Volume 86, Number 12,
December 2025

ISSN 0005-1179
CODEN: AURCAT

AUTOMATION AND REMOTE CONTROL

Editor-in-Chief
Andrey A. Galyaev

http://ait.mtas.ru

Automation and Remote Control

Vol. 86, No. 12, December 2025

Available via license: CC BY 4.0

Automation and Remote Control

ISSN 0005-1179

Editor-in-Chief
Andrey A. Galyaev

Deputy Editors-in-Chief M.V. Khlebnikov and E.Ya. Rubinovich

Coordinating Editor A.S. Samokhin

Editorial Board

F.T. Aleskerov, A.V. Arutyunov, N.N. Bakhtadze, A.A. Bobtsov, P.Yu. Chebotarev, A.G. Chkhartshvili, L.Yu. Filimonyuk, A.L. Fradkov, O.N. Granichin, M.F. Karavai, E.M. Khorov, M.M. Khrustalev, A.I. Kibzun, S.A. Krasnova, A.P. Krishchenko, A.G. Kushner, N.V. Kuznetsov, A.A. Lazarev, A.I. Lyakhov, A.I. Matasov, S.M. Meerkov (USA), R.V. Mescheryakov, A.I. Mikhal'skii, B.M. Miller, O.V. Morzhin, R.A. Munasypov, A.V. Nazin, A.S. Nemirovskii (USA), D.A. Novikov, A.Ya. Oleinikov, P.V. Pakshin, D.E. Pal'chunov, A.E. Polyakov (France), V.Yu. Protasov, L.B. Rapoport, I.V. Rodionov, N.I. Selvesyuk, P.S. Shcherbakov, A.N. Sobolevski, O.A. Stepanov, A.B. Tsybakov (France), D.V. Vinogradov, V.M. Vishnevskii, K.V. Vorontsov, and L.Yu. Zhilyakova

Staff Editor E.A. Martekhina

SCOPE

Automation and Remote Control is one of the first journals on control theory. The scope of the journal is control theory problems and applications. The journal publishes reviews, original articles, and short communications (deterministic, stochastic, adaptive, and robust formulations) and its applications (computer control, components and instruments, process control, social and economy control, etc.).

Automation and Remote Control is abstracted and/or indexed in *ACM Digital Library, BFI List, CLOCKSS, CNKI, CNPIEC Current Contents/Engineering, Computing and Technology, DBLP, Dimensions, EBSCO Academic Search, EBSCO Advanced Placement Source, EBSCO Applied Science & Technology Source, EBSCO Computer Science Index, EBSCO Computers & Applied Sciences Complete, EBSCO Discovery Service, EBSCO Engineering Source, EBSCO STM Source, EI Compendex, Google Scholar, INSPEC, Japanese Science and Technology Agency (JST), Journal Citation Reports/Science Edition, Mathematical Reviews, Naver, OCLC WorldCat Discovery Service, Portico, ProQuest Advanced Technologies & Aerospace Database, ProQuest-ExLibris Primo, ProQuest-ExLibris Summon, SCImago, SCOPUS, Science Citation Index, Science Citation Index Expanded (Sci-Search), TD Net Discovery Service, UGC-CARE List (India), WTI Frankfurt eG, zbMATH.*

Journal website: http://ait.mtas.ru

© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences.

Automation and Remote Control participates in the Copyright Clearance Center (CCC) Transactional Reporting Service.

Available via license: CC BY 4.0

0005-1179/25. *Automation and Remote Control* (ISSN: 0005-1179 print version, ISSN: 1608-3032 electronic version) is published monthly by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 65 Profsoyuznaya street, Moscow 117997, Russia.

Volume 86 (12 issues) is published in 2025.

Publisher: Trapeznikov Institute of Control Sciences, Russian Academy of Sciences.

65 Profsoyuznaya street, Moscow 117997, Russia; e-mail: redacsia@ipu.rssi.ru; http://ait.mtas.ru, http://ait-arc.ru

Contents

Automation and Remote Control

Vol. 86, No. 12, 2025

Reviews

Linear Systems

Nonlinear Systems

Stochastic Systems

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 12, pp. 1089–1117.
© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2025.
Russian Text © The Author(s), 2025, published in Avtomatika i Telemekhanika, 2025, No. 12, pp. 3–41.

REVIEWS

Precise Industrial Photogrammetry Methods Survey

A. V. Gudym[*,a] **and A. P. Sokolov**[*,b]

[*] *Bauman Moscow State Technical University, Moscow, Russia*
e-mail: [a] *anton.v.gudym@yandex.ru,* [b] *alsokolo@bmstu.ru*

Received January 26, 2025
Revised July 4, 2025
Accepted July 8, 2025

Abstract—The survey is about modern and classical methods that forms SOTA photogrammetric pipeline for effectively constructing high-precision 3D point clouds and determining the position of objects in space from a photo or video signal. The work pays special attention to measurement error factors of the output 3D-reconstruction. Depending on the application, the reconstructed 3D-points may correspond to contrasting features of the object's texture, landscape, or special marks applied to the object's surface. After the features matching, the bundle adjustment optimization follows to restore the 3D coordinates of points in space. The survey provides detailed overview of the algorithms and convenient and practical formulations of various camera models and their distortions for bundle adjustment process. The experimental part demonstrates the highest level of accuracy achievable in practice using the methods considered. For close-range measurements repeatability, the proposed pipeline can outperfom professional photogrammetry solution.

Keywords: precise photogrammetry, visual odometry, computer vision, feature matching, bundle adjustment

DOI: 10.7868/S1608303225120011

1. INTRODUCTION

Photogrammetry—the science of measurements from photographs—has a long history and has been actively developed both in Russia and worldwide since the late 19th century. The term "*photogrammetry*" was introduced in 1867 by the German architect Albrecht Meydenbauer (1834–1921), who had previously published his photogrammetric method for measuring buildings in 1858. The mathematical foundations of photogrammetry were laid by the German mathematician S. Finsterwalder (Sebastian Finsterwalder, 1862–1951) [1]. Significant contributions to the development of projection mathematical models, still in use today, were made by the American researcher Duane Brown [2–4]. The founders of photogrammetry in Russia are considered to be the following outstanding scientists and engineers, specialists from the Department of Photogrammetry at the Moscow State University of Geodesy and Cartography, who made significant contributions to the technology's development: Professors N.M. Alexapolsky (1890–1942), F.V. Drobyshev [5], A.N. Lobanov, L.N. Vasilyev, V.B. Dubinovsky [6], and many others.

One of the first domestic stereophotogrammetric systems was the "Talka" photogrammetric system, developed by Soviet engineer D.V. Tyukavkin in the 1960s [7].

Historically, photogrammetry was applied in the fields of geodesy and cartography (Fig. 1). However, over the past two decades, considering the rapid growth of computational power in computer systems and the capabilities of image processing methods, including those using artificial intelligence (AI), photogrammetry as a measurement technology has become widely adopted in industry (*industrial* close-range photogrammetry). The use of photogrammetry methods primarily enables the recovery or reconstruction of a 3D surface model of an object of interest from a set of

Fig. 1. Left: Polygonal mesh (3D model) of landscape surface obtained through photogrammetry. Right: Original photo (crop) where circles indicate positions of SIFT feature points for matching and triangulation in the photogrammetric optimization problem.

photographic images. Furthermore, the technology has a broad range of applications in modern manufacturing processes [8–11]:

1) Contact technologies for measuring surface shape through probe/stylus positioning (tracking) in coordinate measuring machine (CMM) mode;
2) Non-contact measurement of surface shape during laser 3D scanner tracking;
3) Monitoring the position of large-sized parts in the aerospace industry;
4) Control of precision machining of parts in mechanical engineering;
5) Tracking of robot manipulator links and the attached tool [12–15].

Industrial photogrammetry involves the application of numerous algorithms from the fields of computer vision and image analysis (pattern recognition, 3D modeling). A renowned researcher in the field of feature recognition and matching on photographic images is Yu.V. Vizilter [16].

The subject of this work is the description of methods constituting the modern photogrammetric pipeline used in industrial photogrammetry. The methods considered in this work ensure efficient and high-precision recovery of 3D point coordinates (hereinafter referred to as 3D points), matched to various characteristic features identified on the surface of observed objects from photo or video signals. The constructed 3D point cloud is used for reconstructing the object's 3D surface model.

Known alternative technologies for 3D surface reconstruction include:

1) Projection 3D scanners with structured lighting or laser lines;
2) LIDAR scanners, measuring the time-of-flight delay of a laser beam;
3) Hybrid solutions – the use of structured lighting in combination with photogrammetry.

The advantage of alternative solutions is that they do not require the presence of contrasting textural features on the object's surface. At the same time, in terms of the ratio of measurement error magnitude to object size, photogrammetry may only be inferior to LIDAR systems [9] (given the very high cost of the latter), remaining an effective tool for measurements across a wide range of sizes. For example, the same set of tools can measure objects with overall dimensions from 0.1 to 10 m. The error can be 1:100,000 [17] or even 1:200,000 [8] of the object's overall size, i.e., ten or five micrometers per 1 meter, respectively.

Within a short period, photogrammetry has become a standard tool for efficient (accurate and fast) quality control, for example, in aerospace manufacturing. Photogrammetry is often used in conjunction with projection 3D scanners for device positioning and subsequent point cloud merging into a high-resolution output polygonal mesh.

Fig. 2. Left: Example of robot's tool positioning using images from multiple stationary cameras. Right: Circular black-and-white markers (targets) used as features for 3D reconstruction.

Photogrammetry is a key tool for precise video-based navigation and a source of geodetic measurements for unmanned aerial vehicles (UAVs) [14, 18–20]. An example of 3D landscape surface reconstruction using a photogrammetric pipeline [21] with UAV aerial photography data [22] is shown in Fig. 1. The right side of the image shows the positions of textural SIFT features, which are a set of 2D coordinates and a vector of real numbers (a descriptor) that characterizes the properties of the object's texture in a small neighborhood of the feature.

A known advantage of photogrammetry compared to alternative 3D reconstruction methods is its versatility and scalability – the same algorithms and camera models (Section 6.1) are applicable both for long-range geodetic measurements and in completely different close-range scenarios when the distance between the object and the camera does not exceed a few meters. In Fig. 2, the found 3D coordinates of markers on object surfaces are used to estimate the mutual position of bodies, for example, the position of a robot manipulator's end-effector relative to a reference object, whose position is also reconstructed based on the identified positions of it's markers.

This work pays special attention to the qualitative process of feature extraction using the example of artificial circular markers, as this has a decisive impact on the accuracy of the obtained 3D coordinates. The factors determining this process are also relevant for natural textural features in such well-known algorithms as: SIFT [23], SURF [24], SuperPoint [25], and others.

2. SURVEY STRUCTURE

Any photogrammetric system represents a hardware-software system. An example of the hardware part of a photogrammetric system is presented in Section 4. The software part implements algorithms designed to solve problems of the following two types, corresponding to the stages of the photogrammetric pipeline:

1) Processing of input images (Section 5):
   - Detection of the maximum number of feature points on the surface of the observed object (Step 1);
   - Construction of a descriptor invariant to affine transformations of the marker and optical distortions of the signal (Step 2);
   - Matching of descriptors – correspondences search (Step 3);
   - Filtering of the found matches (Step 4);

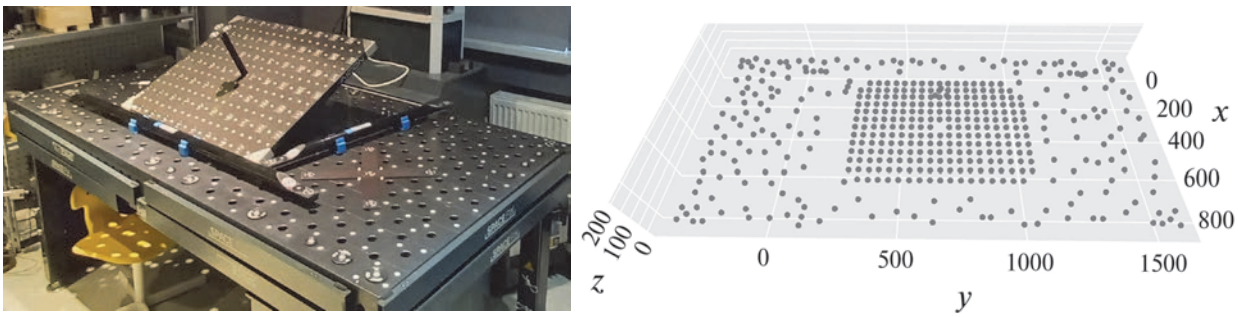

Fig. 3. Left: Test scene includes two measured objects with applied markers: steel welding table ∼2000 mm, carbon fiber calibration plate ∼800 mm, and two carbon fiber scale bars. Right: Photogrammetry result – high-precision reconstruction of markers as a 3D point cloud.

2) Solving the *bundle adjustment problem* (Section 6) for projection rays based on found correspondences:
   - Determination of initial approximations for parameters (coordinates of observed features, internal camera parameters and their positions) or simply auto-calibration (Step 5);
   - Solving the optimization problem (Step 6).

The data-processing step (Section 5) heavily depends on the application scenario. For example, in industrial measurements, simple circular or coded markers are applied to the object (Fig. 3). In aerial or satellite photography, natural contrasting features on the terrain are detected.

The optimization stage, unlike data processing, is sufficiently universal and applicable to almost any operational scenario.

Section 7 is devoted to the results of experiments with various factors of the bundle adjustment problem. It also demonstrates a high level of accuracy, comparable to professional photogrammetric products.

3. MATHEMATICAL NOTATION

Vector quantities are denoted in bold and represent column vectors of scalar quantities, e.g., $\mathbf{p} = [p_1, .., p_n]^{\mathrm{T}} \in \mathbb{R}^n$. Homogeneous coordinates, used in projective geometry and being an extension of Cartesian coordinates, are denoted by the superscript h, e.g.: $\mathbf{p}^h = [p_1^h, \ldots, p_n^h, 1/\lambda]^{\mathrm{T}} \in \mathbb{R}^{n+1}, \lambda \neq 0$. For homogeneous coordinates, the following holds: $\lambda p_i^h = p_i$, where $i \in [1..n]$, which is convenient for concise formulation of various matrix transformations.

Linear operators for coordinate system (CS) transformations typically contain notation indicating from where (subscript **bottom right**) and to where (superscript **top left**) the transition occurs. For example, to describe the position of the object CS (subscript o, $o$bject) relative to the camera CS (superscript c, $c$amera), the matrix ${}^cT_o \in \mathbf{SE3} \subset \mathbb{R}^{4\times4}, \det({}^cT_o) \neq 0$ (six degrees of freedom or *6DoF*) is used:

$$ {}^cT_o \cdot \mathbf{p}_o^h = \begin{bmatrix} {}^cR_o & {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix} = \mathbf{p}_c^h, \tag{1} $$

where:
${}^c\mathbf{t}_o$ is the translation of the object's CS origin relative to the camera (3DoF),
$\mathbf{p}_o^h$ is the point position relative to the object,

$\mathbf{p}_c^h$ is the point position relative to the camera,
${}^cR_o \in \mathbf{SO3} \subset \mathbb{R}^{3\times3}$ is the rotation matrix (three rotation parameters are mapped to the matrix, e.g., by the Rodrigues formula [26] or using Euler angles),
SO3 is the special orthogonal group of rotations (${}^cR_o^{\mathrm{T}} = {}^cR_o^{-1}$, $\det({}^cR_o) = 1$, 3DoF),
SE3 is the Euclidean group of motion [27] or similarity transformation with unit scale, describing possible body movements in space.

The significance of the mathematical "*group*" concept for engineering applications: if it is required to calculate the change or increment of the camera position in the time interval from t_a to t_b, then multiplication by the inverse element[1] should be used:

$$ {}^bT_o \cdot {}^aT_o^{-1} \cdot \mathbf{p}_a^h = {}^bT_o \cdot {}^oT_a \cdot \mathbf{p}_a^h = {}^bT_a \cdot \mathbf{p}_a^h = \mathbf{p}_b^h, $$

where:
${}^aT_o$, ${}^bT_o$ are the positions of the object relative to the camera CS at time t_a and t_b, respectively,
${}^bT_a$ is the position of the camera at time t_a relative to the camera CS at time t_b,
$\mathbf{p}_a^h$, $\mathbf{p}_b^h$ are the positions of the 3D point relative to the camera at time t_a and t_b, respectively.

For better understanding of the formulations, one should assume the simultaneous existence of all CSs associated with the states of the moving body and consider time moments as identifiers of a particular CS. It also means that many definitions below are universal in the following – the indices a, b may correspond to either two moments in time or two different cameras at the same moment in time. $I \in \mathbf{SE3}$ is the identity matrix, describing still object relatively to the camera.

Careful notation for CS relationships in matrix indices is necessary for clarity in formulating the photogrammetric optimization problem, in particular, for describing the camera model (Subsection 6.1).

4. HARDWARE FOR INDUSTRIAL PHOTOGRAMMETRY

Figure 3 shows an experimental scene for a 3D point cloud reconstruction from circular markers (commonly called "*targets*") attached to measured objects. The scene also contains two scale bars – objects with calibrated distance between markers.[2] Scale bars allow determining absolute distance values between points identified in the scene. In the absence of scale bars, it is possible to reconstruct scene or object geometry up to scale from a set of images [26]. Scale bars are often made of carbon fiber, which provides a low coefficient of linear thermal expansion along its axis, as well as low weight and sufficient strength.

For best results, minimizing measurement noise and maximizing operational range, industrial systems often use special circular markers made of retroreflective material, e.g. "*retro-targets*". This property is extremely useful for increasing the contrast or sharpness of the marker's contour. When using a camera flash (to illuminate the object), the retroreflective material returns much more energy strictly in the direction of the source, compared to the inverse-square law of light intensity for ordinary materials.

To obtain high-quality images of key points on the surface of the measured object, a professional DSLR camera can be used. For best results, a monochrome high-contrast sensor and high-resolution optics are required. For example, the professional photogrammetric system "Hexagon DPA Pro"

[1] Matrix elements of **SE3** should not be multiplied by a scalar or subtracted – this is an invalid operation for a group element.
[2] The geometry of markers on the scale bar is measured or calibrated in advance under laboratory conditions.

from Hexagon AB includes a Canon EOS 5DS camera with a monochrome sensor $w \times h = 8700 \times 5800$ (50 Mp); lens f = 28 mm.[3]

To achieve good measurement results, specialized optics (e.g., with low distortion) are not mandatory [3]. The main factors are, naturally, those affecting signal quality (image contrast), as well as geometric stability (rigidity) of the camera-sensor optical system[4] during data acquisition [17, 28]. Geometric stability is influenced by the rigidity of the construction and lens mounting methods, mass-dimensional characteristics, flash mounting method, and the device's operating temperature regime. Thus, for high-precision measurements, careful selection of the equipment is required. Typical industrial photogrammetric system includes:

1) Calibrated carbon fiber scale bar;
2) High-resolution DSLR camera ensuring lens geometric stability;
3) Retroreflective adhesive-based markers;
4) Computational unit or PC implementing the photogrammetric pipeline stages described in the following sections.

5. FEATURE DETECTION AND MATCHING

Feature detection and matching is an extremely broad field of research [21, 29–33] that, among others, has numerous applications in photogrammetry, SfM (Structure from Motion [26]), SLAM[5], augmented reality, image retrieval, and contextual information analysis. This work focuses on the potential applications of these techniques for precise measurements.

The goal of this stage is to identify the maximum number of connections or correspondences between 3D points on the object (future measurements) and their observations – 2D points on images. Many works [25, 32, 34] allow finding correspondences between only image pairs, thus requiring additional grouping of points by their relation to a common surface point. In photogrammetry, it is the correspondence between a 2D point on an image (or projection) $(u, v) \in \mathbb{R}^2$ and a 3D point on the object $(x, y, z) \in \mathbb{R}^3$ that defines the future system of equations in the bundle adjustment problem (Section 6.5). In Fig. 4 correspondences are found by matching binary coded markers (color identify group or unique object point).

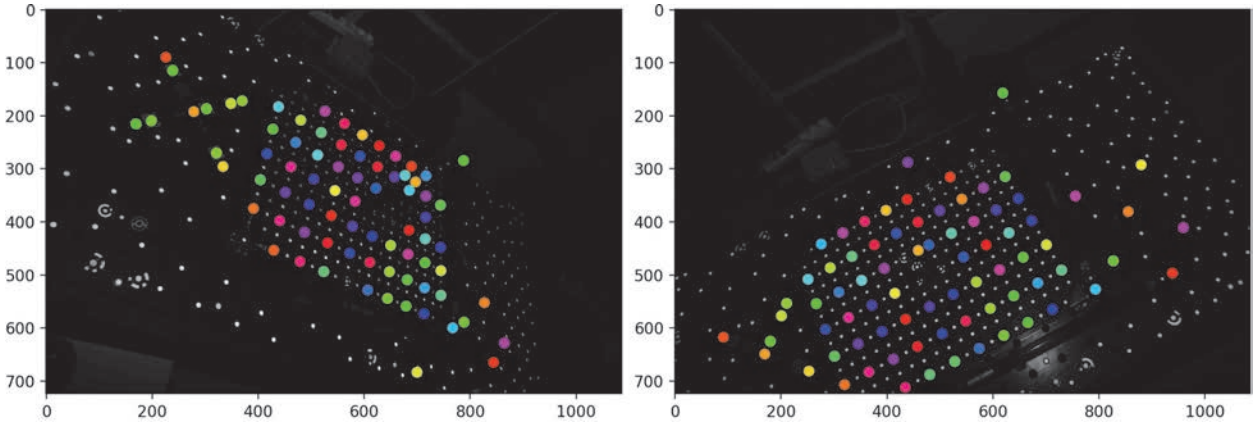

Fig. 4. Original photos of the test scene with 2D points matching results by binary descriptor. Color denotes the group or unique 3D point on the object.

[3] Field of view angles are $\Delta\varphi_{hor} = 64°$, $\Delta\varphi_{ver} = 45°$, $\Delta\varphi_{diag} = 73°$; maximum angular resolution $\approx 2\frac{\Delta\varphi_{diag}}{\sqrt{w^2+h^2}} = 0.014°$. Angular resolution is convenient for comparing sensors of different resolutions or technologies, e.g., SfM and LIDAR.

[4] Methods of dynamic sensor stabilization may be harmful in this context.

[5] SLAM (Simultaneous Localization and Mapping) – a navigation method in mobile autonomous systems.

5.1. Feature Detector

The first step of the photogrammetric pipeline (page 1091) is the extraction or detection of the maximum number N^a_{pts} of feature points $\mathcal{P}_a = \{\mathbf{p}^k_a\}_{k=1}^{N^a_{pts}}$, $\mathbf{p}^k_a = [u^k_a, v^k_a]^{\mathrm{T}} \in \mathbb{R}^2$ on the surface of the measured object, observed in image $a \in [1 \dots N_{im}]$ from a camera.[6] This step is performed by a detection algorithm or simply a detector. In general-purpose photogrammetry [21], without artificial markers, various natural features—points, circles, corners, crosses, or similar structures—are detected on the object texture using Harris [35], GFTT [36], FAST [37] detectors. Some detectors extract both the point and generalized information about the texture of the neighborhood in the form of a multidimensional descriptor vector. The point coordinates together with the descriptor are often called a *feature*. This task is solved by classical approaches SIFT [23], SURF [24] and based on machine learning (ML) SuperPoint [25], DISK [38]. In this case, special areas most suitable for subsequent determination of a stable descriptor are identified. Typically, natural features lack sufficient contrast or size for precise 2D localization and subsequent 3D reconstruction. This can be seen from the difference between the observed 2D coordinates of features and the projections (on the image) of the corresponding 3D object points. This difference is commonly called the reprojection error [26, 39]. For a high-precision measurement task, one standard deviation of the reprojection error does not exceed 0.1 pixels (Fig. 11), hence the feature must be localized even more accurately. The noise of detected coordinates for natural texture features often exceeds one pixel [25, 32, 40]. With careful calibration in [41], but using corners, the reprojection error is still ~ 0.33 pixels. Using ML for texture feature detection in [42] one observes ~ 0.5 pixels. Therefore, for industrial photogrammetric measurements, artificial black-and-white markers of circular and, less frequently, square shape are often applied to the measured object's surface (for further detection).

A circular marker is preferable to corners and similar structures. To understand the reason, let's consider the main factors affecting localization stability, dividing them into two categories for clarity:

1) **Geometric** factors determine observed feature's shape (effects of perspective projection, lens distortion, or object's shape), Fig. 5;
2) **Optical** factors affect feature's contrast, signal-to-noise ratio[7] (illumination level and surface reflectance, focusing, sensor resolution, photon leakage), Fig. 6.

Geometric factors of detection. All texture feature detectors mentioned earlier and similar ones rely on surface's continuity in the neighborhood of the point of interest. That is, the smoother, closer to planar the surface, the more stable and reproducible on other images the found feature will be. For example, the SIFT detector "suppresses" points on lines [23] in order to avoid object boundaries and possible inclusion of background information.

The assumption that the neighborhood of a feature point is a small flat surface area in 3D means that the mapping "3D surface – 2D image"[8] can be performed via an affine transformation of the observed area:

$$ {}^{o}A_a = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix}, \quad \det({}^{o}A_a) \neq 0. $$

It includes six degrees of freedom [26]: 2D translation, 2D scale, 1D diagonal shear, and 1D rotation. In a more general case (without the assumption of a small area size), a projective or homographic

[6] Several N_{im} stationary cameras or one mobile camera in N_{im} positions, depending on the application.
[7] Signal usually refers to pixel intensity or color.
[8] Similar to ray tracing in computer graphics – finding the intersection of a ray "from a pixel" and a 3D plane.

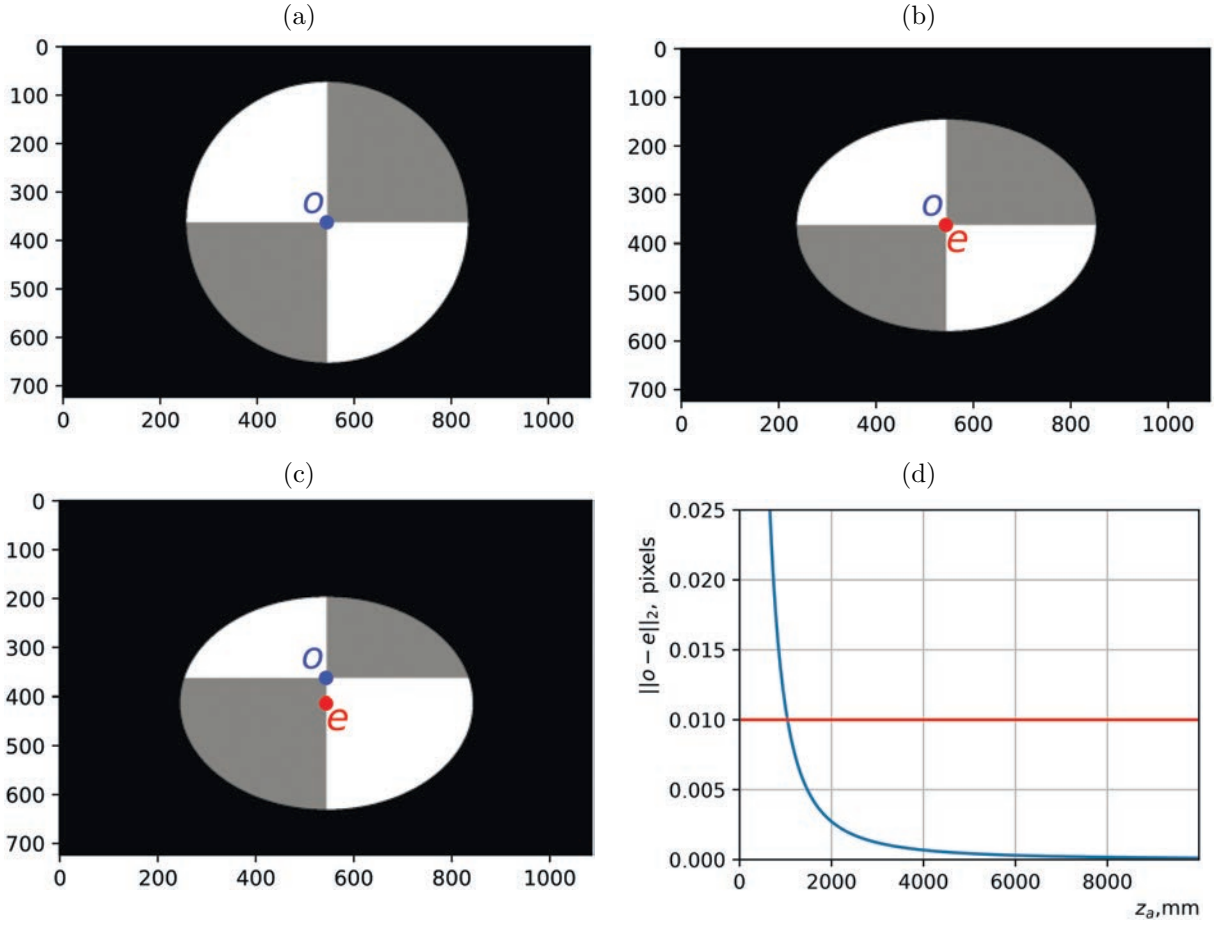

Fig. 5. (a) Circular marker on object, (b) affine projection of marker on screen, (c) perspective projection of marker, (d) dependence of perspective effect $||\mathbf{o} - \mathbf{e}||_2$ on distance to marker z_a.

transformation takes place:

$$ {}^{o}H_a = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}, \quad \det({}^{o}H_a) \neq 0. $$

In computer graphics, affine transformation ${}^{o}A_a$ corresponds to orthogonal projection, and ${}^{o}H_a$ to perspective projection. Transformation ${}^{o}A_a$ transfers a feature from a flat object directly to image a, preserving line parallelism and distance proportions along a line, unlike the more general transformation ${}^{o}H_a$. The model of these transformations in homogeneous coordinates is:

$$ \lambda \begin{bmatrix} u_a \\ v_a \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ 1 \end{bmatrix}, \tag{2} $$

$a, b \in [1..N_{im}]$ – image indices, u_a, v_a – point coordinates on the image (the feature itself or a point in its 3D neighborhood, point index k omitted for brevity), x_o, y_o – point coordinates on the object in the coordinate system of the feature's 3D plane ($z_o = 0$), λ – non-zero scale factor, easily eliminated from the system of linear equations (2) by substitution if ${}^{o}H_a$ needs to be found.

Consider an example. Suppose a circular marker with radius $\mathcal{R} = 5$ mm is depicted on a flat object (Fig. 5a). Under affine projection of the marker onto the screen (with observation angle

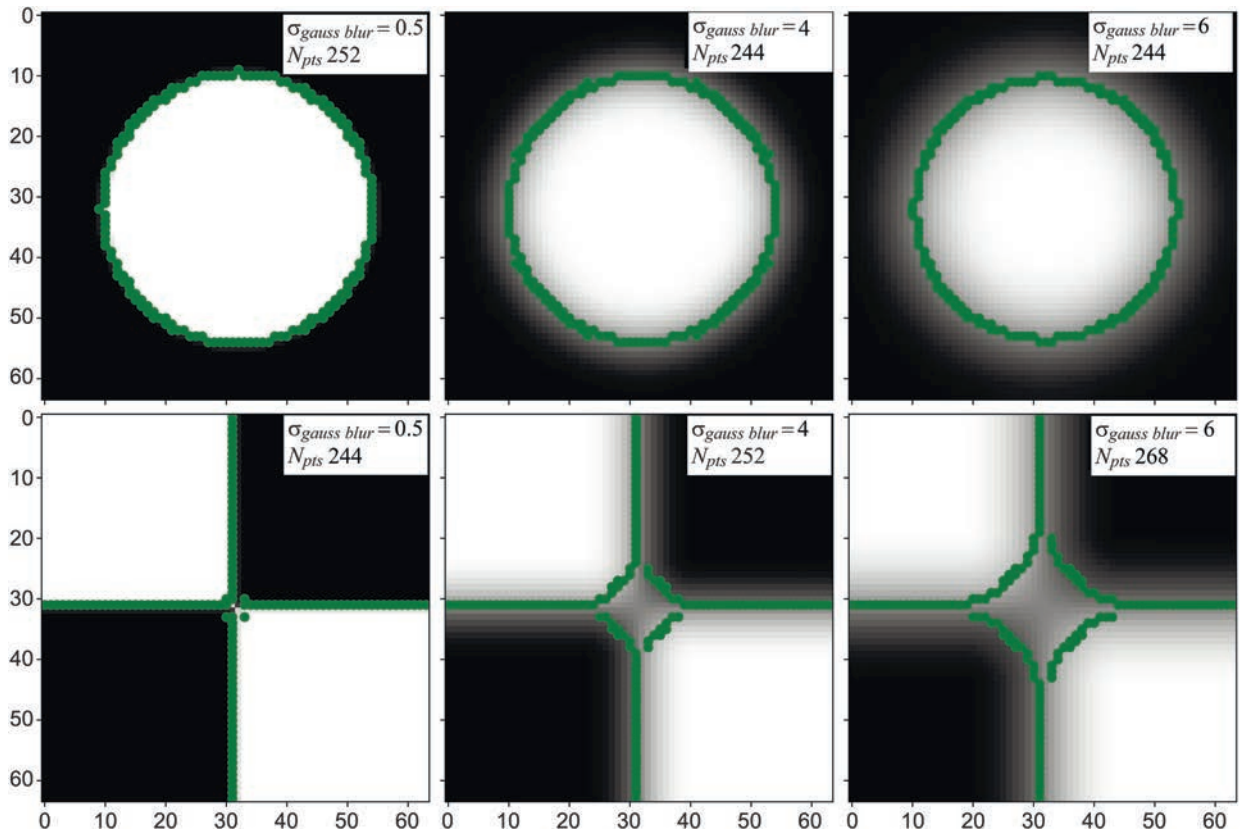

Fig. 6. Effect of sharpness reduction (left to right) for circle and square: unlike circular marker (top row), square structure (bottom row) is significantly distorted. Dots indicate feature contour – pixels with maximum signal gradient amplitude.

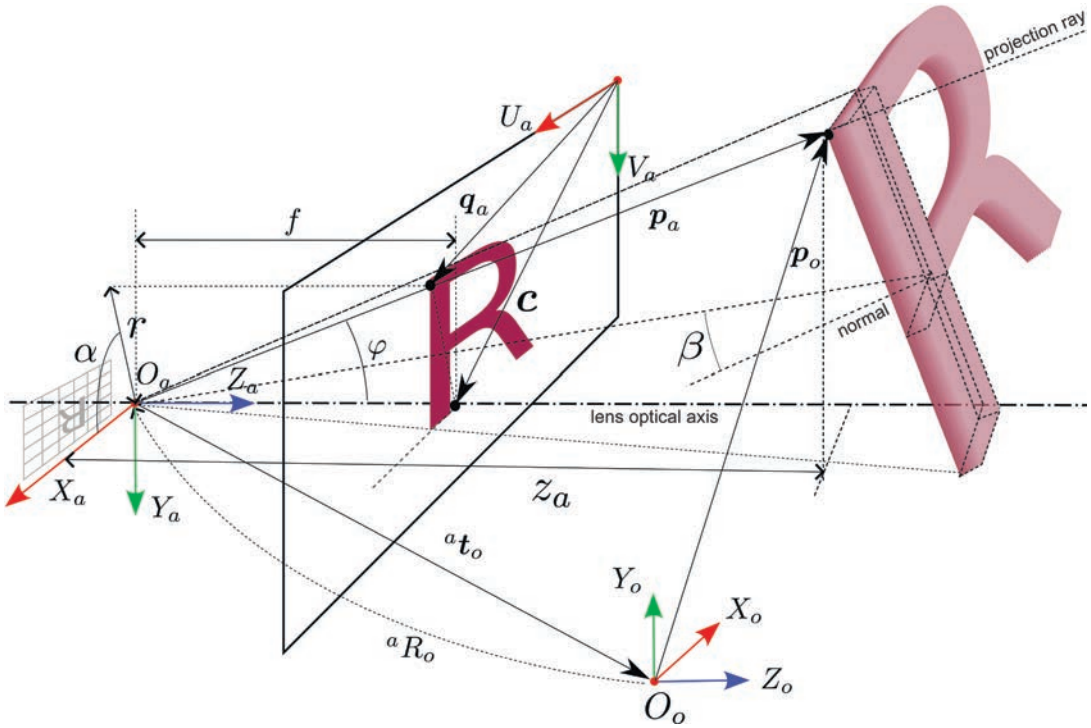

Fig. 7. Perspective or rectilinear camera model (6): 3D object relative to different coordinate systems and its projection onto screen or sensor (detailed description in text).

$\beta = 45°$, explanation in Fig. 7), the circle center $\mathbf{o} \in \mathbb{R}^2$ will perfectly correspond to the center $\mathbf{e} \in \mathbb{R}^2$ of the observed ellipse (Fig. 5b). In reality, at a very close surface distance $z_a = 15$ mm, strong projective distortion is observed (Fig. 5c), the magnitude of which depending on z_a is shown in Fig. 5d. Experimental conditions are presented in Section 7, using a Canon EOS 5DS camera.

The dependence shown in Fig. 5d is valid for a marker with radius $\mathcal{R} = 5$ mm and $\beta = 45°$. To build a similar dependence for a marker of a different radius, e.g., $\mathcal{R}' = 10$ mm, at the same observation angle β, the abscissa on the presented graph (Fig. 5c) must be multiplied (scaled) by the coefficient $\frac{\mathcal{R}'}{\mathcal{R}}$.

The difference between operators ${}^oA_a$ and ${}^oH_a$ is clearly shown in Fig. 5. When the distance to the surface is small and the area for extracting the feature center is large, an increasingly significant deviation is observed between the detected point $\mathbf{e}$ and the projection $\mathbf{o}$ of the estimated 3D point on the object. When the deviation exceeds 1:10 of the standard reprojection error in the bundle adjustment problem (Fig. 5d, red line on the graph), then not only the 3D object point but also the parameters of the surface patch orientation in space need to be optimized (Section 6).

In some applications, projective shape distortions allow reconstruction of object geometry from two or more coplanar circles [43]. Besides projective distortions described above, the deviation $||\mathbf{o} - \mathbf{e}||_2$ can be caused by much more complex nonlinear effects, e.g., camera lens distortion [44] or surface curvature. If the influence of these effects on the local feature geometry is significant, for an accurate bundle adjustment solution, one can directly minimize the residual of intensity values in each image pixel [26, 39]. This approach requires substantially more computational resources. For example, instead of two residual equations for the center of each marker (bundle adjustment problem, Section 6), there will be $O(\pi\mathcal{R}^2)$ equations[9] of color intensity differences in the feature pixels. Moreover, the number of parameters increases: for each optimized 3D point, a 3D normal[10] is added, i.e., at least five parameters per point instead of three. Such a solution is justified when working with a wide-angle camera model That significantly different from the common perspective model (Section 6.1).

Optical factors of detection. Extracting a feature from a large neighborhood creates difficulties described earlier. It seems that a simple solution would be to reduce the area size. For example, using a feature type like a cross, corner, or other line intersections on the object [45]. But under strong optical blur or other contrast loss (e.g., during noise suppression), the structure of such a feature can quickly degrade (Fig. 6).

The issue is that the high-frequency component of the signal,[11] necessary for depicting sharp edges of a corner, is gradually lost. Thus, a circle with a smooth contour structure remains the most universal marker for the most precise measurements (even if only the contour, not the center, is needed). Circular markers can be detected in real time using the following sequence of steps:

1) Coarse localization: "*blobs*" detector [35, 36];
2) Contour extraction [46, 47] or gradient extraction in the ellipse neighborhood [48];
3) Reliable and precise localization: quadratic function approximation of the ellipse as the locus of contour points or gradient field.

It is worth noting that subpixel refinement during contour extraction may not be required. The discretization noise of contour points $e_{edge} \sim \mathcal{U}(-0.5, 0.5)$, $\sigma_{edge} = \frac{1}{\sqrt{12}}$ (in pixels) is significantly averaged, so the noise in determining the marker center (mean of the random variable) is much smaller and amounts to $\sigma_{center} = \frac{\sigma_{edge}}{\sqrt{N_{pts}^{edge}}}$, N_{pts}^{edge} – number of points used to determine the center.

[9] Two equations per pixel, the number of which is proportional to the area of the observed marker.
[10] The orientation of a unit normal is defined by two angles.
[11] In computer vision, spatial signal frequency is typically discussed, unlike time function in electronics.

For example, for a circle with $\mathcal{R} = 25$ pixels, the discretization noise level of the found center coordinates σ_{center} will be $1/\sqrt{12 \cdot 2 \cdot \pi \cdot 25} = 0.02$ pixels.

The number of features does not compensate for their low quality – a general rule for precise photogrammetry. Since the parametric model of the ray (or camera) can be a high-order nonlinear function (Section 6), especially at the edges of the field of view, this inevitably leads to the overfitting.

5.2. Feature Descriptor

After key points detection a description for each point's texture neighborhood $\mathcal{D}_a = \{\mathbf{d}_a^k\}_{k=1}^{N_{pts}^a}$ is constructed (descriptor estimation, **second step** of the photogrammetric pipeline, page 1091). The key property of a descriptor is its stability or invariance to geometric transformations (due to camera or object motion) and to optical factors, e.g., due to illumination changes, defocusing, or sensor noise. That is, under the most diverse image acquisition conditions, the same object point should have identical, yet unique, descriptor. Descriptor invariance, in general, is unattainable, hence there is a wide variety of approaches effective for different scenarios.

For example, in the SfM task for large objects, affine distortions practically do not change angles in the feature structure, but rotations and scale changes are possible, similar to observations of celestial bodies [49]. In close-range photogrammetry for quality control of medium and small objects, on the contrary, noticeable projective distortions of features can occur (Fig. 5). Thus, the effectiveness of descriptors can differ significantly under different conditions [30]. This is important to consider when working with ML-based approaches [25, 31, 32, 34, 38] (when preparing the training database), or when selecting a suitable descriptor based on comparative review results [29].

The descriptor of natural texture features can be a vector of real numbers: $\mathbf{d} \in \mathbb{R}^n$ for SIFT [23], SURF [24], SuperPoint [25], DISK [38] or a binary vector $\mathbf{d} \in \{0,1\}^n$ for BRIEF [50], ORB [51], AKAZE [52]. Also, most artificial coded markers represent a binary descriptor [53–56]: QR-code, ARTag, AprilTag, ArUco, CCTag, Schneider's Coded Target (SCT), etc. SCT coded markers [53] (Fig. 3) are used in the experimental part. Markers with concentric circles (CCTag, SCT, etc.) are often used in industrial photogrammetry. "*Decoding*" markers or determining descriptor's binary sequence $\mathbf{d} \in \{0,1\}^n$ is significantly simplified by the ability to compensate for affine distortions of the feature: knowing the center, rotation angle, and magnitude of the principal axes, the five parameters of the affine transformation can be computed.

Machine learning can be effectively applied for decoding of specific key points and even arbitrary image regions [57]. Architectures based on convolutional networks can be used, e.g., Resnet [58] or Unet [59] encoding parts. As an output layer, a fully-connected bitwise classifier can be used, e.g., 12 output neurons with a sigmoid activation function for a 12-bit binary marker. By batching pixels from marker neighborhoods from several images, the decoding task can be efficiently solved on a GPU in real time.

5.3. Descriptor Matching

At this step, the found descriptor vectors of features from each image $\mathcal{D}_a = \{\mathbf{d}_a^k\}_{k=1}^{N_{pts}^a}$, $\mathcal{D}_b = \{\mathbf{d}_b^k\}_{k=1}^{N_{pts}^b}$, $a, b \in [1..N_{im}]$ are matched against each other (**third step** of the photogrammetric pipeline, page 1091), forming a set of corresponding indices: $\mathcal{M}_{a\to b} = \{(i^k, j^k) \mid i^k \in [1..N_{pts}^a], j^k \in [1..N_{pts}^b]\}_{k=1}^{N_{pairs}^{a\to b}}$. For each descriptor from the source set $\mathcal{D}_a$ with index $i \in [1..N_{pts}^a]$, a search for the nearest neighbor with index j^* in set $\mathcal{D}_b$ is performed ($l2$-norm of the difference as an example):

$$j^* = \underset{j \in [1..N_{pts}^b]}{\operatorname{argmin}} \|\mathbf{d}_a^i - \mathbf{d}_b^j\|_2. \tag{3}$$

In the simplest case, two sets $\mathcal{M}_{a\to b}$, $\mathcal{M}_{a\leftarrow b}$ are built (arrow indicates search direction) with subsequent filtering presented in Section 5.4. In the general case, optimal matching of more than two discrete descriptor sets belongs to the NP-complete transportation problem. Matching methods using machine learning [31, 32, 34, 60] effectively approximate the discrete search problem by combining local and global properties of descriptors.

If a pixel on the contour is used as a feature instead of a distinctive point (e.g., marker center), strict correspondence of 2D points between an image pair may not exist. In this case, the coordinates of the corresponding feature must be interpolated (assuming local smoothness or planarity of the surface).

As a distance function, depending on the nature of the vector, the $l2$-norm (Euclidean distance) [23–25, 38, 61] is often used, or for binary descriptors [50–52] – the number of identical bits (Hamming distance). In [61], an effective transformation of the distance function for the SIFT descriptor $\mathbf{d} = [d^1, \ldots, d^{128}]^{\mathrm{T}}$ is proposed, which significantly increases the probability of finding correct connections between images. Originally, the distance between descriptors is defined via the Euclidean norm, i.e., $\mathrm{dist}_E(\mathbf{d}_a, \mathbf{d}_b) = \|\mathbf{d}_a - \mathbf{d}_b\|_2$. However, considering that $\|\mathbf{d}_a\|_2 = \|\mathbf{d}_b\|_2 = 1$, the following holds:

$$\|\mathbf{d}_a - \mathbf{d}_b\|_2 = \sqrt{2 - 2\sum_{l=1}^{128} d_a^l \; d_b^l}.$$

The basis of SIFT is a frequency histogram or distribution function of some characteristic of the marker neighborhood. Histograms are also used when matching 3D features of point clouds, e.g., FPFH [62]. For comparing distributions similarity, instead of the $l2$-norm, it is better to use the Hellinger's f-divergence [61]. As a result of $l1$-normalization and element-wise square root, the distance between descriptors can be reduced to the corresponding form:

$$\mathrm{dist}_H(\mathbf{d}_a, \mathbf{d}_b) = \sqrt{2 - 2\sum_{l=1}^{128} \sqrt{c_a \; c_b \; d_a^l \; d_b^l}},$$

where c_a, c_b are normalization coefficients. Thus, by varying the distance function, the number of correct correspondences can be significantly increased.

Since the number of images N_{im} and the average number of found features N_{pts} per image can be large, the overall complexity of searching for similar descriptors among all images often becomes unacceptably high $O(N_{im}^2 \cdot N_{pts}^2)$ – as dozens of views (N_{im}) of a specific 3D point are needed for reliable results, and their total number of key points (N_{pts}) can easily exceed 1000. In this case, accelerating structures in the form of random trees with approximate nearest neighbor search [63][12] are primarily applied. For binary descriptors, the distance function differs from Euclidean, so methods based on hash functions [65] are used. Approximate search may allow a significant number of errors, but they can be filtered at the next stage.

Since images may not view the same object region, BoW techniques [61, 66] (abbreviation for "Bag-of-Words") are used to accelerate the selection of suitable pairs, allowing quick exclusion of non-overlapping frames and the need to match all descriptors for an image pair against each other.

5.4. Filtering of Found Correspondences

The results of the correspondence search need to be filtered (**fourth step** of the photogrammetric pipeline, page 1091) regardless of the matching methodology to minimize the number of errors or outliers.

[12] Ordinary kd-trees are inefficient – a problem known as the "curse of dimensionality," where sequential search turns out to be faster than tree traversal due to information distribution in multidimensional data structures [64].

From (3) it follows that the search direction matters and the sets $\mathcal{M}_{a\to b}$ and $\mathcal{M}_{a\leftarrow b}$ may not be identical. This is the basis for the "mutual correspondence" filter [30–32], i.e., mutually nearest descriptors or the intersection of pairs from sets $\mathcal{M}_{a\to b}$ and $\mathcal{M}_{a\leftarrow b}$ are used, while the rest are discarded. Another popular filter is based on descriptor "uniqueness" [23] – it discards match if one of the descriptors (from original image) is relatively close to multiple others (on a different image). Another common method is using repeatedly occurring descriptors forming a sequence (e.g., "tracks" in video data analysis).

These and similar heuristics by themselves are not very effective and can remove a large number of correct correspondences. This is because the local texture properties of the object, represented by descriptors, vary greatly even within a single image. Essentially, the task of such filters is to screen out the coarsest matching errors and accelerate subsequent steps.

The most effective filtering methods rely on global context [21, 30, 42], e.g., checking how well the found correspondences satisfy the geometric properties of space and the perspective camera model (Section 6.1). If the camera model significantly differs from perspective, e.g., for wide-angle lenses, or has large distortion, then fast methods considering these distortions are required [40, 67].

First, the most universal and widely used criterion for geometric consistency is epipolar geometry – all correspondences on an image pair must satisfy the following epipolar line equation:

$$\begin{bmatrix} u_a & v_a & 1 \end{bmatrix} {}^aF_b \begin{bmatrix} u_b \\ v_b \\ 1 \end{bmatrix} = 0, \tag{4}$$

u_a, v_a, u_b, v_b – coordinates of two projections of the same 3D point on images a and b respectively, ${}^aF_b \subset \mathbb{R}^{3\times 3}$ – the well-known fundamental matrix, $\operatorname{rank}({}^aF_b) = 2$. The geometric meaning of equation (4) is that matrix ${}^aF_b$ establishes a point-line correspondence between an image pair: $\begin{bmatrix} u_a & v_a & 1 \end{bmatrix} {}^aF_b$ – parameters of a line on image b, while ${}^aF_b \begin{bmatrix} u_b \\ v_b \\ 1 \end{bmatrix}$ – a line on image a. Substituting eight pairs of corresponding points into (4), a system of linear algebraic equations [68] is built for the unknown matrix ${}^aF_b$. Considering that $\operatorname{rank}({}^aF_b) = 2$, using seven point pairs, a system of nonlinear equations for finding ${}^aF_b$ can be built [26]. Thus, by repeatedly solving these equations for small groups of corresponding points, in statistical methods like RANSAC [69] or PROSAC [70], a significant portion of matching errors can be filtered out.

Second, filtering reliability can be increased if it is known that the points lie on a plane or depth variations on the object surface are significantly smaller than its dimensions. In this case, only 4 correspondences are required to define a 2D point transformation model when using methods like RANSAC or DEGENSAC [71]:

$$\lambda \begin{bmatrix} u_a \\ v_a \\ 1 \end{bmatrix} = {}^aH_b \begin{bmatrix} u_b \\ v_b \\ 1 \end{bmatrix}, \tag{5}$$

where ${}^aH_b \subset \mathbb{R}^{3\times 3}$ is the projective transformation matrix or homography between corresponding projections of a common 3D point, $\det({}^aH_b) \neq 0$.

Machine learning methods allow extraction of distinctive points and building local descriptors [25, 38]. In [31], texture features are extracted on practically homogeneous areas of natural texture. In [31, 32, 34], features are supplemented with contextual information from some image area calculated using a transformer-based neural network architecture or similar modifications for computational efficiency. As a result of pairwise matching, a feature correspondence probability matrix is obtained: $P^{a,b} \in \mathbb{R}^{N^a_{pts}\times N^b_{pts}}$, $\sum_{i=1}^{N^a_{pts}} P^{a,b}_{ij} \leqslant 1$, $\sum_{j=1}^{N^b_{pts}} P^{a,b}_{ij} \leqslant 1$. Thus, correspondences with

the required reliability can be selected, excluding points invisible from both viewpoints. In the context of precise photogrammetry, the presented ML-based solutions are excellent for finding a reliable initial approximation and constructing approximate camera positions and 3D point clouds. But unfortunately, they have low feature localization accuracy – the reprojection error in various tasks often exceeds one pixel, and the angular error in position determination is over 5°; often the training dataset is built on flat scene areas, i.e., heavily relies on (5), and matching results still require filtering [30, 32].

Practically regardless of design, the local descriptor in photogrammetry is used at the preliminary stage of finding an initial approximation in the bundle adjustment problem, in the absence of prior information about scene geometry and camera positions. When these parameters are known with sufficient accuracy, feature matching can be performed along epipolar lines obtained from (4), which significantly increases the number of correct correspondences.

6. PHOTOGRAMMETRIC OPTIMIZATION PROBLEM

6.1. Camera Model and Perspective

To solve the optimization problem in the photogrammetric pipeline, known as bundle adjustment for projection rays, it is necessary to define the key component of the pipeline. The "heart" of this technology, without exaggeration, is the camera model – a function for mapping or projecting points from the surrounding 3D space onto an image (screen or sensor). A key property of any model considered in this review is the rectilinear propagation of light; diffraction or chromatic aberrations are considered as negligible. Through any 2D point on the sensor, a ray can be drawn that will hit the corresponding 3D point on the object surface. Thus, the camera model defines the direction of projection rays based on the 2D sensor point, internal parameters, and camera position in space.

The perspective or rectilinear pinhole camera model [3, 11, 26, 41, 72, 73] is the most common in computer vision. Its characteristic feature is that 3D straight lines in object space are projected into 2D straight lines in image space. This model is often assumed (explicitly or implicitly) as the baseline in various studies. For example, it is used when finding homography (5), when determining mutual position between images [25, 31, 32, 34], or for 3D scene reconstruction [42]. This model serves as the initial approximation for more complex parameterizations of projection rays, considered further. The rectilinear camera model is defined by the projection matrix ${}^aP_o \in \mathbb{R}^{3\times4}$, $\operatorname{rank}({}^aP_o) = 3$ and establishes the following relationship:

$$\lambda \mathbf{q}_a^h = {}^aP_o\ \mathbf{p}_o^h = \begin{bmatrix} f_u & s_{uv} & u_c & 0 \\ 0 & f_v & v_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^aR_o & {}^a\mathbf{t}_o \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}_{4\times4} \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix} = \begin{bmatrix} K & \mathbf{0} \end{bmatrix}_{3\times4} {}^aT_o \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}, \tag{6}$$

$\mathbf{q}_a^h = [u_a, v_a, 1]^{\mathrm{T}}$ – 2D coordinates on image a, result of projecting 3D point $\mathbf{p}_o^h = [x_o, y_o, z_o, 1]^{\mathrm{T}}$ in the object CS;

K – upper-triangular matrix of internal camera parameters $\det(K) \neq 0$ (five degrees of freedom, in pixels), includes focal lengths $f_u,\ f_v$,[13] coordinates of the projection center $\mathbf{c} = [u_c, v_c]^{\mathrm{T}}$ – the point where the optical axis of the lens intersects the sensor (Fig. 7), and also the sensor diagonal distortion s_{uv};

${}^aT_o \in \mathbf{SE3}$ – matrix of external camera parameters (six degrees of freedom) defines the position of the object relative to the camera at the moment of capturing image a and includes rotation ${}^aR_o \in \mathbf{SO3}$ and translation of the object CS origin ${}^a\mathbf{t}_o$;

[13] In computer vision, unlike the classical optical model, two "focal lengths" are distinguished for convenience to account for the possible non-square pixel shape in these quantities.

parameter λ is set equal to z_a, where z_a – z-coordinate of the considered point or distance from the origin O_a along axis Z_a to the point (Fig. 7).[14]

Based on (6), let's define normalized homogeneous projection screen coordinates $\bar{\mathbf{q}}_a^h$, which are useful in the future. In object space, they correspond to the coordinates of 3D points on the plane for which $Z_a = 1$ in front of the camera, or projection onto a camera with internal parameters $K = I_{3\times3}$ ($f = 1$):

$$\bar{\mathbf{q}}_a^h = \begin{bmatrix} \bar{u}_a \\ \bar{v}_a \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{x_a}{z_a} \\ \frac{y_a}{z_a} \\ 1 \end{bmatrix} = \frac{1}{z_a} \begin{bmatrix} {}^aR_o & {}^a\mathbf{t}_o \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}. \tag{7}$$

The operation scheme of the rectilinear camera model is shown in Fig. 7. For clarity in the example, one can set $f_u = f_v = f$, $s_{uv} = 0$ (often these assumptions hold for precise measurements [17]); the physical pixel size, sensor size, and focal length f value are irrelevant for the mathematical formulation, what matters is the ratio of quantities; the physical sensor (with reflected projection) is located behind the camera CS center O_a (optical center of the lens), its mathematical model is conventionally placed between the center and the object [72].

Using the structure of ${}^aP_o$ from (6) and considering that ${}^aR_o^{\mathrm{T}} = {}^aR_o^{-1}$, one can easily construct the projection ray equation for any 2D screen point and compute the 3D coordinates of point $\mathbf{p}_o$ relative to the object CS:

$$\mathbf{p}_o = \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} = {}^aR_o^{\mathrm{T}}(K^{-1}\lambda\mathbf{q}_a^h - {}^a\mathbf{t}_o), \tag{8}$$

where $\mathbf{q}_a^h$ – screen point coordinates, $\lambda = z_a$ – depth, and the internal and external camera parameters are given.

The central equation of photogrammetry – the *triangulation problem* consists in determining the coordinates of a 3D point $\mathbf{p}_o$ from known projections $\mathbf{q}_a$, $\mathbf{q}_b$. For the rectilinear camera model, considering (6), a system of linear algebraic equations can be built:

$$\begin{cases} \lambda_a\mathbf{q}_a^h = \begin{bmatrix} K & \mathbf{0} \end{bmatrix} {}^aT_o\ \mathbf{p}_o^h, \\ \lambda_b\mathbf{q}_b^h = \begin{bmatrix} K & \mathbf{0} \end{bmatrix} {}^bT_o\ \mathbf{p}_o^h, \end{cases} \tag{9}$$

where ${}^aT_o$, ${}^bT_o$ – matrices defining the mutual position of the object and observer/camera, K – internal parameters of the camera (for multiple cameras scenario, the matrices may differ). In the triangulation problem, K, ${}^aT_o$, ${}^bT_o$ are known and the presence of parallax is important, i.e., $\|{}^a\mathbf{t}_b\|_2 \neq 0$. System (9) contains six equations and five unknowns (λ_a, λ_b, $\mathbf{p}_o$). Unknowns λ_a and λ_b are easily expressed, resulting in an overdetermined system of linear algebraic equations (four equations and three unknowns), which can be solved with the least squares approach (e.g., *normal equations*) for the 3D coordinates $\mathbf{p}_o$.

When solving the bundle adjustment problem (Section 6.5) in equation (9), the only known quantities are the detected coordinates $\mathbf{q}_a$, $\mathbf{q}_b$ (two residual equations for each observed 3D object point on the image), all other parameters are optimized.

[14] The three-dimensional representation of the surface is often stored in the form of depth maps relative to the optical center of the lens.

6.2. Basic Projection Models

The rectilinear camera model (6) establishes the relationship between the angle φ between the projection ray and the optical axis with the screen point coordinates $\mathbf{q}_a$ (Fig. 7):

$$\tan(\varphi) = \frac{r}{f} = \frac{||\mathbf{q}_a - \mathbf{c}||}{f} = ||\bar{\mathbf{q}}_a||.$$

In other words, this is the relationship between the angles of the ray from the surrounding space (collinear with the radius vector $\mathbf{p}_a$, entering the camera lens) and the ray falling on the sensor behind the lens; in the perspective mathematical camera model, both rays lie on the same straight line passing through the optical center of the lens O_a in Fig. 7.

In the physical lens model, the scattered light beam is collected from a surface area and undergoes a series of complex refractions; the presented mathematical models approximate this image formation process. Depending on the lens shapes and construction, at least three basic projection models can be distinguished [44]:

1) Rectilinear or perspective model: $r = f\tan(\varphi)$;
2) Stereographic model: $r = 2f\tan(\frac{\varphi}{2})$;
3) Equidistant model: $r = f\varphi$ (ideal wide-angle optics, since sensor resolution does not depend on angle φ).

In popular wide-angle cameras like Insta360 X4, two lenses with equidistant projection models and field of view exceeding 180° each are used to provide a panoramic view in video mode. An example of two projections is shown in Fig. 8.

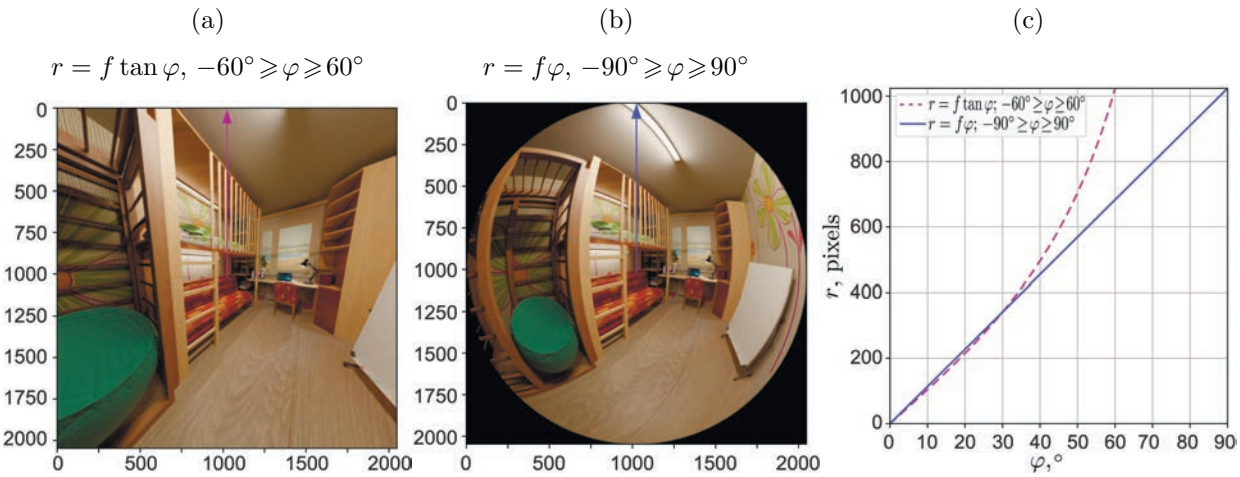

Fig. 8. Realistic 3D model of children's room (rendered in Blender 3D [84]) in two ideal projections without distortion: *a*-rectilinear, *b*-equidistant.

Let us present the equations for mapping a certain 3D point in the camera CS $\mathbf{p}_a = [x_a, y_a, z_a]^{\mathrm{T}}$ to a 2D point $\mathbf{q}_a$ on the original image for various projection models. Let φ, α be the spherical coordinates of the projection ray (3D point on a sphere), where φ is the angle between the optical axis Z_a and the projection ray, α is the rotation angle of the projection ray around axis Z_a (Fig. 7):

$$\cos(\alpha) = \frac{x_a}{\sqrt{x_a^2 + y_a^2}}, \quad \sin(\alpha) = \frac{y_a}{\sqrt{x_a^2 + y_a^2}}, \quad \tan(\varphi) = \frac{\sqrt{x_a^2 + y_a^2}}{z_a}. \tag{10}$$

As a result, the image point coordinates can be computed via polar coordinates α, $\rho(\varphi)$:

$$\mathbf{q}_a^h = K \begin{bmatrix} \rho(\varphi)\cos(\alpha) \\ \rho(\varphi)\sin(\alpha) \\ 1 \end{bmatrix}, \quad \rho(\varphi) = \begin{cases} \tan(\varphi) & \text{– for rectilinear model,} \\ 2\tan(\frac{\varphi}{2}) & \text{– for stereographic model,} \\ \varphi & \text{– for equidistant model.} \end{cases} \tag{11}$$

Substituting expressions (10) into (11), the 2D projection coordinates for any of the presented models can be obtained. It is easy to see that for the rectilinear model, (11) takes the form of (6).

There exist models combining rectilinear, stereographic, and equidistant projections. In [44], a parameter for smooth adjustment of the projection ray refraction model is introduced. This work also emphasizes the importance of accounting for the entrance pupil shift or lens optical center for wide-angle optics. Developing this idea, generalized camera's projection models with individual ray parameterization should be highlighted [45, 74, 75], where each image pixel is assigned its own ray parameters. In practice, only a subset of such pixels is selected, and parameters for the rest are interpolated. Naturally, such models easily "overfit" and require a lot of data for precise results.

In the photogrammetric pipeline, use of an appropriate projection model can significantly reduce the magnitude of distortion and decrease the total number of parameters requiring good initial approximation and careful calibration.

6.3. Distortion of Projection Models

Due to various factors, e.g., complexity of lens and optics manufacturing, sensor curvature, or camera assembly errors, the actual projection often deviates from the model, especially near the field of view or image borders. This phenomenon is called *distortion*. Often, the distortion refers to the difference between basic projection models, e.g., curvature of straight lines or shape of 3D objects in Fig. 8, since the distortion function approximates this effect [67, 76].

Typically, for the rectilinear camera model, radial and tangential distortion components are distinguished [3, 4, 17, 72]. Despite its age and significant technological development, this model works very effectively, as demonstrated in Section 7.

Radial distortion is the most significant factor distorting the rectilinear projection, with straight lines curving in a "barrel" or "pincushion" shape. It is approximated by an even-degree polynomial, as the distortion function is symmetric due to the central symmetry of lenses:

$$\delta\mathbf{r}(\mathbf{q}) = \mathbf{q}(k_r^1 r^2 + k_r^2 r^4 + k_r^3 r^6 + \ldots), \tag{12}$$

$\mathbf{q} = [u, v]^{\mathrm{T}}$ – some 2D point on the screen (distortion center at $[0, 0]^{\mathrm{T}}$);
k_r^γ – radial distortion coefficients ($\gamma \in \mathbb{N}$);
$\delta\mathbf{r}(\mathbf{q})$ – deviation of the observed 2D point from the rectilinear projection $\mathbf{q}$ (projection model (6)) due to radial distortion. The main contribution to the deviation from the rectilinear model is typically made by the first term $k_r^1 r^2$ on the right side of (12), while subsequent terms are often omitted.

Tangential distortion is caused by installation errors of the lens system and, unlike $\delta\mathbf{r}(\mathbf{q})$, creates asymmetric field distortion, formulated by [2]:

$$\delta\tau(\mathbf{q}) = \begin{bmatrix} k_\tau^2(r^2 + 2u^2) + 2k_\tau^1 uv \\ k_\tau^1(r^2 + 2v^2) + 2k_\tau^2 uv \end{bmatrix} (1 + k_\tau^3 r^2 + k_\tau^4 r^4 + \ldots), \tag{13}$$

where k_τ^γ – tangential distortion coefficients ($\gamma \in \mathbb{N}$).

Coefficients k_τ^3, $k_\tau^4, \ldots$ are typically not accounted for, or the $\delta\tau(\mathbf{q})$ factor is entirely ignored. However, the tangential distortion factor should not be completely neglected, especially in the field of precise measurements with photogrammetry: in the foundational work [3], the distortion center coordinates coincide with the projection center $\mathbf{c}$ from the camera's internal parameters K.

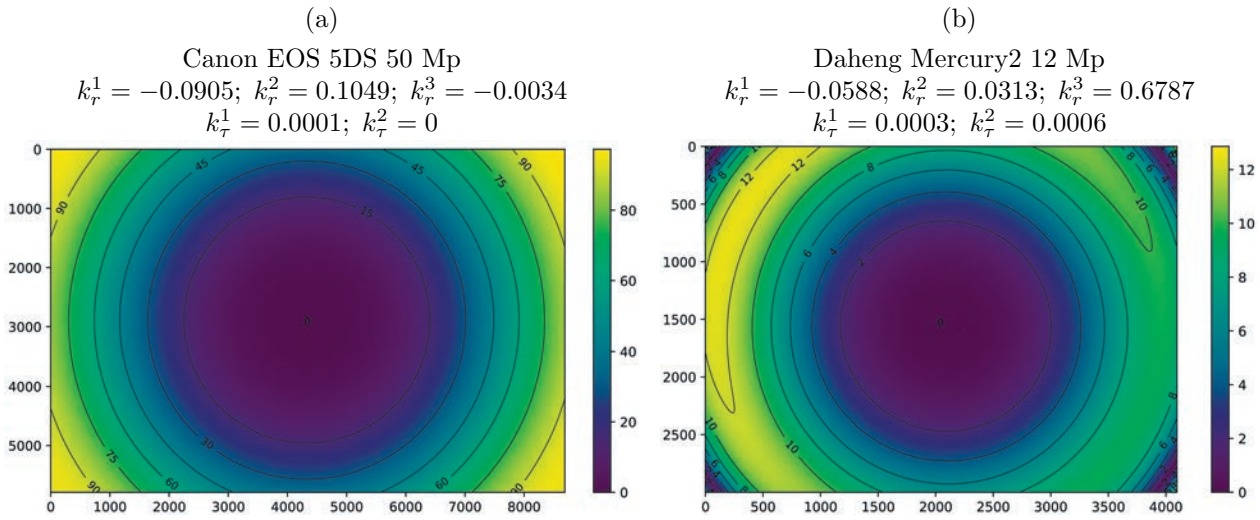

Fig. 9. Magnitude of distortion $||\delta\mathbf{r}(\mathbf{q})+\delta\tau(\mathbf{q})||_2 \cdot f$ at each frame point (in pixels). Both optical systems mainly demonstrate centrosymmetric deviation from the rectilinear model due to radial distortion.

Meanwhile, in works [67, 76, 77] neglecting tangential distortion, it is noted that $\mathbf{c}$ and the distortion center coordinates significantly deviate. Although in [3, 17] it is indicated that tangential distortion strongly correlates with the projection center position. Probably in [67, 76, 77], it is the unaccounted tangential distortion that distorts the estimate of $\mathbf{c}$.

Thus, the precise rectilinear projection model (6) accounting for distortion defines the relationship between 3D coordinates and their 2D projections as follows:

$$K^{-1}\mathbf{q}_a^h + \delta(K^{-1}\mathbf{q}_a^h) = \frac{1}{z_a}\begin{bmatrix} {}^aR_o & {}^{\mathbf{a}}\mathbf{t}_{\mathbf{o}} \end{bmatrix}\begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix} = \bar{\mathbf{q}}_a^h, \tag{14}$$

K – internal parameter matrix of the rectilinear camera model,[15]
$\mathbf{q}_a^h$ – 2D coordinates of the found feature on the original image,
$\bar{\mathbf{q}}_a^h$ – normalized 2D coordinates of the projection of the corresponding 3D point from (7),
$\delta(\mathbf{q}^h) = \begin{bmatrix} \delta\mathbf{r}(\mathbf{q}) + \delta\tau(\mathbf{q}) \\ 0 \end{bmatrix}_{3\times 1}$ – combined distortion for homogeneous coordinates.

The fundamental formula (14) is widespread in photogrammetric literature [2–4, 11, 17], where the polynomial function corrects the distortion of the observed point $\mathbf{q}_a$. This differs from the formulation in the popular category of "non-photogrammetric" works, e.g., [40, 41, 67, 72], where the polynomial function, conversely, adds the distortion effect to normalized coordinates:

$$\mathbf{q}_a^h = K[\ \bar{\mathbf{q}}_a^h + \delta(\bar{\mathbf{q}}_a^h)\ ]. \tag{15}$$

Section 7 provides a comparison of the effectiveness of models based on expressions (14) and (15). A clear example of two calibrated cameras (distortion models) is presented in Fig. 9.

For the Daheng Mercury2 camera, asymmetry is noticeable – the contribution of tangential distortion. Moreover, non-monotonicity (decrease) in the magnitude of radial distortion is observed at the frame corners, and a very large value of k_r^3 (Fig. 9b). All this is the result of instability in the optimization problem solution due to lack of observations (markers), especially in areas with

[15] Essentially, K^{-1} performs normalization of the image coordinate space.

strong distortions. In other words, the distortion is approximated by a high-order polynomial, and it incorrectly extrapolates the distortion magnitude [17].

As follows from the presented results, accounting for distortion is necessary, as its magnitude is very significant. Therefore, rectilinear camera projection models with distortion (14) and (15) are extremely widespread and found in numerous works on photogrammetry, SfM/SLAM, visual odometry, augmented reality, etc. In this work, the experimental part considers the rectilinear model with distortion (14) and (15).

6.4. Autocalibration

In photogrammetry, joint optimization of 3D coordinates of observed features, internal camera parameters, and their positions is performed (Section 6.5). To perform the nonlinear optimization procedure, an initial estimation for optimized parameters is found by the autocalibration procedure [26, 78] (**fifth step** of the photogrammetric pipeline, page 1091). Methods for 3D reconstruction with ML [42] also require such a procedure. For preliminary calibration and camera positioning in professional photogrammetry "Hexagon DPA Pro", special tools are used – groups of coded markers placed on a known cross structures. This approach has significant inconveniences. The algorithms presented below don't need any special structures in the scene for parameters estimation.

In the most common photogrammetric scenario with a single rectilinear camera (page 1104), moved against the measured object, the following optimized quantities can be distinguished:

1) 3D coordinates of observed features on the object surface: $\{\mathbf{p}_o^i \in \mathbb{R}^3\}_{i=0}^{N_{pts}}$;
2) Internal camera parameters (elements of K) and distortion coefficients: $\{f, \mathbf{c}, k_r^1, k_r^2, k_r^3, k_\tau^1, k_\tau^2, ...\}$;
3) Camera positions at different moments of time: $\{{}^aT_o \in \mathbf{SE3}\}_{a=2}^{N_{im}}$, fixing the first position ${}^1T_o = I_{4\times4}$ as the object CS.[16]

Distortion at the autocalibration step can often be neglected – the found parameters will be correct for the central area of the image (or sensor). Precise estimation of distortion parameters will be performed at the next step. For the projection center $\mathbf{c}$ – the image center is typically a good initial approximation.

If camera's distortion is too large and prevents reliable autocalibration, or the projection model differs from rectilinear due to a large field of view, then an iterative process of prior undistortion and refinement of parameters is required. In most cases, it is the rectilinear projection model that is used in autocalibration algorithms [78, 79]. In [79], autocalibration is reduced to determining the mutual camera positions ${}^aR_b,\ {}^a\mathbf{t}_b,\ a, b \in [1...N_{im}]$, using previously found correspondences, with (4) and the structure of the fundamental matrix:

$$ {}^aF_b = K^{-\mathrm{T}}\ {}^aE_b\ K^{-1} = K^{-\mathrm{T}}\ [{}^a\mathbf{t}_b]_\times\ {}^aR_b\ K^{-1}, \tag{16}$$

$[\mathbf{t}]_\times = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$ – skew-symmetric matrix of the vector product.

The translation between cameras ${}^a\mathbf{t}_b$ can be determined only up to scale (the vector specifies only direction) – this is a theoretical limitation of autocalibration. To find ${}^aR_b,\ {}^a\mathbf{t}_b$, it's sufficient to have five corresponding points to calculate residuals from (4), since ${}^aF_b$ has five degrees of freedom [26]. The minimal number of correspondences allows effective application of RANSAC [69] to improve the stability of the found solution. In [79], for autocalibration with two images (or camera positions),

[16] In the general case, the photogrammetry problem is solved up to a **SE3** transformation with an arbitrary scale of the resulting point cloud [26], thus parameters of one position can be fixed.

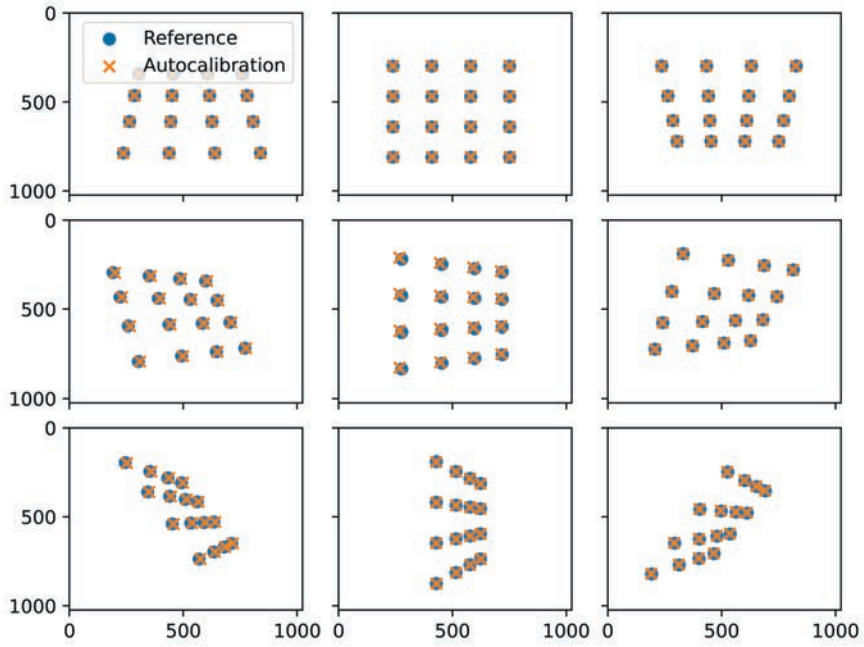

Fig. 10. Model images (1024x1024 pixels) of 9 tilted views for a flat object (circular markers) and projection of found 3D coordinates (crosses) – result of the autocalibration algorithm [78], estimating internal and external camera parameters, 3D coordinates of object points. Root mean square reprojection error according to (6) is 3.35 pixels with input observation noise amplitude of one pixel.

the internal camera parameters (matrix K) are required. At the same time, the focal length f is difficult to estimate in advance, as it can vary significantly depending on the field of view. Moreover, if the observed correspondences (for two views or camera positions) lie on a flat surface, then the internal parameters cannot be estimated [26, 79]. Thus, the most reliable solution from [79] is to use three or more images obtained from different positions.

An alternative approach [78] based on factorization of absolute quadric matrix was implemented in this work. The problem reduces to solving a nonlinear system of equations:

$$K^{-1}\, P_b\, \Omega\, P_b^{\mathrm{T}}\, K^{-\mathrm{T}} = \lambda\, I_{3\times 3} \tag{17}$$

against the elements of the quadratic form matrix $\Omega = HH^{\mathrm{T}} \in \mathbb{R}^{4\times 4}$ and the internal parameter matrix K,
λ – arbitrary scale coefficient,
$P_b = K[\ {}^bR_a|{}^b\mathbf{t}_a]_{3\times 4}H^{-1}$ – known projective transformation matrix (6) (position b, for example), which was obtained from factorization of the fundamental matrix ${}^aF_b$, and ${}^aF_b$, in turn, was found from correspondences between images a, b. Thus, knowing correspondences between several views (at least three), the relative camera position ${}^bR_a$, ${}^b\mathbf{t}_a$ and its parameters K can be determined.

At the next step, unknown 3D point coordinates are computed via the triangulation equation (9), followed by the bundle adjustment problem. Figure 10 shows the result of the autocalibration algorithm for a flat object from correspondences from nine views. Various methods for estimating relative position and camera parameters without initial approximation are also available in the open-source library [80].

6.5. Bundle Adjustment

Measurements obtained by photogrammetry are the 3D coordinates of observed 2D features, as well as the spatial position of objects (corresponding point clouds) relative to cameras. This requires

joint optimization of the multiple parameters listed above. The central problem of photogrammetry is minimizing the sum of squared residuals or reprojection errors (**sixth step** of the photogrammetric pipeline, page 1091):

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\ \mathcal{L}(\theta), \tag{18}$$

where θ – combined vector of optimized parameters (Section 6.4);
$\mathcal{L}(\theta)$ – minimized objective or loss function.

The geometric meaning of the problem is bringing the bundle of projection rays to converge at the corresponding 3D points by minimizing the reprojection error, hence the name "bundle adjustment".

The terms of the loss function are derived from the camera's projection model. The type of projection model is based, ideally, on the camera's lens design, e.g., (6) for a rectilinear camera without distortion (special lens design, for example), (11) for a wide-angle camera, (15) or (14) for a precise rectilinear model with radial and tangential distortion. The loss function with reprojection error from (14) has the following form:

$$\begin{aligned} \mathcal{L}(\theta) &= \sum_{a=1}^{N_{im}} \sum_{i=1}^{N_{pts}} e_{a,i}(\theta) + \ w_{scale} \sum_{(i,j,d)\in S} (||\mathbf{p}_o^i - \mathbf{p}_o^j||_2 - d)^2, \\ e_{a,i}(\theta) &= \left\| K(\theta)^{-1}\mathbf{q}_a^{m_a(i)} + \delta_\theta(K(\theta)^{-1}\mathbf{q}_a^{m_a(i)}) - \frac{1}{z_a}\ {}^aT_o(\theta)\ \mathbf{p}_o^i \right\|_2^2, \end{aligned} \tag{19}$$

where $e_{a,i}(\theta)$ – square of $l2$-norm of 2D residual or reprojection error for one point on the image;
$\mathbf{p}_o^i$ (component of vector θ) – unknown 3D point on the object surface (e.g., marker in Fig. 3), symbol h omitted for brevity;
$m_a(i)$ – mapping of the ith 3D point on the object to the index of its projection on image a (assuming, for brevity, that all object points are visible on the image);
$\mathbf{q}_a^i$ – homogeneous 2D coordinates of the feature in pixels, original observations on image a;
$S = \{(i,j,d)|i,j \in [1..N_{pts}], d \in \mathbb{R}^+\}_{i=1}^{N_{scales}}$ – scale bars (N_{scales} pieces) in the form of known points and calibrated distances between them, e.g., coded markers on two scale bars in Fig. 3;
${}^aT_o(\theta)$ – camera CS position includes rotation matrix ${}^aR_o(\theta)$ and origin ${}^a\mathbf{t}_o$ (component of vector θ) when capturing image a;
w_{scale} – weight of the scale measurement error term. The weight is needed to balance the different numbers of equations of two types. Moreover, the reprojection error can have different units: pixels for (15) and normalized coordinates for (14).

For better numerical stability, it is desirable to normalize image coordinates so that $\mathbf{q}_a \in [-1,1]^2$. This is common practice, but an important nuance can be missed – first, the origin must be shifted from the center of the first pixel to the corner of the image to decouple the coordinate system from the original image pixels. Thus, the resulting camera calibration K will not be tied to a specific resolution.

A number of key aspects of the optimization problem (18) should be noted:

1) Relatively large number of optimized parameters: in the experimental scene, 71 camera positions (six degrees of freedom each), and 430 markers on the object, i.e., at least $71 \cdot 6 + 430 \cdot 3 = 1716$ parameters;
2) Sparsity of the equation system (Jacobian): each term in (19) – reprojection error includes only one position, and not all object points are observed in every frame;
3) Presence of outliers or points with very high reprojection error: Fig. 11 shows that in almost every position there are points with residuals multiple times higher the root mean square error σ_{repr}.

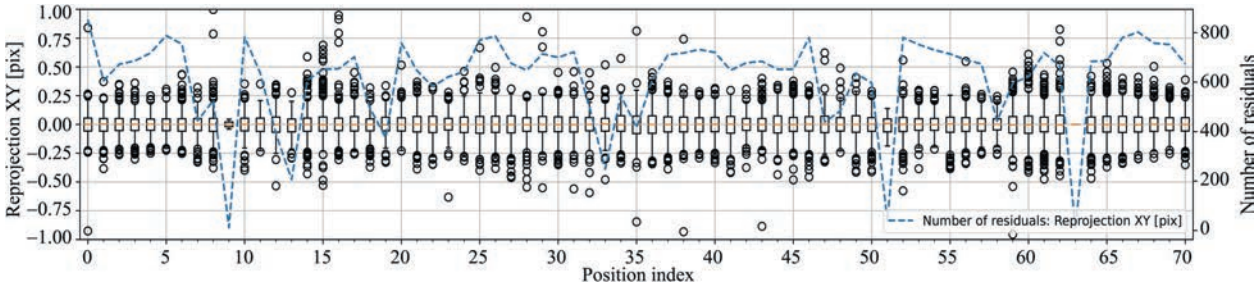

Fig. 11. Distribution of reprojection error of 3D points from (19) (left Y-axis) in pixels across 71 camera positions as box plots covering 99% of points. Circle denotes outlier (less than 1% of points). Dashed line (right Y-axis) denotes number of residuals per position – two for each observed point.

Thus, it is necessary to use optimization methods robust to noise and outliers (even after careful feature detection and filtering). As seen in Fig. 11, outliers occur more frequently than can be predicted based on a normal or Gaussian distribution, as emphasized in [39]. It proposes replacing $e_{a,i}(\theta)$ in (19) with $\rho(e_{a,i}(\theta))$, where $\rho(x)$ is a non-negative scalar function or "robust kernel": $\rho(x) > 0,\ \forall x \in \mathbb{R}, \rho'(0) = 0$, which suppresses gradients of the loss function in case of outliers due to $\rho'(x) < 1$. In the experimental part, the Huber function was used for this purpose:

$$\rho_t(x) = \begin{cases} x, & \text{if } x < t^2, \\ t^2\ (2\sqrt{x/t^2} - 1), & \text{otherwise}, \end{cases} \tag{20}$$

where $t = 0.05$ pixels – normalization coefficient.

For residual values below t, the robust kernel function has no effect on the loss function; however, in the outlier region above t, robust kernel significantly reduces outliers contribution to the loss function (19).

For reliable solution of the bundle adjustment problem (18), approximate Newton methods with regularization are used – at each algorithm's iteration, step direction and magnitude adjustment is performed [39]. The Levenberg–Marquardt optimization algorithm from this category was used in the experimental part. An efficient implementation of various optimization methods, different robust kernels and sparse matrix operations is available in [81].

7. EXPERIMENTAL PART

The experiments goal is to highlight the key factors of the optimization process (18) that affect the accuracy of the reconstructed 3D point cloud. Original data (digital images) was acquired with camera from "Hexagon DPA Pro" (HDP) hardware set, specifically, Canon EOS 5DS with a monochrome sensor $w \times h = 8700 \times 5800$ (50 Mp) and lens $f = 28$ mm. It should be noted that various professional DSLR camera models can be used for precise measurements [28].

The test scene and the reconstructed 3D point cloud of 430 markers are shown in Fig. 3. The scene includes two rigid objects – a steel welding table with attached markers (overall dimension ~2000mm) and a carbon fiber calibration plate (overall dimension ~800mm, markers applied by the manufacturer). Two scale bars of 1390mm and 790mm define metric units for the reconstructed point cloud and provide redundant accuracy verification. Photography was performed carefully with environmental temperature control (20 °C) in the lab. Two nearly identical datasets were collected with a one-day interval (74 and 79 images). Image processing for each day was performed both in the HDP software package and in the authors' own software based on the Python language and open-source mathematical libraries Jax [82], SciPy [83], OpenCV [72].

The photogrammetric pipeline of reconstruction included the following steps:

1) Photography of the test object from different positions (distance between camera and object 1.5–2 m), example in Fig. 4;

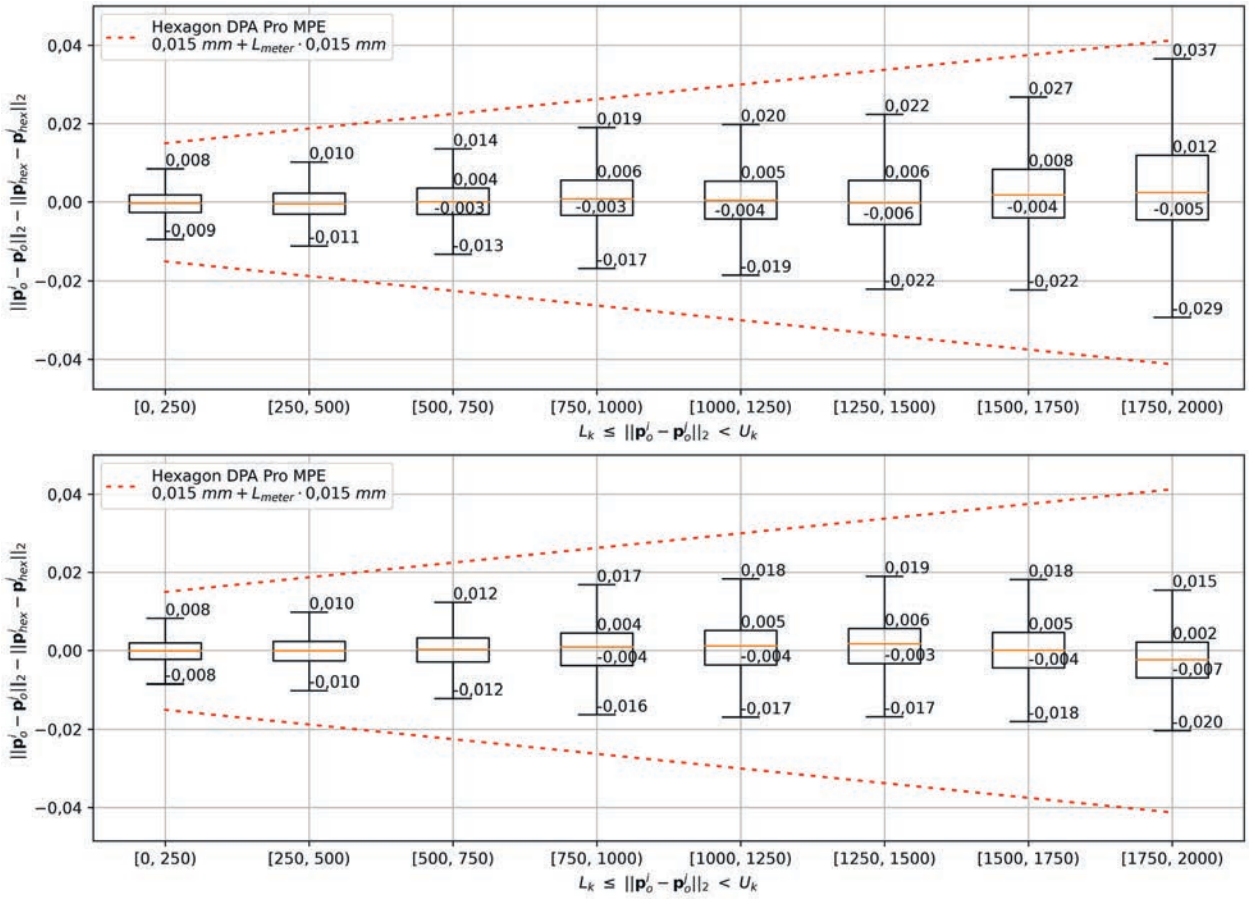

Fig. 12. Comparison of photogrammetric reconstruction results of the test scene from "Hexagon DPA Pro" and the current approach for two experiments (data from two days – upper and lower graphs). Units in **millimeters**. X-axis shows measured distance ranges.

2) Computation of circular marker parameters (coded and regular) on images and matching (grouping) of coded markers by binary code, Section 5;
3) Autocalibration estimating rough camera calibration (without accounting for distortion (12)) and its positions at different moments of time, Section 6.4;
4) Approximate bundle adjustment estimating cloud of 3D coordinates of coded markers and camera calibration 6.5;
5) Matching (grouping) of all markers using the fundamental matrix equation (4), obtained from camera calibration [26];
6) Precise bundle adjustment, result – cloud of 3D coordinates of all markers and accurate camera calibration.

Next, we consider the results – reconstructed 3D point clouds obtained for various configurations of the optimization problem (18). Let's denote Brown's projection model [3] (14) as Brown; with residual stabilization (20) as Brown & $\rho_{0.05}$. For the case without accounting for tangential distortion (13) Brown & $\rho_{0.05}$ & k^0_τ. Model (15) is denoted as OpenCV[17] and, according to previous definitions, we introduce OpenCV & $\rho_{0.05}$, OpenCV & $\rho_{0.05}$ & k^0_τ.

Figure 12 presents a comparison of reconstruction from HDP and the scheme Brown & $\rho_{0.05}$. Each graph includes $k = 8$ distributions (box plots) of distance differences between corresponding point pairs of two clouds:

[17] This formulation is given in [72].

$||\mathbf{p}_o^i - \mathbf{p}_o^j||_2 - ||\mathbf{p}_{hex}^i - \mathbf{p}_{hex}^j||_2,\ \forall i, j \in [1..N_{pts}]$, where $\mathbf{p}_{hex}^{\cdot}$ – points obtained from HDP, $\mathbf{p}_o^{\cdot}$ – result of optimization (18) with model Brown & $\rho_{0.05}$. The distribution of deviations is given for a specific interval $[L_k, U_k)$ of distances between points. The red line indicates the maximum permissible error of HDP for length measurement. It should be emphasized that HDP is a certified measurement system according to the VDI / VDE 2634, part 1 standard. This guarantees volumetric measurement accuracy within the limits indicated by the red line. Thus, the model Brown & $\rho_{0.05}$ yields results within the HDP error margin.

In HDP, both scale bars were used in the bundle adjustment problem. For all results obtained by the author's software, the loss function (19) included the length residual of only the larger bar (1390mm, $N_{scales} = 1$), the 790mm bar was used for additional verification. By increasing w_{scale} in (19), zero error of the bar length can easily be achieved at the cost of projective distortion of the resulting point cloud.

Table provides a comparison of distance deviations between point pairs for two clouds similar to Fig. 12, but with the algorithm being common and data from different days. Thus, the repeatability of results – a key characteristic of any measurement system and photogrammetry in particular – can be evaluated. The scene contained two rigid objects – a steel welding table and a carbon calibration plate; repeatability is evaluated separately for each point subset. Columns σ_{table} and σ_{plate} show the root mean squared error for table and plate points respectively. The largest error in measuring the scale bar length (790mm) over two days e_{bar} is also presented.

As seen from Table, the measurement repeatability of the small calibration plate σ_{plate} is almost identical (with one exception) and does not depend on the algorithm configuration choice. For the table, the difference can be significant – the best repeatability is achieved with optimization suppressing outliers: Brown & $\rho_{0.05}$ and OpenCV & $\rho_{0.05}$ give the best results in their groups. Moreover, without stabilization in configurations Brown and OpenCV, significant errors in scale bar measurements e_{bar} are observed. The results also confirm the necessity of accounting for tangential distortion (13) in the bundle adjustment problem even for high-class optics (Canon EOS 5DS). It's interesting to note that Brown & $\rho_{0.05}$ significantly outperforms HDP in σ_{table}. In conclusion, it should be emphasized that the main components for high-precision results are careful texture feature extraction and accounting for the detection errors in the optimization problem (18).

Repeatability of distances for 3D point clouds obtained by one algorithm from data of different days (smaller is better)

Algorithm	$\sigma_{table}, \mu m$	$\sigma_{plate}, \mu m$	$e_{bar}, \mu m$
HDP	46.5	7.2	–2.9[18]
Brown	80.4	22.5	138.7
Brown & $\rho_{0.05}$	**18.0**	7.5	–6.9
Brown & $\rho_{0.05}$ & k_τ^0	41.1	7.5	3.3
OpenCV	41.8	7.6	–154.2
OpenCV & $\rho_{0.05}$	39.8	7.6	–9.0
OpenCV & $\rho_{0.05}$ & k_τ^0	41.2	7.7	4.3

8. CONCLUSION

Within this work, a broad overview of algorithms constituting the photogrammetric pipeline has been presented. Key factors determining the accuracy of the output 3D point cloud have been demonstrated. A mathematical formulation of the photogrammetric optimization problem for various camera models has been presented. In the experimental part, high accuracy indicators for the described mathematical models on real hardware and application scenario have been confirmed.

The authors express their sincere gratitude to the employees and management of LLC "Digital Assembly" (Russian Federation, St. Petersburg) for their assistance in organizing the experiment. Thanks to the professionalism of the company's engineers, it was possible to collect high-quality data for testing various hypotheses and obtain results of world leaders level in the field of photogrammetry. The authors are ready to provide the source data (images, obtained point clouds) upon a request.

FUNDING

The work was supported by the Ministry of Science and Higher Education of Russia (theme no. FSFN-2024-008Z).

REFERENCES

1. Finsterwalder, S., Die geometrischen grundlagen der photogrammetrie, *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 1897, vol. 6, pp. 1–42.
2. Brown, D., Decentering distoriton of lenses, *Photogrammetric Engineering*, 1966, vol. 32, no. 3, pp. 444–462.
3. Brown, D., Close-range camera calibration, *Photogrammetric Engineering*, 1971, vol. 37, no. 8, pp. 855–866.
4. Fryer, J. and Brown, D., Lens distortion for close-range photogrammetry, *Photogrammetric Engineering and Remote Sensing*, 1986, vol. 52, pp. 51–58.
5. Drobyshev, F.V., Soviet stereophotogrammetric instruments, *Photogrammetria*, 1960, vol. 17, pp. 60–68.
6. Dubinovskiy, V.B., et al., A rigorous method of constructing photogrammetric networks for updating topographic maps, *Izvestiya vysshikh uchebnykh zavedenij. Geodeziya i aerofotosyemka*, 1990, no. 6, pp. 68–72.
7. Alchinov, A.I., "Talka-tdv" company and the "talka" digital photogrammetric station, *Geoprofi*, 2005, no. 1, pp. 10–11.
8. Luhmann, T., Close range photogrammetry for industrial applications, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2010, vol. 65, no. 6, pp. 558–569, iSPRS Centenary Celebration Issue.
9. Bösemann, W., Industrial photogrammetry – accepted metrology tool or exotic niche, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2016, vol. XLI-B5, pp. 15–24.
10. Altukhov, V.G., Study of the accuracy of photogrammetry as a method for determining the volume of an object, *Avtomatika i Programmnaya Inzheneriya*, 2020, vol. 32, no. 2, pp. 69–74.
11. Leizea, I., Herrera, I., and Puerto, P., Calibration procedure of a multi-camera system: Process uncertainty budget, *Sensors*, 2023, vol. 23, no. 2.
12. Balanji, H.M., Turgut, A.E., and Tunc, L.T., A novel vision-based calibration framework for industrial robotic manipulators, *Robot. Comput.-Integr. Manuf.*, 2022, vol. 73, no. C.
13. Puerto, P., et al., Analyses of key variables to industrialize a multi-camera system to guide robotic arms, *Robotics*, 2023, vol. 12, pp. 10–22.
14. Li, Z., et al., A robust camera self-calibration method based on circular oblique images, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2024, vol. X-1-2024, pp. 131–136.
15. Ulrich, M., et al., Vision-guided robot calibration using photogrammetric methods, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2024, vol. 218, pp. 645–662.
16. Visilter, Y.V., et al., Image comparison by shape using diffuse morphology and diffuse correlation, *Kompyuternaya Optika*, 2015, vol. 39, no. 2, pp. 265–274.

17. Fraser, C.S., Automatic camera calibration in close-range photogrammetry, *Photogrammetric Engineering and Remote Sensing*, 2013, vol. 79, pp. 381–388.

18. Lebedev, M.A., et al., A real-time photogrammetric algorithm for sensor and synthetic image fusion with application to aviation combined vision, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2014, vol. XL-3, pp. 171–175.

19. Knyaz, V., et al., Multi-sensor data analysis for aerial image semantic segmentation and vectorization, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2024, vol. XLVIII-1-2024, pp. 291–296.

20. Kobzev, A. and Chibunichev, A., Aerial triangulation using different time images of urban areas obtained from unmanned aerial systems, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2024, vol. XLVIII-2/W5-2024, pp. 87–93.

21. Schonberger, J.L. and Frahm, J.M., Structure-from-motion revisited, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4104–4113.

22. Turki, H., Ramanan, D., and Satyanarayanan, M., Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs, 2022.

23. Lowe, D., Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision*, 2004, vol. 60, no. 2, pp. 91–110.

24. Bay, H., Tuytelaars, T., and Van Gool, L., *Computer Vision – ECCV 2006* A. Leonardis, H. Bischof, and A. Pinz, editors, Surf: Speeded up robust features, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.

25. DeTone, D., Malisiewicz, T., and Rabinovich, A., Superpoint: Self-supervised interest point detection and description, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 337–348.

26. Hartley, R. and Zisserman, A., *Multiple View Geometry in Computer Vision*, New York, NY, USA: Cambridge University Press, 2 edn., 2003.

27. Sola, J., Deray, J., and Atchuthan, D., A micro lie theory for state estimation in robotics, 2021.

28. Rieke-Zapp, D., et al., Evaluation of the geometric stability and the accuracy potential of digital cameras – comparing mechanical stabilisation versus parameterisation, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2009, vol. 64, no. 3, pp. 248–258, theme Issue: Image Analysis and Image Engineering in Close Range Photogrammetry.

29. Tareen, S.A.K. and Saleem, Z., A comparative analysis of sift, surf, kaze, akaze, orb, and brisk, *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2018, pp. 1–10.

30. Jin, Y., et al., Image matching across wide baselines: From paper to practice, *Int. J. Comput. Vision*, 2021, vol. 129, no. 2, pp. 517–547.

31. Sun, J., et al., LoFTR: Detector-free local feature matching with transformers, *CVPR*, 2021, vol. 1, pp. 8918–8927.

32. Lindenberger, P., Sarlin, P.E., and Pollefeys, M., LightGlue: Local Feature Matching at Light Speed, *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, vol. 1, 2023, pp. 17581–17592.

33. Vizilter, Y., Zheltov, S., and Lebedev, M., Image and shape comparison via morphological correlation, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2021, vol. XLIV-2/W1-2021, pp. 207–211.

34. Sarlin, P.E., et al., Superglue: Learning feature matching with graph neural networks, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4937–4946.

35. Harris, C.G. and Stephens, M.J., A combined corner and edge detector, *Alvey Vision Conference*, 1988.

36. Shi, J. and Tomasi, C., Good features to track, *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2000, vol. 600.

37. Rosten, E. and Drummond, T., *Computer Vision – ECCV 2006* A. Leonardis, H. Bischof, and A. Pinz, editors, Machine learning for high-speed corner detection, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 430–443.

38. Tyszkiewicz, M.J., Fua, P., and Trulls, E., Disk: Learning local features with policy gradient, 2020.

39. Triggs, B., et al., Bundle adjustment – a modern synthesis, *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ICCV'99, Berlin, Heidelberg: Springer-Verlag, 1999, pp. 298–372.

40. Fitzgibbon, A., Simultaneous linear estimation of multiple view geometry and lens distortion, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 1–125.

41. Zhang, Z., A flexible new technique for camera calibration, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, vol. 22, pp. 1330–1334, mSR-TR-98-71, Updated March 25, 1999.

42. Wang, J., et al., Vggsfm: Visual geometry grounded deep structure from motion, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21686–21697.

43. Gurdjos, P., Sturm, P., and Wu, Y., *Computer Vision – ECCV 2006* A. Leonardis, H. Bischof, and A. Pinz, editors, Euclidean structure from n $\geqslant$ 2 parallel circles: Theory and algorithms, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 238–252.

44. Gennery, D., Generalized camera calibration including fish-eye lenses, *International Journal of Computer Vision*, 2006, vol. 68, pp. 239–266.

45. Schops, T., et al., Why having 10,000 parameters in your camera model is better than twelve, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA: IEEE Computer Society, 2020, pp. 2532–2541.

46. Canny, J., A computational approach to edge detection, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1986, vol. PAMI-8, pp. 679–698.

47. Shen, Z., et al., Combining convex hull and directed graph for fast and accurate ellipse detection, *Graphical Models*, 2021, vol. 116, pp. 101–110.

48. Ouellet, J.N. and Hebert, P., Precise ellipse estimation without contour point extraction, *Mach. Vis. Appl.*, 2009, vol. 21, pp. 59–67.

49. Mortari, D., Junkins, J., and Samaan, M., Lost-in-space pyramid algorithm for robust star pattern recognition, *Spaceflight Mechanics 2005*, 2001, vol. 120, pp. 10–20.

50. Calonder, M., et al., *Computer Vision – ECCV 2010* K. Daniilidis, P. Maragos, and N. Paragios, editors, Brief: Binary robust independent elementary features, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 778–792.

51. Rublee, E., et al., Orb: An efficient alternative to sift or surf, *2011 International Conference on Computer Vision*, 2011, pp. 2564–2571.

52. Fernandez Alcantarilla, P., Fast explicit diffusion for accelerated features in nonlinear scale spaces, *British Machine Vision Conference (BMVC) at Bristol, UK*, 2013.

53. Schneider, C.T. and Sinnreich, K., Optical 3-d measurement systems for quality control in industry, *XVIIth ISPRS Congress Technical Commission V: Close-Range Photogrammetry and Machine Vision*, 1992, vol. 29, pp. 56–59.

54. Dos Santos Cesar, D.B., et al., An evaluation of artificial fiducial markers in underwater environments, *OCEANS 2015 – Genova*, 2015, pp. 1–6.

55. Calvet, L., et al., Detection and accurate localization of circular fiducials under highly challenging conditions, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 562–570.

56. Tushev, S., Sukhovilov, B., and Sartasov, E., Robust coded target recognition in adverse light conditions, *2018 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, 2018, pp. 1–6.

57. Lin, T.Y., et al., Feature pyramid networks for object detection, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 936–944.

58. He, K., et al., Deep residual learning for image recognition, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

59. Ronneberger, O., Fischer, P., and Brox, T., U-net: Convolutional networks for biomedical image segmentation, *Medical image computing and computer-assisted intervention – MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, Springer, 2015, pp. 234–241.

60. Caetano, T.S., et al., Learning graph matching, *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.

61. Arandjelovic, R. and Zisserman, A., Three things everyone should know to improve object retrieval, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2911–2918.

62. Rusu, R.B., Blodow, N., and Beetz, M., Fast point feature histograms (fpfh) for 3d registration, *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 3212–3217.

63. Muja, M. and Lowe, D., Fast approximate nearest neighbors with automatic algorithm configuration, *VISAPP 2009 – Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, Lisboa, Portugal, February 5–8*, vol. 1, 2009, pp. 331–340.

64. Bishop, C., *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Berlin, Heidelberg: Springer-Verlag, 2006.

65. Lv, Q., et al., Multi-probe lsh: efficient indexing for high-dimensional similarity search, *Proceedings of the 33rd International Conference on Very Large Data Bases*, VLDB '07, VLDB Endowment, 2007, pp. 950–961.

66. Nister, D. and Stewenius, H., Scalable recognition with a vocabulary tree, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 2161–2168.

67. Hartley, R. and Kang, S.B., Parameter-free radial distortion correction with centre of distortion estimation, *Proceedings of IEEE International Conference on Computer Vision. IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1834–1841.

68. Hartley, R., In defense of the eight-point algorithm, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, vol. 19, no. 6, pp. 580–593.

69. Fischler, M.A. and Bolles, R.C., Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, M.A. Fischler and O. Firschein, editors, *Readings in Computer Vision*, San Francisco (CA): Morgan Kaufmann, 1987, pp. 726–740.

70. Chum, O. and Matas, J., Matching with prosac – progressive sample consensus, *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 220–226.

71. Chum, O., Werner, T., and Matas, J., Two-view geometry estimation unaffected by a dominant plane, *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 772–779.

72. Kaehler, A. and Bradski, G., *Learning OpenCV, 2nd Edition*, O'Reilly Media, Inc., 2014.

73. Chibunichev, A., Govorov, A., and Chernyshev, V., Research of the camera calibration using series of images with common center of projection, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2019, vol. XLII-2/W18, pp. 19–22.

74. Grossberg, M. and Nayar, S., A general imaging model and a method for finding its parameters, *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2, 2001, pp. 108–115.

75. Tecklenburg, W., Luhmann, T., and Hastedt, H., Camera modelling with image-variant parameters and finite elements, *Optical 3D-Measurement Techniques V*, 2001, pp. 328–335.

76. Hughes, C., et al., Equidistant ($f\theta$) fish-eye perspective with application in distortion centre estimation, *Image and Vision Computing*, 2010, vol. 28, no. 3, pp. 538–551.

77. Bukhari, F. and Dailey, M.N., Automatic Radial Distortion Estimation from a Single Image, *Journal of Mathematical Imaging and Vision*, 2013, vol. 45, no. 1, pp. 31–45.

78. Triggs, B., Autocalibration from planar scenes, *Proceedings of the 5th European Conference on Computer Vision*, vol. 1 of *ECCV'98*, Berlin, Heidelberg: Springer-Verlag, 1998, pp. 89–105.

79. Nister, D., An efficient solution to the five-point relative pose problem, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, vol. 26, no. 6, pp. 756–770.

80. Larsson, Viktor and contributors, PoseLib – Minimal Solvers for Camera Pose Estimation, 2020.

81. Agarwal, S., Mierle, K., and The Ceres Solver Team, Ceres Solver, 2023.

82. Bradbury, J., et al., JAX: composable transformations of Python+NumPy programs, 2018.

83. Virtanen, P., et al., Scipy 1.0: Fundamental algorithms for scientific computing in python, *Nature Methods*, 2020, vol. 17, pp. 261–272.

This paper was recommended for publication by B.M. Miller, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 12, pp. 1118–1137.
© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2025.
Russian Text © The Author(s), 2025, published in Avtomatika i Telemekhanika, 2025, No. 12, pp. 42–65.

LINEAR SYSTEMS

An Algorithm for Constructing Gain Matrices in the Spectrum Assignment Problem of a Linear Control System

S. P. Zubova[*,a] **and E. V. Raetskaya**[**,b]

[*] *Voronezh State University, Voronezh, Russia*
[**] *Voronezh State University of Forestry and Technologies, Voronezh, Russia*
e-mail: [a] *spzubova@mail.ru,* [b] *raetskaya@inbox.ru*

Received February 24, 2025
Revised July 21, 2025
Accepted July 24, 2025

Abstract—This paper proposes an algorithm for constructing gain matrices in the spectrum assignment problem of a continuous-time linear dynamic control system without any constraints on the matrix coefficients of the system. The algorithm is based on constructing eigenvectors and adjoined vectors corresponding to the given eigenvalues of the corresponding matrix. An algorithm with a minimum number of simple algebraic operations is developed for their construction. As a result, to solve the above problem, a complete set of gain matrices is constructed, depending on a certain number of arbitrary scalar parameters. Cases of the uniqueness of such a matrix are determined. Illustrative examples are provided in the cases of a simple spectrum and a multiple one. Gain matrices are constructed for a dynamic system describing the operation of a multi-chamber heating furnace.

Keywords: continuous-time linear systems, state feedback, spectrum assignment problem, cascade decomposition method, algorithm for constructing a set of gain matrices

DOI: 10.7868/S1608303225120022

1. INTRODUCTION

The problem of stabilizing the programmed motion of a dynamic system is one of the most important challenges in control theory, dictated by pressing practical demands.

The solution of the stabilization problem has a long history of over 150 years and an extensive bibliography, including both purely theoretical and practical works. The history up to 2019 was described in detail in [1], with a comprehensive list of references containing 107 items.

A more general problem is also of great interest: if a linear dynamic feedback control system is described by a set of differential equations with respect to the unknown components of the state vector of the system, it is required to determine an appropriate feedback (gain matrix) under which the spectrum of the matrix coefficient at the state vector will possess definite properties. For example, it should be located in a desired region of the complex plane. In particular, if this region lies in the left half-plane of the complex plane, then the corresponding gain matrix is constructed by stabilizing the programmed motion of the control system under consideration.

In a special case, the region for locating the spectrum points can be a finite set of arbitrarily given numbers. In this case, the problem of finding the corresponding gain matrix is called pole assignment, spectrum control, or spectrum assignment in the literature.

In the monograph [2], V.I. Zubov studied a nonlinear time-varying control system and established sufficient conditions for the existence of additional components that can be introduced into the system to stabilize it; also, he provided a method for finding such additional components.

For a linear time-invariant control system of the form

$$\dot{x} = Ax + Bu, \tag{1.1}$$

where $x = x(t)$, $x \in \mathbb{R}^n$, $u = u(t)$, $u \in \mathbb{R}^m$, $\dot{x} = \frac{dx}{dt}$, $A : \mathbb{R}^n \to \mathbb{R}^n$, $B : \mathbb{R}^m \to \mathbb{R}^n$, and $t \in [t_0, t_k]$, the spectrum assignment problem consists in the following:

For arbitrarily given numbers $\{\lambda_j\}_{j=1}^n$, it is required to construct a state-feedback control law

$$u = Kx \tag{1.2}$$

under which the spectrum of the matrix $A + BK$ will coincide with $\{\lambda_j\}_{j=1}^n$, i.e.,

$$\det(A + BK - \lambda_j I) = 0. \tag{1.3}$$

In the case of complex values λ_j, they must be pairwise ordered and complex conjugate.

For a linear time-invariant system with a one-dimensional control vector (scalar control), V.I. Zubov essentially derived a sufficient condition for the solvability of the spectrum assignment problem: this is the controllability condition of the system. The spectrum was considered arbitrary, and the author passed to the spectrum located in the left half-plane of the complex plane only to stabilize the system [2, pp. 153 and 154].

The complete result on the existence of a solution of problem (1.1)–(1.3) was obtained by W.M. Wonham [3, 4, p. 79]:

Theorem 1. *Problem* (1.1)–(1.3) *is solvable if and only if the pair* (A, B) *is controllable.*

Consequently, the condition

$$\operatorname{rank}(B\, AB\, \ldots\, A^{n-1}B) = n$$

must hold.

The history of the proof of this theorem by some authors, with the aim of simplification, was presented in [1].

A practical solution of problem (1.1)–(1.3) was obtained for the case $m = 1$ as a solution of the equation

$$\det(A + BK - \lambda I) = f(\lambda),$$

where $f(\lambda)$ is a polynomial of degree n with corresponding real coefficients that contains the components of the row matrix $K = (k_1 \quad k_2 \quad \ldots \quad k_n)$. This equation is uniquely solvable with respect to k_i, $i = 1, \ldots, n$, and the solution is given by the exact Bass–Gura and Ackermann formulas [1, p. 579].

A numerical implementation of problem (1.1)–(1.3) in MATLAB, based on the full pole placement method [5], was provided in [6].

For the controllable system (1.1) with given A and B, $\operatorname{Ker} B = \{0\}$, a method for constructing a set of gain matrices was described in [7]. For this purpose, Jordan chains of vectors of the matrix $(A - \lambda I \quad B)$ for each $\lambda = \lambda_j$ were used, and additional parameterizing matrices were introduced.

The so-called cascade decomposition method for solving problem (1.1)–(1.3) without any constraints on A and B was proposed in [8]. (For brevity, it will be referred to as the cascade method.) This method allows one to establish that either system (1.1) is uncontrollable (and the matrix K cannot be constructed) or system controllability holds (in this case, the method yields a complete

manifold of all matrices K corresponding to the given A, B, and $\{\lambda_j\}_{j=1}^n$). Thus, another proof of Theorem 1 was obtained and, moreover, a method for constructing a complete set of gain matrices was developed.

Problem (1.1)–(1.3) is solved as follows [8].

The equation

$$(A + BK)v_j = \lambda_j v_j \tag{1.4}$$

is solved with respect to v_j and K; first, a set of n linearly independent vectors v_j corresponding to the eigenvalues λ_j is constructed; then, the components of the matrix K are found. The cascade method is used only to construct v_j.

However, application of the cascade method is rather computationally intensive: it involves decompositions of spaces into subspaces, projectors onto subspaces, and semi-inverse matrices.

The goal of this paper is to maximally simplify the solution of problem (1.1)–(1.3); the idea is to create an algorithm for constructing n linearly independent eigenvectors and adjoined vectors of the matrix $A + BK$ that requires only solving systems of linear algebraic equations and checking the linear independence of the resulting vectors at the last step of the algorithm, e.g., by computing determinants.

Such a design of gain matrices appreciably simplifies the computational process and creates the necessary prerequisites for developing simpler computational programs.

For the sake of comparison, note the following: Moore–Penrose pseudoinverse matrices were used in [6] to solve the applied problem (1.1)–(1.3) in MATLAB; they were computed as the limit of a sequence of certain matrices. In this case, only one matrix K is constructed, although there exists a set of such matrices [8, p. 2024] for the problem considered in [6].

In this paper, we reveal a complete set of gain matrices for a particular problem by establishing the dependence of the matrix on some numerical parameters, some being arbitrary while the others satisfying certain conditions.

The manifold of gain matrices is very useful in applications: one can select an appropriate matrix, e.g., with a smaller norm or with fewer nonzero components.

The number of steps in the algorithm for finding the eigenvectors and adjoined vectors of the matrix $A + BK$ can be determined in advance: there are exactly p of them, where $p = \min q$ and the number q is given by the condition

$$\operatorname{rank}(B\, AB\, \ldots\, A^q B) = n. \tag{1.5}$$

The algorithm proposed below is based on the cascade method [8] and represents its significant simplification. The algorithm for constructing the eigenvectors and adjoined vectors of the matrix $A + BK$ is justified in the Appendix.

As illustrative examples, we find gain matrices in the cases of a simple spectrum and a multiple one and construct gain matrices for a dynamic system describing the operation of a multi-chamber heating furnace.

2. SOLVING A LINEAR ALGEBRAIC EQUATION WITH A SINGULAR MATRIX AT THE UNKNOWN

A linear system of the form

$$Cx = y \tag{2.1}$$

with a singular matrix C can be solved with respect to x only under some constraint (condition) imposed on y :

$$Qy = 0. \tag{2.2}$$

The solution x may be nonunique, i.e.,

$$x = C^{-}y + z, \tag{2.3}$$

where C^{-} is some matrix, and z is an arbitrary vector such that $Cz = 0$.

The constraint (2.2) will be called the well-posedness condition for system (2.1), and the vector (2.3) will be called the solution of system (2.1).

For example, let $C = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 2 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}$, i.e.,

$$\begin{gathered} x_1 + x_3 = y_1, \\ x_2 = y_2, \\ 2x_1 + 2x_3 = y_3, \\ 0 = y_4; \end{gathered} \tag{2.4}$$

then $x = \begin{pmatrix} y_1 \\ y_2 \\ 0 \end{pmatrix} + \begin{pmatrix} -x_3 \\ 0 \\ x_3 \end{pmatrix}$ is the solution of system (2.4) with an arbitrary value x_3 under the conditions

$$\begin{gathered} -2y_1 + y_3 = 0, \\ y_4 = 0. \end{gathered}$$

Consequently, $Qy = 0$, where $Q = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$. In this case, $z = \begin{pmatrix} -x_3 \\ 0 \\ x_3 \end{pmatrix}$.

System (2.4) can also be solved as follows:

$x = \begin{pmatrix} 0 \\ y_2 \\ y_1 \end{pmatrix} + \begin{pmatrix} x_1 \\ 0 \\ -x_1 \end{pmatrix}$ for any x_1 under the well-posedness condition

$$\begin{gathered} y_1 - \frac{1}{2}y_3 = 0, \\ y_4 = 0. \end{gathered}$$

But the solution set and the well-posedness condition are equivalent in this case; hence, the particular form of the solution with the corresponding well-posedness condition does not matter in practice.

3. THE ALGORITHM FOR CONSTRUCTING A GAIN MATRIX IN THE CASE OF A SIMPLE SPECTRUM

3.1. Preliminary Transformations

A_1. In the case of simple eigenvalues λ_j, $j = 1, \ldots, n$, in equation (1.4)

$$(A + BK)v(\lambda_j) = \lambda_j v(\lambda_j), \tag{3.1}$$

introduce the designation

$$Kv(\lambda_j) = f(\lambda_j). \tag{3.2}$$

A_2. Write equation (3.1) as

$$Bf(\lambda_j) = (\lambda I - A)v(\lambda_j). \tag{3.3}$$

A_3. Solve equation (3.3) with respect to $f(\lambda_j)$:

$$f(\lambda_j) = B^-(\lambda_j I - A)v(\lambda_j) + z(\lambda_j) \tag{3.4}$$

with an arbitrary vector $z = z(\lambda_j)$ under the well-posedness condition

$$Q(\lambda_j I - A)v(\lambda_j) = 0. \tag{3.5}$$

As a result of preliminary transformations, the relation (3.1) is replaced by an equivalent system of the three relations (3.2), (3.4), and (3.5).

3.2. The Algorithm for Solving Equation (3.5) *with Respect to the Vector* $v = v(\lambda_j)$

Forward pass

The first step. A_{11}. In equation (3.5) with the subscript j omitted for now, denote by $w_1 = w_1(\lambda)$ the coefficient at λ (i.e., Qv); then (3.5) is equivalent to the system

$$\begin{cases} Qv = w_1 \\ QAv = \lambda w_1. \end{cases} \tag{3.6}$$

A_{12}. Obtain the solution v of system (3.6) in the form

$$v = G_1(\lambda, w_1) + z_1 \tag{3.7}$$

with some matrix G_1 and an arbitrary vector $z_1 = z_1(\lambda)$ under the well-posedness condition

$$Q_1(\lambda, w_1) = 0. \tag{3.8}$$

The second step. A_{21}. In the relation (3.8), denote by $w_2 = w_2(\lambda)$ the coefficient at λ (i.e., $Q_1 w_1$); then (3.8) is equivalent to the system

$$\begin{cases} Q_1 w_1 = w_2 \\ Q_{11} w_1 = \lambda w_2 \end{cases} \tag{3.9}$$

with some matrix Q_{11}.

A_{22}. Obtain the solution of this system in the form

$$w_1 = G_2(\lambda, w_2) + z_2$$

with some matrix G_2 and an arbitrary vector $z_2 = z_2(\lambda)$ under the well-posedness condition

$$Q_2(\lambda, w_2) = 0.$$

And so on ... .

The ith step. A_{i1}. In the well-posedness condition identified at the $(i-1)$th step,

$$Q_{i-1}(\lambda, w_{i-1}) = 0, \qquad i = 2, 3, \ldots, \tag{3.10}$$

denote by $w_i = w_i(\lambda)$ the coefficient at λ; as a result, (3.10) is equivalent to the system

$$\begin{cases} Q_{i-1}w_{i-1} = w_i \\ Q_{i-11}w_{i-1} = \lambda w_i. \end{cases} \tag{3.11}$$

A_{i2}. Obtain the solution of this system in the form

$$w_{i-1} = G_i(\lambda, w_i) + z_i \tag{3.12}$$

with some matrix G_i and an arbitrary vector $z_i = z_i(\lambda)$ under the well-posedness condition

$$Q_i(\lambda, w_i) = 0.$$

And so on ... .

The pth step. A_{p2}. As proved in [8], if system (1.1) is completely controllable, then for $i = p$ (see (1.5)) system (3.11) is solvable with respect to w_{p-1} without any well-posedness condition:

$$w_{p-1} = G_p(\lambda, w_p) + z_p \tag{3.13}$$

with some matrix G_p and an arbitrary vector $z_p = z_p(\lambda)$. The element $w_p = w_p(\lambda)$ is also arbitrary.

Construct w_{p-1}.

3.3. The Algorithm for Constructing the Vector $v(\lambda)$

Backward pass

B_1. Substitute the vector w_{p-1} constructed at step A_p into (3.12) with $i = p-1$, thereby determining the vector w_{p-2}.

B_2. Substitute the vector w_{p-2} constructed at the previous step into (3.12) with $i = p-2$, thereby determining the vector w_{p-3}.

And so on ... .

B_{p-2}. Substitute the vector w_2 constructed at step B_{p-3} into (3.12) with $i = 2$, thereby determining the vector w_1.

B_{p-1}. Substitute the vector w_1 constructed at step B_{p-2} into (3.7), thereby determining the vector v :

$$v = G\big(w_p(\lambda), z_p(\lambda), z_{p-1}(\lambda), \ldots, z_1(\lambda)\big) \tag{3.14}$$

with some matrix G and arbitrary vector functions $w_p(\lambda)$ and $z_s(\lambda)$, $s = 1, \ldots, p$.

3.4. Constructing the System of n Linearly Independent Vectors $v(\lambda_j)$

If system (1.1) is completely controllable, then for any set $\{\lambda_j\}_{j=1}^n$ there exist elements $w_p(\lambda), z_p(\lambda), z_{p-1}(\lambda), \ldots, z_1(\lambda)$ such that the vectors $v(\lambda_j)$ (3.14) with the above $w_{p-1}(\lambda), z_p(\lambda), z_{p-1}(\lambda), \ldots, z_1(\lambda)$ are linearly independent [8].

The eigenvalues λ_j can also be complex, pairwise ordered.

If system (1.1) is not completely controllable, then n linearly independent vectors $v(\lambda_j)$ cannot be constructed, and the same applies to the matrix K [8].

Here, one should construct a vector $v(\lambda_j)$ such that the vectors $v(\lambda_1)$, $v(\lambda_2)$, ..., $v(\lambda_n)$ are linearly independent for any set $\{\lambda_j\}_{j=1}^n$.

3.5. The Algorithm for Constructing a Gain Matrix in the Case of a Simple Spectrum

C_1. Substitute the vectors $v_j = v(\lambda_j)$ constructed in Section 3.4 into formula (3.4), thereby determining $f_j = f_j(\lambda_j)$, $j = 1, \dots, n$.

C_2. Substitute v_j and f_j into (3.2) for each value of j. As a result, n linear equations with unknowns $k_{ij}, i = 1, \dots, m$ (the components of matrix K) are obtained.

C_3. Extract from the above equations those containing the components k_{1j} as the unknowns, thereby forming a system of linear algebraic equations with the principal determinant $\triangle_1$ made up of the components of the linearly independent vectors $v(\lambda_j)$; hence, $\triangle_1 \neq 0$.

C_4. Solve the system obtained at step C_3 with respect to k_{1j} (the components of the first row of the matrix K).

C_5. Repeat steps C_3 and C_4 for $i = 2, 3, \dots, m$, thereby determining the components of the ith rows of the matrix K.

Example 1. Let $A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$, and $K = (k_{ij})$, $i = 1, 2$, $j = \overline{1, 3}$.

We solve the equation $Bf = (\lambda I - A)v$, where $f = (f_1, f_2)$ and $v = (v_1, v_2, v_3)$, with respect to f (step A_3):

$$\begin{aligned} f_1 + f_2 &= \lambda v_1, \\ f_1 + f_2 &= -v_1 + \lambda v_2, \\ f_1 + f_2 &= -v_2 + \lambda v_3. \end{aligned}$$

Consequently,

$$\begin{aligned} f_1 &= \lambda v_1 - z(\lambda), \\ f_2 &= z(\lambda) \end{aligned} \tag{3.15}$$

with an arbitrary vector $z(\lambda)$ ((3.15) is (3.4)) under the well-posedness condition (3.5):

$$\begin{aligned} \lambda v_1 &= -v_1 + \lambda v_2, \\ \lambda v_1 &= -v_2 + \lambda v_3, \end{aligned}$$

or

$$\begin{cases} \lambda(v_1 - v_2) = -v_1 \\ \lambda(v_1 - v_3) = -v_2. \end{cases} \tag{3.16}$$

Following step A_{11}, we denote

$$\begin{aligned} v_1 - v_2 &= w_{11}, \\ v_1 - v_3 &= w_{12}. \end{aligned} \tag{3.17}$$

Then (3.16) becomes

$$\begin{cases} -v_1 = \lambda w_{11} \\ -v_2 = \lambda w_{12}. \end{cases} \tag{3.18}$$

The system composed of (3.17) and (3.18) is system (3.6).

We solve system (3.17), (3.18) with respect to v_1, v_2, and v_3 (see step A_{12}, formula (3.7)):

$$\begin{aligned} v_1 &= -\lambda w_{11}, \\ v_2 &= -\lambda w_{12}, \\ v_3 &= -\lambda w_{11} - w_{12} \end{aligned} \tag{3.19}$$

(here $z_1 = 0$) under the well-posedness condition of system (3.17), (3.18): $-\lambda w_{11} + \lambda w_{12} = w_{11}$. As a result, we arrive at the relation (3.8):

$$\lambda(-w_{11} + w_{12}) = w_{11}.$$

Following step A_{21}, we denote

$$-w_{11} + w_{12} = w_2;$$

then

$$w_{11} = \lambda w_2.$$

The last two equalities are system (3.9).

We solve this system with respect to w_{11} and w_{12} (see step A_{22}) :

$$\begin{aligned} &w_{11} = \lambda w_2, \\ &w_{12} = (1+\lambda) w_2. \end{aligned} \tag{3.20}$$

A solution exists for any w_2 without any well-posedness condition; hence, step A_{22} is step A_{p2} with $p = 2$.

Substituting (3.20) into (3.19) yields formula (3.14):

$$v(\lambda) = (\lambda^2,\, \lambda + \lambda^2,\, 1 + \lambda + \lambda^2). \tag{3.21}$$

For different values of λ_1, λ_2, and λ_3, the vectors $v(\lambda_j) = (\lambda_j^2,\, \lambda_j + \lambda_j^2,\, 1 + \lambda_j + \lambda_j^2)$, $j = 1, 2, 3$, are linearly independent since the determinant $\triangle_1$ made up of the components of the above vectors for each j reduces to a Vandermonde determinant [9, p. 33], which is nonzero.

Then $v(\lambda_j)$ are substituted into (3.15) (step C_1) and, together with f_1 and f_2, are substituted into (3.2) (step C_2).

Two systems are formed from the resulting equalities.

For $i = 1$ (see step C_3),

$$\begin{cases} \lambda_1^2 k_{11} + (\lambda_1 + \lambda_1^2) k_{12} + (1 + \lambda_1 + \lambda_1^2) k_{13} = \lambda_1^3 - z(\lambda_1) \\ \lambda_2^2 k_{11} + (\lambda_2 + \lambda_2^2) k_{12} + (1 + \lambda_2 + \lambda_2^2) k_{13} = \lambda_2^3 - z(\lambda_2) \\ \lambda_3^2 k_{11} + (\lambda_3 + \lambda_3^2) k_{12} + (1 + \lambda_3 + \lambda_3^2) k_{13} = \lambda_3^3 - z(\lambda_3); \end{cases}$$

for $i = 2$ (step C_4),

$$\begin{cases} \lambda_1^2 k_{21} + (\lambda_1 + \lambda_1^2) k_{22} + (1 + \lambda_1 + \lambda_1^2) k_{23} = z(\lambda_1) \\ \lambda_2^2 k_{21} + (\lambda_2 + \lambda_2^2) k_{22} + (1 + \lambda_2 + \lambda_2^2) k_{23} = z(\lambda_2) \\ \lambda_3^2 k_{21} + (\lambda_3 + \lambda_3^2) k_{22} + (1 + \lambda_3 + \lambda_3^2) k_{23} = z(\lambda_3). \end{cases}$$

And the components of the matrix K are determined accordingly.

The nonunique form of the matrix K is ensured by the arbitrary function $z(\lambda)$.

4. THE ALGORITHM FOR CONSTRUCTING A GAIN MATRIX IN THE CASE OF A MULTIPLE SPECTRUM

For example, let λ_1 have multiplicity k, $k \leqslant n$, and let $\lambda_{k+1}, \lambda_{k+2}, \ldots, \lambda_n$ be distinct arbitrary numbers not equal to λ_1.

The vectors $v_1 = v(\lambda_1)$, $v_{k+1} = v(\lambda_{k+1}), \ldots, v_n = v(\lambda_n)$ are constructed as described in Section 3; as $k-1$ vectors $\tau_j = \tau_j(\lambda)$ we take the vectors $\tau_1, \tau_2, \ldots, \tau_{k-1}$ of the matrix $A + BK$ that are adjoined to v_1 and correspond to the eigenvalue λ_1 :

$$(A + BK)\tau_j = \lambda_1 \tau_j + \tau_{j-1}, \qquad j = 1, \ldots, k-1, \qquad \tau_0 = v_1.$$

4.1. Preliminary Transformations

D_1. Introduce the designation

$$K\tau_j = g_j. \tag{4.1}$$

D_2. For each $j = 1, \dots, k-1$, solve the equations $Bg_j = (\lambda_1 I - A)\tau_j + \tau_{j-1}$ with respect to g_j :

$$g_j = B^-((\lambda_1 I - A)\tau_j + \tau_{j-1}) + z_j, \tag{4.2}$$

with an arbitrary vector z_j such that $Bz_j = 0$, under the well-posedness conditions

$$Q((\lambda_1 I - A)\tau_j + \tau_{j-1}) = 0. \tag{4.3}$$

4.2. The Algorithm for Solving Equations (4.3) *with Respect to* τ_j

Forward pass

The first step. D_{11}. In (4.3), denote by w_1^j, $j = 1, \dots, k-1$, the coefficients at λ_1 (i.e., $Q\tau_j$) :

$$Q\tau_j = w_1^j; \tag{4.4}$$

as a result, (4.3) takes the form

$$QA\tau_j = \lambda_1 w_1^j + Q\tau_{j-1}, \tag{4.5}$$

where $\tau_0 = v_1$ is a known function (see Section 3).

D_{12}. Find the solutions of systems (4.4) and (4.5) in the form

$$\tau_j = G_1^j(\lambda_1, w_1^j) + z_1^j \tag{4.6}$$

with some matrices G_1^j and arbitrary vectors z_1^j under the well-posedness conditions

$$Q_1^j(\lambda_1, w_p, w_1^j) = 0.$$

The ith step. D_{i1}. In the relations obtained at the $(i-1)$th step of the algorithm,

$$Q_{i-1}^j(\lambda_1, w_p, w_{i-1}^j) = 0, \tag{4.7}$$

denote by w_i^j the coefficients at λ_1 :

$$Q_{i-1}^j w_{i-1}^j = w_i^j. \tag{4.8}$$

The expressions (4.7) become

$$Q_{i-11}^j w_{i-1}^j = \lambda_1 w_i^j. \tag{4.9}$$

D_{i2}. Solve systems (4.8) and (4.9) with respect to w_{i-1}^j :

$$w_{i-1}^j = G_i^j(\lambda_1, w_i^j) + z_i^j \tag{4.10}$$

with some matrices G_i^j and arbitrary vectors z_i^j under the well-posedness conditions

$$Q_i^j(\lambda_1, w_p, w_i^j) = 0, \tag{4.11}$$

and so on . . . .

The pth step. D_{p2}. As proved in [8], if system (1.1) is completely controllable, then for each $j = 1, \dots, k-1$ and $i = p$, equations (4.11) are solvable with respect to w_{p-1}^j :

$$w_{p-1}^j = G_p^j(\lambda_1, w_p, w_p^j) + z_p^j, \tag{4.12}$$

where G_p^j are some matrices and z_p^j, w_p, and w_p^j are arbitrary vectors. In this case, well-posedness conditions are absent.

Form w_{p-1}^j.

Backward pass

E_1. For each $j = 1, \dots, k-1$, substitute the vectors w_{p-1}^j (4.12), with as yet arbitrary w_p^j and r_p^j, into (4.10) with $i = p-1$, thereby determining the vectors w_{p-2}^j.

E_2. Substitute w_{p-2}^j into (4.10) with $i = p-2$. Thereby, the vectors w_{p-3}^j are obtained.

And so on . . . .

E_{p-2}. Substitute w_2^j (found at step E_{p-3}) into (4.10) with $i = 2$. Thereby, the vectors w_1^j are obtained.

E_{p-1}. Determine τ_j from (4.6):

$$\tau_j = G(w_p, w_p^j, z_p^j, z_{p-1}^j, \dots, z_1^j), \quad j = 1, \dots, k-1. \tag{4.13}$$

4.3. Constructing the System of n Linearly Independent Vectors in the Case of a Multiple Spectrum

According to [8], in the case of the complete controllability of system (1.1), there exist vectors $w_p, w_p^j, z_p^j, z_{p-1}^j, \dots, z_1^j$ such that the vectors $v_1, \tau_1, \dots, \tau_{k-1}, v_{k+1}, \dots, v_n$ are linearly independent. Here, the vectors $\tau_1, \dots, \tau_{k-1}$ are constructed using formulas (4.13), and $v_j = v(\lambda_j)$, $j = 1, k+1, \dots, n$, using formulas (3.14).

If system (1.1) is not completely controllable, then n linearly independent vectors $v_1, \tau_1, \dots, \tau_{k-1}, v_{k+1}, \dots, v_n$ cannot be constructed for arbitrary values $\lambda_1, \lambda_{k+1}, \dots, \lambda_n$, where λ_1 is an eigenvalue of multiplicity k [8].

Select elements $w_p, w_p^j, z_p^j, z_{p-1}^j, \dots, z_1^j$ such that the vectors $v_1, \tau_1, \dots, \tau_{k-1}, v_{k+1}, \dots, v_n$ are linearly independent.

4.4. The Algorithm for Constructing a Gain Matrix in the Case of a Multiple Spectrum

Let λ_1 be an eigenvalue of multiplicity k of the matrix $A + BK$, and let the other $\lambda_{k+1}, \dots, \lambda_n$ be simple eigenvalues.

G_1. For each $j = 1, k+1, \dots, n$, substitute the vectors $v_j = v(\lambda_j)$ constructed in Section 3 into (3.4), thereby determining f_j.

G_2. Substitute f_j and v_j into (3.2):

$$Kv(\lambda_j) = f(\lambda_j), \quad j = 1, k+1, \dots, n. \tag{4.14}$$

G_3. Substitute the vectors $\tau_1, \tau_2, \dots, \tau_{k-1}$ obtained at step 4.3 into formulas (4.2), thereby determining the vectors g_j.

G_4. Substitute g_j and τ_j into (4.1):

$$K\tau_j = g_j, \quad j = 1, \dots, k. \tag{4.15}$$

G_5. From equalities (4.14) and (4.15) extract those containing the components k_{1j} of the first row of the matrix K, $j = 1, \dots, n$. Thereby, a linear algebraic system with the unknowns k_{1i} is formed, with the principal determinant $\triangle_2$ made up of the components of the linearly independent vectors $v_1, \tau_1, \dots, \tau_{k-1}, v_{k+1}, \dots, v_n$; hence, $\triangle_2 \neq 0$.

G_6. Solve the system constructed at the previous step, thereby obtaining the components of the first row of the matrix K.

G_7. Repeat steps G_5 and G_6 for $i = 2, 3, \dots, m$, thereby determining the components of the ith rows of matrix K.

Example 2. Let A and B be the matrices from Example 1, and let $\lambda_1 = \lambda_2 = \lambda_3$. It is required to construct three linearly independent vectors v_1, τ_1, τ_2 such that

$$(A + BK)v_1 = \lambda_1 v_1, \tag{4.16}$$

$$(A + BK)\tau_1 = \lambda_1 \tau_1 + v_1, \tag{4.17}$$

$$(A + BK)\tau_2 = \lambda_1 \tau_2 + \tau_1. \tag{4.18}$$

The vector v_1 has been constructed in Section 3: $v_1 = (\lambda_1^2, \lambda_1 + \lambda_1^2, 1 + \lambda_1 + \lambda_1^2)$ (formula (3.21).

To construct $\tau_1 = (\tau_{11}, \tau_{12})$, in equation (4.17) we introduce the designation (equality (4.1) with $j = 1$)

$$K\tau_1 = g_1, \quad g_1 = (g_{11}, g_{12}). \tag{4.19}$$

This equation becomes

$$Bg_1 = (\lambda_1 I - A)\tau_1 + v_1;$$

therefore,

$$\begin{aligned} g_{11} &= \lambda_1 \tau_{11} + \lambda_1^2 - z_1^1, \\ g_{12} &= z_1^1 \end{aligned} \tag{4.20}$$

(this is (4.2) with $j = 1$) with an arbitrary value z_1^1 under the well-posedness condition ((4.3) with $j = 1$)

$$\begin{cases} \lambda_1(\tau_{11} - \tau_{12} - 1) = -\tau_{11} \\ \lambda_1(\tau_{11} - \tau_{13} - 1) = -\tau_{12} + 1. \end{cases}$$

Next, the coefficients at λ_1 in the last two relations are denoted as the new unknowns (see step D_{11}), e.g.,

$$\begin{aligned} \tau_{11} - \tau_{12} - 1 &= a, \\ \tau_{11} - \tau_{13} - 1 &= b. \end{aligned}$$

(Here, a and b are used instead of w_{11}^1 and w_{12}^1 for the sake of simple notation, see (4.4).) The further actions are performed as described in subsection 4.2. As a result, the vector $\tau_1 = (2\lambda_1, 1 + 2\lambda_1, 1 + 2\lambda_1)$ is determined.

Next (see subsection 4.1, $j = 2$), with the designations

$$K\tau_2 = g_2, \quad g_2 = (g_{21}, g_{22}), \tag{4.21}$$

equation (4.18) takes the form

$$Bg_2 = (\lambda \cdot I - A)\tau_2 + \tau_1. \tag{4.22}$$

From this equation we find

$$\begin{aligned} g_{21} &= \lambda_1 \tau_{21} + 2\lambda_1 - z_2, \\ g_{22} &= z_2 \end{aligned} \tag{4.23}$$

with an arbitrary value z_2 under the condition

$$\begin{cases} \lambda_1(\tau_{21} - \tau_{22}) = -\tau_{21} + 1 \\ \lambda_1(\tau_{21} - \tau_{23}) = -\tau_{22} + 1. \end{cases}$$

Continuing to solve the last system by the method described in subsection 4.4, we get

$$\tau_2 = (1, 1, 1).$$

As is easily verified, the determinant $\triangle_2$ made up of the components of the vectors v_1, τ_1, τ_2 is nonzero.

Next, using formulas (4.20) and (4.23), we determine $g_1 = (3\lambda_1^2 - z_1^1, z_1^1)$ and $g_2 = (3\lambda^2 - z_2, z_2)$ with arbitrary values z_1^1 and z_2. The vector f is given by (3.15): $f = (\lambda_1^3 - z, z)$ with an arbitrary value z.

Finally, a system consisting of the equations $Kv_1 = f_1$, (4.19), and (4.21) is formed.

As a result, two systems are obtained:

$$\begin{cases} \lambda_1^2 k_{11} + (\lambda_1 + \lambda_1^2)k_{12} + (1 + \lambda_1 + \lambda_1^2)k_{13} = \lambda_1^3 - z \\ 2\lambda_1^2 k_{11} + (1 + 2\lambda_1)k_{12} + (1 + 2\lambda_1)k_{13} = 3\lambda_1^2 - z_1^1 \\ k_{11} + k_{12} + k_{13} = 3\lambda_1 - z_2 \end{cases}$$

and

$$\begin{cases} \lambda_1^2 k_{21} + (\lambda_1 + \lambda_1^2)k_{22} + (1 + \lambda_1 + \lambda_1^2)k_{23} = z \\ 2\lambda_1 k_{21} + (1 + 2\lambda_1)k_{22} + (1 + 2\lambda_1)k_{23} = z_1^1 \\ k_{21} + k_{22} + k_{23} = z_2. \end{cases}$$

They yield the components k_{ij}, $i = 1, 2, 3$, $j = 1, 2, 3$, of the matrix K in a nonunique way, depending on arbitrary values z, z_1^1, and z_2.

In a special case where $\lambda = -1$ is a triple eigenvalue of the matrix $A + BK$, we have $K = \begin{pmatrix} c_1 & -2 + c_2 & -1 - c_3 \\ -c_1 & -c_2 & c_3 \end{pmatrix}$, where c_1, c_2, and c_3 are linear combinations of arbitrary values z, z_1^1, and z_2, therefore being arbitrary as well.

Direct verification shows that, for such K,

$$\det(A + BK - \lambda I) = -(\lambda + 1)^3.$$

5. ON THE NONUNIQUENESS OF THE GAIN MATRIX

The nonuniqueness of K for given A, B, and $\{\lambda_j\}_{j=1}^n$ arises if $\operatorname{Ker} B \neq \{0\}$, i.e., when the equation $Bf = h$ with a fixed vector $h \in \mathbb{R}^n$ has a nonunique solution $f = B^- h + z$ with an arbitrary vector $z \in \mathbb{R}^m$.

Also, the nonuniqueness of K arises if at least one of systems (3.11) or systems (4.8), (4.9) with $i = 1, \ldots, p$ has a nonunique solution w_{i-1} or w_{i-1}^j, $j = 1, \ldots, k$, where k is the multiplicity of λ_s.

However, if $\operatorname{Ker} B = \{0\}$ and all systems (3.11) or (4.8), (4.9) are uniquely solvable, then K has a unique form; see Example 2 in [8].

Note that the higher the multiplicity of an eigenvalue is, the more arbitrary parameters K may contain.

For example, a unique matrix K was constructed in [6] for system (1.1) with the matrices $A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 5 & 0 & 0 \\ 7 & 0 & 0 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the eigenvalue $\lambda = -1$ of multiplicity 4.

However, for this case, the algorithm proposed here yields a matrix K depending on eight parameters, $K = K(c_1, c_2, \ldots, c_8)$, connected only by the condition

$$\begin{vmatrix} c_1 & c_2 & 0 & 0 \\ c_3 & c_4 & c_1 & c_2 \\ c_5 & c_6 & c_3 & c_4 \\ c_7 & c_8 & c_5 & c_6 \end{vmatrix} \neq 0.$$

This condition arises due to the linear independence of the vectors v_1, τ_1, τ_2, and τ_3 [8].

For a particular dynamic system of the form (1.1) and a given arbitrary set $\{\lambda_j\}_{j=1}^n$, the method proposed in this paper constructs precisely the **complete** set of gain matrices. This conclusion follows from the equivalence of the transformations applied.

6. AN ILLUSTRATIVE EXAMPLE: GAIN MATRICES FOR STABILIZING THE OPERATION OF A MULTI-CHAMBER HEATING FURNACE

The dynamic model and operation scheme of three furnaces for heating three chambers were presented in [10].

Consider the case of three furnaces and five chambers:

$$\begin{aligned} \dot{x}_1 &= 2ax_1 + bu_1 + cu_2, \\ \dot{x}_2 &= ax_2 + cu_2 + du_3, \\ \dot{x}_3 &= ax_3 + bu_1 + 2cu_2 + du_3, \\ \dot{x}_4 &= 3ax_4 + 2cu_2 + 2du_3, \\ \dot{x}_5 &= 3ax_5 + 2bu_1 + 3cu_2 + du_3, \end{aligned} \tag{6.1}$$

where $a \leqslant -\frac{1}{10}$, $b = \frac{1}{4}$, $c = \frac{1}{9}$, and $d = \frac{1}{5}$. In the case of the unreachability of a given initial temperature $x_i(t_0) = x_{0i}$ in the ith chambers, $i = 1, 2, \ldots, 5$, it is required to correct the program control (fuel supply) $u_r = u_r(t)$ of the rth furnaces, $r = 1, 2, 3$, so that at a specified time instant $t = t_k$ the temperature $x_i(t_k)$ in the chambers is close to a desired one x_{ki}. For this purpose, a stabilizing control law with a gain matrix K can be used to make the state of system (6.1) exponentially tending to the program state.

Here, $A = \begin{pmatrix} 2a & 0 & 0 & 0 & 0 \\ 0 & a & 0 & 0 & 0 \\ 0 & 0 & a & 0 & 0 \\ 0 & 0 & 0 & 3a & 0 \\ 0 & 0 & 0 & 0 & 3a \end{pmatrix}$ and $B = \begin{pmatrix} 1/4 & 1/9 & 0 \\ 0 & 1/9 & 1/5 \\ 1/4 & 2/9 & 1/5 \\ 0 & 2/9 & 2/5 \\ 2/4 & 3/9 & 1/5 \end{pmatrix}$.

The components of the matrix $K = (k_{ij})$ are found from the equation $Kv = f$, where $f = f(\lambda)$ and $v = (v_1, \ldots, v_5)$, and $v_i = v_i(\lambda)$ are obtained from the equation $Bf = (\lambda I - A)v$, or

$$\begin{aligned} f_1 + f_2 &= (\lambda - 2a)v_1, \\ f_2 + f_3 &= (\lambda - a)v_2, \\ f_1 + 2f_2 + f_3 &= (\lambda - a)v_3, \\ 2f_2 + 2f_3 &= (\lambda - 3a)v_4, \\ 2f_1 + 3f_2 + f_3 &= (\lambda - 3a)v_5, \end{aligned} \tag{6.2}$$

where $f = (4f_1, 9f_2, 5f_3)$ to get rid of fractions in the matrix B.

The solution of system (6.2) is

$$\begin{aligned} f_1 &= (\lambda - 2a)v_1 - z, \\ f_2 &= z, \\ f_3 &= (\lambda - a)v_2 - z \end{aligned} \tag{6.3}$$

(formula (3.4)) with an arbitrary value z under conditions (3.5):

$$\begin{gathered}2(\lambda-a)v_2=(\lambda-3a)v_4,\\(\lambda-2a)v_1+(\lambda-a)v_2=(\lambda-a)v_3,\\2(\lambda-2a)v_1+(\lambda-a)v_2=(\lambda-3a)v_5,\end{gathered}$$

or

$$\begin{gathered}\lambda(2v_2-v_4)=2av_2-3av_4,\\\lambda(v_1+v_2-v_3)=2av_1+av_2-av_3,\\\lambda(2v_1+v_2-v_5)=4av_1+av_2-3av_5.\end{gathered}\tag{6.4}$$

Constructing the vector $v(\lambda)$

The first step. We introduce the designations (see step A_{11})

$$\begin{gathered}2v_2-v_4=aw_{11},\\v_1+v_2-v_3=aw_{12},\\2v_1+v_2-v_5=aw_{13}.\end{gathered}\tag{6.5}$$

Then from (6.4) it follows that

$$\begin{gathered}2v_2-3v_4=\lambda w_{11},\\2v_1+v_2-v_3=\lambda w_{12},\\4v_1+v_2-3v_5=\lambda w_{13}.\end{gathered}\tag{6.6}$$

$\mathbf{A_{12}}$. System (6.5), (6.6), which contains six equations and five unknowns, is used to find v_j (step A_{12}) :

$$\begin{aligned}v_1&=(\lambda-a)w_{12},\\v_2&=-\frac{1}{4}(\lambda-3a)w_{11},\\v_3&=-\frac{1}{4}(\lambda-3a)w_{11}+(\lambda-2a)w_{12},\\v_4&=-\frac{1}{2}(\lambda-a)w_{11},\\v_5&=-\frac{1}{4}(\lambda-3a)w_{11}+2(\lambda-a)w_{12}-aw_{13}\end{aligned}\tag{6.7}$$

(this is formula (3.7)) under the well-posedness condition of this system:

$$(\lambda-3a)w_{11}-4(\lambda-a)w_{12}-2(\lambda-3a)w_{13}=0;$$

hence,

$$\lambda(w_{11}-4w_{12}-2w_{13})=a(3w_{11}-4w_{12}-6w_{13}).$$

The second step. **$\mathbf{A_{21}}$**. Denoting (see step A_{21})

$$w_{11}-4w_{12}-2w_{13}=aw_{21},\tag{6.8}$$

we have

$$3w_{11}-4w_{12}-6w_{13}=\lambda w_{21}.\tag{6.9}$$

A_{22}. System (6.8), (6.9) is solved with respect to w_{1i} (step A_{22}) :

$$\begin{aligned} w_{11} &= \frac{1}{2}(\lambda - a)w_{21} + 2w_{13}, \\ w_{12} &= \frac{1}{8}(\lambda - 3a)w_{21} \end{aligned} \tag{6.10}$$

for any w_{13} and w_{21}. There is no well-posedness condition; consequently, $p = 2$.

Next, we substitute the values w_{11} and w_{12} (6.10) into formulas (6.7):

$$\begin{aligned} v_1 &= \frac{1}{8}(\lambda - a)(\lambda - 3a)w_{21}, \\ v_2 &= -\frac{1}{2}(\lambda - 3a)w_{13} - \frac{1}{8}(\lambda - a)(\lambda - 3a)w_{21}, \\ v_3 &= -\frac{1}{2}(\lambda - 3a)w_{13} - \frac{1}{8}a(\lambda - 3a)w_{21}, \\ v_4 &= -(\lambda - a)w_{13} - \frac{1}{4}(\lambda - a)^2 w_{21}, \\ v_5 &= -\frac{1}{2}(\lambda - a)w_{13} + \frac{1}{8}(\lambda - a)(\lambda - 3a)w_{21} \end{aligned} \tag{6.11}$$

(this is (3.14) for $p = 2$)) with any $w_{13}(\lambda)$ and $w_{21}(\lambda)$ such that $v(\lambda_j)$ are linearly independent:

$$\det(v(\lambda_1)\ v(\lambda_2)\ v(\lambda_3)\ v(\lambda_4)\ v(\lambda_5)) \neq 0.$$

In particular, for $w_{13} = -2$ and $w_{21} = 8\lambda^2$,

$$v(\lambda) = \begin{pmatrix} (\lambda - a)(\lambda - 3a)\lambda^2 \\ (\lambda - 3a)\big(1 - (\lambda - a)\lambda^2\big) \\ (\lambda - 3a)\big(1 - a\lambda^2\big) \\ 2(\lambda - a)\big(1 - (\lambda - a)\lambda^2\big) \\ (\lambda - a)\big(1 + (\lambda - 3a)\lambda^2\big) \end{pmatrix}. \tag{6.12}$$

As is easily verified, the determinant $\triangle$ made up of the components of this vector for different values of $\lambda = \lambda_j$, $j = 1, 2, \ldots, 5$, by transformations reduces to the Vandermonde determinant $W(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5)$ and is therefore nonzero.

Thus, the vectors $v(\lambda_j)$, $j = 1, 2, \ldots, 5$, are linearly independent.

Constructing the gain matrix in the case of a simple spectrum

For each $\lambda = \lambda_j$, $j = 1, 2, \ldots, 5$, we substitute the values $v(\lambda_j)$ (6.12) and $f(\lambda_j)$ (6.3) into the relation $Kv(\lambda) = f(\lambda)$.

The resulting system of fifteen equations is used to determine the components of the matrix K. These equations are split into three systems of the form $V_1 \cdot \begin{pmatrix} k_{i1} \\ k_{i2} \\ k_{i3} \\ k_{i4} \\ k_{i5} \end{pmatrix} = \begin{pmatrix} f_i(\lambda_1) \\ f_i(\lambda_2) \\ f_i(\lambda_3) \\ f_i(\lambda_4) \\ f_i(\lambda_5) \end{pmatrix}$, $i = 1, 2, 3$. In the special case $\lambda_j = -\frac{1}{2}(j + 1)$ and $a = -\frac{1}{10}$, we have

$$V_1 = \begin{pmatrix} 63/100 & -133/100 & -77/100 & -171/50 & -27/100 \\ 189/50 & -249/50 & -147/100 & -581/50 & 119/50 \\ 323/25 & -731/50 & -119/50 & -817/25 & 551/50 \\ 33 & -176/5 & -143/40 & -384/5 & 153/5 \\ 7047/100 & -7317/100 & -513/100 & -7859/50 & 6757/100 \end{pmatrix}, \tag{6.13}$$

$f_1 = (-252/125 - 4z,\ -2457/125 - 4z,\ -11\,628/125 - 4z,\ -1518/5 - 4z,\ -98\,658/125 - 4z),$
$f_2 = (9z,\ 9z\ 9z\ 9z\ 9z),$
$f_3 = (-65\,201/2 - 50z,\ -136\,255/8 - 25z,\ 17\,431,\ -15\,687/4,\ 62\,181/8 + 25z).$

As a result, $V_1^{-1} f_i^T$ are the elements of the ith row of the matrix K. Finally,

$$K = \begin{pmatrix} 128\,394/5 - 40z_1 & 134\,253/10 - 20z_2 & -68\,724/5 & 15\,437/5 & -61\,163/10 + 20z_3 \\ 90z_1 & 45z_2 & 0 & 0 & -45z_3 \\ -65\,201/2 - 50z_1 & -136\,255/8 - 25z_2 & 17\,431 & -15\,687/4 & 62\,181/8 + 25z_3 \end{pmatrix}$$

with arbitrary values z_1, z_2, and z_3. Substituting the above matrix K into the expression $A + BK - \lambda \cdot I$ yields

$$\det(A + BK - \lambda I) = \begin{vmatrix} 12\,839/2 - \lambda & 134\,253/40 & -17\,181/5 & 15\,437/20 & -61\,163/40 \\ -65\,201/10 & -136\,259/40 - \lambda & 17\,431/5 & -15\,687/20 & 62\,181/40 \\ -502/5 & -1001/20 & 499/10 - \lambda & -25/2 & 509/20 \\ -65\,201/5 & -27\,251/4 & 34\,862/2 & -1569 - \lambda & 62\,181/20 \\ 63\,193/10 & 132\,251/40 & -16\,931/5 & 15\,187/20 & -60\,157/40 - \lambda \end{vmatrix}$$

$$= -\frac{45}{2} - \frac{261}{4}\lambda - \frac{145}{2}\lambda^2 - \frac{155}{4}\lambda^3 - 10\lambda^4 - \lambda^5$$

$$= (\lambda + 1)\left(\lambda + \frac{3}{2}\right)(\lambda + 2)\left(\lambda + \frac{5}{2}\right)(\lambda + 3).$$

Note that in this example, the variability of the gain matrix depends on the choice of arbitrary values z and on the values of w_{13} and w_{21} such that the vectors $v(\lambda_j)$ (6.11) are linearly independent.

The case of a multiple spectrum

It is required to construct a matrix K such that $\lambda = -2$ is an eigenvalue of multiplicity 5 of the matrix $A + BK$, where A and B are the matrices appearing in system (6.1).

The vector $v(-2) = \tau_0$ is constructed by formula (6.12) with $a = -\frac{1}{10}$ and $\lambda = -2$; the components of the vector $v(-2) = \tau_0$ are those of the third row of the matrix V_1 (6.13). The vectors τ_j, $j = 1, 2, \ldots, 4$, are obtained from the well-posedness conditions of the equations $BK\tau_j = (\lambda I - A)\tau_j + \tau_{j-1}$.

The components of the vectors τ_j are those of the jth rows of the matrix V_2, where

$$V_2 = \begin{pmatrix} 323/25 & -731/50 & -119/50 & -817/25 & 551/50 \\ -72/5 & 77/5 & 7/5 & 162/5 & -67/5 \\ 4 & -57/10 & -17/10 & -59/5 & 21/10 \\ 0 & 1 & 1 & 2 & 1 \\ 0 & -17/10 & -17/10 & -38/10 & -19/10 \end{pmatrix}.$$

The equations $Bg_j = (\lambda I - A)\tau_j + \tau_{j-1}$ are used to find the solutions g_j; in this case, for a particular choice of arbitrary constants, we obtain

$g_1 = (-11\,628/125 - 4z,\quad 3884/25 - 4z,\quad -432/5 - 4z,\quad 16 - 4z,\quad -4z),$
$g_2 = (9z,\quad 9z,\quad 9z,\quad 9z,\quad 9z),$
$g_3 = (13\,889/100 - 5z,\quad -1097/5 - 5z,\quad 2623/20 - 5z,\quad -38 - 5z,\quad 423/20 - 5z)$

with an arbitrary value z.

Solving the system

$$V_2 \cdot \begin{pmatrix} k_{i1} \\ k_{i2} \\ k_{i3} \\ k_{i4} \\ k_{i5} \end{pmatrix} = g_i^{\mathrm{T}}$$

yields k_{ij}, $i = 1, 2, 3$, $j = 1, 2, \ldots, 5$.

Computations in Mathcad produced the result

$$K = \begin{pmatrix} 16\,704/5 - 112z & 9519/5 - 58z & -8759/5 & 6613/20 & -7973/10 + 54z \\ 252z & 261/2z & 0 & 0 & -243/2z \\ -4163 - 140z & -4765/2 - 145/2z & 8509/4 & -6363/16 & 8101/8 + 135/2z \end{pmatrix}.$$

Direct check shows that

$$\det(A + BK - \lambda \cdot I)$$

$$= \begin{vmatrix} 835 - \lambda & 9519/20 & -8759/20 & 6613/80 & -7973/40 \\ -4163/5 & -2383/5 - \lambda & 8509/20 & -6363/80 & 8101/40 \\ 13/5 & -11/20 & -63/5 - \lambda & 25/8 & 16/5 \\ -8326/5 & -953 & 8509/10 & -1275/8 - \lambda & 8101/20 \\ 4189/5 & 2377/5 & -9009/20 & 6863/80 & -785\,740 - \lambda \end{vmatrix}$$

$$= -32 - 80\lambda - 80\lambda^2 - 40\lambda^3 - 10\lambda^4 - \lambda^5 = -(\lambda + 2)^5.$$

Remark 1. In the case of a nonsingular matrix B, system (1.1) is controllable, and K is determined from the equations

$$Kv_j = B^{-1}(\lambda_j I - A)v_j, \qquad j = 1, \ldots, n,$$

or

$$Kv_s = B^{-1}(\lambda_j I - A)v_s,$$
$$K\tau_j = B^{-1}(\lambda_j I - A)\tau_j + B^{-1}\tau_{j-1}, \quad j = 1, \ldots, k-1, \quad \tau_0 = v_s,$$

where arbitrary linearly independent vectors can be taken for v_j or $v_1, \ldots, v_s, \tau_1, \ldots, \tau_{k-1}, v_{k+1}, \ldots, v_n$.

7. CONCLUSIONS

This paper has proposed a new algorithm for constructing gain matrices in the spectrum assignment problem of a continuous-time linear dynamic system without any constraints on its matrix coefficients, except for the controllability of the pair (A, B).

The algorithm is based on the cascade decomposition method developed in [8]. In contrast to [8] (with expanding spaces into subspaces, constructing projectors onto them, and using semi-inverse matrices), the algorithm in this paper involves elementary algebraic operations: solving linear algebraic equations, making changes of variables, and checking the linear independence of vectors.

Such a solution of the problem significantly simplifies the computational process and allows creating simple computing programs.

In addition, the dependence of the gain matrix on a certain number of arbitrary or conditionally related scalar parameters has been revealed, and the complete set of such matrices for each problem has been determined.

The case of a unique gain matrix has been identified as well.

Illustrative examples of constructing gain matrices have been provided.

Finally, various cases of constructing such a matrix for a dynamic system describing the operation of a multi-chamber heating furnace have been considered.

FUNDING

This work was supported by the Russian Science Foundation, project no. 24-21-20012, https://rscf.ru/project/24-21-20012/.

APPENDIX

JUSTIFICATION FOR THE ALGORITHM FOR CONSTRUCTING LINEARLY INDEPENDENT VECTORS (SUBSECTIONS 3.4 AND 4.2)

Section 2 has presented the solution of a linear algebraic equation of the form $Cx = y$ with a singular matrix $C : \mathbb{R}^s \to \mathbb{R}^l$, $y \in \mathbb{R}^l$. This solution is explained as follows: using a mapping C, one can expand spaces R^s and $\mathbb{R}^l$ into the direct sums of subspaces:

$$R^s = \operatorname{Coim} C \dot{+} \operatorname{Ker} C, \qquad R^l = \operatorname{Im} C \dot{+} \operatorname{Coker}\ C, \tag{A.1}$$

where Ker C, Im C, and Coker C stand for the kernel, image, and defect (defective) subspace of C, respectively, and Coim C is the direct complement to Ker C in R^s. (Mappings and the corresponding matrices are indicated identically.) Here, the narrowing of $\widetilde{C}$ to Coim C has an inverse mapping $\widetilde{C}^-$ [11]. Let P and Q denote projectors onto Ker C and Coker C, respectively; then the mapping $\widetilde{C}^-(I-Q)$ is called semi-inverse [12, p. 164] and is denoted by C^-. Here, I means an identity mapping in any subspace.

The following result is well-known [8, 13–17].

Lemma. *The relation $Cx = y$ is equivalent to the system*

$$\begin{cases} Qy = 0 \\ x = C^- y + z \end{cases} \tag{A.2}$$

for any z from Ker C.

Note that $z = Px$. The relation $Qy = 0$ in (A.2) is the well-posedness condition for the equality $Cx = y$, and the second relation in (A.2) is the solution of this equation.

By this lemma, equation (3.3) with omitted j, $Bf = (\lambda I - A)v$, has the solution

$f = B^-(\lambda I - A)v + z$ (3.4) for any $z \in$ Ker B under the well-posedness condition (3.5):

$$Q(\lambda\, I - A)v = 0. \tag{A.3}$$

The first step. Denoting

$$Qv = w_1, \quad (I - Q)v = v_1, \quad QAQ = A_1, \quad QA(I - Q) = B_1, \tag{A.4}$$

we have

$$v = v_1 + w_1, \tag{A.5}$$

and (A.3) can be written as

$$B_1 v_1 = (\lambda\, I - A_1)w_1. \tag{A.6}$$

Note that the designations $Qv = w_1$ in (A.4) and (3.6) are identical.

Next we study (A.6) with $B_1 : \operatorname{Im} B \to \operatorname{Coker} B$. Based on the decomposition

$$\operatorname{Im} B = \operatorname{Coim} B_1 \dot{+} \operatorname{Ker} B_1, \qquad \operatorname{Coker} B = \operatorname{Im} B_1 \dot{+} \operatorname{Coker} B_1, \tag{A.7}$$

equation (A.6) (see the lemma) has the solution

$$v_1 = B_1^-(\lambda I - A_1)w_1 + \tilde{z}_1, \quad \forall \tilde{z}_1 \in \operatorname{Ker} B_1, \tag{A.8}$$

under the condition

$$Q_1(\lambda I - A_1)w_1 = 0, \tag{A.9}$$

where P_1 and Q_1 are projectors onto $\operatorname{Ker} B_1$ and $\operatorname{Coker} B_1$, respectively, corresponding to the decomposition (A.7), and B_1^- is the semi-inverse of B_1. Formulas (A.5), (A.8), and (A.9) correspond to formulas (3.7) and (3.8).

The second step. Denoting

$$Qw_1 = w_2, \quad (I - Q_1)w_1 = v_2, \quad Q_1A_1Q_1 = A_2, \quad Q_1A_1(I - Q_1) = B_2,$$

we have

$$w_1 = v_2 + w_2, \tag{A.10}$$

and (A.9) is

$$B_2v_2 = (\lambda I - A_2)w_2.$$

Note that the designation $Qw_1 = w_2$ appears in (3.9).

Applying the lemma to the last equation, we express v_2 through w_2 and an arbitrary vector $\tilde{z}_2 \in \operatorname{Ker} B_2$; using (A.10), we express w_1 through λ, w_2, and $\tilde{z}_2$, this is (3.11) with $i = 2$. And so on ....

As a result, the relation (A.3) is equivalent to the system consisting of the relations (A.5), (A.8), (A.10), and

$$w_i = v_{i+1} + w_{i+1}, \tag{A.11}$$

$$v_{i+1} = B_{i+1}^-(\lambda I - A_{i+1})w_{i+1} + \tilde{z}_{i+1}, \quad \forall \tilde{z}_{i+1} \in \operatorname{Ker} B_{i+1}, \tag{A.12}$$

$$B_pv_p = (\lambda I - A_p)w_p, \tag{A.13}$$

where $B_{i+1}^- = Q_iA_i(I - Q_i)$, $A_{i+1} = Q_iA_iQ_i$, $w_i = Q_{i-1}w_{i-1}$, $v_{i+1} = (I - Q_i)w_i$, Q_i and P_i are projectors onto $\operatorname{Coker} B_i$ and $\operatorname{Ker} B_i$, respectively, corresponding to the decomposition

$$\operatorname{Im} B_{i-1} = \operatorname{Coim} B_i \dot{+} \operatorname{Ker} B_i, \qquad \operatorname{Coker} B_{i-1} = \operatorname{Im} B_i \dot{+} \operatorname{Coker} B_i,$$

$B_{i-1}^- = \tilde{B}_{i-1}^{-1}(I - Q_{i-1})$, $\tilde{B}_{i-1}$ is the narrowing of B_{i-1} to $\operatorname{Coim} B_{i-1}$, $i = 1, \ldots, p$, $B_0 = B$, and $A_0 = A$.

Due to the controllability of the pair (A, B), equation (A.13) is solvable with respect to v_p for any $w_p \in \operatorname{Coker} B_{p-1}$ [8, 13–17]. From (A.11) with $i = p - 1$ we find w_{p-1}; next, from (A.12) with $i = p - 2$ we determine v_{p-1}; then using formula (A.11) with $i = p - 2$ we construct w_{p-2}, and so on .... Finally, using formula (A.5) we obtain $v = v(\lambda, w_p, \tilde{z}_1, \tilde{z}_2, \ldots \tilde{z}_p)$ with arbitrary vectors $w_p \in \operatorname{Coker} B_{p-1}$ and $\tilde{z}_s \in \operatorname{Ker} B_s$, $s = 1, \ldots, p$.

Obviously, constructing v by the method proposed in this work—using linear changes of unknown vectors and solving linear algebraic systems—significantly simplifies the design of gain matrices in the spectrum assignment problem.

REFERENCES

1. Shumafov, M.M., Stabilization of Linear Control Systems. Pole Assignment Problem. A Survey, *Vestnik of St. Petersburg University. Mathematics. Mechanics. Astronomy*, 2019, vol. 6(64), no. 4, pp. 564–591.
2. Zubov, V.I., *Teoriia optimal'nogo upravleniia sudnom i drugimi podvizhnymi ob"ektami* (Theory of Optimal Control of a Ship and Other Moving Objects), Leningrad: Sudostroenie, 1966.
3. Wonham, W.M., On Pole Assignment in Multi-Input Controllable Linear Systems, *IEEE Trans. Aut. Contr.*, 1967, vol. AC–12, no. 6, pp. 660–665.
4. Wonham, M., *Linear Multivariable Control Systems. A Geometric Approach*, New York: Springer-Verlag, 1979.
5. Lapin, A.V. and Zubov, N.E., MATLAB Based Implementation of Analytic Algorithms of Modal Control with State-Vector Feedback and Output-Vector Feedback, *Engineering Journal: Science and Innovation*, 2020, no. 1(97), pp. 1–16.
6. Zubov, N.E., Mikrin, E.A., and Ryabchenko, V.N., *Matrichnye metody v teorii i praktike sistem avtomaticheskogo upravleniya letatel'nykh apparatov* (Matrix Methods in the Theory and Practice of Aircraft Automatic Control Systems), Moscow: Bauman Moscow State Technical University, 2016.
7. Schmid, R., Ntogramatzidis, L., Nguyen, T., and Pandey, A., A Unified Method for Arbitrary Pole Placement, *Automatica*, 2014, vol. 50, no. 8, pp. 2150–2154.
8. Zubova, S.P. and Raetskaya, E.V., Solution of the Spectrum Allocation Problem for a Linear Control System with Closed Feedback, *Differential Equations*, 2024, vol. 60, no. 6, pp. 763–781.
9. Il'in, V.A. and Poznyak, E.G., *Lineinaya algebra* (Linear Algebra), Moscow: Fizmatlit, 1999.
10. Andreev, Yu.N., *Upravlenie konechnomernymi lineinymi ob"ektami* (Control of Finite-Dimensional Linear Plants), Moscow: Nauka, 1976.
11. Atkinson, F.V., The Normal Solvability of Linear Equations in Normed Spaces, *Math. Sb.*, 1951, vol. 28(70), no. 1, pp. 3–14.
12. Bortakovskii, A.S. and Panteleev, A.V., *Lineinaya algebra v primerakh i zadachakh* (Linear Algebra in Examples and Problems), Moscow: Vysshaya Shkola, 2005.
13. Zubova, S.P. and Raetskaya, E.V., Solution of the Multi-Point Control Problem for a Dynamic System in Partial Derivatives, *Mathematical Methods in the Applied Sciences*, 2021, vol. 44, no. 15, pp. 11998–12009.
14. Zubova, S.P. and Raetskaya, E.V., Algorithm to Solve Linear Multipoint Problems of Control by the Method of Cascade Decomposition, *Autom. Remote Control*, 2017, vol. 78, no. 7, pp. 1189–1202.
15. Zubova, S.P., Solution of Inverse Problems for Linear Dynamical Systems by the Cascade Method, *Dokl. Math.*, 2012, vol. 86, no. 3, pp. 846–849.
16. Zubova, S.P., Solution of the Control Problem for Linear Descriptor System with Rectangular Coefficient Matrix, *Math. Notes*, 2010, vol. 88, no. 5–6, pp. 844–854.
17. Zubova, S.P., Trung, L.H., and Raetskaya, E.V., On Polynomial Solutions of the Linear Stationary Control System, *Autom. Remote Control*, 2008, vol. 69, no. 11, pp. 1852–1858.
18. Zubova, S.P. and Raetskaya, E.V., On an Algorithm for Constructing a Gain Matrix for a Linear Dynamic Control System, in *Sovremennye metody teorii funktsii i smezhnye problemy* (Modern Methods of Function Theory and Related Problems), Proceedings of the International Conference "Voronezh Winter Mathematical School," Voronezh: Voronezh State University, 2025, pp. 139–141.

This paper was recommended for publication by M.V. Khlebnikov, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 12, pp. 1138–1152.
© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2025.
Russian Text © The Author(s), 2025, published in Avtomatika i Telemekhanika, 2025, No. 12, pp. 66–84.

NONLINEAR SYSTEMS

Comprehensive Method for Angular Super-Resolution of Group Targets

B. A. Lagovsky[*,a] **and E. Y. Rubinovich**[**,b]

[*] *Russian Technological University, Moscow, Russia*
[**] *Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia*
e-mail: [a] *robertlag@yandex.ru,* [b] *rubinvch@gmail.com*

Received November 19, 2024
Revised April 5, 2025
Accepted April 7, 2025

Abstract—A comprehensive method is proposed to enhance the resolution and accuracy of radar angular measurements for detecting and determining the coordinates of objects in the form of closely spaced multiple aerial targets that cannot be resolved by direct observation. Solving this problem improves the quality of controlling various types of unmanned aerial vehicles (UAVs) located near such targets. The practical implementation of the method is particularly important in calculating and modeling flight trajectories of autonomous and controlled aerial vehicles when visual observation is difficult or ineffective. Mathematically, the problem reduces to solving Fredholm integral equations of the first kind of convolution type with additional constraints. Solutions with angular super-resolution are sought in the form of an expansion of the unknown function over chosen systems of orthogonal functions. For group targets with high object density, it is not always possible to obtain an adequate solution to this inverse problem. In such cases, enhancing the achievable super-resolution degree is proposed based on a new method called the separation method. It is based on excluding from the analyzed signal its component formed by reflection from one or several targets distinguished by some means. The use of nonlinear regression methods in research is justified. Results of numerical experiments on a mathematical model are presented and analyzed.

Keywords: Rayleigh criterion, angular super-resolution, stability of inverse problem solutions

DOI: 10.7868/S1608303225120033

1. INTRODUCTION

Angular super-resolution of a measurement or observation system refers to angular resolution capability exceeding the Rayleigh criterion. The Rayleigh criterion is the minimum angular distance θ_R between two point objects at which the measurement system can still register them separately:

$$\theta_R = \lambda/L, \tag{1}$$

where L is the linear size of the receiving system, λ is the wavelength used. The angle θ_R equals the antenna beamwidth $\theta_{0.5}$, defined at the level of received power reduction by a factor of 2.

In measurements of angular coordinates, the obtained resolution does not exceed the Rayleigh criterion. Results of digital signal processing using special algorithms allow systems to detail images of studied objects with accuracy down to a previously unknown angle $\theta_s < \theta_R$. The value θ_s depends on the signal-to-noise ratio (SNR) in the data, the digital processing method used, as well as the angular reflection (or radiation) characteristics of the signals from the studied objects. Achieved angular super-resolution improves measurement accuracy, enhances detection and identification

probability characteristics. It becomes possible to observe and measure coordinates of individual closely located objects within targets, called group targets, which previously, according to (1), merged into a single extended object.

Consequently, the obtained dynamic picture enables improving the effectiveness of controlling unmanned aerial vehicles (UAVs) operating in zones containing many other UAVs and UAV "swarms."

Several dozen numerical methods for achieving super-resolution and their variations are known [1–7]. However, there is no single universal method for solving the super-resolution achievement problem. All methods and algorithms have various limitations. For example, for most methods, super-resolution can only be achieved at SNR above 20–25 dB. The most well-known methods [5–7] are effective only for solving one-dimensional problems. For two-dimensional problems, algorithms become significantly more complex [8] and do not allow real-time use.

For each specific problem, one should choose its own, most effective method for processing measurement or observation data under the given conditions. The quality of the obtained approximate solution can be improved by combining several new methods into a single comprehensive method. It is based on the sequential application, depending on the results obtained, of new specialized processing methods described in Sections 4–6.

2. PROBLEM FORMULATION

Let a large number of closely located UAVs be present in the surveillance zone of a radar station (ground-based or mounted on an aerial vehicle). A significant portion of objects, due to relatively small distances between them, are not angularly resolved by direct observation. In this case, they form a single large spatial object, i.e., a group target.

It is necessary to isolate the maximum possible number of individual objects in the surveillance zone and determine their coordinates. The characteristics of the measurement system and the received signal are assumed known.

Mathematical problem formulation.

Given: antenna pattern (AP) $F(\alpha, \varphi)$, the received signal $U(\alpha, \varphi)$ during scanning of the two-dimensional surveillance sector Ψ in the form of a linear integral transform:

$$U(\alpha, \varphi) = \int\limits_{\Omega} F(\alpha - \alpha', \varphi - \varphi') I(\alpha', \varphi')\, d\alpha' d\varphi', \tag{2}$$

where Ω is an unknown two-dimensional angle within which signal sources are located, and $\Omega < \Psi$. It is required to find the angular distribution of the reflected signal amplitude $I(\alpha, \varphi)$, equal to zero outside the region Ω.

The problem of finding the function $I(\alpha, \varphi)$ represents an inverse problem, ill-posed in the sense of Hadamard, in the form of a Fredholm integral equation (IE) of the first kind. In such problems, the third condition of well-posedness—stability of solutions with respect to input data—is violated.

It is well known that the stability of inverse problems can be improved by using any additional data about the solution. Algebraic methods [9–15] are convenient for introducing various additional constraints into solutions. Furthermore, the mentioned methods are relatively noise-resistant, and their high speed allows real-time use. For the considered class of problems, they are the most promising.

3. REGULARIZATION OF THE PROBLEM USING THE ALGEBRAIC METHOD

Algebraic methods involve representing one- or two-dimensional solutions (2) as expansions over a given orthogonal system of functions in the region of source location.

Consider a one-dimensional problem when scanning is performed along one coordinate. Generalization of the obtained results to the two-dimensional case presents no fundamental difficulties.

Following the ideology of algebraic methods, the desired solution $I(\alpha)$ can be represented as

$$I(\alpha) = \sum_{m=1}^{\infty} b_m g_m(\alpha) \cong \sum_{m=1}^{M} b_m g_m(\alpha), \tag{3}$$

where $g_m(\alpha),\ m = 1, \ldots, M$ is a chosen finite system of functions orthogonal in the region Ω, b_m are the desired expansion coefficients of $I(\alpha)$ in a series. Then the received signal (2) can be represented as

$$U(\alpha) = \sum_{m=1}^{\infty} b_m G_m(\alpha) \cong \sum_{m=1}^{M} b_m G_m(\alpha), \quad G_m(\alpha) = \int_{\Omega} F(\alpha - \alpha') g_m(\alpha')\, d\alpha', \tag{4}$$

and it turns out to be a superposition of non-orthogonal functions $G_m(\alpha)$. Thus, the inverse problem becomes parameterized. Its approximate solution reduces to finding the vector of coefficients B with elements b_m, which are determined from a system of linear algebraic equations (SLAE) by minimizing the mean square deviation of $U(\alpha)$ (4) from the received signal (2) [16–23]:

$$V = HB, \tag{5}$$

$$V_n = \int_{\Omega} U(\alpha) G_n(\alpha)\, d(\alpha), \quad H_{n,m} = \int_{\Omega} G_m(\alpha) G_n(\alpha)\, d(\alpha), \quad n, m = 1, \ldots, M. \tag{6}$$

If the signal is given as a discrete sequence of values, the problem reduces to solving an overdetermined SLAE based on the least squares method.

The boundaries of the angular sector Ω, within which signal sources are located, are initially chosen based on an estimate, considering a significant decrease in the level of the received signal at the boundaries of Ω relative to its maximum value. Later, based on preliminary solutions, the boundaries of sector Ω are refined, and the solution is sought in a new, usually smaller, angular sector. As numerical experiments have shown, after several iterations, the size and boundaries of sector Ω become close to the true ones.

The choice of the system of functions $g_m(\alpha)$ for representing the solution (3) is based on using a priori information about signal sources or, in its absence, on a reasonable model of physically realizable source types.

Enhancing the achievable super-resolution level is based on increasing the number of functions in the representation (3). However, in the considered inverse problems, matrices H in SLAE (5) are ill-conditioned. Their condition numbers grow exponentially with increasing matrix dimension. A stable approximate solution for UAV-type signal sources can only be obtained for the first few functions from (3), which, at high SNR, allows exceeding the Rayleigh criterion by 2–4 times [10, 13, 14].

To further improve the quality of the approximate solution by increasing the number of used functions in (4)–(6), it is proposed to apply Tikhonov regularization method [16] to the stated problem (2).

Formally, solving SLAE (5) is equivalent to minimizing the function

$$\Phi(B) = \| HB - V \|^2, \tag{7}$$

where $\| \cdot \|$ denotes the vector or matrix norm. The Tikhonov regularization method allows finding the normal solution of system (5) using representation (7), i.e., the vector B minimizing the norm of vector $\| HB - V \|$.

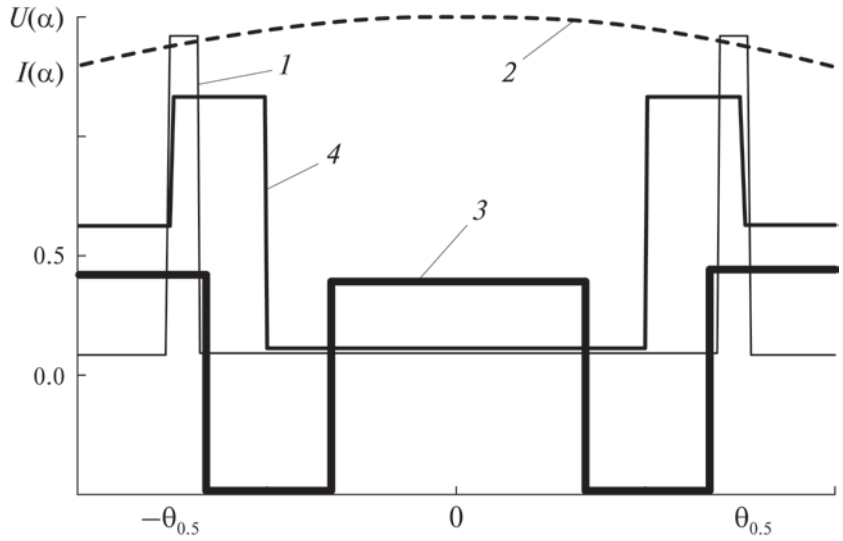

Fig. 1. Solution of the problem for two close objects.

The solution consists in finding the vector B for which the function

$$\Phi(B,\lambda) = \| HB - V \|^2 + \lambda \| B \|^2 \tag{8}$$

reaches the smallest value for a fixed positive λ. It is known that a solution to such a problem exists and is unique [16].

The problem of minimizing function $\Phi(B,\lambda)$ (8), as can be easily shown, is equivalent to solving the SLAE

$$(H^{\mathrm{T}}H + \lambda E)B = H^{\mathrm{T}}V, \tag{9}$$

where E is the identity matrix, H^{T} is the transposed matrix. For $\lambda = 0$, equation (9) transforms into the original SLAE.

The Tikhonov regularization method additionally allows accounting for a possible a priori approximate estimate of the solution, given in the form of expected coordinates of vector B as a vector C.

Then instead of (8), (9) we obtain

$$\begin{gathered}\Phi(B,\lambda) = \| HB - V \|^2 + \lambda \| B - C \|^2, \\ (H^{\mathrm{T}}H + \lambda E)B = H^{\mathrm{T}}V + \lambda C.\end{gathered} \tag{10}$$

Since in solving real problems related to measurement systems, the input data always contain random components, to choose the best solution, it is necessary to perform calculations for various values of the regularization parameter λ, close to the level of random components in the studied signal. As a model example, the problem of achieving angular super-resolution for two closely located identical small-size objects at high SNR was solved.

In Figs. 1 and subsequent figures, the zero direction is taken as the direction to the antenna plane; the vertical axis—the amplitude value of signal sources and the received signal, normalized to unity; the horizontal axis—the scanning angle α of the system in relative units of $\theta_{0.5}$ beamwidth; true signal sources, simulating UAVs, are shown as a thin black solid broken line (curve *1*). The angular dependence of the received signal $U(\alpha)$, showing that during direct observation without signal processing the sources merge into a single object, is presented by a dashed curve (curve *2*). The thick

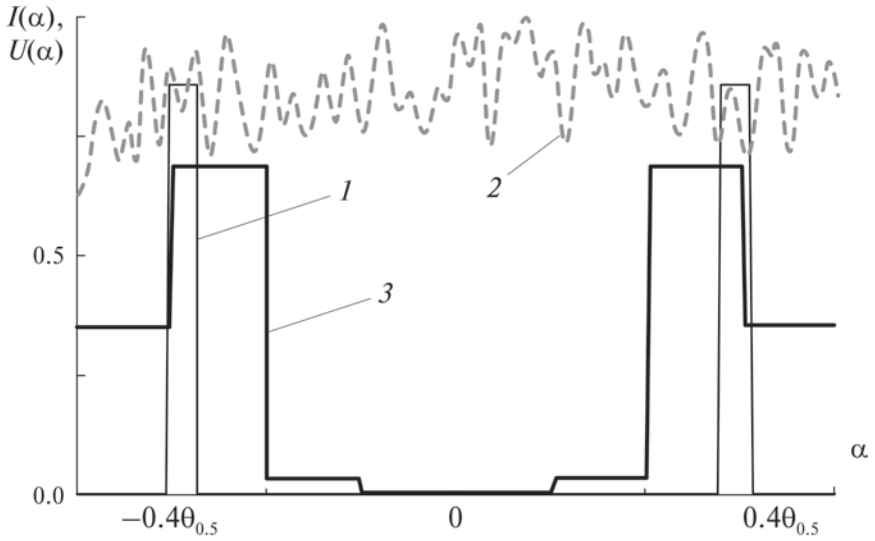

Fig. 2. Solution of the problem for two close objects, SNR = 12dB: *1*—true sources, *2*—received signal $U(\alpha)$, *3*—solution with super-resolution.

stepped line *3* is the solution by the algebraic method (3)–(6) without problem regularization. The obtained inadequate solution is unstable and represents an oscillating function. The broken curve *4* is the stable solution obtained using Tikhonov regularization. Both true sources are observed separately. The Rayleigh criterion is exceeded by a factor of 5.

When comparing super-resolution achievement methods, one of the main quality criteria is the dependence of the degree of exceeding the Rayleigh criterion on SNR. This is because inverse problems are being solved, which are significantly more sensitive to the level of random components in the studied signals than direct problems. By this criterion, Tikhonov regularization allows sharply improving the quality of the obtained solutions.

On Fig. 2, the solution of the same problem is presented in the same notations as in Fig. 1, but at SNR = 12 dB. The obtained stable solution practically did not change, whereas for popular super-resolution achievement methods [1–7] the minimally required SNR level is 20 dB.

4. REGULARIZATION OF THE PROBLEM BY THE SEPARATION METHOD

Suppose that from a group target, using M step functions, it was possible to isolate K separate objects (or one object) and approximately determine their angular positions and amplitudes of reflected signals. Then the number of objects in the new group target will be $M - K$. For the isolated K objects, we assume that their locations $g_k(\alpha)$ and amplitudes b_k have been found:

$$I_k(\alpha) = b_k g_k(\alpha), \quad k = M - K + j, \quad j = 1, \ldots, K. \tag{11}$$

The problem arises of separating the remaining objects in the new group. For this, a method based on extracting from the received reflected signals $U(\alpha)$ only those that belong to the new group target is proposed.

The signal received from the already isolated K sources

$$U_K(\alpha) = \sum_{k=P}^{M} \int_{\Gamma} F(\alpha - \alpha') I_k(\alpha')\, d\alpha' = \sum_{k=P}^{M} b_k \int_{\Gamma} F(\alpha - \alpha') g_k(\alpha')\, d\alpha' = \sum_{k=P}^{M} b_k G_k(\alpha), \tag{12}$$

where $P = M - K + 1$, Γ is the region of location of the isolated sources. For the new problem, signal $U_K(\alpha)$ becomes interference, and strong interference at that, which prevents isolating individual sources from the new group.

It is possible to largely neutralize the influence of this interference. For this, from the received real signal $U(\alpha)$, one should subtract an artificially synthesized signal imitating the signal from the previously separated K sources (12). Then the signal received from the remaining objects, representing the new group target of P objects, becomes equal to

$$U_S(\alpha) = U(\alpha) - U_K(\alpha). \tag{13}$$

The formed signal in the form (13) can be further used in solving the super-resolution achievement problem for the new group target similarly to (3)–(6). Introducing a new system of step functions $h_m(\alpha)$ for this, we finally obtain an SLAE of the form

$$W = QC, \tag{14}$$

where the elements of vector W and matrix Q are represented as:

$$\begin{gathered} W_k = \int\limits_{\Phi} U_S(\alpha) H_k(\alpha)\, d\alpha, \quad Q_{k,q} = \int\limits_{\Phi} H_k(\alpha) H_q(\alpha)\, d\alpha, \\ H_k(\alpha) = \int\limits_{\Phi} F(\alpha - \alpha') h_k(\alpha')\, d\alpha'. \end{gathered} \tag{15}$$

Here C is the vector with elements b_k from (11), Φ is the region of location of the new group target.

Estimation of the position and boundaries of region Φ is now performed based on analysis of the synthesized signal $U_S(\alpha)$ similarly to determining region Ω in (6). If the previously separated K objects are located near the boundary of region Ω, then the size of region Φ becomes smaller than Ω. If the separated objects are not located at the edge of the region, then the new region Φ also turns out to be smaller than Ω, but doubly connected or even multiply connected. Constructing a solution in a multiply connected region is performed according to the same scheme as in the one-dimensional case, by introducing a finite system of functions orthogonal in region Φ.

In the new reduced region Φ, the same number of functions $g_m(\alpha)$ as in region Ω in (3) provides greater resolution without a substantial increase in the condition numbers of matrix H in (5). In many cases, this allows angular resolution of objects of the new group target. The described approach to the problem of analyzing a UAV swarm or other objects constituting a group target can be called the *separation method*.

5. SEPARATION METHOD

It should be noted that the coefficients b_m from (5), (6), and consequently $U_S(\alpha)$, are found with some error. For direct problems, the error may be insignificant and can be neglected. However, the considered problem is inverse, the stability of its solutions is significantly worse. To obtain adequate results, it is necessary to refine the values of b_m before their further use in (7)–(15).

Refinement of the values is proposed to be carried out according to the following scheme. Problem (3)–(6) is solved, but instead of signal $U(\alpha)$, signal $U_S(\alpha)$ is used, i.e., vector V is replaced by vector W from (10). In the idealized case, i.e., if coefficients b_m and $U_S(\alpha)$ were previously found accurately, all coefficients b_k in (7) turn out to be zero.

In the real case, we obtain values b_m and b_k satisfying the relation

$$U_S(\alpha) = U(\alpha) - \sum_{k=M-K+1}^{K} b_k^0 G_k(\alpha) = \sum_{m=1}^{M-K} b_m^1 G_m(\alpha) + \sum_{k=M-K+1}^{M} b_k^1 G_k(\alpha), \tag{16}$$

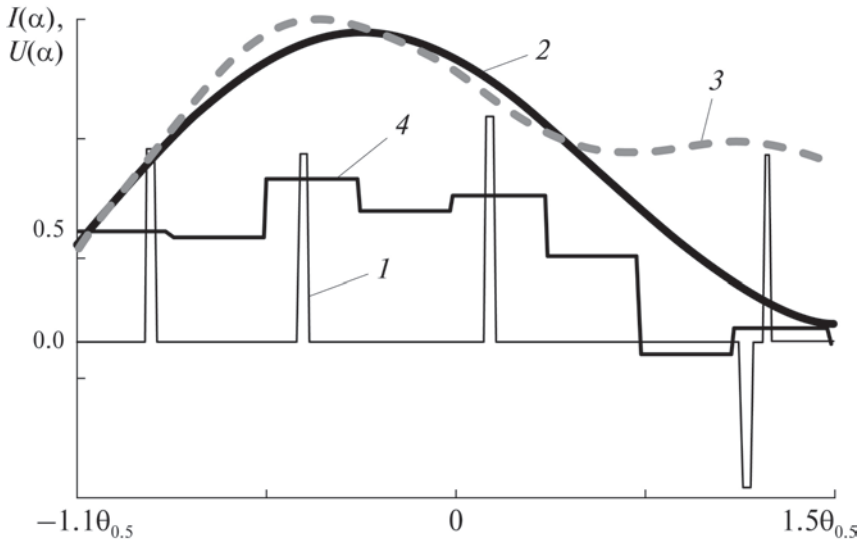

Fig. 3. Separation of one object from a group target. *1*—True sources, *2*—synthesized signal $U_S(\alpha)$, *3*—received signal, *4*—solution with super-resolution.

where b_k^0 are the values of b_k, found with some error when solving problem (8)–(15) with the original signal $U(\alpha)$, b_m^1 are refined values of coefficients b_m when solving the problem with the original signal $U_S(\alpha)$, b_k^1 are residual values of coefficients b_k not equal to zero, arising due to errors in determining b_k^0.

To reduce these errors, an iterative process [17] is then built. Instead of coefficients b_k^0 in (8) at the first step, we use $\widetilde{b}_k^1 = b_k^0 + b_k^1$, and instead of $U_S(\alpha)$ in the form (9) $U_S^1(\alpha)$ i.e.

$$U_S(\alpha) = U(\alpha) - U_K(\alpha), \quad U_K^1(\alpha) = \sum_{k=M-K+1}^{M} \widetilde{b}_k^1 G_k(\alpha). \tag{17}$$

Further, similarly to (14), (15), we obtain the second approximation b_k^2. At the next step, we use $\widetilde{b}_k^1 = b_k^0 + b_k^1 + b_k^2$, etc. At the nth step, we arrive at the relation

$$U_S^n\alpha) = U(\alpha) - U_K(\alpha), \qquad U_K^1(\alpha) = \sum_{k=M-K+1}^{M} \widetilde{b}_k^1 G_k(\alpha). \tag{18}$$

Numerical experiments showed that in several iterations b_k^n decrease significantly compared to the initial values and become comparable to errors in the input data.

As an example of applying the separation method, consider a group target of four point objects located in the angular sector $[-1.1\theta_{0.5};\ 1.5\theta_{0.5}]$, with small differences in signal amplitude values.

In Fig. 3, the original angular dependence of the received signal $U(\alpha)$ is shown as a dashed curve (curve *3*). The objects are not angularly resolved by direct observation. Solution by the algebraic method (3)–(6) using eight step functions allowed isolating one object from the group target.

Using a larger number of functions leads to unstable inadequate solutions that cease to reflect reality. Such solutions can be easily recognized, as coefficients b_m begin to take values orders of magnitude larger than real ones.

The degree of solution stability can be assessed using condition numbers of matrices H from (5). Note that condition numbers of such matrices increase sharply (exponentially) with an increase in the number of used functions. Consequently, the number of functions $g_m(\alpha)$ in representing solution (3) is always limited.

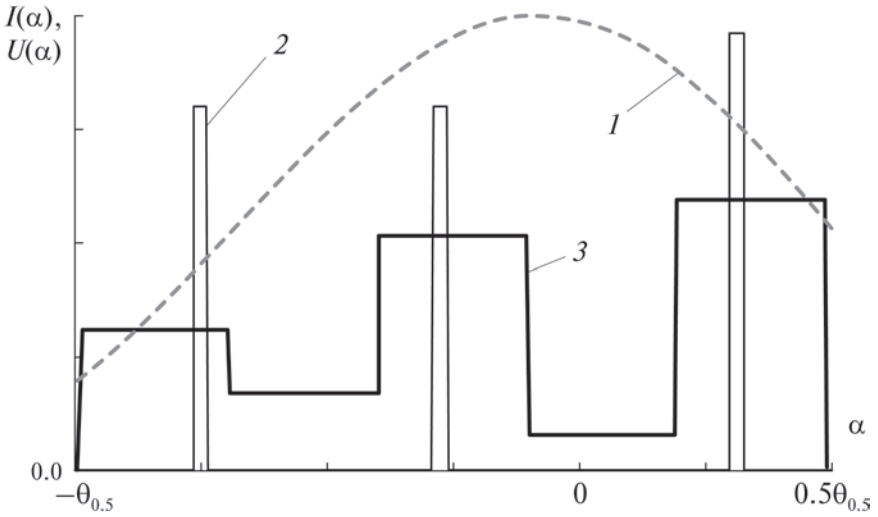

Fig. 4. Solution of the problem based on the object separation method.

Figure 3 shows: as a thin broken line (curve *1*)—positions of true point signal sources and the introduced virtual source $I_N(\alpha)$ (12) with negative polarity; as a thick solid curve (curve *2*)—the synthesized signal $U_S(\alpha)$, which was used to refine the signal amplitude reflected by the fourth object, i.e., refining b_4; dotted curve (curve *3*)—the reflected signal; as a thick stepped curve (curve *4*)—the solution of the inverse problem (2) based on the synthesized signal $U_S(\alpha)$ in the original region Ω.

The solution confirmed the correctness of isolating one object from the group target. However, resolving the remaining objects in angle using the eight introduced step functions $g_m(\alpha)$ in Ω was not possible.

In this case, one should proceed to the second stage of solving the problem by the separation method in the found new region Φ (15).

The second stage consists in using $U_S^n(\alpha)$ instead of $U_S(\alpha)$ in (13)–(15) for the newly formed group target.

On Fig. 4, the solution results are presented. As a thick dashed curve (curve *1*), the synthesized signal $U_S^n(\alpha)$ (18) is shown. Thin broken line (curve *2*)—true point signal sources, thick solid broken line (curve *3*)—the found solution using five step functions $g_m(\alpha)$ in region Φ, the size of which turned out to be 1.75 times smaller than the original region Ω.

Sufficient solution stability and the achieved super-resolution level in the considered example allowed separately registering and determining the coordinates of all objects in the new group target.

The second stage of solution does not always allow resolving objects of the group target. For each specific target and specific measurement system, there exists an angular distance between objects below which their resolution is impossible.

6. NONLINEAR REGRESSION

In cases where resolving objects of a group target using the proposed methodology fails, a more complex approach is proposed in the comprehensive method.

Note that the above separation method is based on determining object amplitudes b_j with high accuracy. However, the accuracy of the angular position of each object does not change, as it is

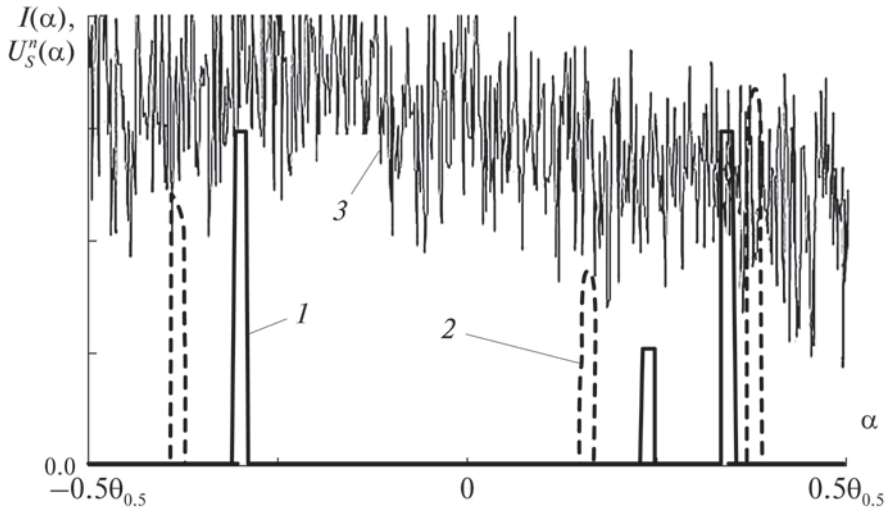

Fig. 5. Solution of the problem for a group target using nonlinear regression.

limited by the angular size of the used step function. More accurate determination of coordinates α_j simultaneously with searching for b_j will improve the quality of obtained solutions.

Assume, as in the given examples, that angular coordinates of objects are negligibly small and can be described using delta-functions $\delta(\alpha - \alpha_j)$, where α_j is the coordinate of the jth object, $j = 1, \dots N$, $N = M - K$. Then instead of (3)–(4) we obtain further (19)–(20)

$$I(\alpha) = \sum_{m=1}^{N} \delta(\alpha - \alpha_m), \tag{19}$$

$$U_S^n(\alpha) = \sum_{m=1}^{N} F(\alpha - \alpha_m). \tag{20}$$

Search for new refined solutions α_j and b_j is proposed to be carried out based on applying nonlinear regression [25–29]. We introduce a regression function similar to (16), i.e.

$$L(\alpha) = \sum_{m=1}^{N} d_m F(\alpha - \gamma_m), \tag{21}$$

where γ_m and d_m are the desired parameters. We also use an additional condition in the form of equality of received signal powers

$$\int_{\Phi} \left(U_S^n(\alpha)\right)^2 d\alpha = \int_{\Phi} \left(\sum_{m=1}^{N} d_m F(\alpha - \gamma_m)\right)^2 d\alpha. \tag{22}$$

Solution based on minimizing the mean square deviation of (21) from $U_S^n(\alpha)$ reduces to solving a system of nonlinear equations. Then, using standard nonlinear regression algorithms, we determine unknown parameters γ_m and d_m minimizing the deviation.

Numerical search for solutions for several objects does not cause significant difficulties, since a good initial approximation is used in the form of previously found b_m and coordinates α_m specified within each previously used step function.

On Fig. 5, an example of solving the problem for three closely located objects with different amplitudes of reflected signals is given. The real location of point sources and their amplitude

values are shown by solid broken lines (curve *1*). The useful signal $U_S^n(\alpha)$ received during scanning together with the noise component within angles $[-\theta_{0.5}/2, \theta_{0.5}/2]$ is shown as a rapidly oscillating curve (curve *3*). Using the previously described methods did not allow resolving the targets. Using nonlinear regression allowed obtaining an adequate solution, shown by a dashed broken line (curve *2*).

All objects are resolved, their location is determined with good accuracy, amounting to 0.07 beamwidth. Amplitude values were found with less accuracy, but they are of secondary importance. The resolution exceeded the Rayleigh criterion by more than four times.

Solving the system of nonlinear equations based on (19)–(22) is noticeably more stable than solving SLAE (14). Numerical experiments showed that adequate solutions are obtained at SNR down to 18 dB, i.e., at higher levels of random components than when using known super-resolution methods.

In some cases, when UAVs are located with high density, resolving objects by the methods described above fails. However, through a more complex approach and some complication of the technical system, achieving super-resolution is possible in these cases as well. For this, it is proposed to use a so-called *harmonic radar*.

7. NONLINEAR SECONDARY RADAR

Secondary, or harmonic radar (HR), upon reception uses a signal frequency two or three times higher than the probing (or interrogation) signal frequency. The received signal is formed due to reflection from nonlinear elements of the studied objects. As a result, the signal represents a superposition of harmonics, multiples of the emitted frequency [24–29].

Ordinary radar objects exhibit nonlinear properties to a very small degree. Currently, HR with specially built-in nonlinear elements are most often used in rescue operations under complex conditions on land and sea for searching and measuring coordinates of objects.

In the considered problems, in the studied angular region at the emitted frequency f_0, there are many signals reflected from a large number of objects, including dangerous mobile objects. Moreover, signals can represent interference at the used frequency f_0, including intentionally created ones. Ultimately, determining coordinates of individual objects by analyzing the total response signal at frequency f_0 turns out to be practically impossible.

At the same time, nonlinear HR detects only targets possessing nonlinear properties, using frequencies $2f_0$ and $3f_0$, and any signals and interference at frequency f_0 do not affect its operation.

The main disadvantage of HR is the weak reflected signal at the used, multiples of f_0, frequencies compared to reflection at the original frequency f_0. This circumstance significantly limits the range D of nonlinear radars. Various methods are proposed to increase it [24–29]. Ultimately, at frequencies $2f_0$ and $3f_0$, a range up to $D \sim 1$–5 km can be achieved.

Ensuring determination of coordinates of individual objects is possible when reducing the number of objects in the studied region. For this, it is proposed to install lightweight small-size nonlinear elements on certain types of UAVs. Simple antennas, including printed ones, in the form of dipoles with a diode, can be used as such. Then, against the background of signals reflected by them at frequencies $2f_0$ and $3f_0$, other UAVs will become practically unnoticeable.

In this case, it becomes possible to measure coordinates of selected types of UAVs at frequencies $2f_0$ and $3f_0$, including using super-resolution methods if necessary. At the main frequency f_0, the above-described method of separating objects in the form of selected types of UAVs with now known coordinates becomes applicable. Consequently, the number of UAVs in the surveillance zone decreases, and the problem of determining coordinates of the remaining UAVs of other types is simplified. Its solution is now carried out according to the above scheme (3)–(6) and (7)–(10).

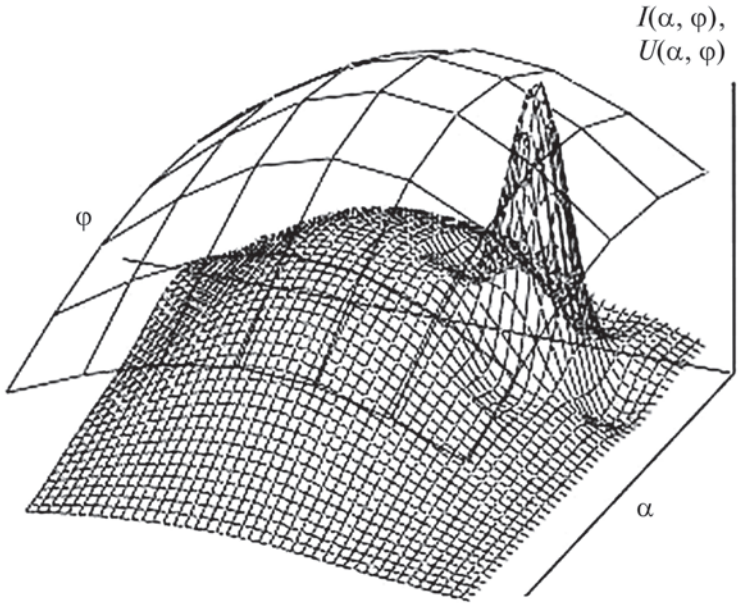

Fig. 6. Improving the accuracy of angular measurements of airborne radars.

Low accuracy of angular measurements for aerial vehicles of relatively small size in autonomous mode at frequencies $2f_0$ and $3f_0$ can significantly reduce the effectiveness of using the presented separation method. This drawback can be largely compensated by using algebraic signal processing methods of airborne radars (3)–(6) to achieve super-resolution at frequencies $2f_0$ and $3f_0$. An example of such compensation is given in Fig. 6.

As a sparse grid, the two-dimensional received signal $U(\alpha, \varphi)$ at frequency $2f_0$ is shown. The true distribution of the emitted signal amplitude was a point source and a small background emission at the second harmonic. When solving direct problems, the background emission can be neglected without any noticeable error. However, when solving inverse unstable problems in the form of IE (2), it can noticeably distort solutions.

The received signal $U(\alpha, \varphi)$ very roughly estimates the angular coordinates of the source. In the considered problem, this is the region covered by the sparse grid. The approximate solution found by the algebraic method (4)–(6), i.e., distribution $I(\alpha, \varphi)$ considering the interfering background emission, is shown as a dense grid. The obtained solution of the IE shows that the point source is located within the peak of $I(\alpha, \varphi)$.

The accuracy of determining the coordinates of signal sources and its localization improved by more than five times. Noise in the form of background emission slightly distorted the true solution and did not allow obtaining even more accurate source coordinates.

For HR, further improvement of angular measurement accuracy and angular resolution is possible. For this, one should simultaneously analyze received signals at the second $U_2(\alpha, \varphi)$ and third $U_3(\alpha, \varphi)$ harmonics. The simplest method of analysis is summing the amplitudes of these signals during scanning:

$$U(\alpha, \varphi) = U_2(\alpha, \varphi) + kU_3(\alpha, \varphi)$$
$$= \int_{\Omega} \left[f_2(\alpha - \alpha', \varphi - \varphi') I_2(\alpha', \varphi') + k f_3(\alpha - \alpha', \varphi - \varphi') I_3(\alpha', \varphi') \right] d\alpha'\, d\varphi', \tag{23}$$

where $I_2(\alpha, \varphi)$ and $I_3(\alpha, \varphi)$ describe the same angular position of the source but with different amplitude values. For signals at the third harmonic, a gain coefficient k is introduced. Since the

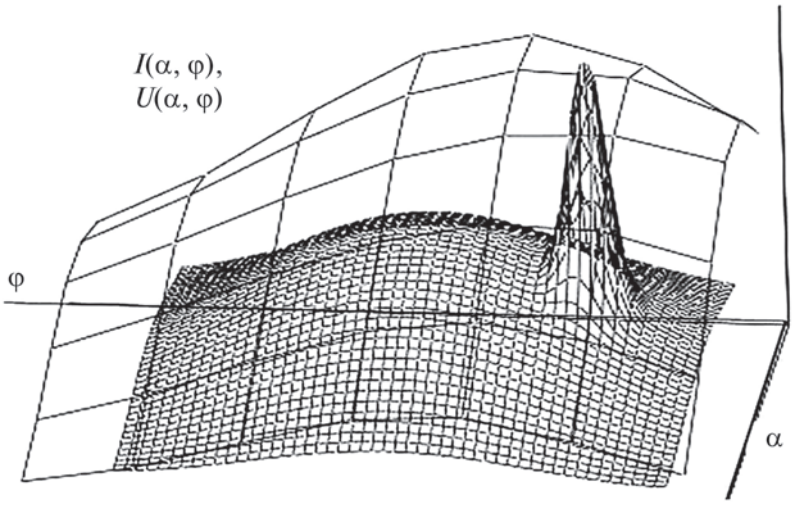

Fig. 7. Solution of IE for the sum of two harmonics.

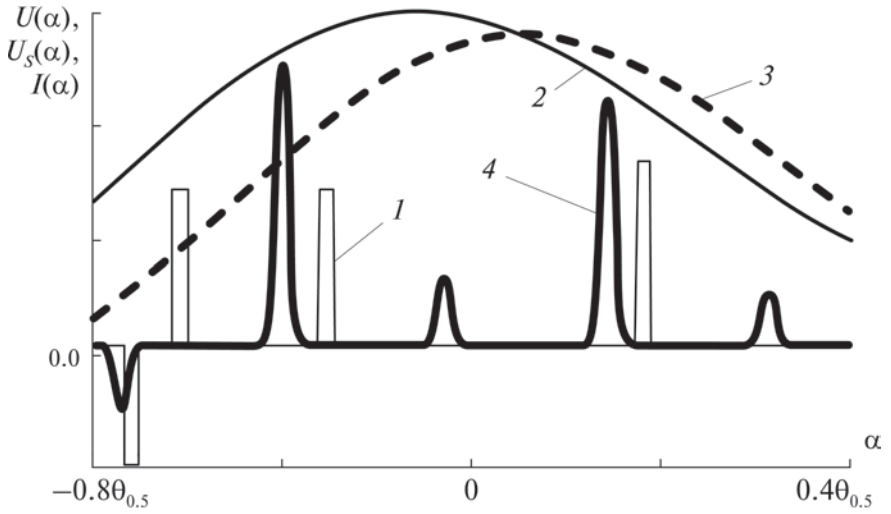

Fig. 8. Solution when separating one object of known type. The solution of the problem of determining angular coordinates of a complex object using HR is presented. The object represented a group target consisting of three closely located point targets—thin broken line (curve *1*). The original signal $U(\alpha)$ (2) is shown as a thin smooth curve (curve *2*).

amplitude of signals at the third harmonic is significantly lower than at the second, coefficient k is chosen from the condition of approximate equality of maximum values of $U_2(\alpha, \varphi)$ and $U_3(\alpha, \varphi)$. The condition of equality of maxima ensures approximately equal contribution of harmonics to the desired solution. An additional advantage of summing harmonics is that noises present in signals $U_2(\alpha, \varphi)$ and $U_3(\alpha, \varphi)$ are non-coherent, and their addition ensures an increase in SNR in (23) compared to SNR at each of frequencies $2f_0$ and $3f_0$. In Fig. 7, results of solving IE (23) for the same problem as in Fig. 6 are shown, in the same notations. As expected, the localization accuracy of the true source noticeably increased—by a factor of two. This ensured an improvement in the accuracy of solving the problem of separating an object of known type from a group target. Resolving targets is not possible either by direct observation or using known super-resolution methods.

One of the objects, namely the one located close to the left boundary of zone Ω, at frequencies $2f_0$ and $3f_0$, is identified by HR as an object of known type. At these frequencies, its angular

coordinate α_1 is measured. According to the separation method (12)–(14) at frequency f_0, a compensating signal with negative polarity is synthesized:

$$U_S(\alpha) = U(\alpha) - AF(\alpha_1). \tag{24}$$

The reflection coefficient A at frequency f_0 is not known in advance and is initially set based on a reasonable estimate. Then it is subject to refinement during the iterative process of solving the problem similar to (16)–(18).

The final signal $U_S(\alpha)$ (14) is shown in Fig. 8 as a thick dashed curve (curve *3*).

For representing the solution, Gaussian functions were used as functions $g_m(\alpha)$ in (3). The found solution without the previously identified UAV with angular coordinate α_1 is shown as a thick curve (curve *4*). In the final solution, the source was represented as a superposition of five Gaussian functions $g_m(\alpha)$:

$$I(\alpha) = \sum_{m=1}^{5} b_m g_m(\alpha - \Delta\alpha m) \tag{25}$$

with distance $\Delta = 0.23\theta_{0.5}$ between maxima of adjacent $g_m(\alpha)$.

True, near-point signal sources together with the virtual compensating source $A\delta(\alpha - \alpha_1)$ are shown as a thin broken line (curve *1*).

The appearance in the solution of many small objects, the radiation intensity $I^2(\alpha)$ of which is an order of magnitude lower than that of the others, is usually caused by the influence of noise and interference. Such targets are considered false.

Consequently, the obtained solution allowed resolving all objects of the group target and determining their angular coordinates with good accuracy.

8. CONCLUSION

1. A new comprehensive method for digital processing of radar signals reflected from multiple objects is theoretically justified and tested in numerical experiments. The method allows measurement systems to achieve angular resolution significantly exceeding the Rayleigh criterion.

2. The proposed separation method, as shown by analytical and numerical results, is applicable for detecting and determining coordinates of individual objects within group targets, including those consisting of UAV "swarms." Creating an artificially synthesized signal during processing of the obtained information allows substantially increasing the achievable super-resolution level compared to known methods.

3. New signal processing algorithms based on nonlinear regression methods allow improving the accuracy of determining angular coordinates and enhancing localization of studied objects by 2–5 times.

4. Using mathematical models, it is shown that nonlinear regression methods allow solving super-resolution achievement problems at higher levels of noise and interference than known methods.

5. The use of harmonic radar for achieving angular resolution allowing the isolation of dangerous UAVs within group targets is justified.

6. Algorithms based on the proposed comprehensive method are relatively simple and can be applied in real time.

FUNDING

The work was partially supported by the Russian Science Foundation (project no. 23-29-00448).

REFERENCES

1. Morse, P. and Feshbach, H., *Methods of Theoretical Physics*, McGraw-Hill, 1953.
2. Uttam, S. and Goodman, N.A., Superresolution of coherent sources in real-beam Data, *IEEE Trans. Aerosp. Electron. Syst.*, 2010, vol. 46, no. 3, pp. 1557–1566.
3. Park, S.C., Park, M.K., and Kang, M.G., Super-resolution image reconstruction: a technical overview, *IEEE Signal Process. Mag.*, 2003, vol. 20, no. 3, pp. 21–36.
4. Kasturiwala, S.B. and Ladhake, S.A., Superresolution: A novel application to image restoration, *Int. J. Comput. Sci. Eng.*, 2010, no. 5, pp. 1659–1664.
5. Waweru, N.P., Konditi, D.B.O., and Langat, P.K., Performance analysis of MUSIC Root-MUSIC and ESPRIT, *Int. J. Electr. Comput. Energ. Electron. Commun. Eng.*, 2014, vol. 8, no. 1, pp. 209–216.
6. Lavate, T.B., Kokate, V.K., and Sapkal, A.M., Performance analysis of MUSIC and ESPRIT, *Proc. 2nd Int. Conf. Comput. Netw. Technol. (ICCNT)*, 2010, pp. 308–311.
7. Almeida, M.S. and Figueiredo, M.A., Deconvolving images with unknown boundaries using the alternating direction method of multipliers, *IEEE Trans. Image Process.*, 2013, vol. 22, no. 8, pp. 3074–3086.
8. Evdokimov, N.A., Lukyanenko, D.V., and Yagola, A.G., Application of multiprocessor systems to solving the two-dimensional convolution-type Fredholm integral equations of the first kind for vector-functions, *Numer. Methods Program.*, 2009, vol. 10, p. 263.
9. Lagovsky, B.A. and Rubinovich, E.Y., Algebraic methods for achieving superresolution by digital antenna arrays, *Mathematics*, 2023, vol. 11, no. 4, pp. 1–9. https://doi.org/10.3390/math11041056
10. Lagovsky, B., Samokhin, A., and Shestopalov, Y., Angular Superresolution Based on A Priori Information, *Radio Sci.*, 2021, vol. 5, no. 3, pp. 1–11. https://doi.org/10.1029/2020RS007100
11. Lagovsky, B.A. and Rubinovich, E.Y., A modified algebraic method of mathematical signal processing in radar problems, *Results Control Optim.*, 2024, vol. 14, no. 3, p. 100405. https://doi.org/10.1016/j.rico.2024.100405
12. Alexandrov, A.E., Borisov, S.P., Bunina, L.V., Bikovsky, S.S., Stepanova, I.V., and Titov, A.P., Statistical model for assessing the reliability of non-destructive testing systems by solving inverse problems, *Russ. Technol. J.*, 2023, vol. 11, no. 3, pp. 56–69.
13. Lagovsky, B. and Rubinovich, E., Algorithms for Digital Processing of Measurement Data Providing Angular Superresolution, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2021, vol. 22, no. 7, pp. 349–356.
14. Lagovsky, B. and Rubinovich, E., Achieving Angular Superresolution of Control and Measurement Systems in Signal Processing, *Adv. Syst. Sci. Appl.*, 2021, vol. 21, no. 2, pp. 104–116.
15. Lagovsky, B.A. and Rubinovich, E.Y., A modified algebraic method of mathematical signal processing in radar problems, *Results Control Optim.*, 2024, vol. 14, no. 3, p. 100405. https://doi.org/10.1016/j.rico.2024.100405
16. Tikhonov, A.N. and Arsenin, V.Y., *Methods for Solving Ill-Posed Problems*, Nauka, 1979.
17. Lagovsky, B.A. and Rubinovich, E.Y., Enhancing angular resolution and range of measurement systems using ultra-wideband signals, *Avtom. Telemekh.*, 2023, no. 10, pp. 72–90. https://doi.org/10.31857/S0005231023100070
18. Kelley, C.T., *Iterative Methods for Optimization*, SIAM, 1999.
19. Marquardt, D.W., An Algorithm for Least-Squares Estimation of Nonlinear Parameters, *J. Soc. Ind. Appl. Math.*, 1963, vol. 11, no. 2, pp. 431–441. https://doi.org/10.1137/0111030
20. Lourakis, M.I.A. and Argyros, A.A., Is Levenberg-Marquardt the Most Efficient Optimization Algorithm for Implementing Bundle Adjustment, *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, 2005, pp. 1526–1531. https://doi.org/10.1109/ICCV.2005.128
21. Nocedal, J. and Wright, S.J., *Numerical Optimization*, Springer, 2006. https://doi.org/10.1007/978-0-387-40065-5

22. Seber, G.A.F. and Wild, C.J., *Nonlinear Regression*, Wiley, 1989.

23. Oosterbaan, R.J., Frequency and Regression Analysis, in *Drainage Principles and Applications*, H.P. Ritzema, Ed., ILRI, 1994, vol. 16, pp. 175–224.

24. Lavrenko, A., Cavers, J.K., and Woodward, G.K., Harmonic Radar With Adaptively Phase-Coherent Auxiliary Transmitters, *IEEE Trans. Signal Process.*, 2022, vol. 70, pp. 1788–1802. https://doi.org/10.1109/TSP.2022.3164183

25. Harzheim, T., Muhmel, M., and Heuermann, H., A SFCW harmonic radar system for maritime search and rescue using passive and active tags, *Int. J. Microw. Wirel. Technol.*, 2021, vol. 13, no. 7, pp. 691–707. https://doi.org/10.1017/S1759078721000520

26. Mazzaro, G.J. and Martone, A.F., Multitone harmonic radar, *Proc. SPIE*, 2013, vol. 8714, p. 87140E. https://doi.org/10.1117/12.2014241

27. Kumar, D., Mondal, S., Karuppuswami, S., Deng, Y., and Chahal, P., Harmonic RFID communication using conventional UHF system, *IEEE J. Radio Freq. Identif.*, 2019, vol. 3, no. 4, pp. 227–235.

28. Mondal, S., Kumar, D., and Chahal, P., Recent advances and applications of passive harmonic RFID systems, *Micromachines*, 2021, vol. 12, no. 4, pp. 1–22.

29. Viikari, V., Sepp, H., and Kim, D.-W., Intermodulation read-out principle for passive wireless sensors, *IEEE Trans. Microw. Theory Tech.*, 2011, vol. 59, no. 4, pp. 1025–1031.

This paper was recommended for publication by B.M. Miller, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 12, pp. 1153–1167.
© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2025.
Russian Text © The Author(s), 2025, published in Avtomatika i Telemekhanika, 2025, No. 12, pp. 85–103.

STOCHASTIC SYSTEMS

Generalized $\mathcal{H}_2$ Control of a Continuous-Time Markov Jump Linear System on a Finite Horizon

R. S. Biryukov[*,a] **and E. S. Bubnova**[*,b]

[*] *Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod, Russia*
e-mail: [a] *ruslan.biryukov@itmm.unn.ru,* [b] *bubnova@itmm.unn.ru*

Received April 2, 2025
Revised June 10, 2025
Accepted June 30, 2025

Abstract—For continuous-time Markov jump linear systems, the concept of the generalized $\mathcal{H}_2$ norm is introduced as the worst-case value of the maximum of the expected squared Euclidean norm of the target output on a finite horizon, provided that the sum of the squared energy of an exogenous disturbance and a quadratic form of the initial state is equal to one. This norm is characterized in terms of coupled Riccati differential matrix equation solutions and in terms of linear matrix inequalities. Linear dynamic state-feedback controllers ensuring an upper bound for the $\mathcal{H}_2$ norm of the closed-loop system are designed by solving a semidefinite programming problem. The effectiveness of the approach is demonstrated by the results of numerical simulations.

Keywords: generalized $\mathcal{H}_2$ norm, linear matrix inequalities (LMIs), homogeneous Markov chains, multi-objective control

DOI: 10.7868/S1608303225120044

1. INTRODUCTION

Random structure systems, particularly Markov jump systems [1–6], are widespread in modern control problems. Such systems have a finite number of distinct operation modes, and the dynamics in each mode are described by a specific system of differential equations. Jumps between modes occur at random time instants, determined by the evolution of a homogeneous Markov chain (Markov jumps, also called Markovian switching in the literature). The simplest problems leading to random structure systems are control with failures and disruptions [2], synchronization in variable topology networks [3, 4], multi-agent control [5, 6], and others.

The stability problem for random structure systems was pioneered by I.Ia. Kats and N.N. Krasovskii [7]. Later, various formulations of control problems for such systems were considered in [8–10]; in particular, a linear-quadratic controller was designed. In [1, 11–14], $\mathcal{H}_\infty$ and $\mathcal{H}_2$ control problems were solved for Markov jump systems. The $\mathcal{H}_\infty$ and $\mathcal{H}_2$ norms allow assessing the quality of transients on average. However, it is often necessary to guarantee that the maximum value of a transient (target output) will not exceed a given threshold, i.e., to estimate the maximum value of the system's target output. One possible approach to solving this problem is based on the generalized $\mathcal{H}_2$ norm.

For continuous-time systems, the concept of the generalized $\mathcal{H}_2$ norm, corresponding to the maximum deviation of a system under an exogenous disturbance of bounded energy and zero initial conditions, was introduced in [15]. The generalized $\mathcal{H}_2$ norm characterizes the system gain when the input signal has a bounded L_2 norm and the output signal is measured by the L_∞ norm

(the maximum value of the Euclidean norm of the target output over time). For a linear continuous time-varying system, the concept of the maximum deviation was introduced in [16, 17] as a natural extension of the generalized $\mathcal{H}_2$ norm to systems with nonzero initial conditions, and an algorithm for its calculation was presented therein as well. As shown in [16, 17], multi-objective control problems with generalized $\mathcal{H}_2$ norms as performance criteria can be effectively solved using the apparatus of linear matrix inequalities (LMIs).

In [18], an estimate of the generalized $\mathcal{H}_2$ norm was calculated for continuous-time Markov jump linear systems on an infinite horizon. For such systems, suboptimal generalized $\mathcal{H}_2$ control was designed in [19]. Another definition of the generalized $\mathcal{H}_2$ norm, differing from the classical one, was used in [20]: estimates for the first absolute moment of the system output components under bounded-energy disturbances were constructed. In [21], the problem of generalized $\mathcal{H}_2$ filtering for semi-Markovian systems was considered. Problems of generalized $\mathcal{H}_2$ filtering and control for discrete-time Markov jump systems were studied in [22–24]. In all the works mentioned, the dynamics of systems in each operation mode were described by time-invariant systems.

In this paper, the concept of the generalized $\mathcal{H}_2$ norm for linear continuous-time Markov jump systems is considered on a finite horizon, and the systems are generally supposed to be time-varying in each operation mode. Several algorithms are proposed for calculating this characteristic. Also, we demonstrate how to find its upper bound rather simply; hence, suboptimal generalized $\mathcal{H}_2$ control can be designed in the case where the state of the Markov chain is available to the controller.

The remainder of this paper is organized as follows. In Section 2, we introduce the concept of the generalized $\mathcal{H}_2$ norm for continuous-time Markov jump linear systems on a finite horizon and present algorithms for its calculation. Section 3 provides the solution of the suboptimal generalized $\mathcal{H}_2$ control design problem in the cases where the state of the Markov chain is available and unavailable to the controller. In Section 4, we solve the multi-objective control problem. Numerical simulations demonstrating the above results are given in Section 5.

2. GENERALIZED $\mathcal{H}_2$ NORM

Let $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in[t_s,t_f]}, \mathsf{P})$ be a probability space with filtration $(\mathcal{F}_t)_{t\in[t_s,t_f]}$. We denote by $L_2^\Omega([t_s,t_f],\mathbb{R}_2^{n_v})$ the space of $\mathcal{F}_t$-adapted processes $v = \{v(t) \in \mathbb{R}_2^{n_v}, t \in [t_s,t_f]\}$ such that

$$\mathbb{E}\|v\|_2^2 = \mathbb{E}\int\limits_{t_s}^{t_f} v^\top(t)v(t)dt < \infty,$$

where $\mathbb{E}(\cdot)$ indicates the mathematical expectation operator.

On a fixed time interval (finite horizon) $[t_s,t_f]$, we consider a continuous-time linear system with a random structure changing according to the evolution of a stationary Markov chain:

$$\begin{aligned} \dot{x} &= A_{\theta(t)}(t)x + B_{\theta(t)}(t)v, \qquad x(t_s) = x_0, \\ z &= C_{\theta(t)}(t)x, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}_2^{n_x}$ is the system state; $v \in L_2^\Omega([t_s,t_f],\mathbb{R}_2^{n_v})$ is an exogenous disturbance; $z \in \mathbb{R}_2^{n_z}$ is the target output; $\theta(t)$ is a homogeneous continuous-time Markov chain defined by the initial distribution $\pi_j = \mathsf{P}\{\theta(t_s) = j\}$ and the matrix of transition rates $P(\tau) = (p_{ij}(\tau))$, $i,j \in \mathfrak{S} = \{1,\dots,S\}$, where $p_{ij}(\tau)$ is the probability that the system, being in state i at some time instant t, will pass to state j in time τ, i.e. $p_{ij}(\tau) = \mathsf{P}\{\theta(t+\tau) = j|\theta(t) = i\}$. Assume that

$$\mathsf{P}\{\theta(t+\tau) = j|\theta(t) = i\} = \begin{cases} \lambda_{ij}\tau + o(\tau), & i \neq j, \\ 1 + \lambda_{ij}\tau + o(\tau), & i = j, \end{cases}$$

where λ_{ij} are the elements of the stationary intensity matrix Λ with the following properties:

$$\sum_{j=1}^{S}\lambda_{ij}=0,\quad \lambda_{ij}\geqslant 0,\quad \lambda_{ii}<0.$$

Note that a continuous-time homogeneous Markov chain can be defined using the intensity matrix, which is related to the transition rate matrix by $P(\tau)=e^{\Lambda\tau}$ [1].

Let the target output of the system be represented as

$$z=\operatorname{column}(z_1,z_2,\dots,z_M),\quad z_m=C_{m,\theta(t)}(t)x,\quad m=1,\dots,M.$$

We define the generalized ∞ norm of the target output by the relation

$$\|z\|_{g\infty}^2=\sup_{t\in[t_s,t_f]}\max_{m=1,\dots,M}\mathbb{E}\{|z_m(t)|_2^2\},\quad |z_m(t)|_2^2=z_m^\top(t)z_m(t). \tag{2}$$

System (1) generates a linear operator mapping the initial conditions and exogenous disturbance into the target output, i.e., $\mathcal{S}:(x_0,v)\mapsto z$. We define the generalized $\mathcal{H}_2$ norm of system (1) as the norm of the operator $\mathcal{S}$ as follows:

$$\|\mathcal{S}\|_{g2}^2=\sup_{(x_0,v)\neq 0}\frac{\|z\|_{g\infty}^2}{\mathbb{E}\|v\|_2^2+x_0^\top Rx_0}, \tag{3}$$

where $R=R^\top\succ 0$ is a given weight matrix reflecting the relative importance of considering uncertainties in the initial conditions and exogenous disturbances.

Theorem 1. *The generalized $\mathcal{H}_2$ norm of the Markov jump linear system* (1) *on the finite horizon $[t_s,t_f]$ can be calculated as*

$$\|\mathcal{S}\|_{g2}=\sup_{T\in[t_s,t_f]}\gamma(T),$$

where $\gamma(T)$ is the solution of the following semidefinite programming problem with respect to the unknown matrices $Q_l(t)=Q_l^\top(t)\succcurlyeq 0$:

$$\begin{aligned}&\inf\gamma^2\\&\begin{bmatrix}\dot Q_l(t)+A_l^\top(t)Q_l(t)+Q_l(t)A_l(t)+\sum\limits_{j=1}^{S}\lambda_{lj}Q_j(t) & Q_l(t)B_l(t)\\ B_l^\top(t)Q_l(t) & -I\end{bmatrix}\preccurlyeq 0,\quad t\in[t_s,T],\\&\sum_{l=1}^{S}\pi_lQ_l(t_s)-R\preccurlyeq 0,\quad \begin{bmatrix}Q_l(T) & C_{m,l}^\top(T)\\ C_{m,l}(T) & \gamma^2 I\end{bmatrix}\succcurlyeq 0,\quad m=1,\dots,M,\quad l\in\mathfrak{S}.\end{aligned} \tag{4}$$

Let

$$t^*=\arg\sup_{T\in[t_s,t_f]}\gamma(T),$$

and let the matrix functions $Q_{\theta(t)}(t)$ be obtained by solving inequalities (4) *on the horizon $[t_s,t^*]$. Then the worst-case exogenous disturbance $v^*(t)$ and the vector of initial conditions x_0^* are given by*

$$x_0^*=e_{\max}\left(R^{-1}\sum_{l=1}^{S}\pi_lQ_l(t_s)\right),\quad v^*(t)=\begin{cases}B_{\theta(t)}^\top(t)Q_{\theta(t)}(t)x(t), & t\in[t_s,t^*],\\ 0, & t\in(t^*,t_f],\end{cases} \tag{5}$$

where $x(t)$ is the solution of the Cauchy problem for the system

$$\dot x=(A_{\theta(t)}+B_{\theta(t)}B_{\theta(t)}^\top(t)Q_{\theta(t)}(t))x(t),\quad x(t_s)=x_0^*. \tag{6}$$

Remark 1. Note that in the case of deterministic exogenous disturbances $v \in L_2([t_s, t_f], \mathbb{R}_2^{n_v})$, the generalized $\mathcal{H}_2$ norm satisfies the estimate

$$\sup_{(x_0,v)\neq 0} \frac{\|z\|_{g\infty}^2}{\|v\|_{L_2}^2 + x_0^\top R x_0} \leqslant \|\mathcal{S}\|_{g2}, \tag{7}$$

since the domain for calculating the supremum is reduced. Thus, by calculating the generalized $\mathcal{H}_2$ norm for stochastic disturbances (3), one obtains an upper bound for the case of deterministic disturbances (7).

To find the generalized $\mathcal{H}_2$ norm using the matrix inequalities (4), we perform discretization. Let us introduce, e.g., a uniform grid with step h :

$$t_0 = t_s, \qquad t_k = t_{k-1} + h, \qquad k = 1, \ldots, K; \qquad h = \frac{T - t_s}{K}. \tag{8}$$

Then the discrete counterparts of inequalities (4) have the form

$$\begin{gathered}
\begin{bmatrix} Q_{l,k+1} - Q_{l,k} + h\big(A_{l,k}^\top Q_{l,k} + Q_{l,k} A_{l,k} + \sum\limits_{j=1}^{S} \lambda_{lj} Q_{j,k}\big) & hQ_{l,k}B_{l,k} \\ hB_{l,k}^\top Q_{l,k} & -hI \end{bmatrix} \preccurlyeq 0, \\
\sum_{l=1}^{S} \pi_l Q_{l,0} - R \preccurlyeq 0, \quad \begin{bmatrix} Q_{l,K} & C_{m,l,K}^\top \\ C_{m,l,K} & \gamma^2 I \end{bmatrix} \succcurlyeq 0, \quad m = 1, \ldots, M, \quad l \in \mathfrak{S},
\end{gathered} \tag{9}$$

where $A_{l,k} = A_l(t_k)$, $B_{l,k} = B_l(t_k)$, $C_{m,l,k} = C_{m,l}(t_k)$, and $Q_{l,k} = Q_l(t_k)$, $k = 0, \ldots, K-1$.

Corollary 1. *The generalized $\mathcal{H}_2$ norm of the Markov jump linear system* (1) *on the finite horizon* $[t_s, t_f]$ *can be calculated as*

$$\|\mathcal{S}\|_{g2} = \sup_{T \in [t_s, t_f]} \gamma(T),$$

where $\gamma(T)$ is the solution of the semidefinite programming problem

$$\begin{gathered}
\inf \gamma^2 \\
\begin{bmatrix} -\dot{Y}_l(t) + A_l(t)Y_l(t) + Y_l(t)A_l^\top(t) + B_l(t)B_l^\top(t) + \lambda_{ll}Y_l(t) & V_l(t) \\ V_l{}^\top(t) & -W_l(t) \end{bmatrix} \preccurlyeq 0, \\
\begin{bmatrix} Y_l(T) & Y_l(T)C_{m,l}^\top(T) \\ C_{m,l}(T)Y_l(T) & \gamma^2 I \end{bmatrix} \succcurlyeq 0, \quad m = 1, \ldots, M, \quad l \in \mathfrak{S}, \quad t \in [t_s, T], \\
\mathcal{L}\big(Y_1(t_s), \ldots, Y_S(t_s)\big) \succcurlyeq 0,
\end{gathered} \tag{10}$$

where

$$\begin{aligned}
V_l(t) &= \begin{bmatrix} \sqrt{\lambda_{l1}}Y_l(t) & \ldots & \sqrt{\lambda_{l,l-1}}Y_l(t) & \sqrt{\lambda_{l,l+1}}Y_l(t) & \ldots & \sqrt{\lambda_{l,S}}Y_l(t) \end{bmatrix}, \\
W_l(t) &= \operatorname{diag}(Y_1(t), \ldots, Y_{l-1}(t), Y_{l+1}(t), \ldots, Y_S(t)), \\
\mathcal{L}(Y_1(t_s), \ldots, Y_S(t_s)) &= \begin{bmatrix} R & \sqrt{\pi_1}I & \ldots & \sqrt{\pi_S}I \\ \sqrt{\pi_1}I & Y_1(t_s) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \sqrt{\pi_S}I & 0 & \ldots & Y_S(t_s) \end{bmatrix}.
\end{aligned} \tag{11}$$

To find the generalized $\mathcal{H}_2$ norm of system (1) using Corollary 1, it is necessary to solve problem (10) for each time instant $T \in [t_s, t_f]$, which is a computationally intensive process. We formulate a theorem yielding an upper bound for the generalized $\mathcal{H}_2$ norm of system (1) rather easily. It will also be used below to design suboptimal generalized $\mathcal{H}_2$ control in the case where the state of the Markov chain is available to the controller.

Theorem 2. *The generalized $\mathcal{H}_2$ norm of the Markov jump linear system* (1) *on the finite horizon $[t_s, t_f]$ satisfies the inequality*

$$\|\mathcal{S}\|_{g2} \leqslant \gamma,$$

where γ is the solution of the following semidefinite programming problem with respect to the unknown matrices $Y_l = Y_l^\top \succcurlyeq 0$, $l \in \mathfrak{S}$:

$$\begin{aligned}
&\inf \gamma^2 \\
&\begin{bmatrix} -\dot{Y}_l(t) + A_l(t)Y_l(t) + Y_l(t)A_l^\top(t) + B_l(t)B_l^\top(t) + \lambda_{ll}Y_l(t) & V_l(t) \\ V_l^\top(t) & -W_l(t) \end{bmatrix} \preccurlyeq 0, \\
&\begin{bmatrix} Y_l(t) & Y_l(t)C_{m,l}^\top(t) \\ C_{m,l}(t)Y_l(t) & \gamma^2 I \end{bmatrix} \succcurlyeq 0, \quad m = 1, \dots, M, \quad l \in \mathfrak{S}, \quad \in [t_s, t_f], \\
&\mathcal{L}(Y_1(t_s), \dots, Y_S(t_s)) \succcurlyeq 0,
\end{aligned} \tag{12}$$

where $V_l(t)$ and $W_l(t)$ are given by formula (11).

3. CONTROL LAW DESIGN

Consider a linear controlled plant with a random structure and dynamics described by the equations

$$\begin{aligned}
\dot{x} &= A_{\theta(t)}(t)x + B_{\theta(t)}(t)v + B^u_{\theta(t)}(t)u, \qquad x(t_s) = x_0, \\
z &= C_{\theta(t)}(t)x + D_{\theta(t)}(t)u,
\end{aligned} \tag{13}$$

where $x \in \mathbb{R}_2^{n_x}$ is the plant's state; $v(t) \in \mathbb{R}_2^{n_v}$ is a stochastic exogenous disturbance; $z \in \mathbb{R}_2^{n_z}$ is the target output; $u \in \mathbb{R}_2^{n_u}$ is the control vector (input); $\theta(t)$ is a continuous-time homogeneous Markov chain defined by the initial distribution $\pi_j = \mathsf{P}\{\theta(t_s) = j\}$ and the transition rate matrix $P(\tau) = (p_{ij}(\tau))$, $i, j \in \mathfrak{S}$.

3.1. Feedback Control Considering the Markov Chain State

Let us pose the following problem: it is required to design a linear state-feedback controller considering the Markov chain state,

$$u(t) = \Theta_{\theta(t)}(t)x(t), \quad \theta(t) \in \mathfrak{S}, \tag{14}$$

that minimizes the generalized $\mathcal{H}_2$ norm of the closed-loop system

$$\begin{aligned}
\dot{x} &= (A_{\theta(t)}(t) + B^u_{\theta(t)}(t)\Theta_{\theta(t)}(t))x + B_{\theta(t)}(t)v, \quad x(t_s) = x_0, \\
z &= (C_{\theta(t)}(t) + D_{\theta(t)}(t)\Theta_{\theta(t)}(t))x
\end{aligned} \tag{15}$$

(system (13) with the controller (14)).

Substituting the matrices of the closed-loop system (15) into inequalities (10), we arrive at the following result.

Theorem 3. *The gain matrices* $\mathbf{\Theta}(t) = (\Theta_1(t), \dots, \Theta_S(t))$ *of the controllers* (14) *minimizing the generalized* $\mathcal{H}_2$ *norm of system* (13) *are obtained by solving the problem*

$$\|\mathcal{S}\|_{g2} = \inf_{\mathbf{\Theta}(t), t\in[t_s,t_f]} \sup_{T\in[t_s,t_f]} \gamma_{\mathbf{\Theta}}(T), \tag{16}$$

where $\gamma_{\mathbf{\Theta}}(T)$ *is the solution of the semidefinite programming problem*

$$\begin{aligned}
&\inf \gamma^2 \\
&\begin{bmatrix} -\dot{Y}_l + A_l Y_l + Y_l A_l^\top + B_l^u \Theta_l Y_l + Y_l \Theta_l^\top B_l^{u\top} + B_l B_l^\top + \lambda_{ll} Y_l & * \\ V_l^\top & -W_l \end{bmatrix} \preccurlyeq 0, \\
&\begin{bmatrix} Y_l(T) & * \\ C_{m,l}(T) Y_l(T) + D_{m,l}(T) \Theta_l Y_l(T) & \gamma^2 I \end{bmatrix} \succcurlyeq 0, \\
&\mathcal{L}\big(Y_1(t_s), \dots, Y_S(t_s)\big) \succcurlyeq 0, \quad t \in [t_s, T], \quad m = 1, \dots, M, \quad l \in \mathfrak{S}.
\end{aligned} \tag{17}$$

For brevity, the argument t *of matrix functions in the first inequality is omitted, and* $*$ *indicates a symmetric element.*

Problem (16) cannot be solved using the existing apparatus since it is necessary to search for the infimum over all possible controller parameters $\mathbf{\Theta}(t)$. This problem vanishes under the conditions of Theorem 2: the feedback parameters can be calculated within the semidefinite programming problem (12). In this case, it seems reasonable to find a controller minimizing the norm bound, the so-called suboptimal generalized $\mathcal{H}_2$ controller.

Theorem 4. *The gain matrices* $\Theta_l(t)$ *of the controllers* (14) *minimizing the bound of the generalized* $\mathcal{H}_2$ *norm of system* (13) *have the form* $\Theta_l(t) = Z_l(t) Y_l^{-1}(t)$, *where* $Y_l = Y_l^\top \succcurlyeq 0$ *and* Z_l *are obtained by solving the semidefinite programming problem*

$$\begin{aligned}
&\inf \gamma^2 \\
&\begin{bmatrix} -\dot{Y}_l + A_l Y_l + Y_l A_l^\top + B_l^u Z_l + Z_l^\top B_l^{u\top} + B_l B_l^\top + \lambda_{ll} Y_l & * \\ V_l^\top & -W_l \end{bmatrix} \preccurlyeq 0, \\
&\begin{bmatrix} Y_l(t) & * \\ C_{m,l}(t) Y_l(t) + D_{m,l}(t) Z_l(t) & \gamma^2 I \end{bmatrix} \succcurlyeq 0, \\
&\mathcal{L}\big(Y_1(t_s), \dots, Y_S(t_s)\big) \succcurlyeq 0, \quad t \in [t_s, t_f], \quad m = 1, \dots, M, \quad l \in \mathfrak{S}.
\end{aligned} \tag{18}$$

3.2. Feedback Control Independent of the Markov Chain State

Next, we design a controller whose parameters are independent of the Markov chain state:

$$u(t) = \Theta(t) x(t). \tag{19}$$

In this case, the corresponding closed-loop system (system (13) with the controller (19)) takes the form

$$\begin{aligned}
\dot{x} &= \big(A_{\theta(t)}(t) + B^u_{\theta(t)}(t)\Theta(t)\big)x + B_{\theta(t)}(t) v, \qquad x(t_s) = x_0, \\
z &= \big(C_{\theta(t)}(t) + D_{\theta(t)}(t)\Theta(t)\big)x.
\end{aligned} \tag{20}$$

Theorem 5. *The generalized $\mathcal{H}_2$ norm of system (20) satisfies the inequality $\|\mathcal{S}\|_{g2} \leqslant \gamma$ with some positive γ if there exists $\rho > 0$ such that the following LMIs are valid for the matrices $P(t) = P^\top(t)$, $Z(t)$, and $Y_l(t) = Y_l^\top \succcurlyeq 0$, $l \in \mathfrak{S}$:*

$$\begin{gathered}
\begin{bmatrix} -\dot{Y}_l + A_l Y_l + Y_l A_l^\top + B_l^u Z + Z^\top B_l^{u\top} + B_l B_l^\top + \lambda_{ll} Y_l & * & * \\ V_l^\top & -W_l & * \\ \rho Z^\top B_l^{u\top} + Y_l - P & 0 & -2\rho P \end{bmatrix} \preccurlyeq 0, \\
\begin{bmatrix} Y_l(t) & * & * \\ C_{m,l}(t) Y_l(t) + D_{m,l}(t) Z(t) & \gamma^2 I & * \\ P - Y_l & \rho Z^\top(t) D_{m,l}^\top(t) & 2\rho P \end{bmatrix} \succcurlyeq 0, \\
\mathcal{L}(Y_1(t_s), \dots, Y_S(t_s)) \succcurlyeq 0, \qquad t \in [t_s, t_f], \quad m = 1, \dots, M, \quad l \in \mathfrak{S},
\end{gathered} \tag{21}$$

where $\Theta(t) = Z(t) P^{-1}(t)$.

4. MULTI-OBJECTIVE CONTROL

Consider a linear controlled plant with a random structure and dynamics described by the equations

$$\begin{aligned}
\dot{x} &= A_{\theta(t)}(t) x + B_{\theta(t)}(t) v + B_{\theta(t)}^u(t) u, \qquad x(t_s) = x_0, \\
z^{(k)} &= C_{\theta(t)}^{(k)}(t) x + D_{\theta(t)}^{(k)}(t) u, \qquad k = 1, \dots, N,
\end{aligned} \tag{22}$$

where $x \in \mathbb{R}_2^{n_x}$ is the plant's state; $v(t) \in \mathbb{R}_2^{n_v}$ is a stochastic exogenous disturbance; $z^{(k)} \in \mathbb{R}_2^{n_{z_k}}$ are the target outputs; $u \in \mathbb{R}_2^{n_u}$ is the control vector (input); $\theta(t)$ is a continuous-time homogeneous Markov chain defined by the initial distribution $\pi_j = \mathsf{P}\{\theta(t_s) = j\}$ and the transition rate matrix $P(\tau) = (p_{ij}(\tau))$, $i, j \in \mathfrak{S}$.

Each target output (vector) $z^{(k)}$, $k = 1, \dots, N$, is represented as the set of vectors

$$\begin{aligned}
z^{(k)} &= \operatorname{column}\bigl(z_1^{(k)}, z_2^{(k)}, \dots, z_{M_k}^{(k)}\bigr), \\
z_m^{(k)} &= C_{m,\theta(t)}^{(k)}(t) x + D_{m,\theta(t)}^{(k)}(t) u, \qquad m = 1, \dots, M_k.
\end{aligned}$$

Assume that the impact of the exogenous disturbance on the kth target output is characterized by the performance criterion

$$J_k^2 = \sup_{(x_0, v) \neq 0} \frac{\|z^{(k)}\|_{g\infty}^2}{\mathbb{E}\|v\|_2^2 + x_0^\top R x_0}, \tag{23}$$

i.e., it represents the generalized $\mathcal{H}_2$ norm. Therefore, it is possible to formulate and solve a multi-objective control problem: $\min\{J_1, \dots, J_N\}$. Following [16], we define the auxiliary performance criterion J_α; then, according to Theorem 4.1 [16], Pareto optimal solutions of the above multi-objective problem can be obtained by solving the problem

$$\begin{gathered}
\min_u J_\alpha, \quad J_\alpha = \max_{k=1,\dots,N} \frac{J_k}{\alpha_k}, \\
\alpha \in \mathcal{A} := \left\{ \alpha = (\alpha_1, \dots, \alpha_N) : \alpha_k > 0, \sum_{k=1}^N \alpha_k = 1 \right\},
\end{gathered} \tag{24}$$

where the performance criterion J_α represents the Germeier convolution and, moreover, is the generalized $\mathcal{H}_2$ norm of the system

$$\begin{aligned} \dot{x} &= \big(A_{\theta(t)}(t) + B^u_{\theta(t)}(t)\Theta_{\alpha,\theta(t)}(t)\big)x + B_{\theta(t)}(t)v, \qquad x(t_s) = x_0, \\ \zeta &= \big(\mathcal{C}_{\theta(t)}(t) + \mathcal{D}_{\theta(t)}(t)\Theta_{\alpha,\theta(t)}(t)\big)x, \end{aligned} \tag{25}$$

where

$$\zeta = \operatorname{column}\Big(\alpha_1^{-1}z_1^{(1)}, \dots, \alpha_1^{-1}z_{M_1}^{(1)}, \alpha_2^{-1}z_1^{(2)}, \dots, \alpha_2^{-1}z_{M_2}^{(2)}, \dots, \alpha_N^{-1}z_1^{(N)}, \dots, \alpha_N^{-1}z_{M_N}^{(N)}\Big),$$

$$\mathcal{C}_{\theta(t)} = \operatorname{column}\Big(\alpha_1^{-1}C^{(1)}_{\theta(t)}, \dots, \alpha_N^{-1}C^{(N)}_{\theta(t)}\Big), \quad \mathcal{D}_{\theta(t)} = \operatorname{column}\Big(\alpha_1^{-1}D^{(1)}_{\theta(t)}, \dots, \alpha_N^{-1}D^{(N)}_{\theta(t)}\Big).$$

A controller $u(t) = \Theta_{\alpha,\theta(t)}(t)x(t)$, $\alpha \in \mathcal{A}$, will be called a multi-objective optimal generalized $\mathcal{H}_2$ controller if it ensures the minimum possible values of the generalized $\mathcal{H}_2$ norm of system (25). Based on Theorem 4, we arrive at the following statement.

Theorem 6. *The gain matrices $\Theta_{\alpha,l}(t)$ of the Pareto suboptimal controllers in terms of the performance criteria J_k, $k = 1, \dots, N$, have the form $\Theta_{\alpha,l}(t) = Z_l(t)Y_l^{-1}(t)$, where $Y_l = Y_l^\top \succcurlyeq 0$ and Z_l are obtained by solving the semidefinite programming problem*

$$\begin{aligned} &\inf \gamma^2 \\ &\begin{bmatrix} -\dot{Y}_l + A_lY_l + Y_lA_l^\top + B_l^uZ_l + Z_l^\top B_l^{u\top} + B_lB_l^\top + \lambda_{ll}Y_l & * \\ V_l^\top & -W_l \end{bmatrix} \preccurlyeq 0, \\ &\begin{bmatrix} Y_l(t) & * \\ C^{(k)}_{m,l}(t)Y_l(t) + D^{(k)}_{m,l}(t)Z_l(t) & \alpha_k^2\gamma^2 I \end{bmatrix} \succcurlyeq 0, \qquad \begin{aligned} m &= 1, \dots, M, \\ k &= 1, \dots, N, \end{aligned} \\ &\mathcal{L}\big(Y_1(t_s), \dots, Y_S(t_s)\big) \succcurlyeq 0, \qquad t \in [t_s, t_f], \qquad l \in \mathfrak{S}. \end{aligned} \tag{26}$$

5. NUMERICAL SIMULATIONS

As an illustration of the above results, we consider a continuous-time linear system with Markov jumps between two states:

$$\begin{aligned} \dot{x} &= A_{\theta(t)}(t)x + B_{\theta(t)}(t)v + B^u_{\theta(t)}(t)u, \qquad x(t_s) = x_0, \\ z^{(k)} &= C^{(k)}_{\theta(t)}(t)x + D^{(k)}_{\theta(t)}(t)u, \qquad k = 1, 2, \end{aligned} \tag{27}$$

with the matrices

$$A_1 = \begin{bmatrix} 1.2 & -2.0 \\ 0.1 & 1.1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.2 & 0.0 \\ 2.0 & 0.3 \end{bmatrix}, \quad B_1 = B_1^u = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad B_2 = B_2^u = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

$$C_1^{(1)} = C_2^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad D_1^{(1)} = D_2^{(1)} = 0, \quad C_1^{(2)} = C_2^{(2)} = 0, \quad D_1^{(2)} = D_2^{(2)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

This system is considered on the horizon $[0, 2]$, the parameters of the Markov chain and the weight matrix are given by

$$\pi = \begin{bmatrix} 0.3 \\ 0.7 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} -10 & 10 \\ 5 & -5 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

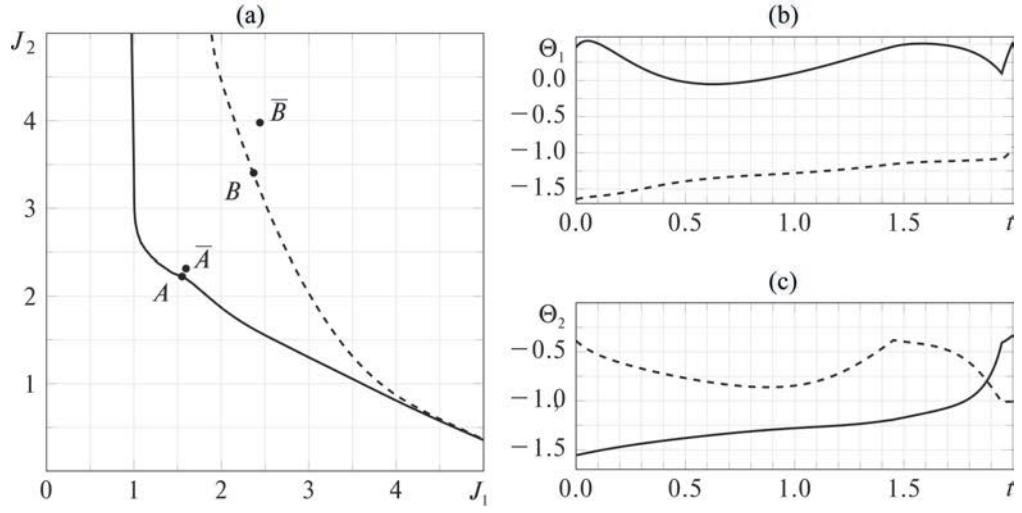

Fig. 1. The Pareto set on the criteria plane (J_1, J_2).

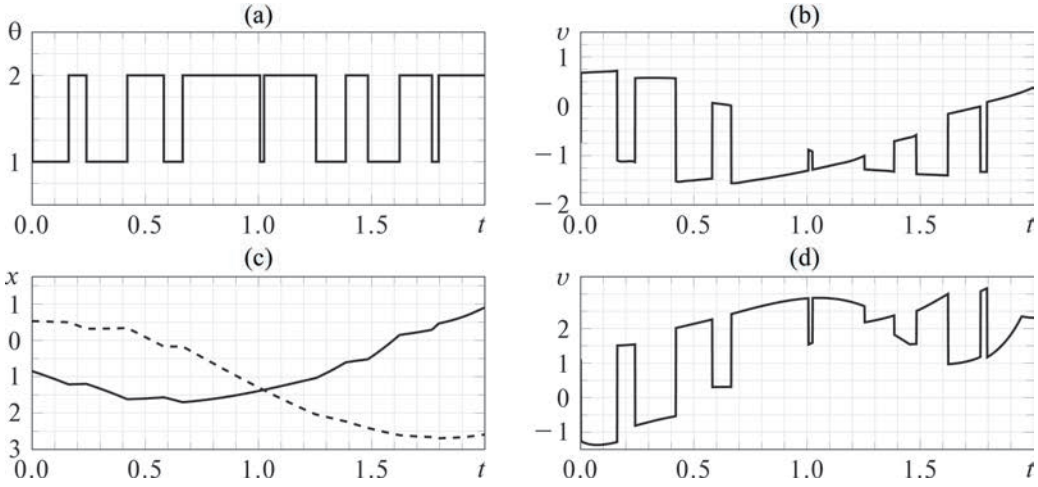

Fig. 2. An example of a Markov process realization.

We introduce two performance criteria J_1 and J_2, which are the generalized $\mathcal{H}_2$ norms of the system with respect to the target outputs $z^{(1)}$ and $z^{(2)}$, respectively:

$$J_1^2 = \sup_{(x_0,v)\neq 0} \frac{\|z^{(1)}\|_{g\infty}^2}{\mathbb{E}\|v\|_2^2 + x_0^\top R x_0}, \qquad J_2^2 = \sup_{(x_0,v)\neq 0} \frac{\|z^{(2)}\|_{g\infty}^2}{\mathbb{E}\|v\|_2^2 + x_0^\top R x_0}. \tag{28}$$

Let us perform discretization on the horizon $[0, 2]$ with a step of $h = 0.001$ and calculate the generalized $\mathcal{H}_2$ norm of system (27) without control. Since the target output is $z^{(2)} = 0$, the generalized $\mathcal{H}_2$ norm of the system coincides with the criterion J_1. The value $\|\mathcal{S}\|_{g2} = 5.7882$ was obtained by solving inequalities (10).

The controllers $\Theta_{\alpha,l}(t) = [\Theta_{\alpha,l}^1, \Theta_{\alpha,l}^2]$, $l = 1, 2$, were designed using Theorem 4, and the corresponding values of the criteria J_1 and J_2 were calculated using Corollary 1. In Fig. 1a, the solid line depicts the Pareto optimal curve on the criteria plane (J_1, J_2). Point $A(1.5509; 2.2492)$ corresponds to the convolution parameter $\alpha = 0.4$. Figures 1b and 1c show the graphs of the Pareto optimal gains $\Theta_{\alpha,l}(t)$ depending on time: the solid curve corresponds to the gain $\Theta_{\alpha,l}^1(t)$ whereas the dashed one to the gain $\Theta_{\alpha,l}^2(t)$.

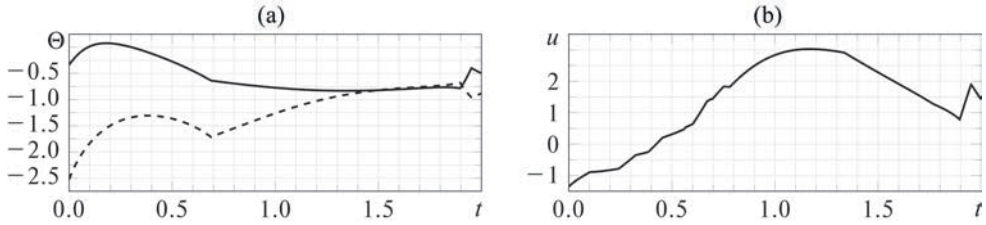

Fig. 3. The coefficients of the gain matrix $\Theta(t)$ and control input $u(t)$.

Next, Fig. 2 presents an example of a Markov process realization: the graphs of the Markov chain state (Fig. 2a), the worst-case disturbances (Fig. 2b), the components x_1 and x_2 (curve) of the system state vector (the solid and dashed curves in Fig. 2c, respectively), and the control input (Fig. 2d).

Note that the gains can be considered slowly varying on the horizon selected. Therefore, it is interesting to compare the values of the criteria under the suboptimal dynamic controller and the static controller corresponding to the average values $\overline{\Theta}_1(t) \equiv [0.2402;\ -1.2928]$ and $\overline{\Theta}_2(t) \equiv [-1.2366;\ -0.6797]$ (see point $\overline{A}(1.5959; 2.3158)$ in Fig. 1a). According to the data presented, the losses in control performance can be considered acceptable, and they are compensated for by the relatively simple-to-implement static controller.

5.1. Control Independent of the Markov Chain State

Now we design a controller whose parameters are independent of the state of the Markov process (19). To solve inequalities (21), let us choose the parameter $\rho = 0.2$, leaving the other simulation parameters unchanged. In Fig. 1a, the upper bound of the Pareto optimal front is plotted by the dashed curve; point $B(2.3697; 3.4030)$ corresponds to the criteria values for $\alpha = 0.4$. For the chosen α, Fig. 3a shows the graphs of the Pareto optimal gains $\Theta_\alpha(t)$ depending on time (the solid curve corresponds to the gain $\Theta^1_\alpha(t)$ whereas the dashed one to the gain $\Theta^2_\alpha(t)$). In Fig. 3b, the control graph for a Markov process realization is presented.

Similar to the previous case, we analyze the behavior of the performance criteria if the dynamic controller is replaced by the static one corresponding to the average values $\overline{\Theta}(t) \equiv [-0.5538;\ -1.2106]$ (see point $\overline{B}(2.4413; 3.9778)$ in Fig. 1a). Clearly, the resulting values of the criteria are worse, but the static controller is simpler to implement, so this approach can be considered justified.

6. CONCLUSIONS

For linear systems with a random structure on a finite horizon, the concept of the generalized $\mathcal{H}_2$ norm has been introduced, and algorithms for its calculation have been presented based on solving both coupled matrix Riccati differential equations and systems of LMIs. Suboptimal generalized $\mathcal{H}_2$ dynamic linear state-feedback control has been designed in the cases where the state of the Markov chain is available and unavailable to the controller. Also, it has been demonstrated how to solve multi-objective control problems if the criteria are generalized $\mathcal{H}_2$ norms.

FUNDING

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (project FSWR-2023-0034) and by the Scientific and Educational Mathematical Center “Mathematics of Future Technologies.”

APPENDIX

Before proceeding to the proofs of the above theorems, for convenience, we introduce the notation

$$\mathbf{X} = (X_1, X_2, \dots, X_S), \qquad X_l(t) = X_l^\top(t) \succcurlyeq 0, \qquad l \in \mathcal{S}, \tag{A.1}$$

$$\mathcal{R}_l(\mathbf{X}) = \dot{X}_l(t) + A_l^\top(t)X_l(t) + X_l(t)A_l(t) + X_l(t)B_l(t)B_l^\top(t)X_l(t) + \sum_{j=1}^{S}\lambda_{lj}X_j(t) \tag{A.2}$$

and establish an auxiliary result.

Lemma 1. *Let* $\mathbf{X} = (X_1, X_2, \dots, X_S)$ *be the solution of the equations*

$$\mathcal{R}_l(\mathbf{X}) + M_l = 0, \quad X_l(t_f) = X_l^0, \quad l \in \mathcal{S}, \qquad t \in [t_s, t_f], \tag{A.3}$$

and let $\mathbf{Y} = (Y_1, Y_2, \dots, Y_S)$ *be the solution of the equations*

$$\mathcal{R}_l(\mathbf{Y}) + N_l = 0, \quad X_l(t_f) = Y_l^0, \quad l \in \mathcal{S}, \quad t \in [t_s, t_f], \tag{A.4}$$

where $0 \preccurlyeq M_l(t) \preccurlyeq N_l(t)$ *and* $0 \preccurlyeq X_l^0 \preccurlyeq Y_l^0$, $l \in \mathcal{S}$, $t \in [t_s, t_f]$. *Then*

$$X_l(t) \preccurlyeq Y_l(t), \quad l \in \mathcal{S}, \quad t \in [t_s, t_f]. \tag{A.5}$$

Proof of Lemma 1. Consider the system of Lyapunov equations

$$\begin{aligned}\dot{P}_l + (A_l + B_lB_l^\top X_l)^\top P_l + P_l(A_l + B_lB_l^\top X_l) + \sum_{j=1}^{S}\lambda_{lj}P_j \\ + (X_l - Y_l)B_lB_l^\top(X_l - Y_l) + N_l - M_l = 0\end{aligned} \tag{A.6}$$

on the time interval $[t_s, t_f]$ with the boundary conditions $P_l(t_f) = Y_l^0 - X_l^0$, $l \in \mathcal{S}$. This equation has a unique solution $P_l(t)$, $l \in \mathcal{S}$, with $P_l(t) \succcurlyeq 0$ since $(X_l - Y_l)B_lB_l^\top(X_l - Y_l) + N_l - M_l \succcurlyeq 0$ and $P_l(t_f) \succcurlyeq 0$ [1, 25]. Note that $P_l(t) = Y_l(t) - X_l(t)$ is the solution of equation (A.6), which finally gives (A.5).

Proof of Theorem 1. We write the functional (3) in the following form:

$$\|\mathcal{S}\|_{g2} = \sup_{T\in[t_s,\,t_f]}\max_{m=1,\dots,M}\gamma_m(T), \qquad \gamma_m^2(T) = \sup_{(x_0,v)\neq 0}\frac{\mathbb{E}|z_m(T)|_2^2}{\mathbb{E}\|v\|_2^2 + x_0^\top Rx_0}.$$

Due to the linearity of the operator $\mathcal{S}$, the last equality can be written as

$$\sup_{(x_0,v)\neq 0}\mathbb{E}\left\{|z_m(T)|_2^2 - \gamma^2\left(\|v\|_2^2 + x_0^\top Rx_0\right)\right\} = 0. \tag{A.7}$$

(Hereinafter, for brevity, the argument and index $\gamma_m(T)$ are omitted.) We introduce the Bellman function at a time instant t :

$$V(t, x_t, l) = \sup_v \mathbb{E}\left\{z_m^\top(T)z_m(T) - \gamma^2\int_t^T v^\top(\tau)v(\tau)d\tau\Big|x(t) = x_t,\ \theta(t) = l\right\}. \tag{A.8}$$

Then the relation (A.7) becomes

$$\sup_{x_0\neq 0}\mathbb{E}\left\{V(t_s, x_0, \theta_0) - \gamma^2 x_0^\top Rx_0\right\} = 0, \qquad \theta_0 = \theta(t_s). \tag{A.9}$$

Let us calculate $V(t_s, x_0, \theta_0)$ using the stochastic Bellman equation [26]

$$\begin{aligned} \max_v \{\mathcal{L}^v V(t, x(t), l) - \gamma^2 v^\top(t) v(t)\} = 0, \\ V(T, x(T), l) = x^\top(T) C_{m,l}^\top(T) C_{m,l}(T) x(T), \end{aligned} \tag{A.10}$$

where the infinitesimal generator $\mathcal{L}^v$ has the form

$$\mathcal{L}^v g(t, x, l) = \frac{\partial g(t, x, l)}{\partial t} + (A_l(t)x + B_l(t)v)^\top \nabla_x g(t, x, l) + \sum_{j=1}^{S} \lambda_{lj} g(t, x, j). \tag{A.11}$$

We seek a solution in the class of quadratic forms $V(t, x(t), l) = x^\top(t) X_l(t) x(t)$, $X_l(t) = X_l^\top(t) \succcurlyeq 0$. From this point onwards, for brevity again, the arguments of the functions $x(t)$ and $v(t)$ will be omitted. Substituting the infinitesimal generator (A.11) into equation (A.10) yields

$$\max_v \left\{ x^\top \dot{X}_l(t) x + 2(A_l(t)x + B_l(t)v)^\top X_l(t) x + \sum_{j=1}^{S} \lambda_{lj} x^\top X_j(t) x - \gamma^2 v^\top v \right\} = 0. \tag{A.12}$$

Since the expression in curly braces is a concave functional in the variable v, a solution of this problem does exist. To obtain it, we find the stationary point v^* :

$$v^* = \gamma^{-2} B_l^\top(t) X_l(t) x. \tag{A.13}$$

Omitting the arguments of matrix functions and substituting (A.13) into (A.12), after straightforward simplifications, we get

$$x^\top \left(\dot{X}_l + A_l^\top X_l + X_l A_l + \gamma^{-2} X_l B_l B_l^\top X_l + \sum_{j=1}^{S} \lambda_{lj} X_j \right) x = 0. \tag{A.14}$$

Equality (A.14) must hold for any value of x; therefore, one arrives at the differential matrix equation

$$\dot{X}_l + A_l^\top X_l + X_l A_l + \gamma^{-2} X_l B_l B_l^\top X_l + \sum_{j=1}^{S} \lambda_{lj} X_j = 0 \tag{A.15}$$

with the boundary conditions $X_l(T) = C_{m,l}^\top(T) C_{m,l}(T)$. Then, at the initial time, we have $V(t_s, x_0, \theta_0) = x_0^\top X_{\theta_0}(t_s) x_0$, and the calculation of (A.9) reduces to

$$\sup_{x_0 \neq 0} x_0^\top \left(\sum_{l=1}^{S} \pi_l X_l(t_s) - \gamma^2 R \right) x_0. \tag{A.16}$$

This expression is a quadratic form in the variable x_0 and reaches its maximum at the point

$$x_0^* = e_{\max} \left(R^{-1} \sum_{l=1}^{S} \pi_l X_l(t_s) \right) \tag{A.17}$$

under the condition

$$\sum_{l=1}^{S} \pi_l X_l(t_s) - \gamma^2 R \preccurlyeq 0; \tag{A.18}$$

in this case, the value of γ is given by

$$\gamma = \lambda_{\max}^{1/2}\left(R^{-1}\sum_{l=1}^{S}\pi_l X_l(t_s)\right). \tag{A.19}$$

Writing condition (A.19) using an LMI with the minimum value of γ, after the change of variables $X_l(t) = \gamma^2 Q_l(t)$, we obtain the following optimization problem on the interval $[t_s, T]$ for calculating $\gamma_m(T)$:

$$\begin{aligned}&\inf\gamma^2\\ &\mathcal{R}_l(\mathbf{Q}) = 0,\ \ \sum_{l=1}^{S}\pi_l Q_l(t_s) - R \preccurlyeq 0,\ \ Q_l(T) = \gamma^{-2}C_{m,l}^{\top}(T)C_{m,l}(T),\ \ l\in\mathfrak{S}.\end{aligned} \tag{A.20}$$

Next, we show that $\gamma_m(T)$ can be found by solving the following semidefinite programming problem on the interval $[t_s, T]$:

$$\begin{aligned}&\inf\gamma^2\\ &\mathcal{R}_l(\mathbf{Q}) \preccurlyeq 0,\ \ \sum_{l=1}^{S}\pi_l Q_l(t_s) - R \preccurlyeq 0,\ \ Q_l(T) \succcurlyeq \gamma^{-2}C_{m,l}^{\top}(T)C_{m,l}(T),\ \ l\in\mathfrak{S}.\end{aligned} \tag{A.21}$$

Let γ_1 be the solution of problem (A.20), and let γ_2 and $\mathbf{Q} = (Q_1, Q_2, \ldots, Q_S)$ be obtained by solving (A.21). The solution of (A.20) is a solution of (A.21) if the corresponding inequalities hold as equalities, therefore $\gamma_2 \leqslant \gamma_1$.

Assume that $\gamma_2 < \gamma_1$. Let $\mathbf{X} = (X_1, X_2, \ldots, X_S)$ be the solution of the equations $\mathcal{R}_l(\mathbf{X}) = 0$, $l \in \mathfrak{S}$, on the interval $[t_s, T]$ with the boundary conditions $X_l(T) = \gamma_2^{-2}C_{m,l}^{\top}(T)C_{m,l}(T)$. Then, due to Lemma 1, we have

$$X_l(t) \preccurlyeq Q_l(t), \quad l\in\mathcal{S}, \quad t\in[t_s, T]. \tag{A.22}$$

Since $\pi_l \geqslant 0$, $l \in \mathfrak{S}$, the relations

$$\sum_{l=1}^{S}\pi_l X_l(t_s) \preccurlyeq \sum_{l=1}^{S}\pi_l Q_l(t_s) \preccurlyeq R \tag{A.23}$$

hold at the initial time instant. Hence, $\mathbf{X}$ is a solution of the problem (A.20) for $\gamma_2 < \gamma_1$, which contradicts the condition $\gamma_1^2 = \inf\gamma^2$; consequently, the assumption $\gamma_2 < \gamma_1$ is false, and $\gamma_1 = \gamma_2$.

Note that for a fixed m, solving problem (A.21) yields $\gamma_m(T)$. To find $\max\limits_{m=1,\ldots,M}\gamma_m(T)$, it is necessary to supplement inequalities (A.21) with enumeration over all possible values of m, i.e., solve (A.21) for $m = 1, \ldots, M$. Applying the Schur complement lemma to inequalities (A.21), we finally arrive at conditions (4). The proof of Theorem 1 is complete.

Proof of Corollary 1. In inequalities (A.21), we make the change of variables $Y_l(t) = Q_l^{-1}(t)$, $l \in \mathfrak{S}$, and multiply the first and third inequalities by $Y_l(t)$ on the left and right. After that, using the Schur complement lemma, we get the expressions (10). The proof of Corollary 1 is complete.

Proof of Theorem 2. Let the generalized $\mathcal{H}_2$ norm of system (1), $\|\mathcal{S}\|_{g2}$, be achieved at a time instant t^*, i.e.,

$$\|\mathcal{S}\|_{g2} = \sup_{T\in[t_s,t_f]}\gamma(T), \quad t^* = \arg\sup_{T\in[t_s,t_f]}\gamma(T),$$

where $\gamma(T)$ is the solution of problem (10). Thus, for $T = t^*$, we obtain $\inf\gamma^2 = \gamma^2(t^*) = \|\mathcal{S}\|_{g2}^2$ in problem (10).

Let solving problem (12) yield a value $\widehat{\gamma}$ and matrices $\widehat{Y}_l(t)$, $t \in [t_s, t_f]$. Assume that $\widehat{\gamma} < \|\mathcal{S}\|_{g2}$; then for $T = t^*$ and $Y_l(t) = \widehat{Y}_l(t)$, $t \in [t_s, t^*]$, problem (10) has a solution $\widehat{\gamma}^2 < \|\mathcal{S}\|_{g2}^2$, which contradicts the condition $\inf \gamma^2 = \|\mathcal{S}\|_{g2}^2$. Hence, the assumption $\widehat{\gamma} < \|\mathcal{S}\|_{g2}$ is false, and $\|\mathcal{S}\|_{g2} \leqslant \widehat{\gamma}$. The proof of Theorem 2 is complete.

Proof of Theorem 4. We substitute the matrices of the closed-loop system (15) into inequalities (12) and apply the changes $\Theta_l(t)Y_l(t) = Z_l(t)$, making the inequalities linear. These transformations lead to inequalities (18). The proof of Theorem 4 is complete.

Proof of Theorem 5. Let us utilize the approach outlined in [27]. Multiplying the first and second inequalities of (21) on the left and right by

$$\begin{bmatrix} I & 0 & \rho^{-1}(Y_l - P)P^{-1} \\ 0 & I & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} I & 0 & \rho^{-1}(Y_l - P)P^{-1} \\ 0 & I & 0 \end{bmatrix}^\top,$$

respectively, we obtain

$$\begin{bmatrix} -\dot{Y}_l + (A_l + B_l^u\Theta)Y_l + Y_l(A_l + B_l^u\Theta)^\top + B_lB_l^\top + \lambda_{ll}Y_l & V_l \\ V_l^\top & -W_l \end{bmatrix} \preccurlyeq 0,$$
$$\begin{bmatrix} Y_l & Y_l(t)(C_{m,l} + D_{m,l}\Theta)^\top \\ (C_{m,l} + D_{m,l}\Theta)Y_l & \gamma^2 I \end{bmatrix} \succcurlyeq 0. \tag{A.24}$$

Together with the third inequality of (21), they are the conditions for calculating the bound of the generalized $\mathcal{H}_2$ norm of system (20) using Theorem 2. The proof of Theorem 5 is complete.

REFERENCES

1. Costa, O.L., Fragoso, M.D., and Todorov, M.G., *Continuous-Time Markov Jump Linear Systems*, Springer, 2014.
2. Stoica, A.M. and Stoicu, S.C., H_∞ State-Feedback Control of Multi-Agent Systems with Data Packet Dropout in the Communication Channels: A Markovian Approach, *Entropy*, 2022, vol. 24, no. 12, pp. 1734.
3. Xiaowu, M., Baojie, Z., and Kai, L., $L_2 - L_\infty$ Containment Control of Multi-Agent Systems with Markovian Switching Topologies and Non-Uniform Time-Varying Delays, *IET Control Theory & Applications*, 2014, vol. 8, no. 10, pp. 863–872.
4. Yan, Z., Sang, C., Fang, M., and Zhou, J., Energy-to-Peak Consensus for Multi-Agent Systems with Stochastic Disturbances and Markovian Switching Topologies, *Transactions of the Institute of Measurement and Control*, 2018, vol. 40, no. 16, pp. 4358–4368.
5. Abdollahi, F. and Khorasani, K., A Decentralized Markovian Jump H_∞ Control Routing Strategy for Mobile Multi-Agent Networked Systems, *IEEE Transactions on Control Systems Technology*, 2011, vol. 19, no. 2, pp. 269–283.
6. Wan, H., Luan, X., Karimi, H.R., and Liu, F., Dynamic Self-Triggered Controller Codesign for Markov Jump Systems, *IEEE Transactions on Automatic Control*, 2021, vol. 66, no. 3, pp. 1353–1360.
7. Kats, I.Ia. and Krasovskii, N.N., On Stability of Systems with Random Parameters, *Journal of Applied Mathematics and Mechanics*, 1960, vol. 24, no. 5, pp. 1225–1246.
8. Kazakov, I.E. and Artem'ev, V.M., *Optimizatsiya dinamicheskikh sistem sluchainoi struktury* (Optimization of Dynamic Systems with Random Structure), Moscow: Nauka, 1980.
9. Katz, I.Ya., *Metod funktsii Lyapunova v zadachakh ustoichivosti i stabilizatsii sistem sluchainoi struktury* (The Lyapunov Function Method in Stability and Stabilization Problems of Systems with Random Structure), Yekaterinburg: Ural State Transport University, 1998.

10. Mariton, M., *Jump Linear Systems in Automatic Control*, Taylor & Francis, 1990.

11. Hinrichsen, D. and Pritchard, A.J., Stochastic H_∞, *SIAM J. Control*, 1998, vol. 36, no. 5, pp. 1504–153.

12. Petersen, I.R., Ugrinovskii, V.A., and Savkin, A.V., *Robust Control Design Using H_∞ Methods*, London: Springer, 2000.

13. Costa, O. and Fragoso, M., A Separation Principle for the H_2-Control of Continuous-Time Infinite Markov Jump Linear Systems with Partial Observations, *J. Math. Anal. Appl.*, 2007, vol. 331, pp. 97–120.

14. De Oliveira, A.M. and Costa, O.L.V., Mixed H_2/H_∞ State-Feedback Control of Continuous-Time Markov Jump Systems with Partial Observations of the Markov Chain, *IFAC-PapersOnLine*, 2020, vol. 53, no. 2, pp. 2249–2254.

15. Wilson, D.A., Convolution and Hankel Operator Norms for Linear Systems, *IEEE Trans. Autom. Control*, 1989, vol. 34, pp. 94–97.

16. Balandin, D.V. and Kogan, M.M., Pareto Optimal Generalized $\mathcal{H}_2$-Control and Vibroprotection Problems, *Autom. Remote Control*, 2017, vol. 78, no. 8, pp. 1417–1429.

17. Balandin, D.V., Biryukov, R.S., and Kogan, M.M., Optimal Control of Maximum Output Deviations of a Linear Time-Varying System on a Finite Horizon, *Autom. Remote Control*, 2019, vol. 80, no. 10, pp. 1783–1802.

18. De Oliveira, A.M., Costa, O.L.V., Gabriel, G.W., and Barros Dos Santos, S.R., Energy-to-Peak Reduced Order Filtering for Continuous-Time Markov Jump Linear Systems with Partial Information on the Jump Parameter, *IEEE Access*, 2022, vol. 10, pp. 79124–79133.

19. Costa, O.L.V., De Oliveira, A.M., Gabriel, G.W., and Barros Dos Santos, S.R., Energy-to-Peak Static Output Control for Continuous-Time Hidden Markov Jump Linear Systems, *IFAC-PapersOnLine*, 2023, vol. 56, no. 2, pp. 8141–8146.

20. Todorov, M.G., A New Approach to the Energy-to-Peak Performance Analysis of Continuous-Time Markov Jump Linear Systems, *IEEE Control Systems Letters*, 2024, vol. 8, pp. 1024–1029.

21. Xu, Z., Wu, Z.-G., Su, H., Shi, P., and Que, H., Energy-to-Peak Filtering of Semi-Markov Jump Systems with Mismatched Modes, *IEEE Transactions on Automatic Control*, 2020, vol. 65, no. 10, pp. 4356–4361.

22. Liu, H., Sun, F., and Sun, Z., Reduced-Order Filtering with Energy-to-Peak Performance for Discrete-Time Markovian Jumping Systems, *IMA J. Math. Control Inform.*, 2004, vol. 21, no. 2, pp. 143–158.

23. Feng, J., and Han, K., Robust Full- and Reduced-Order Energy-to-Peak Filtering for Discrete-Time Uncertain Linear Systems, *Signal Processing*, 2015, vol. 108, pp. 183–194.

24. Zhang, Zh., Zhang, Z., and Yang, S., Robust Reduced-Order l_2-l_∞ Filtering for Network-Based Discrete-Time Linear Systems, *Signal Processing*, 2015, vol. 109, pp. 110–118.

25. Fragoso, M. and Baczynski, J., Lyapunov Coupled Equations for Continuous-Time Infinite Markov Jump Linear Systems, *Journal of Mathematical Analysis and Applications*, 2002, vol. 274, pp. 319–335.

26. Fragoso, M.D. and Hemerly, E.M., Optimal Control for a Class of Noisy Linear Systems with Markovian Jumping Parameters and Quadratic Cost, *Int. J. Syst. Sci.*, 1991, vol. 22, no. 12, pp. 2553–2561.

27. Zhou, J., Park, J.H., and Ma, Q., Non-Fragile Observer-Based H_∞ Control for Stochastic Time-Delay Systems, *Applied Mathematics and Computation*, 2016, vol. 291, pp. 69–83.

This paper was recommended for publication by P.V. Pakshin, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 12, pp. 1168–1179.
© The Author(s), 2025 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2025.
Russian Text © The Author(s), 2025, published in Avtomatika i Telemekhanika, 2025, No. 12, pp. 104–118.

STOCHASTIC SYSTEMS

On a Problem Related to the Time of First Reaching a Given Level by a Random Process

S. L. Semakov[*,**,a]

[*] *National Research University Higher School of Economics, Moscow, Russia*
[**] *Moscow Institute of Physics and Technology, Dolgoprudny, Russia*
e-mail: [a] *slsemakov@yandex.ru*

Received April 16, 2025
Revised June 17, 2025
Accepted October 3, 2025

Abstract—This paper considers the problem of estimating the probability of the following event: a continuous random process will first reach a given level at some time from a given variation interval of the independent variable. The general results obtained previously are specified for a smooth Gaussian process. The estimates are calculated for different values of process parameters, and the corresponding numerical results are presented.

Keywords: probability, random process, level crossing

DOI: 10.7868/S1608303225120055

1. INTRODUCTION. PROBLEM STATEMENT AND PREVIOUS RESULTS

Let $\xi(x)$ be a random process continuous with probability 1, and let y be a given number. As the domain of the process $\xi(t)$ we consider two intervals, namely, a) a half-interval $(x_0, x'']$ or b) a closed interval $[x_0, x'']$. In case a), by assumption,

$$\lim_{x \to x_0} \mathbf{P}\{\xi(x) > y\} = 1, \tag{1}$$

and x_0 can be either a finite number or $-\infty$. In case b), we suppose $\mathbf{P}\{\xi(x_0) > y\} = 1$.

Consider an arbitrary value $x' \in (x_0, x'')$. Let us define the events

$$Z = \{\exists\, \tilde{x} \in (x', x'') \;\; \forall x \in (x_0, \tilde{x}) \;\; \xi(x) > y,\, \xi(\tilde{x}) = y\}$$

and

$$L = \{\exists\, \hat{x} \in (x_0, x''] \;\; \forall x \in (x_0, \hat{x}) \;\; \xi(x) > y\}.$$

Event L means that at the initial time, the trajectory $\xi(x)$ is above level y; and event Z means that level y will be first reached by the trajectory at some time within the interval (x', x'') (see Fig. 1). It is required to find the conditional probability $\mathbf{P}\{Z|L\}$ of event Z given the occurrence of event L.

This problem is a special case of the one posed in [1]. For non-Markov smooth processes, it was studied, in particular, in the author's publications [2–5]; related problems were considered in many well-known works, e.g., [6–13]. A more detailed bibliography can be found in [5]. For diffusion Markov processes, this problem can be reduced to solving a mixed problem for a partial differential equation, which was shown back in [1]. In this paper, as in [2–5], we consider the case of a non-Markov smooth random process $\xi(x)$. Note that, in contrast to [3–5], a somewhat different procedure for forming lower estimates of the desired probability was proposed in [2]. This paper is a continuation of [2].

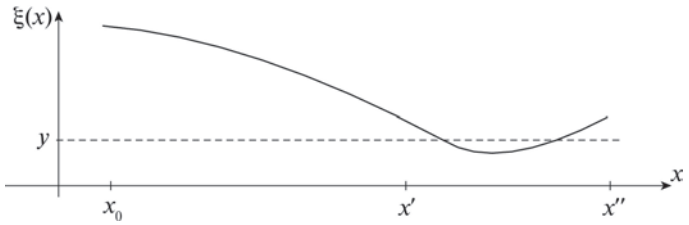

Fig. 1. Some realization of the process $\xi(x)$.

In applications, the above problem arises when investigating various stochastic systems; for example, see [14]. For instance, the probabilistic estimation of the accuracy and safety of aircraft landing reduces to this problem [15–19], which confirms its practical importance.

Following [10], we denote by $G_y(x_0, x'')$ the set of scalar functions continuous on $[x_0, x'']$ or $(x_0, x'']$ (depending on the domain of the process $\xi(x)$ selected) that are not identically equal to y on any subinterval of (x_0, x''). For functions from $G_y(x_0, x'')$, we define the concepts of a crossing of level y, a touching of level y, an upcrossing of level y [10], and a downcrossing of level y [10], as done in [10].

If[1] 1) the sample functions $\xi(x)$ belong to the set $G_y(x_0, x'')$ and have no touchings of level y with probability 1 and 2) the mean[2] number $N(x_0, x'')$ of crossings of level y by the process $\xi(x)$ on the interval (x_0, x'') is finite, then, in view of condition (1), it is not difficult to show that $\mathbf{P}\{Z|L\} = \mathbf{P}\{Z\}$. In other words, one can deal with estimating the unconditional probability $\mathbf{P}\{Z\}$ instead of estimating the conditional one $\mathbf{P}\{Z|L\}$.

For the process $\xi(x)$, we denote by $N^+(x_1, x_2)$ and $N^-(x_1, x_2)$ the mean number of upcrossings and downcrossings of level y, respectively, on an interval (x_1, x_2). Also, we denote by $m_y(x_1, x_2)$ the mean number of local maxima of the process $\xi(x)$ located above level y on an interval (x_1, x_2). The following result was established in [2].

Theorem. *Assume that:*

1) *The sample functions $\xi(x)$ belong to the set $G_y(x_0, x'')$ and have no touchings of level y on the interval (x_0, x'') with probability* 1*, and $N(x_0, x'') < \infty$.*

2) $\mathbf{P}\{\xi(x') = y\} = 0$.

3) *Condition* (1) *holds.*

Let x^* be any point from the interval (x_0, x'') such that $\mathbf{P}\{\xi(x^*) = y\} = 0$, $m_y(x^*, x'') < \infty$. Then

$$N^-(x', x'') - \Big(N^+(x_0, x^*) + m_y(x^*, x'')\Big) \leqslant \mathbf{P}\{Z\} \leqslant N^-(x', x'').$$

If the process $\xi(x)$ is mean-square differentiable, then the numbers $N^-(x_1, x_2)$, $N^+(x_1, x_2)$, and $m_y(x_1, x_2)$ can be calculated by Rice's formulas [6, 8, 10]:

$$N^-(x_1, x_2) = -\int_{x_1}^{x_2} dx \int_{-\infty}^{0} v f_x(y, v) dv, \tag{2}$$

$$N^+(x_1, x_2) = \int_{x_1}^{x_2} dx \int_{0}^{\infty} v f_x(y, v) dv, \tag{3}$$

$$m_y(x_1, x_2) = -\int_{x_1}^{x_2} dx \int_{y}^{\infty} du \int_{-\infty}^{0} z w_x(u, 0, z) dz, \tag{4}$$

[1] Sufficient conditions for assumptions 1) and 2) were formulated in [2].

[2] Here, "mean" refers to the mathematical expectation of a random variable.

where $f_x(u,v)$ is the joint probability density of the random variables $\xi(x)$ and $\xi'(x)$, and $w_x(u,v,z)$ is the joint probability density of the random variables $\xi(x), \xi'(x)$, and $\xi''(x)$. In addition, $\xi'(x)$ and $\xi''(x)$ stand for the first and second mean-square derivatives of the process $\xi(x)$.

2. SPECIFICATION FOR A GAUSSIAN PROCESS

Let the process $\xi(x)$ have the form

$$\xi(x) = a_1 x + a_0 + \eta(x), \quad x \in (-\infty, x''],$$

where $a_1 < 0$ and a_0 are constants, $\eta(x)$ is a stationary centered Gaussian process with continuous realizations and the correlation function

$$r(\tau) \equiv \mathbf{E}\{\eta(x)\eta(x+\tau)\}/\sigma^2,$$

where σ^2 indicates the variance of the processes $\xi(x)$ and $\eta(x)$, and $\mathbf{E}$ is the mathematical expectation operator. If there exists a finite second derivative $r''(0)$, then the process $\xi(x)$ is mean-square differentiable, and Rice's formulas (2) and (3) can be used to calculate the numbers N^- and N^+. If there exists a finite fourth derivative $r^{IV}(0)$, then the process $\xi(x)$ is twice mean-square differentiable, and the number m_y can be calculated by Rice's formula (4).

In the case under consideration, the conditions of the theorem are satisfied, and for any x^* from the interval $(-\infty, x'')$, we have the inequalities

$$N^-(x', x'') - \Big(N^+(-\infty, x^*) + m_y(x^*, x'')\Big) \leqslant \mathbf{P}\{Z\} \leqslant N^-(x', x''), \tag{5}$$

where $N^+(-\infty, x^*) = \lim\limits_{x \to -\infty} N^+(x, x^*)$.

We denote by σ_1^2 and σ_2^2 the variance of the processes $\xi'(x)$ and $\xi''(x)$, respectively, and by ρ the correlation coefficient of the random variables $\xi(x)$ and $\xi''(x)$. Note that

$$\sigma_1^2 = -\sigma^2 r''(0), \quad \sigma_2^2 = \sigma^2 r^{IV}(0), \quad \rho = r''(0)/\sqrt{r^{IV}(0)}.$$

According to these relations, in the nondegenerate case (i.e., $\sigma \neq 0$, $\sigma_1 \neq 0$, and $\sigma_2 \neq 0$), which is studied here, the correlation coefficient ρ takes only negative values.

For the process $\xi(x)$ under consideration, the formulas for $f_x(u,v)$ and $w_x(u,v,z)$ become[3]

$$f_x(u,v) = \frac{1}{2\pi\sigma\sigma_1} \exp\left\{-\frac{(u-a_0-a_1x)^2}{2\sigma^2} - \frac{(v-a_1)^2}{2\sigma_1^2}\right\}, \tag{6}$$

$$\begin{aligned} w_x(u,v,z) &= \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{(v-a_1)^2}{2\sigma_1^2}\right\} \frac{1}{2\pi\sigma\sigma_2\sqrt{1-\rho^2}} \\ &\times \exp\left\{-\frac{1}{2(1-\rho^2)}\left[\frac{(u-a_0-a_1x)^2}{\sigma^2} - 2\rho\frac{(u-a_0-a_1x)z}{\sigma\sigma_2} + \frac{z^2}{\sigma_2^2}\right]\right\}. \end{aligned} \tag{7}$$

[3] Since the process $\xi(x)$ is Gaussian, the probability densities $f_x(u,v)$ and $w_x(u,v,z)$ will be Gaussian (e.g., see [10, 15]); in other words, $f_x(u,v)$ and $w_x(u,v,z)$ are two- and three-dimensional Gaussian probability densities, respectively. In addition, for any x, the correlation coefficient of the random variables $\xi(x)$ and $\xi'(x)$ and that of the random variables $\xi'(x)$ and $\xi''(x)$ are equal to zero due to the stationarity of the process $\eta(x)$ (e.g., see [14, 15]). These circumstances finally bring to formulas (6) and (7).

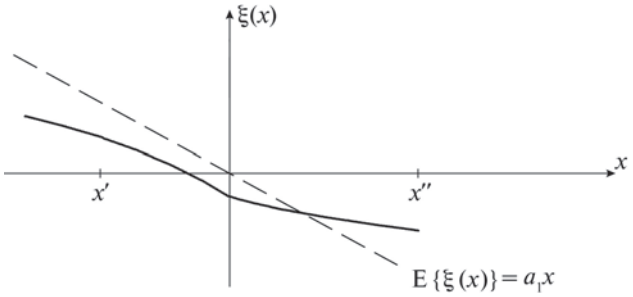

Fig. 2. Crossing of the zero level $y = 0$ by some realization of the process $\xi(x)$ and by its mean $\mathbf{E}\{\xi(x)\}$ for $a_0 = 0$.

For the sake of definiteness, assume that $y = 0$, $a_0 = 0$, and $x'' > 0$ (Fig. 2). Then event Z means that the zero level will be first reached by the process $\xi(x) = a_1 x + \eta(x)$ at some point from the interval (x', x'').

We introduce the parameters[4]

$$\alpha = -a_1/\sigma_1 \quad \text{and} \quad \beta = -a_1 x''/\sigma.$$

Then, see the Appendix, formulas (2)–(4) yield

$$N^-(x', x'') = \left[\frac{1}{\sqrt{2\pi}\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\} + \Phi(\alpha)\right]\left[\Phi(\beta) - \Phi\left(\beta\frac{x'}{x''}\right)\right], \tag{2′}$$

$$N^+(-\infty, x^*) = \left[\frac{1}{\sqrt{2\pi}\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\} - \Phi(-\alpha)\right]\Phi\left(\beta\frac{x^*}{x''}\right), \tag{3′}$$

$$\begin{aligned} m_y(x^*, x'') &= -\frac{\beta}{2\pi\rho\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\} \\ &\times \int\limits_{x^*/x''}^{1}\left[\Phi\left(-\frac{\beta}{\sqrt{1-\rho^2}}v\right) - \rho\Phi\left(-\frac{\rho\beta v}{\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{\beta^2 v^2}{2}\right\}\right]dv, \end{aligned} \tag{4′}$$

where

$$\Phi(z) = \frac{1}{\sqrt{2\pi}}\int\limits_{-\infty}^{z}\exp\{-x^2/2\}dx. \tag{8}$$

3. PROBABILITY ESTIMATES: ACCURACY ANALYSIS

Inequalities (5) can be written as

$$N^-(x', x'') - f(x^*) \leqslant \mathbf{P}\{Z\} \leqslant N^-(x', x''),$$

where

$$f(x^*) = N^+(-\infty, x^*) + m_y(x^*, x''), \quad x^* \in (-\infty, x''].$$

[4] In a physical interpretation, the value of the parameter α shows how strongly, on average, the slope of the realizations of the process $\xi(x)$ differs from that of its mean $\mathbf{E}\{\xi(x)\}$: for small α, this difference is large, and vice versa.

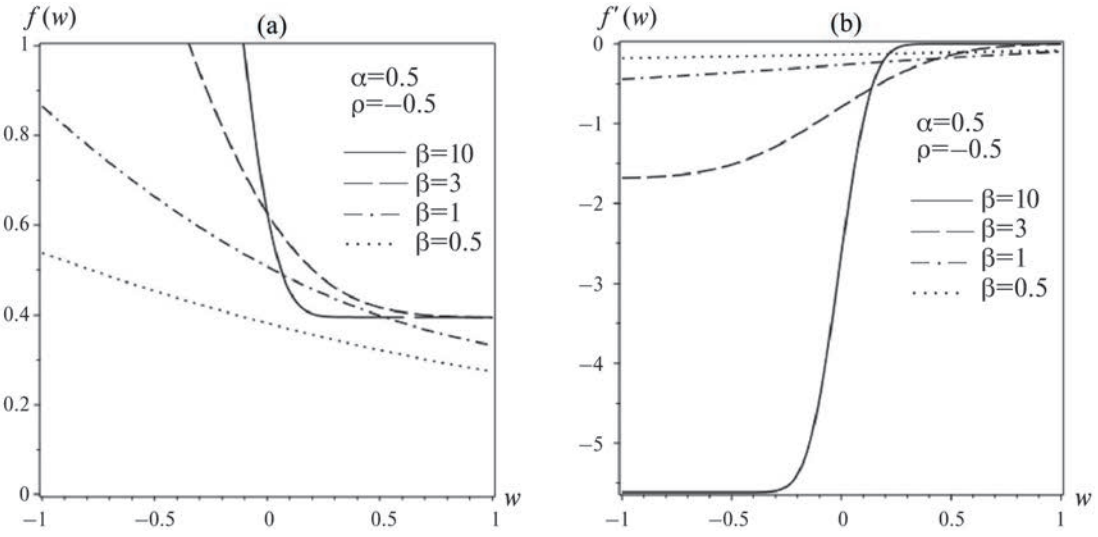

Fig. 3. (a) Curves $f(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.5$ and (b) curves $f'(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.5$.

With the dimensionless variable $w = x^*/x''$, the estimates (5) take the form

$$N^-(x', x'') - f(w) \leqslant \mathbf{P}\{Z\} \leqslant N^-(x', x''), \tag{5'}$$

where $w \in (-\infty, 1]$ and

$$f(w) = N^+(-\infty, w) + m_y(w, 1),$$

$$N^+(-\infty, w) = \left[\frac{1}{\sqrt{2\pi}\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\} - \Phi(-\alpha)\right]\Phi(\beta w),$$

$$m_y(w,1) = -\frac{\beta}{2\pi\rho\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\}\int\limits_w^1\left[\Phi\left(-\frac{\beta}{\sqrt{1-\rho^2}}v\right) - \rho\Phi\left(-\frac{\rho\beta v}{\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{\beta^2 v^2}{2}\right\}\right]dv.$$

Let us investigate the behavior of the function $f(w)$ on the interval $(-\infty, 1]$. The smaller the values of $f(w)$ are, the higher accuracy the estimates (5′) of the probability $\mathbf{P}\{Z\}$ will have.

Consider the behavior of the function $f(w)$ depending on three parameters: α, β, and ρ. The region of interest is $w \in [-1, 1]$ since, by the physical meaning of the numbers $N^+(-\infty, w)$ and $m_y(w, 1)$, the function $f(w), w < -1$, is monotonically decreasing, so the minimum values of $f(w)$ will be achieved for $w > -1$.

Figure 3a shows the behavior of the function $f(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.5$. As it turned out, for any value combination of the parameters α, β, and ρ, the function $f(w)$ is monotonically decreasing on the interval $(-\infty, 1]$. This conclusion follows from the analysis of the derivative $f'(w)$. Based on the well-known formula for the derivative of a definite integral with variable limits,

$$\frac{d}{dt}\int\limits_{\phi_1(t)}^{\phi_2(t)} h(v)dv = -h(\phi_1(t))\frac{d\phi_1(t)}{dt} + h(\phi_2(t))\frac{d\phi_2(t)}{dt},$$

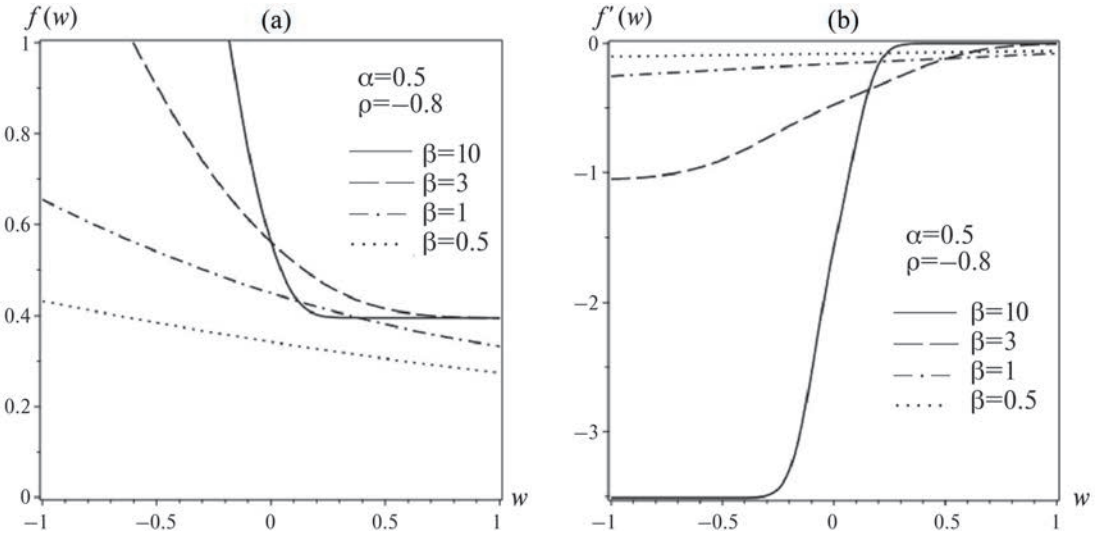

Fig. 4. (a) Curves $f(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.8$ and (b) curves $f'(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.8$.

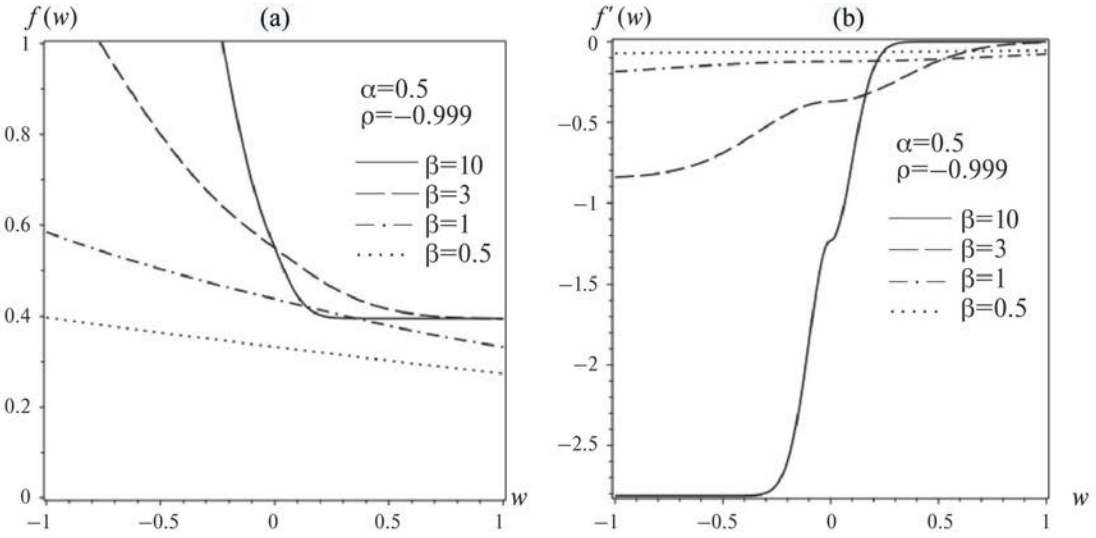

Fig. 5. (a) Curves $f(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.999$ and (b) curves $f'(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.999$.

we obtain the following expression for the derivative of the function $f(w)$:

$$\frac{df(w)}{dw} = \left[\frac{1}{\sqrt{2\pi}\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\} - \Phi(-\alpha)\right]\frac{1}{\sqrt{2\pi}}\exp\left\{-\frac{\beta^2 w^2}{2}\right\}\beta$$

$$+\frac{\beta}{2\pi\rho\alpha}\exp\left\{-\frac{\alpha^2}{2}\right\}\left[\Phi\left(-\frac{\beta w}{\sqrt{1-\rho^2}}\right) - \rho\Phi\left(-\frac{\rho\beta w}{\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{\beta^2 w^2}{2}\right\}\right].$$

Figure 3b shows the behavior of the derivative $f'(w)$ for different β, $\alpha = 0.5$, and $\rho = -0.5$. According to the numerical calculations, $f'(w) < 0$ for all $w \in (-\infty, 1]$. So the minimum values of the function $f(w)$ are achieved at $w = 1$. However, the nature of change of the derivative $f'(w)$ strongly depends on the parameters α, β, and ρ. For example, for $\beta = 10$, the values of $f'(w)$ are negligibly small on the interval $(0.5, 1)$; therefore, $f(w)$ is almost independent of w on this interval.

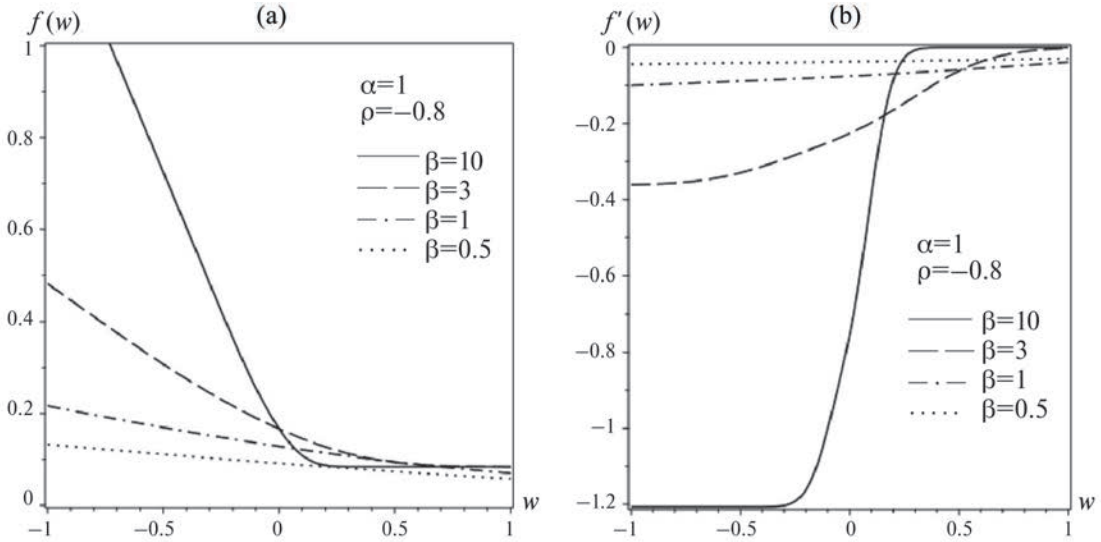

Fig. 6. (a) Curves $f(w)$ for different β, $\alpha = 1$, and $\rho = -0.8$ and (b) curves $f'(w)$ for different β, $\alpha = 1$, and $\rho = -0.8$.

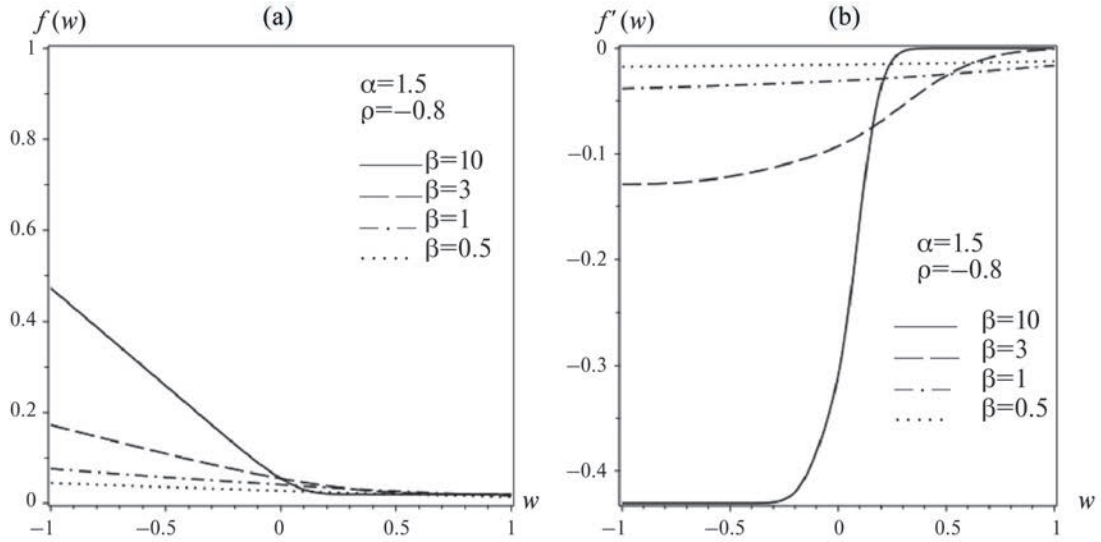

Fig. 7. (a) Curves $f(w)$ for different β, $\alpha = 1.5$, and $\rho = -0.8$ and (b) curves $f'(w)$ for different β, $\alpha = 1.5$, and $\rho = -0.8$.

Direct comparison of Figs. 3a–5a demonstrates changes in the behavior of the function $f(w)$ depending on the correlation coefficient ρ : $\rho = -0.5$ for Fig. 3a, $\rho = -0.8$ for Fig. 4a, and $\rho = -0.999$ for Fig. 5a. The corresponding changes in the behavior of the derivatives $f'(w)$ are shown in Figs. 3b–5b. If w is close to 1, then the values of $f(w)$ change very weakly. If w is below 1, then the values of $f(w)$ decrease under the transition $\{\rho = -0.5\} \to \{\rho = -0.8\} \to \{\rho = -0.999\}$; the smaller w is, the greater this decrease will be. Moreover, this conclusion holds for all β.

Direct comparison of Figs. 4a and 6a–9a demonstrates changes in the behavior of the function $f(w)$ depending on the parameter α : $\alpha = 0.5$ for Fig. 4a, $\alpha = 1$ for Fig. 6a, $\alpha = 1.5$ for Fig. 7a, $\alpha = 2$ for Fig. 8a, and $\alpha = 3$ for Fig. 9a. The corresponding changes in the behavior of the derivatives $f'(w)$ are shown in Figs. 4b and 6b–9b. Clearly, for any fixed β, increasing α leads to a decrease in $f(w)$. If $\alpha = 3$ (Fig. 9a), then the values of $f(w)$ are less than 0.008 for $\beta = 10$ and

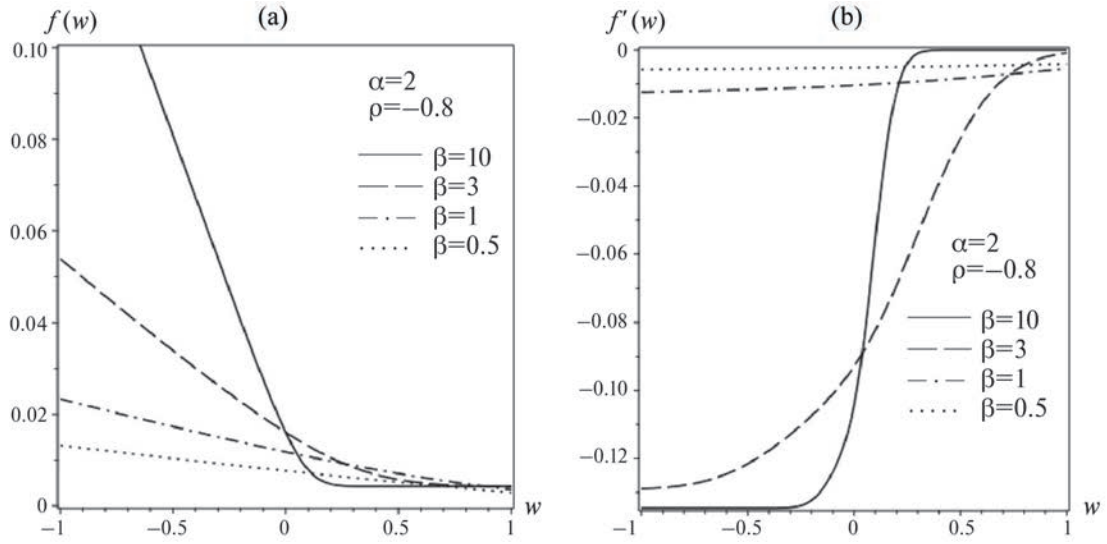

Fig. 8. (a) Curves $f(w)$ for different β, $\alpha = 2$, and $\rho = -0.8$ and (b) curves $f'(w)$ for different β, $\alpha = 2$, and $\rho = -0.8$.

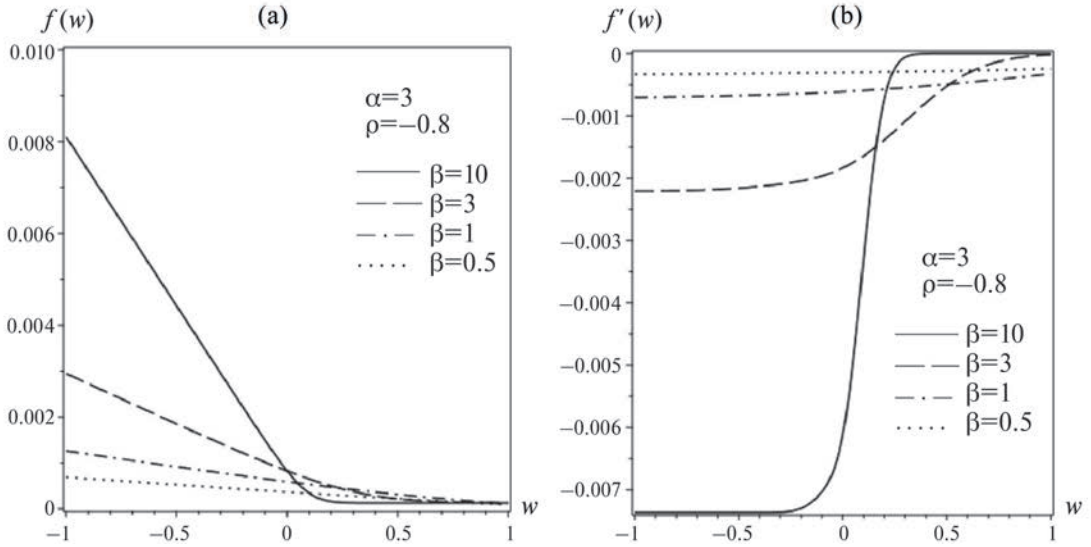

Fig. 9. (a) Curves $f(w)$ for different β, $\alpha = 3$, and $\rho = -0.8$ and (b) curves $f'(w)$ for different β, $\alpha = 3$, and $\rho = -0.8$.

$\beta = 3$ and less than 0.001 for $\beta = 1$ and $\beta = 0.5$. For all value combinations of the parameters α, β, and ρ under consideration, the minimum value of the function $f(w)$ is achieved at $w = 1$. Also note that for $\alpha > 3$, the values of $f(w)$ are negligibly small. In this case, inequalities (5′) allow determining the probability $\mathbf{P}\{Z\}$ almost exactly.

The Maple package was used for the numerical calculations.

4. CONCLUSIONS

A special stationary Gaussian process $\xi(x)$ with drift has been considered, and the estimates (5) for the probability $\mathbf{P}\{Z\}$ [2] of the process have been analyzed numerically. That is, the accuracy of these estimates has been studied depending on the choice of the point x^* and the parameters

α, β, and ρ of the process $\xi(x)$. The accuracy is given by $f(w)$, $w \in (-\infty, 1]$, (see (5′)), where the value w is uniquely determined by the choice of the point x^*. The smaller value $f(w)$ takes, the more accurate the resulting probability $\mathbf{P}\{Z\}$ will be. As it turned out, for *Gaussian* processes, the minimum value of $f(w)$ is obtained at $w = 1$ for any value combination of the parameters α, β, and ρ. The values of $f(w)$ strongly depend on the parameter α : the larger α is, the smaller $f(w)$ will be. If $\alpha > 3$, then the values of $f(w)$ are negligibly small, and the probability $\mathbf{P}\{Z\}$ is determined almost exactly.

APPENDIX

We find the number $N^+(x_1, x_2)$ by formula (3), substituting the expression (6) for the density $f_x(u, v)$ with $u = y$. As a result,

$$N^+(x_1, x_2) = \frac{1}{2\pi\sigma\sigma_1} \int\limits_{x_1}^{x_2} \exp\left\{-\frac{(y - a_0 - a_1 x)^2}{2\sigma^2}\right\} dx \int\limits_0^\infty v \exp\left\{-\frac{(v - a_1)^2}{2\sigma_1^2}\right\} dv.$$

Both integrals here are easily calculated:

$$\int\limits_{x_1}^{x_2} \exp\left\{-\frac{(y - a_0 - a_1 x)^2}{2\sigma^2}\right\} dx = -\frac{\sigma}{a_1}\sqrt{2\pi}\left[\Phi\left(\frac{y - a_0 - a_1 x_2}{\sigma}\right) - \Phi\left(\frac{y - a_0 - a_1 x_1}{\sigma}\right)\right],$$

$$\int\limits_0^\infty v \exp\left\{-\frac{(v - a_1)^2}{2\sigma_1^2}\right\} dv = \sigma_1^2 \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} + \sqrt{2\pi} a_1 \sigma_1 \Phi\left(\frac{a_1}{\sigma_1}\right),$$

where the function $\Phi(\cdot)$ is given by (8). Consequently,

$$N^+(x_1, x_2) = \left[\frac{\sigma_1}{\sqrt{2\pi} a_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} + \Phi\left(\frac{a_1}{\sigma_1}\right)\right]\left[\Phi\left(\frac{y - a_0 - a_1 x_1}{\sigma}\right) - \Phi\left(\frac{y - a_0 - a_1 x_2}{\sigma}\right)\right].$$

In this relation, letting $y = 0$, $a_0 = 0$, and $x_2 = x^*$ and denoting $\alpha = -a_1/\sigma_1$ and $\beta = -a_1 x''/\sigma$ (by analogy with the main text of the paper), one finally arrives at formula (3′) as $x_1 \to -\infty$ since $a_1 < 0$.

Similarly, substituting the expression (6) for the density $f_x(u, v)$ with $u = y$, by formula (2) we obtain the number

$$N^-(x_1, x_2) = \left[\frac{\sigma_1}{\sqrt{2\pi} a_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} - \Phi\left(-\frac{a_1}{\sigma_1}\right)\right]\left[\Phi\left(\frac{y - a_0 - a_1 x_1}{\sigma}\right) - \Phi\left(\frac{y - a_0 - a_1 x_2}{\sigma}\right)\right].$$

In this relation, letting $y = 0$, $a_0 = 0$, $x_1 = x'$, and $x_2 = x''$ and denoting $\alpha = -a_1/\sigma_1$ and $\beta = -a_1 x''/\sigma$ (by analogy with the main text of the paper), one finally arrives at formula (2′).

Now, using formula (4), let us determine the number $m_y(x_1, x_2)$ if the density $w_x(u, 0, z)$ is calculated by (7) with $v = 0$. With the new variables $\tilde{u} = u - a_0 - a_1 x$, $\tilde{z} = z$, and $\tilde{x} = x$, we obtain

$$m_y(x_1, x_2) = -\int\limits_{x_1}^{x_2} d\tilde{x} \int\limits_{y - a_0 - a_1\tilde{x}}^{\infty} d\tilde{u} \int\limits_{-\infty}^{0} \frac{\tilde{z}}{\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}$$

$$\times \frac{1}{2\pi\sigma\sigma_2\sqrt{1 - \rho^2}} \exp\left\{-\frac{1}{2(1 - \rho^2)}\left[\frac{\tilde{u}^2}{\sigma^2} - 2\rho\frac{\tilde{u}\tilde{z}}{\sigma\sigma_2} + \frac{\tilde{z}^2}{\sigma_2^2}\right]\right\} d\tilde{z}.$$

After additional changes of the variables, $\bar{u} = \tilde{u}/\sigma$, $\bar{z} = \tilde{z}/\sigma_2$, and $\bar{x} = \tilde{x}$, it follows that

$$m_y(x_1, x_2) = -\int_{x_1}^{x_2} d\bar{x} \int_{\frac{y-a_0-a_1\bar{x}}{\sigma}}^{\infty} \sigma d\bar{u} \int_{-\infty}^{0} \frac{\sigma_2 \bar{z}}{\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}$$

$$\times \frac{1}{2\pi\sigma\sigma_2\sqrt{1-\rho^2}} \exp\left\{-\frac{\bar{u}^2 - 2\rho\bar{u}\bar{z} + \bar{z}^2}{2(1-\rho^2)}\right\} \sigma_2 d\bar{z}$$

$$= -\frac{\sigma_2(1-\rho^2)^{-\frac{1}{2}}}{2\pi\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} \int_{x_1}^{x_2} d\bar{x} \int_{\frac{y-a_0-a_1\bar{x}}{\sigma}}^{\infty} \exp\left\{-\frac{\bar{u}^2}{2(1-\rho^2)}\right\} d\bar{u} \int_{-\infty}^{0} \bar{z} \exp\left\{-\frac{\bar{z}^2 - 2\rho\bar{u}\bar{z}}{2(1-\rho^2)}\right\} d\bar{z}.$$

The exponent index in the last integral can be transformed as

$$-\frac{\bar{z}^2 - 2\rho\bar{u}\bar{z}}{2(1-\rho^2)} = -\frac{(\bar{z} - \rho\bar{u})^2 - \rho^2\bar{u}^2}{2(1-\rho^2)}.$$

Then we make the change of variables

$$\lambda = \bar{u}, \quad \mu = \bar{z} - \rho\bar{u}, \quad \tau = \bar{x}$$

to get

$$m_y(x_1, x_2) = -\frac{\sigma_2(1-\rho^2)^{-\frac{1}{2}}}{2\pi\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}$$

$$\times \int_{x_1}^{x_2} d\tau \int_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty} \exp\left\{-\frac{\lambda^2}{2(1-\rho^2)}\right\} d\lambda \int_{-\infty}^{-\rho\lambda} (\mu + \rho\lambda) \exp\left\{-\frac{\mu^2 - \rho^2\lambda^2}{2(1-\rho^2)}\right\} d\mu$$

$$= -\frac{\sigma_2(1-\rho^2)^{-\frac{1}{2}}}{2\pi\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} \int_{x_1}^{x_2} d\tau \int_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty} \exp\left\{-\frac{\lambda^2}{2}\right\} d\lambda \int_{-\infty}^{-\rho\lambda} (\mu + \rho\lambda) \exp\left\{-\frac{\mu^2}{2(1-\rho^2)}\right\} d\mu.$$

Let $g(\lambda)$ denote the inner integral with respect to μ. Direct calculations yield

$$g(\lambda) \equiv \int_{-\infty}^{-\rho\lambda} (\mu + \rho\lambda) \exp\left\{-\frac{\mu^2}{2(1-\rho^2)}\right\} d\mu$$

$$= -(1-\rho^2) \exp\left\{-\frac{\rho\lambda^2}{2(1-\rho^2)}\right\} + \rho\lambda \int_{-\infty}^{-\rho\lambda} \exp\left\{-\frac{\mu^2}{2(1-\rho^2)}\right\} d\mu,$$

or, using the function $\Phi(\cdot)$ introduced above,

$$g(\lambda) = -(1-\rho^2) \exp\left\{-\frac{\rho\lambda^2}{2(1-\rho^2)}\right\} + \sqrt{2\pi(1-\rho^2)}\rho\lambda\Phi\left(-\frac{\rho\lambda}{\sqrt{1-\rho^2}}\right).$$

Thus,

$$m_y(x_1, x_2) = -\frac{\sigma_2(1-\rho^2)^{-\frac{1}{2}}}{2\pi\sqrt{2\pi}\sigma_1} \exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\} \int_{x_1}^{x_2} I(\tau) d\tau, \tag{A.1}$$

where

$$I(\tau) = I_1(\tau) + I_2(\tau),$$

$$I_1(\tau) = \int\limits_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty} -(1-\rho^2)\exp\left\{-\frac{\lambda^2}{2(1-\rho^2)}\right\}d\lambda,$$

$$I_2(\tau) = \int\limits_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty} \sqrt{2\pi(1-\rho^2)}\rho\lambda\exp\left\{-\frac{\lambda^2}{2}\right\}\Phi\left(-\frac{\rho\lambda}{\sqrt{1-\rho^2}}\right)d\lambda.$$

Based on the integration by parts, we bring $I_2(\tau)$ to the form

$$I_2(\tau) = \sqrt{2\pi(1-\rho^2)}\rho\Phi\left(-\frac{\rho(y-a_0-a_1\tau)}{\sigma\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{(y-a_0-a_1\tau)^2}{2\sigma^2}\right\}$$

$$-\rho^2\int\limits_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty}\exp\left\{-\frac{\lambda^2}{2(1-\rho^2)}\right\}d\lambda,$$

which allows writing $I(\tau)$ as

$$I(\tau) = -\int\limits_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty}\exp\left\{-\frac{\lambda^2}{2(1-\rho^2)}\right\}d\lambda$$

$$+\sqrt{2\pi(1-\rho^2)}\rho\Phi\left(-\frac{\rho(y-a_0-a_1\tau)}{\sigma\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{(y-a_0-a_1\tau)^2}{2\sigma^2}\right\}.$$

Expressing the first term here through the above function $\Phi(\cdot)$, we find

$$-\int\limits_{\frac{y-a_0-a_1\tau}{\sigma}}^{\infty}\exp\left\{-\frac{\lambda^2}{2(1-\rho^2)}\right\}d\lambda = -\sqrt{2\pi(1-\rho^2)}\left[1-\Phi\left(\frac{y-a_0-a_1\tau}{\sigma\sqrt{1-\rho^2}}\right)\right].$$

Hence, due to (A.1),

$$m_y(x_1,x_2) = -\frac{\sigma_2}{2\pi\sigma_1}\exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}$$

$$\times\int\limits_{x_1}^{x_2}\left[-1+\Phi\left(\frac{y-a_0-a_1\tau}{\sigma\sqrt{1-\rho^2}}\right)+\rho\Phi\left(-\frac{\rho(y-a_0-a_1\tau)}{\sigma\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{(y-a_0-a_1\tau)^2}{2\sigma^2}\right\}\right]d\tau.$$

In this relation, letting $y=0$, $a_0=0$, $x_1=x^*$, and $x_2=x''$ and utilizing the identity $1-\Phi(z)=\Phi(-z)$, we obtain

$$m_y(x^*,x'') = \frac{\sigma_2}{2\pi\sigma_1}\exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}\int\limits_{x^*}^{x''}\left[\Phi\left(\frac{a_1\tau}{\sigma\sqrt{1-\rho^2}}\right)-\rho\Phi\left(\frac{\rho a_1\tau}{\sigma\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{(a_1\tau)^2}{2\sigma^2}\right\}\right]d\tau.$$

With the new integration variable $\nu=\tau/x''$, it follows that

$$m_y(x^*,x'') = \frac{\sigma_2 x''}{2\pi\sigma_1}\exp\left\{-\frac{a_1^2}{2\sigma_1^2}\right\}\int\limits_{x^*/x''}^{1}\left[\Phi\left(\frac{a_1x''\nu}{\sigma\sqrt{1-\rho^2}}\right)-\rho\Phi\left(\frac{\rho a_1x''\nu}{\sigma\sqrt{1-\rho^2}}\right)\exp\left\{-\frac{(a_1\nu x'')^2}{2\sigma^2}\right\}\right]d\nu.$$

Finally, letting $\alpha=-a_1/\sigma_1$ and $\beta=-a_1x''/\sigma$ (by analogy with the main text of the paper), one finally arrives at formula (4′).

REFERENCES

1. Pontryagin, L.S., Andronov, A.A., and Vitt, A.A., On the Statistical Consideration of Dynamic Systems, *Zhurnal Eksperimental'noi i Teoreticheskoi Fiziki*, 1933, vol. 3, no. 3, pp. 165–180.
2. Semakov, S.L. and Semakov, I.S., Estimating the Probability That a Random Process First Reaches the Boundary of a Region on a Given Time Interval, *Proc. 2018 IEEE 57th Conference on Decision and Control (CDC-2018)*, Miami Beach, USA, 2018, pp. 256–261.
3. Semakov, S.L., The Probability of the First Attainment of a Level by a Component of a Multidimensional Process on a Prescribed Interval under Restrictions on Its Other Components, *Theory of Probability and Its Applications*, 1990, vol. 34, no. 2, pp. 357–361.
4. Semakov, S.L. and Semakov, I.S., The Probability of First Reaching a Desired Level by a Random Process on a Given Interval, *Automation and Remote Control*, 2019, vol. 80, no. 3, pp. 459–473.
5. Semakov, S.L., The First Achievement of a Given Level by a Random Process, *IEEE Transactions on Information Theory*, 2024, vol. 70, no. 10, pp. 7162–7178.
6. Rice, S.O., Mathematical Analysis of Random Noise, *The Bell System Technical Journal*, 1944, vol. 23, no. 3, pp. 282–332; 1945, vol. 24, no. 1, pp. 46–156.
7. Itô, K., The Expected Number of Zeros of Continuous Stationary Gaussian Processes, *Journal of Mathematics of Kyoto University*, 1964, vol. 3, no. 2, pp. 207–221.
8. Leadbetter, M.R., On Crossings of Levels and Curves by a Wide Class of Stochastic Processes, *Annals of Mathematical Statistics*, 1966, vol. 37, no. 1, pp. 260–267.
9. Belyaev, Yu.K., On the Number of Level Crossings by a Gaussian Stochastic Process I, II, *Theory of Probability and Its Applications*, 1966, vol. 11, no. 1, pp. 106–113; 1967, vol. 12, no. 3, pp. 392–404.
10. Cramér, H. and Leadbetter, M.R., *Stationary and Related Stochastic Processes*, New York: Wiley, 1967.
11. Pickands, J., Upcrossing Probabilities for Stationary Gaussian Processes, *Transactions of the American Mathematical Society*, 1969, vol. 145, pp. 51–73.
12. Blake, I. and Lindsey, W., Level-Crossings Problems for Random Processes, *IEEE Transactions on Information Theory*, 1973, vol. 19, no. 3, pp. 295–315.
13. Hüsler, J., Extreme Values and High Boundary Crossings of Locally Stationary Gaussian Processes, *Annals of Probability*, 1990, vol. 18, no. 3, pp. 1141–1158.
14. Tikhonov, V.I., *Vybrosy sluchaynykh protsessov* (Outliers of Random Processes), Moscow: Nauka, 1970.
15. Semakov, S.L., *Vybrosy sluchainykh protsessov: prilozheniya v aviatsii* (Outliers of Random Processes: Applications in Aviation), Moscow: Nauka, 2005.
16. Semakov, S.L. and Semakov, I.S., Estimating the Probability of Safe Landing for Aircrafts, *Proc. 2019 IEEE 58th Conference on Decision and Control (CDC-2019)*, Nice, France, 2019, pp. 2568–2573.
17. Semakov, S.L. and Semakov, I.S., Method of Calculating the Probability of a Safe Landing for Ship-Based Aircraft, *IEEE Transactions on Aerospace and Electronic Systems*, 2022, vol. 58, no. 6, pp. 5425–5442.
18. Semakov, S.L. and Semakova, M.V., Thrust Control for Aircraft Landing on a Carrier, *Automation and Remote Control*, 2023, vol. 84, no. 3, pp. 331–340.
19. Semakov, S.L., *Crossings Problems in Random Processes Theory and Their Applications in Aviation*, 2nd ed., Newcastle, UK: Cambridge Scholars Publishing, 2024.

This paper was recommended for publication by A.V. Nazin, a member of the Editorial Board