DEPARTMENT OF DEFENCE

DEFENCE INSTITUTE „PROF. TZVETAN LAZAROV”

ing. Alexander Genchov Ranov

A MODEL FOR ROUTING AND TRAFFIC CONTROL OF AN AUTONOMOUS COMBAT PLATFORM

DISSERTATION ABSTRACT

for the acquisition of the educational and scientific degree Ph.D.

in the scientific specialty "Automated systems for information processing and management"

Research supervisor:

Assoc. Prof. Dr. Alexander Assenov Kolev

Sofia - 2024

The dissertation work was accepted and directed for defense after discussion by the Scientific Council of the Directorate "Development of C4I Systems" at the Institute of Defense "Professor Tsvetan Lazarov" with a protocol № 71 / 19.12.2023 г.

The dissertation contains an introduction, four chapters with results and conclusions to each of them, results and contributions of the whole work and a conclusion, set out in 147 pages of academic text, 62 figures and 11 tables, 12 appendices. The bibliography includes 111 titles in Bulgarian and English. The total volume of the dissertation is 170 pages.

The numbers of chapters, figures, tables, formulas and cited literature in the abstract correspond to those in the dissertation work. The numbering of the literature used in the abstract corresponds to that in the dissertation.

The defense of the dissertation will take place on ....
...............hours in the hall.................of the Defense Institute "Professor Tsvetan Lazarov", Sofia, at open session of a scientific jury in composition:
1. Prof. Dr. Georgi Sotirov
2. Prof. Dr. Rosen Iliev
3. Col. Assoc. Prof. Dr. Sevdalin Spasov
4. L. Col. Assoc. Ph.D. Angel Genchev
5. Assoc. Prof. Dr. Ivan Hristozov

Reserve Members:
1. Col. Assoc. Prof. Dr. Ivo Radulov
2. Assoc. Prof. Dr. Maya Bozhilova

Research supervisor:
Assoc. Prof. Dr. Alexander Asenov Kolev

Author:
Eng. Alexander Genchov Ranov

Title: A model for routing and traffic control of an autonomous combat

platform

I. GENERAL CHARACTERISTICS OF THE DISSERTATION

ACTUALITY
One of the latest high-tech trends worldwide is the creation of autonomous vehicles. Results of tests and trial operation of similar projects in the civil sphere are known from publicly available sources. It is apparent that the initial optimistic expectations are met with setbacks of a different nature, which make it problematic to fit the developed autonomous vehicle control systems into the existing land transport safety requirements and regulations. In terms of military applications, autonomous platforms could be productive in certain transport-combat tasks. Information from current military conflicts shows successful attempts to implement combat vehicles with different levels of autonomy, operating in air, water, as well as land-based environments. Within the scope of the assigned dissertation topic, the research is focused on the issues of the military application of autonomous ground platforms.

SUBJECT OF THE DISSERTATION
The subject of the research in the dissertation is the synthesis of a routing model when preparing a transport plan for the movement of a ground autonomous combat platform, as well as to propose methods for the implementation of traffic control during the execution of a transport mission.

OBJECT OF RESEARCH
The object of research is the modeling of the routing process of a ground autonomous combat platform by applying information processing of digital data on the height of the earth's surface within the scope of carrying out a transport-combat task. Motion control processes involve processing primary data from a set of on-board sensors: for inertial navigation, computer vision, energy security.

PURPOSE AND OBJECTIVES OF THE RESEARCH
The aim of the dissertation work is to create a routing model, and to propose methods for controlling the movement of an autonomous combat platform, taking into account the specifics of a military application.
To achieve the goal, the following tasks were completed:
• To develop a mathematical apparatus and to algorithmize procedures for routing a ground platform without using the existing road network and avoiding visibility from an enemy observation post;
• To synthesize a model for processing the information in the routing of an autonomous combat platform;
• The model should be validated in the conditions of a test software environment;
• To propose methods of inertial navigation and algorithms for controlling the movement of an autonomous platform;

• To present an application of the model in the performance of transport-combat tasks.

II. BRIEF CONTENTS OF THE DISSERTATION

CHAPTER 1: A Survey of Current Advances in Theory and Practice in the Implementation of Autonomous Platforms

The broad theme of research on the achievements of autonomous platforms can be summarized by the potential advantages of a built system of autonomous vehicles: safety, optimization of street traffic, reduced costs.

The navigation system for mobile robots and autonomous platforms is functional if there is a correct determination of the current position of the moving apparatus, as well as the presence of a sufficiently accurate and adequate map of the working environment. The navigation process consists of two levels of influence: at the global level it is traffic planning, at the local level it is the implementation of traffic control. Motion planning needs a digital terrain model, or an electronic map of the work environment. The presence of these prerequisites allows to determine the coordinates of the geometric positions of the movement path, along which the mobile robot should pass following them from the starting point to the set end position of movement. Motion control is a process in which data from onboard sensors is applied and a direction is determined in which various unforeseen objects or difficult terrain sections can be bypassed.

The military application of autonomous platforms has been brought to the attention of relevant specialists. Objects of consideration in this case, in addition to the main challenges, are also some special requirements. The tactical and technical requirements for the autonomous platforms with military purpose can be directed to the routing for movement in the mode without using the existing road network, requirements for the passage bridge , based on the design data of the machine, combat radius and load capacity, weight corresponding to the intended class of ballistic protection and many. others. From a military point of view, there is the question of determining a route using the defensive properties of the area. In a more limited view, this would involve conducting a spatial analysis of the terrain with the determination of areas of visibility, by notified enemy observer positions. Avoidance of unmapped obstacles is achievable with the application of 3D and lidar computerized observation, simultaneous localization and mapping, and other methods close in terms of problems to systems with the application of artificial intelligence. In a purely military aspect, the emphasis should primarily be on the so-called passive methods for detecting unmapped obstacles, in order to minimize unmasking signs. Such a method, despite certain limitations, is the application of a stereo camera and software solutions for computer vision.
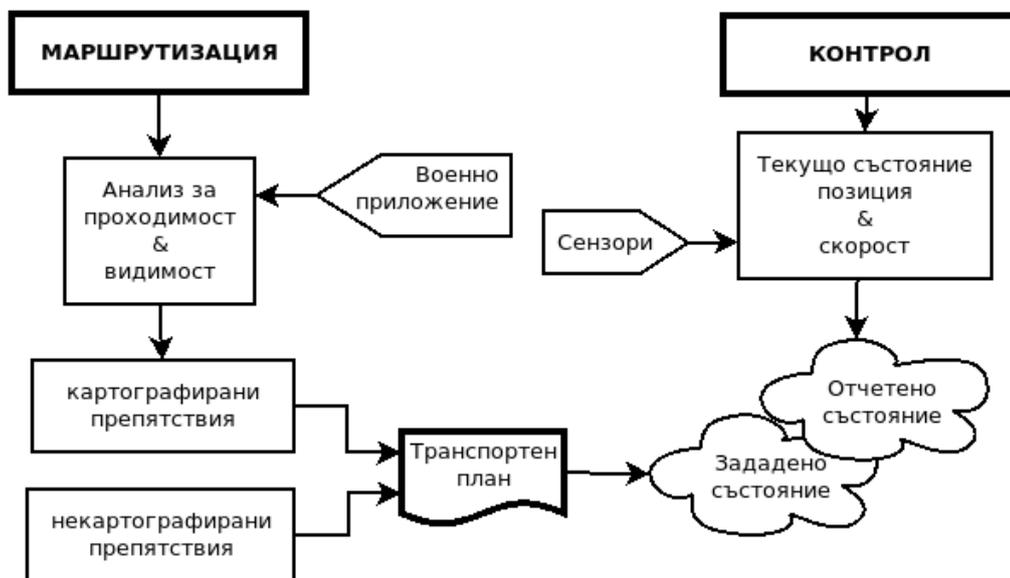
Fig. 1.7. Summary structure of studies

The study of known approaches to creating autonomous systems gives an overview of the problem. It is possible to estimate the necessary scientific and technical resources for conducting full-fledged research. Restrictions have been placed on the development of the dissertation work, within the scope of the subject of the present scientific research, and taking into account the available material provision. The goal of the dissertation research is defined and the tasks leading to the fulfillment of the goal thus set are set.

CHAPTER 2: Autonomous Combat Platform Routing Issues and Solutions

We consider a path planning approach hidden from a specific observation point of the likely adversary and according to the tactical-technical capabilities of the autonomous platform to overcome the slope.

The "stealth movement planning" task defined above implies specific inputs, among which the minimum required are:

• Starting position of the autonomous platform, designated as Start Point (SP);

• Final position of the autonomous platform, designated as End Point (EP);

• Height of the autonomous platform above the ground, Platform Height (PH);

• Maximum tilt angle (MSA);

• Observer Position (OP);

• Maximum detection distance of similar target type, maximum observer range (MOR);

• Height of the observer's position above the earth's surface (OH).

The necessary digital data for the height of the earth's surface are presented in matrix form in the ASCII / Binary Grid format [24] and are

available both from public sources [97] and from departmental specialized databases.

A three-dimensional analysis of the digital data for the height of the earth's surface in a given area of interest (Area of Interest, AoI) was performed, applying algorithms for:

• Determination of the slope of the earth's surface in the area of interest;

• Determination of visibility zones relative to a known observer point;

• Determining a route according to the maximum tilt angle of the autonomous platform and avoiding the designated visibility areas.
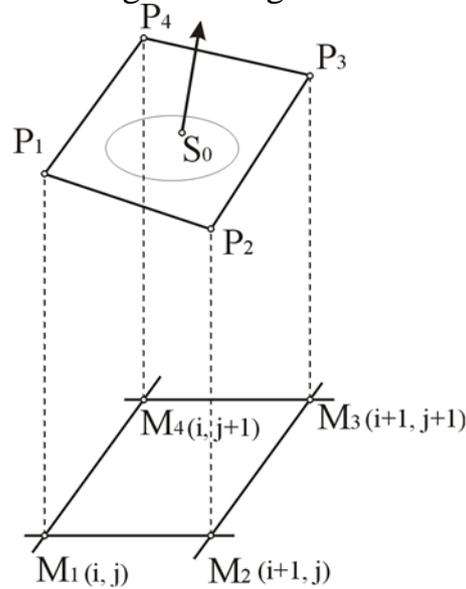
Fig. 2.1. Digital surface data and approximating plane S_o

To verify the results achieved, the digital analysis product was converted into a standard vector file format for geographic data representation (ShapeFile) and visualized using specialized open source software.

We assume that the numerical data for the height of the earth's surface are represented in the form of a rectangular two-dimensional matrix {M}, [18] and in fig. 2.1 shows one of its unit cells with nodes M1, M2, M3, M4}. We also assume that for each of the nodes M1, M2, M3, M4, we know the altitude with sufficient accuracy for practice. We also know that the matrix {M} is georeferenced in a known coordinate system. Let the total number of nodes in the height matrix be N = m x n, where m is the number of columns and n is the number of rows in the two-dimensional matrix.

Determining the slope of the earth's surface using an approximating plane

Let the unit cell of the surface in the digital model be defined by the points:

$$
\left.
\begin{aligned}
P_1\left(x_i, y_j, z_{i,j}\right) \\
P_2\left(x_{i+1}, y_j, z_{i+1,j}\right) \\
P_3\left(x_{i+1}, y_{j+1}, z_{i+1,j+1}\right) \\
P_4\left(x_i, y_{j+1}, z_{i,j+1}\right)
\end{aligned}
\right\}
\tag{2.1}
$$

We need to determine the normal vector $\vec{n}$ on the approximating plane $S_0\{P1, P2, P3, P4\}$.

By applying the method of least squares [45], the required approximating plane S_0 is represented by the expression:

$$S_0 \equiv z = ax + by + c, \tag{2.2}$$

And also with the coefficients a and b, for which the function is minimized

$$F \equiv \sum_{k=1}^{4}(aX_k + bY_k - Z_k)^2 \tag{2.3}$$

where:

$$
\left.
\begin{aligned}
X_1 &= X_4 = x_i \\
X_2 &= X_3 = x_{i+1} \\
Y_1 &= Y_2 = y_j \\
Y_2 &= Y_4 = y_{j+1} \\
Z_1 &= z_{i,j} \\
Z_2 &= z_{i+1,j} \\
Z_3 &= z_{i+1,j+1} \\
Z_4 &= z_{i,j+1}
\end{aligned}
\quad
\begin{aligned}
X_1 &= X_4 = x_i \\
X_2 &= X_3 = x_{i+1} \\
Y_1 &= Y_2 = y_j \\
Y_2 &= Y_4 = y_{j+1} \\
Z_1 &= z_{i,j} \\
Z_2 &= z_{i+1,j} \\
Z_3 &= z_{i+1,j+1} \\
Z_4 &= z_{i,j+1}
\end{aligned}
\right\}
\tag{2.4}
$$

To determine the coefficients a and b for which F has a minimum value, the system of equations is solved:

$$
\left|
\begin{aligned}
\frac{\partial F}{\partial a} &\equiv 2\sum_{k=1}^{4}(aX_k - bY_k - Z_k)X_k = 0 \\
\frac{\partial F}{\partial b} &\equiv 2\sum_{k=1}^{4}(aX_k - bY_k - Z_k)Y_k = 0 \\
\frac{\partial F}{\partial c} &\equiv 2\sum_{k=1}^{4}(aX_k - bY_k - Z_k) = 0
\end{aligned}
\right.
\tag{2.5}
$$

After determining the coefficients a and b by applying the expressions (2.3), (2.4), and (2.5) we find that the surface (2.2) has a normal vector

$$\vec{N} = (-a, -b, 1) \tag{2.6}$$

As a result, the angle γ on the slope of the plane S_0 relative to the horizon is determined by the expression:

$$\cos\gamma = \frac{1}{\sqrt{a^2 + b^2 + 1}} \tag{2.7}$$

Therefore, the ground surface angle for the investigated unit cell from the digital surface model in degrees, γ^{deg} is defined as:

$$\gamma^{deg} = acos\left(\frac{1}{\sqrt{a^2 + b^2 + 1}}\right)\frac{180}{\pi} \tag{2.8}$$

Determining the slope of the earth's surface using triangulation

Let the surface S be defined by the surfaces S_1 and S_3, which are defined by the triangles $P_1P_2P_4$ and $P_3P_4P_2$ (Fig. 2.2a), or by the surfaces S_2 and S_4, which are defined by the triangles $P_2P_3P_1$ and $P_4P_1P_3$ (Fig. 2.2b).

We will determine $\cos\gamma_k$ for each of the planes S_k, k=1,2,3,4.

The plane S_1 is defined by the points

$P_1(x_i, y_i, z_{i,j})$, $P_2(x_{i+1}, y_i, z_{i+1, j})$, $P_4(x_i, y_{j+1}, z_{i, j+1})$, and the expression $z = a_1x+b_1y$, where:

$$a_1 = \frac{\partial z}{\partial x}(M_1) = \frac{z_{i+1,j}-z_{i,j}}{\Delta x}, \tag{2.9}$$

$$b_1 = \frac{\partial z}{\partial y}(M_1) = \frac{z_{i,j+1}-z_{i,j}}{\Delta y}. \tag{2.10}$$

The angle of inclination γ_1 for the plane S_1 similarly to the expression (2.7) is defined by:

$$\cos\gamma_1 = \frac{1}{\sqrt{a_1^2+b_1^2+1}} \tag{2.11}$$

Similarly, we find the angles of inclination γ_k on the plains S_k, k=2,3,4 represented by their cosines:

$$\cos\gamma_2 = \frac{1}{\sqrt{a_1^2+b_2^2+1}} \tag{2.12}$$

$$\cos\gamma_3 = \frac{1}{\sqrt{a_2^2+b_2^2+1}} \tag{2.13}$$

$$\cos\gamma_4 = \frac{1}{\sqrt{a_2^2+b_1^2+1}} \tag{2.14}$$

where a_1, b_1 are defined by dependencies (2.9), (2.10), and

$$a_2 = \frac{z_{i+1,j+1}-z_{i,j+1}}{\Delta x}, \tag{2.15}$$

$$b_2 = \frac{z_{i+1,j+1}-z_{i+1,j}}{\Delta y}. \tag{2.16}$$
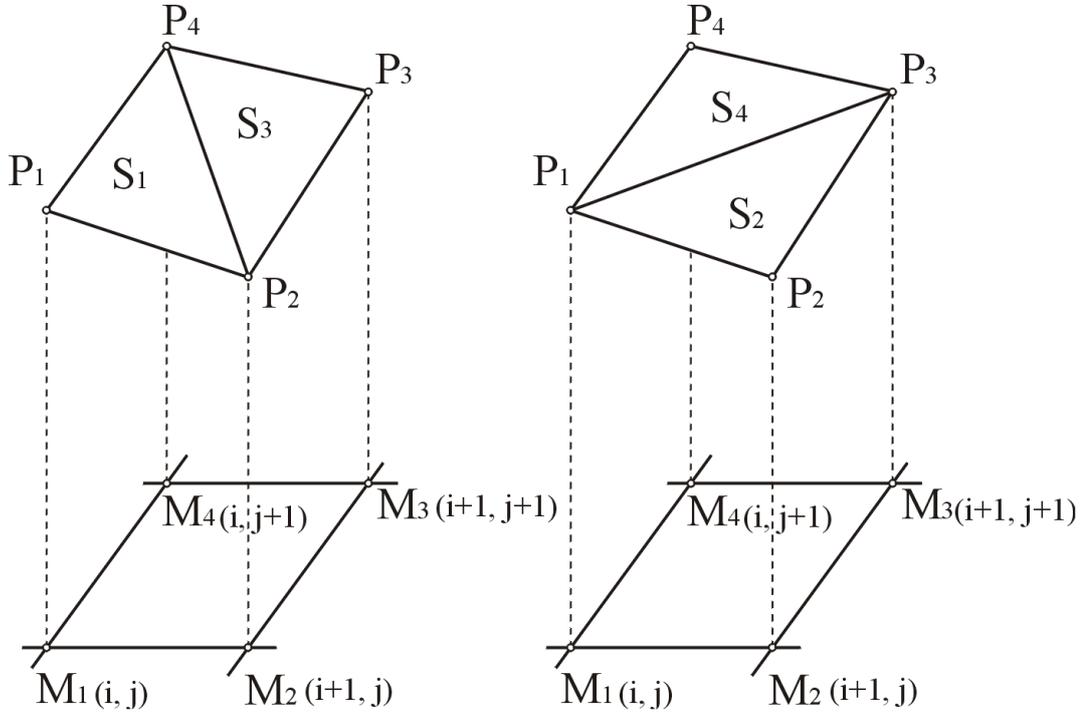
Fig. 2.2. Triangulation of a cell from the height matrix

After determining the $\cos\gamma_k$, k=1,2,3,4 for expected magnitude of $\cos\gamma$ of the cell GRID (i, j) value can be taken:

$$\cos\gamma = \min(\cos\gamma_1, \cos\gamma_2, \cos\gamma_3, \cos\gamma_4). \tag{2.17}$$

The slope of the earth's surface for the study clect from the height matrix in decimal degrees γ^{deg} presents itself as:

$$\gamma^{deg} = acos\big(min(\cos\gamma_k)\big)\frac{180}{\pi} \tag{2.18}$$

We denote the patency matrix by the symbol A. Applying expressions (2.8) or (2.18), for each cell of {M} we have:

$$A \equiv \left\{\gamma_{i,j}^{deg}\right\}, \ i = 1,2\ldots,m-1, \ j = 1,2\ldots,n-1 \tag{2.23}$$

Defining a field of view

In order to be able to define a visibility area, it is necessary to know the height of the earth's surface in all coordinates of the modeled surroundings. Considering that the height model of the earth's surface is represented by discrete values of the nodes of a two-dimensional rectangular matrix {M}, and each unit cell is represented by four points with three-dimensional coordinates (2.1), it is necessary to be able to know the height z at an arbitrary point P(x, y, z) with known x and y coordinates at the boundaries of the region {P₁, P₂, P₃, P₄}, fig. 2.4. Thus, the stated requirement is achieved by applying the mathematical formulation known as bilinear interpolation. [76, 86]

$P_4(X_{i,j+1}, Y_{i,j+1}, Z_{i,j+1})$ $P_3(X_{i+1,j+1}, Y_{i+1,j+1}, Z_{i+1,j+1})$

$P(x, y, z)$

$P_1(X_{i,j}, Y_{i,j}, Z_{i,j})$ $P_2(X_{i+1,j}, Y_{i+1,j}, Z_{i+1,j})$

Fig. 2.4. Height determination with bilinear interpolation
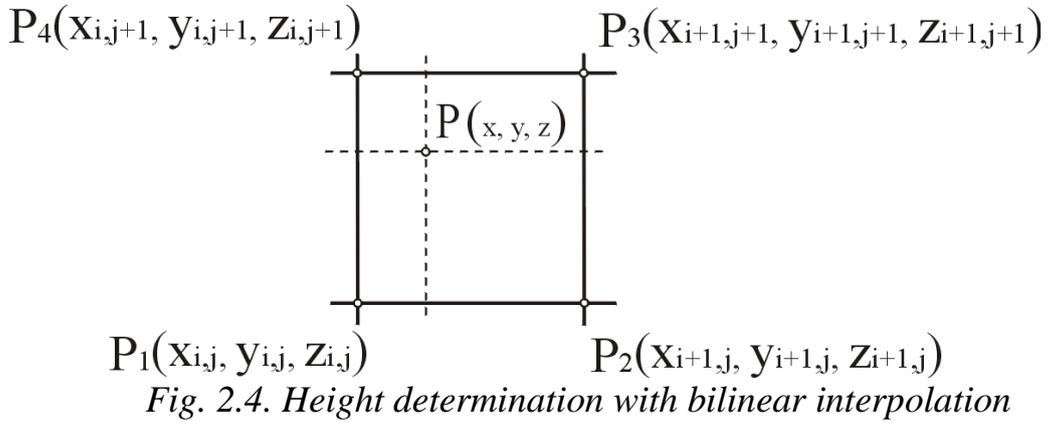
Applying the notation on expressions (2.4), we can define:
$\Delta X = X_2 - X_1$, $\Delta Y = Y_2 - Y_1$.

We introduce dimensionless coefficients t, u as:

$$t \equiv \frac{x - X_1}{\Delta X}, \tag{2.24}$$

$$u \equiv \frac{y - Y_1}{\Delta Y}. \tag{2.25}$$

Applying the thus defined dimensionless coefficients t, u, the required height z is determined by bilinear interpolation as:

$$z = (1-t)(1-u)Z_1 + t(1-u)Z_2 + tuZ_3 + (1-t)Z_4. \tag{2.26}$$

By applying the expression (2.26), it is possible to determine a series of three-dimensional points in one direction, called the relief profile {p}. Using the relief profile[3], we proceed to determine the line of sight in the given direction.
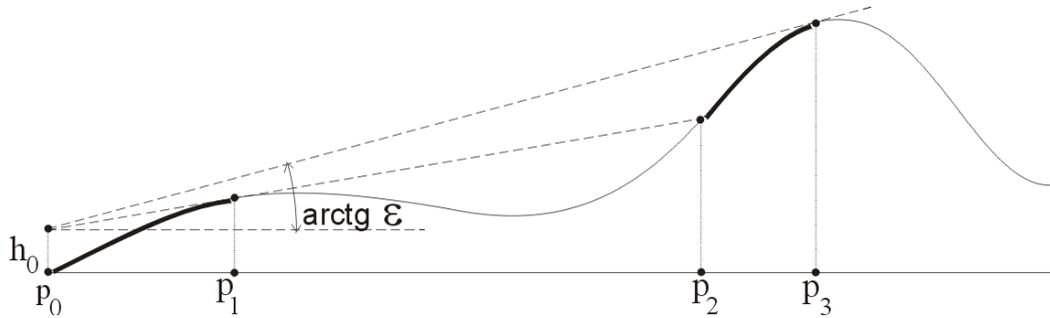
Fig. 2.5. Determining line of sight

Figure 2.5 schematically shows the procedure for determining visibility for targets located on the earth's surface. Let the point p_0 is the position of the observer, at height h_0 above the earth's surface. The observer from a point p_0 there is an open line of sight between the points (p_0, p_1), as well as between points (p_2, p_3). Between the dots (p_1, p_2) has a closed line of sight.

To specifically determine the line of sight, we enter the magnitude of the angle of coverage between the observer and the points of the profile of the earth's surface. Let the closing angle be represented by its tangent and denoted by ε.

Let with p_k a current target point on the earth's surface is indicated.

Control angle of closure ε^k_{contr} relative to a current point p_k of the profile we call the maximum angle of coverage between the observer p_0 and the target p_k. It is expressed as:

$$\varepsilon^k_{contr}= \max (\varepsilon_i), \ 0 < i \leq k. \tag{2.27}$$

The line of sight between the observer p_0 and the target p_k is open when:

$$\varepsilon_k \geq \varepsilon^{\kappa}_{contr} \tag{2.28}$$

The line of sight between the observer p_0 and the target p_k is closed, when:

$$\varepsilon_k < \varepsilon^{\kappa}_{contr} \tag{2.29}$$

By applying the dependencies (2.28), (2.29) for all possible lines of sight in a circular sector with center OP and radius MOR, a visibility matrix denoted by B is determined, where:

$$B \equiv \begin{cases} 1, & \varepsilon_k \geq \varepsilon^k_{contr} \\ 0, & \varepsilon_k < \varepsilon^k_{contr} \end{cases} \tag{2.30}$$

A model for processing information in route determination

The thus defined visibility matrix B and patency matrix A are included in the information processing to obtain a hidden movement matrix.

In fig. 2.6 a) the proposed information processing model is shown graphically. According to the presented mathematical apparatus, matrix A contains the values of the slope of the earth's surface, in decimal degrees, discretized to the cells of the used digital terrain model.

Matrix B is defined by dependence (2.30), represents visibility conditions, and consists of discrete values {0,1}.

The matrix marked in the figure with B^{90} is produced from the visibility matrix, its elements multiplied by a certain numerical value corresponding to a slope unsuitable for the autonomous platform.

The stealth routing matrix is produced by applying elementwise summation of the traversability matrix with the modified visibility matrix.
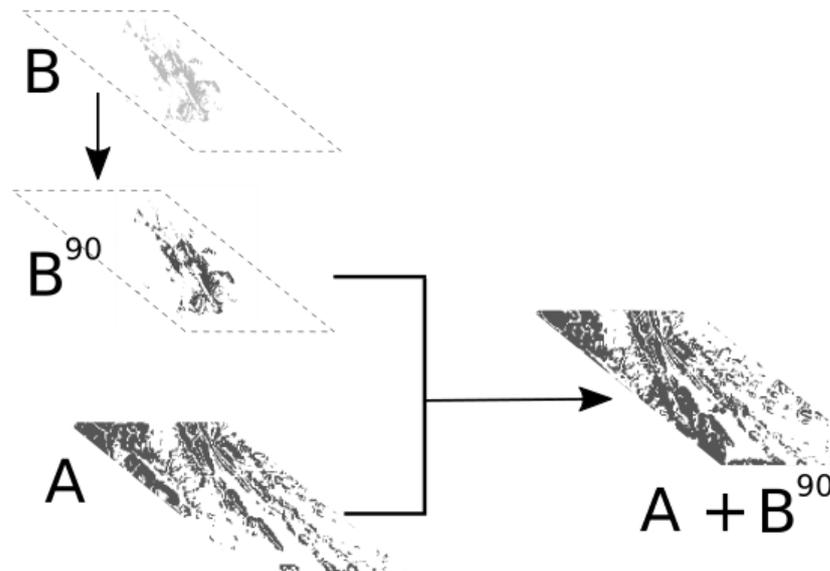
Fig. 2.6 a) Model for information processing during covert movement

In fig. 2.6 b) shows an algorithmic sequence of processing the output data for routing of an autonomous platform with movement hidden in relation to a known observation point - Secure Path Planning. The algorithm sequence was developed and proposed at the scientific conference DIGILIENCE 21 and published in an issue of "Information&Security: An International Journal" [90].

The data required for further processing is contained in the digital terrain model used, shown in the figure as MTS100. The model includes electronic map data in the form of GeoTIFF raster files and DTED (Digital Terrain Elevation Data) data in the form of an ASCII Grid. The initial settings required to achieve a certain solution are provided by a specially developed user interface.

The software solution is a functional extension (plugin) to the specialized product Quantum GIS [87] and the use of programming languages Python and C++. Standard geographic data processing is performed with modules from the GDAL (Geographic Data Abstraction Layer) software library [43], which is a product of the international organization OSGeo (Open Source Geospatial Foundation).[85]

The processing sequence is as follows:

- Selection of an area of interest - AoI determined from the initial data and its presentation in a digital form suitable for processing - a matrix of heights M in GeoTIFF file format. Processing is done by implementing the gdal_merge.py and gdal_warp.py modules from the GDAL library;
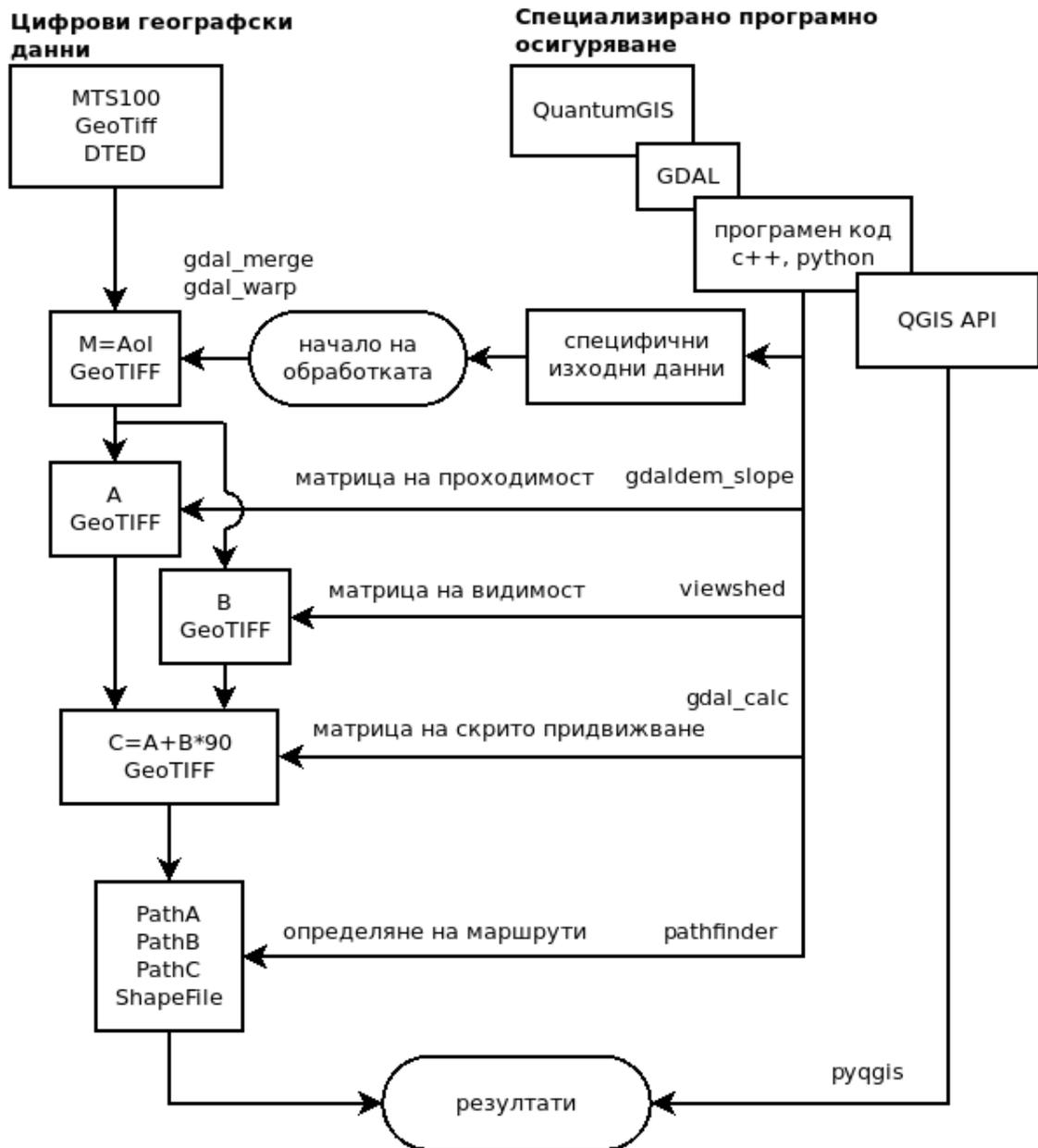
Fig. 2.6 b) Routing algorithm with hidden movement

- Determination of patency matrix A by applying the module gdaldem-slope.py;
- Defining a visibility matrix B by applying the C^{++} code viewshed.c;
- Determination of hidden motion matrix C, for this purpose the mathematical processing of the data from matrix A and matrix B is performed by applying the module gdal_calc.py.
- Execution of the path planning algorithm from pathfinder.py, the routes - results are stored in the form of vector graphics files of type ShapeFile.
- Visualization of the results - vector layers PathA, PathB and PathC op type "line" are the determined routes, taking into account the walkability matrix A, taking into account the visibility matrix B, and taking into account the complex (walkability and visibility) road planning, matrix C.

Fig. 2.10. Graphical complex routing result at a certain slope to overcome and hidden from a known observer

A basic form of path planning algorithm using raster data, known in specialized sources as Deterministic Eight (D8), has been implemented.[34, 110]

The graphical routing results according to the set MaxSlopeAngle value of the autonomous platform, and according to the visibility conditions relative to the determined position of the opposing observer, is the combined option of traversability. This complex variant of routing, avoiding areas visible to the enemy, is shown in fig. 2.10.

Table 2.1 presents the more important numerical results of the software experiments conducted.

Table 2.1. Numerical results of the routing experiment.

Indicators	According to slope	According to visibility	Complex
Processing time	0.068 s	0.042 s	0.107 s
Estimated route length	34439.3 m	38475.7m	38818.3 m

Overcoming mapped obstacles by vegetation type, applying digital vegetation cover data

A geo database, a product of the Military Geographical Service of the Bulgarian Army, is used.
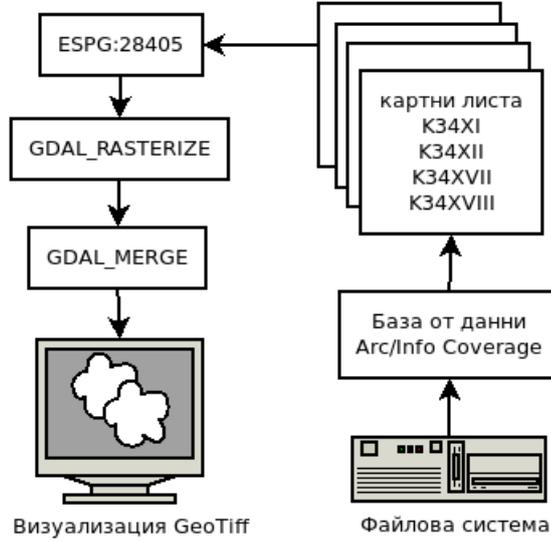
Fig. 2.11. Preparation of data when determining patency

The D8 algorithm [30, 36] is applicable for determining patency in various conditions, including for military purposes. An important feature when applying the D8 algorithm is the need for the input data to be represented in bitmap form.

Overcoming unmapped obstacles, a stereo camera is used and computer vision methods are applied [33, 53].

Procedure for calibrating a stereo camera

Mathematically, the distortion can be represented by the expression:

$$P = K \times [R|T], \tag{2.31}$$

where K denotes the so-called internal matrix (intrinsic matrix) of the camera, [R|T] presents the so-called combined extrinsic matrix and R is responsible for the radial distortion and T is responsible for the tangential distortion.

The internal matrix K consists of camera-specific parameters. It includes information such as focal length (f_x, f_y), optical centers (c_x, c_y), also called the camera matrix. The K matrix depends only on the camera type, so once determined, it can be saved for future calculations. It is expressed as:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.32}$$

The radial distortion R is solved as follows:

$$x^R_{corrected} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6), \tag{2.33}$$

$$y_{corrected}^R = y(1 + k_1r^2 + k_2r^4 + k_3r^6), \tag{2.34}$$

Tangential distortion T occurs because the image capturing lens is not aligned perfectly parallel to the image plane. So some areas of the image may appear closer than expected. It is solved by applying the expressions:

$$x_{corrected}^T = x + \left(2p_1xy + p_2(r^2 + 2x^2)\right), \tag{2.35}$$

$$y_{corrected}^T = y + (p_1(r^2 + 2y^2) + 2p_2xy). \tag{2.36}$$

Therefore, we need to find five parameters (k_1 k_2 p_1 p_2 k_3), known as distortion coefficients, which was performed using procedures from OpenCV. The exact values of the experimentally determined coefficients are presented in the Appendix № 2.1.
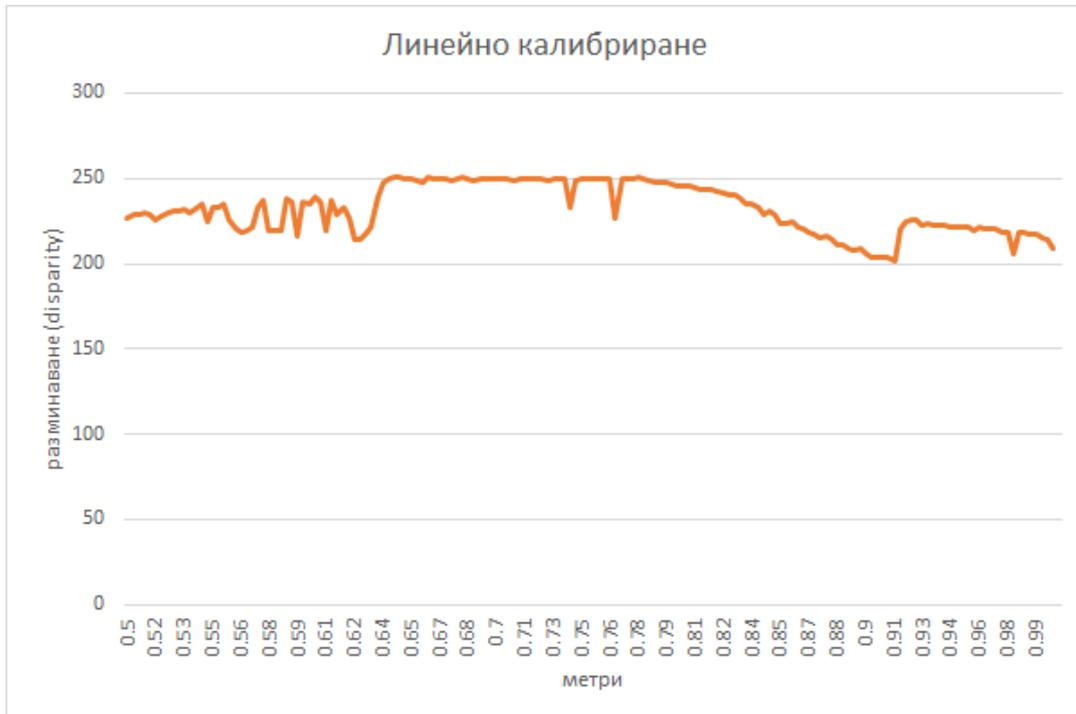
Fig. 2.19. Raw data from the linear calibration

Reference to the mathematical dependencies (2.28 ... 2.39 in the dissertation) gives grounds for determining Disparity values in the synchronous stereo image, and from there determining Depth Map values [25].

In order to achieve a practical result, a linear calibration of the used stereo camera was carried out.

The exact numerical values of the data obtained from the measurement are presented in Appendix No. 2.2. For distances to the barrier between 0.5 m and 1 m, the data are graphically presented in fig. 2.19.

The analysis of the graphically presented experimental data shows that in the range from 0.75 m to 0.91 m the character of the graph qualitatively

corresponds to the theoretically expected dependence according to expressions (2.38, 2.39). A mathematical interpolation method was applied to determine an analytical form of the dependence between the distance to the obstacle and the pixel gap in the image. Applying the research math software GNU Octave [46], the fourth degree interpolation polygon coefficients were determined using the polyfit method. The coefficients of the interpolation polygon are presented in Table. 2.4. Figure 2.20 is a graphical visualization of the interpolation dependence.

Table 2.4. Coefficients of an interpolating polynomial

a_0	a_1	a_2	a_3	a_4
2.9180e-03	-9.6307e-01	1.1888e+02	-6.5077e+03	1.3361e+05
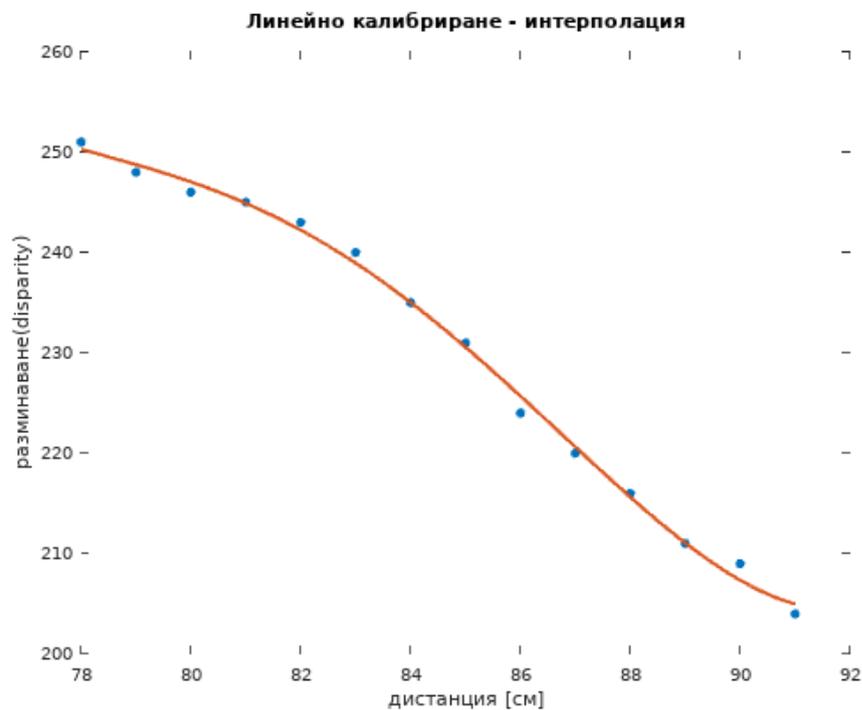
Fig. 2.20. Interpolation dependence

The analytical form of the interpolation dependence is as follows:

$$dsp = a_4 + a_3 y + a_2 y^2 + a_1 y^3 + a_0 y^4, \qquad (2.40)$$

The thus obtained interpolation dependence (2.40) between the distance to the obstacle and the pixel gap in the Disparity image makes it possible to algorithmize the detection of an unmapped obstacle when the autonomous combat platform moves along a certain route. In fig. 2.21 graphically presents an algorithm for detecting unmapped obstacles.
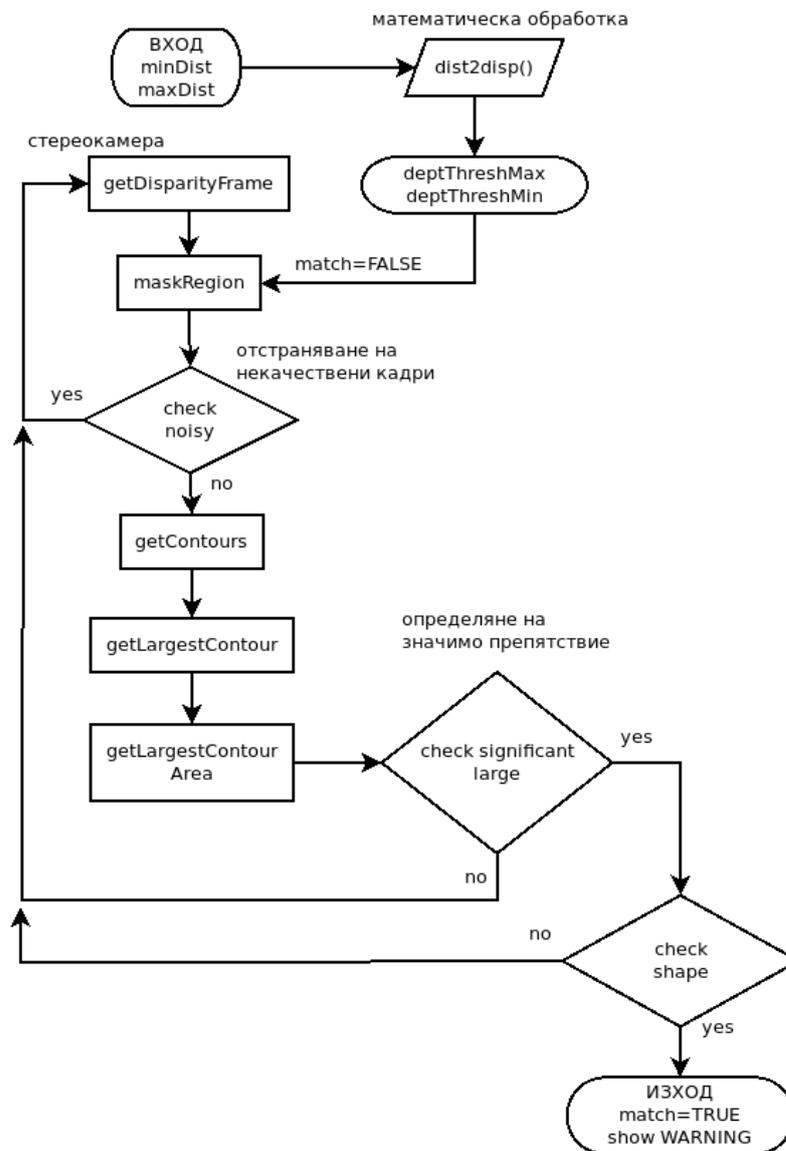
Fig. 2.21. Obstacle detection algorithm

The algorithm is tested using programming code in the Python scripting language, Appendix No. 2.3. In Fig. 2.22 presents the result of the execution of the algorithm described above with selected input data minDist = 0.82 m and maxDist = 0.85 m.

Conclusions and results

A model is proposed for processing the information when planning the path of an autonomous combat platform with a requirement for covert movement in relation to a known enemy observation point and according to the technical possibilities to overcome a certain slope of the slope in the direction of movement.
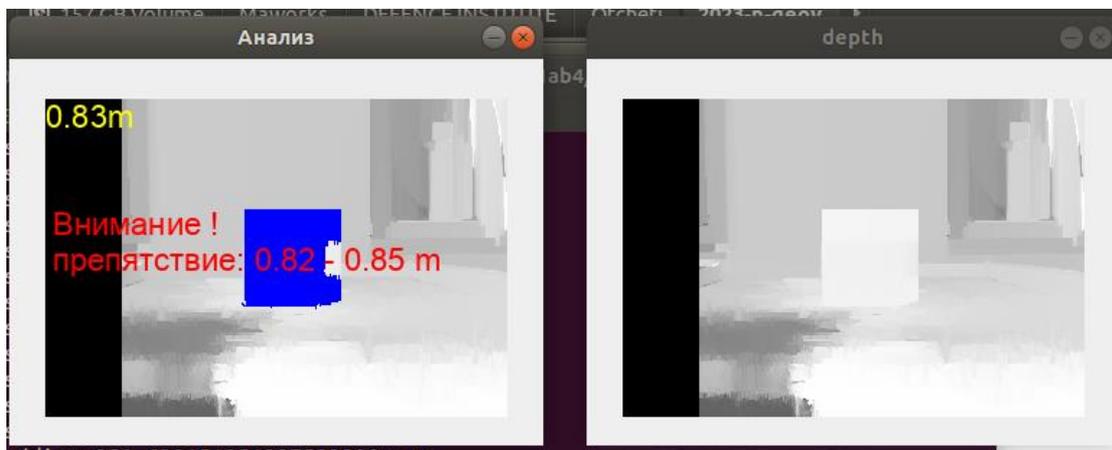
Fig. 2.22. Analysis for an unmapped obstacle

A suitable mathematical apparatus for representing the slope of the earth's surface and for defining visibility zones is defined.

A workstation is configured with QuantumGIS specialized software installed. The proposed algorithms have been tested with a created functional extension (plugin) to QuantumGIS using Python and C^{++} programming languages.

The mathematical foundations for the localization of unmapped nearby obstacles with the application of computer vision methods through the use of a stereo camera are discussed. Known procedures for the preparation of a stereo camera are presented and corresponding experimental work is carried out with the available and specially designed at the Institute of Defense Laboratory Equipment.

CHAPTER 3: Methods and Algorithms for Motion Control of an Autonomous Combat Platform

Mathematical model of the state vector of the autonomous platform

We assume that the autonomous platform follows a predetermined trajectory that is defined in a global planar system (X_e, Y_e). We assume that the autonomous platform moves with a linear velocity $\upsilon = (\upsilon_x, \upsilon_y, 0)$ and rotates with angular velocity $\omega = (0, 0, \omega_z)$, expressed in a local system (X, Y) with center of mass (COM).

The linear velocity υ and the angular velocity ω are defined in the local system (X, Y). The state vector $q = (x, y, \theta)$ is represented by the actual coordinates (x, y) of the autonomous platform in the global system, and θ describes the orientation of the local system relative to the global coordinate system.

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} \int_0^t \dot{x}\,dt \\ \int_0^t \dot{y}\,dt \\ \int_0^t \dot{\theta}\,dt \end{bmatrix} \qquad (3.9)$$

As can be seen from the expressions (3.5 … 3.9), the state vector q of an autonomous platform moving in a plane is completely defined with a known linear velocity υ in the local system and an angular velocity ω of rotation of the local system relative to the global coordinate system.

A closed-loop feedback control model for motion speed
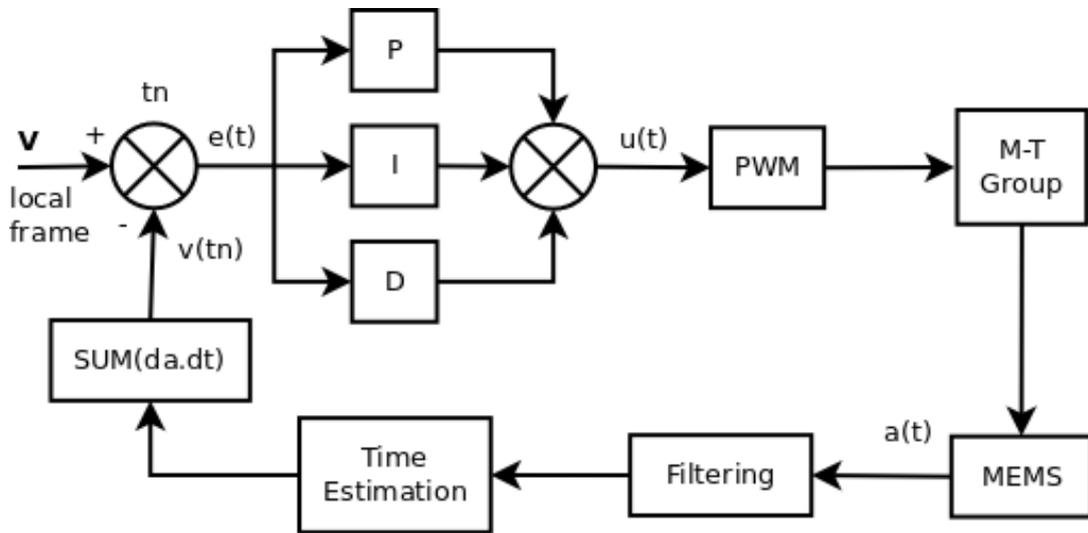
Fig. 3.2. Feedback with PID regulation

For a certain period of time, the on-board equipment of the autonomous platform must provide movement at the selected speed, or close to it. Figure 3.2 schematically presents a speed-adjusting feedback system using the most commonly applied principle of PID (Proportional Integral Derivative) regulation [21, 71, 105]. For example, we consider movement along only one axis.

In the schematic shown, the input control signal is the desired speed V to be maintained by the autonomous platform. The symbol v(tn) denotes the current speed of the autonomous platform. The difference between the desired and current speed (also called error) is denoted by e(t) = V − v(tn). As described in the references cited above in the text, the control mechanism labeled P, I, and D displays the proportional, integral, and differential values for the error compensation. The interpretation of these values is that P expresses current error, I indicates accumulation of past errors, and D is a prediction of future errors. A mathematical expression of the regulator adjustment process can be shown as:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{3.10}$$

where:

u(t) is the value of the control action;

K_p, K_i and K_d are respectively the proportional, integral, and differential component of the control action;

t is the current time.

If we have an array $\{a_n\}$, n = 1,2,...,N from measured acceleration values, the current speed $v(t_n)$ at the instant time t_n can be determined by the expression:

$$\vartheta(t_n) = \vartheta(t_{n-1}) + \frac{a_n - a_{n-1}}{2}(t_n - t_{n-1}), \tag{3.11}$$

with the assumption that $v(t_0) = 0$ and $a_0 = 0$.

As presented in FIG. 3.2, the application of the expression (3.11) is possible to give the required accuracy when the time interval of the control cycle $(t_n - t_{n-1})$ is as small as possible and the acceleration values $\{a_n\}$ are measured with minimal errors.

Prerequisites for improving the speed control process

For the purposes of the present research, highly configurable Inertial Measurement Unit (IMU) components with the possibility of digital output of type MPU9250/6050 were used.

As stated in publication [75], typical measurement errors are possible when using this IMU. These errors can be systematic and random errors. The main types of systematic errors are static bias errors, scale factor errors, and fitting errors. Systematic errors are predictable and in most cases the manufacturer provides hardware preloaded compensation values.

Table 3.1. Digital Low Pass Filter (DLPF)

Mode of DLPF	0(off)	1	2	3	4	5	6	7
Frequency [Hz]	1.13	420	184	92	41	20	10	5
Delay [ms]		0.75	1.94	5.80	7.80	11.80	19.80	35.70	66.96

In IMU measurements, the occurrence of random errors is a priori undeterminable, discussed in [31, 32] and represented as part of the system noise. An unavoidable component of noise, according to the authors, is the natural vibration of the autonomous platform's mechanical structure. These natural vibrations depend on many different factors and are generally difficult to predict.

The implementation of noise reduction techniques is widely used in IMU-based navigation and orientation applications. IMU manufacturers offer as an option a configurable hardware built-in digital low pass filter (Digital

Low Pass Filter - DLPF). The exact values for the DLPF characteristics of the MPU9250 [79] are presented in Table 3.1.

Let the array of possible cutoff frequencies with the IMU's DLPF enabled be denoted as $P = \{P_1 .. P_n\}$, where $n = 1,2,\ldots,7$.

The correct choice of cutoff frequency for low-pass filtering is possible by taking into account the characteristic frequencies of the sensor noise caused by the inherent vibrations during the movement of the autonomous platform. If the array of platform-specific frequencies is specified $H = \{H_1...H_n\}$, where $n = 1,2,...,N$, then the desired value of the shear frequency Oc can be defined as:

$$O_c = \max(P \mid H) . \tag{3.12}$$

To determine the characteristic frequencies H during motion of the autonomous platform, a software implementation of Fourier transform on accelerometer data of an identical second MEMS sensor mounted on the same platform is proposed in this study.

Similar to what is presented in the literature [19, 73, 98], the input signal x(k) (amplitude values at successive moments of time obtained from the sensor) should be represented as H(n) – amplitude values for a discrete sequence of frequencies. The transformation is represented by the mathematical expression:

$$H(n) = \sum_{k=0}^{N-1} x(k) e^{\frac{2\pi jkn}{N}} \tag{3.13}$$

or equivalently applying Euler's formula:

$$H(n) = \sum_{k=0}^{N-1} x(k) \left(\cos\frac{2\pi kn}{N} - j\sin\frac{2\pi kn}{N} \right) \tag{3.14}$$

where H(n) is the searched set of values of characteristic frequencies when moving the autonomous platform, N is the number of determined frequencies and $n = 0,1,\ldots, N-1$.

A software approach with an adaptive digital filter for motion speed control
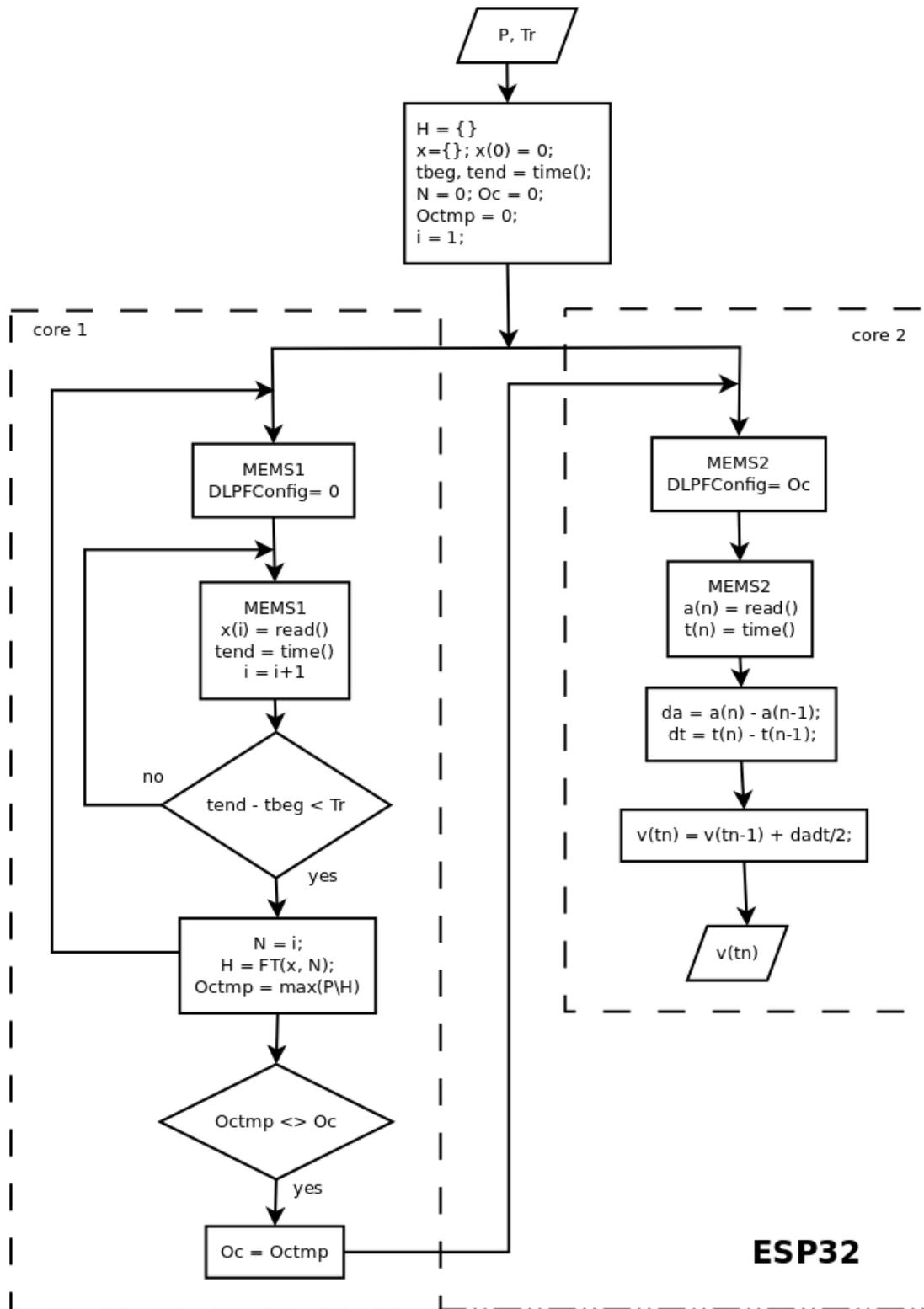
Fig. 3.4. Data flow in adaptive filtering

In fig. 3.4 schematically presents the data flow and digital processing sequence for improved adaptive filtering of the primary data flow from the MEMS sensor accelerometer with a representative MPU9250. The numerical processing sequence was developed and presented at the scientific

conference CompSysTech'20 and published in a publication of the "Association for Computing Machinery"[19]. A dual-core ESP32 microcontroller was used as the main computing module.

Core 1 is designed to extract primary sensor data from the MEMS 1 accelerometer and perform Fourier frequency analysis. Runs with DLPF off with the highest possible data refresh rate. By applying dependence (3.12), the cutting frequency Oc current for the measured time period is determined.

Core 2 provides the calculation procedures in the speed controller subsystem. Applying numerical integration of accelerometric data from MEMS2 determines the velocity $v(t_n)$. For improved adaptive noise reduction, the MEMS 2 sensor uses its built-in DLPF set at the "core 1" determined cutoff frequency O_c.

Experimental determination of operating characteristics

Of scientific and practical interest is the determination of typical disturbing frequencies in the operation of a ground autonomous platform. A built physical model of a ground autonomous platform was used to conduct a practical experiment.

We use the assumption that the main sources of vibration disturbances is the power drive of the model. The power drive of the model is carried out with four motor-transmission groups, each of which has a transmission ratio of 1:8. The three-dimensional model of one of the motor-transmission groups is shown in fig. 3.9.
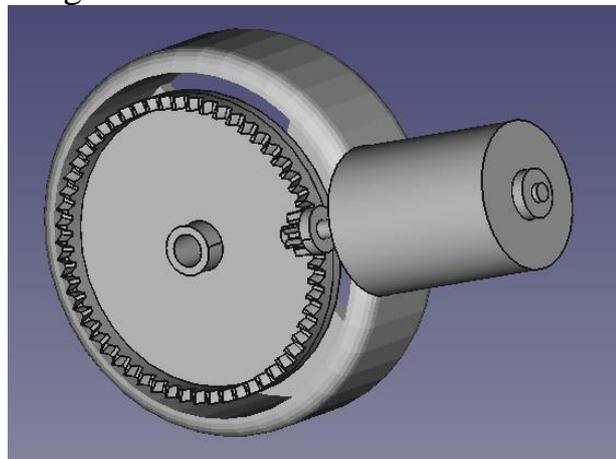
Fig. 3.9. Three-dimensional model of a motor-transmission group

Using a specialized measuring device - an optical tachometer type DT-2234C (shown on Appendix No. 3.2) - the rotational speed of one of the driving wheels was measured when 80% of the maximum power was supplied to the DC electric motor of this wheel. With the received readings from the tachometer in the range of 543 to 599 RPM, we determine an approximate frequency of disturbance from wheel rotation ~ 9.5 Hz.

Taking into account the known gear ratio of the motor-transmission group, we determine the approximate frequency of disturbance from the operation of the DC electric motor of ~ 76 Hz.
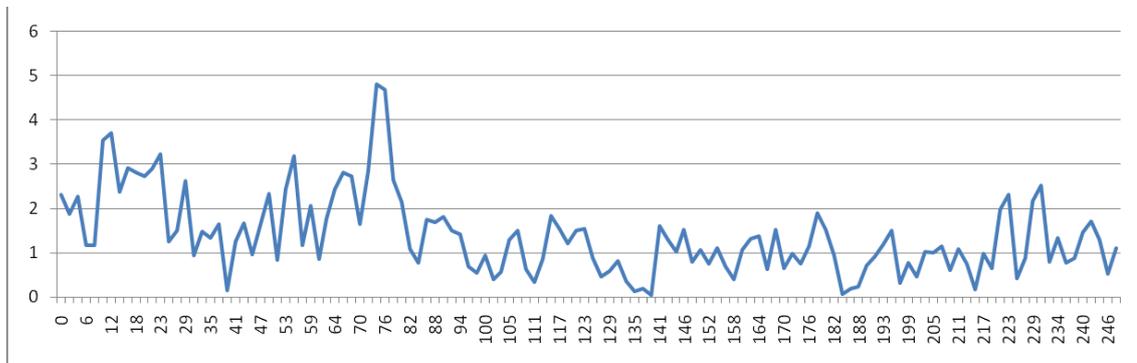
Fig. 3.10. Experimentally obtained frequency diagram in operation of an autonomous platform model

The expected values of vibration disturbance frequencies obtained by direct measurement during movement of the ground autonomous platform were used to test the action of the proposed adaptive digital filter in its part of performing a Fourier frequency analysis. A separate firmware module of the ESP32 on-board microcontroller is implemented by implementing the "arduinoFFT" software library[42] and the Arduino development environment ver. 1.8.15. The source code written in the C^{++} programming language is presented in Appendix No. 3.3. The result of the frequency analysis performed by the on-board microcontroller is shown graphically in fig. 3.10. Confirmation of the expected results are the noticeable peaks in the signal amplitude in the frequency ranges 6 to 12 Hz and 70 to 82 Hz, which corresponds well with the directly measured 9.5 Hz and 76 Hz.

In order to estimate the error in the linear positioning, a setup of the so-called "static" measurement. The autonomous ground platform model is stationary with actuated sensor and microcontroller subsystems. The positioning error is represented as a calculated virtual displacement, which should ideally be zero meters over an arbitrarily long time period. In the conditions of the experiment, the resulting displacement was 279.4 m in a time of 25.8 min (1553 s).

Control of the direction of movement using a magnetometric sensor

To achieve a result sufficient for practical application, modern mathematical methods use three-dimensional data for instantaneous values of acceleration (accelerometer), instantaneous values of the speed of angular rotation (gyroscope) and instantaneous values of the intensity of the earth's magnetic field (magnetometer)..
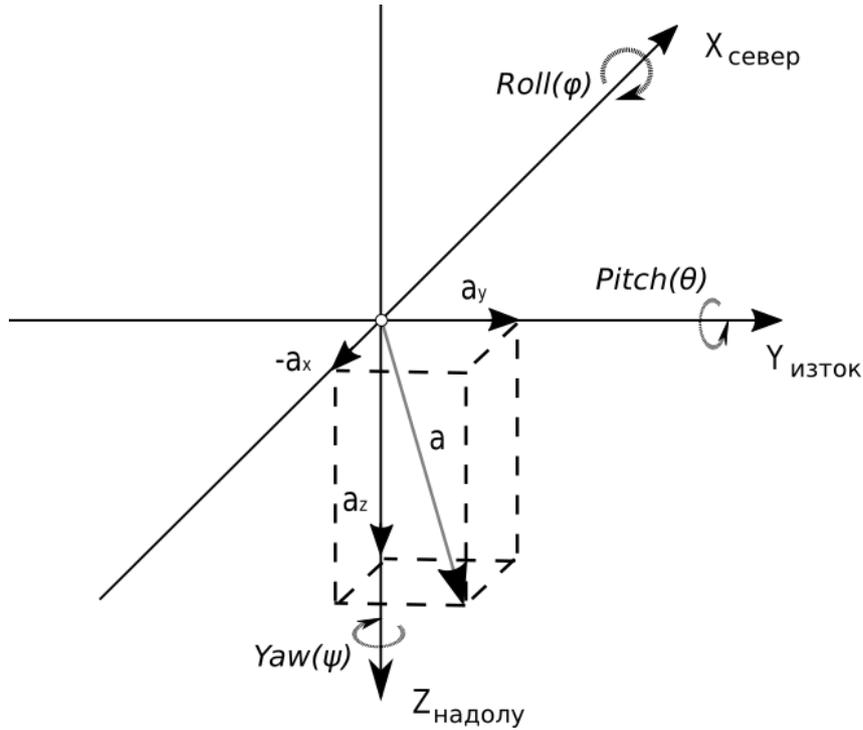
Fig. 3.13. Convention for determining orientation in space

The software implementation includes digital filtering and iterative algorithms applied to the sensor data, as well as a series of geometric transformations. The result of the operation of this computing module is a sufficiently accurate determination of the orientation of the autonomous platform in space - azimuth (Yaw), longitudinal (Pitch) and transverse (Roll) inclination. The YPR implementation convention graphically represented in Fig. 3.13 is the basis of the navigation systems of autonomous devices of different types. [60].

To determine the YPR orientation, dependencies apply:

$$\begin{pmatrix} \theta \\ \varphi \\ \psi \end{pmatrix} = \begin{pmatrix} \tan^{-1}\left(\frac{a_x}{\sqrt{(a_y^2 + a_z^2)}} \right) \\ \tan^{-1}\left(\frac{a_y}{a_z} \right) \\ \tan^{-1}\left(\frac{m_y}{m_x} \right) \end{pmatrix}. \tag{3.18}$$

In the case where the initial resting orientation of the autonomous platform is known θ_0, φ_0, ψ_0, the current values can also be determined by the gyroscope readings:

$$\begin{pmatrix} \theta_{n+1} \\ \varphi_{n+1} \\ \psi_{n+1} \end{pmatrix} = \begin{pmatrix} \theta_n + \omega_x \Delta t \\ \varphi_n + \omega_y \Delta t \\ \psi_n + \omega_z \Delta t \end{pmatrix}, \tag{3.19}$$

where Δt is a sufficiently small time interval, often determined experimentally.

For the most correct determination of the azimuth, it is necessary to take into account the current longitudinal and transverse inclination of the autonomous platform, or to perform the procedure known in the specialized literature as "tilt compensation"[72, 101]. In fig. 3.14 graphically presents the construction for determining the slope compensation.
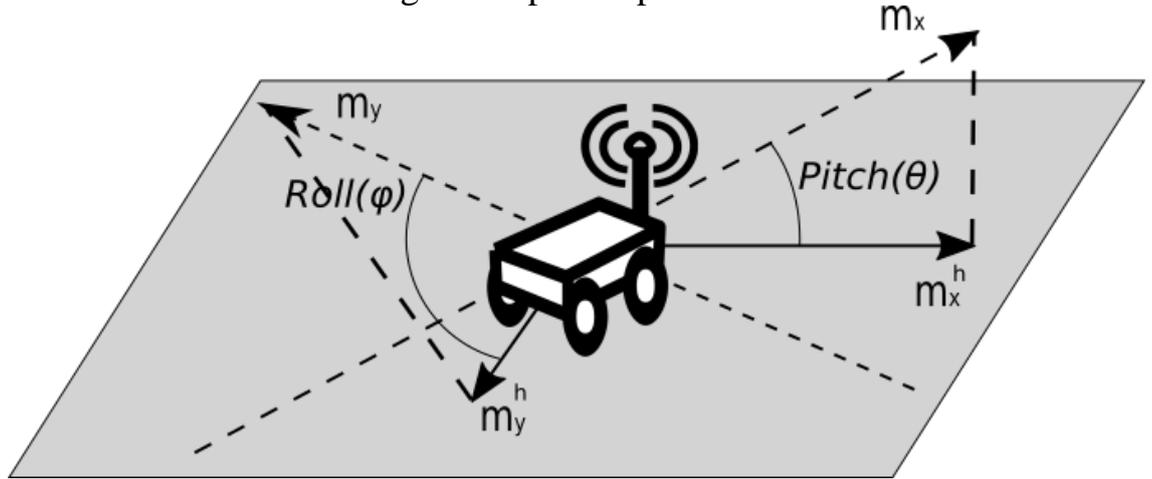
Fig. 3.14. Determine the slope compensation

Then the coefficients m_x and m_y from the expression (3.18) take the form:

$$m_x^h = m_x \cos\theta + m_z \sin\theta, \tag{3.20}$$

$$m_y^h = m_x \sin(\varphi)\sin(\theta) + m_y \cos(\varphi) + m_z \sin(\varphi)\cos(\theta). \tag{3.21}$$

In order for the user of sensor data to be confident in the correct functioning of the IMU, and for the received data to be as close as possible to the real ones, the sensor manufacturer recommends that self-diagnosis procedures and determination of expected deviations in the measured values be carried out before each start-up.

In fig. 3.16 in the form of a diagram, the actions performed when turning on an inertial sensor of the MPU9250 type are shown in their sequence.

MPU9250
Начало

Измервателен цикъл

MPU9250_ADDRESS, WHO_AM_I_MPU9250
0x68 0x75

a, g, m = f(aRes, aBias, gRes, gBias, mRes, mBias, mScale)

Проверка за
наличие на комуникация
по I2C

не ← 0x71 → да → Проверка за
функционалност
(self test)

Калибриране на магнетометър
в условията на работната реда

Работен режим
Обхват на магнитометър
4800µT
резолюция mRes

MPU9250_ADDRESS, ACCEL_CONFIG, 0x00
0x68 0x1C

MPU9250_ADDRESS, GYRO_CONFIG, 0x00
0x68 0x1B

AK8963_ADDRESS, AK8963_CNTL, 0x26
0x0C 0x0A

Обхват на акселерометър
2 g
резолюция aRes

Обхват на жироскоп
250 dps
резолюция gRes

Извлича стойностите на фабрично
определени отклонения
mBias

MPU9250_ADDRESS, ACCEL_XOUT_H
0x68 0x3B

MPU9250_ADDRESS, GYRO_XOUT_H
0x68 0x43

AK8963_ADDRESS, AK8963_ASAX
0x0C 0x10

Чете текущи усреднени стойности
aAvg, gAvg

Сервизен режим

MPU9250_ADDRESS, ACCEL_CONFIG, 0xE0
0x68 0x1C

MPU9250_ADDRESS, GYRO_CONFIG, 0xE0
0x68 0x1B

AK8963_ADDRESS, AK8963_CNTL, 0x0F
0x0C 0x0A
AK8963_ADDRESS, AK8963_CNTL, 0x00
0x0C 0x0A

Задава режим на проверка

Конфигуриране на магнетометър

MPU9250_ADDRESS, ACCEL_XOUT_H
0x68 0x3B

MPU9250_ADDRESS, GYRO_XOUT_H
0x68 0x43

Чете текущи усреднени стойности
aSTAvg, gSTAvg

Определя процентна разлика
aDiff = f(aSTAvg, aAvg,fTrim)
gDiff = f(gSTAvg, gAvg, fTrim)
Diff = max(aDiff, gDiff)

да

Diff < 5%

не

MPU9250_ADDRESS, SELF_TEST_ACCEL
0x68 0x0D...0X0F

MPU9250_ADDRESS, SELF_TEST_GYRO
0x68 0x00...0X03

Извлича стойностите на фабрично определени
допустими отклонения fTrim

Край

Устройството е
неработоспособно

Fig. 3.16. Self-diagnosis and compensation of factory-set deviations in the MPU9250

A method for spatial compensation of deviations in a magnetometric sensor

To determine the azimuth ψ by applying the dependencies (3.18), we consider the readings mx and my radiated during operation of the magnetometric sensor. In the process of calibrating the magnetometer, a data array is stored with the instantaneous values of the magnetic field measured along each of the three {x, y, z} measurement axes. Let's perform a geometric analysis on the accumulated data.

If we consider the data as a pointcloud structure {mX_i, mY_i, mZ_i}, it is possible to determine the corresponding approximating plane when rotating the sensor about from z. Let the approximating plane be represented by the expression:

$$ax + by + d = z, \tag{3.31}$$

where the sought a and b are the angular coefficients along the respective axes, the sought d is the distance to the origin of the coordinate system.

The coefficients a, b, d of the approximating plane are represented as:

$$F = \begin{bmatrix} a \\ b \\ d \end{bmatrix}. \tag{3.34}$$

Then the obvious equality A* F = B holds, or described as a matrix division procedure [63, 80]:

$$F = (A^T * A)^{-1} * A^T * B. \tag{3.35}$$

The values of the required coefficients from the expression (3.31) were determined using software code in the Python programming language, Appendix No. 3.4. The results of the numerical analysis to determine the approximating planes of the magnetometric data when rotating about the z, y and x axes are seen in a table 3.5.

Table 3.5. Coefficients of approximating plane

Axis of rotation	a	b	d
z	-0.001729	-0.042330	52.239663
y	6.648565	-0.611659	202.464604
x	-0.099913	-9.486075	-290.099274

The experimentally obtained numerical results allow a three-dimensional visualization of the downloaded magnetometer data and the corresponding approximating planes. In fig. 3.17 shows the general appearance of the pointcloud structure when rotating around the z axis, in fig. 3.18 and 3.19 the angular deviations in xz and yz views are presented.

After the additional tests thus performed, the readings of the magnetometer compensated for the slope m_x^h and m_y^h from the expressions (3.20, 3.21) take the form:

$$m_x^h = m_x \cos(\theta + \bar{\theta}) + m_z \sin(\theta + \bar{\theta}), \tag{3.36}$$

$$\begin{aligned} m_y^h = m_x \sin(\varphi + \bar{\varphi}) \sin(\theta + \bar{\theta}) + m_y \cos(\varphi + \bar{\varphi}) + \\ m_z \sin(\varphi + \bar{\varphi}) \cos(\theta + \bar{\theta}), \end{aligned} \tag{3.37}$$

where $\bar{\theta} = \tan^{-1}(a)$ and $\bar{\varphi} = \tan^{-1}(b)$,
coefficients a and c are determined by dependence (3.35).
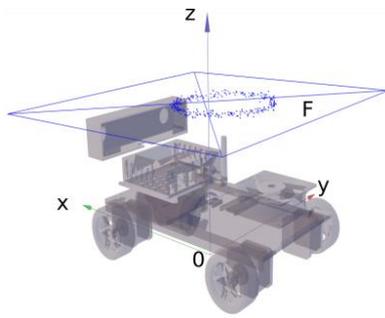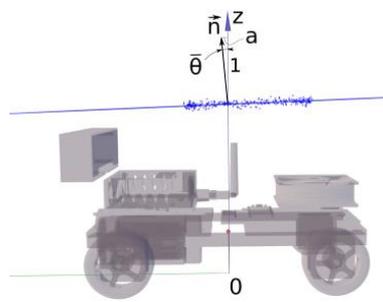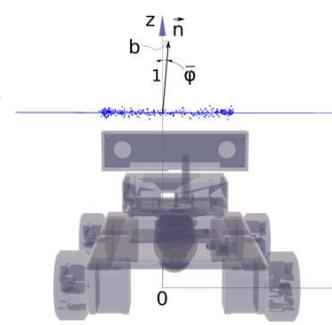
| Fig. 3.17 | Fig. 3.18 | Fig. 3.19 |

Results of the approbation of the spatial compensation method

The given dependencies (3.37) and the obtained numerical data for coefficients of geometric deviations (Table 3.5) provide the basis for conducting an approbation experiment. The purpose of the experiment is to evaluate the application of the proposed spatial compensation method, how it affects the accuracy of determining the azimuth of the experimental autonomous platform.
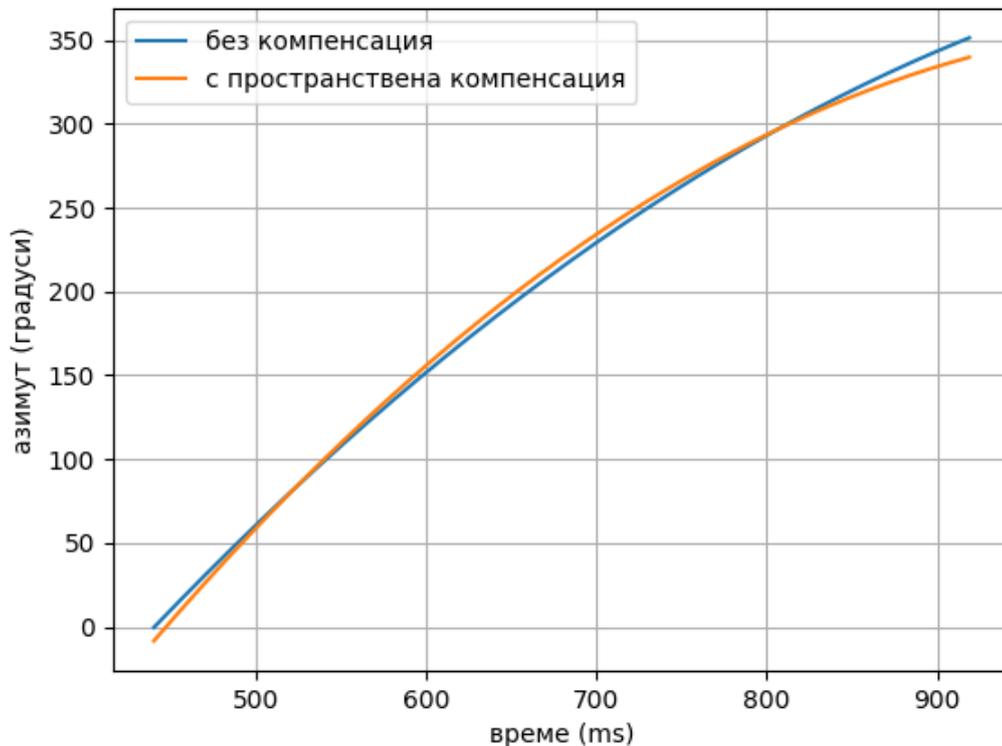
Fig. 3.20. Spatial compensation of deviations in magnetometric sensor readings

In fig. 3.20 with a different color graphically shows how the determined azimuth of the autonomous platform changes when making one complete revolution around its vertical axis. The curve "without compensation" is a visualization of the obtained magnetometric data with the application of dependencies (3.20, 3.21), the curve "with spatial compensation" is obtained with the application of dependencies (3.36, 3.37). A numerical value of the visually distinct differences between the two curves was obtained by the well-known statistical method "root mean square deviation".

Conclusions and results

A study of problematic moments when performing the procedures for controlling the movement of a ground autonomous platform was carried out. The mathematical foundations for determining the components of the state vector are presented. A motion speed control model of the autonomous platform based on an inertial navigation sensor (MEMS) and applying closed-loop feedback is proposed.

The means for filtering the sensor noise - a digital low-pass filter (DLPF) - built into the used inertial navigation unit (IMU) are examined. A software implementation of Fourier transform was applied to determine the characteristic disturbance frequencies during motion of the ground autonomous platform.

An innovative software approach and computational procedure with the application of an adaptive digital filter in the motion control process of an autonomous platform are proposed.

An original method for spatial compensation of deviations in the readings of a magnetometric sensor is proposed. The method is applicable to all construction variants of autonomous platforms, where the magnetometric sensor is physically separated from the sensor with accelerometric and gyroscopic action.

The proposed method was tested and as a result a numerical evaluation was obtained for the improvement in the accuracy of the results when determining the azimuth of the ground autonomous platform.

CHAPTER 4: Analysis of combat capabilities when performing tasks in the area

The research conducted in the second and third chapters allows, within the framework of the generalized structure of the thesis, to propose an architecture for routing and conducting control of the movement of an autonomous combat platform. In graphic form, the proposed architecture is presented in fig. 4.1.

The architecture is in line with modern views on the use of autonomous devices in the preparation and conduct of combat operations. There are three main functional components, through the interaction of which it is possible to perform tasks within the framework of a combat mission. The main functional components are: command post, mission operator, and autonomous combat platform.

In the command post, which can be located at a different level in the hierarchical chain of command and control, based on a received tactical task, combat mission planning is carried out.

The mission operator is a unit in the executive branch of the command and control system. A tactical portable computer is available to the mission operator. Using this computer, military specialist contractors receive the transportation plan to accomplish the combat mission. The transport plan represents the set state at the beginning of mission execution.
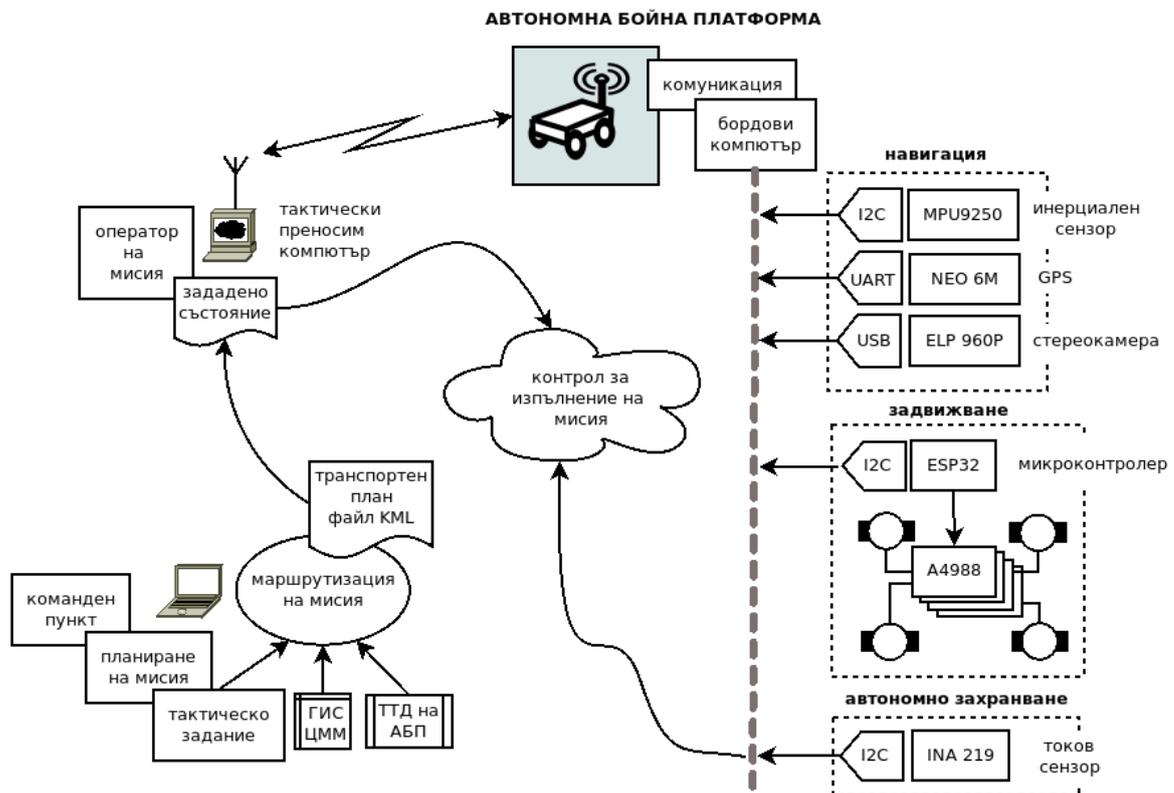
Fig. 4.1. Architecture of a traffic routing and control system

The on-board computer of the autonomous combat platform is of the SBC (Single Board Computer) type and is the integrating unit of all subsystems. The on-board computer runs under the control of a standard Linux-type operating system, with pre-installed system and application software, and configured automated access to the peripheral sensor and executive devices.

Experimental set-up for conducting model studies

In fig. 4.2. schematically shows the architecture for building the on-board system of the autonomous combat platform. The main module in the system is the on-board computer. The on-board computer hardware is of the SBC type and specifically the popular Jetson Nano model was used. The specified hardware is intended more as a specialized video-processing environment, with increased graphics capabilities. The implementation of this hardware in the control modules of autonomous systems is desired by the developers of such devices due to the compact size, low weight, low voltage power supply, and all the necessary digital interfaces to the sensor and executive subsystem.
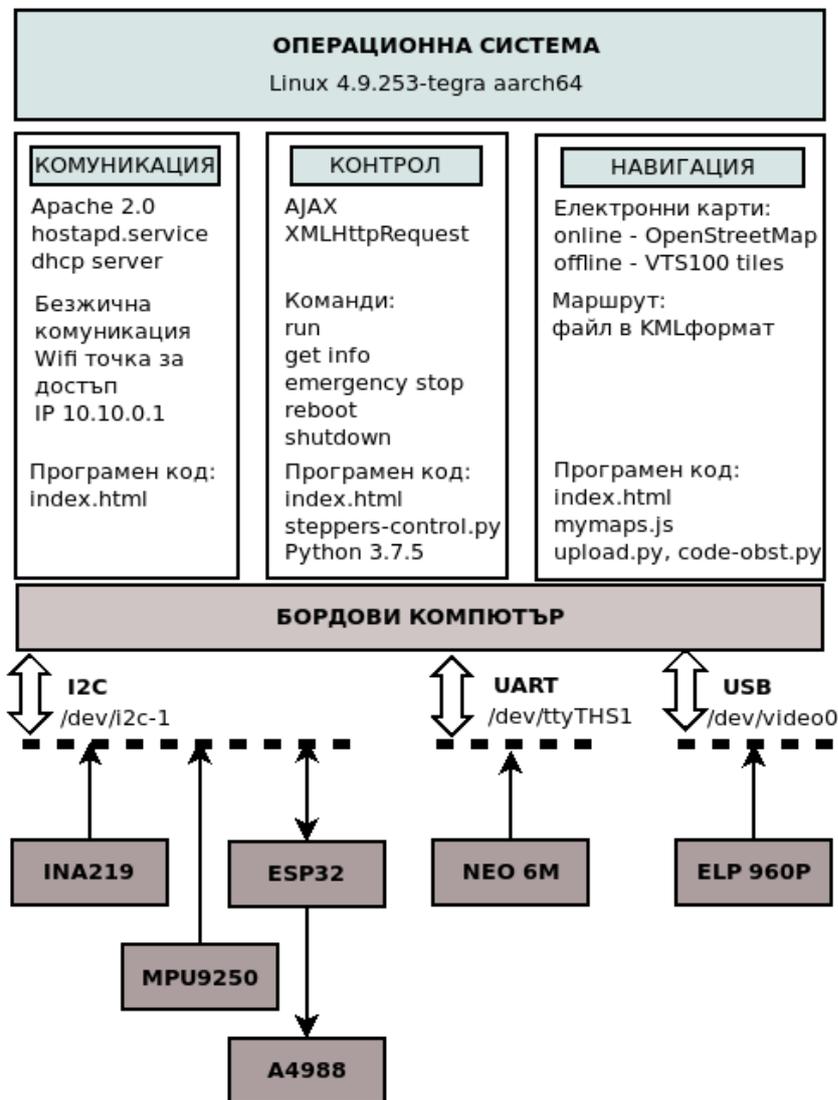
Fig. 4.2. Onboard System Architecture

In fig. 4.3 shows a screen with the interface of the navigation subsystem of the autonomous combat platform. In the process of preparing for a mission, the operator loads a transport plan (route), and for this purpose, with the "Browse" button, he indicates a file in KML format submitted by the command post. The operator selects the visualization mode of the electronic geographic map. From the OSD menu, the choice of the main rendering layer is between Mapnik, CycleMap and LocalTiles visualization. The first two visualizations of electronic geographic maps are from the OSM geo-information database system (OpenStreetMap), a widely used project of the OpenStreetMap Foundation initiative [84]. In case of lack of Internet connectivity, which would be an expected situation in a real production, the basic visualization of an electronic geographic map is in offline mode. Electronic map layers from the VTS100 system of the Military Geographical Service of the Bulgarian Army for the area of movement are converted to a view compatible for visualization in the OSM Tiles format and are physically located in the file system of the on-board computer.
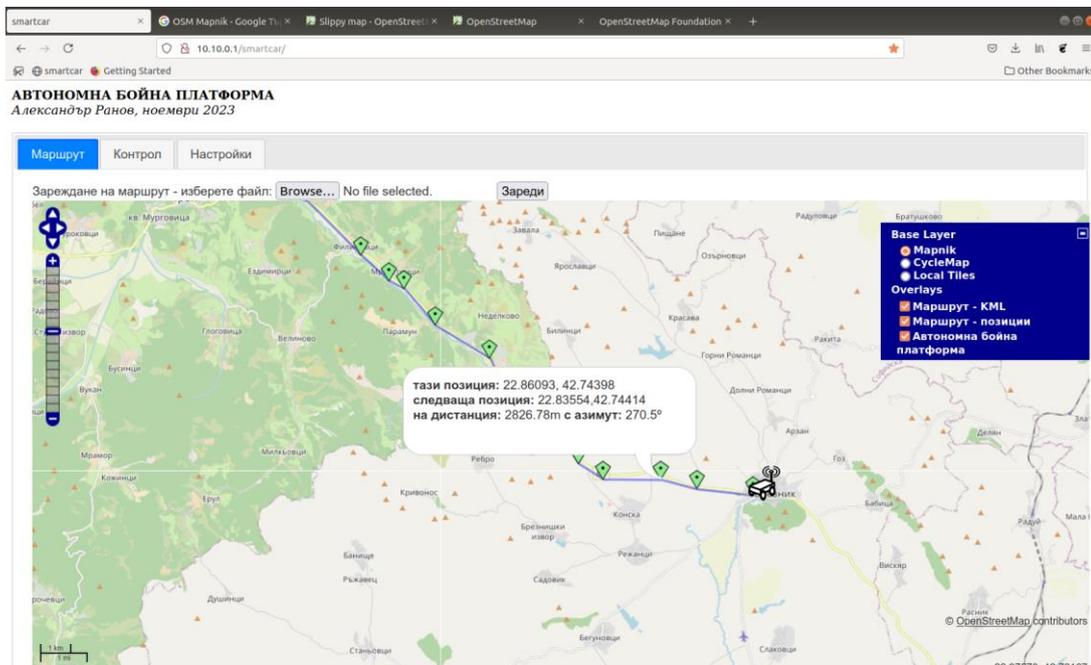
Fig. 4.3. View from the navigation subsystem screen
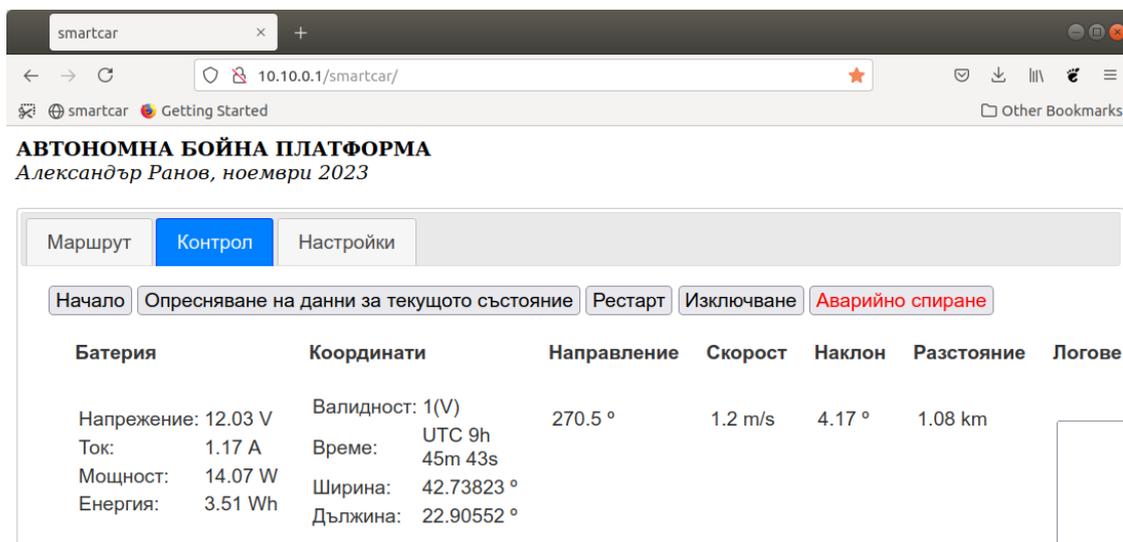
Fig. 4.4. Control interface of the autonomous platform

The procedures for controlling the movement of the autonomous combat platform are carried out by executing commands sent through the web interface (Fig. 4.4), and automatic interaction between the sensor and executive subsystem. The commands from the web page are executed in the server part of the on-board computer system by programmatically generating requests using the AJAX (Asynchronous JavaScript and XML) system, which by applying the javascript functionality XMLHttpRequest makes a parameterized reference to the program code steppers-control.py.

The "Start" button is provided for starting to move along an already set route. To move to the next control point of the route, the ESP32 microcontroller receives formatted data for speed, distance and acceleration at start and stop for the left and right groups of electric motors, respectively. The microcontroller communicates with the on-board computer over the I2C

data channel, using modified firmware that presents it as an I2C slave device[91].

The current status data of the autonomous platform is requested from the on-board computer system with the "Refresh current status data" button, which starts the get info command. By reading the readings of the INA 219 current sensor, the current values of the voltage supplied by the autonomous power supply, the value of the current consumption of electric current, the instantaneous power consumed, and the electricity consumed since the start of movement are visualized. The emergency stop function ("Emergency stop" button) can be activated at the operator's decision, or in the event of a critical voltage drop in the autonomous power supply system.

Improving the ability to perform transport-combat tasks

In the created experimental model of an autonomous combat platform, the on-board single-board computer has an available electronic carrier of the operating system and specialized software, a micro SD (Secure Digital) card with a typical storage volume of 32 GB.

In the event of an emergency power failure, or in combat conditions (radio-electronic countermeasures with EW means by the enemy), the electronic medium - the SD card of the operating system and the installed specialized software - is at risk of damage.

In order to ensure a problem-free conduct of the experimental work on the dissertation research, the fully configured and verified working SD card is reserved.

The backup and restore experiments were performed in a desktop Linux operating system environment. A flowchart of the sequence of actions for creating a backup SD card is presented, with the means of the operating system used.

Energy characteristics

The autonomous combat platform with electric drive is expected to achieve the goal of the transport-combat mission after preliminary preparation. The autonomous platform must travel a certain distance, following a set route, and within the allotted time. In ideal conditions, without the influence of the enemy and accidental technical failures, the execution of the mission is directly dependent on the availability of sufficient energy in the autonomous power supply system.

In the physical model of an autonomous combat platform used for experiments, the power supply is a type of LiIon (Lithium - Ion) battery according to the 3S scheme with a total nominal voltage of 12 V and a declared capacity of 2000 mAh, type 18650.

A load characteristic of a single LiIon cell with the same capacity is published in the source [37]. Based on the published data, and with the application of software code (Appendix No. 4.3), a digitized load characteristic in 3D format shown in fig. 4.9.

The physical model of an electrically powered autonomous combat platform, created for experimental work in the current research on the doctoral program "Automated information processing and control systems", is primarily designed to test the processes of information interaction in routing and conducting traffic control. Without entering into theoretical statements from other scientific specialties, the assessment of energy consumption during the implementation of a transport-combat plan could be based on practical recording of the so-called "external speed characteristic" discussed in the cited material. [13]
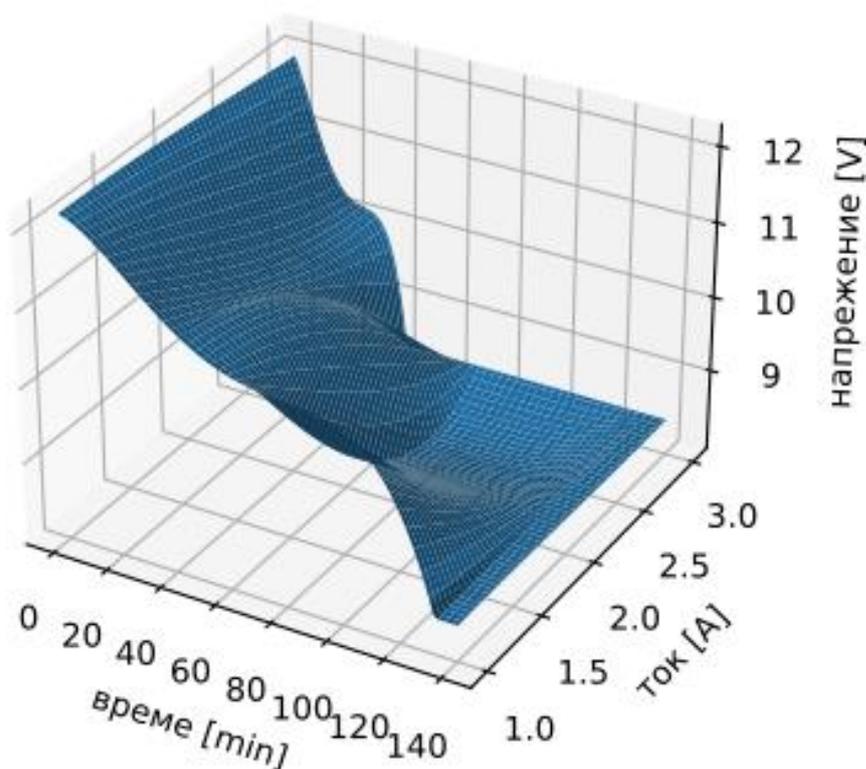
Fig. 4.9. Digitized load characteristic in 3D format

Table 4.2 presents experimentally obtained values for the external speed characteristic, when moving on a different slope and with a different set speed. The data were obtained by direct measurement of electrical quantities from the on-board sensor INA 219, and an additional measuring device. Values are averaged over the period of each measurement.

Table 4.2. Experimentally determined values for external speed characteristic

Slope	Speed[m/s]	U[V]	I[A]	P[W]
	0.5	11.89	2.20	26.16
	1.0	11.85	2.33	27.61
	1.5	11.84	2.41	28.53
	0.5	11.80	2.42	28.56
	1.0	11.70	2.56	29.95
	1.5	11.65	2.62	30.52
	0.5	11.60	2.62	30.39
	1.0	11.44	2.75	31.48
	1.5	11.37	2.81	31.95

a)   *b)*
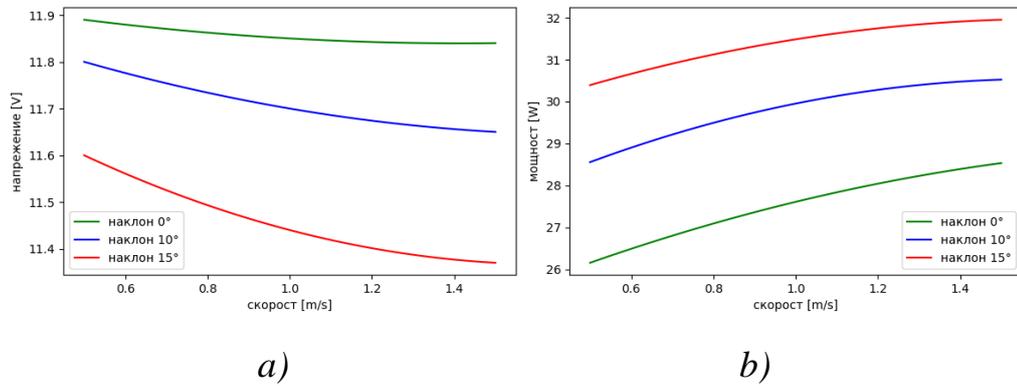
Fig. 4.10. External speed characteristics by voltage and power

Methodology for determining the possibility of executing a given route

The ability to completely overcome the set route, at a generally assigned travel speed, is determined by the current level of the supply voltage. Two main factors for the success of the transport-combat mission can be defined: general energy armament, and momentary overload. For normal operation of the on-board computer, the supply voltage level must not fall below 8.5 V, for all sections of the route. At the same time, as the time of movement increases, in the order of exhaustion of the capacity of the storage battery, the supplied voltage progressively decreases, visible from the graph in fig. 4.9. The experimentally recorded speed characteristic, in the form shown in fig. 4.10 a) graphically shows the relationship between the ability of the battery to maintain voltage under the conditions of different loads caused by movement.

Let U_i denote the voltage that is expected to be present when overcoming the i-th section of the set route.

$$U_i = U_{(v, \gamma_i)} , \qquad\qquad\qquad (4.1)$$

where U is the functional dependence graphically shown in Fig. 4.10 a).

Let U_{min} denote the minimum allowable working voltage, determined with a value of 8.5V.

Then the condition for operability under momentary overload is in the form:

$\min(U_i) > U_{min}$ (4.2)

The total energy armament of the autonomous power system must be sufficient to perform the requested displacement, expressed as mechanical work.

Let the travel route be defined by N number of elementary segments. For each i-th leg, i = 1…N in the routing process are defined:

S_i – travel distance;

γ_i – slope to climb;

$v_i = v$ – set movement speed;

$\Delta\alpha_i$ – change in the azimuth of the direction of travel;

$\omega_i = \omega$ - angular velocity when changing the direction of movement;

r – turning radius when changing the direction of movement.

Then the theoretically required energy A_v is determined by the expression:

$$A_v = \sum_{i=1}^{N} \left(P_{(v,\gamma_i)} \frac{S_i}{cos\gamma_i v} + P_{(v=r\omega,\gamma=0)} \frac{\Delta\alpha_i}{\omega} \right) \quad , \tag{4.3}$$

and more:

$P_{(v,\gamma_i)}$ is the instantaneous power consumption according to the graphical representation in fig. 4.10b).

The given route can be reached if the condition is met:

$$E > A_v , \tag{4.4}$$

where E denotes the claimed battery capacity of the autonomous power supply, measured in watt hours (Wh).

The simultaneous fulfillment of the conditions under dependencies (4.2) and (4.4) ensures the possibility of executing the set route from the autonomous combat platform.

Conclusions and results

An architecture for routing and control of the movement of an autonomous combat platform is proposed, consistent with modern views on the application of autonomous devices in the preparation and conduct of various operations. For the components of the "navigation" and "control" architecture, an experimental set-up for conducting model studies is presented. The created physical model of an autonomous platform was used in the experimental activity.

An analysis of the energy characteristics of the autonomous combat platform was performed. The known load characteristic of the energy source used in the physical model is compared with the experimentally recorded external speed characteristic of the autonomous combat platform in different modes of movement. As a result of the research, a methodology was proposed for determining the possibilities of executing a given route.

III. CONCLUSION

The conducted research on the topic of the dissertation reveals a number of problem areas in the theory and practice of autonomous systems. Attention was paid to the issues of application of autonomous systems in the military field.

Achieved scientific-applied and applied contributions:

1. A mathematical apparatus was defined for deriving the dependences "slope" and "visibility" using a digital model of the height of the earth's surface.

2. On the basis of a mathematical analysis, information structures "passability matrix" and "visibility matrix" were defined. A model for processing information structures with the result "hidden movement matrix" in autonomous combat platform routing is synthesized.

3. By applying the routing model, an algorithm was developed for drawing up a transport plan of a ground autonomous combat platform, without using an existing road network and in the conditions of hidden movement towards a known enemy observation point.

4. The routing model is validated in a developed test software application. Graphical and numerical results are presented, confirming the working ability of the model.

5. An algorithm was developed for the detection of unmapped obstacles using "computer vision" methods. The algorithm was tested in a specially created test environment.

6. A mathematical model of the state vector of an autonomous platform is developed. A closed-loop feedback control model for motion speed is proposed.

7. An analysis of the sources of errors in the processing of primary sensor data from an inertial navigation module was performed. An original approach for implementing digital filtering in speed control of an autonomous combat platform is proposed. The approach has been tested in its part of experimental determination of operating characteristics.

8. A method for compensation of deviations in the readings of a magnetometric sensor designed to determine the azimuth in the direction of movement of an autonomous combat platform is proposed. The method was tested by applying spatial graphic analysis on experimentally obtained data.

9. An experimental set-up was created for conducting model studies, and energy characteristics during movement of a physical model of an autonomous platform were recorded. On the basis of the energy characteristics, a methodology is proposed to determine the possibilities for the implementation of a given route.

LIST OF DISSERTATION-RELATED PUBLICATIONS

[10]. Ranov A., Autonomous Combat Platform and Computer Vision, International Scientific Conference "Modern Research and Technologies for Defense" ARTDEF 2022, pp. II-202 - II-210, ISSN 2815-2581.

[11]. Ranov A., Sensor data processing in an autonomous combat platform, CIO ICT Media vol.7, year XV, ISSN 1312-5605, Sofia 2020, pp. 80-84, 2020.

[19]. Alexander Kolev and Alexander Ranov. 2020. Software Improved State Control of an Autonomous Platform. In Proceedings of the 21st International Conference on Computer Systems and Technologies (CompSysTech '20). Association for Computing Machinery, New York, NY, USA, 101–105. https://doi.org/10.1145/3407982.3408002.

[90]. Ranov, Alexander, and Alexander Kolev. "A Security Approach to a Military Autonomous Platform Path Planning." Information & Security: An International Journal 50, no. 2 (2021): 193-203. https://doi.org/10.11610/isij.5022.