(t_i) = v(t_{i-1}) + \int_0^{\Delta t} \tilde{a}\,dt = v(t_{i-1}) + \left(5\tilde{a}(t_i) + 8\tilde{a}(t_{i-1}) - \tilde{a}(t_{i-2})\right) \cdot \frac{\Delta t}{12}; \qquad (2.20)$$

$$\begin{aligned} r(t_i) &= r(t_{i-1}) + v(t_{i-1})\Delta t + \int_0^{\Delta t} v(t_i)\,dt = \\ &= r(t_{i-1}) + \left(3\tilde{a}(t_i) + 10\tilde{a}(t_{i-1}) - \tilde{a}(t_{i-2})\right) \cdot \frac{\Delta t^2}{24} + \Delta t \cdot v(t_{i-1}). \end{aligned} \qquad (2.21)$$

Summarizing, we note that relations (2.5), (2.15)–(2.19), (2.11), (2.13), (2.14), (2.20) – (2.21) determine the INS operation algorithm, i.e., allow using the results of RGs and accelerometers readings at instants of time t_i to estimate navigation parameters at time t_i. This, in turn, allows using traditional GPS and INS integration algorithms to correct the results of operation of the INS under study.

2.3.3. Estimating the Integration Accuracy of Kinematic Equations

Following (Avraamenko and Larin, 1983), (Larin and Naumenko, 1983), let us consider in more detail the procedure of numerical integration of Eq. (2.10) (constructing the quaternion $\delta\lambda$ that appears in formula (2.15)). In the problem about motion of a rigid body relative to a fixed point, the procedure of numerical integration of Eq. (2.10) is often based on the Picard method, (see, for example, (Branets and Shmyglevsky, 1973), Sec. 4.1). Namely, as we have already mentioned, having approximated the solution of Eq. (2.10) with some accuracy on a small time interval δt, we find the rigid body position by successive multiplication of such elementary quaternions. In case of such technique of numerical integration of kinematic equations, quaternions are usually considered as four-dimensional vectors, which predetermines approximation accuracy criteria. However, in the problem under consideration this vector contains information about axis direction and value of rotation angle; hence, when approximating an elementary quaternion, it is necessary to minimize losses of such information. To consider these problems in more detail, we will use an elementary problem example of integration of Eq. (2.10) in case of a body rotating about a fixed axis (Larin and Naumenko, 1983). This example is of interest because as it was mentioned in (Branets and Shmyglevsky, 1973), (Chelnokov, 1977), the hypothesis of such motion is often used in constructing an elementary quaternion (2.17).

Example 2.1

The purpose of this example consists of obtaining analytical expressions for estimation of accuracy of the aforementioned methods of rotational mechanization in a case of some given specific rigid body rotation. These expressions will be used for further numerical examples. Thus, let the body rotate about axis x, i.e., $\omega_1 = \omega_0$, and $\omega_2 = \omega_3 = 0$. Then in time δt the body will turn about axis x by angle $\chi = \int \omega_0 dt$ (the integration time is from 0 to δt). The solution of

Eq. (2.10) (the quaternion corresponding to body rotation in time δt) for the initial condition $\lambda(0)=\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T$ has the form

$$\lambda=\begin{bmatrix}\cos{}^1\!/\!_2\,\chi & \sin{}^1\!/\!_2\,\chi & 0 & 0\end{bmatrix}^T \tag{2.22}$$

The value χ is usually supposed to be small and $\sin{}^1\!/\!_2\,\chi$ and $\cos{}^1\!/\!_2\,\chi$ that appear in quaternion (2.22) are approximated by a certain number of terms of Taylor series, i.e., not quaternion (2.22) but quaternions (2.17) are taken as elementary ones,

$$\lambda^1=\begin{bmatrix}1 & {}^1\!/\!_2\,\chi & 0 & 0\end{bmatrix}^T,\ \lambda^2=\begin{bmatrix}1-{}^1\!/\!_8\,\chi^2 & {}^1\!/\!_2\,\chi & 0 & 0\end{bmatrix}^T, \tag{2.23}$$

and are called first-order, second-order, etc. approximation (Branets and Shmyglevsky, 1973). It is natural to consider that along with Taylor approximation, other approximations can be used for this purpose, in particular, it is possible to use the Pade approximation of function $\exp\left({}^{i\chi}\!/\!_2\right)$. For example, the first-order Pade approximation leads to the expression

$$\lambda_p^1=\frac{1}{1+{}^1\!/\!_{16}\,\chi^2}\begin{bmatrix}1-{}^1\!/\!_{16}\,\chi^2 & {}^1\!/\!_2\,\chi & 0 & 0\end{bmatrix}^T \tag{2.24}$$

When we call (2.23) for small χ an approximate value of (2.22) of first, second, etc. order, we mean $\left\|\lambda-\lambda^1\right\|=O(\chi),\left\|\lambda-\lambda^2\right\|=O(\chi^2)$, i.e., approximation accuracy of vector (2.22), (2.23) or (2.24) in the sense of Euclidean norm. Since in the problem under study vector (2.22) is a quaternion, it is necessary to clarify how vectors (2.23) after an appropriate normalization (in case of Pade approximation, the norm of λ_p^1 is equal to one) approximate (2.22) as a quaternion, i.e., to find errors in axis orientation and rotation angle. Note that in this case all the approximating quaternions do not contain errors in the orientation of rotation axis (it is clear from their structure that the rotation is about the axis). Let us find the angles φ_1,φ_2 and φ_p of rotation about axis x, corresponding to λ^1,λ^2 and λ_p^1 (according to (2.22), the true value is χ).

It is obvious that in case of rotation about axis x ($\lambda_2=\lambda_3=0$), we can find the rotation angle φ as

$$\varphi=2arctg\left(\lambda_1/\lambda_0\right)\ , \tag{2.25}$$

hence

$$\varphi_1==2arctg\left({}^1\!/\!_2\,\chi\right)=\chi-{}^1\!/\!_2\,\chi^3+\ldots$$

$$\varphi_2 == 2arctg\left[\chi/\left(2\left(1-\tfrac{1}{8}\chi^2\right)\right)\right] = \chi + \tfrac{1}{24}\chi^3 + ...$$

$$\varphi_p == 2arctg\left[\chi/\left(2\left(1-\tfrac{1}{16}\chi^2\right)\right)\right] = \chi - \tfrac{1}{48}\chi^3 + ...$$

These formulas show that, generally speaking, increase in the approximation accuracy of quaternion (2.22) considered as a vector does not respectively increase the approximation accuracy of the rotation angle (φ_2 and φ_p differ from χ by the value of the same order as

φ_1). Thus, in this example, when constructing an elementary quaternion, it is necessary to minimize the residue $\varphi - \chi$ (φ is defined by expression (2.25)) instead of increasing the approximation accuracy of vector λ . In line with these reasons, we can use vector $\lambda_m = \left[1-\tfrac{1}{12}\chi^2 \quad \tfrac{1}{2}\chi \quad 0 \quad 0\right]^T$ as an approximation of elementary quaternion. Its structure does not differ from λ^2, but it better approximates the rotation angle

$$\varphi_m == 2arctg\left[\chi/\left(2\left(1-\tfrac{1}{12}\chi^2\right)\right)\right] = \chi - \tfrac{1}{120}\chi^5 + ...$$

To illustrate the efficiency of different approximations of quaternion (2.22), we will use the following numerical example.

Example 2.2

A body rotating with constant angular velocity about axis x makes a complete revolution in time T, i.e., $\lambda(0) = \left[1 \quad 0 \quad 0 \quad 0\right]^T$ and $\lambda(T) = \left[\cos\pi \quad \sin\pi \quad 0 \quad 0\right]^T = \left[-1 \quad 0 \quad 0 \quad 0\right]^T$. Let interval T be divided into n equal time intervals, i.e. $\chi = \Delta\omega_0 = 2\pi/n$. In this example, procedure (2.16) of the multiplication of elementary quaternions has the form

$$\begin{bmatrix}\lambda_0(i\Delta)\\ \lambda_1(i\Delta)\end{bmatrix} = \begin{bmatrix}1-q\chi^2 & \tfrac{1}{2}\chi\\ \tfrac{1}{2}\chi & 1-q\chi^2\end{bmatrix} \cdot \begin{bmatrix}\lambda_0((i-1)\Delta)\\ \lambda_1((i-1)\Delta)\end{bmatrix}, \qquad (2.26)$$

since as $\lambda_2(i\Delta) = \lambda_3(i\Delta) = 0$ (i=1,2,...,n).

In (2.26), parameter takes the following values: q=0, $\tfrac{1}{8}$, $\tfrac{1}{16}$ and $\tfrac{1}{12}$ in a case of the quaternion approximation (2.22) by vectors $\lambda^1, \lambda^2, \lambda^1_p$ and λ_m respectively. We will estimate the approximation accuracy by the formula $\Delta_n(\lambda) = \lambda_1(n\Delta)10^2/\left[\lambda_0(n\Delta)\right]$ (for precise integration $\Delta_n(\lambda) = 0$). Table 2.1 shows the results of numerical integration. Note that $\left\|\lambda - \lambda_m\right\| = O(\chi)$, i.e., vector λ_m approximates vector λ worse than λ^2 does. However, as follows from Table 2.1, angle φ_m approximates angle χ much better than φ_2 does. Thus, if we drop the normalization requirement $\|\lambda(t_i)\| = 1$ at each integration step (for example, only carry out normalization at the end of the integration process), we can increase the approximation accuracy or simplify the expression for $\delta\lambda$. In this regard, the following modification of quaternion $\delta\lambda^2$ was proposed in (Larin and Naumenko, 1983):

$$\delta\lambda_m^2 = \begin{bmatrix} 1 - 1/12\|\nabla\theta_i\|^2 \\ 1/2\nabla\theta_i \end{bmatrix}.$$

In (Larin and Naumenko, 1982) it was proposed the following approximation for quaternion $\delta\lambda^3$:

$$\delta\lambda_m^3 = \begin{bmatrix} 1 - 1/12\|\nabla\theta_i\|^2 \\ 1/2\nabla\theta_i - 1/24(\nabla\theta_i \times \nabla\theta_{i-1}) \end{bmatrix}. \qquad (2.27)$$

There are general relations that allow estimating the integration accuracy of kinematic equations (2.6) (see, for example, (Bodanskiy and Furman, 1970)). However, these relations are rather cumbersome. For the case of conical motion, it became possible in (Avraamenko and Larin, 1983) to obtain simple formulas to estimate average drift rate (the velocity of accumulation of integration error) when one integration algorithm or another is used. Moreover, the examples considered in (Avraamenko and Larin, 1983) give an idea about the accuracy of integration of Eq. (2.6) in case of using different algorithms to construct $\delta\lambda$.

Table 2.1.

n	$\Delta_n(\lambda^1)$	$\Delta_n(\lambda^2)$	$\Delta_n(\lambda_p^1)$	$\Delta_n(\lambda_m)$
5	35,0	-18,1	9,79	0,986
6	25,7	-13,0	6,91	0,491
7	19,1	-9,92	5,12	0,270
8	14,9	-7,72	3,95	0,161
9	12,0	-6,14	3,13	0,102
10	9,79	-5,01	2,55	0,0664
11	8,16	-4,17	2,11	0,0455

12	6,91	-3,51	1,78	0,0323

Example 2.3

A body undergoes precession motion with parameters $\phi = \pi/2 - 1.25t$, $\psi = 3.75t$, $tg\,\theta = 4/3$, i.e., $\omega = \|3\cos(1.25t); 3\sin(1.25t); 1\|'$ The integration time is $T = 1.6\pi$, i.e., $\lambda(T) = \lambda(0)$. For the frame that should be turned according to quaternion $\lambda(t)$, we will take the system coinciding with moving frame at the initial instant of time, i.e., $\lambda(0) = \|1, 0, 0, 0\|^T$.

Table 2.2 shows the results of integration demonstrating normalized values of quaternion $\lambda(T)$ when elementary quaternions $\delta\lambda^1, \delta\lambda^2, \delta\lambda^2_m, \delta\lambda^3$ $(\delta\lambda = T/50)$ are used (Larin and Naumenko, 1982).

Table 2.2

$\lambda(T)$	$\delta\lambda^1$	$\delta\lambda^2$	$\delta\lambda^2_m$	$\delta\lambda^3$
$\lambda_0(T)$	0.997	1.00	1.00	1.00
$\lambda_1(T)$	$-5.64 \cdot 10^{-2}$	$1.89 \cdot 10^{-2}$	$-6.47 \cdot 10^{-3}$	$2.87 \cdot 10^{-4}$
$\lambda_2(T)$	$1.93 \cdot 10^{-6}$	$3.43 \cdot 10^{-6}$	$2.79 \cdot 10^{-6}$	$8.08 \cdot 10^{-5}$
$\lambda_3(T)$	$-4.22 \cdot 10^{-2}$	$1.41 \cdot 10^{-2}$	$-4.84 \cdot 10^{-3}$	$-3.59 \cdot 10^{-4}$

As it is seen from Table 2.2, the vectors of orientation errors (as the coordinates of these vectors in the moving frame whose rotation at instant of time $t = T$ is defined by Euler's angles $\varphi = \frac{\pi}{2}, \psi = 0, \tan\theta = 4/3$, we can take $2\lambda_1(T), 2\lambda_2(T), 2\lambda_3(T)$ are almost collinear to axis Z of the fixed coordinate system XYZ and moreover are directed in opposite directions. Thus, if we consider the procedure of integration of Eq. (1.1) with "elementary" quaternion

$$\delta\lambda_q^2 = \left\| \begin{matrix} 1 - q\|\nabla\theta \quad \|^2 \\ {}^1\!/_2\, \nabla\theta_i \end{matrix} \right\|, \tag{2.28}$$

which coincides with $\delta\lambda^2$ for $q = 1/8$ and with $\delta\lambda_m^2$ for $q = 1/12$, we may expect that for the appropriate value of parameter q between $1/8$ and $1/12$, integration error will substantially decrease. The numerical optimization procedure ($\min(\delta\lambda^1, \delta\lambda^2, \delta\lambda^2_m, \delta\lambda^3)$) yields the optimal value $q = 0.0939$ and, respectively $\lambda(T) = \left\| 1,\ -6.2\cdot 10^{-5},\ 9\cdot 10^{-13},\ -4.6\cdot 10^{-5} \right\|^T$. Comparing this result with the last column in Table 2.2, we conclude that for such value of q and for $t = T$ the errors of the algorithm under study are commensurable with errors of the algorithm that takes into account the non-commutativity of rotations ($\delta\lambda^3$). Let us explain this effect. The difference of displacements of the rigid body in time δt, defined by the quaternions $\delta\lambda_m^2$ and $\delta\lambda_m^3$ (the approximation error), is characterized by the vector of small rotation $\varepsilon_i = -{}^1\!/_{12}\left(\nabla\theta_i^* \times \nabla\theta_{i-1}^*\right)$ since

$$\delta\lambda_m^3 \circ (\delta\lambda_m^2)' = \left\| \begin{matrix} 1 \\ -{}^1\!/_{24}(\nabla\theta_i^* \times \nabla\theta_{i-1}^*) \end{matrix} \right\|^3,$$

The task is to analyze the process of summation of these vectors. To this end, we can accept the following approximation: $\nabla\theta_i^* = \boldsymbol{\omega}(t_i)dt$. Hence, when subscript i varies, vector ε_i (as well as vector $\boldsymbol{\omega}(t_i)$) will rotate about axis Z of the fixed coordinate system with angular velocity $\dot\psi$ (the term "rotate" implies that as subscript i increases by one, the vector ε_i turns about axis Z by angle $\dot\psi\delta t$). Thus, it is convenient to investigate the process of summation of vectors ε_i (the process of accumulation of integration errors) by decomposing vector ε_i into two orthogonal components, one lying in plane XY of fixed coordinate system, and second being directed along axis Z. The sums with respect i of each of the marked component behave differently. For example, absolute value of the vector sum of the component errors ε_i lying in plane XY is a periodic function of i whose mean value over the time interval is zero. Projections of vector ε_i onto axis Z will be added, i.e., will be the main cause of systematic drift. This conclusion agrees well with the note made in the analysis of the results in the example of numerical integration that error vectors, when quaternions $\delta\lambda^2$ and $\delta\lambda_m^2$ are used, are directed along axis Z. Thereupon, the above possibility of "adjusting" parameter q in quaternion $\delta\lambda_q^2$ can be explained by the effect of compensation of the projection of error (vector of small rotation) onto axis Z. Indeed, with accuracy within third-order small values we have:

$$\delta\lambda_q^2 = \left(1 - \tfrac{\mu}{12}\|\nabla\theta_i^*\|^2\right) \left\| \begin{matrix} 1 - \frac{1}{12}\|\nabla\theta_i^*\|^2 \\ \frac{1}{2}\nabla\theta_i^*\left(1 + \frac{\mu}{12}\|\nabla\theta_i^*\|^2\right) \end{matrix} \right\|^3, \mu = 12q - 1. \tag{2.29}$$

Comparing this expression for $\delta\lambda_q^2$ with $\delta\lambda_m^2$ (we can disregard factor $\left(1 - \frac{1}{12}\|\nabla\theta_i^*\|^2\right)$ in the last expression for $\delta\lambda_q^2$; its effect is eliminated by the subsequent normalization (Larin and Naumenko, 1982)). We conclude that when using quaternion $\delta\lambda_q^2$, we can obtain average drift corresponding to the procedure of integration with quaternion $\delta\lambda^3$ if the appropriate choice of coefficient μ ensures the equality of the last components of these quaternions. We can obtain similar relation for finding coefficients μ or q from the condition that the last component of quaternion $\delta\lambda_m^3 \circ (\delta\lambda_m^2)'$ (projection of the error onto axis Z) is zero. After transformations, this condition becomes $\mu\|\boldsymbol{\omega}\|^2\omega_z = (\|\boldsymbol{\omega}\|^2 - \omega_z)\dot{\psi}$, or:

$$\mu(\dot{\varphi}\cos\theta + \dot{\psi})\left(\dot{\varphi}^2 + \dot{\psi}^2 + 2\dot{\varphi}\dot{\psi}\cos\theta\right) = \dot{\psi}\,\dot{\varphi}^2\sin^2\theta, \tag{2.30}$$

where $\omega_z = \dot{\psi} + \dot{\varphi}\cos\theta$ is projection of vector $\boldsymbol{\omega}$ onto axis Z. If not Euler's angles are specified but projections of the vector of angular velocity onto axes of moving coordinate system (for example, $\omega_1 = a\cos ct$, $\omega_2 = a\sin ct$, $\omega_3 = b$)), then taking into account that $\dot{\varphi} = -c,\ b = \dot{\psi}\cos\theta - c,\ \dot{\psi}^2 = a^2 + (b+c)^2, \dot{\psi} = {}^{a}/_{\sin\theta},$ in this case as well, we can rearrange relation (2.30) as $a^2c^2 = \mu(a^2+b^2)(a^2+b^2+bc)$, i.e., the optimal value of q can be expressed in terms of motion parameters as follows:

$$q^* = \frac{1}{12}\left[1 + \frac{\dot{\psi}\dot{\varphi}^2\sin^2\theta}{(\dot{\varphi}\cos\theta + \dot{\varphi}\,)\left(\dot{\varphi}^2 + \dot{\psi}^2 + 2\dot{\varphi}\dot{\psi}\cos\theta\right)}\right]$$

$$\mathrm{q}^* = \frac{1}{12}\left[1 + \frac{\mathrm{a}^2\mathrm{c}^2}{(\mathrm{a}^2+\mathrm{b}^2)(\mathrm{a}^2+\mathrm{b}^2+\mathrm{bc})}\right] \tag{2.31}$$

For the initial data of the considered numerical example ($a = 3,\ b = 1,\ c = 1.25$), the value of q^* calculated by formula (2.31) turns out to be equal to $9.375 \cdot 10^{-2}$, which coincides well with the above value of $q$obtained by means of numerical optimization.

Similar reasoning allows us to obtain the approximate formula for the drift rate (the velocity of accumulation of integration error) when elementary quaternion $\delta\lambda_q^2$ is used for the values of q that are quite different from q^* (formula (2.31)). Since for $q = q^*$ algorithm (2.28) ensures much higher accuracy than for other values of q, when deriving the approximate relation we will consider the error for $q = q^*$ equal to zero. Moreover, like above, we assume that average drift is defined by the projection of small rotation onto axis Z, which characterizes integration error. According to (2.29), this error is defined by the projection of the vector of small rotation onto the same axis:

$$\nabla\theta_i^*\left(1+\frac{12q-1}{12}\|\nabla\theta_i^*\|^2\right)-\nabla\theta_i^*\left(1+\frac{12q^*-1}{12}\|\nabla\theta_i^*\|^2\right).$$

Thus, in the first approximation, we can assume relative integration error to be equal to $(q-q^*)\|\nabla\theta_i^*\|^2$. Finally, taking into account the above assumptions, we can find average drift rate U for algorithm (2.28) as follows:

$$U=|(q-q^*)\omega_z|\frac{\|\boldsymbol{\omega}\|^2}{f^2} \tag{2.32}$$

where f is the sampling frequency ($\delta t = {}^1\!/\!_f$).

Table 2.3 gives some idea about the accuracy of this formula. It presents the values of drift rate (dimension here and in Table 2.4 is angular second per second) for two types of motion: the upper number is calculated by formula (2.32), the lower one as a result of computer modeling of the integration process (Larin and Naumenko, 1982, 1983). The results are obtained for two types of motion. In the first case (I): $\boldsymbol{\omega}'=\|\pi;\ \ 0;\ \ 0\|,\ f=10\text{ Hz}$, i.e., $\delta t=0.1\ s$. In the second case (II), motion parameters are the same as in the example considered above, i.e., $\boldsymbol{\omega}'=\|3\cos 1.25t;\ \ 3\sin 1.25t; 1\|,\ f=50/1.6\pi$ Hz.

Table 2.3

Motion type	$\delta\lambda^1$	$\delta\lambda^2$	$\delta\lambda_m^2$
I	5329 5130	2665 2626	 26
II	5863 5797	1954 1938	651 660

Table 2.4

γ , deg	$\delta\lambda^1$	$\delta\lambda^2$	$\delta\lambda_m^2$
0	5863 5766	1954 1927	651 660
30	5077 4998	1692 1673	564 573
45	4145 4084	1381 1368	460 468
60	2931 2890	979 970	326 331
90	0 10	0 2	0 0.8

The assumptions made while deriving formula (2.32) make possible to obtain similar estimate of average drift rate when the process of integration of kinematic equations is accompanied at each step by correction of integration results by measurement of projections onto the axes related to the body of any vector m whose position is known in a fixed coordinate system (Larin and Naumenko, 1983). Considering the results of measurement of projection of this vector accurate, we can suppose in functional (3.3) from (Larin and Naumenko, 1983) $\alpha = 0, \beta_2 = 1$. In this case, the effect of correction of the error of integration results (the vector of small turn $\varepsilon_i = -\frac{1}{12}(\nabla\theta_i^* \times \nabla\theta_{i-1}^*)$) will be elimination of the component of vector ε_i that is orthogonal to vector m (according to (3.13) from (Larin and Naumenko, 1983) for $\alpha = 0,\ I_k = 0$), i.e., projection of vector ε_i onto the direction of vector m. Thus, the resultant average drift rate will be defined by the sum of projections of vector ε_i onto the direction of vector m.

Let us decompose vector ε_i into two components: one directed along axis Z and the other being orthogonal to this axis. Since the second component uniformly rotates in plane XY with angular velocity $\dot{\psi}$, the sum of its projections onto the direction of vector m, when the vector is fixed in the fixed frame, will be a periodic function of time with zero contribution to the average drift rate. As to the first component, as we already mentioned when deriving formula (2.32), the sum of projections of vector ε_i onto axis Z will be an increasing function of time and will define average drift rate. Hence, if the position of vector m is fixed in the fixed coordinate system and γ is the angle between axis Z and vector m, then average drift rate in the integration of kinematic equations according to algorithm (2.28) and correction of the integration results according to (Larin and Naumenko, 1983) is as follows:

$$U_m = U\cos\gamma\,. \tag{2.33}$$

The value U appearing in (2.33) is defined by formula (2.32). It is possible to ensure the same average drift rate by making correction not at each integration step but once during the change of angle ψ by 2π.

The efficiency of formula (2.33) is characterized by the data presented in Table 2.4 for the motion of a rigid body with the parameters $\varphi = \pi/2 - 1.25t,\ \ \psi = 3.75t,\ \ \tan\theta = {}^4\!/_3$, i.e., $\boldsymbol{\omega}' = \|3\cos 1.25t;\ 3\sin 1.25t; 1\|$. For each value of γ, the upper number is calculated by formula (2.33), the lower one is obtained by modeling the integration process. Integration time is $T = 1.6\pi$, sampling frequency is $f = 50/1.6\pi$ Hz. The correction was made at the end of integration interval. Table 2.3 gives the value of average drift rate for integration without correction.

According to the report (Bogdanov et.al, 2011), creating a navigation system that can efficiently operate with very fast (about 5000 °/sec and above) rotating objects where a "classical" SINS, constructed using AVSs and accelerometers, is inefficient represents a very important problem. An alternative here is SINS constructed with accelerometers only. We will consider it in the next chapter.

2.4. Algorithms of the Accelerometer-Based INS (without Rate Gyros Usage)

As well as in (Larin and Tunik, 2010 a), we will consider two problems of determining kinematic parameters of motion of a rigid body. In the first one, the results of measurement of the velocities of three points of the body are used to find the vectors of angular velocity and velocity of the point accepted as the origin of body-fixed moving coordinate system. In the second problem, the results of observation of the acceleration of three points of the body and known angular velocity of the body are used to calculate angular accelerations and accelerations of the origin of moving frame.

Below, based on the obtained results, we will analyze INS model that does not use angular velocity sensors. Such INS scheme can be efficient, for example, in case of motion of an object with large angular velocity when, as is mentioned in (Bogdanov et.al, 2011), using RGs may be problematic.

2.4.1 Finding the Velocities

Several authors considered the problem of finding angular velocity of a rigid body and velocity of its one point from the results of observation of the velocities of three points of the body (see (Fenton and Willgoss, 1990), (Laub and Shiflett, 1983) and references therein). The problem statement looks like follows (see Fig. 2.4).

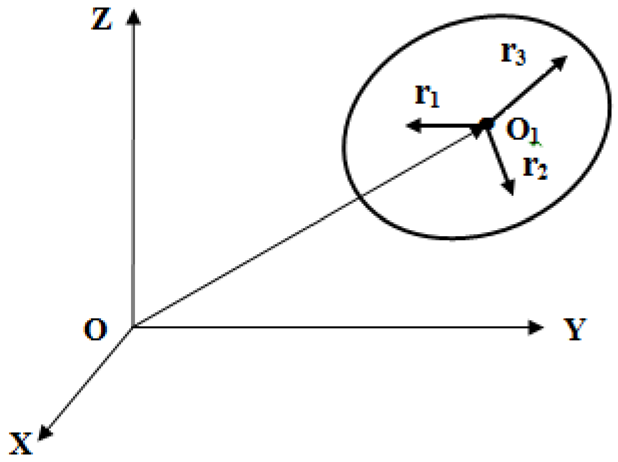


Fig. 2.4. Installation of velocity sensors

Three vectors r_1, r_2, r_3 determine points of a rigid body at which linear velocity is measured. From the results of these measurements it is necessary to find the vectors of body's angular velocity ($\omega=[\omega_1\, \omega_2\, \omega_3]^T$) and of linear velocity ($v_0=[v_1\, v_2\, v_3]^T$) of the origin of body-fixed coordinate system O_1. Taking into account the well-known relation (see, for example (2.7.8) in (Bogdanov et.al, 2011), (2) in (Avraamenko and Larin, 1983)) for the velocity of a point of rigid body specified by vector r:

$$v = v_0 + \omega \times r\,, \tag{2.34}$$

we can write the following linear relations (Eqs. (6) in (Krasovskiy, 1993), (4) in (Avraamenko and Larin, 1983)), which associate the required components of vectors ω, v and the results of observation of the velocities of selected points

$$V = \Omega P + v_0 h^T\,, \tag{2.35}$$

where $\Omega = \omega\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, P = [r_1\ r_2\ r_3], h = [1\ \ 1\ \ 1]^T$, and V is the matrix whose columns are velocity vectors of points defined by vectors r_1, r_2, r_3 .

Let $\beta_1, \beta_2, \beta_3$ and $\gamma_1, \gamma_2, \gamma_3$ be columns of matrices P^T and V^T: $P^T = [\beta_1, \beta_2, \beta_3]$, $V^T = [\gamma_1, \gamma_2, \gamma_3]$.

In this case, we can write relation (2.35) as a system of linear equations with respect to ω, v_0:

$$A_v x = B$$

$$x = \begin{bmatrix} \omega \\ v_0 \end{bmatrix},\ A_v = \begin{bmatrix} o & \beta_3 & \beta_2 & h & o & o \\ -\beta_3 & o & \beta_1 & o & h & o \\ \beta_2 & -\beta_1 & o & o & o & h \end{bmatrix},\ \ B = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}, \tag{2.36}$$

where o is 3×1 zero matrix.

Since velocity measurements involve errors, we will rearrange (2.36) as

$$A_v x = B_0 + n_v\ , \tag{2.37}$$

where n_v are measurement errors and vector B_0 consists of exact values of velocities of the considered points.

2.4.2. Finding the Accelerations

By analogy with the problem described above, we can consider finding body's angular acceleration and acceleration of its one point based on the results of observation of the acceleration of three points of the body. We will formulate the problem as follows. Let three vectors ρ_1, ρ_2, ρ_3 define rigid body's points at each of which three accelerometers are located that allow measuring components of the acceleration vector of that point. It is necessary to use the results of these measurements and the value of angular rate vector ($\omega=[\omega_1\,\omega_2\,\omega_3]^T$) to find angular acceleration$\left(\varepsilon=[\varepsilon_1\ \varepsilon_2\ \varepsilon_3]^T=\dfrac{d\omega}{dt}\right)$ and linear acceleration ($w_0=[w_1\,w_2\,w_3]^T$) of the origin of the body-fixed coordinate system. As applied to the problem under study, an analog of relation (2.34) is relation ((2.17.9) (Bogdanov et.al, 2011)), which specifies the acceleration of rigid body's point defined by vector (w):

$$\mathrm{w}=\mathrm{w}_0+\varepsilon\times\rho+\omega\times(\omega\times\rho). \tag{2.38}$$

Denoting $U=[w_1\,w_2\,w_3]$, w_i are acceleration vectors of points defined by ρ_i $(i=1,2,3)$. Then based on (2.38), we can write an analog of relation (2.35)

$$U=\Omega^2P_w+EP_w+w_0h^T. \tag{2.39}$$

Here $\mathrm{P_w}=[\rho_1\,\rho_2\,\rho_3]$, $\mathrm{E}=\varepsilon\times=\begin{bmatrix}0 & -\varepsilon_3 & \varepsilon_2\\ \varepsilon_3 & 0 & -\varepsilon_1\\ -\varepsilon_2 & \varepsilon_1 & 0\end{bmatrix}$ and matrices Ω,h are similar to those appearing in (2.35).

As well as (2.35), we can represent relation (2.39) as a system of linear equations with respect to ε,w_0. Let $\alpha_1,\alpha_2,\alpha_3$; $\delta_1,\delta_2,\delta_3$; $\sigma_1,\sigma_2,\sigma_3$ be columns of matrices $\mathrm{U^T},\mathrm{P_w^T},(\Omega^2\mathrm{P_w})^\mathrm{T}$, i.e., $\mathrm{U^T}=[\alpha_1\,\alpha_2\,\alpha_3]$, $\mathrm{P_w^T}=[\delta_1\,\delta_2\,\delta_3]$, $(\Omega^2\mathrm{P_w})^\mathrm{T}=[\sigma_1\,\sigma_2\,\sigma_3]$.

Then we can write relation (2.39) similarly to (2.36), namely:

$$A_wx=B_\omega+B_w,$$

$$x=\begin{bmatrix}\varepsilon\\ w_0\end{bmatrix},\quad A_w=\begin{bmatrix}o & \delta_3 & \delta_2 & h & o & o\\ -\delta_3 & o & \delta_1 & o & h & o\\ \delta_2 & -\delta_1 & o & o & o & h\end{bmatrix},\quad B_\omega=\begin{bmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\end{bmatrix},\quad B_w=\begin{bmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{bmatrix} \tag{2.40}$$

o is 3×1 zero matrix, as well as in (2.36).

Like in case of relation (2.36), we suppose that readouts of the accelerometers are corrupted with errors. So we rearrange (2.40) as

$$\mathrm{A_wx=B_\omega+B_{wo}+n_w} \tag{2.41}$$

where $\mathrm{n_w}$ are measurement errors and components are formed by exact values of accelerations.

2.4.3. Inertial Navigation System

Taking into account $w_0 = \frac{dv_0}{dt}$, $\varepsilon = \frac{d\omega}{dt}$, we can consider system (2.41) as a system of nonlinear differential equations with respect to ω. In other words, considering signals of accelerometers (B_w) as known external inputs, under given initial conditions we can find $\omega(t)$ and $v_0(t)$ by integrating (2.41). Thus, the approach under study allows us to obtain information about angular velocity of the object without using angular velocity sensors. However, in this case when creating an inertial navigation system, it is necessary to take into account the following circumstances.

The vector X appearing in (2.41) is specified in a moving frame. Since we are investigating object's position in the inertial frame, it is expedient to map the second component of vector x (vector w_0) into the inertial coordinate system followed by integration in this system. Then it is possible to find the velocity and coordinates of object's point O_1 (see Fig.1) in the moving frame. The point O_1 is assumed to be the origin of the moving coordinate system. As to the first part of vector X (vector ω), it should be used to find current body's orientation, which can be defined both by the Rodriguez-Hamilton parameters (2.2) and by the DCM (2.4) (they are related by (2.5) and (2.7)). In our case, it is convenient to find the Rodriguez-Hamilton parameters by integrating Eq. (2.10) where components of vector ω can be found from integration of Eq. (2.41). Then the value of matrix A , which is used to map into the inertial coordinate system, can be found by (2.5). This matrix allows us to map vector w_0 into the inertial coordinate system and, as we have mentioned above, use integration to find current object's velocity and coordinates. Thus, realization of this type of inertial system includes the following:

- finding ω by integrating three differential equations (the first three relations in (2.41));
- finding the quaternion λ, which defines according to (2.5) the DCM A (which allows mapping the acceleration vector w_0 into the inertial coordinate system) by integrating system (2.10) (four equations);
- finding object's velocity and coordinates by integrating six equations.

In other words, it is necessary to integrate a system of differential equations of the 13th order. The initial conditions for this system are the values of the following quantities at the initial instant of time: object's position ($\mathrm{r}_0 = [\mathrm{x}_0\ \mathrm{y}_0\ \mathrm{z}_0]^{\mathrm{T}}$), initial orientation (quaternion $\bar{\lambda}$) or respective DCM $\mathrm{A}(\bar{\lambda})$, object's linear velocity ($\bar{\mathrm{v}}_0 = [\mathrm{v}_{\mathrm{x}0}\ \mathrm{v}_{\mathrm{y}0}\ \mathrm{v}_{\mathrm{z}0}]^{\mathrm{T}}$), and angular velocity vector ($\omega_0 = \lfloor \omega_{\mathrm{x}0}\ \omega_{\mathrm{y}0}\ \omega_{\mathrm{z}0} \rfloor$).

Note that we can find $\bar{\mathrm{v}}_0$, ω_0 from GPS measurement of the velocity of three points of the object by the algorithm described in Sec. 3.1.

Since the size of matrix A_{w} in (2.41) is equal 9×6, we can eliminate three rows in system (2.41) (not to consider the readings of three accelerometers). In other words, we can create an inertial navigation system using six accelerometers only (see Example I).

Under considerable operation time, such inertial system using rather simple and cheap hardware may not ensure an adequate accuracy of navigation parameters. In such situations, it is expedient to adjust the results of operation of such system by means of GNSS (Phillips and Schmidt, 1996), (Greenspan, 1996), (Schmidt, 2008, 2011).

Example 2.4

To illustrate the above algorithms, we will consider the following navigation problem. Let the navigation frame Oxyz be related to the Earth surface. In this frame, the object (to which system $\mathrm{Ox'y'z'}$ is related) circles in the plane xy with the velocity $\bar{v} = 30\, m/s$ and period $T = 60\ s$. During the motion, its orientation (frame $\mathrm{Ox'y'z'}$) is defined by the following time dependence of the Euler angles: $\psi = \dot{\psi}t,\ \dot{\psi} = \dfrac{2\pi}{T},\ \varphi = 0,\ \vartheta = 0$. According to (2.8), projections of the angular velocity onto axes of the moving coordinate system are as follows: $\omega_1 = \omega_2 = 0,\ \omega_3 = \dot{\psi}$. According to (2.2), initial object's orientation is specified by the quaternion $\lambda = [1\ 0\ 0\ 0]^{\mathrm{T}}$ and hence, according to (2.5), the DCM is a unit matrix. At time $\mathrm{t} = 0$, the object is located on axis y at the distance $\mathrm{R}_0 = \dfrac{\mathrm{T}\bar{\mathrm{v}}}{2\pi}$ and its velocity (vector appearing in (2.35)) and acceleration in (2.39) are defined by the following relations $\mathrm{v}_0 = [-\bar{\mathrm{v}}\ 0\ 0]^{\mathrm{T}}$, $w_0 = \begin{bmatrix} 0 & -\dfrac{\bar{v}^2}{R_0} & -g \end{bmatrix}^T$, where $\mathrm{g} = -9{,}81\,\mathrm{m/sec}^2$ is gravity acceleration. Accelerometers are installed at the trihedron axes so that matrix P_{w} in (2.39) looks like follows:

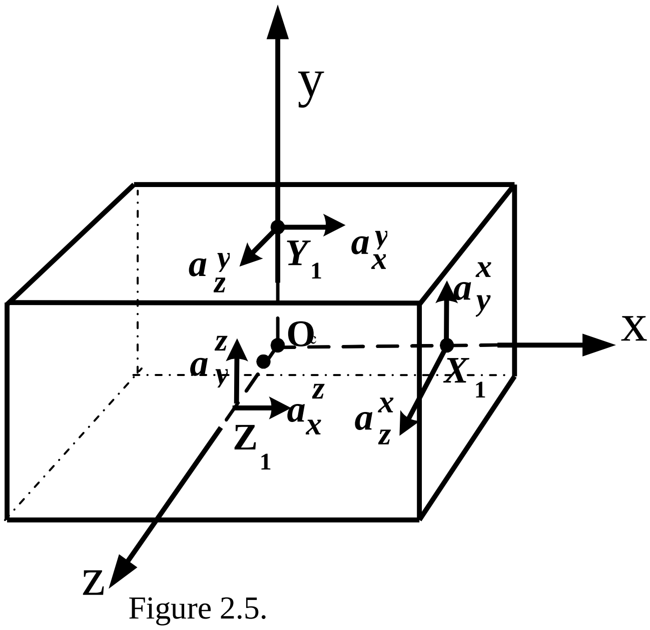


Figure 2.5.

$$P_w = 0{,}1 \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

We assume that there are six accelerometers and they are located so that matrix in (2.39) has the following structure:

$$\mathrm{U} = \begin{bmatrix} - & w_{21} & w_{31} \\ w_{12} & - & w_{32} \\ w_{13} & w_{23} & - \end{bmatrix},$$

where dashes denote missing components of acceleration measurements. We can illustrate this arrangement of accelerometers by the instrumental trihedron presented in Fig. 3, where X_1, Y_1, Z_1 are points on trihedron axes at which pairs of accelerometers are installed. The superscript specifies the axis where the accelerometer is installed, and the subscript specifies the direction of its measurement axis. For example, a_x^y means that the accelerometer measures acceleration of point Y_1 in the direction of OX axis.

Errors of accelerometers (n_w in (2.41)) can be modeled by uniformly distributed uncorrelated random numbers with zero expectation and variance $\sigma_w = 10^{-3}\ m/\sec^2$.

Thus, the above initial conditions and the assumptions about accelerometer errors allow us to model the operation of the inertial navigation system under study, which does not contain angular velocity sensors. However, to illustrate the algorithm described in Sec. 3.1, we will consider the situation where initial values of object velocities (angular and linear) can be found by GNSS by velocity measurements data processing of three points of the object for $t = 0$. In that regard, we assume that matrix in (2.35) is as follows:

$$\mathrm{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Note that since matrix P^{-1} does not exist, it is impossible to use algorithm (Laub and Shiflett, 1983).

Measurement errors (components of vector n_v in (2.37)) are assumed to be uncorrelated uniformly distributed random numbers with zero expectation and variance $\sigma_v = 10^{-1}$ m/sec. Using the algorithm in Sec. 3.1 for these initial data, we obtained the following estimates:

$$\omega(0) = [-10^{-3} \quad 6{,}4\cdot 10^{-3} \quad 0{,}1024]^T, \; v_0(0) = [-29{,}9955 \quad -3{,}4\cdot 10^{-3} \quad 0{,}0101]^T$$

and accepted them as respective initial conditions in modeling the inertial navigation system.

For other parameters, namely parameters that determine object's initial position and attitude, their initial values are accepted as their exact values for $t = 0$.

Simulation of the system's operation during 15 sec with the use of MATLAB procedures, namely, used procedure ode 45.m to integrate the system of differential equations and procedure rand.m, to generate random numbers.

The results of modeling are presented in Figs. 2.6–2.14. Figures 2.6–2.8 show how projections of the estimate of vector ω (deg/sec) vary in time. The horizontal line (6 deg/sec) in Fig. 2.8 corresponds to the exact value of ω_z. Figures 2.10–2.12 present the errors of determining object's coordinates. Figures 2.9–2.11 show the errors of finding the orientation μ_x, μ_y, μ_z (in degrees), which are upper diagonal elements of matrix $A^T(\bar{\lambda})A(\lambda)$ approximated in the form of (2.6) (Wittenburg, 1977):

$$A^T(\bar{\lambda})A(\lambda) \cong \begin{bmatrix} 1 & \mu_z & -\mu_y \\ -\mu_z & 1 & \mu_x \\ \mu_y & -\mu_x & 1 \end{bmatrix},$$

where λ denotes the value of the quaternion obtained by integration, $\bar{\lambda}$ is the exact value. Respectively, $A(\bar{\lambda}), A(\lambda)$ are the exact value of the cosine matrix and its estimate obtained as a result of integration.

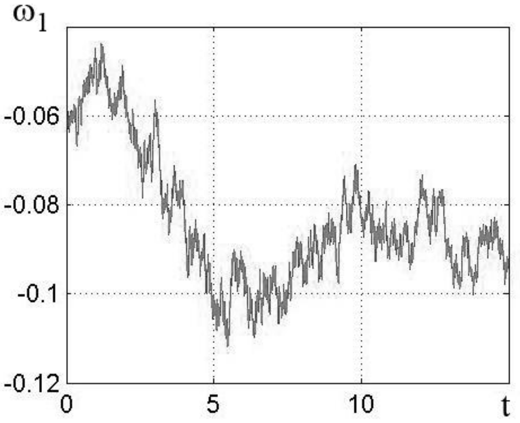


Fig. 2.6. Error of estimate of ω_x

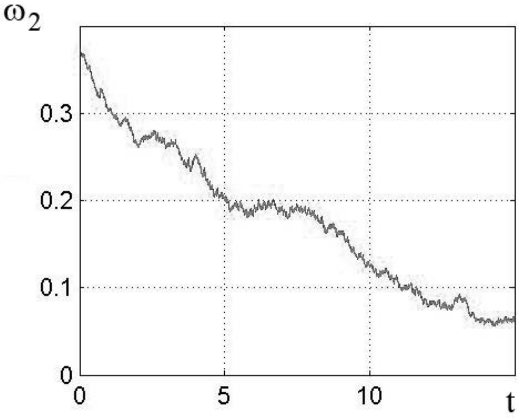


Fig. 2.7 Error of estimate of ω_y

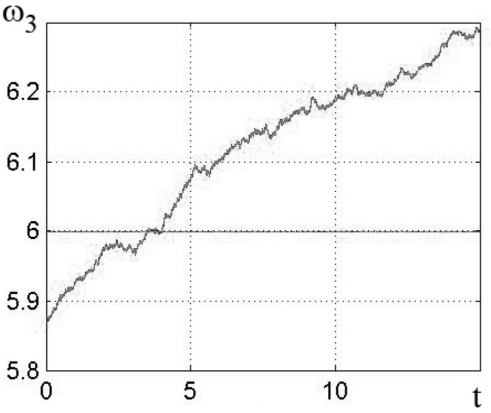


Fig. 2.8. Estimate of ω_z

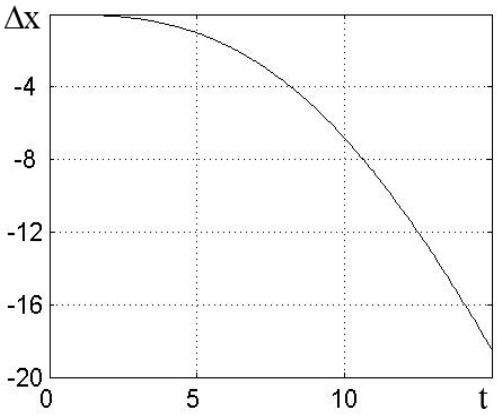


Fig. 2.9. Error of estimate of coordinate X

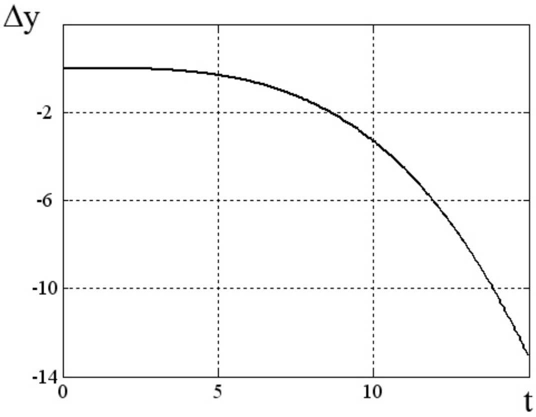


Fig. 2.10. Error of estimate of coordinate Y

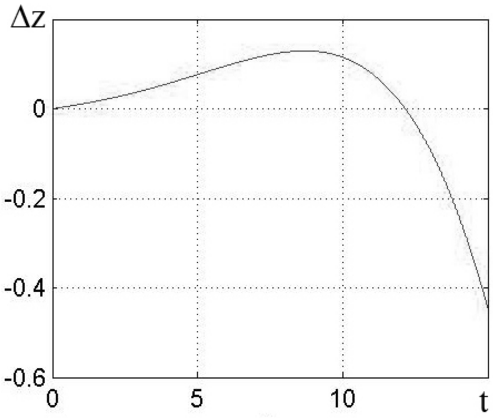


Fig. 2.11. Error of estimate of coordinate Z

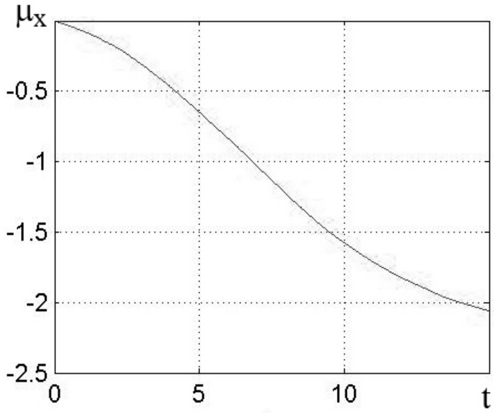


Fig 2.12. Estimate of rotation error about axis X .

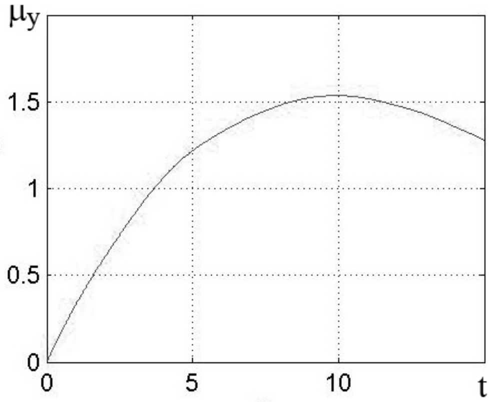


Fig. 2.13. Estimate of rotation error about axis Y .

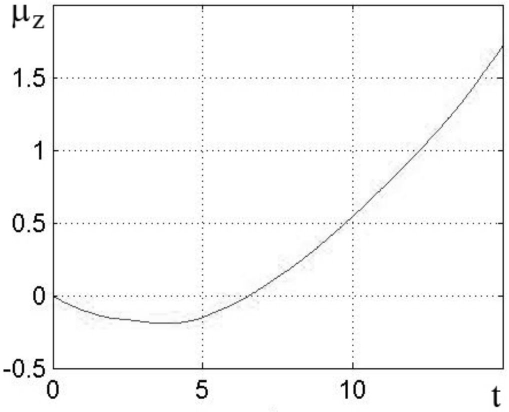


Fig. 2.14. Estimate of rotation error about axis Z .

Thus, as we can see from the above results of simulation, such inertial system using rather simple hardware (six accelerometers as in the example), under considerable operation time, may not ensure an adequate accuracy of navigation parameters. In such situations, it is expedient to correct the results of operation of such system by means of GPS. To this end, we will consider creation of the integrated system GPS/INS as applied to the INS under study. Thereupon, it is necessary to consider the "discrete" variant of INS described in Sec. 2.4.4.

2.4.4. Discrete Variant of INS

As we mentioned in Sec. 2.4.3, operation of the INS under study involves integration of a non-linear system of differential equations of 13th order. From the point of view of realization of such INS, it is expedient to consider "sampling" of this system, i.e., the case where the readings are taken not continuously but in regular intervals Δt , i.e., with frequency $f = 1/\Delta t$. Respectively, the required navigation parameters (DCM $A(\lambda)$, velocity V , and position $\mathbf{r}$) are calculated in time interval Δt .

In Sec. 2.3.1 we have presented expressions for quaternions $\delta\lambda(t_i)$ in terms of the vector of quasicoordinates $\nabla\theta_i$, which provide one approximation quality or another depending on the complexity.

In what follows, we will use elementary approximations (Euler's method) for quaternion $\delta\lambda(t_i)$ and for $\nabla\theta_i$:

$$\delta\lambda(t_i)=\begin{bmatrix} 1 \\ \frac{1}{2}\theta_i \end{bmatrix}, \quad \nabla\theta_i=\frac{\omega(t_i)+\omega(t_{i-1})}{2}\Delta t,$$
$$\omega(t_i)=\omega(t_{i-1})+\varepsilon(t_i)\Delta t, \tag{2.42}$$

where $\varepsilon(t_i)$ is the vector of angular accelerations defined by (2.40), from the results of accelerometer readings at instant of time t_i (in (2.39), components of vector $\omega(t_{i-1})$ are taken as elements of matrix Ω). Given the estimate of quaternion $\delta\lambda(t_i)$ obtained from (2.42), we use (2.15) and (2.16) to find the quaternion $\lambda(t_i)$ and then due to (2.5) to obtain matrix $A(\lambda(t_i))$. Using matrix $A(\lambda(t_i))$ defined by expression (2.40) to map vector $w_0(t_i)$ into the fixed frame, we can find estimates of the velocity $(v(t_i))$ and coordinates $(v(t_i))$ of the object (analogs of relations (2.20) and (2.21)):

$$v(t_i)=v(t_{i-1})+\tilde{w}(t_i)\Delta t,$$
$$r(t_i)=r(t_{i-1})+\frac{v(t_i)+v(t_{i-1})}{2}\Delta t.$$

Thus, the above relations allow using accelerometers' readouts at time instants t_i to estimate navigation parameters at the same t_i. This, in turn, allows using traditional integration algorithms GPS and INS to correct the results of operation of the INS under study.

2.4.5. Algorithms to Increase the Accuracy of Finding Object's Angular Velocity

We will show that increase in the number of accelerometers can be used to increase the accuracy of finding object's angular velocity ω (Larin and Tunik, 2013).

Let us consider the case of nine accelerometers. We will supplement the scheme of six accelerometers in Fig. 2.5 by three accelerometers at the point O whose sensitivity axes are directed along axes OX, OY, OZ, respectively, i.e., these accelerometers measure the acceleration of the origin of coordinates. Let us denote the readouts of these accelerometers by

a_x^0, a_y^0, a_z^0 and assume that the distance of each point X_1, Y_1, Z_1 from the origin of coordinates is L.

Denote: $n_y^x = a_y^x - a_y^0$, $n_x^y = a_x^y - a_x^0$, $n_z^x = a_z^x - a_z^0$, $n_z^y = a_z^y - a_y^0$, $n_x^z = a_x^z - a_x^0, n_y^z = a_y^z - a_y^0$. With such installation of the accelerometers, (2.38) or (2.39) yield the following relations:

$$\begin{gathered} 2L\varepsilon_1 = n_z^y - n_y^z, \\ 2L\varepsilon_2 = n_x^z - n_z^x \\ 2L\varepsilon_3 = n_y^x - n_x^y. \end{gathered} \tag{2.43}$$

$$\begin{gathered} 2L\omega_2\omega_3 = n_z^y + n_y^z, \\ 2L\omega_1\omega_3 = n_x^z + n_z^x \\ 2L\omega_1\omega_2 = n_y^x + n_x^y. \end{gathered} \tag{2.44}$$

Noteworthy is that Eqs. (2.43) coincide with (3.390) from (Andreev, 1967). Thus, in case of nine accelerometers, relations (2.44) define three more values: $\omega_1\omega_2$, $\omega_1\omega_3$, $\omega_2\omega_3$. It is expedient to use this information to adjust the results of integration of the angular acceleration ε. Note that if two out of three components of vector ω are zero (rotation about a fixed axis), relations (2.44) cannot be used to adjust the results of integration.

Thereupon, it is expedient to supplement the above system of nine accelerometers with three more accelerometers such that acceleration along axis OX is also measured at the point X_1, and respectively: along axis OY - in the point Y_1, and along axis OZ - in the point Z_1. Note that this layout of accelerometers coincides with that in Fig. 3.7 (Andreev, 1967). Let readouts of these three accelerometers be a_x^x, a_y^y, a_z^z . Denote:

$$n_x^x = a_x^x - a_x^0,\ n_y^y = a_y^y - a_y^0,\ n_z^z = a_z^z - a_z^0.$$

In the measurement system of 12 accelerometers under study, relations (2.43) and (2.44) should be supplemented with the following ones:

$$\begin{gathered} 2L\omega_1^2 = n_x^x - n_y^y - n_z^z, \\ 2L\omega_2^2 = -n_x^x + n_y^y - n_z^z \\ 2L\omega_3^2 = -n_x^x - n_y^y + n_z^z \end{gathered} \tag{2.45}$$

Thus, in the case under consideration (12 accelerometers), we can use relations (2.44) and (2.45) to adjust the results of integration of the angular acceleration.

Let us briefly consider the problem of using relations (2.44) to increase the accuracy of determination of ω in case of nine accelerometers and a similar problem in case of 12 accelerometers. In the latter case, relations (2.45) are used along with the above-mentioned relations (4.2). Thus, let us consider the case of nine accelerometers whose readings determine both the angular acceleration vector ε (relations (2.43)) and components of vector $\Omega_n = [\omega_2\omega_3 \;\; \omega_1\omega_3 \;\; \omega_2\omega_1]^T$ (relations (2.44)). Assuming that $\Delta\omega_i = \omega(t_i) - \omega(t_{i-1})$ is a small value, we can write the following relations:

$$\Omega_n = H\Delta\omega_i + \Omega_{n0} \quad H = \begin{bmatrix} 0 & \omega_3 & \omega_2 \\ \omega_3 & 0 & \omega_1 \\ \omega_2 & \omega_1 & 0 \end{bmatrix} \quad \Omega_{n0} = [\omega_2\omega_3 \;\; \omega_1\omega_3 \;\; \omega_1\omega_2]^T . \tag{2.46}$$

In (2.46) Ω_n is defined by (2.44), and the values of components of vector ω, appearing in H and Ω_{n0}, correspond to the values of components of vector $\omega(t_{i-1})$. In other words, as a result of the assumption about the smallness of, we obtain a standard problem of parameters estimation by the weighed least squares method (Bryson and Ho-Yu-Chi, 1969). Namely, there is some initial estimate $\Delta\overline{\omega}_i = \frac{\varepsilon(t_i) + \varepsilon(t_{i-1})}{2}\Delta t$. Vector z is observed according to (2.46) as follows:

$$z = \Omega_n - \Omega_{n0} = H\Delta\omega + \nu, \tag{2.47}$$

where ν is the vector of measurement errors. The estimate of $\Delta\widehat{\omega}_i$ is defined by relation (12.2.7) (Bryson and Ho-Yu-Chi, 1969):

$$\Delta\widehat{\omega}_i = \Delta\overline{\omega}_i + PH^T R^{-1}(z - H\Delta\overline{\omega}_i) \tag{2.48}$$

$$P^{-1} = M^{-1} + H^T R^{-1} H .$$

Here, M is the covariance matrix of errors of estimate $\Delta\overline{\omega}_i$, R is the covariance matrix of errors of measurements ν in (2.47). Finally, the value of vector at time is defined by the relation

$$\omega(t_i) = \omega(t_{i-1}) + \Delta\widehat{\omega}_i \tag{2.49}$$

where $\Delta\widehat{\omega}_i$ can be found from (2.48).

Matrix P^{-1} can be ill-conditioned; therefore, to find matrix P appearing in (2.48) it may be expedient to use the approach from (Larin, 1999, 2006).

Since matrices M, R are symmetric and positive definite, we can represent them as $M = m^2,\ R = r^2$, т.е. $m = M^{1/2},\ r = R^{1/2}$. Respectively, we can write the expressions for matrix P^{-1} as

$$P^{-1} = \left[m^{-1}\ \ H^T r^{-1}\right]\left[m^{-1}\ \ H^T r^{-1}\right]^T. \tag{2.50}$$

Using the procedure of QR-factorization, we will rearrange matrix $\left[m^{-1}\ \ H^T r^{-1}\right]^T$ as follows:

$$\left[m^{-1}\ \ H^T r^{-1}\right]^T = Q[\rho\ 0]^T, \tag{2.51}$$

where Q is an orthogonal matrix and ρ is an invertible matrix.

With regard for $Q^T Q = I$, substituting (2.50) into (2.51) yields $P^{-1} = \rho^T \rho$ or: $P = \rho^{-1}\rho^{-T}$. Thus, we can represent expression (2.48) as

$$\Delta\hat{\omega}_i = \Delta\bar{\omega}_i + \rho^{-1}\rho^{-T} H^T R^{-1}(z - H\Delta\bar{\omega}_i). \tag{2.52}$$

If we assume that $M = \mu^2 I, R = \gamma^2 I$, then we can write relations (2.52) as

$$\Delta\hat{\omega}_i = \Delta\bar{\omega}_i + \rho^{-1}\rho^{-T} H^T(z - H\Delta\bar{\omega}_i)\ , \tag{2.53}$$

where ρ is defined by the QR-factorization of the following matrix

$$\left[\lambda I\ \ H^T\right]^T \quad \lambda = \frac{\gamma}{\mu}. \tag{2.54}$$

Note that the aforementioned correction algorithm can also be used in case of 12 accelerometers. In this case, the matrix H and vector Ω_{n0} appearing in (2.46) are as follows:

$$H = \begin{bmatrix} 0 & \omega_3 & \omega_2 & 2\omega_1 & 0 & 0 \\ \omega_3 & 0 & \omega_1 & 0 & 2\omega_2 & 0 \\ \omega_2 & \omega_1 & 0 & 0 & 0 & 2\omega_3 \end{bmatrix}^T, \quad \Omega_{n0} = \left[\omega_2\omega_3 \quad \omega_1\omega_3 \quad \omega_1\omega_2 \quad \omega_1^2 \quad \omega_2^2 \quad \omega_3^2\right]^T.$$

As well as in case of nine accelerometers, components H and Ω_{n0} are defined by components of vector $\omega(t_{i-1})$. Components of vector Ω_n are defined by relations (2.44) and (2.50).

Example 2.5

Nine accelerometers are mounted at the object, namely, six accelerometers as is shown in Fig. of 3.2 and three more accelerometers at the origin of coordinates. The origin of coordinates of the instrumental trihedron fixed to the moving frame $(Ox'y'z')$ is specified by vector $R = [0\ 1\ 0]^T$. The orientation of the moving coordinate system is specified by the following time dependence of Euler angles $\psi(0) = 0,\ \theta(0) = \frac{\pi}{4},\ \varphi(0) = 0,\ \ \dot{\psi} = 1,\ \dot{\theta} = 0,\ \dot{\varphi} = 10$. Projections of the angular velocity onto axes of moving coordinate system are defined by (2.8):

$$\begin{aligned} \omega_1 &= \dot{\psi}\ \sin\vartheta\ \sin\varphi + \dot{\vartheta}\cos\varphi, \\ \omega_2 &= \dot{\psi}\ \sin\vartheta\ \cos\varphi - \dot{\vartheta}\sin\varphi, \\ \omega_3 &= \dot{\psi}\ \cos\vartheta + \dot{\varphi}. \end{aligned} \tag{2.55}$$

In the example under consideration, absolute value of the angular velocity vector exceeds 600 deg/sec. In this regard, in the example we assume that $\Delta t = 10^{-3}$ sec. The initial orientation (quaternion) is defined by relations (2.2) and we assume that $\sigma_w = 10^{-3}$ m/sec^2 and $L = 0{,}1$ m. We modeled the error of the initial alignment of angular velocity as follows. As the initial value we took

$$\tilde{\omega}(0) = \frac{1}{2}\omega(0) \tag{2.56}$$

where ω is the exact value defined by (2.55).

We assume that $\lambda = 0{,}1$ in (2.54). The results of the errors of determination of kinematic parameters of motion obtained in modeling of the motion during 15 sec are presented in Figs. 2.15 and 2.16.

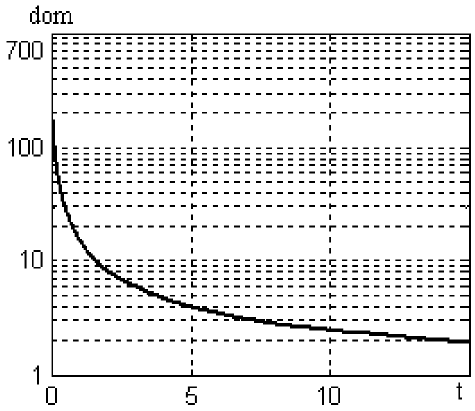


Fig. 2.15. Error of the estimate of ω

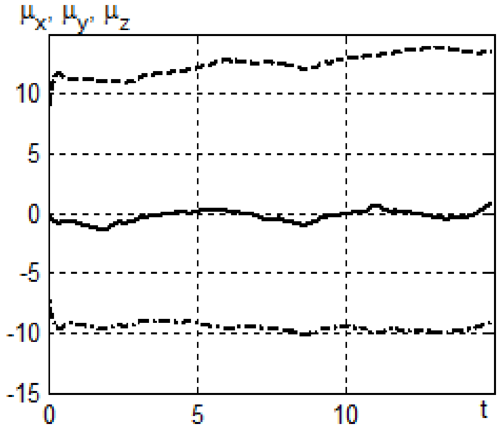


Fig. 2.16. Errors of finding the orientation

Figure 2.15 shows the values (dimension deg/sec) of following parameter:

$$\mathrm{dom}(t_k) = \left(\sum_{i=1}^{k} \| \omega(t_i) - \tilde{\omega}(t_i) \| \right) \Big/ k,$$

where $\tilde{\omega}(t_i)$ is the estimate of angular velocity obtained according to (2.49) and $\omega(t_i)$ is the exact value of the angular velocity vector, i.e., the quantity characterizes the accuracy of the estimate of current value of the angular velocity vector obtained according to (2.49) (in Fig. 2.16 the dimension of dom is deg/sec). Figure 2.16 shows the values of errors of finding the orientation (μ_x– dashed line, μ_y– dash-and-dot line, μ_z – solid line, dimension is deg). Thus, based on the above results we may state that in the example under study using the correction algorithm (2.49) has substantially increased the accuracy of finding the current value of ω and thus increased the accuracy of finding the orientation. According to the graph dom (Fig. 2.15), the 15 sec error of ω is about 2 deg/sec, while according to (2.56), the error of the initial ω is about 300 deg/sec. As a result, we may state an increase in the accuracy of determination of the orientation (Fig. 2.16). For example, the error of finding the orientation initially increases in different ways due to coarse initial setting of ω. However, after a decrease in the error of determination of the current value of ω we may state that errors of finding the orientation do not undergo substantial changes.

Let us emphasize that while the error of initial setting of angular velocity is about 300 deg/sec, the error of finding object’s orientation for 15 sec is about 10 degrees.

Example 2.6

Let us consider a measurement system containing 12 accelerometers. We will keep the initial data from Example 2 (motion parameters, error of the initial setting, etc.) and only change the values of σ_w and λ. We assume that $\sigma_w = 10^{-1}$m/sec^2 and $\lambda = 700$, i.e., the accuracy of accelerometers is two orders of magnitude less. The results of modeling are presented in Figs. 2.17 and 2.18 (the notation coincides with that accepted in Figs. 2.15 and 2.16).

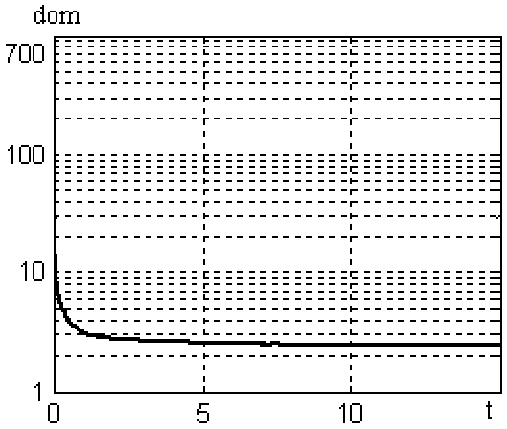


Fig 2.17 –Error of the estimate of ω.

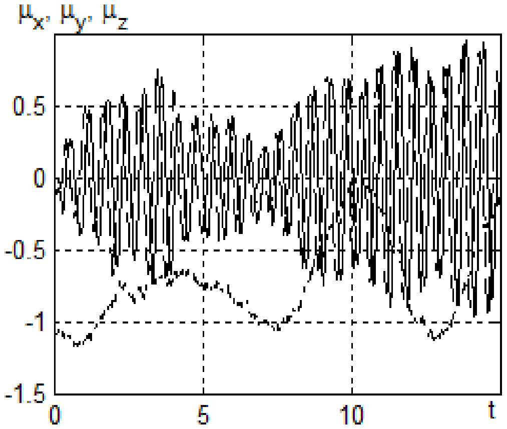


Fig. 2.18 –Error of the orientation finding

These results witness a much higher efficiency of the measurement system containing 12 accelerometers. As we can see from Fig. 2.17, despite of the fact that acceleration measurement errors have increased by two orders of magnitude, the error of current value of ω for 15 sec is of almost the same order as in Example 2. We may state that in such system the error of the initial setting of ω decreases much faster and hence the errors of finding the orientation decrease (comparing Fig. 2.18 and Fig. 2.16).

Example 2.7

Let us continue considering Example 2.4. We will keep all the input data (motion parameters, accelerometers accuracy, etc.) from Example 2.4. However, we will assume that the measurement system contains not 6 but 12 accelerometers (like in Example 2.6). We assume in (2.54) that $\mu = 7$. Figure 2.19 shows the simulation results.

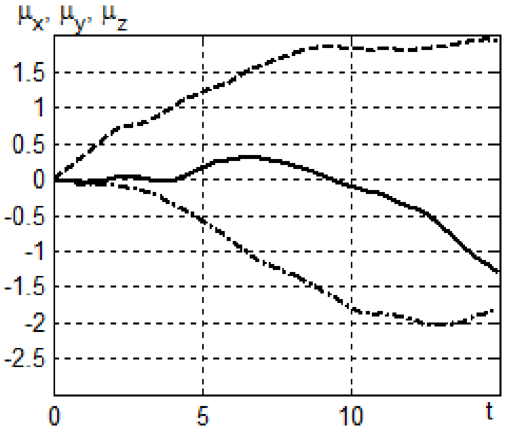


Fig. 2.19

Comparing them with the results presented in Fig. 2.16, we may state that the accuracy of finding the orientation substantially increases. In this example, unlike Example 2.4, to estimate ω, we used the procedure of integration of a nonlinear differential equation (ode 45.m) (Mathworks.com, 2018) for the finite-difference scheme (2.47).

We have presented the algorithms of operation of autonomous inertial navigation systems that do not contain angular velocity sensors. We have considered systems containing six, nine, and 12 accelerometers. Since six accelerometers are enough to measure angular acceleration of the object, for a system that contains nine or 12 accelerometers there is a possibility to increase the accuracy of finding the angular velocity vector of the object by using additional information due to additional accelerometers. We have presented respective correction algorithms and have used examples to show that such systems can be efficient if the object moves with a large angular velocity (when using RGs is known to become problematic (Bogdanov et.al., 2011)).

2.5. Summary for Chapter 2

2.5.1. In the 2nd chapter it was considered the SINS software structure and the basic relations, which are used for implementation of separate algorithms in this structure.

2.5.2. The distinctive feature of considered algorithms of the rotational mechanization consists of procedures, using analytical expressions for approximation of quaternions corresponding to the rigid body small turn. Using several methods of approximation of these quaternions, it is possible to vary the accuracy of the rotational mechanization algorithms.

2.5.3. Procedure of approximation of the quasi-coordinates by quadratic splines is used in the rotational as well as in the translational mechanization algorithms allowing obtaining the results of computing in quadratures instead of direct integration of the rotational and translational differential kinematic equations.

2.5.4. Distinctive features mentioned in the items 2.5.3 and 2.5.4 allow us to simplify essentially the SINS algorithms and simultaneously to preserve their acceptable accuracy.

2.5.5. In this chapter the algorithms of the rotational and translational mechanization for the gyro-free accelerometer based SINS were considered also. They are essentially effective for UAV spinning with respect to the longitudinal axis with comparatively great angular rate. It was shown that increasing of number of redundant accelerometers above minimal number leads to significant increasing of accuracy of the gyro-free SINS integration algorithms.

2.5.6. Algorithms presented in this paper were tested by results of the mathematical modeling, which witness sufficiently high accuracy of the rotational and translational mechanization algorithms for the traditional as well as for the gyro-free SINS.

CHAPTER 3 – ALGORITHMS OF CORRECTION OF INERTIAL NAVIGATION SYSTEM

The design process of GNSS/INS integrated navigation system includes the tradeoff between performance and cost, and the cost may be significantly influenced by the level of modifications required to be done inside of both systems (Grewal, Weill, and Andrews, 2001). There is a wide variety of integration approaches, from loosely coupled and to ultra-deep integration (Schmidt, 2008, 2011), (Schmidt and Phillips, 2008, 2011 a, 2011 b).

The majority of loosely coupled implementations use only standard outputs of the GNSS receiver (like position and velocity) and INS (position, velocity, and attitude) as inputs to a system integration filter (typically, Kalman filter). More tightly coupled implementations use less standard subsystem outputs such as pseudoranges from GNSS receivers or raw accelerations from INS and usually require more intervention in the subsystem operation. For more details regarding integration techniques, see (Grewal, Weill, and Andrews, 2001).

Figure 3.1 presents the scheme of loosely coupled approach (compensation scheme) of GNSS/INS integration.

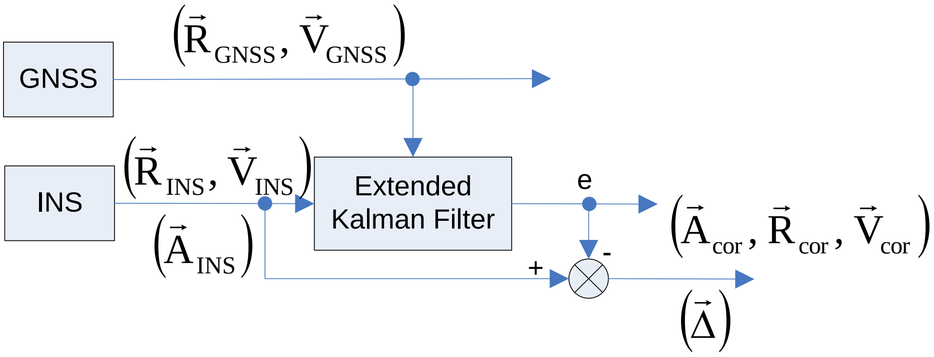

GNSS
$\left(\vec{R}_{GNSS}, \vec{V}_{GNSS}\right)$
INS
$\left(\vec{R}_{INS}, \vec{V}_{INS}\right)$
$\left(\vec{A}_{INS}\right)$
Extended Kalman Filter
e
$\left(\vec{A}_{cor}, \vec{R}_{cor}, \vec{V}_{cor}\right)$
-
+
$\left(\vec{\Delta}\right)$

Figure 3.1 – Scheme of loosely coupled approach of GNSS/INS integration (Schmidt, 2011), (Schmidt and Phillips, 2011 a, 2011 b), (Titterton and Weston, 2004, fig. 13.18)

3.1. Algorithms of correction of inertial navigation system

Let us write filter equations. Denote by $\mu, \delta v, \delta r$ vectors of INS errors in the inertial frame (μ is the vector of small turn of attitude error, $\delta v, \delta r$ are vectors of velocity and coordinate errors). Let $\tilde{a} = [\tilde{a}_1, \tilde{a}_2, \tilde{a}_3]^T$ be vector of specific force (hereinafter, the superscript T means transposition). We accept the equation of INS errors propagation, in the form similar to (7.149) (Grewal and Andrews, 1993):

$$\dot{x} = Fx + n,$$

$$x = \begin{bmatrix} \mu \\ \delta v \\ \delta r \end{bmatrix}, \qquad F = \begin{bmatrix} 0 & 0 & 0 \\ C & 0 & 0 \\ 0 & I & 0 \end{bmatrix}, \qquad \boldsymbol{C} = \begin{bmatrix} 0 & -\tilde{a}_3 & \tilde{a}_2 \\ \tilde{a}_3 & 0 & -\tilde{a}_1 \\ -\tilde{a}_2 & \tilde{a}_1 & 0 \end{bmatrix}, \qquad (3.1)$$

n is white noise vector. Hereinafter, 0 and I are zero and unit matrices of respective sizes (3x3).

Expressions for matrix F (3.1) and (7.149) in (Grewal and Andrews, 1993) are analogous. Thus, in (3.1) we neglect the Earth rotation rate Ω and the square of the Schuler frequency ω_s^2. This neglecting is justified by low sensitivity of primary MEMS sensors, which are applied in small UAV navigation systems.

As a discrete analog of (3.1), i.e., relation that expresses errors propagation in terms of a small time interval Δt, the following equation is taken:

$$x_{k+1} = \Phi_k x_k + n_k,$$

$$\Phi_k = I + F\Delta t + \frac{(\Delta t)^2}{2} F^2 = \begin{bmatrix} I & 0 & 0 \\ C \cdot \Delta t & I & 0 \\ C \cdot \frac{(\Delta t)^2}{2} & I \cdot \Delta t & I \end{bmatrix}, \qquad (3.2)$$

where n_k is vector of random errors of INS operation. The subscript k corresponds to instant of time $k\Delta t$. We may suppose that Δt is a step of INS operation and (3.2) is the initial equation of errors. Assume that at the k_{th} step of INS operation, GPS provides information about estimates of coordinates and velocity of the object, i.e., the following observation process takes place:

$$z_k = Hx_k + v_k$$

$$H = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}, \qquad (3.3)$$

where v_k is the measurement error.

If INS errors are corrected at the same step, then the error modification equation corresponding to (3.2) is

$$x_{k+1} = \Phi_k x_k + \Gamma_k u_k + n_k, \qquad (3.4)$$

where u_k is a correcting signal. The choice of matrix Γ_k determines the correction process.

Thus, using relations (3.3) and (3.4), we can formulate the INS correction problem as a standard linear quadratic Gaussian problem. It is generally known (see for example, item 14.7 in (Bryson and Ho-Yu-Chi, 1969)) that solution of this problem is given by

$$\begin{aligned} u_k &= -D_k \hat{x}_k, \\ \hat{x}_k &= \bar{x}_k + K_k\left(z_k - H\bar{x}_k\right), \\ \bar{x}_{k+1} &= \Phi_k \hat{x}_k + \Gamma_k u_k, \end{aligned} \tag{3.5}$$

where the matrix D_k of feedback loop coefficients is defined by the form of the functional being optimized. The filter gain matrix (K_k) generating the vector of optimal estimate $\hat{x}_k$ is defined as follows (the filter equations) (Grewal and Andrews, 1993):

$$K_k = M_k H^T\left(HM_k H^T + R_k\right)^{-1}, \tag{3.6}$$

$$M_{k+1} = \Phi_k P_k \Phi_k^T + Q_k, \tag{3.7}$$

$$P_k = M_k - K_k\left(HM_k H^T + R_k\right)K_k^T. \tag{3.8}$$

Matrices Q_k, R_k are covariance matrices of noise n_k , ν_k , which appear in (3.3) and (3.4). M_0 is a given covariance matrix of initial estimate of vector x. Note that correction of INS errors usually happens after m> 1 steps. In this case, between the times of correction, INS errors are modified according to Eq. (3.2), and their complete correlation matrix is modified according to (3.7) (we may assume that $H = 0$ at these steps). At the correction step, modifications of the complete correlation matrix are described by Eq. (3.8).

A significant feature of the problem under study is that matrices Φ_k and H form an incompletely observable pair in observability matrix (Larin, 1999). The observability matrix has the following form (Katsuhiko, 1997):

$$Ob = \begin{bmatrix} H \\ H\,\Phi_k \\ \dots \\ H\,(\Phi_k)^{n-1} \end{bmatrix},$$

and can be calculated in Matlab by means of "obsv" function ("MATLAB Documentation", 2018).

Let us consider a bit more detailed matrices Φ_k and H. Since the skew-symmetric C appeared in Eqs. (3.1) (3.2) is singular (its rank is equal to 2), some orthogonal matrix U exists, which can transform C as follows:

$$U^T CU = \begin{bmatrix} 0 & 0 \\ 0 & \lambda \end{bmatrix}.$$

Here λ is an invertible 2x2 matrix. If we form 9x9 matrix V (containing matrices U at the main diagonal) from the equation above and two unit matrices ($V = diag\{U, E, E\}$) and then apply it to similarly transform matrices Φ_k and H, they will have the following structure:

$$\overline{\Phi}_k = V^T \Phi_k V = \begin{bmatrix} 1 & 0_{[1x8]} \\ 0_{[8x1]} & \tilde{\Phi}_{k[8x8]} \end{bmatrix}, \qquad \overline{H} = HV = H = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}.$$

It might be seen that matrix V transformed Φ_k in such a manner that the first element of new matrix $\overline{\Phi}_k$ is equal to 1, and all other elements of the first column and row are zero. Due to the specific form of matrix H, multiplication by matrix V did not change its form. As is seen, elements of the first column of this matrix are zero.

Hence the pair (Φ_k, H) has one unobservable mode. Specific meaning of this mode depends on the structure of matrix C, i.e., on the specific value of acceleration vector w. This requires various maneuvers of the object to be used to correct all the components of its attitude vector (see, for example, (Bar-Itzhack, 1982)).

3.1.1 Substantiating the Necessity of Computing the Cholesky Multipliers in the Kalman Filtering Algorithms

Ensuring the convergence of evaluations is one of the basic problems in using Kalman filtration in traditional form in integrated navigation systems. A significant feature of the Kalman filtration is that the covariance matrix of state variables P_k is ill-conditioned. Also, as was shown above, the pair of matrices (Φ_k, H) has one unobservable mode. Moreover, the other modes may be weakly observable. These circumstances can lead to accumulation of computing error and cause divergence of calculations in Kalman filtration algorithms.

As is generally known, loss of positive definiteness of a priori $P(-)$ and a posteriori $P(+)$ covariance matrices of state variables of expanded Kalman filter is the main reason that the algorithm of expanded Kalman filter stops converging (Watanabe and Tzafestas, 1989). To avoid this phenomenon (which stops expanded Kalman filter), attention is paid to perform all operations with these matrices in the form of generalized Cholesky multipliers, whose product is always a positive definite matrix. Moreover, using Cholesky multipliers improves the convergence of Kalman filtration problem since the condition number of Cholesky multiplier is the square root less than the condition number of the respective matrix.

The study (Grewal and Andrews, 1993) analyzes roundoff error propagation in Kalman filters. The Kalman filtering algorithm is considered as two computational loops: estimation loop (where $\hat{x}_k$ is calculated) and gain loop (where P_k and K_k are calculated). Experimental analysis shows that roundoff errors in the estimation loop are compensated by the feedback mechanism if gain loop is correct. This correction applies to all type of errors, including those introduced by roundoff, due to noise and apriori estimation errors. In the gain loop the effects of roundoff error cannot be detected and corrected and, therefore, they propagate and accumulate unchecked (Grewal and Andrews, 1993). Due to big number of matrix multipliers in the gain loop it suffers more from roundoff errors propagation in "conventional" form of Kalman filter. Even bounded errors in the computed value of P_k may momentarily destabilize the estimation loop. Those roundoff errors can cause the computed value of P_k to have a negative characteristic value. And since P_k is a factor in the Kalman gain, its negative characteristic value can cause gain in the prediction error feedback loop to have the wrong sign, which may momentarily destabilize the estimation loop. And even after recovery the convergence would be slower than it should be (Grewal and Andrews, 1993).

Therefore, it makes sense to increase the accuracy of computing procedures. Thereupon, such problems usually use algorithms that allow calculating the Cholesky multipliers (factors) of respective covariance matrices.

According to (Grewal and Andrews, 1993), techniques to solve the Riccati equation in more numerically stable implementations of Kalman filter include the following ones:

1. Factoring the *covariance matrix of state estimation uncertainty* P (the dependent variable of the Riccati equation) in Cholesky factors or into modified Cholesky factors (unit triangular and diagonal factors).
2. Factoring the *covariance matrix of measurement noise* R to reduce the computational complexity of the observational update implementation.
3. *Block matrix factorizations* of matrix expressions in the Riccati equation. In the common approach, two different factorizations are used to represent two sides of equation

$$CC^T = AA^T + BB^T = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} A^T \\ B^T \end{bmatrix}$$

The alternative Cholesky factors C and $\begin{bmatrix} A & B \end{bmatrix}$must then be related by orthogonal transformations (triangulizations). A QR decomposition of $\begin{bmatrix} A & B \end{bmatrix}$ will yield the corresponding solution of the Riccati equation in terms of Cholesky factor of the covariance matrix. Also it is necessary to mention about another method of the Cholesky factors computing via Bierman's $UD-$factorization (Bierman, 1977), which will be used further.

Here are listed the techniques implemented in our approach. For more details regarding other techniques, see (Grewal and Andrews, 1993). Note that for description of a matrix transformation process into an equivalent product of factors both terms decomposition and factoring (factorization) are used usually interchangeably (Grewal and Andrews, 1993).

3.1.2 Evaluating the Cholesky multipliers by means of QR factorization

A number of algorithms for evaluating the Cholesky multipliers of the covariance matrix of filter (3.7), (3.8) are known (see, for example (Grewal and Andrews, 1993)). Below we will describe an algorithm (similar to that presented in (Larin, 1992, 1993), (Larin and Aliev, 1993), (Aliev and Larin, 1998)) based on QR decomposition of matrix (Voyevodin and Kuznetsov, 1984), (Lawson and Hanson, 1974).

QR decomposition (or QR factorization) of a matrix is its decomposition into an orthogonal matrix and a triangular matrix.

Any real square matrix A may be decomposed as

$$A=QR,$$

where Q is an orthogonal matrix (i.e., $Q^TQ=I$) and R is an upper triangular matrix (or right triangular matrix). If A is nonsingular (i.e., invertible), then this factorization is unique.

More generally, it is possible to decompose a complex rectangular [mxn] (m≥n) matrix A as a product of unitary [mxm] matrix Q and upper triangular [mxn] matrix R. The bottom (m-n) rows of [mxn] upper triangular matrix entirely consist of zeros.

There are several methods to actually compute QR decomposition, such as the Gram–Schmidt process, Householder transformations, and Givens rotations. Each has a number of advantages and disadvantages. For more details about actual computing of QR decomposition see (Grewal and Andrews, 1993). In Matlab, the QR decomposition algorithm is implemented by "qr" function (Mathworks.com, 2018).

Let us outline two variants of the algorithm of computing the Cholesky multipliers of the covariance matrix of the filter. In the first case, we will assume that matrix R_k is invertible, in the second we will not assume this.

Thus, let us consider the case of invertibility of matrix R_k**.** Let m_k, p_k, q_k, r_k be the Cholesky multipliers of matrices M_k, P_k, Q_k, R_k, respectively, i.e.,

$$M_k = m_k m_k^T,\ \ \mathrm{P}_k = p_k p_k^T,\ \ \mathrm{Q}_k = q_k q_k^T,\ \ \mathrm{R}_k = r_k r_k^T. \tag{3.9}$$

In the case of invertibility of matrix R_k, we can rearrange relation (3.8) as

$$p_k p_k^T = m_k \left(I + m_k^T H^T R_k^{-1} H m_k\right)^{-1} m_k^T . \tag{3.10}$$

Let us present the expression in parentheses as the product of two rectangular matrices

$$\begin{aligned} &I + m_k^T H^T R_k^{-1} H m_k = N_k N_k^T , \\ &N_k = \left[I \quad m_k^T H^T r_k^{-1}\right] \end{aligned}$$

By means of orthogonal matrix U and using the algorithm of QR decomposition, we will transform matrix N^T as follows:

$$\begin{bmatrix} S_k \\ 0 \end{bmatrix} = U_k N_k^T , \tag{3.11}$$

where S_k is an invertible matrix.

Thus, according to (3.10) we get

$$p_k = m_k S_k^{-1}. \tag{3.12}$$

Similarly, we will represent the right-hand side of (3.7) as the product of two rectangular matrices and will use QR decomposition of these matrices (Voyevodin and Kuznetsov, 1984), (Lawson and Hanson, 1974) by the orthogonal matrix V_k :

$$m_{k+1} m_{k+1}^T = T_k T_k^T ,$$

$$T_k = \left[\Phi_k p_k \quad q_k\right]; \tag{3.13}$$

$$\begin{bmatrix} X_k^T \\ 0 \end{bmatrix} = V_k T^T , \tag{3.14}$$

$$m_{k+1} = X_k . \tag{3.15}$$

Thus, we will calculate the multiplier p_k from the given m_k, r_k according to (3.11) and (3.12) and then calculate multiplier m_{k+1} according to (3.13-3.15).

Now, let us remove the assumption about invertibility of matrix R_k. Generally, to exclude the operation of inversion of matrix r_k, we will transform matrix N_k. Using the orthogonal matrix Ω_k in the QR decomposition, we will bring matrix $m_k^T H^T$ to the following form:

$$\begin{bmatrix} Y_k \\ 0 \end{bmatrix} = \Omega_k m_k^T H^T .$$

Assuming that matrix Y_k is invertible, we will introduce the following square matrix with $\tilde{W}_k = diag\{r_k^T \cdot Y_k^{-1}, I\}$ and will use it to transform matrix N_k.

$$N_k = \Omega_k^T \tilde{W}_k^{-1} \tilde{N}_k,$$

$$\tilde{N}_k = \begin{bmatrix} \tilde{W}_k \Omega_k & \begin{bmatrix} I \\ 0 \end{bmatrix} \end{bmatrix}.$$

Obviously, $\left(N_k N_k^T\right)^{-1} = \Omega_k^T \tilde{W}_k^T \left(\tilde{N}_k \tilde{N}_k^T\right)^{-1} \tilde{W}_k \Omega_k$.

Thus, if the orthogonal matrix $\tilde{U}$ transforms matrix $\tilde{N}^T$ similarly to (3.11), i.e.,

$$\begin{bmatrix} \tilde{S}_k^T \\ 0 \end{bmatrix} = \tilde{U}_k \tilde{N}_k^T,$$

we can write the following expression for matrix p_k, which does not contain operation of inversion of matrix r_k:

$$p_k = m_k \Omega_k^T \tilde{W}_k^T \tilde{S}_k^{-1}.$$

The relations that define m_{k+1} in the considered case of singular matrix r_k obviously will not change, i.e., matrix m_{k+1} will be defined by relations (3.13)-(3.15).

Derivation of the relations (3.10)-(3.15) is given in Appendix A.

3.1.3 Calculating the Cholesky Multipliers by the Cholesky and Bierman Methods

(Applying the Cholesky and Bierman methods in Kalman filtration algorithms)

Along with QR factorization, Cholesky decomposition algorithms are also used.

The Cholesky decomposition algorithm is a procedure to calculate elements of triangular Cholesky factor of a symmetric, nonnegative definite matrix. It solves the Cholesky decomposition equation $P = CC^T$ for a triangular matrix C, given the matrix P (Grewal and Andrews, 1993). Two variants of Cholesky decomposition algorithm are presented in (Grewal and Andrews, 1993), (Table 6.3): with lower and upper triangular results. A matrix is called upper triangular if its nonzero elements are on and above its main diagonal and lower triangular if they are on or below the main diagonal. In Matlab, the Cholesky decomposition algorithm is implemented by function "chol" (Mathworks.com, 2018).

Kalman filtration also widely uses modified Cholesky decomposition algorithm. Modified Cholesky decomposition (or as it is often called, Bierman’s UD decomposition) of a symmetric positive definite matrix M is product decomposition $M = UDU^T$ such that U is unit upper triangular and D is diagonal (Bierman, 1977). The upper triangular matrix U is called unit upper triangular if all its diagonal elements are 1. Similarly, a lower triangular matrix L is called unit lower triangular if all its diagonal elements are 1. (Grewal and Andrews, 1993).

A procedure for implementing UD decomposition is presented in Table 6.4 from (Grewal and Andrews, 1993). As is stated, this algorithm is only slightly different from the upper triangular Cholesky decomposition algorithm presented in Table 6.3 from (Grewal and Andrews, 1993). An important distinction of this modified Cholesky decomposition algorithm is that it does not require taking square roots. That is why it has been called as “square root filtering without square roots” (Grewal and Andrews, 1993).

Let us consider here the Bierman method for calculation of Cholesky multipliers.

Using the Bierman method in factorization of covariance matrices substantially reduces the amount of operations of the Kalman filtration algorithm (in comparison with the QR-factorization algorithms) and respectively, accelerates the process of calculation of the navigation solution as a whole. This method deals with matrix elements rather than entire matrices or vectors. This method also does not include the operation of calculating the inverse matrix and square root. Thus, the main advantage of the Bierman method as compared with QR factorization is the reduced amount of operations in calculating the Cholesky multipliers.

However, a special feature of the Bierman method as applied for factorization of covariance matrix is that this method can only be used for decomposition of symmetric covariance matrices.

But this inconvenience is resolved by using the Joseph form to calculate a posteriori covariance matrix of state variables:

$$P_k(+) = [I - K_k H_k] P_k(-) [I - K_k H_k]^T + K_k R_k K_k^T . \tag{3.16}$$

Though calculation of covariance matrices by means of the Joseph form (3.16) is more complicated and requires a greater amount of mathematical operations as compared with concatenation of matrices, such form has advantages as well. As it is shown in (Grewal and Andrews, 1993), Joseph form allows overcoming the problem of matrix $P_k(+)$ ill-conditioning making it a positive definite matrix, thus increasing robustness of the computational procedure.

Now, we will consider the Kalman filtration algorithm with factorization by the Bierman method. The EKF equation with the use of Joseph form is as follows:

$$K_k = P_k(-) H' \left(H P_k(-) H^T + R_k \right)^{-1} . \tag{3.17}$$

$$P_k(+) = [I - K_k H_k] P_k(-) [I - K_k H_k]^T + K_k R_k K_k^T . \tag{3.18}$$

$$P_{k+1}(-) = \Phi_k P_k(+)\Phi_k^T + Q_k, \tag{3.19}$$

Similarly to the previous filter, m_k, p_k, q_k, r_k are Cholesky multipliers of the corresponding matrices $P_k(-), P_k(+), Q_k, R_k$, i.e., $P_k(-) = m_k m_k^T$, $\mathrm{P}_k(+) = p_k p_k^T$, $\mathrm{Q}_k = q_k q_k^T$, $\mathrm{R}_k = r_k r_k^T$.

Then we can rearrange relations (3.17), (3.18), and (3.19) as

$$K_k = m_k m_k^T H^T \left(H m_k m_k^T H^T + r_k r_k^T\right)^{-1}. \tag{3.20}$$

$$p_k p_k^T = \left[I - K_k H_k\right] m_k m_k^T \left[I - K_k H_k\right]^T + K_k r_k r_k^T K_k^T. \tag{3.21}$$

$$m_{k+1} m_{k+1}^T = \Phi_k p_k p_k^T \Phi_k^T + q_k q_k^T, \tag{3.22}$$

Thus, given m_k, q_k, r_k, we will use the Bierman method and calculate the multiplier p_k from Eq. (3.21) and then find the multiplier m_{k+1} from Eq. (3.22).

3.1.4 Advantages and Disadvantages of Different Methods for Calculation of Cholesky Multipliers and their Application in the Kalman Filtration algorithms

QR factorization as applied to calculate the Cholesky multipliers is a universal and qualitative method. This method allows calculating the Cholesky multiplier for nonsymmetric matrices. Due to this feature of the QR factorization method, the expression for calculation of a priori and a posteriori matrices of state variables of the expanded Kalman filter was factored in the paper (Larin and Tunik, 2012) using the matrix concatenation procedure. Thus, expression was obtained to calculate the Cholesky multipliers of the a priori and a posteriori covariance matrices without calculating the covariance matrices (see subsection 3.1.2). Such simplification allows us to only deal with the Cholesky multipliers of covariance matrices with considerably larger condition numbers. Avoiding the procedure of calculation of covariance matrices increases the calculation accuracy as a whole and improves the convergence of calculations. However, such form of calculations (concatenation of matrices) increases matrix size, which increases the amount of mathematical operations in calculating the Cholesky multipliers.

A significant shortcoming of the QR factorization method is that it uses a considerable amount of mathematical operations to calculate the Cholesky multipliers:

$$E_{QORT} = \sum_{j=n-m}^{n}\left(5j^2 - j + 8\right) + 2n^2\left(n + m + 1\right)m. \tag{3.23}$$

The reason is that this method deals with vectors and matrices in calculating Cholesky multipliers.

To compare the efficiency and to estimate the computational complexity of the Bierman method and QR factorization method, we performed analytical and program calculation of the amount of mathematical operations used by these methods to calculate Cholesky multipliers.

Using analytical methods to find the computational complexity allowed making a preliminary estimation of the efficiency of the Bierman method for factorization of covariance matrices (Bierman, 1977). These methods resulted in the following analytical formula for finding the computational complexity of factorization by the Bierman method:

$$E_{UD} = \sum_{j=1}^{m}\left(m - j + \sum_{i=j}^{m} 2(m-i) \right) = \frac{2}{3}m^3 + \frac{1}{2}m^2 - \frac{7}{6}m, \tag{3.24}$$

where m is the rank of covariance matrix. To check the correctness of the analytical formula, we modeled the algorithm of the Bierman method for factorization of covariance matrix and calculated the amount of mathematical operations by means of *flops* function from Matlab software package. For modeling, we took a 9x9 symmetric positive definite matrix and decomposed it into Cholesky multipliers. As a result of modeling in Matlab 5 with the use of function *flops*, we obtained 497 operations. Calculating the amount of operations by the analytical formula (3.24) yielded 516 operations. These results are rather close; therefore, we can consider them reliable.

Similarly, we used mathematical methods to find the computational complexity and derived the analytical formula for the computational complexity of the QR factorization algorithm:

$$\begin{aligned} E_{QORT} &= \sum_{j=n-m}^{n}\left(5j^2 - j + 8\right) + 2n^2(n+m+1)m = \\ &= \frac{5}{3}n^3 + 2n^2 + \frac{25}{3}n - \frac{5}{3}(n-m)^3 - 2(n-m)^2 - \frac{25}{3}(n-m) + 2n^2 m(n+m+1), \end{aligned} \tag{3.25}$$

where n, m are dimensions of the covariance matrix. We performed similar modeling in the Matlab environment with the use of *flops* function for the algorithm with QR factorization. As a result of modeling and calculation by the analytical formula (3.25) with 9x9 symmetric positive definite matrix, we obtained the same number, namely, 29 154 operations.

Thus, having analyzed expressions (3.24) and (3.25) for calculation of the computational complexity of both methods, we may state that the Bierman method is more efficient since it needs less operations to calculate Cholesky multipliers of covariance matrices.

Let us compare the efficiency of the Bierman method with that of the QR factorization method used in the Kalman filtering problems with factorization of a priori and a posteriori covariance matrices of state variables in inertial-satellite navigation system. In the first case, we will consider the Kalman filter with QR factorization and calculate the amount of necessary mathematical operations. And then we will consider the Kalman filter with factorization by the Bierman method and perform a similar procedure.

The Kalman filter equations for the first variant are presented above by (3.6)–(3.8). Modeling the Kalman filtration algorithm with QR factorization and using *flops* function to calculate the amount of mathematical operations result in 292,267 operations.

Similarly, modeling the Kalman filtration algorithm with factorization by the Bierman method in Matlab software environment and using *flops* function to find the amount of mathematical operations, we obtain 22,799 operations.

To assess the factorization quality and find error accumulation in calculating the Cholesky multipliers, we will find the conditional numbers of a posteriori covariance matrix of state variables. To this end, let us model the algorithm of operation of an integrated navigation system in Matlab software environment and calculate conditional numbers using "cond" function. Modeling both algorithms of operation of the integrated navigation system yields the following results (Tunik and Valdenmayer, 2011):

- maximum value of the conditional number: 959,790 by the Bierman method and 959,820 by the QR factorization method;
- mean value of the conditional number: 209,320 by the Bierman method and 209,310 by the QR factorization method.

As we can see from the obtained results, the conditional numbers of Cholesky multipliers obtained by both methods are almost identical. This means that the values of factorization quality of both methods are proportional (Tunik and Valdenmayer, 2011).

However, using the QR factorization method has some advantages since this algorithm does not have a procedure of calculation of covariance matrices in renewal of Cholesky multipliers. This circumstance reduces error accumulation probability and improves the convergence of Kalman filtrations algorithm.

Example 3.1

Example of Finding the Attitude in INS and GPS Joint Operation

To illustrate the efficiency of the mechanization algorithms presented in Sec. §2 and correction algorithms described in the current subsection, we can use an elementary example of finding object's attitude in case of joint operation of INS and GPS. To this end, we will consider the following navigation problem. Choose a frame $Oxyz$ fixed to the Earth surface, with the axes being oriented to the north, east, and downwards (North, East, Down (NED) tangent plane navigation frame (Bronkhorst, 1978), (Schmidt, 1978)). In this frame, the object circles in the plane xy with period T = 300 sec and velocity 60 m/sec. During the motion, its attitude is defined by the following time dependence of Euler angles:

$$\psi = \frac{2\pi t}{T}, \qquad \vartheta = \frac{\pi}{2}, \qquad \varphi = 0{,}3\sin(10\psi)$$

Projections of the angular velocity of the object to axes of the body frame (without angular velocity of the Earth) are defined by Eqs. (2.8). These data are used to model readouts of rate gyros (RGs) mounted at the object. Namely, random numbers uniformly distributed with zero expectation and preset value of variance σ_ω are added to the values obtained according to (2.8). In integrating the kinematic equations (2.10), according to algorithm (2.15), quaternion λ_m^3 (2.27) was chosen as "elementary" one. Vectors of quasicoordinates $\nabla\theta_i$, necessary to find λ_m^3, were calculated with the use of quadratic spline (Simpson's formula). In other words, time dependence of the angular velocity vector was approximated on the time interval $[t_{i-2}, t_i]$ by a second-degree polynomial whose coefficients were calculated using “RG readouts” at times t_{i-2}, t_{i-1}, t_i.

Readouts of accelerometers, including measurement errors, which were assumed to be uniformly distributed random numbers with zero expectation and variance σ_a, were modeled similarly.

As the equation that describes variation in object’s coordinates in the NED system, we took Eq. (31) from (Bronkhorst, 1978), which reflects the theorem about addition of accelerations in a complicated motion of a point (the Coriolis theorem):

$$\frac{d\boldsymbol{V}}{dt} = \boldsymbol{a}_m + \bar{\boldsymbol{g}} - 2\Omega\times\boldsymbol{V}\,, \tag{3.26}$$

$$\tilde{\boldsymbol{a}} = \boldsymbol{a}_m - \bar{\boldsymbol{g}}, \qquad \bar{\boldsymbol{g}} = \boldsymbol{g} - \Omega\times\Omega\times\boldsymbol{R}\,,$$

In (3.26), which is duplication of (2.14), $\tilde{\boldsymbol{a}}$ - specific force, $\boldsymbol{a}_m$ - accelerometers readouts, transformed to navigation frame, $\boldsymbol{g}$ is the gravity acceleration, $\bar{\boldsymbol{g}}$ is the result of subtraction of centripetal Earth’s acceleration from gravity acceleration, $\boldsymbol{V}$ - ground speed, Ω is angular velocity of the Earth rotation, and $\boldsymbol{R}$ is position vector of a point in geocentric coordinate system.

Integration of (3.26) involved quadratic spline approximation of the variation of acceleration on the time interval $[t_{i-2}, t_{i-1}]$. It was considered earlier that the Coriolis’ acceleration was neglected. However it is possible to include the estimate of this acceleration $2\Omega\times V$ considering it as a small correction like it was done in (Bronkhorst, 1978) evaluating it at the *ith* step by value of velocity V at the *(i – 1)th* step. This has allowed us to simplify integration procedure of Eq. (3.26) using calculations in quadratures.

In the example, we assumed INS operation frequency to be 20 Hz, i.e., $\Delta t = 5 \cdot 10^{-2}$ sec in equation (3.2). The graphs in Figs. 3.2 – 3.4 give an idea about the accuracy characteristics of the described INS algorithm and show the results of modeling of INS operation in the absence of measurement noises ($\sigma_\omega = 0$, $\sigma_a = 0$). In these figures, solid line corresponds to axis x (attitude error (small turn $\delta\varphi$) with respect to axis x (Fig. 3.2), velocity errors (δV) and coordinate errors (δr) in the direction of axis x (Figs. 3.3, 3.4)), dashed line corresponds to axis y, and dot-and-dash line corresponds to axis z. Modeling was made with the use of MATLAB package. Noteworthy is the high accuracy of integration of the kinematic equations (2.10) when the quaternion λ_m^3 (2.27) was used (according to Fig. 3.2, the attitude error is of order 10^{-3} angular second).

Then we modeled the situation where sensor readouts were accompanied by noise ($\sigma_\omega = 1$ ang. min/sec, $\sigma_a = 10^{-2}$ m/sec^2) and corrected (according to the algorithms described above) the results of INS operation in 2 seconds. In (3.4) and (3.5) we assumed that $\Gamma_k = I$ and $D_k = I$. According to (Lachapelle, 1996), GPS system ensures the following errors of velocities and object coordinates: 0.1 m/sec, 50 m. Using these data and accepted values of σ_ω and σ_a, we have chosen the following values of Cholesky multipliers q_k and r_k (3.9):

$$q_k = 10^{-3} \cdot diag\{\alpha \cdot I, \beta \cdot I, \gamma \cdot I\}, \qquad r_k = diag\{0.1 \cdot I, 50 \cdot I\}.$$

$$\alpha = 0.0073, \qquad \beta = 0.25, \qquad \gamma = 0.0063.$$

The dimensions of identity matrices I in q_k and r_k are 3x3.

At the initial instant of time (t=0), we assumed errors of coordinates and velocities to be absent and attitude errors to be of order 2000 ang. sec along each axis (the (non-normalized) quaternion that defines the error of initial attitude has the form $[1 \quad 5 \cdot 10^{-3} \quad -5 \cdot 10^{-3} \quad 5 \cdot 10^{-3}]^T$). In this regard, the following matrix was taken as the Cholesky multiplier m_0:

$$m_0 = diag\{2 \cdot 10^{-2} \cdot I, 0, 0\},$$

where the identity matrix I and zero matrices 0 are 3x3.

To demonstrate the effect of correction of INS by GPS signals, correction was not performed on the time interval [300 sec, 600 sec]. Figures 3.5 – 3.8 show the results of modeling. The symbols in these figures are similar to those in Figs. 3.2 – 3.4. Figure 3.8 shows the variances (σ) of attitude errors calculated as square roots of the first three diagonal elements of matrix M_k (3.9).

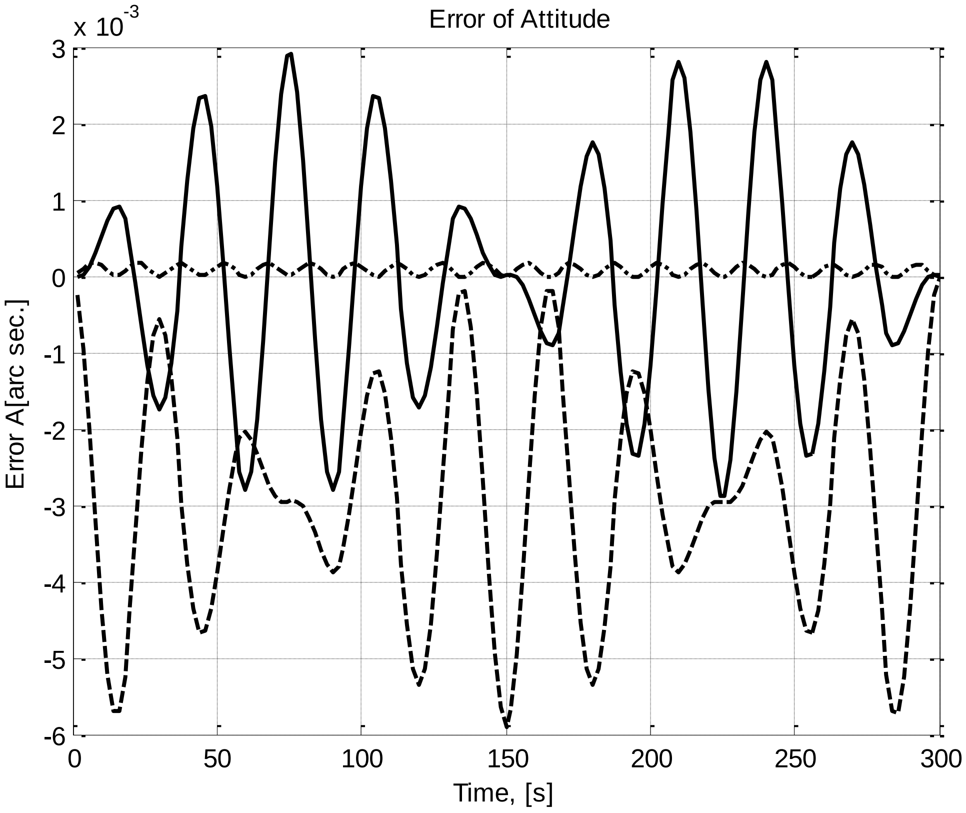


Figure 3.2 – Attitude error in the absence of measurement noise

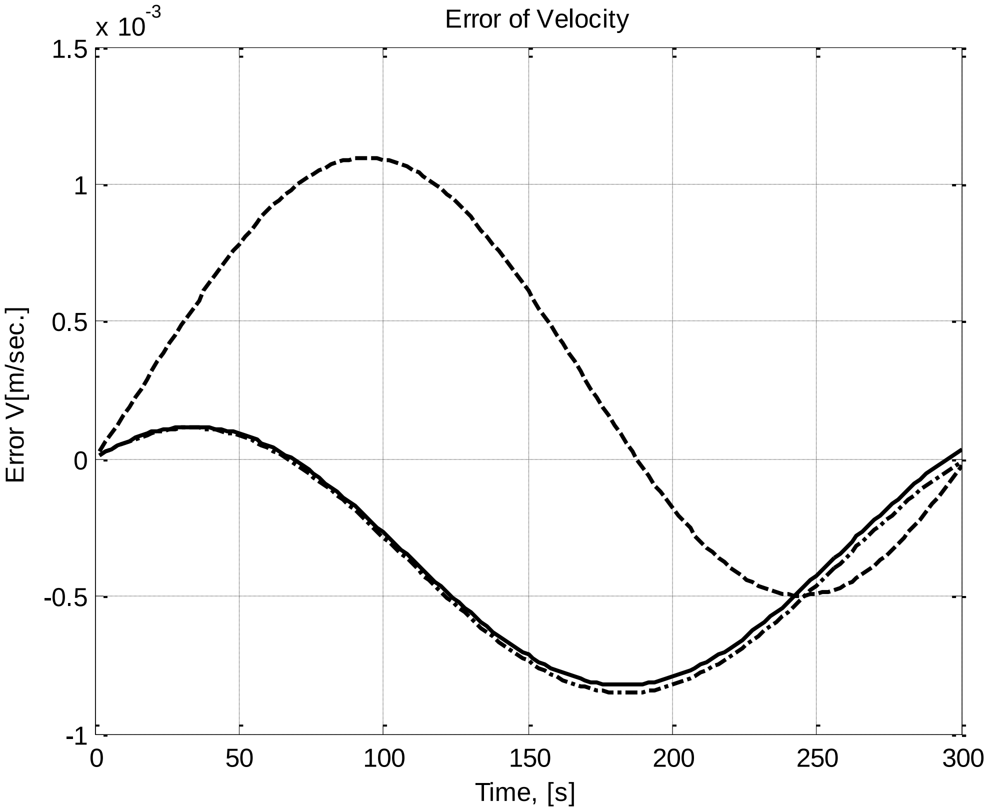


Figure 3.3 – Velocity error in the absence of measurement noise

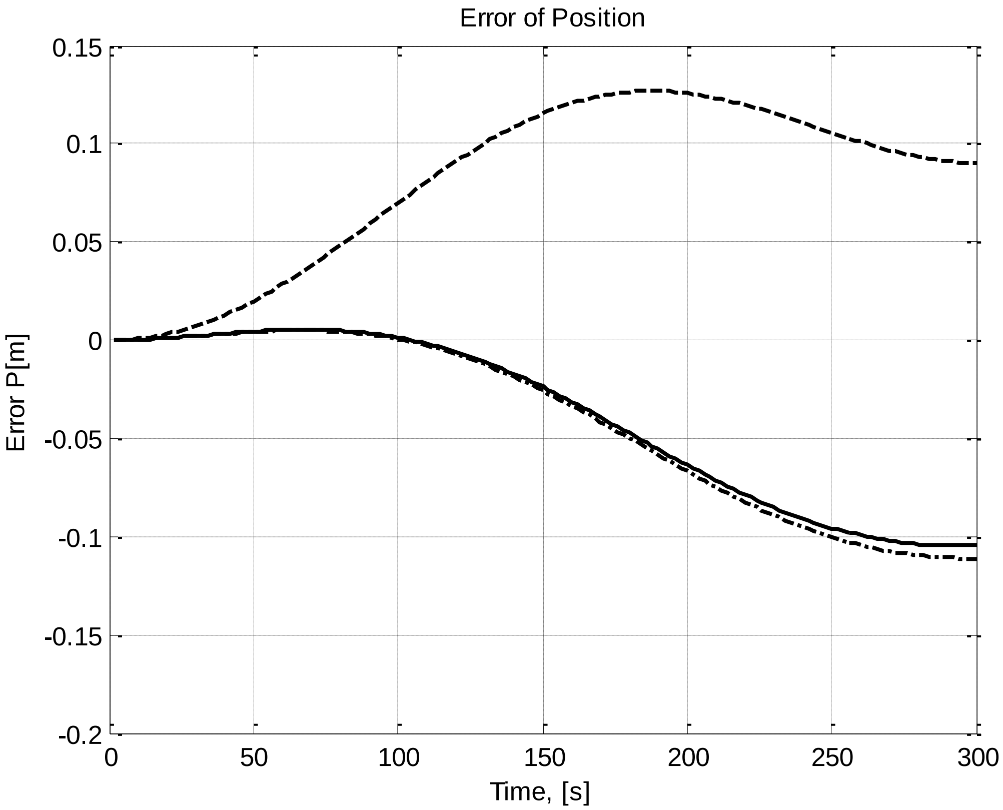


Figure 3.4 – Position error in the absence of measurement noise

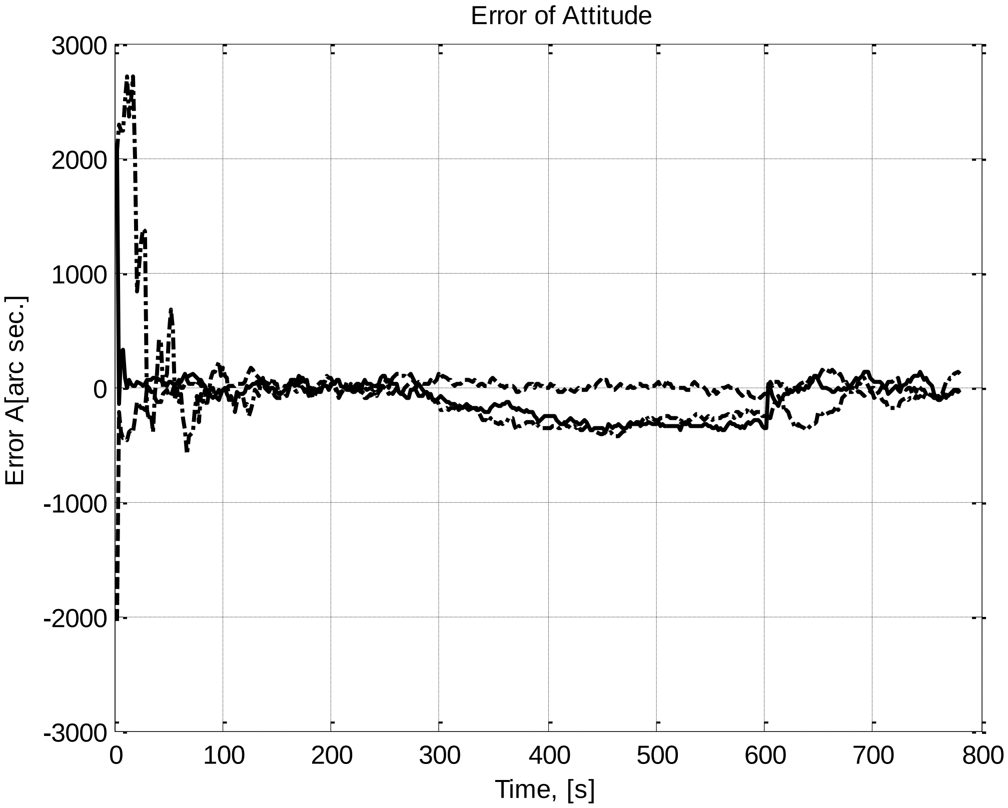


Figure 3.5 – Attitude error with measurement noises

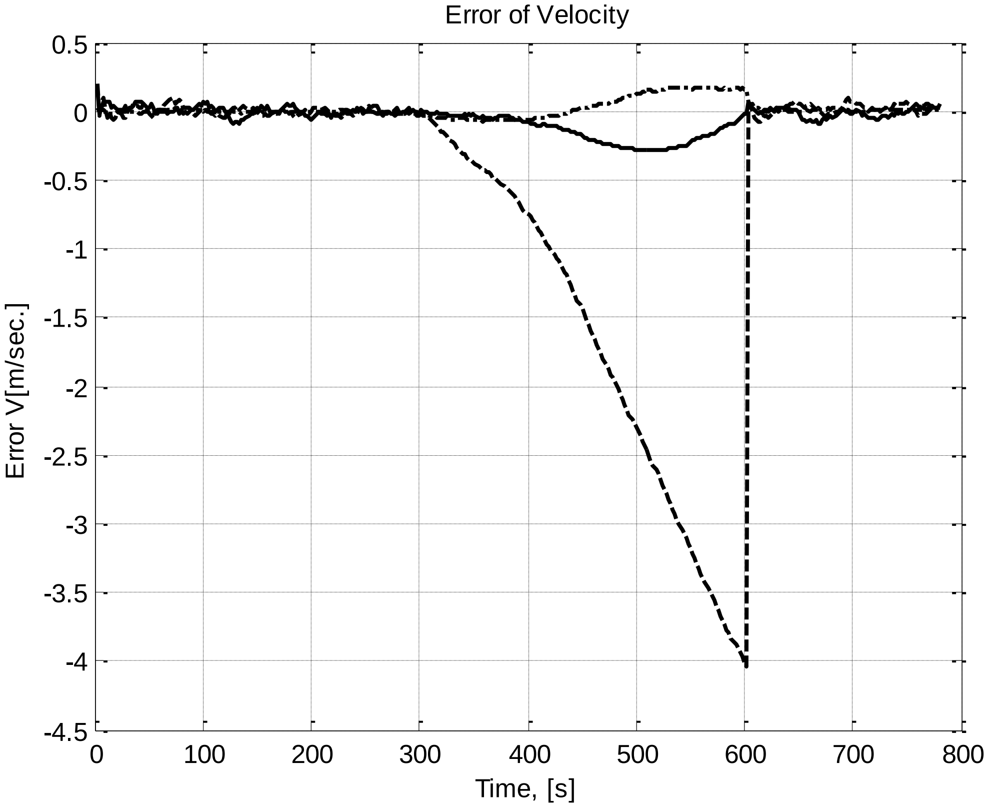


Figure 3.6 – Velocity error with measurement noises

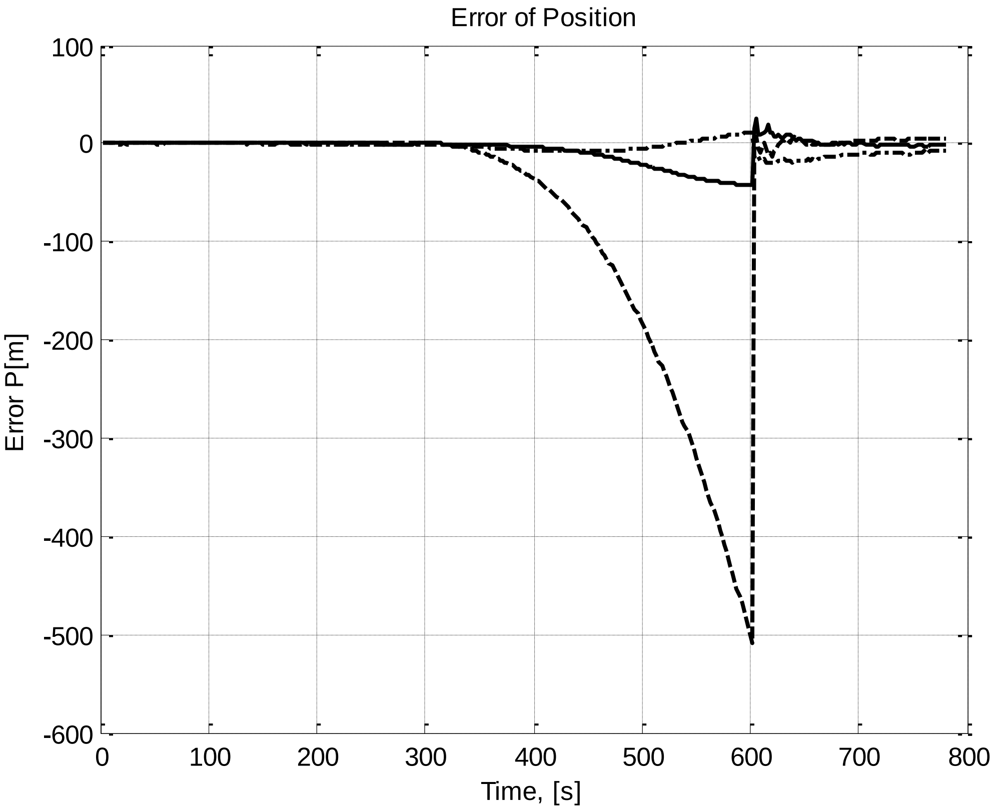


Figure 3.7 – Position error with measurement noises

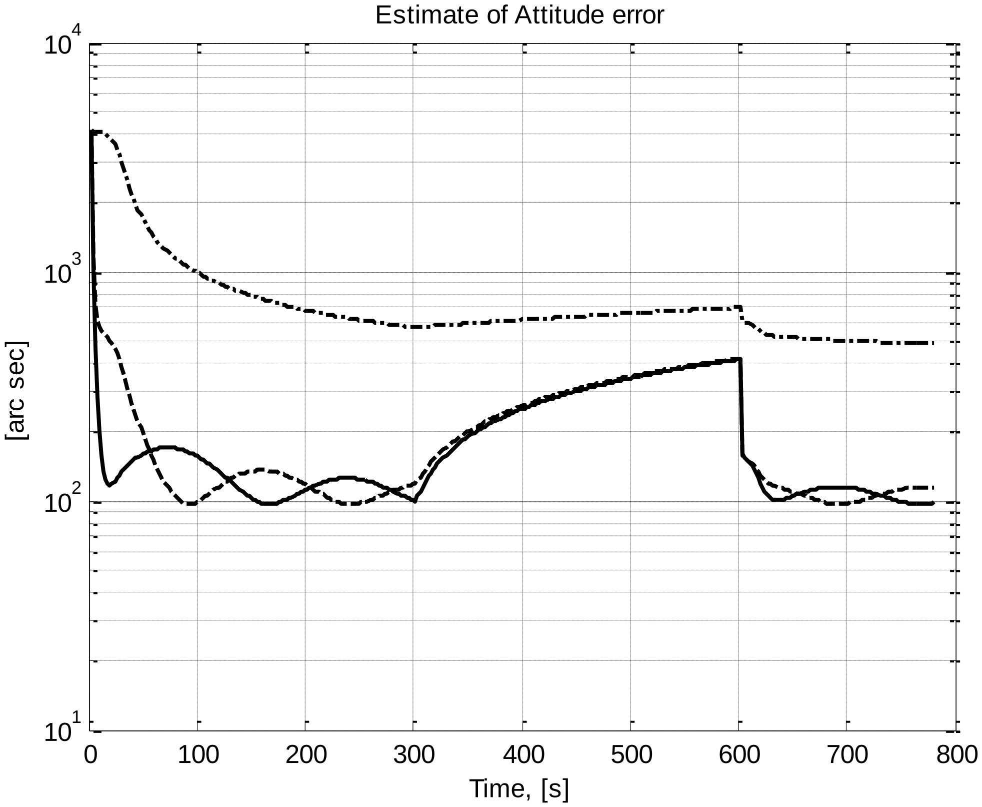


Figure 3.8 – The variances of attitude errors

3.2. Algorithms of Strapped Down Inertial Navigation Systems Correction based on Global Navigation Satellite System, Magnetometer, and Altimeter Signals

Subsection 3.1 describes the process of correction (relations (3.1) – (3.8)) of INS operation according to the information obtained from GPS. We will generalize the problem statement, assuming that along with GPS signals, readouts of magnetometer and altimeter can be used to adjust INS operation. Thus, along with statistical parameters of signals and measurement noise, the initial information for the correction algorithm is residual vector (ε_k), which can be calculated as the difference between GPS signals (vector z_k) and estimate of current values of coordinates and velocities of the object (vector $H \cdot \bar{x}_k$):

$$\varepsilon_k = z_k - H \cdot \bar{x}_k . \tag{3.27}$$

It is natural that generalization of the problem statement under study should be related to generalization of computation of the corresponding residual vector. For example, if measurement channels are supplemented by altimeter readouts, this generalization reduces to the corresponding expansion of vector z_k and matrix H in (3.27).

However, taking into account the readouts of magnetometer requires additional reasoning. To simplify the computations, in what follows we will consider (model) the information channel related to the magnetometer as follows. Assume that on the object, we measure a vector $(\overline{m})$ that in the local topocentric frame is a unit vector directed along the axis *ox* $(m = [1 \quad 0 \quad 0]^T)$. Thus, given the values of $(\overline{m})$ and estimate of the direction cosine matrix $\overline{A}$, we will find the estimate γ of the vector of small turn angle, which defines the object attitude error. To this end, we can use the following relation (see, for example (1.8) in (Larin and Naumenko, 1987)):

$$\overline{A}\overline{m} - m = m \times \gamma . \tag{3.28}$$

Relation (3.28) can be interpreted as formalization of the fact that the small turn vector γ turns vector m until it coincidences with vector $\overline{A}\overline{m}$. Multiplying both sides of relation (3.28) by $m\times$ yields the following expression for γ:

$$-\gamma = m \times (\overline{A}m - m). \tag{3.29}$$

As follows from the assumption about orthogonality of vectors m and γ, the first element of vector γ is zero and we can eliminate it from the consideration. The other components of vector γ can be interpreted as a result of measurement of two respective components of vector μ appearing in equation (3.1).

Thus, taking into account the remarks, in the case where information about readouts of magnetometer and altimeter is available along with GPS signals, the measurement process is as follows:

$$\boldsymbol{z}_k = \boldsymbol{H}\boldsymbol{x}_k + \xi_k; \quad \boldsymbol{H} = \begin{bmatrix} 0_{8\times1} & I_{8\times8} \\ 0_{1\times8} & 1 \end{bmatrix} \tag{3.30}$$

where $\boldsymbol{z}_k = [-\tilde{\gamma} \quad v_{GPS} \quad r_{GPS} \quad h]^T$ is measurement vector, where $\tilde{\gamma}$ is a 2×1 vector that consists of the two last components of vector $\bar{\gamma}$ obtained from magnetometer measurement, using relation (3.29), v_{GPS} is 3×1 velocity estimate obtained from GNSS, r_{GPS} is 3×1 coordinate estimate obtained from GNSS; $\boldsymbol{h}$ is height estimate obtained from barometric altimeter; $\boldsymbol{H}$ is measurement matrix, $\boldsymbol{x}_k$ is state vector, ξ_k is measurement error vector; and 0_{ixj} and I_{ixj} are zero and unity matrices of respective dimensions.

Let us continue considering the problem of INS correction based on magnetometer signals. When we deal with real magnetometer readouts, we actually obtain not the basis vector of X axis but some other vector dependent on the place and time of the measurement. The Earth's magnetic field at some arbitrary point is characterized by the magnetic field intensity vector, $H_e = [H_x, H_y, H_h]^T$ and two angles definig the attitude of this vector with respect to the geographic frame, namely: magnetic declination λ and inclination δ (Fig. 3.9).

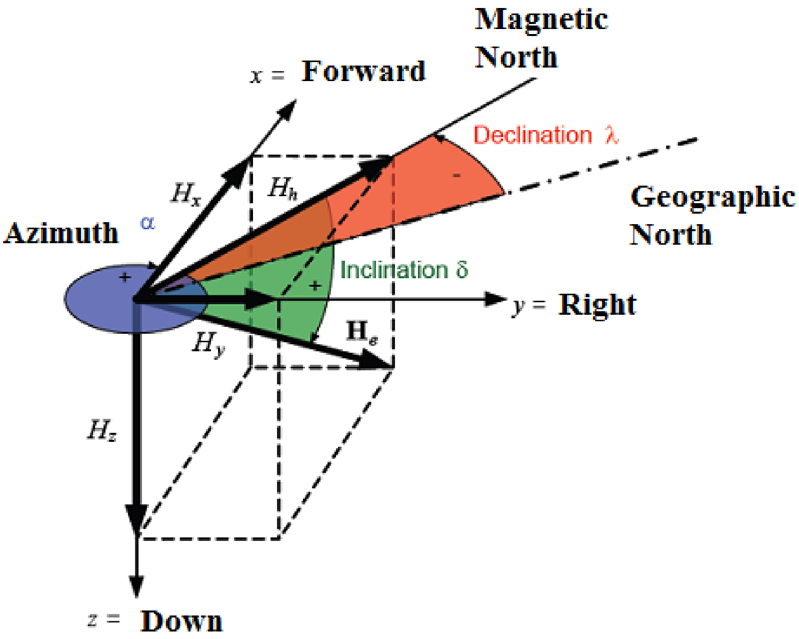


Figure 3.9 – Elements of Earth magnetic field intensity

There are geomagnetic maps of Earth, developed by specialized institutions such as the National Geophysical Data Center (NGDC, Boulder CO, USA) (now the National Centers for Environmental Information (NCEI)) and the British Geological Survey (BGS, Edinburgh, Scotland). In our study, we propose to use the World Magnetic Model (WMM), which is a joint product of the United States' National Geospatial-Intelligence Agency (NGA) and the United Kingdom's Defence Geographic Centre (DGC) and is the standard navigation model for the U.S. and U.K. Departments of Defense and NATO (Ngdc.noaa.gov, 2018), (Woods, 2018). There is also International Geomagnetic Reference Field (IGRF) model, which is the international research reference model (Thebault, 2014). Both of them allow calculating the geomagnetic field in the specified coordinates, taking into account sources inside the Earth, without regard for external currents.

This means that given the latitude, longitude, height, and date of the experiment with the use of the model, we obtain characteristics of the geomagnetic field. For example, for 30 April, 2018, for the point Latitude 50 º N, and Longitude – 30 º E, and zero elevation, magnetic declination is 7 º 26', inclination is 66 º 55', and the vector of magnetic field intensity was be the following: $\vec{\boldsymbol{H}}_e = \begin{bmatrix} H_x & H_y & H_z \end{bmatrix}^T = \begin{bmatrix} 19546.5 & 2554.2 & 46277.9 \end{bmatrix}^T nT$. Thus, the normalized vector of magnetic field intensity at this point has the following form:

$$\overrightarrow{m_\tau} = \frac{\vec{\boldsymbol{H}}_e}{\left\| \vec{\boldsymbol{H}}_e \right\|} = \left[m_x, m_y, m_z \right]^T = \begin{bmatrix} 0.3886 & 0.0508 & 0.9200 \end{bmatrix}^T \tag{3.31}$$

Based on the aforesaid, we will now remove the assumption that the magnetic field vector in the original frame is defined by the unit vector of axis *ox*, i.e., $\vec{m} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$. Let us show that under respective modification in this general case we can also use the algorithm presented above in this subsection.

Thus, let m_τ be a unit vector that defines the magnetic field intensity but does not coincide with the basis vector of axis *ox*. Let orthogonal matrix τ be such that

$$\tau \cdot \vec{m}_\tau = \vec{m} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T . \tag{3.32}$$

Here, the turn matrix τ is defined on the basis of the normalized vector of magnetic field intensity (3.31) as follows:

$$\boldsymbol{\tau} = \begin{bmatrix} m_x & m_y & m_z \\ -m_y & 1-\dfrac{m_y^2}{(1+m_x)} & -\dfrac{m_y m_z}{(1+m_x)} \\ -m_z & -\dfrac{m_y m_z}{(1+m_x)} & 1-\dfrac{m_z^2}{(1+m_x)} \end{bmatrix} \tag{3.33}$$

Let us substitute now the numerical values from (3.31) and check the validity of equation (3.32).

$$\boldsymbol{\tau} \cdot \vec{\boldsymbol{m}}_\tau = \begin{bmatrix} 0.3886 & 0.0508 & 0.9200 \\ -0.0508 & 0.9981 & -0.0337 \\ -0.9200 & -0.0337 & 0.3905 \end{bmatrix} \cdot \begin{bmatrix} 0.3886 \\ 0.0508 \\ 0.9200 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Magnetic inclination is derived from the vector of magnetic field intensity $\vec{\boldsymbol{H}}_e$ in the following way (Chulliat et.al., 2015), relation (19):

$$\delta = \operatorname{atan}\left(\frac{H_z}{\sqrt{H_x^2 + H_y^2}} \right)$$

It is necessary to take in account that the Magnetic North doesn’t coinside with the Geographic (or True) North, and the magnetic declination λ varies according to different locations. Therefore we should compensate it accordingly, when using magnetometer measurements for navigation. The National Geospatial-Intelligence Agency (NGA) (Ngdc.noaa.gov, 2018) provides the source code written in C that is based on the World Magnetic Model (WMM) and provides the Earth magnetic field intensity, together with inclination and declination angles. This correction could be easily performed, that is why we don’t consider it for the sake of simplicity.

In this case, it is expedient to consider the first three components of vector $x_k = [\mu \quad \delta v \quad \delta r]^T$, which define the small turn vector μ, in the frame defined by matrix $\boldsymbol{\tau}$ that appears in (3.32). In other words, it is necessary to introduce a small turn vector $\overline{\mu}$, which is related to vector μ as follows:

$$\overline{\mu} = \tau \cdot \mu. \tag{3.34}$$

Thereupon, it is necessary to subject matrix Φ_k appearing in (3.2) to the following linear transformation:

$$\overline{\Phi}_k = \theta \Phi_k \theta^T, \qquad \theta = diag\{\tau, \quad I, \quad I\}. \tag{3.35}$$

Matrix $\overline{\Phi}_k$ thus obtained should be used in the relations (3.2, 3.4, 3.5 and 3.7) in Subsection 3.1.

Let us consider modifications that should be introduced in the procedure described above in this subsection. An analog of relation (3.28) in the case under study is

$$\overline{A}\overline{m}_\tau - m_\tau = m_\tau \times \gamma_\tau . \tag{3.36}$$

In (3.36), $\overline{m}_\tau$ is the result of measurement of magnetic field vector in the moving frame and γ_τ is the corresponding vector of small turn. Multiplying (3.36) by τ yields

$$\tau \cdot \overline{A} \cdot \overline{m}_\tau - m = m \times \bar{\gamma}, \quad \bar{\gamma} = \tau \cdot \gamma_\tau . \tag{3.37}$$

From (3.37), an analog of relation (3.28) follows:

$$-\bar{\gamma} = m \times \tau \cdot \overline{A} \cdot \overline{m}_\tau . \tag{3.38}$$

Since vector γ is orthogonal to vector m, the first component of vector $\bar{\gamma}$ is zero. Thereupon, vector $\tilde{\gamma}$ appearing in (3.30) will only have two components, which coincide with the last two components of vector $\bar{\gamma}$. Thus, in the general case under study, vector z_k in equation (3.30) is defined.

Then, estimate of vector x_k obtained according to (3.4) should be multiplied by matrix θ^T. This is because the first three components of vector x_k correspond to vector $\overline{\mu}$ related to the small turn vector μ by relation (3.34) in the original frame.

Example 3.2

Example of INS Correction by Means of GPS, Magnetometer, and Altimeter

Let us illustrate the described algorithm of INS correction by means of GPS, magnetometer, and altimeter. In this connection, let us consider an example similar to that in Sec. 10 (Larin, 2001). Assume that xyz frame represented in in Fig. 2.2 (Chapter 2) is oriented as follows: axis *x* to South, axis *y* to East, and axis *z* to Zenith. The origin of coordinates of this frame (point *O*) is located on the Earth surface at 45^0 north latitude. In this frame, the object circles in *xy* plane with period $T = 300\,s$ and velocity $V = 60\,m/s$. During the motion, its attitude is described by the following time dependence of Euler angles: $\psi = \frac{2\pi t}{T}$, $\vartheta = 0$, $\varphi = 0{,}3\sin(10\psi)$. Projections of the angular velocity of the object onto the axes of body frame

(without regard for the angular velocity of the Earth) are defined by Eqs. (5). These data are used to model readouts of RGs mounted at the object. Namely, random numbers, which are uniformly distributed with zero expectation and variance σ_ω, are added to the values obtained according to (2.8). In integrating the kinematic equations (2.10), according to algorithm (2.15), quaternion (2.18) is chosen as "elementary" one. The necessary vectors of quasicoordinates $\nabla\theta_i$ were calculated according to (2.19).

Errors of accelerometers were assumed to be uniformly distributed random numbers with zero expectation and variance σ_a.

In integrating Eq. (2.14), relations (2.20) and (2.21) were used (Coriolis acceleration was not taken into account).

Errors of magnetometer and altimeter readouts were modeled similarly. For magnetometer, errors were assumed to have identical variance σ_m for each of the coordinates. Readouts of altimeter are accompanied by errors with variance σ_v.

Like in (Larin, 2001), INS is supposed to operate with frequency of 20 Hz, i.e., time interval is $\Delta t = 5\cdot 10^{-2}\sec$. Readouts of RGs and accelerometers are accompanied by noise ($\sigma_\omega = 3$ ang. min/sec, $\sigma_a = 10^{-2}\ m/\sec^2$), INS is corrected every 2 sec. Since we only consider the navigation problem, it is assumed that matrices Γ_k and D_k in (3.4) and (3.5) are zero. As well as in (Larin, 2001), GPS is supposed to ensure the following errors of velocity and coordinates of the object: 0.1 m/sec, 50 m. The variance of magnetometer (finding the coordinates of vector $\overline{m}$) and altimeter errors are assumed as follows: $\sigma_m = 0.0524$, $\sigma_v = 1m$. Based on these data, the following values for Cholesky multipliers q_k, η_k were accepted:

$$\bar{q}_k = 10^{-3}diag\{\alpha_q\cdot I,\ \beta_q\cdot I, \gamma_q\cdot I\},\qquad \bar{\eta}_k = diag\{\alpha_\eta\cdot I_2,\ \beta_\eta\cdot I, \gamma_\eta\cdot I, 1\} \tag{3.39}$$

$\alpha_q = 0.0218, \beta_q = 0.25,\ \gamma_q = 0.0063,\ \alpha_\eta = 0.0524, \beta_\eta = 0.1,\ \gamma_\eta = 50.$ In these expressions, matrix I is a 3x3, matrix I_2 is 2x2. We suppose that $g = 9.81\,m/\sec^2$ and at the initial instant of time $(t = 0)$ the object is located on axis y, at distance $\frac{VT}{2\pi}$, i.e., the initial position of the object is described by vector $r_0 = \begin{bmatrix} 0 & \frac{VT}{2\pi} & 0 \end{bmatrix}^T$. The velocity vector of the object has the form $v_0 = \begin{bmatrix} -V & 0 & 0 \end{bmatrix}^T$. The initial INS set up is accompanied by the following errors. The error of initial attitude is determined by the (non-normalized) quaternion: $\lambda(0) = [1\quad 0{,}05\quad -0{,}05\quad 0.05]^T$, the error of set up of initial values is determined by the relative errors $\varepsilon_r = 0.01$ and $\varepsilon_v = 0.01$, i.e., the following values of initial coordinates (r) and velocity (v) are assigned:

$$r = r_0(1+\varepsilon_r)\,, \quad v = v_0(1+\varepsilon_v)\,.$$

In this connection, the following matrix is taken as the Cholesky multiplier m_0:

$$m_0 = diag\{\alpha_m I,\ \beta_m I, \gamma_m I\},$$

$$\alpha_m = 0.02,\ \beta_m = 1.2,\ \gamma_m = 57.3.$$

As well as in (3.39), matrices I are 3x3. Noteworthy is that the observation vector z_k and measurement matrix H are formed according to equation (3.30).

For such initial data, INS operation during 500 sec was modeled. On the time interval $[0, 60]$sec, INS operated in autonomous mode, i.e., no correction of its operation was made. On the interval $[60, 180]$sec, INS operation was corrected using signals of GPS, magnetometer, and altimeter. Further, on the interval $[180, 420]$sec, information from magnetometer and altimeter was only used for correction of INS operation. At the final stage, $t \in [420, 500]$sec, INS operation was corrected using signals of GPS, magnetometer, and altimeter. The results of modeling (time dependence of the components of vector x appearing in equation (3.1)) are presented in Figs. 3.10 – 3.15.

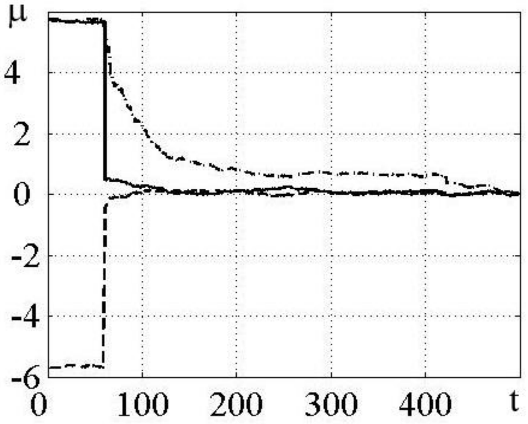


Figure 3.10 – Attitude error

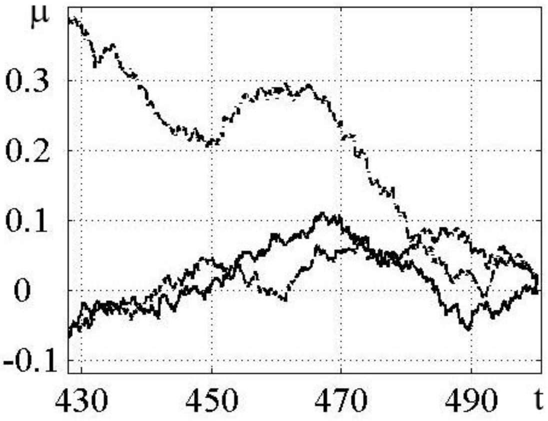
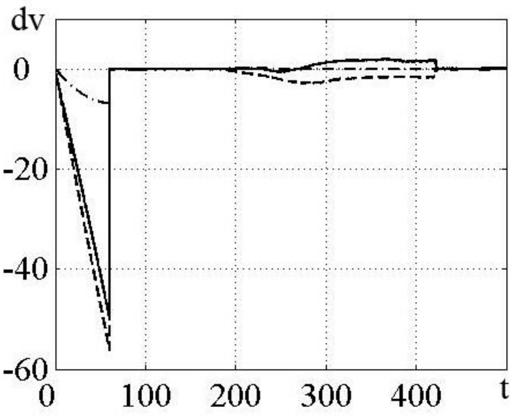


Figure 3.11 – Scaled attitude error



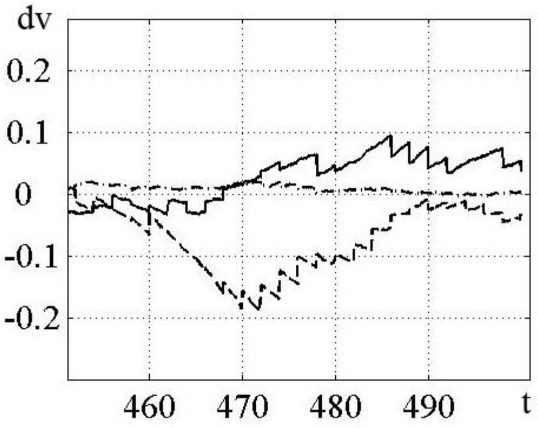


Figure 3.12 – Velocity error

Figure 3.13 – Scaled velocity error

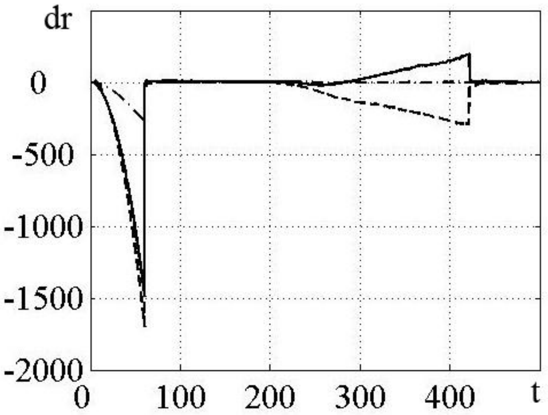


Figure 3.14 – Position error

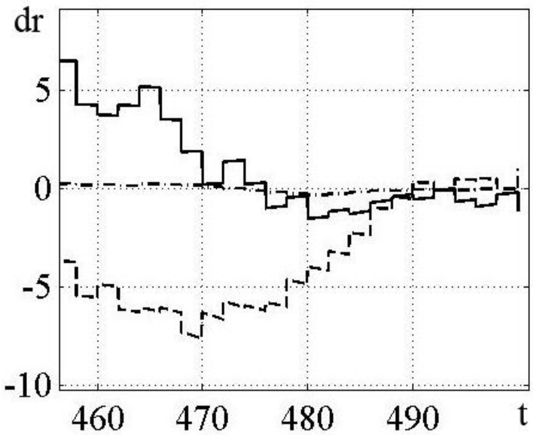


Figure 3.15 – Scaled position error

Figure 3.10 shows time dependence of components of vector μ, (μ_x, μ_y, μ_z), degrees dimension). Figures 3.12 and 3.14 show components of vectors $\delta v, \delta r$ (dimensions m/sec and m, respectively). To estimate the accuracy characteristics of the integrated system under steady operation, Figs. 3.11, 3.13, 3.15 present (with rescaling along the vertical axis) fragments of Figs. 3.10, 3.12, 3.14, respectively. In these figures, the notation is as follows for coordinates of vectors $\mu, \delta v, \delta r$: the solid line corresponds to axis x, dashed line to axis y, and dot-and-dash line to axis z.

As follows from these figures, the algorithm proposed for INS correction is rather efficient, in particular, it successfully operates under considerable perturbations. Errors at the steady mode (as follows from Figs. 3.11, 3.13, 3.15) have the following order: the error of attitude estimate is of shares of degree, velocity error is ~ 0.2 m/sec, and coordinate error is less than one meter.

We have considered a simple algorithm of integration of inertial navigation system, satellite navigation system, magnetometer, and barometric altimeter. A number of simplifying assumptions is introduced. This is because, on the one hand, gauges in the system under study are not high-precision, and on the other hand, such systems are supposed to be used for objects that move with low speed and at rather small distances. In particular, these can be cheap unmanned aerial vehicles. To illustrate the capabilities of such system, an example is considered.

3.3. Strapped Down Inertial Navigation Systems Correction Algorithm Taking into Account Systematic Errors of Rate Gyros

Let us supplement the Kalman filtration algorithm described in the previous subsection (3.2) by including calculation of systematic errors of rate gyros (RGs). Input data of the correction block are the same data about position, velocity, and attitude from SINS, data about position and velocity from GNSS, covariance SINS error matrix, covariance GNSS error matrix, and measurements sensitivity matrix. Output data are optimal estimates of UAV position, velocity, and attitude and systematic errors.

Let us write the corresponding equations of the Kalman filter. Denote by $\mu, \delta v$ and δr vectors of INS errors in the same frame where Eq. (2.14) is written (μ is the vector of small turn of attitude error and δv and δr are vectors of the errors of object's velocity and coordinates). Let δc be vector of systematic RG error, $\tilde{a} = [\tilde{a}_1, \tilde{a}_2, \tilde{a}_3]^T$ be vector of full acceleration, and cosine matrix be defined by (2.5). We accept the equation of INS errors propagation in the form similar to (7.149) in (Grewal and Andrews, 1993), i.e.,

$$\dot{x} = Fx + n, \tag{3.40}$$

$$\boldsymbol{x} = \begin{bmatrix} \mu \\ \delta \boldsymbol{v} \\ \delta \boldsymbol{r} \\ \delta c \end{bmatrix} \quad F = \begin{bmatrix} 0 & 0 & 0 & A^T \\ C & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} 0 & -\tilde{a}_3 & \tilde{a}_2 \\ \tilde{a}_3 & 0 & -\tilde{a}_1 \\ -\tilde{a}_2 & \tilde{a}_1 & 0 \end{bmatrix},$$

n is the vector of white noise. Hereinafter, 0 is zero matrix of respective size and is identity matrix of respective dimension.

As a discrete analog of (3.40) (i.e., relation that expresses variation of errors on a small time interval Δt), we take the following equation:

$$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \boldsymbol{n}_k; \quad \Phi_k = I + F\Delta t + \frac{(\Delta t)^2}{2} F^2 = \begin{bmatrix} I & 0 & 0 & A^T \Delta t \\ C\Delta t & I & 0 & CA^T \dfrac{(\Delta t)^2}{2} \\ C\dfrac{(\Delta t)^2}{2} & I\Delta t & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix}; \tag{3.41}$$

$\boldsymbol{n}_k$ is the vector of random errors of INS operation. The subscript k corresponds to the instant of time $k \cdot \Delta t$. We may suppose that Δt is a time step of INS operation, and the initial equation of errors is (3.41).

Assume that at the k th step of INS operation, GPS provides information about the estimate of coordinates and velocity of the object, i.e., the following observation process takes place :

$$\boldsymbol{z}_k = \boldsymbol{H}\boldsymbol{x}_k + \xi_k; \quad \boldsymbol{H} = \begin{bmatrix} 0_{3x3} & I_{3x3} & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & I_{3x3} & 0_{3x3} \end{bmatrix}; \tag{3.42}$$

ξ_k is measurement error. It might be seen that equation (3.42) is an extended version of the equation (3.3).

If information about the readouts of magnetometer and altimeter is available along with GPS signals, the measurement process is as follows:

$$\mathrm{z}_k = \mathrm{Hx}_k + \xi_k; \quad \mathrm{H} = \begin{bmatrix} 0_{8\times1} & I_{8\times8} & 0_{8\times3} \\ 0_{1\times8} & 1 & 0_{1\times3} \end{bmatrix} \tag{3.43}$$

where $\boldsymbol{z}_k = \begin{bmatrix} -\tilde{\gamma} & v_{GPS} & r_{GPS} & h \end{bmatrix}^T$ is measurement vector, where $\tilde{\gamma}$ is a 2×1 vector that consist of the two last components of vector $\bar{\gamma}$ obtained from the magnetometer, v_{GPS} is 3×1 velocity estimate obtained from GNSS, r_{GPS} is 3×1 coordinate estimate obtained from GNSS; h is height estimate obtained from barometric altimeter; $\boldsymbol{H}$ is measurement matrix, $\boldsymbol{x}_k$ is state vector, ξ_k is measurement error vector; and and are zero and unity matrices of respective sizes. It might be seen as well that the equation (3.43) is an extended version of the equation (3.30).

Thus, using relations (3.3), (3.30), (3.42) or (3.43), we can formulate the problem of INS correction as an optimal filtration problem.

It is generally known (see, for example, item 12.4 in (Bryson, and Ho-Yu-Chi, 1969)) that the solution of this problem has the form

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{z}_k - \boldsymbol{H}\bar{\boldsymbol{x}}_k\right), \quad \bar{\boldsymbol{x}}_{k+1} = \Phi_k \hat{\boldsymbol{x}}_k. \tag{3.44}$$

The filter gain matrix ($\boldsymbol{K}_k$) that generates optimal estimate vector $\boldsymbol{x}_k$ is defined in the same way as in Sec. 3.1, where filter equations are described by (3.6)–(3.8). Note that INS error correction usually takes place in steps. In this case, on the intervals between the corrections, INS errors are modified according to Eq. (3.2), and their correlation matrix is modified according to (3.6) (we may suppose that at these steps H = 0). At the step at which correction takes place, modifications of correlation matrix are described by Eq. (3.7).

Thus, the first nine components of vector $\bar{\boldsymbol{x}}_k$ in (3.44) determine estimates of error vectors $\mu_k, \delta\boldsymbol{v}_k, \delta\boldsymbol{r}_k$, and hence estimates of attitude, velocity and coordinates of the object at instant of time t_k. As to the last three components of vector $\bar{\boldsymbol{x}}_k$ (vector δc_k), noteworthy is that these components determine estimates of systematic error of RG readouts. It is expedient to use this estimate for correction of RG readouts. For example, if vector $\bar{\omega}(t_k)$ is the output signal of RGs at time t_k, then it is necessary to use the following value of angular velocity vector in relations (2.19):

$$\omega(t_k) = \bar{\omega}(t_k) - \delta c_k \tag{3.45}$$

Let us emphasize that correction δc_k in (3.45) varies only at the instant of time of INS correction by GPS data, i.e., when $H \neq 0$.

Example 3.3

Example of INS Correction by Means of GPS, Magnetometer, and Altimeter

Let us illustrate the algorithm of INS correction by means of GPS, magnetometer, and altimeter described above. To illustrate the influence of systematic RG error, we will use the example from (Larin and Tunik, 2010 b) to compare the results of modeling of GPS/INS operation in case of the algorithm described above and of the algorithm from (Larin and Tunik, 2010 b) where systematic RG errors are not taken into account. The value of systematic RG error is assumed the same as in the example from (Ahn et.al, 2003). Thus, let us consider an example similar to the example from (Larin and Tunik, 2010 b). Assume that the frame Oxyz represented in Fig. 2.2 (Chapter 2) is oriented as follows: axis x is south, axis y is east, and axis z is zenith. The origin of coordinates (point O) is located at the Earth surface at $45°$ northern latitude. In this frame, the object circles in the plane xy with period $T = 300\,s$ and velocity $V = 60\,m/s$. During the motion, its attitude is described by the following time dependence of Euler angles: $\psi = \frac{2\pi t}{T}$, $\vartheta = 0$, $\varphi = 0{,}3\sin(10\psi)$. Projections of the angular velocity of the object to the axes of the body frame $(x'y'z')$ (without regard for the angular velocity of the Earth) are defined by Eqs. (2.8). These data are used to model the observations of RGs mounted at the object. Namely, the vector of systematic error $n_c = [\sigma_1\ \sigma_2\ \sigma_3]^T$ and the 3×1 vector n_ω that models random errors are added at time t_k to the angular velocity vector $n_\omega = [\omega_1\ \omega_2\ \omega_3]^T$ obtained according to (2.8). Elements of vector n_ω are random numbers uniformly distributed with zero expectation and variance σ_ω. When integrating the kinematic equations (2.10), according to algorithm (2.15), quaternion (2.18) was taken as "elementary" one. The necessary vectors of quasicoordinates $\nabla\theta_i$ were calculated according to (2.19). Accelerometer errors were assumed to be uniformly distributed random numbers with zero expectation and variance σ_a.

In the integration of Eq. (2.14), relations (2.20) and (2.21) were used (the Coriolis acceleration was not taken into account).

Magnetometer and altimeter errors are modeled similarly. For magnetometer, we supposed that errors have identical variance σ_m for each of the coordinates. Altimeter readouts are accompanied by errors with variance σ_v.

Like in (Larin and Tunik, 2010 b), we assume that INS operates with frequency 20 Hz, i.e., time interval is $\Delta t = 5 \cdot 10^{-2}\sec$. Readouts of RGs and accelerometers are accompanied by noise ($\sigma_\omega = 3$ ang. min/sec, $\sigma_a = 10^{-2}\ m/\sec^2$). Like in (Ahn et.al, 2003), the values of components of vector n_c are as follows: $\sigma_1 = 94$ ang.sec/sec, $\sigma_2 = -56$ ang.sec/sec, and $\sigma_3 = 22$ ang.sec/sec. Note that the accepted values of systematic error are less than σ_ω. INS operation is corrected every 2 sec. Like in (Larin and Tunik, 2010 b), we assume that GPS will ensure the following errors of object's velocity and coordinates: 0.1 m/sec and 50 m. The variance of the errors of magnetometer (determining the coordinates of vector $\overline{m}$) and of altimeter are taken as follows: $\sigma_m = 0.0524, \sigma_v = 1\,m$. Guided by these data, we accepted the following values for the Cholesky multipliers q_k, η_k:

$$\overline{q}_k = 10^{-3} diag\{\alpha_q \cdot I,\ \beta_q \cdot I, \gamma_q \cdot I, \delta_q \cdot I\}, \quad \overline{\eta}_k = diag\{\alpha_\eta \cdot I_2,\ \beta_\eta \cdot I, \gamma_\eta \cdot I, 1\}, \quad (3.46)$$

$\alpha_q = 0.0218, \beta_q = 0.25,\ \gamma_q = 0.0063, \delta_q = 5 \cdot 10^{-5},\ \alpha_\eta = 0.0524, \beta_\eta = 0.1,\ \gamma_\eta = 50.$ In these expressions, matrix dimensions are 3×3 for I and 2×2 for I_2. We assume that $g = 9.81\,m/\sec^2$ and at the initial instant of time $(t = 0)$ the object is located on axis y, at distance $\frac{VT}{2\pi}$, i.e., the initial position of the object is described by vector $r_0 = \begin{bmatrix} 0 & \frac{VT}{2\pi} & 0 \end{bmatrix}^T$. The velocity vector of the object has the form $v_0 = [-V \quad 0 \quad 0]^T$. The initial INS set up is accompanied by the following errors. The error of initial attitude is determined by the (non-normalized) quaternion: $\lambda(0) = [1 \quad 0{,}05 \quad -0{,}05 \quad 0.05]^T$, the error of set up of initial values is determined by the relative errors $\varepsilon_r = 0.01$ and $\varepsilon_v = 0.01$, i.e., the following values of initial coordinates (r) and velocity (v) are assigned: $r = r_0(1+\varepsilon_r)$, $\quad v = v_0(1+\varepsilon_v)$.

Thereupon, the following matrix is taken as the Cholesky multiplier m_0:

$$m_0 = diag\{\alpha_m \cdot I,\ \beta_m \cdot I, \gamma_m \cdot I, \delta_m \cdot I\};\ \alpha_m = 0.04, \beta_m = 0.6,\ \gamma_m = 28.65, \delta_m = 0.4.$$

Here matrix I has size $[3\times3]$. The observation vector z_k and $[9\times12]$ measurement matrix has the structure as in eq. (3.43).

Under such initial data, we modeled INS operation during 250 sec. The results of modeling (time dependence of the first nine components of vector x that appears in (3.1)) are presented in Figs. 3.16, 3.18, 3.20.

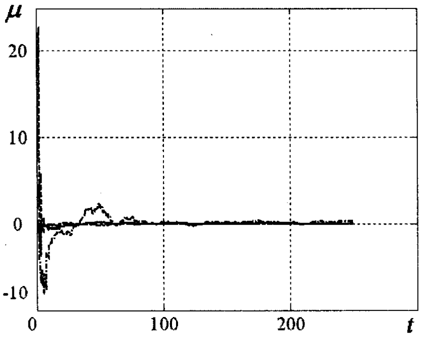


Figure 3.16 – Attitude error. Algorithm compensating systematic errors of RG

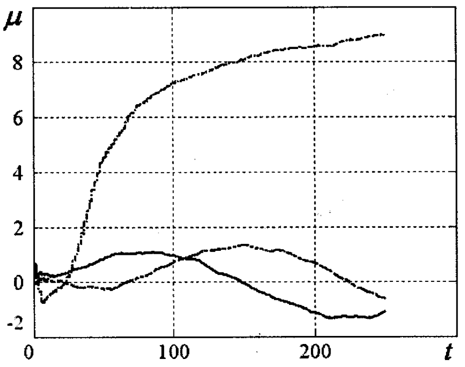


Figure 3.17 – Attitude error. Algorithm disregarding systematic errors of RG

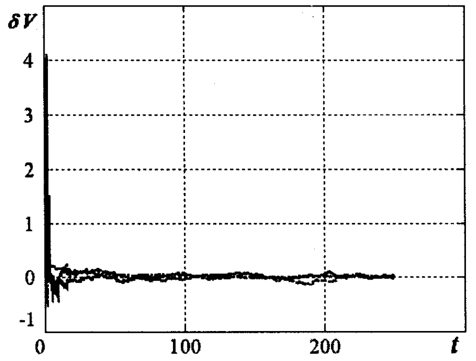


Figure 3.18 – Velocity error. Algorithm compensating systematic errors of RG

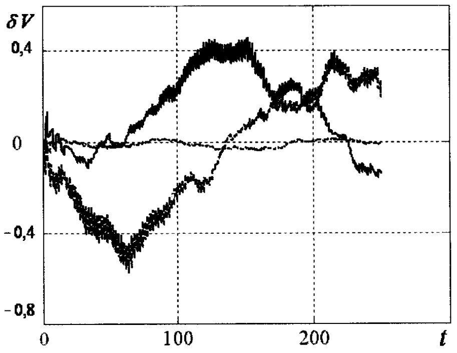


Figure 3.19 – Velocity error. Algorithm disregarding systematic errors of RG

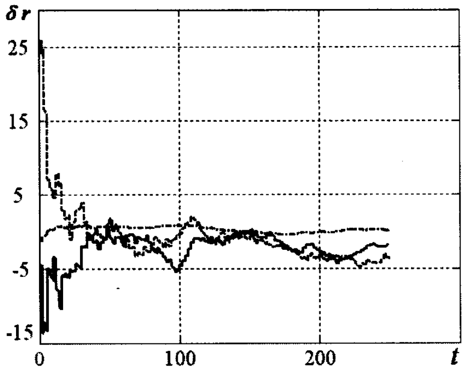


Figure 3.20 – Position error. Algorithm compensating systematic errors of RG

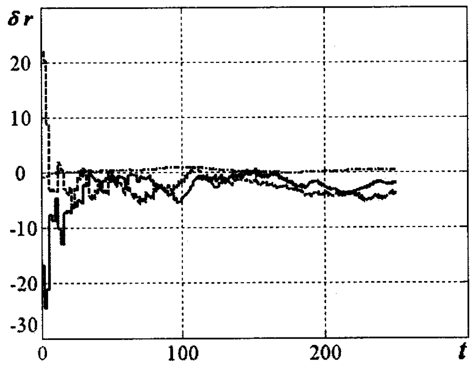


Figure 3.21 – Position error. Algorithm disregarding systematic errors of RG

Figure 3.16 shows the time dependence of components of vector $\mu(\mu_x, \mu_y, \mu_z)$, degrees dimension), and Figs. 3.18, 3.20 show components of vectors $\delta v, \delta r$ (dimensions m/sec and m, respectively).

To estimate the influence of systematic RG error in Figs. 3.17, 3.19, 3.21, similar modeling results are presented when the algorithm (Larin and Tunik, 2010 b) is used that disregards the systematic error. The following notation is introduced in Figs. 3.16 – 3.21 for coordinates of vectors $\mu, \delta v, \delta r$: the solid line corresponds to axis x, dashed line to axis y, and dash-and-dot line to axis z.

We can assess the efficiency of compensation for the systematic RG error in Fig. 3.22, where the time dependence $\alpha = \log\left(\left\|n_c - \delta c\right\| / \left\|n_c\right\|\right)$ is shown, δc is current value of the estimate of systematic RG error, and $\|\cdot\|$ denotes the norm of a vector.

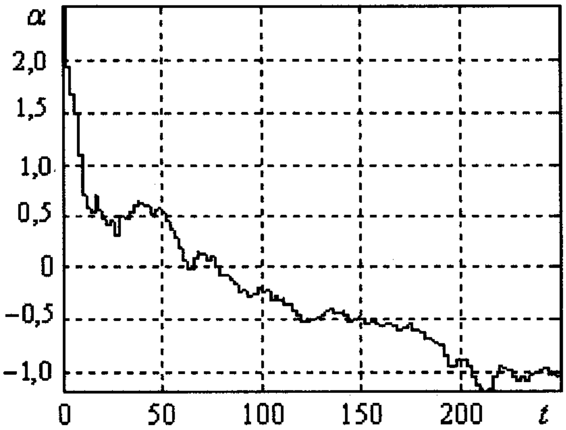


Figure 3.22 – Characteristics of the compensation efficiency

As follows from the results in Figs. 3.16, 3.18, 3.20 the proposed algorithm of INS correction is rather efficient in case of systematic RG errors compensation as well. For example, after rather short (2–3 steps of GPS correction) transient caused by robust estimation of systematic RG errors by the system, which ensures rather high accuracy of the estimate of attitude, velocity, and coordinate parameters of the object.

Comparing the data in Figs. 3.16, 3.18, 3.20 and Figs. 3.17, 3.19, 3.21, we may state that it is expedient to use the algorithm that ensures compensation of the systematic RG error.

We have considered a simple algorithm of integration of inertial navigation system, satellite navigation system, magnetometer, and a barometric altimeter. The algorithm allows compensating systematic errors of rate gyros. A number of simplifying assumptions have been introduced. This is because, on the one hand, gyros in the system under study are not highly accurate, and on the other hand, such systems are supposed to be used on objects that move with low velocity and on rather small distances. In particular, these can be cheap unmanned aerial vehicles. An example has demonstrated the expediency of compensating the systematic error of rate gyros.

3.4. Summary for Chapter 3

3.4.1. In the 3rd chapter the problem of SINS correction from external sources of information is considered. Primarily this problem is considered from the point of view of usage of GPS only as the external source of information. In this case traditional Kalman filter can be applied for GPS and SINS fusion. However, it was shown that the covariance matrix of the state estimation errors is ill conditioned and the state vector estimation includes one unobservable mode. In order to overcome these difficulties the Cholesky factorization of this covariance matrix was considered and several method of finding generalized Cholesky factors were analyzed from the point of view of their computational expenses and possibilities of application for solution problem of improvement of Kalman filtering convergence.

3.4.2. Further application of integrated SINS and GPS correction includes application of other external sensors likewise the magnetometer and barometric altimeter. Some practical algorithms were proposed for solution of this problem.

3.4.3. Correction methods mentioned above didn't take in account the systematic errors of inertial sensors. However the compensation of gyros biases is inevitable for sustainable operation of navigation system. Corresponding correction algorithms were proposed also for solution of this problem.

3.4.4. The efficiency of proposed algorithms was proved by results of mathematical modeling, which are given in this chapter, as well as by results of laboratory, ground and flight experiments, which are considered in the 6th chapter.

CHAPTER 4 – METHODS OF INITIAL ALIGNMENT AND CALIBRATION OF THE COMPONENTS OF SENSORS OF THE INERTIAL NAVIGATION SYSTEM

4.1 Initial Alignment of Strap-Down Inertial Navigation Systems

Preliminary initial alignment of SINS, i.e., finding initial values of attitude, position, and velocity parameters is known to be one of the necessary conditions when solving navigation equations.

We assume that linear coordinates (position and velocity) are known. Therefore, researchers mostly pay attention to procedures of finding the initial attitude parameters by using measurements of available onboard sensors and well-known information about magnetic and gravitational fields of the Earth. Initial alignment usually means finding initial attitude parameters in one form of representation of another.

The essence of this procedure is to find angles (or the quaternion or direction cosine matrix) that describe spatial attitude of UAV (and SINS mounted on it, respectively) in the chosen navigation frame. Navigation frame is the system that is used for numerical integration of differential navigation equations; this can be Earth centered inertial frame, Earth centred Earth fixed frame or local-level topocentric frame, where the axes are directed North East Down (NED) or East North Up (ENU) (Lurie, 2002), (Siouris, 2007).

As the practice of UAV construction and operation in various application fields shows, the procedure of initial alignment of SINS is important since the correctness of determining object’s spatial position during flight depends on its accuracy. That is why efforts of many researchers all over the world are currently aimed at the development of the initial alignment procedure that would satisfy two principal conditions. On the one hand, such procedure should be executed in minimum time and be implemented with minimum complexity on onboard computers; and on the other hand, it should ensure the maximum accuracy of determining the spatial position of the UAV.

To derive attitude parameters, it is necessary to execute the procedure of initial alignment, possibly in two stages. Rough alignment is possible when MEMS-type rate gyros are used, which do not allow measurement of the angular velocity of the Earth. If more precise and expensive sensors (for instance, fiber-optics gyros) are used, it is possible to use the second stage of precise alignment. Since applying MEMS-type rate gyros is most typical for small UAV, we will first consider the variants of rough alignment.

4.1.1 Rough Alignment with the Use of Accelerometers and Rate Gyros

As to rough alignment, the literature proposes an approach to determining the attitude that consists of two basic components: accelerometer levelling (i.e., determining roll and pitch angles) and yaw/azimuth angle with the use of magnetometer (Titterton and Weston, 2004), (Petovello, 2003), (Shin, 2005), Ilnytska S. (2010).

The following relations are used in SINS to find roll and pitch angles from available measurements by accelerometers of the projections of acceleration a_x^b and a_y^b onto axes x, y of the body frame (Shin, 2005):

$$\begin{aligned} \phi &= \sin^{-1}\left(\frac{a_x^b}{g}\right) \cong \frac{a_x^b}{g}, \quad \delta\phi = \frac{b_{ax}}{g}, \\ \theta &= \sin^{-1}\left(\frac{a_y^b}{g}\right) \cong \frac{a_y^b}{g}, \quad \delta\theta = \frac{b_{ay}}{g}, \end{aligned} \tag{4.1}$$

where $\delta\phi$ and $\delta\theta$ are the errors of roll ϕ and pitch θ angles, respectively, b_{ax} and b_{ay} are accelerometer's biases, and g is the gravity acceleration.

If the accuracy of RG is enough to measure the components of Earth angular rate, then it is possible to determine yaw angle after the end of the previous stage, using RG measurements ω_y^b and ω_x^b from the following expressions (Shin, 2005):

$$\psi = \tan^{-1}\left(-\frac{\omega_y^b}{\omega_x^b}\right), \quad \delta\psi = \frac{b_{\omega_y^b}}{\omega_e \cos\varphi}, (\psi \cong 0), \tag{4.2}$$

where $\delta\psi$ is yaw angle error, $b_{\omega_y^b}$ is RG bias, ω_e is the Earth angular rate, ω_x^b, ω_y^b are components of Earth angular rate measured in a body frame and φ is current latitude. Noteworthy is that for low-cost inertial sensors, measurement errors often considerably exceed the Earth rotation velocity. Because of rather low accuracy and resolution of MEMS-type RGs, they cannot precisely measure components of the Earth angular velocity. In turn, this means that to find the yaw angle, it is also necessary to involve some external information sources, for example, a magnetometer.

A shortcoming of the above method of rough orientation is close dependence of the accuracy on measurement errors of accelerometers and impossibility to find yaw angle by means of MEMS-type RGs.

To eliminate this issue, data from accelerometers and magnetometers are proposed to be used for the initial alignment procedure.

4.1.2 Refined Alignment with the Use of Magnetometer and Accelerometer Readouts

This method assumes that along with onboard measurements of accelerometers and magnetometers, there is also information about vectors of gravitational and magnetic fields of the Earth in the initial frame.

Let us now consider the problem of determining the matrix of transition from the fixed frame to UAV frame. Denote the values of measurement vectors of onboard accelerometers and magnetometer (in the fixed frame) by a and m respectively. Let us construct the third vector, which is equal to the vector product of vectors a and m, i.e., vector $d = a \times m$. Thus, we have three vectors: a, m, d with their positions known in the fixed frame. It is important that that it is possible to measure the values of vectors a and m in the UAV frame (denote them by a_1, m_1) and to calculate vector $d_1 = a_1 \times m_1$. Thus, we have two matrices, $Z = [a \quad m \quad d]$ and $Z_1 = [a_1 \quad m_1 \quad d_1]$.

If we denote by A the matrix of transition from the fixed frame to the UAV frame, we can write

$$Z_1 = AZ. \tag{4.3}$$

Assuming that matrix Z is invertible allows us to obtain the following relation that defines matrix A:

$$\boldsymbol{A} = \boldsymbol{Z}_1 \boldsymbol{Z}^{-1}. \tag{4.4}$$

This matrix defines the UAV attitude.

Accounting for Measurement Errors.

Relation (4.4) defines the transition matrix A regardless of the measurement errors of vectors that define matrix Z_1. It is important that matrix A is orthogonal, i.e., $A' = A^{-1}$. With regard for measurement errors, the orthogonality property of matrix A defined by (4.4) can be violated. However, this property can be used to estimate matrix A.

Thus, let matrix Z_1 have the following structure as a result of measurements of vectors a_1 and m_1:

$$Z_1 = \tilde{Z}_1 + \varepsilon Z_1, \tag{4.5}$$

where εZ_1 is the matrix of measurement errors and $\tilde{Z}_1$ is the matrix of true measurements. In this case, taking into account (4.5), we can represent relation (4.4) as

$$A_1 = (\tilde{Z}_1 + \varepsilon Z_1)Z^{-1} = \tilde{Z}_1 Z^{-1} + \varepsilon Z_1 Z^{-1}. \tag{4.6}$$

The first and second terms in (4.6) define the exact value of the transition matrix and determination errors of this matrix. To derive estimate $\tilde{A}$ of matrix A, by the measurement results from (4.6), we can use the property of orthogonality of the transition matrix, namely, the condition $\tilde{A}^{-1} = \tilde{A}'$.

To obtain such estimate, let us construct singular value decomposition of matrix A_1:

$$A_1 = USV', \tag{4.7}$$

where U and V are orthogonal matrices and S is a diagonal matrix. In case of precise measurements, matrix A_1 will be orthogonal; hence matrix S will be an identity matrix. Based on this reasoning, we can take matrix

$$\tilde{A} = UV', \tag{4.8}$$

(where matrices U and V are defined by (4.7) and S is assumed to be identity matrix) as estimate $\tilde{A}$ of matrix A.

Example 4.1

Let vectors a and m be as follows:

$$a = [0 \quad 0 \quad 1]', \quad m = [1 \quad 1 \quad 1]'.$$

Respectively, vector $d = a \times m = [-1 \quad 1 \quad 0]'$.

To find matrix A, we will specify the following values of Euler angles:

$$\varphi = \psi = \theta = \frac{\pi}{4}.$$

The following matrix corresponds to these values of Euler angles (see, for example, Chapters 2, 3 (Lurie, 2002)):

$$A = \begin{bmatrix} 0.1464 & 0.8536 & 0.5000 \\ -0.8536 & -0.1464 & 0.5000 \\ 0.5000 & -0.5000 & 0.7071 \end{bmatrix}.$$

As matrix εZ_1, we take a matrix whose elements are random numbers with zero expectation and variance 10^{-2} (this matrix is generated by rand.m procedure from MATLAB package (Mathworks.com, 2018)). As a result of modeling, we obtained the following value of matrix S that appears in (4.7):

$$S = \begin{bmatrix} 1.0102 & 0 & 0 \\ 0 & 0.9889 & 0 \\ 0 & 0 & 0.9753 \end{bmatrix}.$$

Using relation (4.8), we found the following estimate $\tilde{A}$ of matrix A:

$$\tilde{A} = \begin{bmatrix} 0.1410 & 0.8516 & 0.5048 \\ -0.8549 & -0.1523 & 0.4959 \\ 0.4992 & -0.5015 & 0.7066 \end{bmatrix}.$$

Note that the norm of the difference of A and $\tilde{A}$ is of order 10^{-3}:

$$\left\| A - \tilde{A} \right\| = 0.0075 .$$

The corresponding initial quaternion is calculated from the obtained DCM by formulas (2.5) and is used as the initial condition for integration of the kinematic rotation equations presented in Chapter 2.

Similar methods of determining the attitude of a rigid body as applied to spacecraft were considered in (Avraamenko, Larin and Bordug, 1983).

To increase the accuracy of determining the initial attitude, it is obviously necessary to possess additional information sources. In this book, we consider navigation systems based rather rough RGs of MEMS type. These gyros are known to have rather low sensitivity, which does not allow measuring the angular velocity of the Earth. If more precise sensors can be applied (for example, fiber-optics gyros (FOG)) whose sensitivity allows measuring the angular velocity of the Earth at the start point, more precise methods of initial alignment are available.

4.2 Calibration Methods and Mathematical Models of Measurements of Components of Sensors of Inertial Navigation System

Strap-down inertial navigation system (SINS) is constructed so that integrating the measured angular velocities and linear accelerations yields the navigational solution. SINS accuracy depends on many factors such as presence of systematic errors, computing errors, and errors of raw information sensors. The last component contributes the most to degradation of INS accuracy since the integration process is very sensitive to regular sensor errors. For example, accelerometer bias causes a velocity error proportional to time t and a position error proportional to t^2. RG sensor bias leads to a velocity error proportional to t^2 and position error proportional to t^3 (Artese and Trecroci, 2008), (Kharchenko, Larin and Ilnytska, 2012). In particular, the study (Artese and Trecroci, 2008) proposes the following approximate formula to estimate the degradation of position accuracy found with the use of errors of RGs and accelerometers:

$$\delta p(t) \approx \delta p_0 + \delta v_0 \Delta t + \delta b_{0a} \frac{\Delta t^2}{2} + \delta b_{0g} \frac{\Delta t^3}{6} + \delta \alpha_0 g \frac{\Delta t^2}{2} + \\ + \delta H_{0z} (V \Delta t) + SF_{0a} \frac{\Delta t^2}{2} + SF_{0g} (\Delta H_z)(V \Delta t) \,, \tag{4.9}$$

where

δp_0 is position error at time t_0;

δv_0 is velocity error at time t_0;

$\Delta t = t - t_0$ is the time passed;

δb_{0a} is accelerometer's bias at time t_0

δb_{0g} is RG bias at time t_0

$\delta \alpha_0$ is horizontal misalignment at time t_0

δH_{0z} is yaw misalignment multiplied by approximate distance;

SF_{0a}, SF_{0g} are the errors of scale coefficients of accelerometers and RGs;

F is measured acceleration and g is acceleration of gravity ($\approx 9.81\, m/s^2$).

The above equation demonstrates the importance of determining biases, errors of scale factor, and non-orthogonality (misalignment) for adequate SINS operation. Determinig of these parameters is the ultimate goal of the calibration procedure.

Calibration is usually performed by comparing certain reference values with sensor measurements if respective sensor model is available. Depending on model's complexity and the number of unknown parameters, experimental measurements are formulated so that the number of obtained independent equations exceeds the number of unknown error parameters.

Earth gravity and its projections are used as reference signals for accelerometers whose measurement range is not much wider than ±1 g (Meleshko and Nesterenko, 2011). For rate gyros, reference signals are the Earth rotation velocity or only the prescribed rotation velocity of a rotating motion table in case of "rough" sensors that cannot detect the Earth rotation.

To perform the calibration procedure, it is necessary to have a sensor measurement model that includes all information about it. Such model can be absolutely simple, for example, include only scale factor. A simple model can be applied to a big class of sensors while a complicated model is based on more specific information about the sensor. The measurement accuracy of a calibrated sensor increases as the model of its measurements is complicated; hence, it is necessary to reach a compromise between model's complexity and accuracy.

Inertial sensors and systems are developed for various applications, which include providing precise autonomous long-time navigation in planes, ships, and submarines as well as operating as a part of integrated inertial-satellite navigation system for small UAV. The performances of sensors' components will obviously substantially differ from each other in these cases. In the first case, these will be expensive, larger, and precise sensor. In the second case, most likely, these will be small-size, rather inexpensive MEMS-type sensors.

It is natural that different classes of sensors should have their own calibration methods. For more expensive and precise systems, it is expedient to use big calilbration motion tables and corresponding calibration techniques as it was considered in (Lee, Tunik and Kim, 1999). And small-size, rather inexpensive MEMS-type sensors demand somewhat simplified calibration techniques with the use of available tools. Below, we will consider calibration methods intended for inexpensive accelerometers and MEMS-type RGs with representation of experimental results.

4.2.1 Calibration Method and Mathematical Model of Measurements of Accelerometers

We propose to base the accelerometer error model on the model presented in (Lee, Tunik and Kim, 1999), (IEEE Std 1293-1998, R2008, 2011), (Choi, Jang and Kim, 2010) but without regard for nonlinearity error. It was shown in (Kharchenko, Larin and Ilnytska, 2012) that its contribution is insignificant (at the level of sensitivity of the last bit); however, it makes calculations much more complicated:

$$\begin{aligned} a_x^m &= (1+s_{ax})(\mathrm{a}_x + \varepsilon_{xz} a_y - \varepsilon_{xy} a_z + b_{ax}) + \eta_{ax}; \\ a_y^m &= (1+s_{ay})(a_y + \varepsilon_{yx} a_z - \varepsilon_{yz} a_x + b_{ay}) + \eta_{ay}; \\ a_z^m &= (1+s_{az})(a_z + \varepsilon_{zy} \mathrm{a}_x - \varepsilon_{zx} a_y + b_{az}) + \eta_{az}, \end{aligned} \tag{4.10}$$

where $a^m = \left[a_x^m, a_y^m, a_z^m\right]^T$ are measured acceleration values; $s_a = [\, s_{ax}, s_{ay}, s_{az} \,]^T$ are scale factors of accelerometers; $a = \left[a_x, a_y, a_z\right]^T$ are true acceleration values; ε_{ij} are misalignment errors of sensitivity axes; $b_a = \left[b_{ax}, b_{ay}, b_{az}\right]^T$ are biases; and η is noise components of measurement errors.

In the matrix form, (4.10) becomes

$$\mathrm{a^m} = \mathrm{A} \cdot \mathrm{a} + \mathrm{b} + \eta, \tag{4.11}$$

where $A = SA_{\varepsilon}$; $b = Sb_{a}$; $S = \mathrm{diag}\{1+s_{ax}, 1+s_{ay}, 1+s_{az}\}$ is the diagonal error matrix of scale factors; $A_{\varepsilon} = \begin{bmatrix} 1 & \varepsilon_{xz} & -\varepsilon_{xy} \\ \varepsilon_{yx} & 1 & -\varepsilon_{yz} \\ \varepsilon_{zy} & -\varepsilon_{zx} & 1 \end{bmatrix}$ is matrix of the misalignments of accelerometer's sensitivity axes.

The vector of unknown parameters is $\vec{X} = [A_1\ A_2\ A_3\ b_1\ b_2\ b_3]^T$ $\dim[\vec{X}] = 12\times1$, where $A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}$, $\dim[A_1] = \dim[A_2] = \dim[A_3] = 1\times3$. Thus, there are 12 unknown parameters: nine for A and three for b. To find 12 unknown parameters, it is necessary to have $n_e \geq 12$ independent equations.

To collect the necessary data, the following measurements are performed. Rotation about axes X and Y with $45°$ intervals and about 2 min fixation at each position is carried out to obtain full turn as a result. Fig. 4.1 schematically represents the calilbration motion table with inertial measuring module (IMU) mounted so that rotation is carried out around axis X. As we see from the figure, to perform rotation around axis Y, it is necessary to make a 90 ° turn of IMU around axis Z of the body frame.

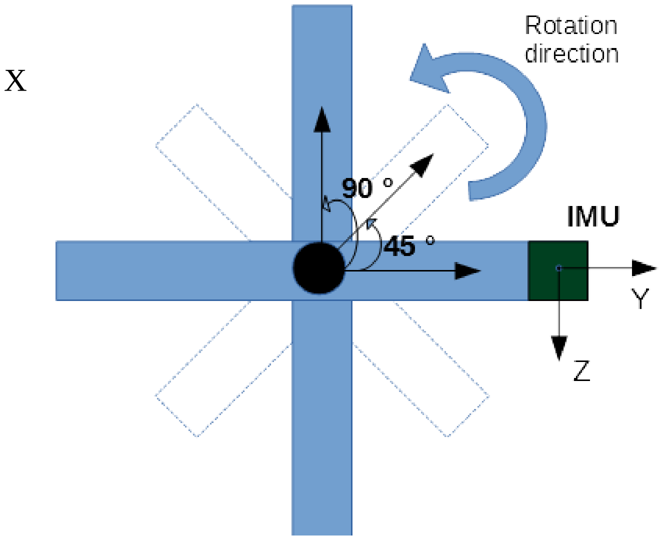


Figure 4.1. Schematic diagram of the calilbration motion table

Thus, we obtain eight positions for each of the axes:

$$\begin{matrix} \phi_1(i) = i\cdot 45° \\ \phi_2(i) = i\cdot 45° \end{matrix},\ i = 1:8$$

Rotation around both axes yields 48 equations. In the matrix form, they are as follows:

$$B_\Sigma = A_\Sigma \vec{X} + \eta\,, \tag{4.12}$$

where $B_\Sigma = \begin{bmatrix} a_{1x}^m \\ \vdots \\ a_{8x}^m \\ a_{1y}^m \\ \vdots \\ a_{8y}^m \end{bmatrix}$ and $A_\Sigma = \begin{bmatrix} A_{1x} \\ \vdots \\ A_{8x} \\ A_{1y} \\ \vdots \\ A_{8y} \end{bmatrix}$ are the measurement vector and the matrix of reference signals, respectively; η is noise component; $a_{ix,y}^m$ are the measured accelerations with respect to three axes, obtained from the i-th position (i = 1: 8) as a result of rotation around axes X and Y, respectively; $A_{ix,y}$ is [3x12] matrix of reference signals for the i-th position (i = 1: 8) as a result of rotation around axes X and Y, respectively. Matrix A_{ix} takes the following form:

$$A_{ix} = \begin{bmatrix} a_{ix}^T & 0_{1\times3} & 0_{1\times3} & 1 & 0 & 0 \\ 0_{1\times3} & a_{ix}^T & 0_{1\times3} & 0 & 1 & 0 \\ 0_{1\times3} & 0_{1\times3} & a_{ix}^T & 0 & 0 & 1 \end{bmatrix},$$

where $a_{ix} = C_{ix} g_{ref}$, $g_{ref} = [0 \quad 0 \quad 9{,}81]^T$, $C_{ix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\phi(i)) & \sin\phi(i) \\ 0 & -\sin\phi(i) & \cos(-\phi(i)) \end{bmatrix}$.

Here, i denotes position number, $\phi(i) = i \cdot 45^\circ,\ i = 1:8$. Rotation around axis Y yields similar results:

$$A_{iy} = \begin{bmatrix} a_{iy}^T & 0_{1\times3} & 0_{1\times3} & 1 & 0 & 0 \\ 0_{1\times3} & a_{iy}^T & 0_{1\times3} & 0 & 1 & 0 \\ 0_{1\times3} & 0_{1\times3} & a_{iy}^T & 0 & 0 & 1 \end{bmatrix},$$

where $a_{iy} = C_{iy} g_{ref}$, $C_{iy} = \begin{bmatrix} \cos(-\phi(i)) & 0 & -\sin\phi(i) \\ 0 & 1 & 0 \\ \sin\phi(i) & 0 & \cos(-\phi(i)) \end{bmatrix}$.

Obviously, using the reference value g_{ref} makes it possible to calculate all the elements of A_Σ. Considering the redundant number of measurements, we apply the least squares method to find vector $\vec{X}$ from the system of equations (4.12):

$$\vec{X} = \left(A_\Sigma{}^T A_\Sigma\right)^{-1} \cdot A_\Sigma{}^T \cdot B_\Sigma\,. \tag{4.13}$$

We select S, A_ε, b_a from vector $\vec{X}$ as follows:

$$S = diag\{a_{11}, a_{22}, a_{33}\};\ A = \lfloor a_{ij} \rfloor.$$

$$b_a = S^{-1} \cdot b;\ A_\varepsilon = S^{-1} \cdot A.$$

To increase the calculation accuracy of vector $\vec{\mathrm{X}}$, we propose to use QR-factorization. Matrix A_Σ can be decomposed into two matrices: $A_\Sigma{}^{qr} = Q^T R$, where $QQ^T = I$, $R = \begin{bmatrix} R_0 \\ 0 \end{bmatrix}$, and R_0 is a square matrix, i.e., $size(R_0) = n \times n$. Let us represent the product of two matrices $\mathrm{Q} \cdot \mathrm{B}_\Sigma$ as two column vectors:

$$Q \cdot B_\Sigma = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \tag{4.14}$$

where $\mathrm{size}(\mathrm{z}_1) = \mathrm{n} \times 1$. Then we will calculate the vector of unknown parameters as

$$\vec{\mathrm{X}} = \mathrm{R}_0^{-1} \cdot \mathrm{z}_1 \tag{4.15}$$

Noteworthy is that the condition numbers of matrix R_0 obtained by means of QR-factorization of the overall measurement matrix A_Σ are related as follows:

$$\mathrm{cond}(\mathrm{R}_0) = \sqrt{\mathrm{cond}\left(\mathrm{A}_\Sigma^{\mathrm{T}} \mathrm{A}_\Sigma\right)} \tag{4.16}$$

To carry out an experiment according to the technique described above, we used a not too much precise rotating motion table in a vertical plane and an inertial measurement unit ADIS16362 by Analog Device Company, which includes three orthogonally arranged accelerometers and three rate gyros.

ADIS16362 is a functionally complete inertial measurement unit, which outputs 14-digit data corresponding to projections of angular velocity and linear acceleration with respect to three axes (Analog.com, 2018).

The accelerometer calibration procedure described above repeated two times. To estimate the adequacy of the measured data, we will use vectors from Eq. (4.14) and calculate so-called data "deviation":

$$\boldsymbol{dev} = \frac{\|\boldsymbol{z}_2\|}{\|\boldsymbol{z}_1\|} \cdot 100\%, \tag{4.17}$$

where $\|\cdot\|$ is vector norm. It is obvious that ideal deviation $\boldsymbol{dev}$ should be 0%; however, the first time we obtained 3% and the second time 4%. This can be explained by low accuracy of the motion table, rotation axes backlashes (Figs. 4.2 – 4.3), and certain errors in angle alignments, for example, not exactly 45º but 45.4º.

Since the results of the two experiments were similar, in what follows we will describe data processing and results of only the first experiment. During rotation around axis X, component g_x should be equal to zero; however, Fig. 4.2 shows that this is not the case in practice, and certain deviations are observed, which can be explained by rotation axes backlashes, which contribute to data deviation. Similar results were obtained during rotation around axis Y (see Fig. 4.3).

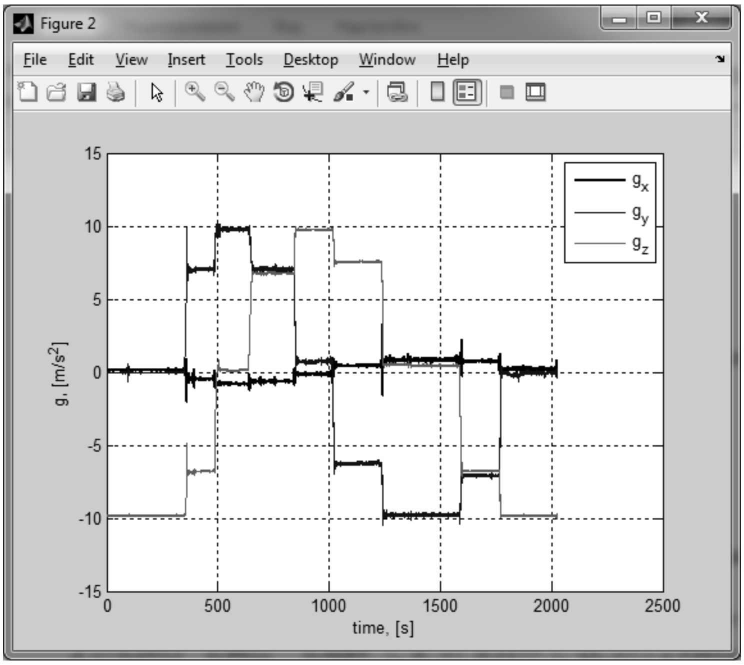


Fig. 4.2. Measurements of accelerometers during rotation around axis X

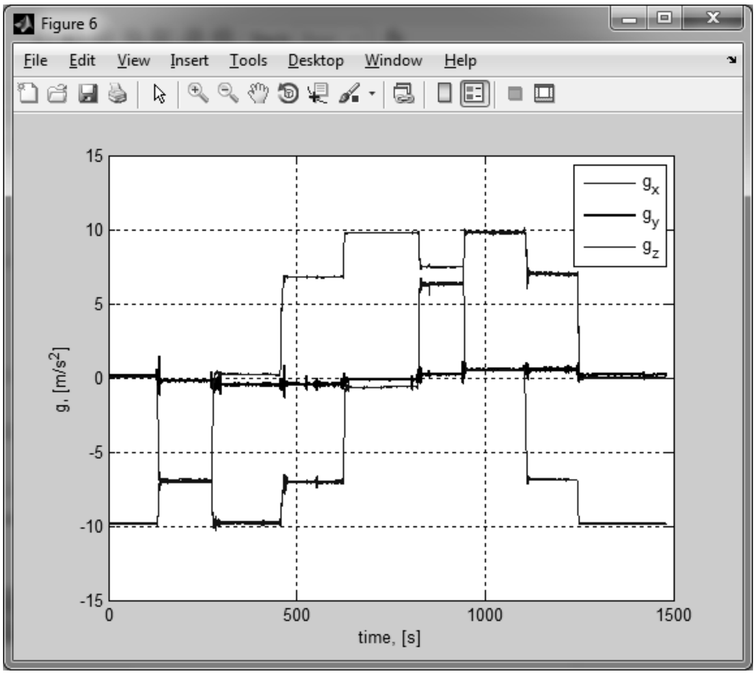


Fig. 4.3. Measurements of accelerometers during rotation around axis Y

Having processed the measurement data according to the formulas above, we obtained the following estimates of the accelerometer sensitivity matrix and bias:

$$A = \begin{bmatrix} 0{,}9917 & -0{,}0830 & -0{,}0229 \\ 0{,}0513 & 0{,}9922 & 0{,}0085 \\ 0{,}0194 & -0{,}0239 & 1{,}0026 \end{bmatrix}; \qquad b_a = \begin{bmatrix} -0{,}0538 \\ 0{,}1177 \\ 0{,}3372 \end{bmatrix}.$$

Note that matrix A is very close to the identity matrix I; therefore, it is expedient to represent it as

$$A = I + \varepsilon B, \tag{4.18}$$

where εB, in turn, can be represented as the sum of a symmetric and a skew-symmetric matrices:

$$\varepsilon \mathrm{B} = \varepsilon \mathrm{B}_{\mathrm{sym}} + \varepsilon \mathrm{B}_{\mathrm{skew}}, \tag{4.19}$$

$$\varepsilon \mathrm{B}_{\mathrm{sym}} = \left(\frac{\varepsilon \mathrm{B} + \varepsilon \mathrm{B}^{\mathrm{T}}}{2} \right); \quad \varepsilon \mathrm{B}_{\mathrm{skew}} = \left(\frac{\varepsilon \mathrm{B} - \varepsilon \mathrm{B}^{\mathrm{T}}}{2} \right)$$

With regard for the small order of magnitudes in matrices εB_{sym}, εB_{skew}, we can consider the following relation to be true (to within ε^2):

$$\mathrm{I} + \varepsilon \mathrm{B} \approx (\mathrm{I} + \varepsilon \mathrm{B}_{\mathrm{sym}})(\mathrm{I} + \varepsilon \mathrm{B}_{\mathrm{skew}}) = (\mathrm{I} + \varepsilon \mathrm{B}_{\mathrm{skew}})(\mathrm{I} + \varepsilon \mathrm{B}_{\mathrm{sym}}). \tag{4.20}$$

The first product characterises actual errors of sensor's axes misalignment and of scale factors as themselves:

$$\left(I + \varepsilon B_{sym}\right) = \begin{bmatrix} 0{,}9850 & 0{,}0249 & -0{,}0033 \\ 0{,}0249 & 0{,}9905 & -0{,}0083 \\ -0{,}0033 & -0{,}0083 & 1{,}0038 \end{bmatrix},$$

which can be considered as an estimate of the sensitivity matrix of the accelerometer unut.

The second product characterises the error of misalignment of sensor's sensitivity axes with respect to calibration plane:

$$\left(I + \varepsilon B_{skew}\right) = \begin{bmatrix} 1 & -0{,}1032 & -0{,}0161 \\ 0{,}1032 & 1 & 0{,}0140 \\ 0{,}0161 & -0{,}0140 & 1 \end{bmatrix}.$$

Thus, $\left(I + \varepsilon B_{skew}\right)$ is a turn matrix, which connects the trihedron of sensors' sensitivity axes and the trihedron of calibration plane. Since the rotation values are small, we can use the following form for this matrix (Wittenburg, 1977):

$$\left(I + \varepsilon B_{skew}\right) \cong \begin{bmatrix} 1 & \mu_3 & -\mu_2 \\ -\mu_3 & 1 & \mu_1 \\ \mu_2 & -\mu_1 & 1 \end{bmatrix}, \qquad (4.21)$$

where μ_1, μ_2, μ_3 are small angles of rotation of the trihedron $Oxyz$ of sensors' sensitivity axes around axes x, y, z , respectively.

The technique described in this study was approved by the data of two experiments. Similar data were obtained both times, which shows that accelerometer error estimates are repeated and calibration technique is adequate. However, these experiments were carried out more likely for the previous "rouhgh" estimate of sensors' errors. To make it possible to talk about more exact results and to compensate these errors in handling algorithms, it is necessary to repeat the experiment at one of the special calibration motion tables, for example three- or bi-axial motion tables, like (Acutronic.com, 2018), (Ixblue.com, 2018), (Actidyn.com, 2018) for example.

Considering rather small value of nonlinearity error estimate, it makes no sense to take it into account under the described calibration conditions. This will probably be expedient in calibration at precision motion tables.

4.2.2. Calibration Method and Mathematical Model of RG Measurements

We propose to be based on the model of RG errors like in (Lee, Tunik and Kim, 1999) but to simplify it like in (Kharchenko et.al., 2012 a) and neglect the nonlinearity error since its contribution to MEMS-type sensors is rather insignificant:

$$\begin{aligned}\omega_x^m &= (1+s_{\omega x})(\omega_x + \alpha_{xz}\omega_y - \alpha_{xy}\omega_z + b_{\omega x}) + \eta_{\omega x};\\ \omega_y^m &= (1+s_{\omega y})(\omega_y + \alpha_{yx}\omega_z - \alpha_{yz}\omega_x + b_{\omega y}) + \eta_{\omega y};\\ \omega_z^m &= (1+s_{\omega z})(\omega_z + \alpha_{zy}\omega_x - \alpha_{zx}\omega_y + b_{\omega z}) + \eta_{\omega z},\end{aligned} \tag{4.22}$$

where $\omega^m = \left[\omega_x^m, \omega_y^m, \omega_z^m\right]^T$ is measured angular velocity signal; $s_\omega = [\, s_{\omega x}, s_{\omega y}, s_{\omega z}\,]^T$ is vector of RG scale factors; $\omega = \left[\omega_x, \omega_y, \omega_z\right]^T$ are reference values of angular velocity; α_{ij} are the errors of misalignment of sensitivity axes; $b_\omega = \left[b_{\omega x}, b_{\omega y}, b_{\omega z}\right]^T$ is bias vector; and η is noise components of measurement errors.

The problem where nonlinear component of sensor’s errors should be taken into account is considered in (Avraamenko, Larin and Bordug, 1983).

We can write the above system of equations in the matrix form as

$$\omega^m = A \cdot \omega + b + \eta, \tag{4.23}$$

where $\mathrm{A} = \mathrm{S}_\omega \mathrm{A}_\alpha; \mathrm{b} = \mathrm{S}_\omega \mathrm{b}_\omega;$ $\mathrm{S}_\omega = \mathrm{diag}\{1 + \mathrm{s}_{\omega x}, 1 + \mathrm{s}_{\omega y}, 1 + \mathrm{s}_{\omega z}\}$ is the diagonal matrix of errors of scale factor; $A_\alpha = \begin{bmatrix} 1 & \alpha_{xz} & -\alpha_{xy} \\ \alpha_{yx} & 1 & -\alpha_{yz} \\ \alpha_{zy} & -\alpha_{zx} & 1 \end{bmatrix}$ is the matrix of misalignment errors of RG sensitivity axes.

The vector of unknown parameters is $\vec{\mathrm{X}} = \left[\mathrm{A}_1\ \mathrm{A}_2\ \mathrm{A}_3\ \mathrm{b}_1\ \mathrm{b}_2\ \mathrm{b}_3\right]^\mathrm{T}$, $\dim[\vec{\mathrm{X}}] = 12 \times 1$, where $\mathrm{A} = \begin{bmatrix} \mathrm{A}_1 \\ \mathrm{A}_2 \\ \mathrm{A}_3 \end{bmatrix}$ $\dim[\mathrm{A}_1] = \dim[\mathrm{A}_2] = \dim[\mathrm{A}_3] = 1 \times 3$. Thus, there are 12 unknown parameters: nine for A and three for b. To find 12 unknown parameters, we should have $N_e \geq 12$ independent equations.

To accumulate the necessary amount of data, a series of measurements is carried out at a motion table (its kinematic scheme is presented in Fig. 4.4). The platform of the motion table rotates around the vertical axis with constant angular velocity ω_{ref} . The calilbration motion table is alternately inclined by chosen angles with respect to axes X and Y (Table 4.1) and rotates around vertical axis (Fig. 4.4) about 1 to 2 minutes clockwise and counterclockwise. Thus, there are seven positions of the motion table. At each of them we measure positive and negative angular velocities of motion.

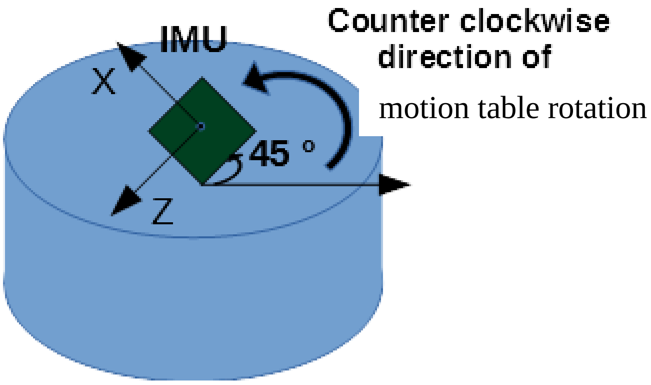


Figure 4.4. Schematic representation of the motion table for calilbration

Table 4.1 - Parameters of the experiment

No.	Inclination with respect to axis X, $\varphi 1\,[\mathrm{rad}]$	Inclination with respect to axis Y $\varphi 2\,[\mathrm{rad}]$	Reference values of angular velocities $[\omega_x\ \omega_y\ \omega_z]^T$
1	0	0	$\pm\omega_{ref}\cdot[0\quad 0\quad 1]^T$
2	0	-π/4	$\pm\omega_{ref}\cdot\frac{\sqrt{2}}{2}\cdot[1\quad 0\quad 1]^T$
3	0	π /4	$\pm\omega_{ref}\cdot\frac{\sqrt{2}}{2}\cdot[-1\quad 0\quad 1]^T$
4	0	π /2	$\pm\omega_{ref}\cdot[-1\quad 0\quad 0]^T$
5	- π /4	0	$\pm\omega_{ref}\cdot\frac{\sqrt{2}}{2}\cdot[0\quad -1\quad 1]^T$
6	π /4	0	$\pm\omega_{ref}\cdot\frac{\sqrt{2}}{2}\cdot[0\quad 1\quad 1]^T$
7	π /2	0	$\pm\omega_{ref}\cdot[0\quad 1\quad 0]^T$

In the table, $\varphi 1, \varphi 2$ are inclination angles (in radians) with respect to axes X and Y, respectively; ω_{ref} is the given velocity of motion table's rotation.

Carrying out all the measurements yields 42 linearly independent equations: the number of experiments is $N = 7 \cdot 2 = 14$, the number of equations is $N_e = 14 \cdot 3 = 42$.

In the matrix form, they can be written like expression (4.12), where $B_{\Sigma} = \begin{bmatrix} \omega_1^m \\ \vdots \\ \omega_{14}^m \end{bmatrix}$ and $A_{\Sigma} = \begin{bmatrix} \tilde{A}_1 \\ \vdots \\ \tilde{A}_{14} \end{bmatrix}$ are the vector of measurements and the matrix of reference signals, respectively, η is the noise component of measurement error. Here, ω_i^m represents the angular velocities measured with respect to three axes, obtained from the i th experiment ($\mathrm{i} = 1:14$), $\tilde{\mathrm{A}}_{\mathrm{i}}$ is the [3x12] matrix of reference signals for the i th experiment ($\mathrm{i} = 1:14$):

$$\tilde{A}_j = \begin{bmatrix} \omega_j^T & 0_{1\times 3} & 0_{1\times 3} & 1 & 0 & 0 \\ 0_{1\times 3} & \omega_j^T & 0_{1\times 3} & 0 & 1 & 0 \\ 0_{1\times 3} & 0_{1\times 3} & \omega_j^T & 0 & 0 & 1 \end{bmatrix},$$

where ω_j^T are reference values of angular velocities with respect to the three axes for the i th experiment and $0_{1\times 3}$ is zero vector of corresponding dimension.

Vector $\vec{X}$ can be found from the system of equations (4.12) similarly to (4.13). From vector $\vec{X}$ we separate out $S, A_{\alpha}, b_{\omega}$ similarly to how it has been done for accelerometers. To increase the calculation accuracy of vector $\vec{X}$, we propose to use QR-factorization similarly to how it has been done for accelerometers.

To carry out the experiment according to the technique described above, we used one axis rotating UPG-48 motion table for checking and testing of gyroscopic devices, miniature attitude and heading reference system IG-500N by SBG Systems (Sbg-systems.com, 2018) as a signal reference, and inertial measuring module ADIS16362 by Analog Device (Analog.com, 2018).

According to the technique described above, we carried out a series of measurements at two different angular rates in the beginning of the measurement range (± 3.9 deg/s) and then we generated the system of equations (4.12) in accordance with conditions indicated in the Table 3.2, which consisted of 84 (42 for each angular rate) linearly independent equations. We have calculated the condition number of the obtained experimental matrix. In addition, to assess the adequacy of the measured data, we have calculated so-called data “deviation” according to (4.17). It is clear that ideally it should be 0%; however, we have obtained 0.82%. This can be explained by some rotation axis backlashes (Fig. 4.5 – 4.6) and certain angle misalignment errors, for example, not exactly 45° but 45.3° since the UPG-48 motion table was not precise enough.

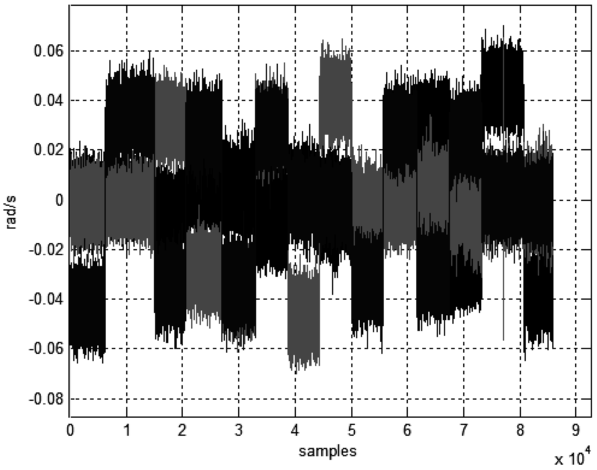


Fig. 4.5. RG measurements for the given rotation velocity of 3 deg/sec

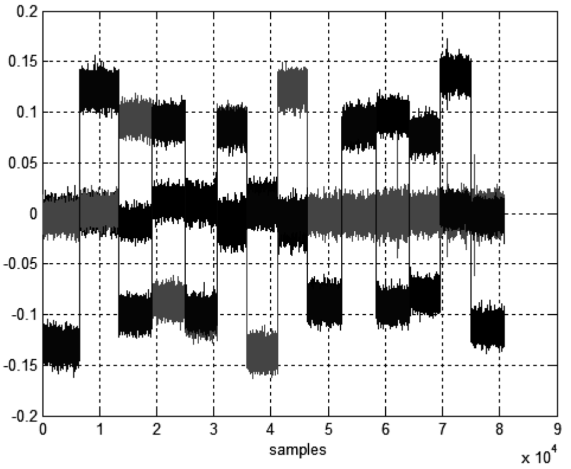


Fig. 4.6. RG measurements for the given rotation velocity of 9 deg/sec

Having processed the measurement data according to the above formulas, we obtained estimates of the RG misalignment matrix, errors of scale factor and bias (Table 4.2).

Table 4.2. Calibration results

Rotation velocities [°/s]	Misalignment matrix, A_α			Scale factor S	Bias, b_ω $[°/s]$
3	1.0000	-0.0062	0.0160	1.0015	0.0006
9	0.0063	1.0000	0.0118	1.0007	-0.0012
	-0.0171	-0.0071	1.0000	1.0003	-0.0003

Let us consider in more detail matrix A, which is obtained directly from vector $\vec{X}$, Eq. (4.15). Since the matrix is very close to the identity matrix I, it is expedient to represent it in the form (4.20), where the first product characterises actually the errors of misalignment and scale factors of sensors and the second product characterizes the error of sensor's sensitivity axes misalignment with respect to the calibration plane:

$$\left(I+\varepsilon B_{sym}\right)=\begin{bmatrix}1{,}0015 & 0{,}0001 & -0{,}0005\\ 0{,}0010 & 1{,}0007 & 0{,}0023\\ -0{,}0005 & 0{,}0023 & 1{,}0003\end{bmatrix} \qquad \left(I+\varepsilon B_{skew}\right)=\begin{bmatrix}1{,}0000 & -0{,}0063 & 0{,}0165\\ 0{,}0063 & 1{,}0000 & 0{,}0095\\ -0{,}0165 & -0{,}0095 & 1{,}0000\end{bmatrix}$$

We consider matrix $\left(I+\varepsilon B_{sym}\right)$ as an estimate of the sensitivity matrix of RG unit and $\left(I+\varepsilon B_{skew}\right)$ as a turn matrix, which relates the trihedron of sensors' sensitivity axes and trihedron of calibration plane. Taking into account that turn values are small, we can write this matrix in the same form as in (4.21).

Let us also investigate the stationarity of the RG bias error, which can be done without use of special equipment. The bias error adds fixed errors to initial sensors' measurements. It looks as if RG rotates while it is fixed. This results in permanent accumulation of angle error obtained from the multiplication of RG bias error by time (Looneys, 2010):

$$\varphi_{BIAS} = \int_0^{t_1} b_\omega dt = b_\omega \cdot t_1 \ . \tag{4.24}$$

One of the techniques to estimate RG bias errors is averaging of sensors' measurements while the device is at rest. When determining the length of measurement's sampling to obtain the maximum accuracy, it is possible to use the Allan variance plot, which relates bias estimate with integration time (Looneys, 2010), (Allan, 1966).

The Allan variance (AVAR), which is also known as two-sample variance, is a measure of frequency stability in clocks, oscillators, and amplifiers and is expressed mathematically as $\sigma_y^2(\tau)$ (Allan, 1966), (Allan and Barnes, 1981), (El-Sheimy, Hou and Niu, 2008).

Many manufacturers of inertial sensors present Allan variance plots in technical specifications of sensors, in particular for RG in ADIS16362, we have such plot (Fig. 4.7) from which we can see that optimal (with respect to obtaining the least bias error) integration time is 100 to 200 sec.

Taking this information into account, we will analyze how stable RG bias errors are in time. To this end, we will analyse data from RG that were motionless during 15 hours. In the beginning of each 30 min, we take a data file with interval of 120 sec and integrate angular velocities. The results of the analysis are presented in Fig. 4.8.

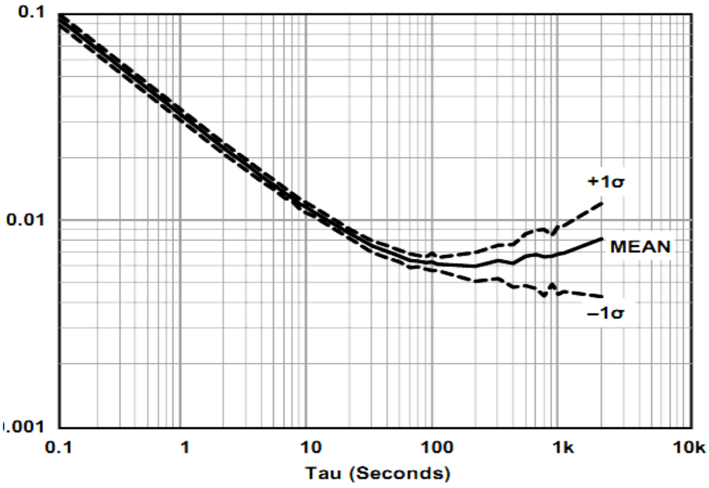


Fig. 4.7 Allan variance plot for rate gyros (Analog.com, 2018)

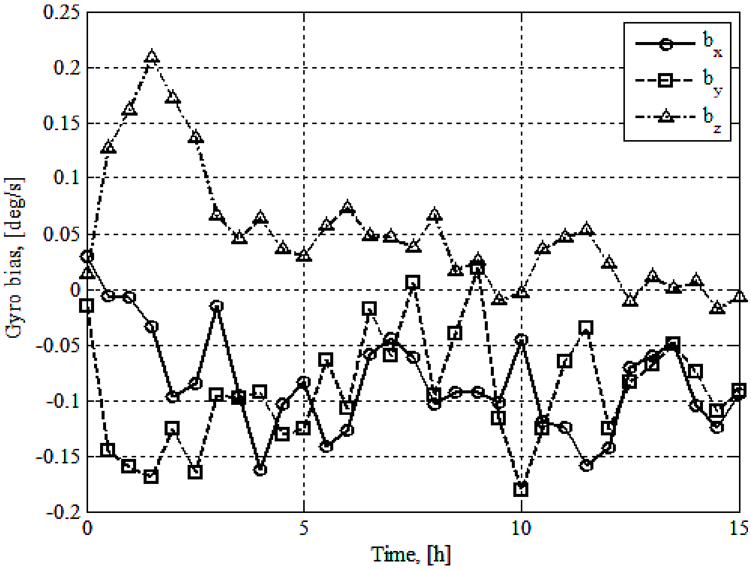


Fig. 4.8. Drift of RG bias in time

Here, $b_{\omega} = [b_x, b_y, b_z]$ is values of RG bias in % with respect to axes x, y, z (solid line with circles, dashed line with quadrates, and dot-and-dash line with triangles, respectively). As is seen from Fig. 4.8, errors of RG bias vary in time slowly and rather insignificantly under steady states (time interval from 5 to 15 hours). The data in Fig. 4.8 and in Table 3.3 are of similar order; however, there are some differences, which can be due to nonstationary RG bias, temperature drifts, etc., as is seen from Fig. 4.8.

As a result of this observation, we may assume that having defined bias errors, we can compensate for them rather efficiently for some time, without involving additional means (like extension of the state vector of Kalman filter to estimate RG bias, etc.) since they stay within the limits of certain established value for a long time.

The technique of calibration at a rotating motion table described above was tested for two different rotation velocities. To estimate the compatibility of the obtained system of equations, we calculated the condition number of the cumulative experimental matrix as well as data variation percent; the obtained values were within admissible norms. We also analyzed finding RG bias errors with the use of static measurements on a long time interval. The order of the errors found by two techniques coincided and corresponded to specifications of the sensors (Analog.com, 2018). However, such experiments were rather carried out for the previous "rough" estimate of sensors' errors. To make it possible to talk about more accurate results and further compensation of these errors in handling algorithms, it is necessary to repeat the experiment at a more precise calilbration motion table, for example, at one of three- or bi-axial motion tables, like (Acutronic.com, 2018), (Ixblue.com, 2018), (Actidyn.com, 2018) for example.

To more accurately assess the contribution of sensors’ calibration into the improvement of calculation of the navigation solution of integrated SINS/GNSS system, it is necessary to perform simultaneous calibration of all the components of inertial measuring module and to carry out a number of experiments with usage a certain reference trajectory.

4.3. Summary for Chapter 4

4.3.1. We have analyzed different variants of initial SINS alignment: rough alignment with the use of accelerometers and RGs in the form of the Euler angles; alignment by means of vector multiplication of measurement vectors of magnetometers and accelerometers, which allows finding the direction cosine matrix and initial angular attitude of the body frame with respect to the navigation system.

4.3.2. We have analysed and improved calibration methods for accelerometers, rate gyros, and MEMS-type magnetometers. They differ from the available methods since we proposed simplified sensor measurement models, used QR-factorization to increase the computing accuracy, and in addition introduced calculation of measurement deviation to assess the calibration adequacy. The calibration methods for RGs and accelerometers were tested against real data, and the corresponding results are presented. Determing the components of sensor errors as a result of calibration increases the computing accuracy of SINS navigation parameters by accounting for these errors.

CHAPTER 5 – DETECTING SENSOR FAILURES

Problems of detecting malfunction (failures, damageability, etc.) of system elements have been continuously attracting researchers' attention (Chow and Willsky, 1984), (Desai, Deckert and Deyst, 1979), (Deyst et.al., 1981), (Johnson, 1989), (Kreinovich et.al, 2012), (Kuznetsov and Shumskaya, 2013), (Larin, 1994), (Larin and Tunik, 2015), (Lebedev, 1992), (Mhaskar, Liu and Christofides, 2013), (Tanaka and Muller, 1990), (Tao, 2014). These issues are very important in navigation problems (Ahn et.al, 2003), Grip et.al, 2012), (Larin, 1999), (Larin and Tunik, 2012, 2013, 2015), (Lebedev, 1992), in particular, in identifying failed rate gyros (RG) and accelerometers in navigaton system (Deyst et.al., 1981), (Lebedev, 1992). In what follows, we will use the results from (Chow and Willsky, 1984), (Desai, Deckert and Deyst, 1979), (Deyst et.al., 1981), (Lebedev, 1992) to show how it is possible to identify a failed RG using computational procedures based both on analytical redundancy methods and on procedures similar to Kalman filter (Lee, 1964). Below we will detail the case where five RGs are used to measure the angular velocity of an object (Lebedev, 1992) and will show that it is possible to simplify the computing procedures in this case (to use only procedures of calculating the determinant of some matrix or its condition number).

5.1 Solving the Sensor Failure Identification Problem by Means of the Traditional Approach

First of all, we should mention that the sensor failure identification problem can only be solved in case of sensor redundancy. This is the traditional approach to the solution of the problem of identification of a failed sensor. Simple hardware redundancy, where n identical sensors are used instead of one sensor, is the best known. A failed sensor is determined by pairwise comparison of readings of individual sensors, and various procedures of detecting it (Johnson, 1989) are applied in case of a mismatch. Note that scalar values are compared in these procedures. And the cost, dimensions, mass, and power consumption of the navigation system sharply increase.

To avoid this, the studies (Chow and Willsky, 1984), (Desai, Deckert and Deyst, 1979), (Deyst et.al., 1981), (Potter and Suman, 1977) propose the analytical redundancy principle, which uses the fact that vector values (rather than scalar ones) are subject to measurement in navigation systems. This principle is based on using the readings of a system of sensors with minimum possible redundancy to calculate a parity vector such that the difference between components of the measured vector y with redundant number of components and parity vector

p is zero in the absence of measurement noise. If one of the redundant sensors fails, this difference will be nonzero, which is indicative of an error. Noteworthy is that in case of analytical redundancy, sensors are oriented on the object not orthogonally according to the trihedron of gyro instrument, but at some angles with respect to the body frame (the angles can be found by a special technique (Desai, Deckert and Deyst, 1979), (Potter and Suman, 1977), not considered in this monograph). Let output $y \in R^n$ of the measurement module be related to the measurand $\omega \in R^{n-m}$ and error vector $e \in R^n$ by the following linear stationary relation:

$$y = A\omega + e \, , \tag{5.1}$$

where m is the number of redundant sensors and constant matrix $A \in R^{n \times m}$ is defined by the spatial arrangement of sensors on the mobile object (Chow and Willsky, 1984), (Deyst et.al., 1981), (Johnson, 1989), (Potter and Suman, 1977). In case of failure of the i th sensor, error e_{*i} is assumed to have the form

$$e_{*i} = [0 \ldots 0 \; \rho_i \, 0 \ldots 0]' \, . \tag{5.2}$$

Hereinafter, prime denotes transposition. Let p be the set of parity vectors with respect to y, which are described by the relation (Deyst et.al., 1981):

$$p = Vy \, , \tag{5.3}$$

where matrix $V \in R^{m \times n}$ satisfies the condition

$$VA = 0_{m \times m} \, . \tag{5.4}$$

Substituting (5.1) into (5.2) and taking into account (5.3), we obtain

$$p = Ve \, , \tag{5.5}$$

i.e., in case of operational sensors, vector p will be equal to zero vector; and in case of failure (5.2), the corresponding component of vector p will be nonzero, which will be indicative of a failed sensor. The algorithm of finding matrix $V \in R^{m \times n}$ that satisfies relation (5.3) is proposed in (Potter and Suman, 1977).

In (Lebedev, 1992), another approach to detect a failed sensor is proposed. Following (Lebedev, 1992), we will construct an orthogonal projector G that maps error vector e_* (5.2) into the corresponding minimum-norm vector e_0, i.e.,

$$e_0 = Ge_* \, . \tag{5.6}$$

Assuming that the error vector has the structure (5.2), we propose the following algorithm to find the i th number of failed sensor. Denote columns of matrix G by $k=1,\ldots,n$, i.e, $G=[g_1 \quad \ldots \quad g_n]$. Find the minimum value of the functional:

$$J(k)=\|q_k\|^2, \quad q_k=\rho_k g_k - e_0; \tag{5.7}$$

$$\rho_k=\frac{g'_k e_0}{\|g_k\|^2}. \tag{5.8}$$

Hereinafter, $\|\cdot\|$ is vector norm, i.e., $\|x\|^2=x'x$. The value of $k=k_*$ for which $k_*=\arg\min J(k)$ corresponds to the number of failed sensor whose error ρ_k is defined by relation (5.8). Algorithm implementation assumes finding one of possible variants of the error vector e_* (5.2), which can be found using the value of vector (5.5). In (Lebedev, 1992), this problem is proposed to be reduced to a linear programming problem.

5.2 An Algorithm Using Singular Value Decomposition

As an alternative to the above approaches, we will consider an algorithm that uses singular value decomposition of the matrix A appearing in (5.1).

Let singular value decomposition (Voyevodin and Kuznetsov, 1984) of the matrix A appearing in (5.1) have the form

$$A=U\begin{bmatrix}\Sigma \\ 0\end{bmatrix}W', \tag{5.9}$$

where U, W are $n\times n$ and $(n-m)\times(n-m)$; orthogonal matrices, respectively; Σ is an $(n-m)\times(n-m)$ diagonal matrix with diagonal elements greater than zero. Let us divide matrix U into blocks $U=[U_1 U_2]$, with the size of block U_2 being equal to $n\times m$. Since matrix U is orthogonal, i.e.,

$$U^T U=\begin{bmatrix}U_1^T \\ U_2^T\end{bmatrix}[U_1 U_2^T]=\begin{bmatrix}I & 0 \\ 0 & I\end{bmatrix}, \tag{5.10}$$

after premultiplying matrix A in (5.1) by U_2^T, we will find an analog of (5.4)

$$U_2^T A=0. \tag{5.11}$$

In (5.10) and in what follows, I and 0 are an identity and a zero matrix, respectively, of the corresponding dimensions. Multiplying the left- and right-hand sides of Eq. (1.1) by U_2 and taking into account (5.11), we obtain an analog of (5.7):

$$p=U_2^T y=U_2^T e. \tag{5.12}$$

Vector e_0 having the minimum norm and satisfying (5.12) can be expressed in terms of vectors y, e as follows:

$$e_0 = U_2 U_2^T y = U_2 U_2^T e \,. \tag{5.13}$$

According to (5.13)**,** projector G that appears in (5.6) has the form

$$G = U_2 U_2^T \,. \tag{5.14}$$

If we denote by U_{2k}^T, $k = \overline{1, n}$, the columns of matrix U_2^T, i.e., $U_2^T = \left[U_{21}^T U_{22}^T \ldots U_{2n}^T\right]$, then the columns of projector G in (5.14) can be written as

$$g_k = U_2 U_{2k}^T \,. \tag{5.15}$$

Hence, $\|g_k\|^2 = U_{2k} U_2^T U_2 U_{2k}^T = U_{2k} U_{2k}^T$ and in what follows

$$\rho_k = \frac{U_{2k} U_2' U_2 U_2' y}{U_{2k} U_{2k}'} = \frac{U_{2k} U_2' y}{U_{2k} U_{2k}'} \,. \tag{5.16}$$

Thus, substituting expressions (5.13), (5.15), and (5.16) into (5.5) yields

$$J(k) = \|g_k\|^2 = y' U_2 (\alpha_k U_{2k}' U_{2k} - E)^2 U_2' y = y' U_2 (E - \alpha_k U_{2k}' U_{2k}) U_2' y, \; \alpha_k^{-1} = U_{2k} U_{2k}' \,. \tag{5.17}$$

Using the singular value decomposition (5.9) allows, according to relations (5.16) and (5.17), explicit expression (in terms of the results of observation y) of functional (5.5) and sensor's error estimate (5.6). Thus, using singular value decomposition of a matrix substantially simplifies the computing procedure of failure identification.

Note that to find the orthogonal matrix U, it is possible to use not only singular value decomposition (5.9) but also a simpler computing procedure, namely, QR decomposition (Grewal and Andrews, 1993).

Example 5.1

Let us illustrate the efficiency of using relation (5.17) in the sensor failure detection procedure. To illustrate the above algorithm, we will use data from example (Deyst et.al., 1981), which considers the case of an antisymmetric arrangement of five RGs on the surface of a cone with the cone angle half of which is equal to 54.7^o. In this case, matrix A in (5.1) is defined as follows (Deyst et.al., 1981):

$$A^T = \begin{bmatrix} 0.97204 & -0.60075 & 0 & 0 & -0.60075 \\ 0 & -0.77653 & 0.47992 & -0.47992 & 0.77653 \\ -0.23482 & -018997 & 0.87731 & 0.87731 & -0.18997 \end{bmatrix}.$$

Assume that ω in (5.1) is $\omega = [1 \quad 2 \quad 3]'$. The quantity appearing in (5.2) $\rho_i = 1$ for $i = \overline{1.5}$, i.e., all possible failure cases are considered. Block U_2 of matrix U (see (5.9) and (5.10)) has the form

$$U_2^T = \begin{bmatrix} -0.0752 & -0.4300 & -0.6205 & 0.5740 & 0.3082 \\ 0.6280 & 0.4638 & 0.1225 & 0.2656 & 0.5523 \end{bmatrix}.$$

Table 1 presents the results of computation with the use of relation (5.17), which envelopes all five variants of sensor failure. In Table 5.1, k_j denotes the number of failed sensor and $J(k_j)$ are calculated by formula (5.17). As seen, the minimum (zero) values of $J(k)$ are only on the diagonal of this table.

Table 5.1

k / $J(k)$	1	2	3	4	5
$J(k_1)$	0.0000	0.1382	0.3618	0.3618	0.1382
$J(k_2)$	0.1382	0.0000	0.1382	0.3618	0.3618
$J(k_3)$	0.3618	0.1382	0.0000	0.1382	0.3618
$J(k_4)$	0.3618	0.3618	0.1382	0.0000	0.1382
$J(k_2)$	0.1382	0.3618	0.3618	0.1382	0.0000

Thus, we may state that the failed sensor detection procedure described by relation (5.17) is efficient.

5.3 Algorithm of Sensor Failure Identification under Measurement Noise

Let us return to system (5.1); however, like in (Deyst et.al., 1981), (Larin, 2015) we will consider that measurement of angular velocity is accompanied with random noise w, namely:

$$y = A\omega + e + w, \tag{5.18}$$

where w is a vector of random quantities with the characteristics

$$< w >= 0, \; < ww' >= \sigma^2 I .$$

Hereinafter, $< >$ denotes expectation.

Thus, we can consider vector y as an available vector of actual measurements that contain errors.

The solution of overdetermined system

$$Ax = b \tag{5.19}$$

by means of the least squares method is known to be written as

$$\bar{x} = (A^T A)^{-1} A^T b. \tag{5.20}$$

Consider the recurrent solution scheme (5.20) for system (5.19). Let the first k equations of system (5.19) yield the solution estimate $\bar{x}_k$:

$$\bar{x}_k = (A_k^T A_k)^{-1} A_k^T b_k ,$$

where matrices A_k and vector b_k are defined by the first k equations (5.19).

$$\bar{x}_k = (A_k^T A_k)^{-1} A_k^T b_k .$$

In this case, to obtain the sequence of solutions x_k, it is possible to use the recurrent procedure (Deyst et.al., 1981).

Let us consider the problem of deriving $k+1$ estimate of vector x, following Sec. 3 (Lee, 1964). Assume that a new $k+1$ measurement of signal is available, i.e., the system will be supplemented with one more equation:

$$A_{k+1}x = b_{k+1}, \quad A_{k+1} = \begin{bmatrix} A_k \\ \alpha' \end{bmatrix}, \; b_{k+1} = \begin{bmatrix} b_k \\ z_{k+1} \end{bmatrix}.$$

According to (3.45), (3.46), and (3.48) in (Lee, 1964) the optimal estimate $\bar{x}_{k+1}$ of vector x obtained as a result of the $k+1$ measurement of signal is related to the optimal estimate $\bar{x}_k$ obtained as a result of using k measurements as follows:

$$\bar{x}_{k+1} = \bar{x}_k + \tilde{P}_k \alpha (\alpha^T \tilde{P}_k \alpha + \sigma^2)^{-1} (z_{k+1} - \alpha^T \bar{x}_k) , \tag{5.21}$$

$$\tilde{P}_{k+1} = \tilde{P}_k - \tilde{P}_k \alpha (\alpha^T \tilde{P}_k \alpha + \sigma^2)^{-1} \alpha^T \tilde{P}_k ,$$

where $\bar{x}_k = (A_k^T A_k)^{-1} A_k^T b_k \,. P_k = (A_k^T A_k)^{-1} \; \tilde{P}_k = P_k \sigma^2$

As a matter of fact, relations (5.21) describe the Kalman filter algorithm.

As is mentioned in Sec. 8.3.1.2 (Grewal and Andrews, 1993), the Kalman filter considered above allows calculating also the parameters that make it possible to detect a failure. Denote

$$Y_{\nu k} = (\alpha^T \tilde{P}_k \alpha + \sigma^2)^{-1} ,$$

$$\nu_k = z_{k+1} - \alpha^T \bar{x}_k ,$$

(the notation coincides with that in (5.21)).

In this case, the likelihood function has the form (relation (8.25) (Grewal, Weill and Andrews, 2001)):

$$S(\nu_k) = exp\left(-\frac{1}{2} \nu_k' Y_{\nu k} \nu_k \right).$$

This function is associated with the statistics (relation (8.27) (Grewal, Weill and Andrews, 2001))

$$k_s = \frac{\nu'_k Y_{\nu k} \nu_k}{\ell} . \tag{5.22}$$

In (3.5), ℓ is the dimension of vector ν_k. In (Grewal and Andrews, 1993), (Grewal, Weill and Andrews, 2001) it is mentioned that if the models are chosen correctly in the filtration algorithm, noise is centered normal random processes, then k_s has χ^2-distribution.

As these assumptions are not always true in practice, the following procedure is proposed in (Grewal and Andrews, 1993), (Grewal, Weill and Andrews, 2001) for failure detection. Pick some value $k_{s\max}$ that determines the variation interval of k_s corresponding to system's normal operation. If $k_s > k_{s\max}$, the system is considered to have a failure.

Let us show how relation (5.22) can be used in the problem of determining the number of failed RG, i.e., when $n-m=3$ in (1.1). Let us supplement procedure (5.21) with the calculation of k_s defined in (5.22). Thus, we will obtain the sequence of values $k_{s1}, k_{s2}, \ldots, k_{s,n-3}$. This sequence allows finding the values $\mu_1 = \frac{k_{s2}}{k_{s1}}\ \mu_2 = \frac{k_{s3}}{k_{s2}}, \ldots$. To demonstrate the essence of the approach, we assume that $\mathrm{i} = \mathrm{n}$ in (5.2), i.e., the equation corresponding to the failed RG is the last in system (5.18). It is obvious that for a sufficiently small noise level w in (5.18), the last value in the sequence μ_j will be the maximum in this case. In turn, this will indicate that the failed RG has the number n. When the number of failed RG is $i\ (i \le n)$, it is possible to generalize the procedure described above by including the following steps in it. At each step, cyclic transposition of the equations in system (5.18) is carried out: the last equation becomes the first, the first one becomes the second, etc. After this transposition, the corresponding value of sequence μ_j is calculated at the same step (with the number $r\,(0 \le r \le n-1)$). Denote the last value in this sequence by μ_r^*. The number r of step at which the maximum value μ_r^* is obtained is related to the number of failed RG as follows:

$$i = n - r . \tag{5.23}$$

Let us illustrate this procedure by an example.

Example 5.2

As well as in the previous example, matrix A in (5.18) is as follows (Deyst et.al., 1981):

$$A^T = \begin{bmatrix} 0,97204 & -0,60075 & 0 & 0 & -0,60075 \\ 0 & -0,77653 & 0,47992 & -0,47992 & 0,77653 \\ -0,23482 & -0,18997 & 0,87731 & 0,87731 & -0,18997 \end{bmatrix}$$

The angular velocity $\omega = [1\ 2\ 3]'$ and the value of ρ_i in (5.2) is accepted to be equal to 1. To generate vector w in (5.18), procedure rand.m from MATLAB package is used, $\sigma = 0.1$. The results of modeling are presented in Table 2. As well as in the previous example, all the cases of sensor failure are considered. As follows from Table 5.2, the results of the experiment confirm relations (5.23), i.e., the algorithm considered above, which is based on the computing procedures of Kalman filter, allows specifying the number of failed RG. This number corresponds to the maximum value of parameter μ_j^* in the corresponding row.

Table 5.2

i	μ_0^*	μ_1^*	μ_2^*	μ_3^*	μ_4^*
1	0.2849	0.0303	0.3440	1.4821	367.6578
2	0.0091	0.2664	1.1265	24.3626	0.6189
3	0.6098	3.5288	5.7698	0.0232	0.1194
4	2.8948	10.2930	0.0438	0.0988	0.5483
5	58.3723	0.1144	0.0634	0.4445	2.0705

For example, in case of failure of the first RG $(\mathrm{i} = 1)$, the maximum number from the first row of Table 5.2 corresponds to column μ_4^*. Hence, the obtained value $r = 4$ and according to (5.23) $i = 1$.

5.4 Algorithm of Sensor Failure Identification by Estimating the Systematic Error of the Sensor.

Simple inertial navigation systems (INS) (Barbour et.al, 2008), (Barbour, 2011), (Schmidt, 2011) are used, in particular, in cheap unmanned aerial vehicles (UAV) (see, for example (Ahn et.al., 2003)). However, such INS, under considerable time of independent operation, may not provide sufficient accuracy of navigation parameters. Therefore, it is expedient to integrate these INS with satellite global positioning system (GPS) (Grewal and Andrews, 1993), (Schmidt, 2011), (Schmidt and Phillips, 2011 b), i.e., to consider them as an element of a GPS/INS navigation complex (Phillips and Schmidt, 1996), (Schmidt, 2008, 2011), (Schmidt and Phillips, 2011 a, 2011 b). It is important that additional channels of navigation information (magnetometers, pressure altimeters, etc. (Coopmans, 2009), (Coopmans, Chao and Chen, 2009), (Kortunov et.al, 2008), (Kharchenko et.al., 2013, 2014)) are often used in UAV along with GPS signals to correct the results of INS operation.

As well as in (Larin and Tunik, 2012), below we will consider a simple algorithm of integration of GPS, INS, magnetometer, and pressure transducer. We will also introduce a number of simplifying assumptions (disregard the Coriolis acceleration and consider the motion in a rectangular frame). This is because, on the one hand, sensors in the considered INS are not high-precision and on the other hand, such systems are supposed to be used at objects that move slow at rather small distances.

It is important that unlike (Larin and Tunik, 2012), along with the problem of compensating for the systematic RG error, below we will consider the possibility to increase the reliability of system operation by increasing the number of RGs. In this case, the problem is to detect a failed RG, which in turn allows eliminating its readings from the consideration.

We have considered an example that shows the efficiency of the proposed algorithms.

Detecting a Failed RG

We have assumed above that the INS contains three RGs, i.e., does not contain redundant RGs. However, it may appear expedient (in order to enhance system's reliability) to increase the number of RGs (see, for example (Deyst et.al., 1981)) to make it possible to exclude the readings of failed RG. Thereupon, we will consider the problem of identification of failed RG. Thus, let the output of measurement module of the RGs be related to the measured angular velocity ω by the following relation (an analog of relation (5.1)):

$$y = A\omega + e\,, \tag{5.24}$$

where vectors $y,\ e \in R^n$ $\omega \in R^{n-m}$, constant matrix $A \in R^{n\times(n-m)}$, i.e., m is the number of redundant RGs. In case of failure of the i th sensor, error e appearing in (5.24) is supposed to have the form

$$e_{*i} = \left[0,\ldots,a_i,0,\ldots,0\right]^T . \tag{5.25}$$

The nonzero element a_i in vector e_{*i} defined in (5.25) is at the i th place. To identify failed RG, it is necessary to find the value of subscript i and find a_i if necessary.

Let according to (5.24) the value of ω be defined by the following relation:

$$\omega = Z(y-e)\,,\ Z = (A^T A)^{-1} A^T . \tag{5.26}$$

If vector e is defined by (5.25), then

$$Ze = a_i z_i , \tag{5.27}$$

where z_i is the i th column of matrix Z defined by (5.27).

Noteworthy is that the algorithm of compensating for RG systematic error described in Sec. 3 allows deriving estimate $\delta\widehat{c}$ of vector δc. Given this estimate, it is possible to determine vector e_{*i}, i.e., the value of subscript i, and estimate $\widehat{a}_i$ of a_i.

Let vector δc be caused by the systematic error of the i th RG defined by (5.25), i.e., $\delta c = a_i z_i$ according to (5.27).

Assuming that estimate $\delta\widehat{c}$ is known, we consider the problem of finding the values of $\widehat{a}_i$ and of subscript i.

Let estimate $\delta\widehat{c}$ be given. The problem is to choose vector z_i (find subscript i) and select $\widehat{a}_i$ that would approximate $\delta\widehat{c}$ in the best way. In other words, it is necessary to find z_i and $\widehat{a}_i$ that minimize the value of the following residual (discrepancy):

$$\begin{aligned} dis = \left\| \delta\widehat{c} - \widehat{a}_i z_i \right\|^2 = (\delta\widehat{c} - \widehat{a}_i z_i)^T (\delta\widehat{c} - \widehat{a}_i z_i) = \\ = \widehat{a}_i^2 z_i^T z_i - 2\widehat{a}_i z_i^T \delta\widehat{c} + \delta\widehat{c}^T \delta\widehat{c}. \end{aligned} \tag{5.28}$$

According to (5.28), the value of $\widehat{a}_i$ that minimizes dis is defined by the following relation:

$$\widehat{a}_i = z_i^T \delta\widehat{c} / z_i^T z_i . \tag{5.29}$$

Thus, the problem of choosing subscript i reduces to selecting vector z_i $(i = 1, \ldots, n)$ that minimizes (5.28) provided that $\widehat{a}_i$ is defined by (5.29), i.e.,

$$i = \arg\min_i \left\| \delta\widehat{c} - \widehat{a}_i z_i \right\| . \tag{5.30}$$

Example 5.3

Assume that frame xyz is oriented as follows: axis x to the South, axis y to the East, and axis z to the Zenith. The beginning of coordinates of this system (point O) is located on the Earth surface at 45^0 north latitude. In this frame, the object circles in the plane xy with period $T = 300\ c$ sec and velocity $V = 60$ m/sec. During the motion, its orientation is described by the following time dependence of the Euler angles: $\psi = \dfrac{2\pi t}{T}$, $\vartheta = 0$, $\varphi = 0{,}3 sin(10\psi)$. Projections of the angular velocity of the object onto the axes of body frame $(x'y'z')$ (without regard for the angular velocity of the Earth) are described by Eqs. (2.8). This data is used to model the readings of RGs installed at the object. As well as in (Deyst et.al., 1981), it is supposed that five RGs are mounted at the object. The form of matrix A in (5.24) is the same as in the previous examples:

$$A^T = \begin{bmatrix} 0{,}97204 & -0{,}60075 & 0 & 0 & -0{,}60075 \\ 0 & -0{,}77653 & 0{,}47992 & -0{,}47992 & 0{,}77653 \\ -0{,}23482 & -0{,}18997 & 0{,}87731 & 0{,}87731 & -0{,}18997 \end{bmatrix}$$

Matrix Z corresponding to it in (5.26) has the form

$$Z = \begin{bmatrix} 0{,}5832 & -0{,}3605 & 0{,}0000 & 0{,}0000 & -0{,}3605 \\ -0{,}0000 & -0{,}4659 & 0{,}2880 & -0{,}2880 & 0{,}4659 \\ -0{,}1409 & -0{,}1140 & 0{,}5264 & 0{,}5264 & -0{,}1140 \end{bmatrix}.$$

The angular velocity vector $\omega = [\omega_1 \quad \omega_2 \quad \omega_3]^T$ obtained at time t_k is supplemented with the vector of systematic error $\delta c = [\sigma_1 \quad \sigma_2 \quad \sigma_3]^T$ defined by (5.27) and with 3×1 vector n_ω, which models random errors. Elements of vector n_ω are random numbers uniformly distributed with zero expectation and variance σ_ω. The kinematic equations are integrated according to the algorithms described in Sec. 2 (see also (Larin, 1999, 2001), (Larin and Tunik, 2012)). (The Coriolis acceleration is disregarded). The errors of accelerometers' readings are assumed to be uniformly distributed random numbers with zero expectation and variance σ_a.

The errors of magnetometer and altimeter were modeled similarly. For the magnetometer we supposed that errors have the same variance σ_m for each coordinate. Altimeter's readings are accompanied by errors with variance σ_v.

As well as in [26], we suppose that INS operates with frequency 20 Hz, i.e., time interval $\Delta t = 5\cdot 10^{-2}\,c$. Readings of RGs and accelerometers are accompanied by noise ($\sigma_\omega = 0.05\ deg/sec,\ \sigma_a = 10^{-2}\,m/\sec^2$). The systematic error of RGs $\delta c = a_1 z_1$, i.e., according to (5.25) it is caused by the systematic error of the first RG, which we will consider as a failed one. We assume that $a_1 = 0{,}1$.

INS operation is corrected every 2 sec. Like in (Larin and Tunik, 2012), we suppose that GPS will provide the following errors of object's velocity and coordinates: 0.1 m/sec and 50 m, respectively. The variances of errors of magnetometer (determining the coordinates of vector $\bar{m}$) and of altimeter are assumed the following: $\sigma_m = 0{,}0524\ rad,\ \sigma_h = 1\ m$. Based on this data, we accepted the following values for the Cholesky multipliers q_k, η_k:

$$\bar{q}_k = 10^{-3} diag\left\{\alpha_q I, \beta_q I, \gamma_q I, \delta_q I\right\},\ \ \eta_k = diag\left\{\alpha_\eta I_2, \beta_\eta I, \gamma_\eta I, 1\right\}, \qquad (5.31)$$

$\alpha_q = 0.0218$, $\beta_k = 0.25$, $\gamma_q = 0.0063$, $\delta_q = 5 \cdot 10^{-5}$, $\alpha_\eta = 0.0524$, $\beta_\eta = 0.1$, and $\gamma_\eta = 50$. In these expressions, matrices I are 3×3, matrix I_2 is 2×2. We suppose that $g = 9.81\,\text{m/s}^2$, initially (at $(t = 0)$) the object is located on axis y at the distance $\frac{VT}{2\pi}$, i.e., the initial position of the object is described by the vector $r_0 = \begin{bmatrix} 0 & \frac{VT}{2\pi} & 0 \end{bmatrix}^T$. The vector of object's velocity is $v_0 = [-V \quad 0 \quad 0]^T$. The INS initial setting is accompanied by the following errors. The error of initial orientation is defined by the (non-normalized) quaternion $\lambda(0) = [1 \quad 0.005 \quad -0.005 \quad 0.005]^T$, the error of setting of initial values is defined by relative errors $\varepsilon_r = 0.01$ and $\varepsilon_v = 0.01$, i.e., the following values of initial coordinates (r) and velocity (v) are set:

$$r = r_0(1 + \varepsilon_r), \quad v = v_0(1 + \varepsilon_v).$$

Thereupon, the following matrix is accepted as the Cholesky multiplier m_0:

$$m_0 = diag\{\alpha_m I, \ \ \beta_m I, \ \ \gamma_m I, \ \delta_m I\},$$

$$\alpha_m = 0.04, \quad \beta_m = 0.6, \quad \gamma_m = 28.65, \ \ \delta_m = 0.4.$$

As well as in (8.1), matrices I are 3×3. The observation vector z_k is generated according to (5.4) and the 9×12 matrix H has the following structure:

$$H = [H_1 \quad O_3],$$

where $H = \begin{bmatrix} O & I \\ O^T & 1 \end{bmatrix}$, O_3 is a 9×3 zero matrix, O is an 8×1 zero matrix, and I is an 8×8 identity matrix.

Under such initial data, the INS operation was modeled during 20 sec. The results of the modeling (time dependence of the first nine component of the vector x appearing in (5.18)) are presented in Figs. 5.1–5.3.

Figure 5.1 shows the time dependence of the components of vector μ (μ_x, μ_y, μ_z, degrees dimension). Figures 5.2 and 5.3 show components of vectors $\delta r, \delta v$ (dimensions m and m/s, respectively). In Figs. 5.1–5.3, we accept the following notation for the coordinates of vectors $\mu, \delta v, \delta r$: the solid line corresponds to axis x, dashed line to axis y, and dash-and-dot line to axis z.

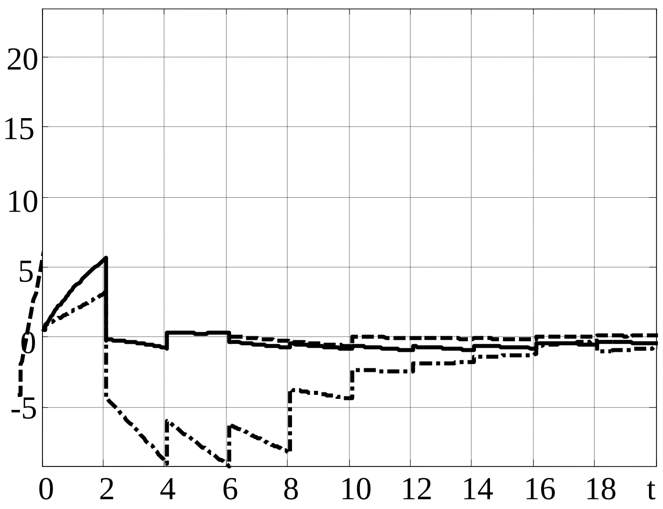


Fig. 5.1

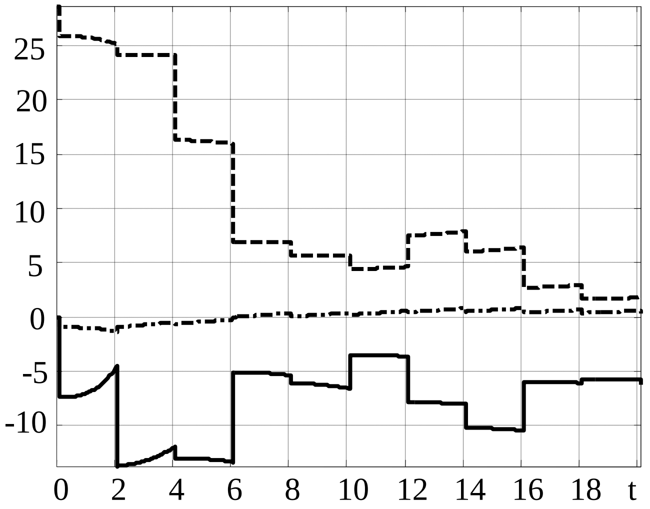
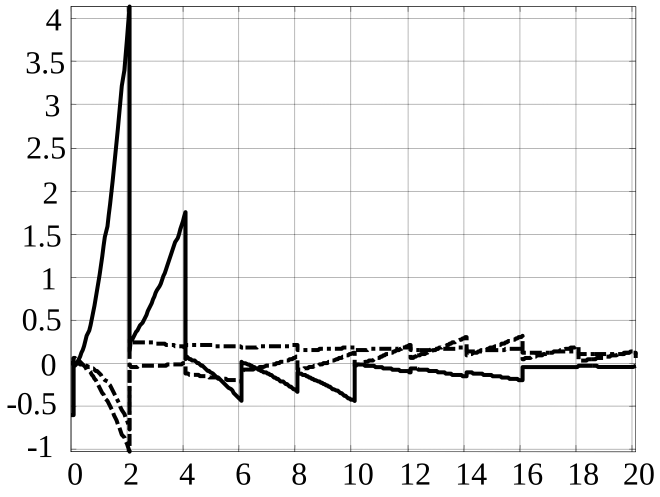


Fig. 5.2



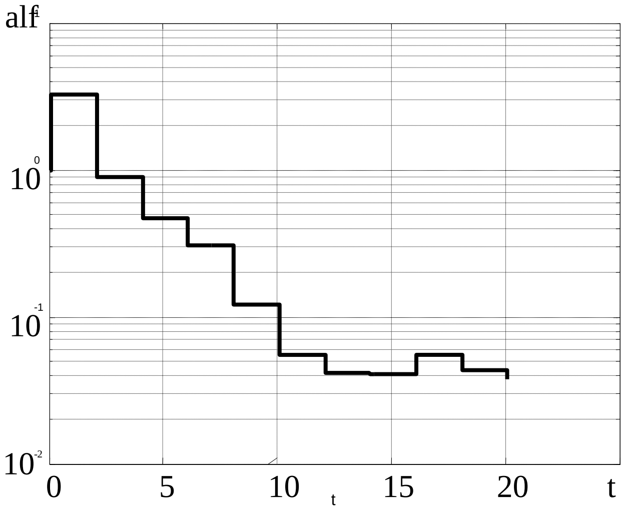


Fig. 5.3

Fig. 5.4

Figure 5.4. allows us to judge about the efficiency of compensation of the systematic error of RGs. It shows the time dependence of $\alpha\ell f = \left\|\|\delta c-\delta\widehat{c}\|/\|\delta c\|\right\|$, ($\delta\widehat{c}$ is the current value of the estimate of systematic error of RGs).

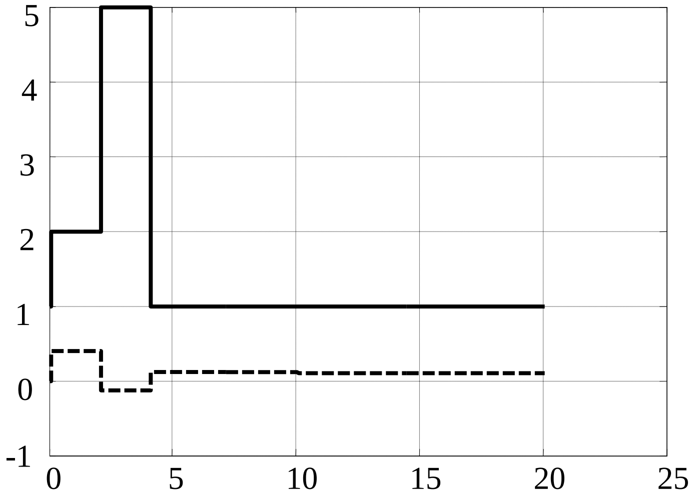


Fig. 5.5

Figure 5.5 presents the values i of the number of failed sensor (solid line) and current estimate $\widehat{a}$ (dashed line), which are defined by relations (5.29) and (5.30). As we can see from Fig. 5.5 (the solid line), after the completion of the transient in the failure identification system, the steady state value of the failure indicator (5.30) is equal to 1, which corresponds to the assumption about failure of the first sensor.

5.5 A Simplified Procedure of Sensor Failure Identification

Let us consider in more detail the case where five RGs are used to measure angular velocity, i.e., where $n=5, m=2$ in system (5.1). In case of such a redundant measurement system $(m=2)$, we can substantially simplify the procedure of detecting a failed sensor. Let us consider the 5×4 matrix corresponding to system (5.1):

$$\tilde{A}=\begin{bmatrix} y & A \end{bmatrix}. \tag{5.32}$$

If we delete one row in matrix $\tilde{A}$, then the rank of the resultant matrix will be equal to 4 if the number of the deleted row is not equal to i and it will be equal to 3 if we deleted the row number i, i.e., the one containing ρ_i^*, according to (5.2). Taking this into account, we will consider the corresponding procedure. Deleting the rows of matrix $\tilde{A}$ one by one, we obtain s 4×4 matrices $\tilde{A}_j,\ j=2,\ldots,5$. Denote the determinants of these matrices by $D_j = det(\tilde{A}_j)$. It is obvious that if $\mathrm{j}=\mathrm{i}$, then $D_j=0$; otherwise $D_j \neq 0$. If we consider not system (5.1) but system (5.18), then in case of rather small σ we may state that the value of d_j in sequence $d_j=|D_j|$ will be minimum for $\mathrm{j}=\mathrm{i}$. Thus, in the five-sensor case under consideration, the procedure of determining the number of failed sensor reduces to generating the sequence $d_j,\ j=1,\ldots,5$, and determining the minimum term of this sequence d_j^*. The values of the subscript of this term correspond to the number of failed sensor.

In general, it is possible to construct a similar algorithm in which the number of failed sensor can be found by using the sequence of not matrix determinants but the conditional numbers of matrices $\tilde{A}_j$ (to find them, singular value decomposition (5.9) can be used (Larin, 1994), (Onishchenko, 1983)). However, such a procedure related to calculating the singular value decomposition for each of the matrices $\tilde{A}_j$ seems to be more labor consuming than the procedure described above, which involves computing the determinants.

Let us illustrate the described procedure on the example.

Example 5.4

The initial data coincides with those accepted in the previous example. The results of numerical modeling of the procedure described above are shown in Table 5.3.

Table 5.3.

j	1	2	3	4	5
1	0.0814	0.6730	1.0075	0.9572	0.5413
2	0.4245	0.1671	0.6949	0.9572	0.8539
3	0.7371	0.3388	0.1890	0.6446	0.8539
4	0.8999	0.9857	0.6949	0.1387	0.4705
5	0.5873	0.9857	1.0075	0.6446	0.0354

This table presents the absolute values of determinants d_j in the $i-\breve{u}$ th row and the minimum value of d_j corresponds to the number of failed sensor. The numbers of columns $(\,j\,)$ correspond to the number of the deleted row in matrix $\tilde{A}$ defined by (5.32). As we can see, the minimum (in each row) value of elements of this table are located on the diagonal, i.e., where $j=i$. In other words, the minimum value of d_j takes place for $i=j$. Thus, the algorithm described above allows us to determine the number of failed sensor.

Note that in this example the results of measurement of useful signal are accompanied by random errors (vector $w\neq 0$ in (5.18)).

We have presented the algorithms to identify a failed sensor and have shown that computing procedures similar to Kalman filter procedures can be used for this purpose. We have considered the case where five sensors are used to measure the angular velocity of the object and have shown that the computing procedures of identification of the failed sensor can be simplified in this case. The efficiency of the proposed algorithms has been illustrated by the examples.

5.6 Summary for Chapter 5

5.6.1. In this chapter several algorithms for identification of the faulty inertial sensors are proposed. They are based on the principle of analytic redundancy. It is necessary to note that the ultimate goal of this chapter is faulty sensor detecting only; meanwhile the problems of system reconfiguration after solving this problem are beyond the content of this chapter.

5.6.2. Two very simple algorithms and two more sophisticated algorithms are proposed in this chapter. The 1st simple algorithm is based on the singular value decomposition; meanwhile the 2nd one uses very simple calculations of certain matrices determinants. They could be applied directly to the redundant sensors, providing the “raw” measurements.

5.6.3. Two other algorithms have to be applied for data processing of the “refined” measurements, i.e. the output data of the integrated SINS. They are based on the sensor bias estimation in presence of sensor noise. The choice of this or that algorithms depends on the preferences of the SINS designer.

5.6.4. The efficiency of these algorithms was proved by mathematical modeling results, which are shown in examples presented in this chapter.

CHAPTER 6 – EXPERIMENTAL STUDIES OF THE INTEGRATED INERTIAL-SATELLITE NAVIGATION SYSTEM

In this chapter, we will describe the results of comprehensive analysis of the integrated inertial-satellite navigation system (IISNS) are described. In the beginning, we will present the results of static laboratory studies using only inertial and satellite navigation signals (Sec 6.1). In the subsequent sections, we will additionally use the readings of magnetometers and altimeter. Sections 6.2, 6.3-6.4, and 6.5 represent the results of IISNS trials by means of a manual rotating motion table, ground tests, and flight trials, respectively To assess the possibilities of practical application of the developed IISNS, the latter was tested under the same conditions and in parallel with standard IISNS IG-500N by SBG-Systems (Sbg-systems.com, 2018), in order to compare their accuracy characteristics.

6.1. Analysis of the Proposed Integrated System under Laboratory Conditions

The initial stage of testing of the integrated inertial-satellite navigation system is executed in laboratory with the use of actual data under steady-state conditions by performing the corresponding experiment whose results are published in (Kharchenko and Ilnytska, 2010).

The following equipment was used during the experiment: inertial measurement unit (IMU) ADIS16362 by Analog Device (Analog.com, 2018 a) and GPS board of OEM-V1 receiver by Novatel (Novatel.com, 2018 a). GPS/IMU data were divided into messages of two types: (i) measurement data from the inertial measurement unit with 100 Hz information output frequency and (ii) measurement data from the GPS receiver with 1 Hz information output frequency.

Figure 6.1 represents the scheme of reading and post-processing of data from IMU and GPS receiver. As is seen from the figure, sensor signals arrive at a personal computer (PC) in binary format *.dat, where C++ software is used to read out messages from IMU and GPS receiver and then data are converted into text formats *.txt. The inner time clock of the device is synchronized with global time by means of GPS receiver signals. The GPS receiver data file contains the following: GPS time (seconds of week), XYZ coordinates in ECEF (Earth Centered Earth Fixed) frame, RMS error of coordinates (m), linear velocities with respect to the three axes in ECEF frame, RMS error of the velocity (m/sec), solution status, and solution type of the navigation problem according to (Novatel.com, 2018). The IMU data file contains: time (seconds of week), angular velocities (rad/sec), and acceleration (m/sec^2) with respect to three measurement axes. In such form, they arrive at the Matlab software environment for further processing.

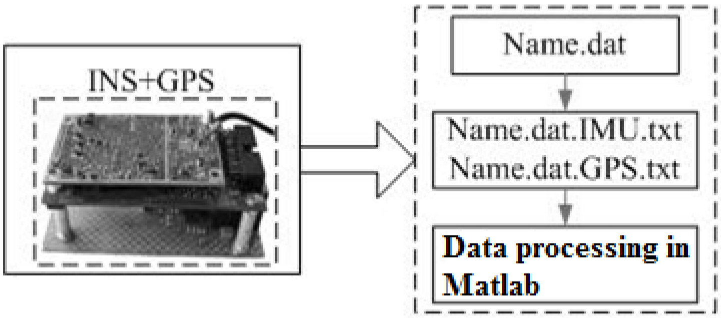


Fig. 6.1. Block diagram of experimental data acquisition and processing.

All the calculations were performed according to the algorithms presented in Secs. 2–3 of this monography in the NED frame with the origin point at the National Aviation University: 50.4391181288º north latitude, 30.4301208368º east longitude and 215.909 m elevation WGS-84. During the experiment, the object was fixed, i.e., its position and attitude did not vary. Measurements of accelerometers and RGs contained errors with nearly zero expectations and with the RMSD no greater than 0.1 º/sec for RGs and no greater than 0.5 m./sec2 for accelerometers. The errors of determining object's coordinates and velocity by the GPS receiver were as follows: expectation ≈ 0.3 m and 10-3 m/sec, RMS no greater than 0.1 m and 0.03 m/sec.

Data with the total time of 300 sec were processed as follows: the first 60 sec the SINS operated independently, the next 120 sec it was corrected every second by the GPS receiver (integrated mode), and the last 120 sec it operated independently again. The results of data processing in the NED frame are presented in Figs. 6.2-6.4.

Figure 6.2 shows time dependence of Euler angles in the NED frame. The solid, dot-and-dash, and dashed lines denote roll, pitch, and heading angles, respectively. The data in Fig. 6.2a are full-scale and in Fig. 6.2b are with vertical axis zoom during integrated ISNS operation. Figure 6.3 shows time dependence of linear velocities in the NED frame. The solid, dot-and-dash, and dashed lines denote the linear velocities along the longitudinal, transverse, and vertical axes, respectively. The data in Fig. 6.2a are full-scale and in Fig. 6.2b are with vertical axis zoom during integrated ISNS operation. Figure 6.4 shows time dependence of object's position. The solid, dot-and-dash, and dashed lines denote the position with respect to axes X, Y, and Z, respectively. The data in Fig. 6.4a are full-scale. Figure 6.4b presents a fragment with 2.5 times vertical axis zoom for a more convenient perception of experimental data, and only the time of integrated operating mode of the ISNS (from 60 to 180 sec) is shown.

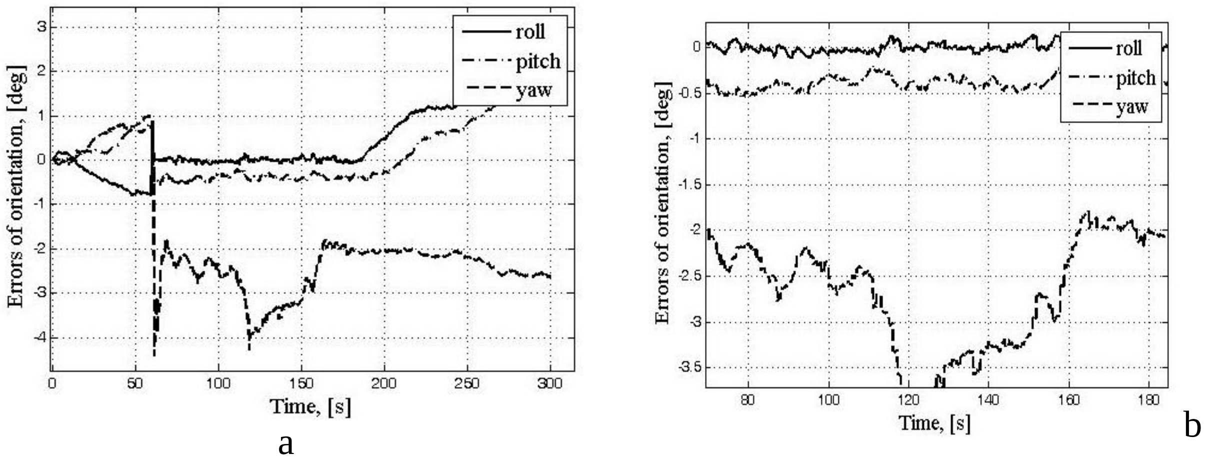


Fig. 6.2. Errors of Euler angles in NED frame: (a) without zoom, (b) with 2.5 times vertical axis zoom

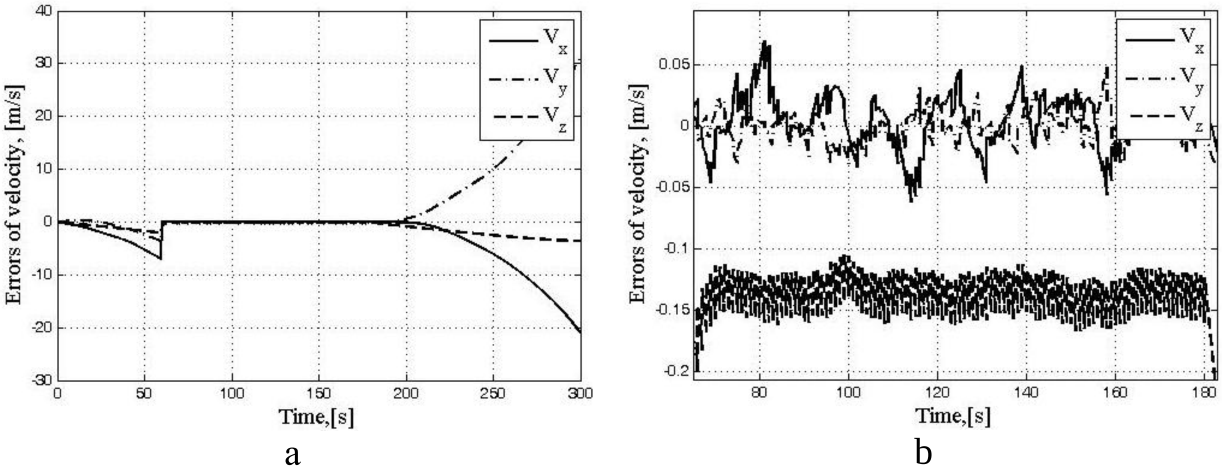


Fig. 6.3. Velocity errors in NED frame: (a) without zoom, (b) with 2.5 times vertical axis zoom

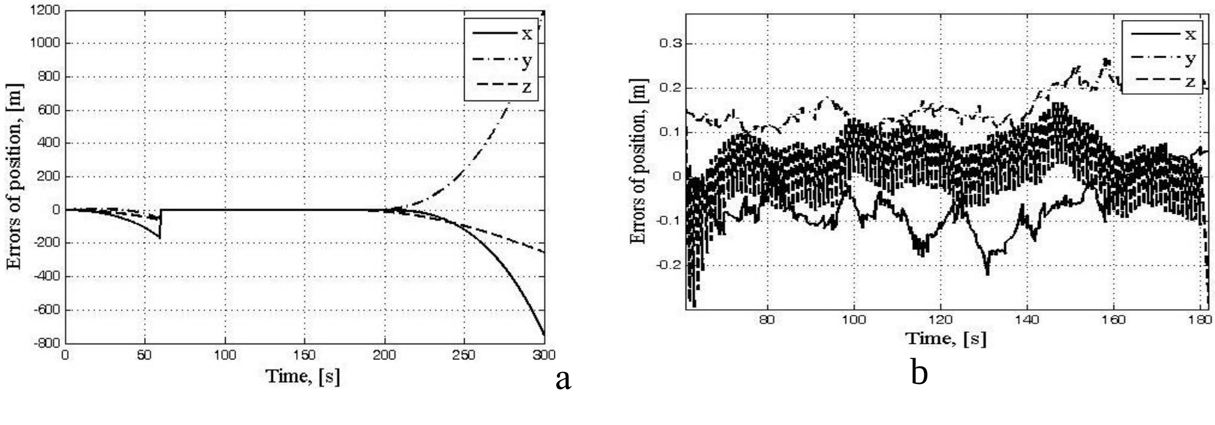


Fig. 6.4. Coordinate errors in NED frame: (a) without zoom, (b) with 2.5 times vertical axis zoom

As is seen from Figs. 6.2-6.4, the proposed algorithm of INS correction is rather efficient and is successful even in case of significant noise of inertial sensors.

The errors in steady state mode are of the following order: attitude error - no greater than 0.1° for roll and pitch angles, velocity error - less than 0.05 m/sec in the horizontal plane and about 0.15 m/sec on the vertical axis, coordinate error - basically no greater than 0.25 m.

No considerable improvement is observed for the heading angle during the time of SINS correction based on GPS signals. This may be due to the fact that RGs sensitivity is rather low and their drift exceeds the Earth rotation velocity. Therefore, to determine the heading angle more accurately, correction should be done based on signals of not only GPS but magnetometer as well.

6.2. Investigating the integrated navigation system by means of a rotating motion table

The purpose of testing the integrated inertial-satellite navigation system (IISNS) by means of an elementary monoaxial manually operated rotating motion table is estimating the errors of coordinates and velocities of the developed system. To assess the possibilities of practical application of the developed IISNS, we compared its accuracy with the accuracy of the commercial-off-the-shelf (COTS) French system IG-500N by SBG-Systems (Sbg-systems.com, 2018).

To estimate the accuracy of coordinates of the developed IISNS, we used a standard trajectory obtained from raw measurements of GPS receiver OEM-V1 that rotates at the stand. These measurements were processed by means of NovAtel GrafNav/GrafNet software by Waypoint Products Group (Novatel.com, 2018 b). It is a powerful software package with a set of adjustable configurations of data processing, which allows obtaining the maximum accuracy with the use of all possible GNSS data. The program uses all-around quality control, which makes solution reliability undoubted. An embedded utility allows public data, as well as files of exact values of satellite ephemeris and values of drift of on-board clocks, to be downloaded from thousands of continuously operating base stations. Data from the mobile receiver and base stations (1 to 8 stations) allow calculating the coordinates with centimeter accuracy.

Reference coordinates can be found by the relative navigation method. To specify certain coordinates, this method uses data of the navigation receiver of base station with a priori known coordinates. In this case, we used the receiver being a part of the experimental facility of monitoring of global navigation satellite systems at the National Aviation University (NAU) (Fig. 6.5). NovAtel GrafNav/GrafNet software (Novatel.com, 2018 b) not only determines coordinates but also allows estimating their accuracy as well as performing visual quality control based on a wide choice of graphic records (number of satellites, their visibility, geometric factors of accuracy aggravation, signal/noise ratio, etc.). The accuracy of finding the coordinates by means of the experimental monitoring facility for global navigation satellite systems at the NAU and NovAtel GrafNav/GrafNet software was within ±12 *cm* depending on experimental conditions (quantity of visible satellites, etc.). This considerably exceeds the accuracy of coordinates found by both satellite and inertial systems. Therefore, the trajectory determined by means of the described relative navigation method can be considered as a reference one to find errors of the IISNSs being compared.

Coordinates of the reference trajectory were obtained as a text file. It contains time in GPS seconds of week, XYZ coordinates in meters, and linear velocities in m/sec with respect to three axes in the ECEF frame. Each point of both reference trajectory and the spatial trajectory being estimated is related to the corresponding timestamp, which makes it possible to estimate the error of instantaneous values of linear velocities and positions when rotating the rotary stand manually.

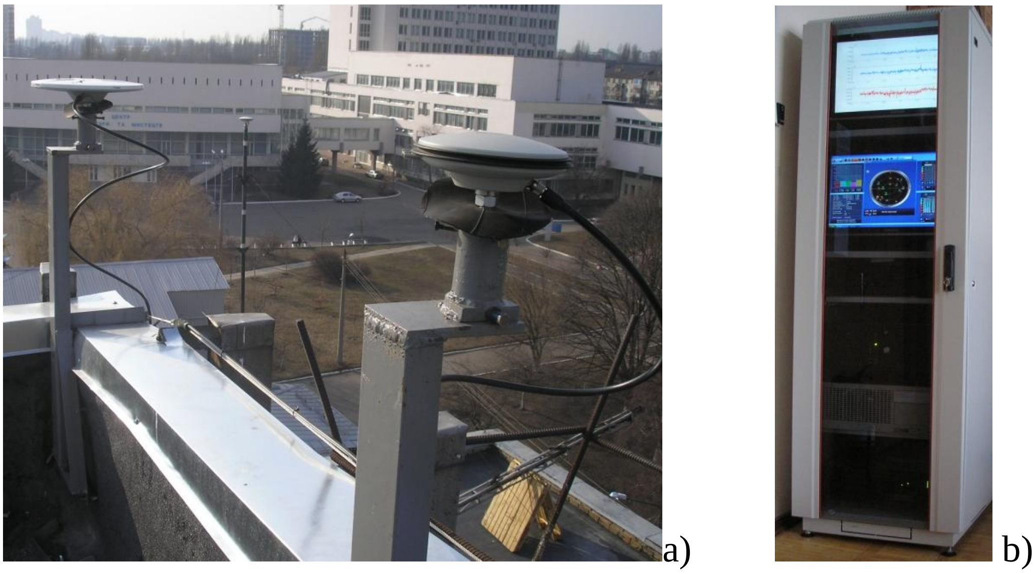


Fig. 6.5. Experimental facility for monitoring of navigation satellite systems on the NAU grounds: (a) GNSS antennas; (b) base station

The results of the experiment are also published in (Kharchenko et.al., 2012 b). The one-axis rotating table with a 2.35 m shoulder (Fig. 6.6) was fixed at a point located on the NAU grounds (coordinates +50.438874248 º north latitude, +30.428294371 º east longitude, and 195.943 m elevation WGS-84).

Figure 6.6 shows the arrangement of the probationary equipment at the rotating motion table, 1 denotes inertial sensors and GPS receiver of both the breadboard model of the proposed ISSN and IG-500N system, 2 denotes laptops and accumulators, and 3 denotes the axis or rotation of the motion table in horizontal plane. Figure 6.7 shows the equipment connection scheme. The following notation is used: 1 is antenna GPS-703-GGG by Novatel; 2 is splitter of

GNSS signal (distributor of signal from GPS antenna) 4G12155-XS4-X by Antcom; 3 is breadboard model of the probationary IISNS; 4 and 6 are laptops for data recording; and 5 is French IISNS available in (Sbg-systems.com, 2018).

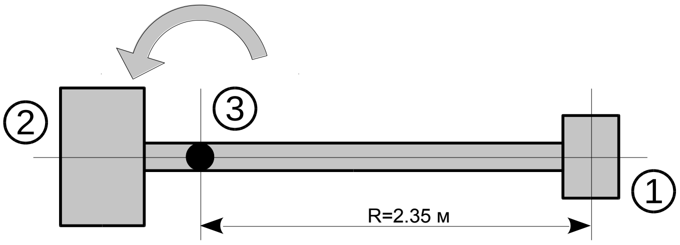


Fig. 6.6. The motion table scheme (top view)

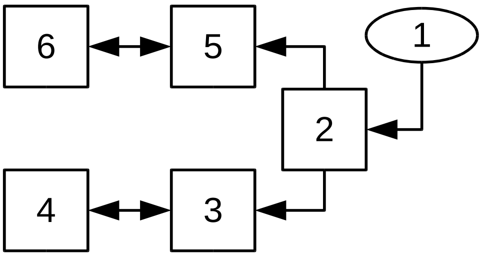


Fig. 6.7. Equipment connection scheme

In the experiment, at first, static measurements of navigation parameters (with fixed motion table) were carried out for about 5 min, then the device was rotated clockwise (top view), i.e., the direction of turn around Z axis of the body frame was positive, the height of the device remaining unchanged. Signals from the navigation satellites received by the antenna through a splitter arrived at navigation receivers of the probationary breadboard model and French IISNS (Sbg-systems.com, 2018).

Data from sensors of the breadboard model of the navigation system were recorded to flash memory of the device, data from the French analog of IISNS were recorded to the laptop by means of the sbgCenterApplication software (Sbg-systems.com, 2018) provided by the manufacturer. Raw GPS measurements of OEM-V1 receiver for deriving a reference trajectory were also saved to the laptop. After the end of the experiment, data post-processing was carried out according to the algorithms presented in Chapters 2 and 3 of the present monograph.

System IG-500N (Sbg-systems.com, 2018) used for comparison provides data about angular orientation of the object, as well as its rate, position, and attitude with data renewal frequency up to 50 Hz. IG-500N includes three-axis RGs, accelerometers, magnetometers, GPS receiver, and a barometric altimeter. According to its engineering specifications (Sbg-systems.com, 2018), attitude RMS of the in statics is 0.5º for roll and pitch angles, and 1º for heading, in dynamics it is 1 º if a GPS signal is present. The accuracy of horizontal positioning is 2.5 m, of vertical is 5.0 m. In case of loss of GPS signals, the accuracy of positioning is aggravated almost the next second. Data are transmitted to the microcontroller through serial interface RS-232.

In what follows, by nominal IISNS mode we will understand teamwork of GNSS and INS. The mode where GNSS signal disappears and INS operates independently is called abnormal mode.

The results of data processing in NED frame are presented in Figs. 6.8-6.11. Relative time of the experiment in seconds is laid along the *oh* axis. Note that time was initially presented in seconds of week, as is customary in GPS. In Fig. 6.8, grey color marks the run time dependence of roll, pitch, and heading angles, calculated by the developed integrated navigation system, and black color marks those calculated by the French system IG-500N.

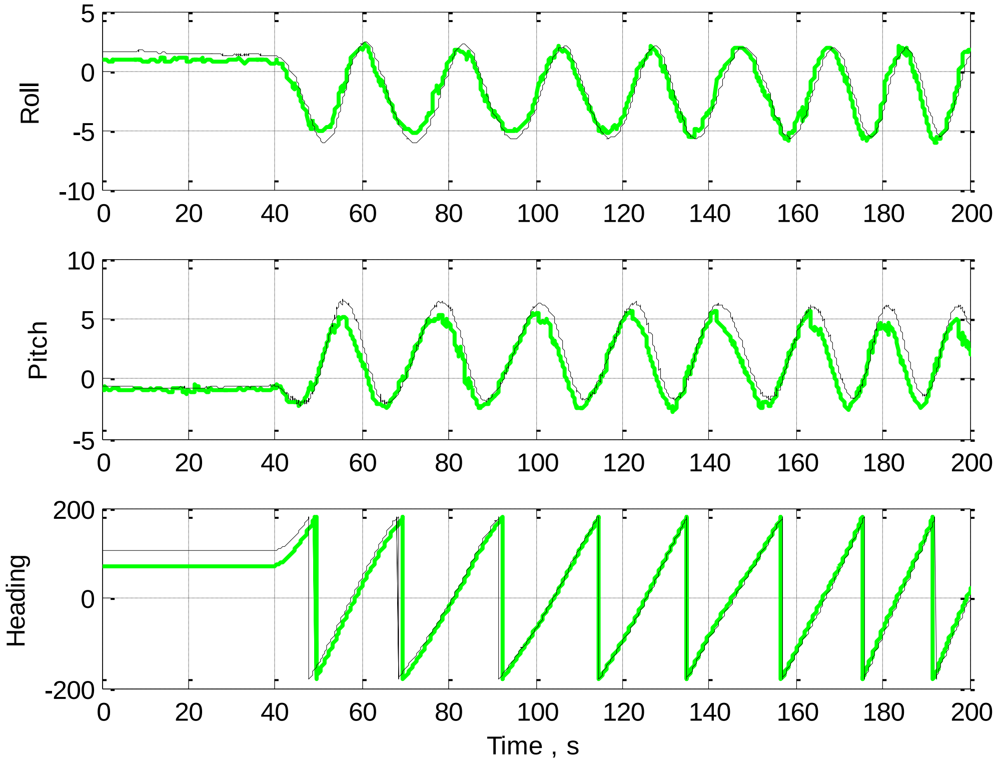


Fig. 6.8. Roll, pitch, and heading angles of the integrated navigation system

Figure 6.9 shows the run time dependence of linear velocities with respect to three axes in the NED frame. The black dots connected by solid line denote the linear velocity obtained from the GPS receiver, and thick grey line denotes linear velocities calculated by the developed integrated navigation system. Figure 6.10 shows the run time dependence of the coordinates of the objects with respect to three axes in NED frame. The notation is the same as in Fig. 6.9.

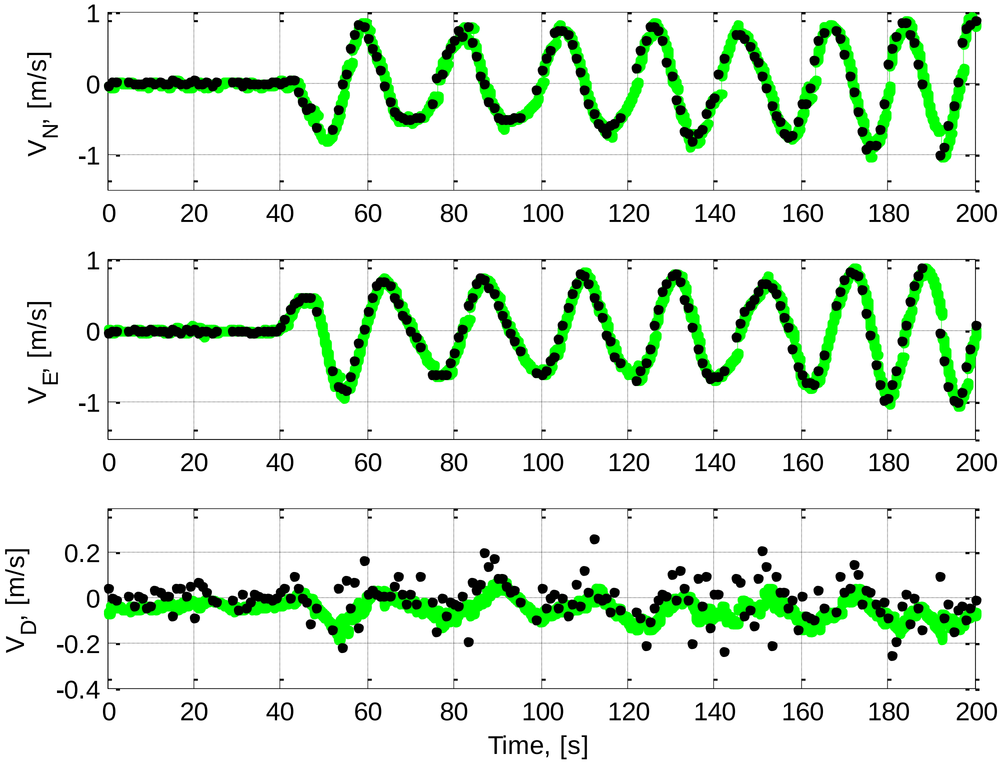


Fig. 6.9. Linear velocity of the integrated navigation system in NED frame

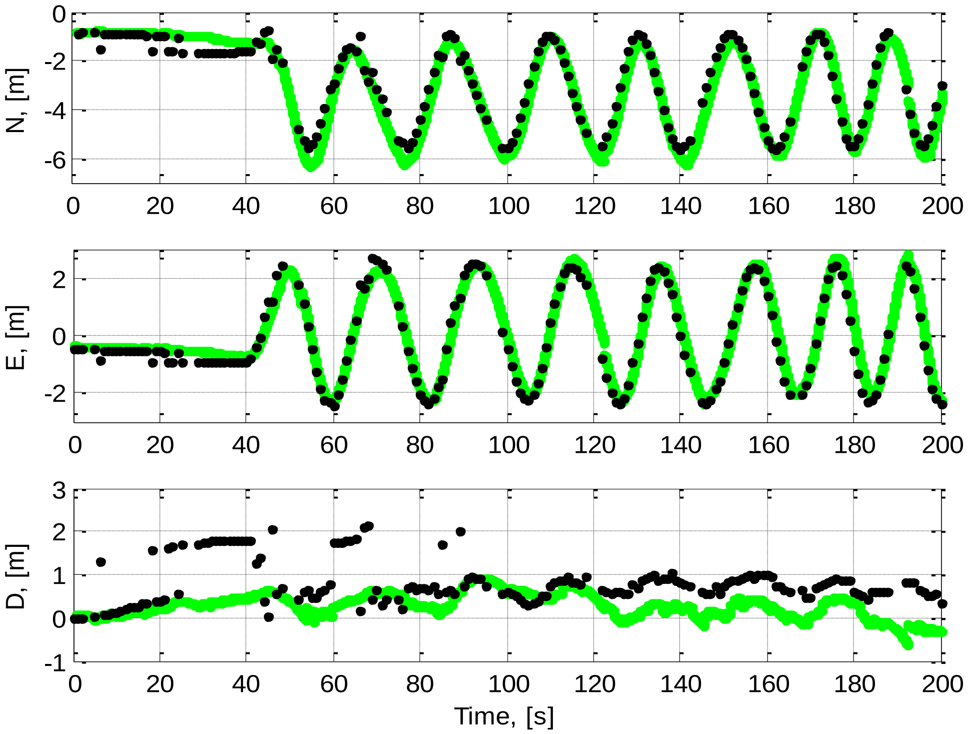


Fig. 6.10. Coordinates of the integrated navigation system in NED frame

From Figs. 6.9–6.10 it can be seen that in the nominal mode the velocities and coordinates virtually coincide. Note also that short loss of GPS signal (during the following intervals: 49-51 sec, 73-75 sec, 96-98 sec, 119-122 sec, 142-145 sec, 165-168 sec, 189-192 sec) did not render a significant influence on the quality of the obtained navigation solution.

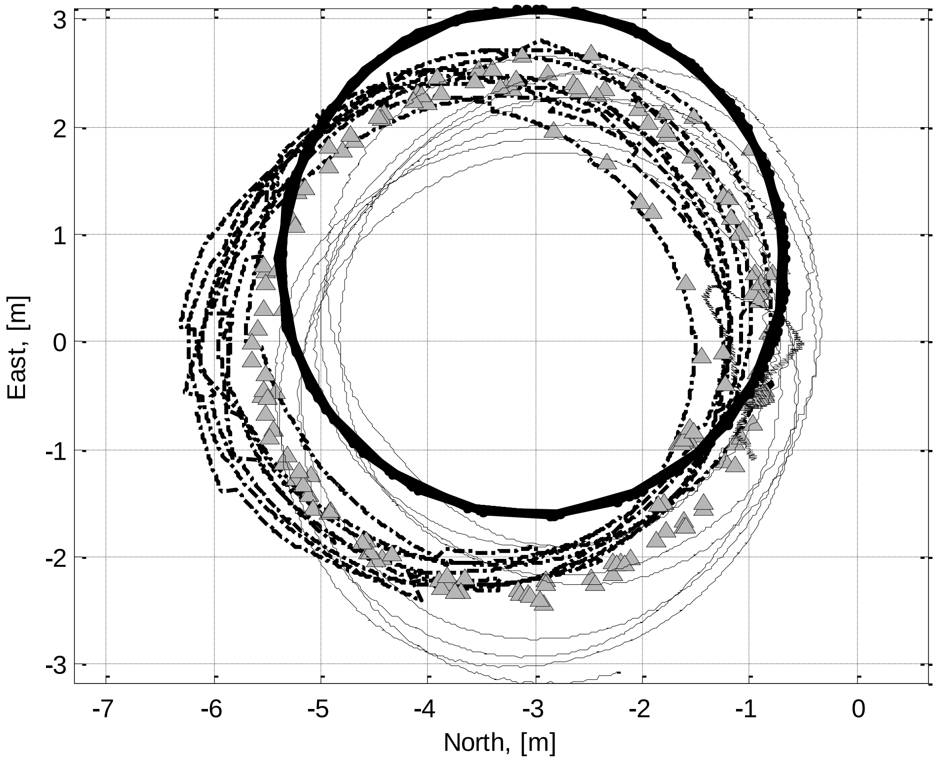


Fig. 6.11. Trajectories of the devices in horizontal plane

Figure 6.11 shows the trajectory in the North-East plane. All the results are presented in meters. The thick black solid line denotes reference trajectory calculated with the use of GrafNav/GrafNet software (Novatel.com, 2018 b), the triangles denote GPS measurements of the receiver, the dash-and-dot line represents the trajectory calculated by the developed IISNS breadboard model, and thin solid line represents the trajectory calculated by the French commercial system (Sbg-systems.com, 2018).

As is seen from Fig. 6.11, the developed integrated navigation system quite accurately traces signals of the GPS receiver. We can also see that GPS signals disappeared for several seconds during which small shift was observed in determining the coordinates in the integrated navigation system with respect to the reference trajectory. When GPS signals appeared, the navigation system corrected the coordinates according to current GPS measurements of the receiver, which testifies to the efficiency of the SINS correction algorithms.

Figure 6.12 shows the errors of coordinates of the integrated navigation system (grey) and separately GPS (black) with respect to the reference trajectory calculated by means Novatel GrafNav/GrafNet software according to the technique described above. As is seen from the figure, solution of the integrated navigation system is more smoothed, without height spikes.

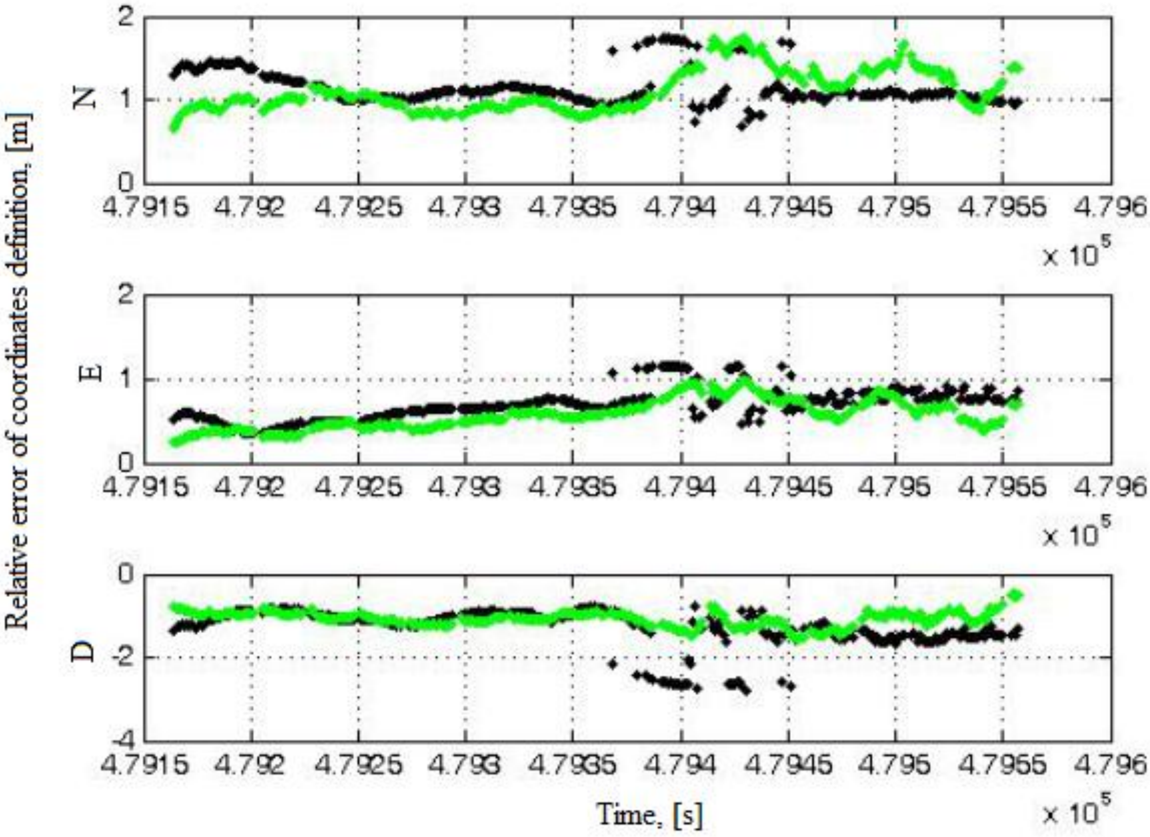


Fig. 6.12. Errors of coordinates of the integrated navigation system and separately GPS with respect to the reference trajectory

Average values of $\bar{x}$, root-mean-square deviations of s, and maximum values $\max$ of the relative errors of coordinates of the integrated navigation system ($\bar{x}_{err\_INS+GPS}$, $s_{err\_INS+GPS}$, $m_{err\_INS+GPS}$) and separately GPS ($\bar{x}_{err\_GPS}$, s_{err_GPS}, max_{err_GPS}) were also calculated by means of the corresponding MATLAB commands (mean, std, and max, respectively):

$$\bar{x}_{err\_GPS} = [1.1481 \quad 0.6909 \quad -1.2968]^T \quad s_{err\_GPS} = [\,0.2084 \quad 0.1849 \quad 0.4627]^T$$

$$\max_{err\_GPS} = \max(|x_{err\_GPS}|) = [\,1.7463 \quad 1.1432 \quad 2.7938]^T$$

$$\bar{x}_{err\_INS+GPS} = [\,1.1038 \quad 0.5694 \quad -1.0793]^T \quad s_{err\_INS+GPS} = [\,0.2446 \quad 0.1736 \quad 0.1778]^T$$

$$m_{err\_INS+GPS} = \max(|x_{err\_INS+GPS}|) = [\,1.7344 \quad 0.9804 \quad 1.5620]^T$$

Both Fig. 6.12 and the calculations above used the array of rarefied data from the integrated navigation system since the information output frequency of the integrated system was 50 Hz, and that of the GPS receiver was only 1 Hz. The time interval in Fig. 6.12 is from 479 158 to 479 557 sec of GPS week. Note also that for comparison we took the maximum deviations in order to derive guaranteed estimates and minimize the risk of UAV collision with the surface or another flying object.

As is seen from the results in Figs. 6.8-6.12 and calculations, the proposed methods of complex data processing in the integrated navigation system are generally quite efficient and successful even in case of significant noise and some regular shifts of inertial sensors, but under permanent presence of GPS signal.

6.3. Ground Studies of the Integrated Navigation System

The facility connection scheme is similar to that described for the experiment from the previous section 6.2. The only difference is that instead of two laptops, one laptop was used to record all the specified data. The equipment is shown in Fig. 6.13.

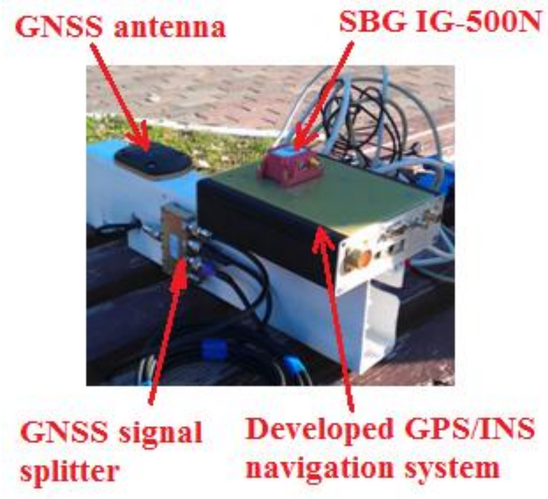


Fig. 6.13. Equipment used in the experimental study

In the experiment, at first, static measurements of navigation parameters were performed during 2 min, then the device was carried in hands with average velocity of 2-3 km/hour. Figure 6.14 shows the reference trajectory of the device, calculated by means of Novatel GrafNav/GrafNet software (Novatel.com, 2018 b).

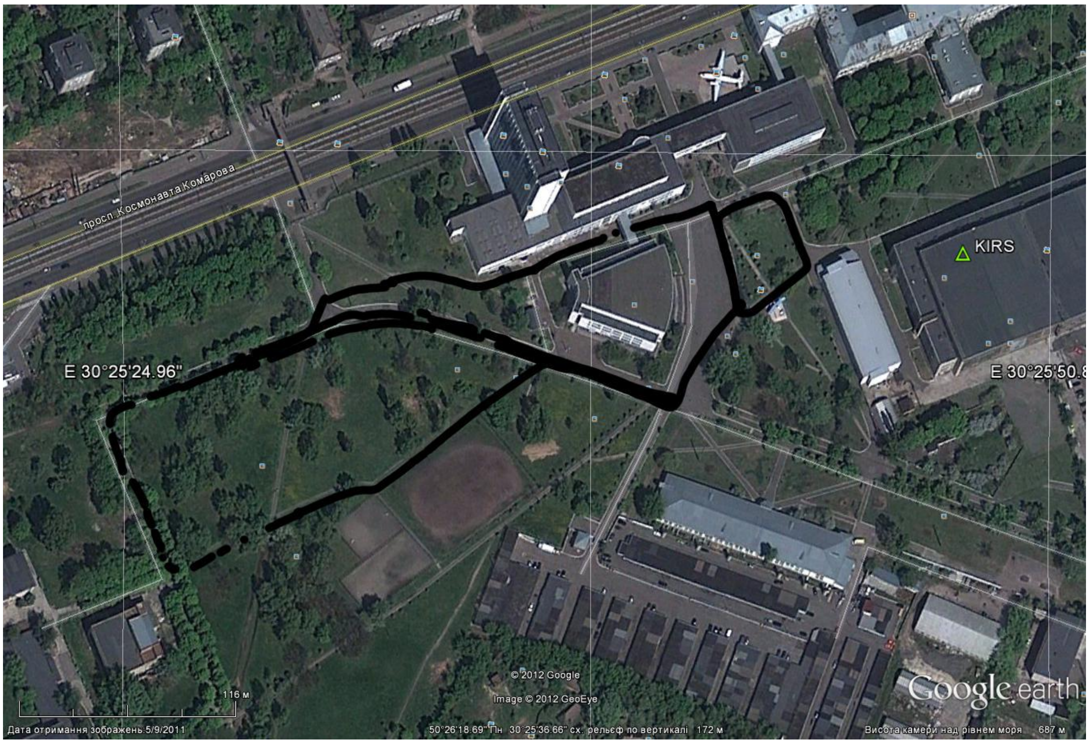

просп. Космонавта Комарова
KIRS
E 30°25'24.96"
E 30°25'50.8
© 2012 Google
Image © 2012 GeoEye
116 м
Google earth
Дата отримання зображень 5/9/2011
50°26'18.69" Пн 30°25'36.66" сх. рельєф по вертикалі 172 м
Висота камери над рівнем моря 687 м

Fig. 6.14. Reference trajectory in Google Earth

During the experiment, nominal mode of the integrated navigation system was analyzed, i.e., a mode where INS was permanently corrected by GPS, and there were no long failures of satellite navigation signals.

The results of data processing in NED frame are presented in Fig. 6.15-6.18. In Fig. 6.15, light grey color denotes the run time dependence of roll, pitch, and heading angles, calculated by the designed breadboard model of IISNS, and black color denotes those for the French IISNS ((Sbg-systems.com, 2018). Figure 6.16 shows the run time dependence of linear velocities with respect to three axes in NED frame. Black dots connected by solid line denote linear velocity obtained from the GPS receiver, and grey thick line denotes linear velocities calculated by the integrated navigation system. As is seen from the figure, these rated velocities almost coincide in steady state mode. So-called data spikes (anomalous measurements) of the GPS receiver took place, which aggravated the accuracy of speed and hence of coordinates (Fig. 6.17).

Figure 6.18 shows the trajectory in the North-East plane. All the results are presented in meters. The thick light-grey line denotes calculation of the trajectory by the developed IISNS breadboard model, the thick black line shows measurements of the GPS receiver.

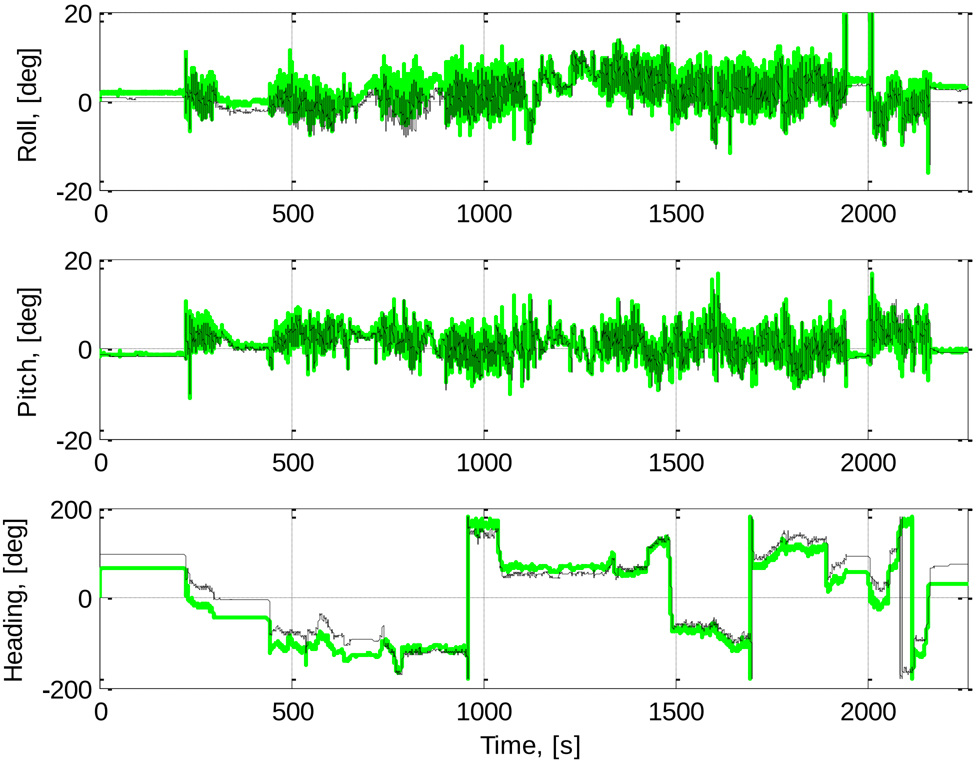


Fig. 6.15. Roll, pitch, and heading angles of the integrated navigation system

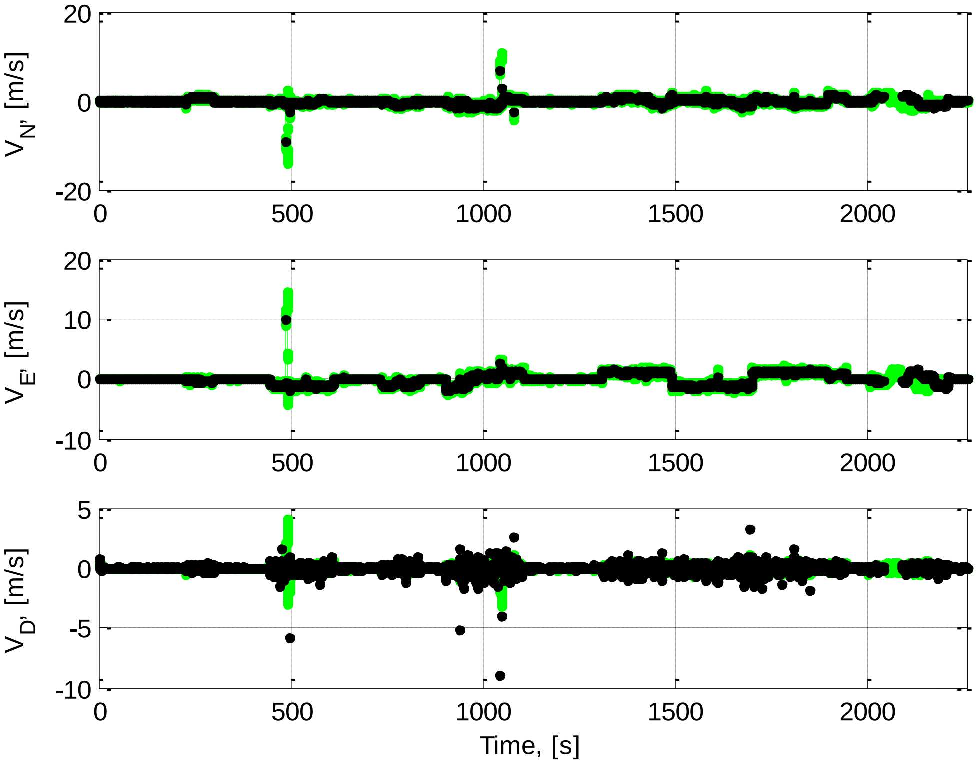


Fig. 6.16. Linear velocity of the integrated navigation system in NED frame

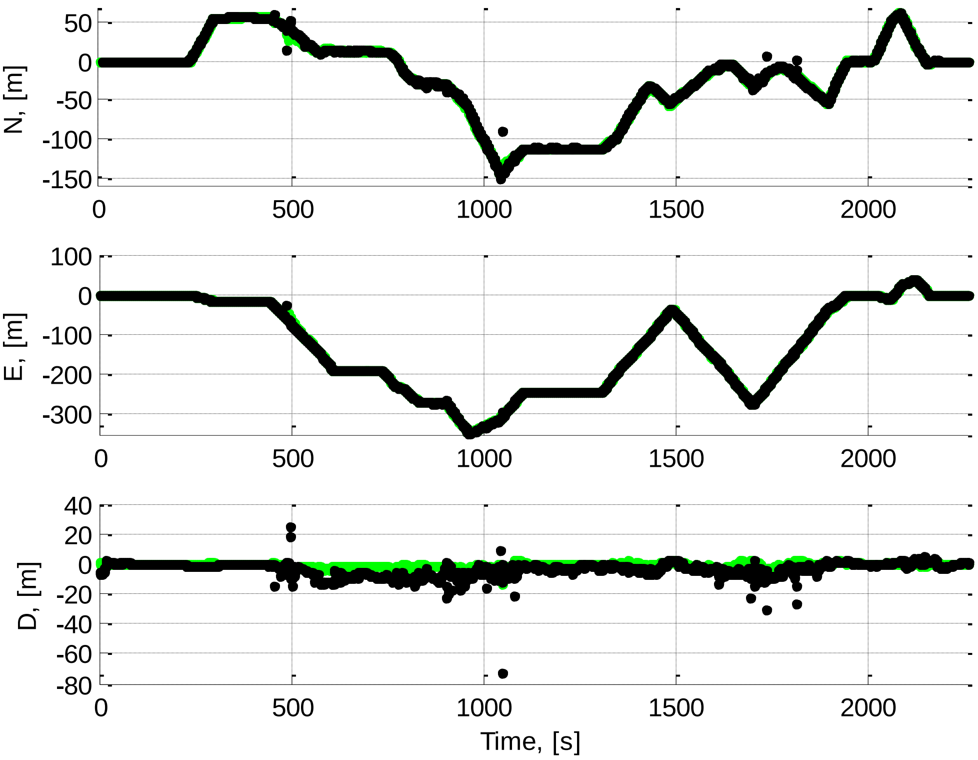


Fig. 6.17. Coordinates of the integrated navigation system in NED frame

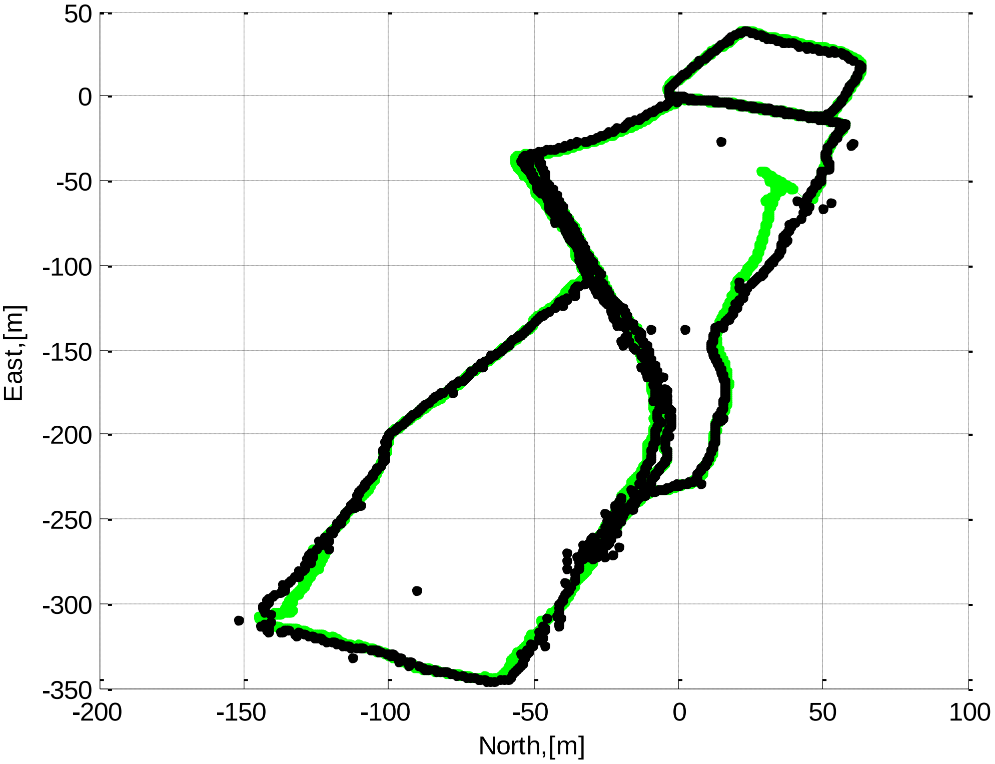


Fig. 6.18. Trajectory of the device in horizontal plane

6.4. Ground Studies of the Integrated Navigation System with the Use of a Car.

This section presents the results of ground tests, with the use of a car, of the following version of the IISNS breadboard model developed at the Scientific and Education Center "Aerospace Center" of the National Aviation University (Fig. 6.19). The component sensors and systems are as follows: single-frequency GNSS receiver and inertial measurement unit ADIS16488 (Analog.com, 2018 b) containing three-axial accelerometer, angular velocity sensor, magnetometer, internal temperature meter, and barometric altimeter. Software for calculation of the navigation solution in the integrated navigation system was implemented in a microcontroller Freescale Kinetis K-60 family (Nxp.com, 2013).

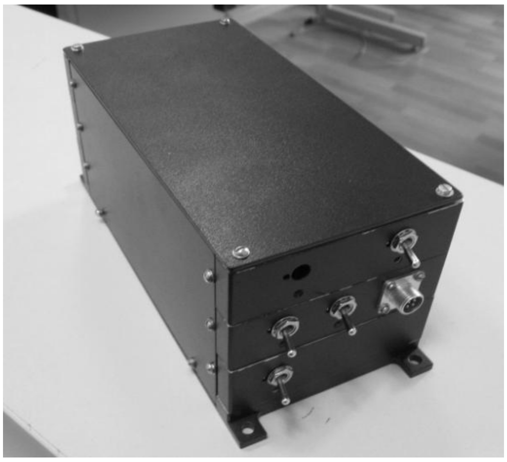


Fig. 6.19 Breadboard model of the integrated inertial-satellite navigation system

In the experiment, the GNSS antenna was placed on a car roof (Fig. 6.20), the device was mounted inside, and the *oh* axis coincided with the longitudinal axis of the car. To make it possible to calculate reference trajectory, an additional GNSS receiver was placed in the car and provided raw measurements. In Fig. 6.20, 1 is GNSS antenna, 2 is splitter (GNSS antenna signal distributor), 3 is IISNS breadboard model, and 4 is GNSS receiver. The equipment connection scheme in this experiment is similar to that presented in Fig. 6.7. The only difference is that internal flash memory cards were also used along with the laptop to save data from the devices.

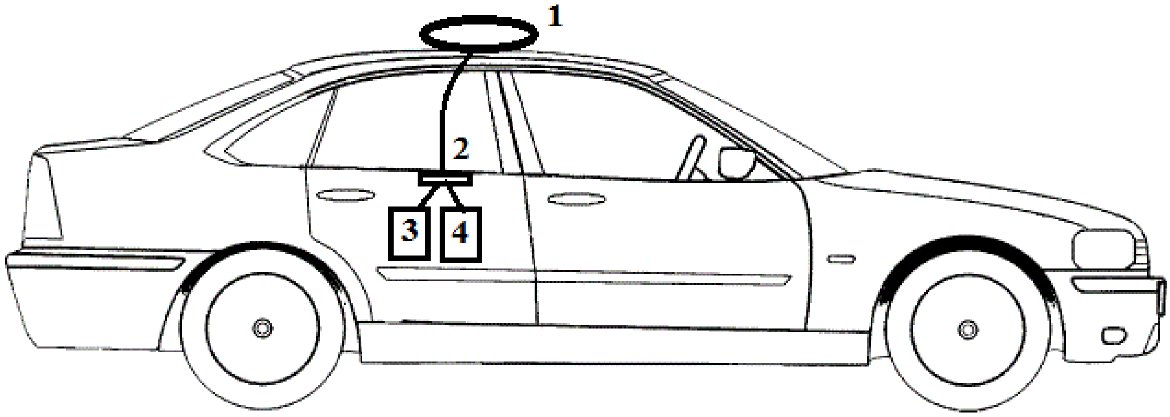


Fig. 6.20. Schematic arrangement of the equipment during the experiment

Noteworthy is that all the navigation calculations were carried out in real time in the device. In the experiment, at first, static measurements of navigation parameters were carried out during several minutes, then the device was moved in a car with average velocity of 6-15 km/hour. Figure 6.21 presents the following: (a) reference trajectory of the car in Google Earth (in latitude of longitude parameters), obtained from raw GNSS measurements by means of GrafNav/GrafNet (Novatel.com, 2018 b) together with the photo of the probationary ISNS sample, (b) calculated trajectory in the North-East plane, obtained from the device (presented in the NED frame). The experiment run time was about 16 minutes, the length of the trajectory was about 4 km.

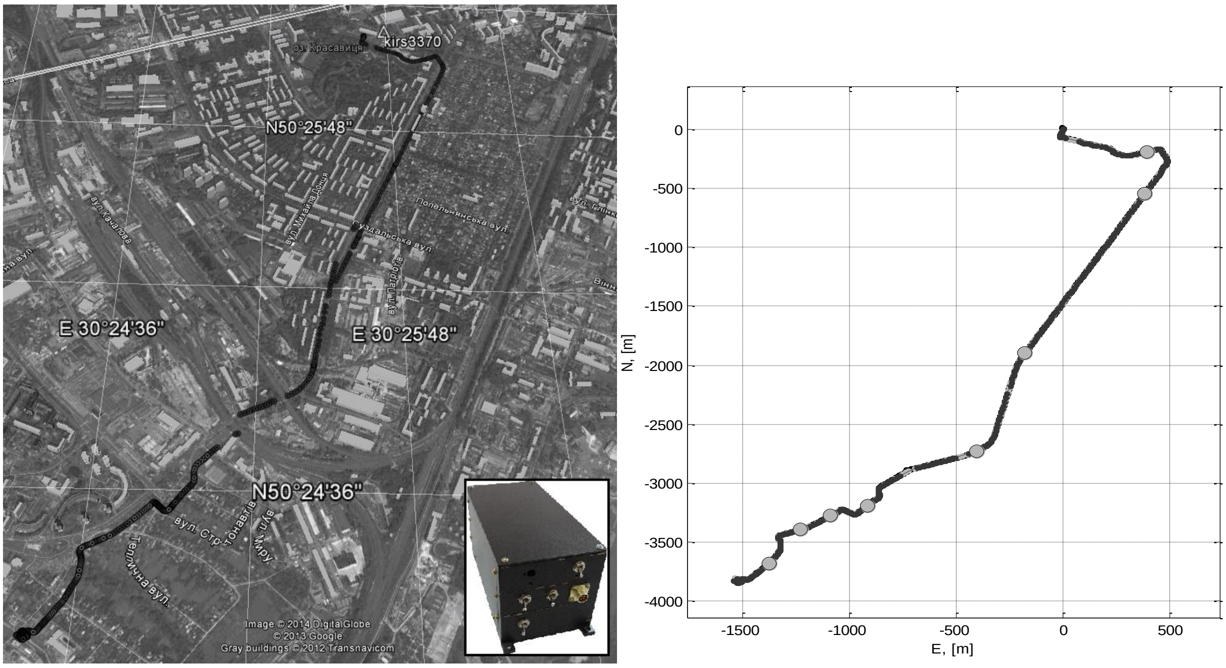


Fig. 6.21. Reference trajectory in Google Earth and photo of the probationary ISNS

The dots in Fig. 6.21 denote the beginning and the end of four sections of the trajectory, which are of interest since they contain turns and respectively contain variations in the velocity and attitude parameters.

The results of the experiment are presented in Figs. 6.22-6.24, where dots mark the same sections of the trajectory as in Fig. 6.21. Figure 6.22 shows the run time dependence of roll, pitch, and heading angles, calculated by the developed IISNS breadboard modelWe can see from the figure how heading angle varied during turns. Figure 6.23 shows the run time dependence of linear velocities with respect to three axes in NED frame. The thick grey line denotes the linear velocity obtained from the GPS receiver, and thin black line denotes linear velocities calculated by the integrated navigation system. We can see from the figure that the velocities almost coincide in the nominal mode, only the vertical component of the velocity in the IISNS was smoother. Figure 6.24 shows the run time dependence of the coordinates of the object in the NED frame. The notation is the same as in Fig. 6.23. In the nominal mode, the coordinates almost coincide here as well, with an insignificant difference in the vertical component.

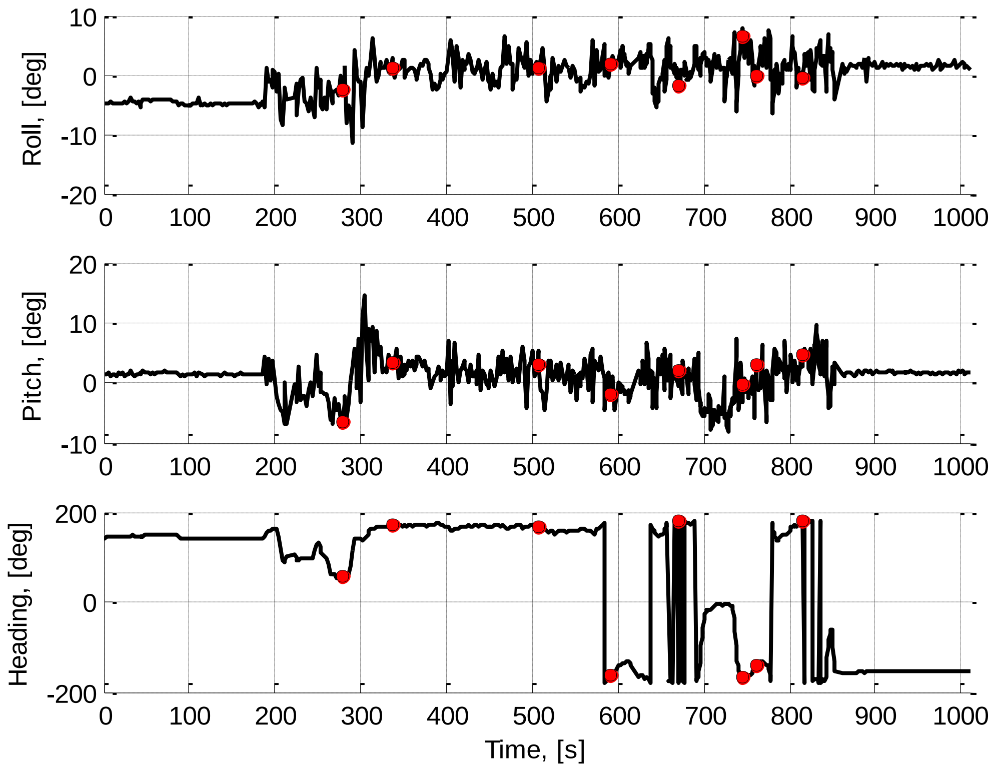


Fig. 6.22. Roll, pitch, and heading angles of the integrated navigation system

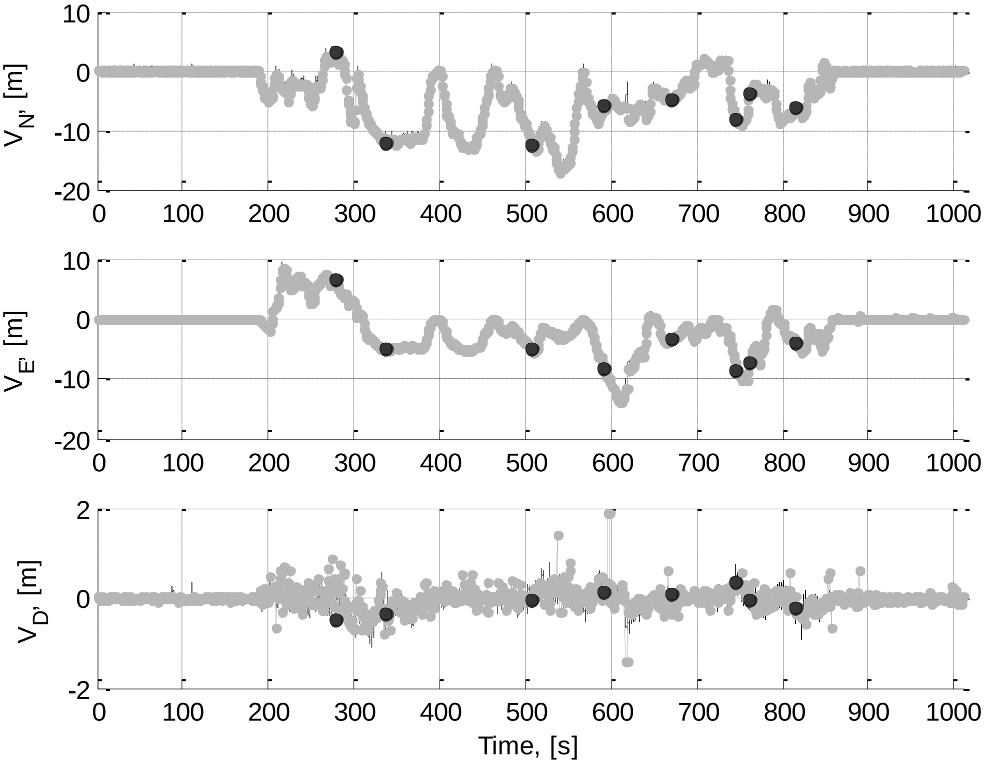


Fig. 6.23. Linear velocity of the integrated navigation system in NED frame

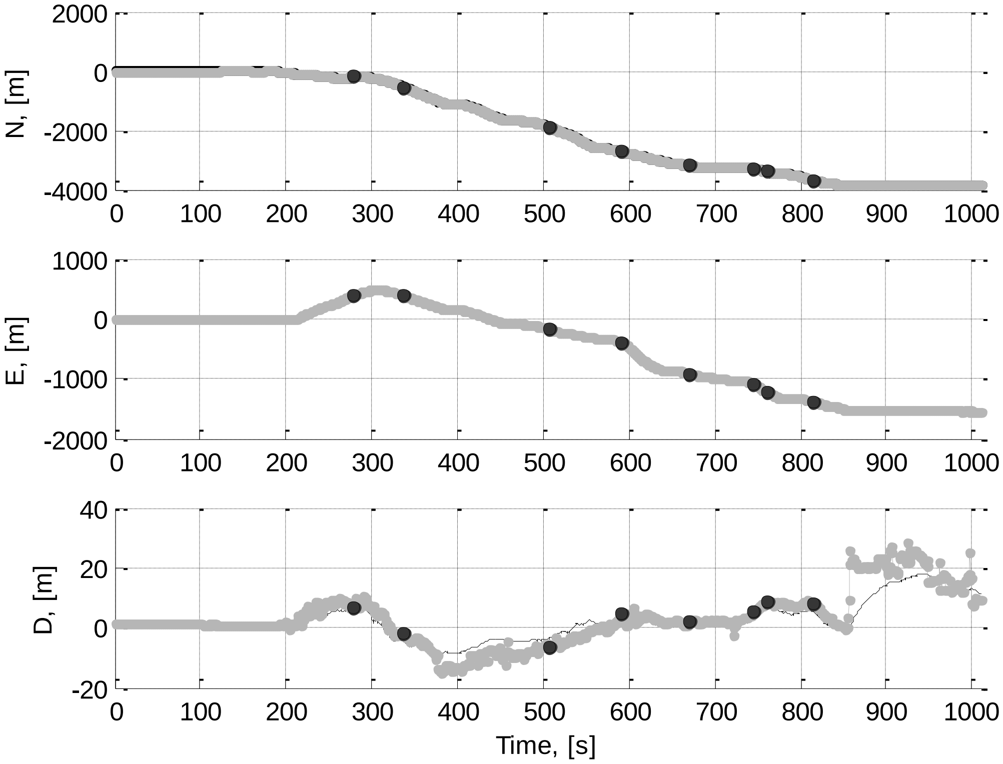


Fig. 6.24. Coordinates of the integrated navigation system in NED frame

6.5. Flight Test of the Integrated Navigation System

In this section, we will discuss two flight experiments, both carried out at the Borodyanka airdrome, which is a class "D" airdrome and is certificated according to the Air Code of Ukraine. The geographical and administrative data of the airdrome are presented in Table 6.1.

Table 6.1 – Borodyanka airdrome

Reference point and coordinates at the AD	503957.19N 295600.83E
Direction and distance from populated areas	30 km North West from Kyiv, 2 km North from Borodyanka
Airdrome altitude/ designed temperature	149.2 m/490 ft/ 23.6 °C
Magnetic declination	4° E
Threshold altitude	102° – 149.2 m/490 ft; 282° – 145.6 m/478 ft
Type of authorized flights	VFR

Experiment 1

The results of this experiment are also published in (Kharchenko and Ilnytska, 2013) and partially in (Kharchenko et.al, 2013). A breadboard model of the integrated navigation system and GPS antenna (Fig. 6.26) were placed on a small-size UAV (Fig. 6.25). During the experiment, at first, static measurements of navigation parameters were carried out during 1-2 min to perform initial setting, then the UAV flied under manual control. Futaba panel was used for UAV control.

Data from sensors of the breadboard model of the integrated navigation system were saved to flash memory card of the device. After the end of the experiment, data post-processing according to the algorithms presented in Chapters 2 and 3 of the present monograph was carried out. The flight trajectory is shown in Fig. 6.27.

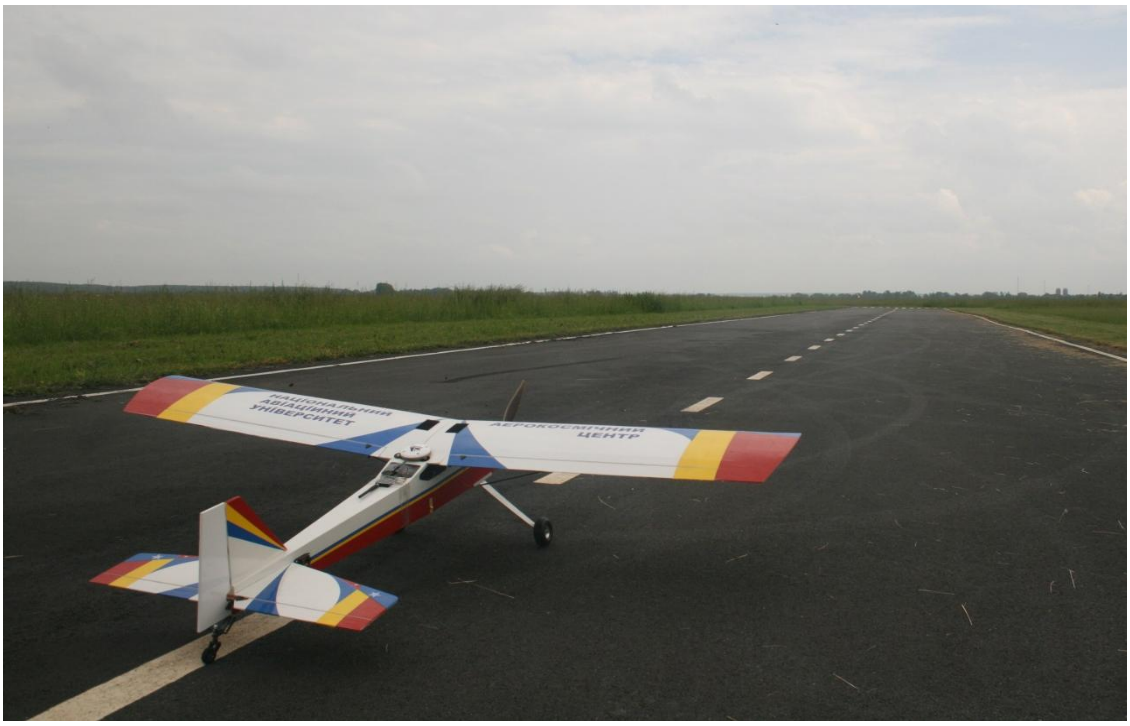


Fig. 6.25. The UAV used for flight tests

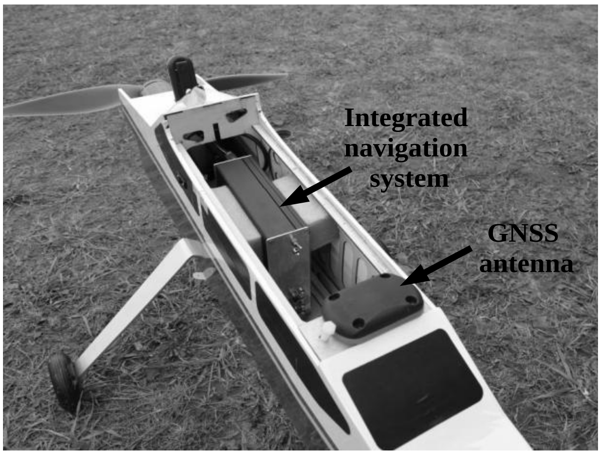


Fig. 6.26. Equipment located at the UAV

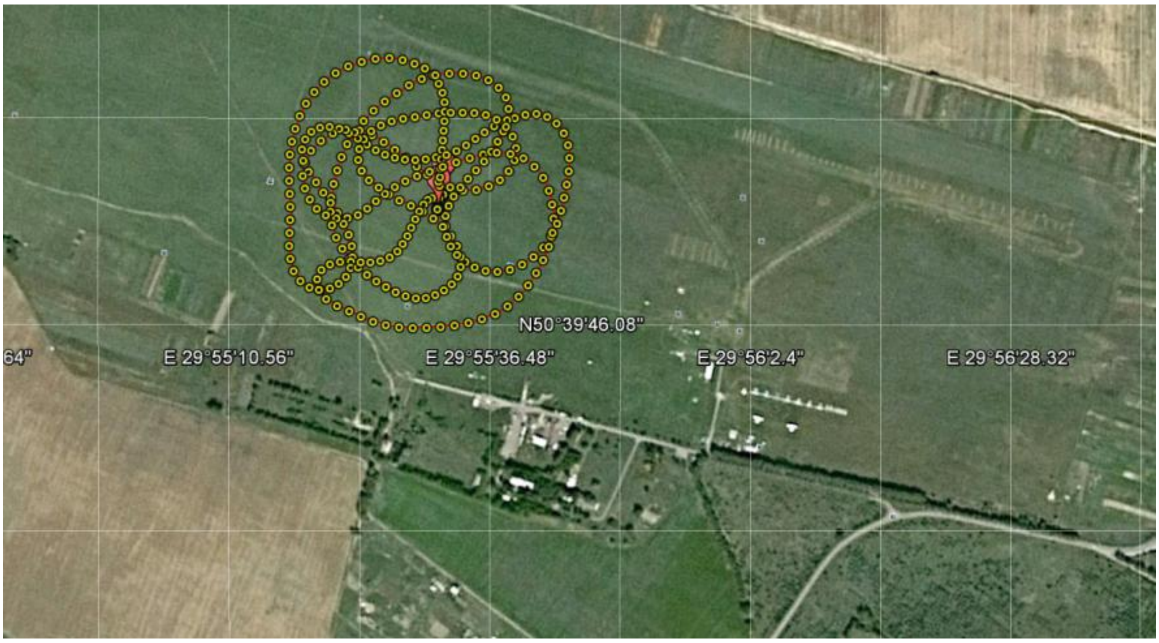


Fig. 6.27. The flight trajectory in Google Earth

The results of data processing in the NED frame are presented in Figs. 6.28 - 6.32. Figure 6.28 shows the run time dependence of roll, pitch, and heading angles, calculated by the breadboard model of the integrated navigation system.

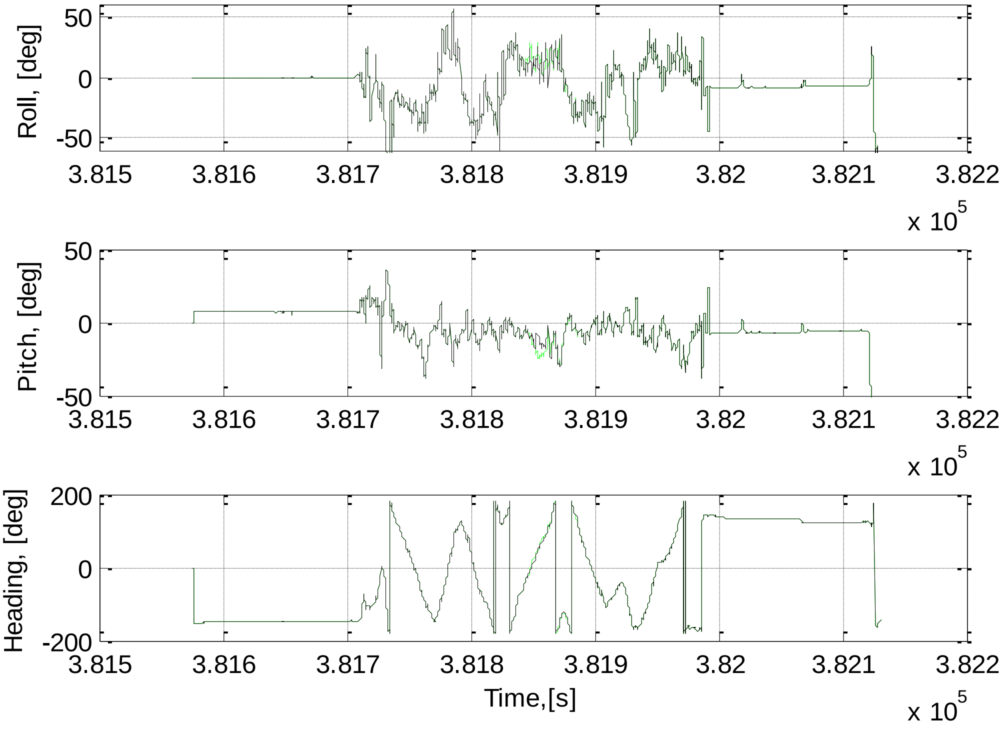


Fig. 6.28. Roll, pitch, and heading angles of the integrated navigation system

Figure 6.29 shows the run time dependence of linear velocities with respect to three axes in the NED frame. The black points connected by solid line designate the linear velocity obtained from the GPS receiver, and thick grey line denotes the linear velocities calculated by the developed integrated navigation system. As is seen from the figure, the velocities almost coincide in the nominal mode. The thick red line in Fig. 6.29 marks the section where data from the GPS receiver were not taken into account in the complex data processing of the integrated navigation system in order to simulate GPS signal loss.

It can be seen that during autonomous SINS operation, velocity directions remained the same; however, insignificant amplitude deviations were observed, which can be explained by the specifications of MEMS-type inertial sensors. Note also that when signal GPS appeared again, the integrated navigation system immediately corrected the solution according to it and then continued operating in nominal mode.

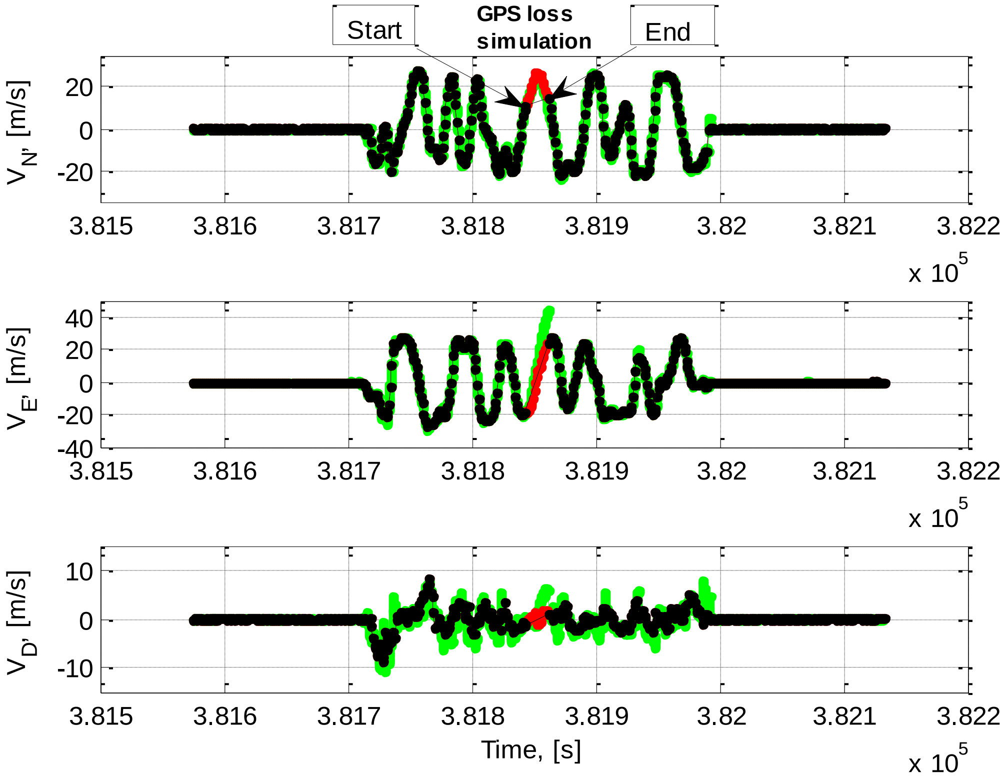


Fig. 6.29. Linear velocity of the integrated navigation system in NED frame

Figure 6.30 shows the run time dependence of coordinates of the object with respect to three axes in the NED frame. The notation is the same as in the previous figure. We can see that some time after GPS signal loss the integrated navigation system continued comprehensible tracing of the coordinates, and then, after signal from GPS receiver appeared, it continued operating in nominal mode.

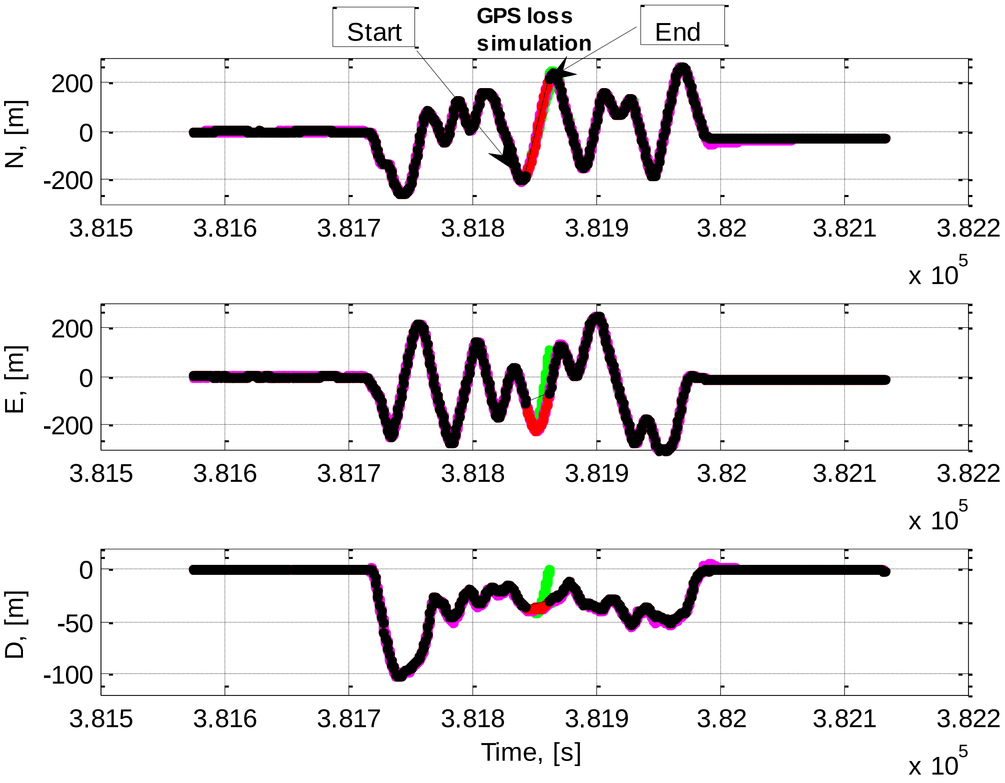


Fig. 6.30. Coordinates of the integrated navigation system in NED frame

Figures 6.31 and 6.32 show the UAV trajectory (in meters) in the North-East plane and in space, respectively. The green asterisks denote measurements by GPS receiver that were taken into account in the complex data processing; the solid thick dark blue line denotes the trajectory calculated by small-size integrated navigation system; red asterisks mark the section where data from the GPS receiver were disregarded in the complex integrated navigation system data processing in order to simulate GPS signal loss. As is seen from Figs. 6.31-6.32, in the nominal mode the small-size integrated navigation system rather accurately traces signals from the GPS receiver, and in abnormal mode it keeps turn direction but forecasted (expected) aggravation of coordinate accuracy is observed. We can also see that after GPS signals appeared, the integrated navigation system corrected its coordinates according to current measurements of the GPS receiver, which testifies the efficiency of the SINS correction algorithms.

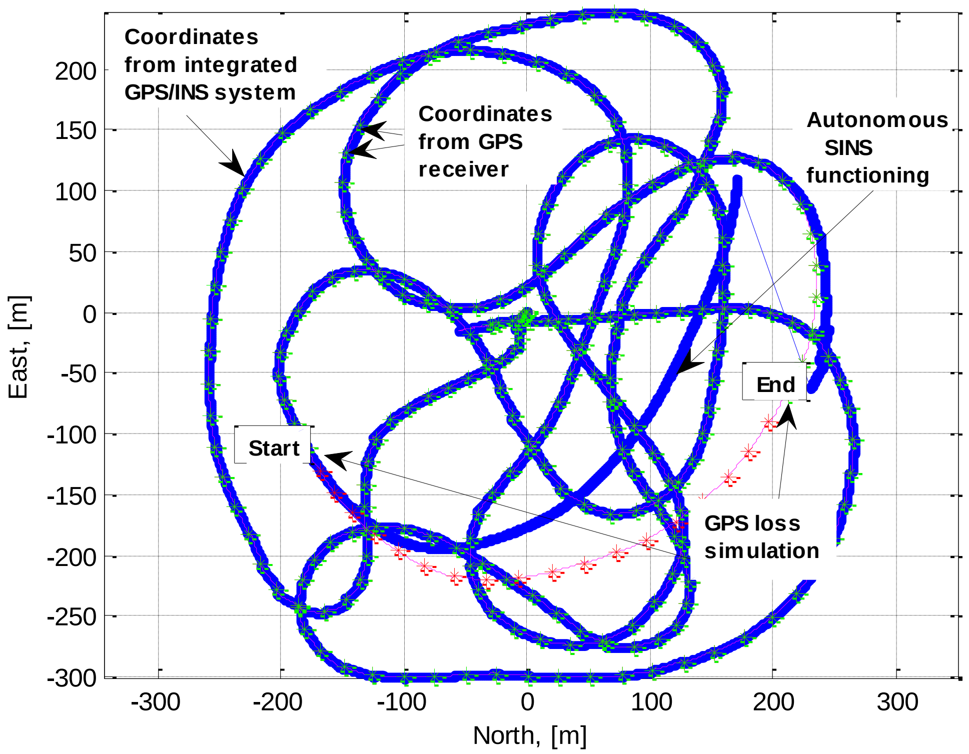


Fig. 6.31. UAV trajectory in horizontal plane

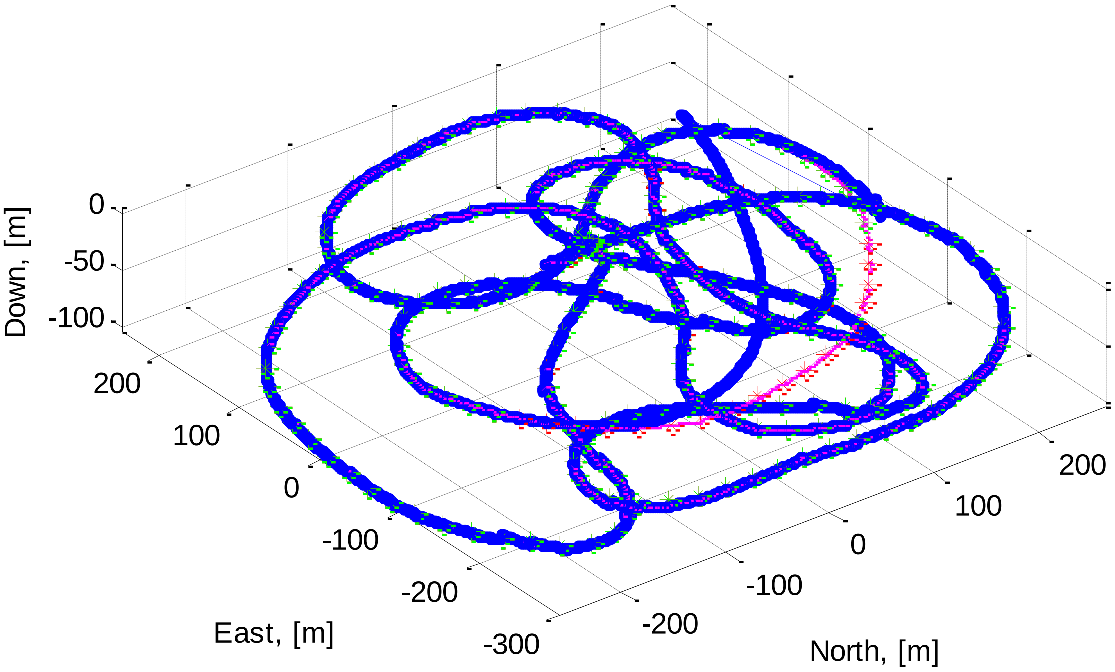


Fig. 6.32. UAV spatial trajectory

During 20 sec of simulation of GPS receiver signal loss, the largest deviations from the stationary mode flight trajectory had the following values:

$$\max\left(\left|err\_Posit\right|\right)=\left[61.3043 \quad 131.8389 \quad 9.8646\right]m,$$

$$\max(|err\_Velocity|) = [10.3377 \quad 16.2613 \quad 3.7761]\, m/s,$$

which agree well with the modeled values presented in Sec. 3. Here $|\cdot|$ denotes the absolute value of a vector.

Errors of the components of IISNS velocity and coordinates with respect to GNSS for nominal mode are presented in Figs. 6.33-6.34. As a result of data processing in static mode, the following estimates of RMS errors of the velocity and coordinates were obtained:

$$\sigma_{errVeloc\_static} = [\,\mathrm{V_N} \;\; \mathrm{V_E} \;\; \mathrm{V_D}\,] = [\,0.0074 \quad 0.0064 \quad 0.0432]\, m/s,$$

$$\sigma_{err\mathrm{Posit}\_static} = [\,\mathrm{N} \;\; \mathrm{E} \;\; \mathrm{D}] = [\,0.1511 \quad 0.1187 \quad 0.2530]\, m.$$

The same estimates for flight conditions had the following values:

$$\sigma_{errVeloc\_dynamic} = [\,0.3938 \quad 0.3949 \quad 1.1242]\, m/s,$$

$$\sigma_{err\mathrm{Posit}\_dynamic} = [\,0.5466 \quad 0.4877 \quad 0.8075]\, m,$$

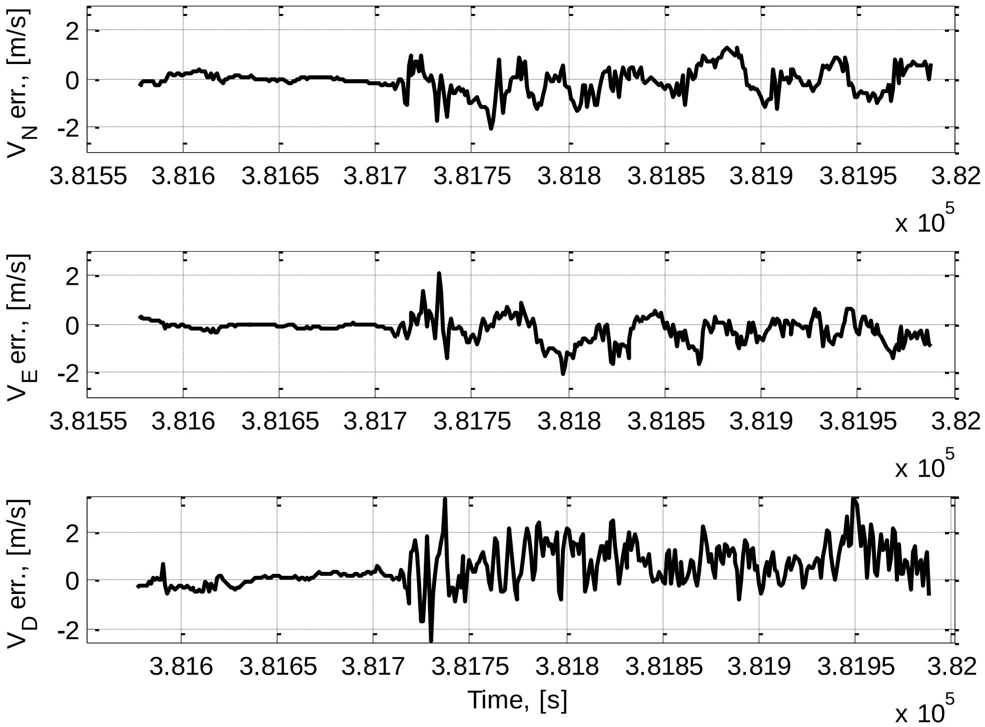


Fig. 6.33. Relative errors of IISNS velocity in NED frame during nominal mode

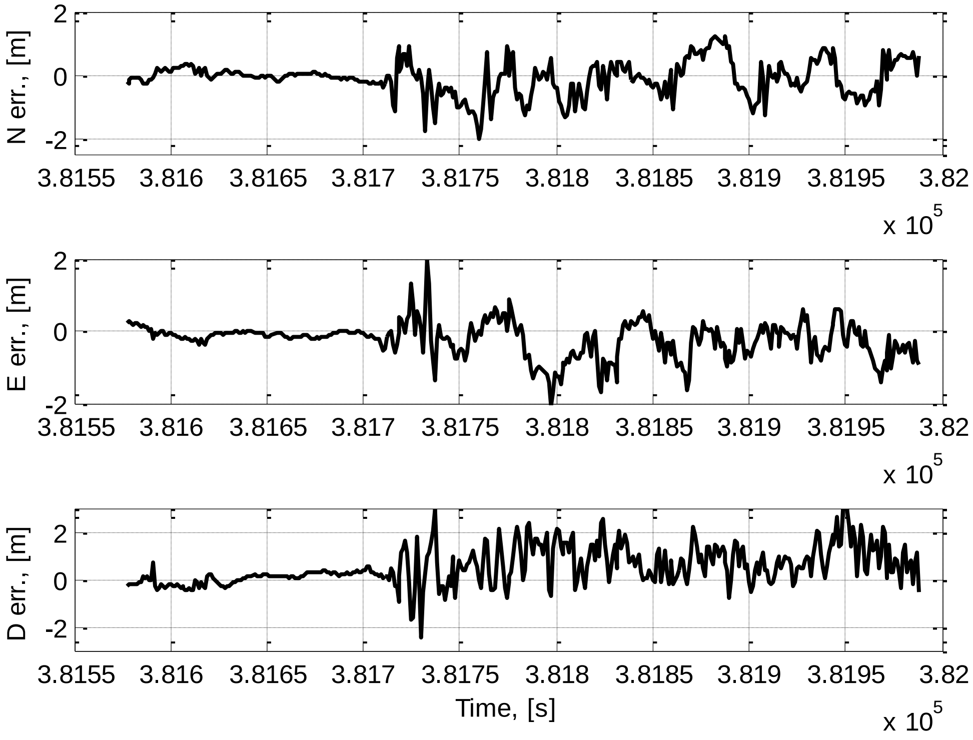


Fig. 6.34. Relative errors of IISNS coordinates in NED frame during nominal mode

As is seen from the results presented in Figs. 6.28-6.34, the proposed methods of complex data processing of the integrated navigation system are generally rather efficient and successful even in case of significant noise and some systematic shifts of inertial sensors; however, this is only ensured under permanent presence of GPS signal. Noteworthy is also that it is necessary to pay special attention to vibration insulation of the navigation equipment used on the UAV to enhance the computing accuracy.

Experiment 2

In this experiment, the developed IISNS breadboard model (Fig. 6.35) was tested together with the French commercial system (Sbg-systems.com, 2018) by means of the same small-size UAV as in the previous experiment (Fig. 6.25).

In the experiment, at first, static measurements of navigation parameters were carried out during 3 min to perform initial setting, then the UAV flied under manual control by means of Futaba panel. In this experiment, nominal mode of the IISNS was investigated.

All the calculations of navigation data were performed in real time immediately in the designed IISNS breadboard model, which is detailed in Sec. 6.4.

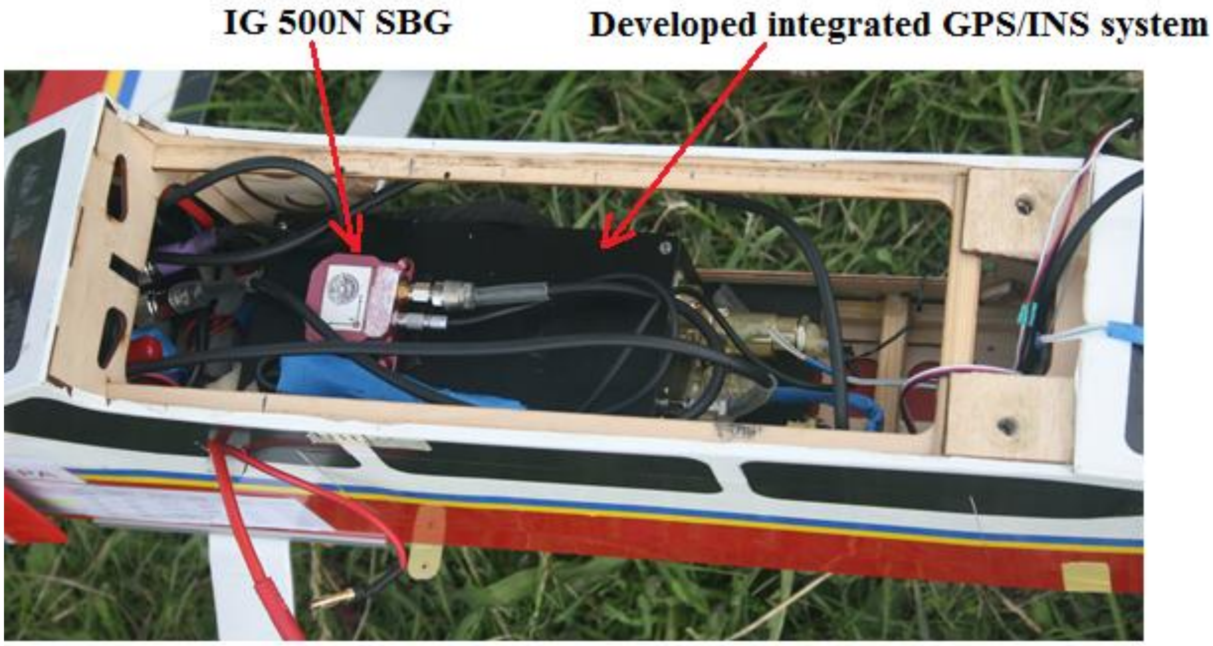


Fig. 6.35. The probationary IISNS breadboard model together with the commercial French sample

The results of the experiment are presented in Figs. 6.36-6.38. Figure 6.36 shows the run time dependence of roll, pitch, and heading angles, calculated by the breadboard model of the integrated navigation system (black solid line) and the commercial French sample (Sbg-systems.com, 2018) (dark blue solid line).

Figure 6.37 shows the run time dependence of coordinates of the plant with respect to three axes in the ENU frame. The black solid line denotes the coordinates obtained by the designed IISNS breadboard model and green (grey) thick line denotes the coordinates calculated by the commercial French sample. As is seen from the figure, coordinates obtained by both samples almost coincide.

Figure 6.38 shows the UAV trajectory in the East-North plane (in meters). The green (grey) asterisks denote measurements of the GPS receiver, solid thick black line is the trajectory calculated by the developed IISNS breadboard model. We omit here the trajectory obtained by the commercial French sample since (as can be seen from Fig. 6.37) it almost repeats the trajectory of the developed IISNS breadboard model.

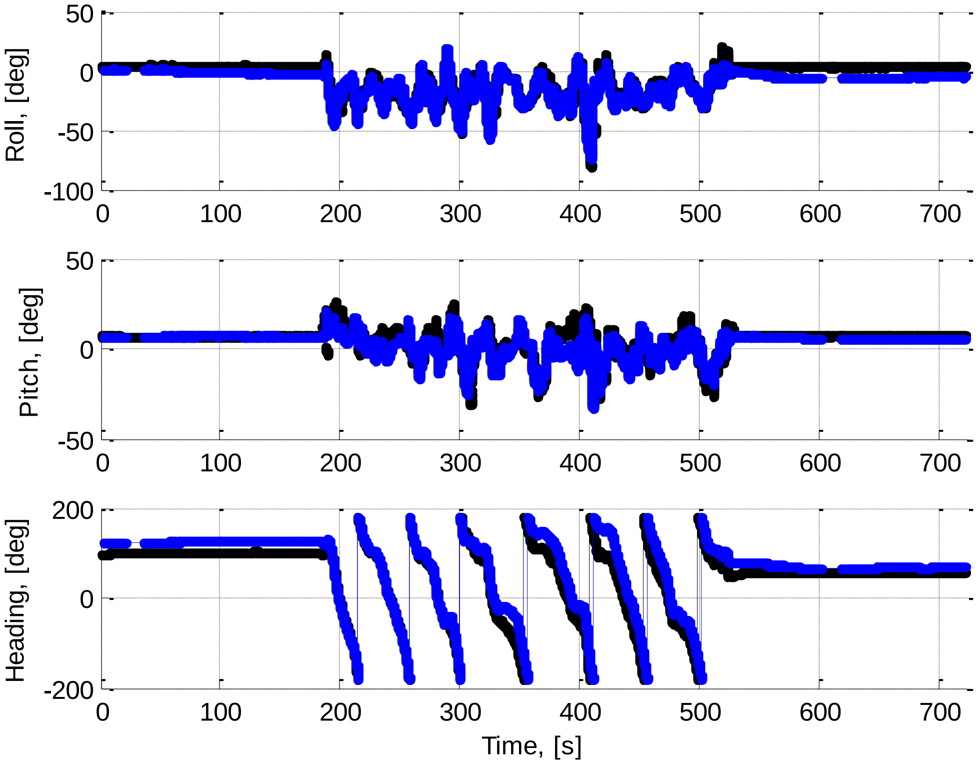


Fig. 6.36. Roll, pitch, and heading angles of the integrated navigation system

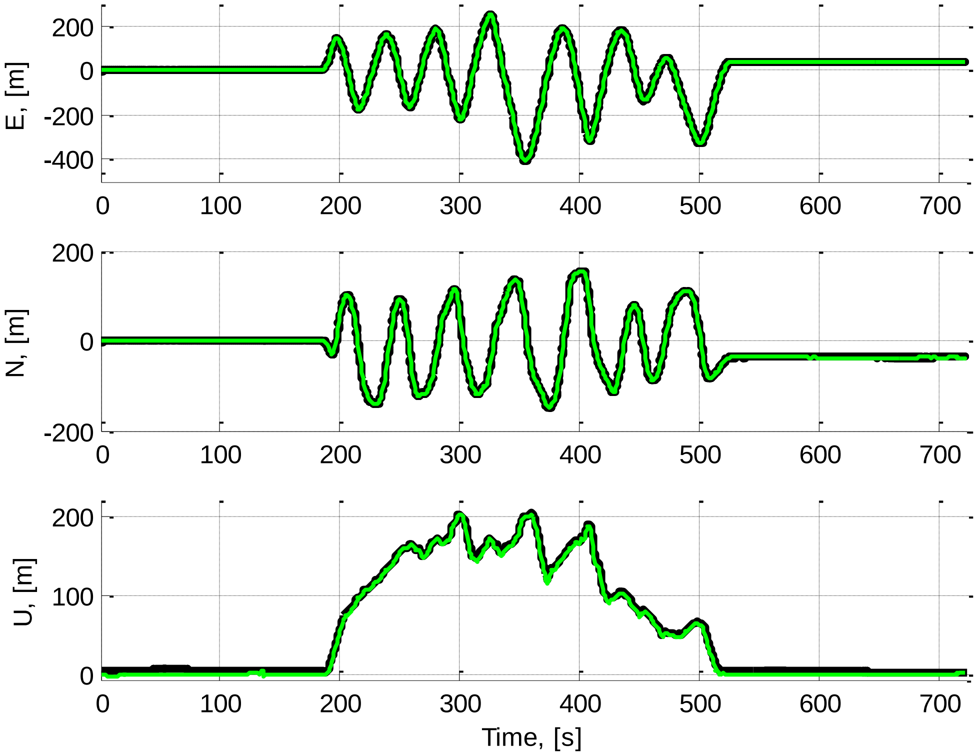


Fig. 6.37. Coordinates of the integrated navigation system in ENU frame

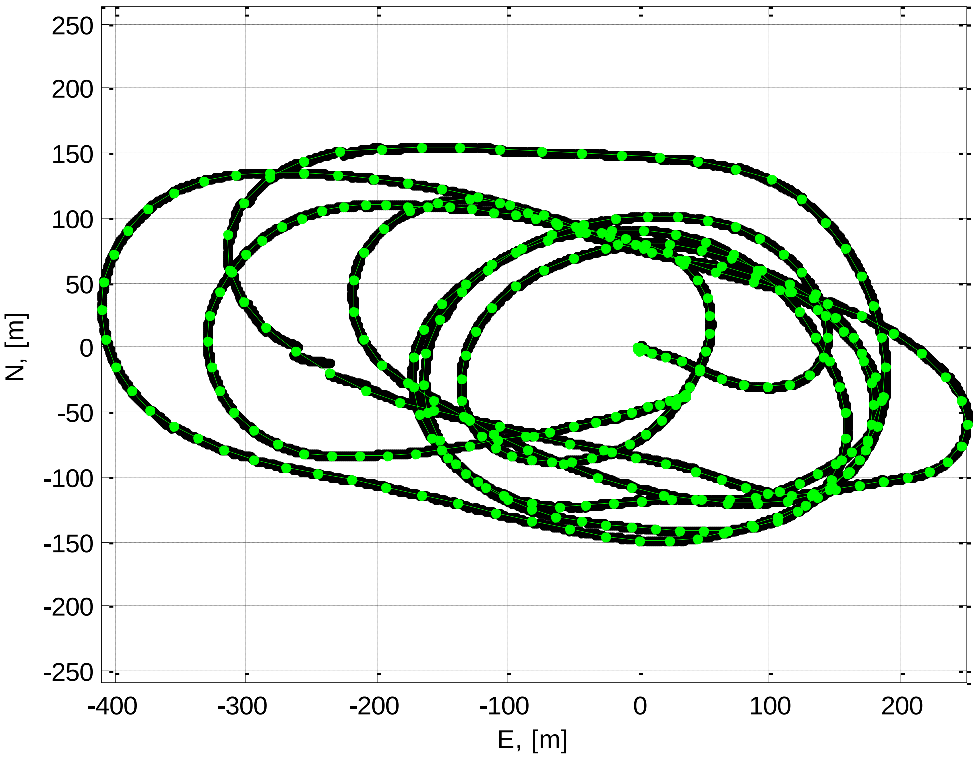


Fig. 6.38. UAV trajectory in horizontal plane

As we can see from the above results, in the nominal mode, which was tested in the experiment, the developed IISNS breadboard model operated well, no worse than the French commercial system.

Similar practical results of integrated navigation system flight validations together with some recommendations have been published in (Kharchenko et.al, 2014).

6.6. Summary for Chapter 6

6.6.1. We have presented the results of laboratory, ground, and flight tests of the developed IISNS.

6.6.2. The laboratory tests were carried out at an elementary rotating motion table with the use of GrafNav / GrafNet high-precision equipment manufactured by Waypoint (Novatel.com, 2018 b) and its software, which allows constructing a reference trajectory with respect to which system errors were calculated.

6.6.3. Ground tests were carried out under slow motion of the IISNS in GrafNav / GrafNet operating zone, which allowed calculating accuracy characteristics of the system by means of the reference trajectory.

6.6.4. Flight trials were performed by means of arranging the IISNS experimental model on a small UAV with postprocessing of data obtained in flights on a stationary PC.

6.6.5. The results of these trials allow us to make a conclusion that accuracy characteristics of the proposed IISNS meet the requirements for navigation equipment of small-size UAV.

6.6.6. To assess the competitive capacity of the proposed IISNS in all above types of trials, its accuracy characteristics were compared with respective characteristics of the commercial (COTS) IISNS of the French system IG-500N. This comparison testifies that the proposed IISNS is at least no worse than available inertial-satellite navigation systems.

REFERENCES

Acutronic.com. (2018). 3-Axis Rate and Position Tables, 3-Axis Motion Simulators - ACUTRONIC / CH. [online] Available at: http://www.acutronic.com/ch/products/3-axis-motion-simulators.html [Accessed 9 Jul. 2018].

Actidyn.com. (2018). Motion Simulation and Control - Actidyn. [online] Available at: http://www.actidyn.com/products/motion-simulation-and-control/ [Accessed 9 Jul. 2018].

Ahn, I.K., Ryu, H., Larin, V.B. and Tunik, A.A. (2003) "Integrated Navigation, Guidance and Control Systems for Small Unmanned Aerial Vehicles", The world Congress "Aviation in the XXI-st Century". Ukraine, Kyiv, 2003, pp. 14 - 16.

Allan, D.W. (1966) "Statistics of atomic frequency standards", Proc. IEEE, vol. 54, no. 2, pp. 221–230, Feb. 1966.

Allan, D.W. and Barnes, J.A. (1981) "A modified "Allan Variance" with increased oscillator characterization ability", in Proc. 35th Annu. Freq. Control Symp., May 1981, pp. 470–475.

Aliev, F.A. and Larin, V.B. (1998) *Optimization of linear control systems: Analytical methods and computational algorithms*, Amsterdam: Gordon and Breach.

Aliev, F.A., Larin, V.B., Naumenko, K.I. and Suntsev, V.N. (1978) *Optimization of linear time invariant control systems*. [in Russian], Kyiv: Naukova dumka.

Analog.com. (2018 a). ADIS16362 Datasheet and Product Info | Analog Devices. [online] Available at: http://www.analog.com/en/mems-sensors/mems-inertial-measurement-units/adis16362/products/product.html [Accessed 9 Jul. 2018].

Analog.com. (2018 b). ADIS16488 Data Sheet Rev.G. Tactical Grade Ten Degrees of Freedom Inertial Sensor, Analog Devices. [online] Available at: http://www.analog.com/media/en/technical-documentation/data-sheets/ADIS16488.pdf [Accessed 9 Jul. 2018].

Andreev, V.D. (1967) *Theory of inertial navigation.Corrected systems.* [in Russian], M.: Nauka.

Artese, G. and Trecroci, G. (2008) "Calibration of a low cost MEMS INS sensor for an integrated navigation system". The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B5. Beijing. [online] Available at http://www.isprs.org/proceedings/XXXVII/congress/5_pdf/153.pdf [Accessed 09 Jul. 2018]

Austin, R. (2010) *Unmanned Aircraft Systems. UAVS Design, Development and Deployment*, John Wiley & Sons.

Avraamenko, L.G. and Larin, V.B. (1983) "On question about integration the kinematic equations in the parameters of Rodrigues-Gamilton", Sb.: Sistemy navigatsii i upravleniya, K.: Inst. Matematika AN USSR, - pp. 9 – 16. [in Russian]

Avraamenko, L.G., Larin, V.B. and Bordug, B.A. (1983) "About the algorithm of orientation determination of a rigid body", Sb.: Matematicheskaya fizika i nelineynaya mekhanika, V. 11(45), Kiev: Naukova dumka, pp. 12-13. [in Russian]

Bar-Itzhack I.B. (1982) "Minimal order time sharing filters for INS in-flight alignment", Journal Guidance, 5, N4, pp. 396 – 402.

Barbour, N., Hopkins, R., Kourepenis, A. and Ward, P., (2008) "Inertial MEMS System Applications". NATO RTO Lecture Series, RTOEN-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Prague, October 2008, pp.7 - 1, 7 - 14.

Barbour, N. (2011) "Inertial Navigation Sensors". NATO RTO Lecture Series, RTO-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Kiev, September 2011.- P. 2 - 1, 2 - 24.

Beard, R.W. and McLain, T.W. (2012) *Small unmanned aircraft: theory and practice,* Princeton University Press.

Bodanskiy, E.D. and Furman, V.D. (1970) "About errors of numerical integration of kinematic equations of Puasson" [in Russian]. Kosmicheskiye issledovaniya, **8**, № 6.– pp. 944 – 948.

Bierman, G.J. (1977) *Factorization Methods for Discrete Sequential Estimation*, Academic Press, New York, San Francisco, London.

Branets, V.N. and Shmyglevsky I.P. (1973) *Application of quaternions in the problems of the rigid body attitude determination*, (in Russian), Nauka, Moscow.

Bryson, Jr. A.E. and Ho-Yu-Chi (1969) *Applied optimal control. Optimization, estimation and control*, Braisdell Publising Company, Waltham, Massachusetts.

Bronkhorst, V. (1978) "Strapdown system algorithms," Ser. 95, AGARD Lecture, NATO. pp. 3 – 1, 3 – 22.

Bogdanov, M.B., Galkin, A.A., Prohortsov, A.V., Saveliev, V.V., Udakova, N.D. (2011) "Integrated inertial/satellite orientation and navigation system on accelerometer-based SINS", Proc. of 18th Saint-Petersburg Conf. on Integrated Navigation Systems, pp. 216 – 218.

Chelnokov, Yu.N. (1977) "About orientation determination of an object in parameters of Rodrigues-Hamilton by its angular rate" [in Russian]. Izv. AN USSR. Mechanika tverdogo tela, № 3.– pp. 11 – 20.

Chow E.Y. and Willsky, A.S. (1984) "Analytical redundancy and the design of robust failure detection systems". IEEE Trans. on Auto. Control, AC-29 (7), pp. 603-614.

Coopmans, C. (2009) "Aggienav: A Small, Well Integrated Navigation Sensor System for Small Unmanned Aerial Vehicles", Proceeding of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference. IDETC/CIE 2009, August 30 - September 2, 2009, San Diego, California, USA, DETC2009- 87636.

Coopmans, C., Chao, H., Chen, Y.Q. (2009) "Design and Implementation of Sensing and Estimation Software in Aggienav, a Small UAV Navigation Platform", Proceeding of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference. IDETC/CIE 2009, August 30 - September 2, San Diego, California, USA DETC2009-87675.

Choi, K., Jang, S. and Kim, Y. (2010) "Calibration of inertial measurement units using pendulum motion", International Journal of Aeronautical and Space Science. No. 11(3), pp. 234–239.

Chulliat, A., Macmillan, S., Alken, P., Beggan, C., Nair, M., Hamilton, B., Woods, A., Ridley, V., Maus, S. and Thomson, A. (2015) "The US/UK World Magnetic Model for 2015-2020". Technical Report, National Geophysical Data Center, NOAA. doi: 0.7289/V5TB14V7.

Desai, M.N., Deckert, J.C. and Deyst, J.J. (1979) "Dual-Sensor Failure Identification Using Analytic Redundancy", AIAA Journal of Guidance and Control, Vol. 2, № 3, May-June 1979, pp. 213-220.

Deyst, J.J., Harrison, J.V., Gai, E. and Daly, K.C. (1981) "Fault Detection, Identification and Reconfiguration for Spacecraft Systems", Journal of the Astronautical Sciences, XXIX, N 2. – pp. 113 – 126.

El-Sheimy, N. , Hou, H. and Niu, X. (2008) "Analysis and modeling of inertial sensors using Allan variance," Instrumentation and Measurement, IEEE Transactions on, vol. 57, no. 1, pp.

140-149, Jan. 2008.

Farrell, J.A. (2008) *Aided navigation: GPS with high rate sensors*. New York: McGraw-Hill.

Fenton, R.G. and Willgoss, R.A. (1990) "Comparison of Methods for Determining Screw Parameters of Infinitesimal Rigid Body Motion from Position and Velocity Data". Journal of Dynamic Systems, Measurement and Control, 112. – pp. 711 – 716.

Gantmacher, F.R. (1959) *The theory of matrices*. Vol. I., English translation by K.A. Hirsch, of the Russian-language book Teoriya Matrits by F.R. Gantmacher. Chelsea publishing company, New York.

Greenspan, R.L. (1996) "Global navigation satellite systems", Ser. 207, AGARD Lecture, NATO, 1996, pp. 1 - 1, 1 - 9.

Grewal, M.S. and Andrews, A.P. (1993) *Kalman Filtering*, Englewood Cliffs, N.J.: Prentice Hall.

Grewal, M.S., Weill, L.R. and Andrews, A.P. (2001) *Global Positioning Systems, Inertial Navigation and Integration*, New York: John Wiley and Sons, Inc.

Grip, H.F., Fossen, T.I., Johansen, T.A. and Saberi, A. (2012) "Attitude Estimation Using Biased Gyro and Vector Measurements With Time-Varying Reference Vectors", IEEE Trans. on Automat. Control. 2012. 57, N5. P.1332 – 1338.

Hajiyev, C. and Guler, D.C. (2017) "Review on Gyroless Attitude Determination Methods for Small Satellites". Progress in Aerospace Sciences, 2017. [online] Available at www.elsevier.com/locate/paerosci [Accessed 08 May 2018]

IEEE Std 1293-1998, R2008. (2011) "IEEE standard specification format guide and test procedure for linear, single-axis, non-gyroscopic accelerometers", pp. 1-249. DOI: 10.1109/IEEESTD.2011.5960745

Ilnytska S. (2010) "Initialization and alignment of strapdown inertial navigation system". Polit: X international scientific practical conference for students and young scientists, 7-9 April 2010, Kyiv, Ukraine. Proceedings, Vol. 1.– pp. 41.

Ilnytska S.I. (2012) *The unmanned aerial vehicle integrated navigation system efficiency improvement*. – Manuscript. Thesis for a Ph.D. degree in technical sciences of specialty 05.22.13 – Navigation and traffic control. – National Aviation University, Kyiv, 2012.

Ixblue.com. (2018). Multi-axis position and rate tables | iXblue. [online] Available at: https://www.ixblue.com/products/multi-axis-position-and-rate-tables [Accessed 9 Jul. 2018].

Johnson, B.W. (1989) *Design and Analysis of Fault Tolerant Digital Systems*. Addison-Wesley Publishing Company.

Kantor, I.L. and Solodovnikov, A.S. (1973) Hypercomplex numbers [in Russian]. M.:Nauka.

Katsuhiko, O. (1997) Modern control engineering. Third edition, Prentice hall, New Jersey.

Kharchenko, V. and Ilnytska, S. (2010) "Efficiency analysis of integrated inertial and satellite navigation system algorithms", Mechanics of gyroscopic systems. Scientific-technical proceedings "Kyiv polytechnic institute", Issue 22. Kyiv 2010 – pp. 32-43 [In Ukrainian]

Kharchenko, V. and Ilnytska, S. (2013) "Multipurpose remotely piloted aircraft system integrated navigation system development and testing", Logistics and Transport Journal. No 3(19), pp. 85 – 90.

Kharchenko, V.P., Larin, V.B. and Ilnytska, S.I. (2012) "Calibration of accelerometers for integrated navigation system of small unmanned aerial vehicle". Systemy upravlinnya, navigatsii i zvyazku, Vypusk 1(21). V. 2. – pp. 25–29. [in Ukrainian]

Kharchenko, V.P., Larin, V.B., Ilnytska S.I. and Kutsenko, O.V. (2012 a) "Calibration of rate gyros for small integrated navigation system", Visnyk inzhenernoi academii Ukrainy, №2, pp. 30–35.

Kharchenko, V., Kondratyuk, V., Ilnytska S. and Kutsenko, O. (2012 b) "Experimental validation of small integrated navigation system", Proc. 5th World Congress "Aviation in the XXI-st century": "Safety in Aviation and Space Technologies". Vol. 2, Kyiv 2012 - pp. 3.2.30-3.2.35.

Kharchenko, V.P, Larin, V.B., Kondratyuk, V.M., Ilnytska, S.I. and Kutsenko, O.V. (2013) "Urgent problems of UAV navigation system development and practical implementation", 2013 IEEE 2nd International Conference "Actual Problems of Unmanned Air Vehicles Developments" Proceedings, K.: Education of Ukraine, pp. 157 – 161.

Kharchenko, V.B., Kondratyuk, V.M., Ilnytska, S.I. and Kutsenko, O.V. (2014) "Recommendations to UAV navigation system test validation and some practical results", 2014 IEEE 3rd International Conference "Actual Problems of Unmanned Air Vehicles Developments" Proceedings, K.: Education of Ukraine, pp. 31 – 34.

Kortunov, V., Dybska, I., Proskura, G. and Kravchuk, A. (2008) "Integrated Mini INS Based on MEMS Sensors for UAV Control", Materials of the Jubilee 15th Saint Petersburg's International Conference on Integrated Navigation Systems. Saint Petersburg, Russia, May 26–28, 2008.– pp.276 – 278.

Krasovskiy, A.A. (1993) "Analytic adjustment of accelerometer based SINS" [in Russian], Izv. RAN. Tekhnicheskaya kibernetica, № 6.- pp. 39 - 47.

Kreinovich, V., Jacob, C., Dubois, D., Cardoso, J. and Ceberio, A. (2012) "Failure Analysis of a Complex System Based on Information about Subsystems with Potential Applications to Aircraft Maintenance", Appl.and Comput. Math, 2012, 11, N 2. – pp. 165 – 179.

Kuznetsov, N.Yu. and Shumskaya, A.A. (2013) "Risk estimation of redundant system failure by the accelerated modeling method", Problemy upravleniya i informatiki. – 2013.– № 3.– pp. 50 – 62. [in Russian]

Lachapelle, G. (1996) "Navigation Accuracy for Absolute Positioning", AGARD-IS-207, NATO, 1996, pp. 4-1, 4-10.

Larin, V.B. (1992) "Construction of a nonnegative-definite solution of the Lyapunov equation", Doklad at the SSSR Acad. Sci.., 332, № 3.- pp. 469 – 471. [in Russian]

Larin, V.B. (1993) "Construction of a solution of the generalized Lyapunov equation", Russian Acad. Sci. Dokl. Math, 47, № 3.- pp.17 - 20.

Larin, V.B. (1994) "Singular decomposition in failures identification task", Kibernetika i vychislitelnaya tekhnika, 1994, Iss. 101. – pp. 86 – 88. [in Russian]

Larin, V.B. (1999) "On Integrating Navigation Systems", Journal of Automation and Information Sciences, 31, N 10. – pp. 95 – 98.

Larin, V.B. (2001) "Attitude-Determination Problems for a Rigid Body", Int. Appl. Mech., 37, N7. – pp. 870 – 898.

Larin, V.B. (2006) "Tasks of orientation determination of a rigid body", [in Russian], Uspekhi mekhaniki, Vol. 2. – pp. 512 – 538.

Larin, V.B. (2015) "Identification of Faults of Navigation Sensors", Int. Appl. Mech., 51, N 6. – pp. 696-701.

Larin, V.B. and Aliev, F.A. (1993) "Construction of square root factor for solution of the Lyapunov matrix equation", Systems & Control Letters, № 20. - pp. 109 – 112.

Larin, V.B. and Naumenko, K.I. (1982) "About integration of kinematic equations in Rodrigues-Hamilton parameters [Ob integrirovanii kinematicheskikh uravneniy v parametrakh Rodriga-Hamiltona]", Kn.: Navigatsiya i upravleniye, K.: In-t matematiki AN USSR, pp. 62 – 71. (in Russian)

Larin, V.B. and Naumenko, K.I. (1983) "About orientation determination of a rigid body [Ob opredelenii orientatsii tverdogo tela]", Izv. AN SSSR, Mekh.Tv. Tela, No 3, pp. 24 – 32. (in Russian)

Larin, V.B. and Naumenko, K.I. (1987) "Suboptimal filtering in determining the attitude of a rigid body", Izv. AN SSSR, Mekh.Tv. Tela, No. 1, pp. 32–42.

Larin, V.B. and Tunik, A.A. (2010 a) "About Inertial-Satellite Navigation System without Rate Gyros", Appl. and Comp. Math., 9, N 1. – pp. 3 – 18.

Larin V.B. and Tunik A.A. (2010 b) "On Inertial Navigation System Error Correction". Problemy upravleniya i informatiki, №4, pp. 130–142. [in Russian]

Larin V.B. and Tunik A.A. (2012). "On Inertial Navigation System Error Correction", Int. Appl. Mech., 48, N 2. – pp. 213 –223.

Larin, V.B. and Tunik, A.A. (2013) "On Inertial Navigation System without Angular-Rate Sensors", Int. Appl. Mech., 49, N4. – pp. 488 – 500.

Larin, V.B. and Tunik, A.A. (2015) "Fault–Tolerant Strap-Down Inertial Navigation Systems with External Corrections", Applied and Computational Mathematics. An International Journal, Vol.14, No.1, 2015, pp. 23 – 38.

Laub, A.J. and Shiflett, G.R. (1983) "A Linear Algebra Approach to the Analysis of Rigid Body Velocity from Position and Velocity Data", Transactions of the ASME, 1983, 105. pp. 92 – 95.

Lawson, C.L. and Hanson, R.J. (1974) *Solving Least Squares Problem*. Prentice-Hall, Englewood Cliffs, NJ.

Lebedev, D.V. (1992) "Identification of sensitive elements failures in inertial navigation system". Automatika, 1992, № 2, pp. 39 – 44. [in Russian]

Lee, R.C.K. (1964) *Optimal Estimation, Identification and Control*. Cambridge: The M.I.T. Press. 1964. N 28. 176 p.

Lee, S.J, Tunik, A.A., Kim, J.C. (1999) "Determination of the Error Parameters of Strap-down INS Sensor Unit Using the 3-axis Motion Table". Proceedings of KSAS (Korean Society of Aeronautics and Space Scientists), Spring Annual Meeting, Daejeon, 1999, KARI, pp. 485 – 488.

Looneys, M. (2010) "A simple calibration for MEMS gyroscopes". Analog Devices. EDN EUROPE, July 2010, pp. 28 – 31. [online] Available at http://www.analog.com/media/en/technical-documentation/technical-articles/GyroCalibration_EDN_EU_7_2010.pdf [Accessed 9 July 2018]

Lurie, A.I. (2002) Analytical Mechanics, Springer, Berlin–New York.

Mathworks.com. (2018). MATLAB Documentation. [online] Available at: https://www.mathworks.com/help/ [Accessed 8 Jul. 2018].

Meleshko, V.V. and Nesterenko, O.I. (2011) *Strapdown inertial navigation systems*, Uchebnoe posobiye. Kirovograd: Polymed-Servis. [in Russian]

Mhaskar, P., Liu, J. and Christofides, P.D. (2013) Fault-Tolerant Process Control Methods and Applications. Springer-Verlag London.

Ngdc.noaa.gov. (2018). World Magnetic Model | NCEI. [online] Available at: https://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml [Accessed 8 Jul. 2018].

Novatel.com. (2018 a). OEMVR family Installation and Operation User Manual [online] Available at: https://www.novatel.com/assets/Documents/Manuals/om-20000093.pdf [Accessed 9 Jul. 2018].

Novatel.com. (2018 b). Waypoint products group. A NovAtel Precise Positioning Product. GrafNav / GrafNet User Manual. [online] Available at: https://www.novatel.com/assets/Documents/Waypoint/Downloads/GrafNav-GrafNet-User-Manual-870.pdf [Accessed 9 Jul. 2018].

Nxp.com. (2013). K60P144M100SF2V2 Data Sheet: Technical Data, Rev. 3. [online] Available at: https://www.nxp.com/docs/en/data-sheet/K60P144M100SF2V2.pdf [Accessed 9 Jul. 2018].

Onishchenko, S.M. (1983) Hypercomplex Numbers in Inertial Navigation Theory [in Russian], Naukova Dumka, Kyiv.

Petovello, M.G. (2003) Real-Time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation, Department of Geomatics Engineering. Thesis for the degree of Doctor of Philosophy / Mark G. Petovello, April 2003. – 269 p. http://www.geomatics.ucalgary.ca/links/GradTheses.html

Phillips, R.E. and Schmidt, G.T. (1996) "GPS/INS Integration," Ser. 95, AGARD Lecture, NATO, pp. 9 - 1, 9 - 18.

Potter, J.E. and Suman, M.C. (1977) "Thresholdless Redundancy Management with Arrays of Skewed Instruments", AGARDOGRAPH-224, Integrity in Electronic Flight Control Systems, 1977, pp. 15-1 to 15-24.

Sbg-systems.com. (2018). IG-500N: miniature INS/GPS. [online] Available at: https://www.sbg-systems.com/products/ig500n-miniature-ins-gps [Accessed 9 Jul. 2018].

Schmidt, G.T. (1978) "Strapdown inertial systems – theory and applications. Introduction and overview," Ser. 95, AGARD Lecture, NATO, pp. 1 - 1, 1 - 6.

Schmidt, G.T. (2008) "INS/GPS Technology Trends", NATO RTO Lecture Series, RTO-EN-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Prague, October 2008, pp.1 - 1, 1 - 18.

Schmidt, G.T. (2011) "INS/GPS Technology Trends", NATO RTO Lecture Series, RTO-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Kiev, September 2011, pp.1 - 1, 1 - 23.

Schmidt, G. and Phillips, R. (2008) "INS/GPS Integration Architecture Performance Comparison", NATO RTO Lecture Series, RTOEN-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Prague, October 2008, pp. 5 - 1, 5 - 18.

Schmidt, G.T. and Phillips, R.E. (2011 a) "INS/GPS Integration Architectures ", NATO RTO Lecture Series, RTO-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Kiev, September 2011, pp. 5 - 1, 5 - 18.

Schmidt, G.T. and Phillips, R.E. (2011 b) "INS/GPS Integration Architecture Performance Comparison", NATO RTO Lecture Series, RTO-SET-116, Low-Cost Navigation Sensors and Integration Technology, Publ., Kiev, September 2011, pp. 6 - 1, 6 - 22.

Shin, E.H. (2005) "Estimation techniques for low-cost inertial navigation," Ph.D. dissertation, University of Calgary, Alberta, Canada, may 2005.

Siouris, G. (1993) *Aerospace avionics systems*. San Diego, Calif.: Academic Press.

Siouris, G.M. (2007) *Aerospace Avionics Systems: a modern synthesis*. Academic Press, Inc.

Shuster, M.D. (1993) "Survey of attitude representations". Journal of the Astronautical Sciences, 41(4), pp. 439 – 517.

Tanaka, S. and Muller, J.C. (1990) "Fault detection in linear discrete dynamic systems by a pattern recognition of generalized-likelihood-ratio", Transactions of the ASME, Journal of Dynamic Systems. Measurement and Control, 1990, V. 112, pp. 276 – 292.

Tao, G. (2014) "Direct adaptive actuator failure compensation control: a tutorial" in Journal of Control and Decision, 2014, Vol. 1, No. 1, pp. 75–101.

Titterton, D.H. and Weston, J.L. (2004) *Strapdown Inertial Navigation Technology*, Peter Peregrinus Press, London.

Tunik, A.A. and Valdenmayer, G.G. (2011) "Analysis of procedures for covariance matrix factorization of problems of extended kalman filter in strapdown-satellite navigation system" Systemy upravlinnya, navigatsiyi ta zvyazku, 2011. vyp. 1(17). - pp. 10 - 15. [in Ukrainian]

Thebault, E. (2014). IAGA V-MOD Geomagnetic Field Modeling: International Geomagnetic Reference Field IGRF-12. [online] Ngdc.noaa.gov. Available at: https://www.ngdc.noaa.gov/IAGA/vmod/igrf.html [Accessed 8 Jul. 2018].

Voyevodin, V.V. and Kuznetsov, Yu.A. (1984) *Matrices and calculations*, M.: Nauka. [in Russian]

Watanabe, K. and Tzafestas, S.G. (1989) "New computationally efficient formula for backward-pass fixed-interval smoother and its UD factorization algorithm", IEEE Proceedings, Vol. 136, No. 2, pp. 73 – 78.

Wittenburg, J. (1977) *Dynamics of Systems of Rigid Bodies*, B.G. Teunbern, Stuttgart.

Woods, A. (2018). WMM Magnetic Elements (2015-2020). [online] Geomag.org. Available at: http://geomag.org/models/wmm-field.html [Accessed 8 Jul. 2018].

APPENDIX A

Here we derive the relations (3.10)-(3.15) presented in Sec. 3.1.2.

Let us derive the algorithm with the use of QR factorization (Sec. 3.1.2) for the case where R_k is an invertible matrix (Larin and Tunik, 2010 b). First, we will make some transformations of Eq. (3.6). To this end, we ill use the following information. For some block matrix Z of the form

$$Z = \begin{bmatrix} P & -M \\ A & B \end{bmatrix},$$

according to the Frobenius formula, there are two equivalent (to each other) inverse matrices Z^{-1} that can be written as (Gantmacher, 1959), (Aliev, Larin, Naumenko and Suntsev, 1978):

$$Z^{-1} = \begin{bmatrix} P^{-1} - P^{-1}M(B + AP^{-1}M)^{-1}AP^{-1} & P^{-1}M(B + AP^{-1}M)^{-1} \\ -(B + AP^{-1}M)^{-1}AP^{-1} & (B + AP^{-1}M)^{-1} \end{bmatrix} \tag{3.47}$$

$$Z^{-1} = \begin{bmatrix} (P + MB^{-1}A)^{-1} & (P + MB^{-1}A)^{-1}MB^{-1} \\ -B^{-1}A(P + MB^{-1}A)^{-1} & B^{-1} - B^{-1}A(P + MB^{-1}A)^{-1}MB^{-1} \end{bmatrix} \tag{3.48}$$

For matrix Z, let us introduce the notation: $A = H,\quad M = H^T,\ P = P_k(-)^{-1} = P_k^{-1}$ (to simplify the notation, we have omitted the "minus" sign in the brackets), . We will rearrange matrices Z^{-1} in (3.47)–(3.48) taking into account the introduced notation:

$$Z^{-1} = \begin{bmatrix} P_k - P_kH^T(R_k + HP_kH^T)^{-1}HP_k & P_kH^T(R_k + HP_kH^T)^{-1} \\ -(R_k + HP_kH^T)^{-1}HP_k & (R_k + HP_kH^T)^{-1} \end{bmatrix}; \tag{3.49}$$

$$Z^{-1} = \begin{bmatrix} (P_k^{-1} + H^TR_k^{-1}H)^{-1} & (P_k^{-1} + H^TR_k^{-1}H)^{-1}H^TR_k^{-1} \\ -R_k^{-1}H(P_k^{-1} + H^TR_k^{-1}H)^{-1} & R_k^{-1} - R_k^{-1}H(P_k^{-1} + H^TR_k^{-1}H)^{-1}H^TR_k^{-1} \end{bmatrix}. \tag{3.50}$$

Comparing the upper right blocks of matrices Z^{-1} in (3.49)–(3.50) yields

$$K_k = (P_k^{-1} + H^TR^{-1}H)^{-1}H^TR_k^{-1}. \tag{3.51}$$

By analogy, we will also transform Eq. (3.8). First, we will substitute (3.6) into (3.8) and make respective simplifications. Hereinafter, by P_k we mean $P_k(-)$:

$$\begin{aligned} P_k(+) &= P_k - K_k\left(HP_kH^T + R_k\right)K_k^T = \\ &= P_k - P_kH^T\left(HP_kH^T + R_k\right)^{-1}\left(HP_kH^T + R_k\right)\left(P_kH^T\left(HP_kH^T + R_k\right)^{-1}\right)^T = \\ &= P_k - P_kH^T\left(P_kH^T\left(HP_kH^T + R_k\right)^{-1}\right)^T = \\ &= P_k - P_kH^T\left(\left(HP_kH^T + R_k\right)^{-1}\right)^T\left(P_kH^T\right)^T = \\ &= P_k - P_kH_k^T\left(\left(HP_kH^T + R_k\right)^{-1}\right)HP_k \end{aligned} \tag{3.52}$$

In the transformations, we have used the following properties of transposition of matrices $\left(A^T\right)^T = A$, $(AB)^T = B^T A^T$, $C^T = C$ if matrix C is symmetric. Comparing the upper left blocks of matrices Z^{-1} in (3.49)–(3.50) yields

$$P_k(+) = (P_k^{-1} + H^T R_k^{-1} H)^{-1} \tag{3.53}$$

We can rearrange Eq. (3.51) with regard for (3.53) as

$$K_k = P_k(+) H^T R_k^{-1}. \tag{3.54}$$

Now, denote by m_k, p_k, q_k, η_k the Cholesky multipliers of the corresponding matrices $P_k(-) = m_k m_k^T$, $\mathrm{P}_k(+) = p_k p_k^T$, $\mathrm{Q}_k = q_k q_k^T$, $\mathrm{R}_k = \eta_k \eta_k^T$. Taking into account the chosen notation, we will rearrange relation (3.52) as follows:

$$P_k(+) = p_k p_k^T = m_k (I - m_k^T H^T (H m_k m_k^T H^T + R_k)^{-1} H m_k) m_k^T. \tag{3.55}$$

To simplify the relation, we will use matrices Z^{-1} (3.47) – (3.48) and introduce the following notation: $P^{-1} = I$, $M = m_k^T H^T$, $B = R_k$, $A = H m_k$. Comparing the corresponding elements of the matrices yields

$$\begin{aligned} P_k(+) = p_k p_k^T &= m_k (I - m_k^T H^T (H m_k m_k^T H^T + R_k)^{-1} H m_k) m_k^T = \\ &= m_k \left(I + m_k^T H^T R_k^{-1} H m_k\right)^{-1} m_k^T. \end{aligned} \tag{3.56}$$

It is obvious that elements $m_k(\ldots)m_k^T$ were not taken into account in the comparison. Now, let us present the bracketed expression as a product of two rectangular matrices:

$$\begin{aligned} &I + m_k^T H^T R_k^{-1} H m_k = N_k N_k^T; \\ &N_k = \begin{bmatrix} I & m_k^T H^T \eta_k^{-1} \end{bmatrix}. \end{aligned}$$

By means of orthogonal matrix U and QR factorization algorithm, we will rearrange matrix N^T as follows:

$$\begin{bmatrix} \Lambda_k \\ 0 \end{bmatrix} = U_k N_k^T, \tag{3.57}$$

where Λ_k is an invertible matrix. Now we will rearrange Eq. (3.56) taking into account (3.57) as

$$\begin{aligned} p_k p_k^T &= m_k \left(I + m_k^T H^T R_k^{-1} H m_k\right)^{-1} m_k^T = \\ &= m_k \left(N_k U_k^T U_k N_k^T\right)^{-1} m_k^T = \\ &= m_k \left(\Lambda_k^T \Lambda_k\right)^{-1} m_k^T = m_k \left(\Lambda_k\right)^{-1} \left(\Lambda_k^T\right)^{-1} m_k^T = \\ &= m_k \left(\Lambda_k\right)^{-1} \left(\Lambda_k^{-1}\right)^T m_k^T. \end{aligned}$$

From here it is obvious that

$$p_k = m_k \Lambda_k^{-1}. \tag{3.58}$$

Similarly, we will represent the right-hand side of Eq. (3.56) as a product of two rectangular matrices and will use QR factorization of these matrices, performed by the orthogonal matrix Z_k.

$$m_{k+1}m_{k+1}^T = T_k T_k^T; \qquad T_k = [\Phi_k p_k \quad q_k]; \tag{3.59}$$

$$\begin{bmatrix} X_k^T \\ 0 \end{bmatrix} = Z_k T^T; \tag{3.60}$$

$$m_{k+1}m_{k+1}^T = T_k Z_k^T Z_k T_k^T = X_k X_k^T \, .$$

So,

$$m_{k+1} = X_k . \tag{3.61}$$

Thus, according to the given m_k, η_k and Eqs. (3.57 – 3.58), we calculate the multiplier p_k and then we can find the multiplier m_{k+1} according to (3.59 – 3.61).