Volume 86, Number 2,
February 2025

ISSN 0005-1179
CODEN: AURCAT

A JOURNAL OF RUSSIAN ACADEMY OF SCIENCES

AUTOMATION AND REMOTE CONTROL

Automation and Remote Control

Vol. 86, No. 2, February 2025

Editor-in-Chief
Andrey A. Galyaev

http://ait.mtas.ru

Automation and Remote Control

ISSN 0005-1179

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

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. 2, 2025

Nonlinear Systems

Stochastic Systems

Control in Technical Systems

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 2, pp. 99–112.

NONLINEAR SYSTEMS

Optimal Absolute Stabilization of Unknown Lurie Systems Based on Experimental Data and A Priori Information

M. M. Kogan[*,a] **and A. V. Stepanov**[**,b]

[*] *Sirius University of Science and Technology, Sochi, Russia*
[**] *Nizhny Novgorod State University of Architecture and Civil Engineering, Nizhny Novgorod, Russia*
e-mail: [a] *mkogan@nngasu.ru,* [b] *andrey8st@yahoo.com*

Received January 18, 2024
Revised August 20, 2024
Accepted November 26, 2024

Abstract—This paper considers Lurie systems composed of an unknown linear subsystem and unknown nonlinear functions belonging to given sectors. For such systems, a method is developed to design an optimal absolutely stabilizing control law based on experimental data and a priori information. The method involves the minimax approach in which an integral quadratic performance index is maximized at the intersection of two matrix ellipsoidal sets selected from experimental data and a priori information. The simulation results of a nonlinear oscillator show the advantage of the obtained control law over the classical robust controller designed based on a priori information.

Keywords: Lurie systems, uncertainty, robust control, experimental data, linear matrix inequalities (LMIs)

DOI: 10.31857/S0005117925020017

1. INTRODUCTION

Nonlinear Lurie systems consist of unknown linear subsystems and unknown nonlinear functions belonging to given sectors [1]. In this paper, we construct control laws for this class of systems using the control design methods for unknown linear dynamic systems developed in [2–4]. The absence of any mathematical model of the controlled system is compensated here by measurement data of the system trajectory on a finite horizon and some a priori information. In this case, the identification problem is not solved for the system, and the control objective is achieved even under non-identifiability conditions. These methods involve the minimax robust control design approach in which the guaranteed value of the performance index is found for any system from some uncertainty set. In contrast to classical robust controllers (e.g., see the review [5]), where this set is chosen based on only a priori information, here the uncertainty set is separated by using both experimental data and a priori information. As a result, one obtains a control law with a much smaller guaranteed value of the performance index; see below.

In recent time, the use of experimental information for the direct design of control laws has received much attention; for example, see [6–8] and the bibliography provided in [2–4]. These research works mainly addressed control laws for linear systems. To the authors' best knowledge, the paper [9] is currently the only one devoted to experimental data-based control design for systems with a nonlinear vector function satisfying a quadratic inequality. According to the assumptions made in [9], there are no disturbances during the experiment and the control law ensures the so-called persistent excitation condition, necessary for the identifiability of the unknown parameters. Furthermore, the dimension of the variables in the linear matrix inequalities (LMIs) for computing the controller parameters grows with the number of measurements, which complicates control law implementation. In contrast, the approach discussed below covers systems with several nonlinear sector functions, considers disturbances during the experiment, and requires no system identifiability; in addition, the dimensions of the variables are determined only by those of the state and control vectors and do not depend on the number of measurements.

2. PROBLEM STATEMENT

Consider a controlled uncertain nonlinear Lurie system consisting of a linear subsystem of the form

$$\begin{aligned} \partial x(t) &= Ax(t) + Bu(t) + Fv(t), \\ z(t) &= Cx(t) + Du(t) \end{aligned} \tag{2.1}$$

closed by a nonlinear continuous vector function of the form

$$v(t) = \varphi(y(t), t), \quad y(t) = L^{\mathrm{T}} x(t), \tag{2.2}$$

with the following notations: ∂ stands for the differentiation operator in the continuous-time case or the shift operator in the discrete-time case; $x(t) \in \mathrm{R}^{n_x}$ is the state vector; $y(t) \in \mathrm{R}^{n_y}$ is the output; $u(t) \in \mathrm{R}^{n_u}$ is the control vector (input); $z(t) \in \mathrm{R}^{n_z}$ is the performance output; finally, $\varphi(y,t) \in \mathrm{R}^{n_y}$ is an unknown vector function such that $\varphi(0,t) \equiv 0$. For all $t \geqslant 0$, each component $\varphi_i(y_i,t)$ of the function (2.2) is located in a corresponding finite sector $[\alpha_i, \beta_i]$, i.e.,

$$\alpha_i \leqslant \frac{\varphi_i(y_i,t)}{y_i} \leqslant \beta_i, \quad i = 1, \ldots, n_y. \tag{2.3}$$

By assumption, the system matrices A, B, and F are unknown and the initial state $x(0) = x_0$ is uncertain. The problem statement will be further refined; generally speaking, it is required to design linear state-feedback controllers $u(t) = \Theta x(t)$ based on a priori information and experimental data under which the closed loop system will be absolutely stable, i.e., the trivial equilibrium $x = 0$ of system (2.1), (2.2) will be asymptotically stable for all functions $\varphi(y,t)$ from the specified class, and the transient will satisfy the following upper bound under arbitrary initial conditions:

$$\sup_{x_0 \neq 0} \frac{\|z\|^2}{x_0^{\mathrm{T}} R^{-1} x_0} < \gamma^2, \tag{2.4}$$

where $R = R^{\mathrm{T}} > 0$ is a weight matrix and $\|\xi\|^2 = \sum_{t=0}^{\infty} |\xi(t)|^2$ in the continuous-time case or $\|\xi\|^2 = \int_{t=0}^{\infty} |\xi(t)|^2 \, dt$ in the discrete-time case.

3. EXPERIMENTAL DATA AND A PRIORI INFORMATION

Information about the unknown parameters of system (2.1) is extracted from a finite set of measurements of its trajectory. Let a disturbance $w(t)$ affect the system during an experiment so that the system equations take the form

$$\begin{aligned} \partial x(t) &= Ax(t) + Bu(t) + Fv(t) + B_w w(t), \\ z(t) &= Cx(t) + Du(t). \end{aligned} \tag{3.1}$$

Suppose that in the experiment preceding the control design procedure, it is possible to measure the values of the system's nonlinear function belonging to given sectors. In the discrete-time case, there are available measurements of the state and nonlinear function, $x_0, x_1, \ldots, x_N$ and

$\varphi(y_0, 0), \ldots, \varphi(y_{N-1}, N-1)$, respectively, under chosen controls $u_0, \ldots, u_{N-1}$ and some unknown disturbances $w_0, \ldots, w_{N-1}$. We compile the matrices

$$X = (x_0 \cdots x_{N-1}), \quad X_+ = (x_1 \cdots x_N), \quad U = (u_0 \cdots u_{N-1}),$$
$$\Phi = (\varphi(y_0, 0) \cdots \varphi(y_{N-1}, N-1), \quad W = (w_0 \cdots w_{N-1}).$$

In the continuous-time case, there are available measurements of the state, its derivative, and nonlinear function, $x(t_0), \ldots, x(t_{N-1})$, $\dot{x}(t_0), \ldots, \dot{x}(t_{N-1})$, and $\varphi(y(t_0), t_0), \ldots, \varphi(y(t_{N-1}), t_{N-1})$, respectively, at time instants $t_0, \ldots, t_{N-1}$ under chosen controls $u(t_0), \ldots, u(t_{N-1})$ and some unknown disturbances $w(t_0), \ldots, w(t_{N-1})$. We compile the matrices

$$X = (x(t_0) \cdots x(t_{N-1})), \quad X_+ = (\dot{x}(t_0) \cdots \dot{x}(t_{N-1})), \quad U = (u(t_0) \cdots u(t_{N-1})),$$
$$\Phi = (\varphi(y(t_0), t_0) \cdots \varphi(y(t_{N-1}), t_{N-1})), \quad W = (w(t_0) \cdots w(t_{N-1})).$$

In both cases, the experimental data matrices satisfy the relations

$$X_+ = A_{real} X + B_{real} U + F_{real} \Phi + B_w W, \tag{3.2}$$

where A_{real}, B_{real}, and F_{real} are the real unknown system matrices. With the notations

$$\Delta_{real} = (A_{real}\, B_{real}\, F_{real}), \quad \widehat{X} = \operatorname{col}(X, U, \Phi),$$

equations (3.2) can be written as the linear matrix regression

$$X_+ = \Delta_{real} \widehat{X} + \widehat{W}, \quad \widehat{W} = B_w W. \tag{3.3}$$

Assume that the disturbances, which also include the approximate calculation errors of the derivatives, satisfy the condition

$$\widehat{W}\widehat{W}^{\mathrm{T}} \leqslant \Omega. \tag{3.4}$$

In particular, if $\|w(t)\|_\infty \leqslant d_w$ for all t and a given value d_w (the error level), then $\Omega = d_w^2 n_w N B_w B_w^{\mathrm{T}}$. In the case $\sum_{i=0}^{N-1} |w(t_i)|^2 \leqslant \nu^2$ (the total "energy" of the disturbances is bounded above during the experiment), we have $\Omega = \nu^2 B_w B_w^{\mathrm{T}}$.

Let us define the set $\mathbf{\Delta_p}$ of all matrices Δ of dimensions $n_x \times (n_x + n_u + n_y)$ that could generate the experimental matrices Φ, Φ_+, and Z under the chosen controls U and some admissible errors $\widehat{W}$ satisfying the constraint (3.4). For these matrices, equality (3.3) must hold under some matrix $\widehat{W}$ satisfying (3.4). Hence,

$$\mathbf{\Delta_p} = \left\{ \Delta : X_+ = \Delta \widehat{X} + \widehat{W}, \quad \widehat{W}\widehat{W}^{\mathrm{T}} \leqslant \Omega \right\},$$

and $\Delta \in \mathbf{\Delta_p}$ iff

$$(X_+ - \Delta \widehat{X})(X_+ - \Delta \widehat{X})^{\mathrm{T}} \leqslant \Omega. \tag{3.5}$$

It is obvious that $\Delta_{real} \in \mathbf{\Delta_p}$. For further use, we represent this last inequality as

$$(\Delta \quad I)\, \Psi^{(1)}\, (\Delta \quad I)^{\mathrm{T}} \leqslant 0, \tag{3.6}$$

where the symmetric matrix $\Psi^{(1)}$ of order $2n_x + n_u + n_y$ is partitioned into appropriate blocks $\Psi_{ij}^{(1)}$, $i, j = 1, 2$, as follows:

$$\Psi^{(1)} = \left(\begin{array}{c|c} \widehat{X}\widehat{X}^{\mathrm{T}} & \star \\ \hline -X_+ \widehat{X}^{\mathrm{T}} & X_+ X_+^{\mathrm{T}} - \Omega \end{array} \right). \tag{3.7}$$

Thus, the set of all matrices Δ consistent with the experimental data satisfies inequality (3.6).

Generally speaking, the set $\mathbf{\Delta_p}$ is unbounded. To establish its boundedness conditions, we denote by $\mathrm{Im}(\cdot)$, $\mathrm{Ker}(\cdot)$, $\mathrm{span}(\cdot)$, and $\mathrm{rank}(\cdot)$ the image, kernel, linear column subspace, and column rank of an appropriate matrix, respectively. Under the assumption $\mathrm{rank}\,\widehat{X} = s \leqslant \min\{n_x + n_u + n_y, N\}$, the matrix $\widehat{X}$ admits the singular decomposition [10]

$$\widehat{X} = (M_1\, M_2)\begin{pmatrix} \Sigma & 0_{s\times(N-s)} \\ 0_{(n_x+n_u+n_y)\times s} & 0_{(n_x+n_u+n_y)\times(N-s)} \end{pmatrix}\begin{pmatrix} G_1^{\mathrm{T}} \\ G_2^{\mathrm{T}} \end{pmatrix} = M_1\Sigma G_1^{\mathrm{T}},$$
$$M_1 \in \mathrm{R}^{(n_x+n_u+n_y)\times s},\ \ M_2 \in \mathrm{R}^{(n_x+n_u+n_y)\times(n_x+n_u+n_y-s)},\quad M = (M_1\, M_2)\,, \tag{3.8}$$

where $\Sigma = \mathrm{diag}(\lambda_1, \ldots, \lambda_s) > 0$, λ_i are the eigenvalues of the information matrix $\widehat{X}\widehat{X}^{\mathrm{T}}$, $\mathrm{span}\, M_1 = \mathrm{Im}\,\widehat{X}$, $\mathrm{span}\, M_2 = \mathrm{Ker}\,\widehat{X}^{\mathrm{T}}$, $\mathrm{span}\, G_1 = \mathrm{Im}\,\widehat{X}^{\mathrm{T}}$, $\mathrm{span}\, G_2 = \mathrm{Ker}\,\widehat{X}$, and $M^{\mathrm{T}}M = I$. Choosing the orthonormal basis of the columns of the matrix M, we introduce the corresponding variables

$$\widehat{\Delta} = \Delta\,(M_1\, M_2) = \left(\widehat{\Delta}^{(1)}\,\widehat{\Delta}^{(2)}\right),\quad \widehat{\Delta}^{(1)} \in \mathrm{R}^{n_x\times s},\quad \widehat{\Delta}^{(2)} \in \mathrm{R}^{n_x\times(n_x+n_u+n_y-s)}$$

and denote $\widehat{X}^{(1)} = M_1^{\mathrm{T}}\widehat{X} = \Sigma G_1^{\mathrm{T}}$. In the new variables, the linear matrix regression (3.3) takes the form

$$X_+ = \widehat{\Delta}^{(1)}_{real}\widehat{X}^{(1)} + \widehat{W}, \tag{3.9}$$

where the matrix $\widehat{X}^{(1)}$ of dimensions $(s \times N)$ has a full column rank, and $\widehat{\Delta}^{(1)}_{real}$ is the "projection" of the matrix $\widehat{\Delta}_{real}$ into the subspace $\mathrm{Im}\,\widehat{X}$, i.e., its rows $\widehat{\Delta}^{(1)}_{real}$ are the projections of the rows of the matrix $\widehat{\Delta}_{real}$ into the subspace $\mathrm{Im}\,\widehat{X}$.

Lemma 3.1. *The set $\mathbf{\Delta_p}$ of all matrices consistent with the experimental data $\widehat{X} = \mathrm{col}\,(X, U, \Phi)$ that satisfy* (3.8) *is an unbounded degenerate "matrix ellipsoid" given by*

$$(\widehat{\Delta}^{(1)} - \widehat{\Delta}^{(1)}_{LS})\Sigma^2(\widehat{\Delta}^{(1)} - \widehat{\Delta}^{(1)}_{LS})^{\mathrm{T}} \leqslant \Gamma,\quad \widehat{\Delta}^{(2)} \in \mathrm{R}^{n_x\times(n_x+n_u+n_y-s)}, \tag{3.10}$$

where

$$\Gamma = \Omega + X_+[\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}\widehat{X}^{(1)} - I]X_+^{\mathrm{T}} \geqslant 0, \tag{3.11}$$

and $\widehat{\Delta}^{(1)}_{LS} = X_+\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}$ is the least-squares estimate of the matrix $\widehat{\Delta}^{(1)}_{real}$ in (3.9).

Corollary 3.1. *The set $\mathbf{\Delta_p}$ is bounded iff the rank condition*

$$\mathrm{rank}\begin{pmatrix} X \\ U \\ \Phi \end{pmatrix} = n_x + n_u + n_y \tag{3.12}$$

holds. In this case, the set $\mathbf{\Delta_p}$ consists of the matrices given by inequality (3.10) *in which $\widehat{\Delta}^{(1)} = \widehat{\Delta}$ and $\widehat{\Delta}^{(1)}_{LS} = \widehat{\Delta}_{LS}$.*

The proof of Lemma 3.1 is provided in the Appendix. By this lemma, in the general case, only the projection $\widehat{\Delta}^{(1)}_{real}$ of the unknown matrix into the subspace $\mathrm{Im}\,\widehat{X}$ can be identified from the obtained data. Under the rank condition (3.12), the matrix Δ_{real} in (3.3) is identifiable, and the matrix ellipsoid and the set $\mathbf{\Delta_p}$ are bounded. Note that the rank condition (3.12) holds only if the number of measurements is not smaller than the sum of the dimensions of the state, output, and

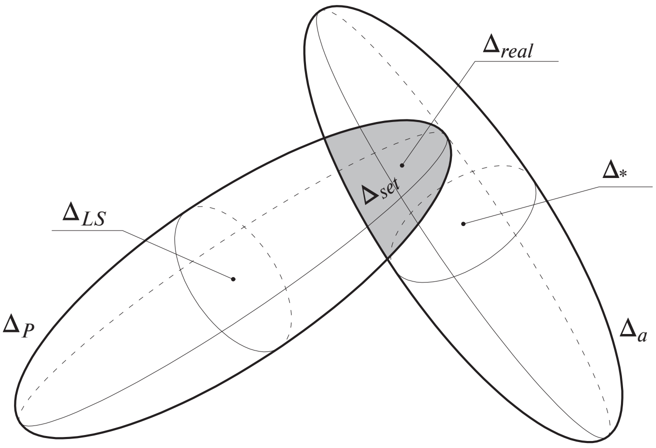

Fig. 1. The set $\mathbf{\Delta_{set}}$ of all unknown parameters Δ consistent with experimental data and a priori information.

control vectors: $N \geqslant n_x + n_u + n_y$. The robust control design procedure considered in this paper does not require the rank condition, and the number of measurements can therefore be less than $n_x + n_u + n_y$.

Now let there exist additional information that the unknown matrix Δ_{real} satisfies the constraint

$$(\Delta - \Delta_*)(\Delta - \Delta_*)^{\mathrm{T}} \leqslant \rho^2 I, \quad \Delta_* = (A_* \, B_* \, F_*)\,, \tag{3.13}$$

where the matrix Δ_* contains the corresponding ones of the nominal system and the parameter ρ characterizes the size of the uncertainty domain. We write this inequality as

$$(\Delta \quad I)\, \Psi^{(2)}\, (\Delta \quad I)^{\mathrm{T}} \leqslant 0, \tag{3.14}$$

where the matrix $\Psi^{(2)}$ consists of the blocks $\Psi^{(2)}_{ij}$, $i, j = 1, 2$, and has the form

$$\Psi^{(2)} = \left(\begin{array}{c|c} I & \star \\ \hline -\Delta_* & \Delta_* \Delta_*^{\mathrm{T}} - \rho^2 I \end{array} \right). \tag{3.15}$$

We introduce the following notations: $\mathbf{\Delta_a}$ is the set of all matrices Δ satisfying inequality (3.14), and $\mathbf{\Delta_{set}} = \mathbf{\Delta_p} \bigcap \mathbf{\Delta_a}$ is the set of all matrices Δ satisfying inequalities (3.6) and (3.14). Obviously, $\Delta_{real} \in \mathbf{\Delta_{set}}$. Figure 1 illustrates a possible arrangement of the sets $\mathbf{\Delta_p}$ and $\mathbf{\Delta_a}$ and their intersection $\mathbf{\Delta_{set}}$.

In the current notations, the optimal absolute stabilization problem of the unknown Lurie system (2.1) can be formulated as follows: it is required to design, directly from input and state measurements, a state-feedback controller $u = \Theta x$ under which, for all systems whose matrices are consistent with the a priori information and experimental data and whose nonlinear functions belong to the given sectors (2.3), the closed loop system will be absolutely stable and the performance index $J(\Theta)$ will be bounded above by a given constant:

$$J(\Theta) = \sup_{\Delta \in \mathbf{\Delta_{set}}} \; \sup_{\varphi(y,t)} \; \sup_{x_0 \neq 0} \frac{\|z\|^2}{x_0^{\mathrm{T}} R^{-1} x_0} < \gamma^2. \tag{3.16}$$

4. PRELIMINARY TRANSFORMATIONS AND AUXILIARY STATEMENTS

Consider the class of Lurie systems in which the components of the vector function $\varphi(y,t)$ satisfy condition (2.3). Before proceeding to the problem solution, we make some transformations to simplify the further presentation. Let us introduce a vector function $\widehat{\varphi}(y,t)$ with the components

$$\widehat{\varphi}_i(y_i,t) = \frac{1}{\beta_i - \alpha_i}[\varphi_i(y_i,t) - \alpha_i y_i], \quad i = 1,\dots,n_y. \tag{4.1}$$

Then equations (2.1) and (2.2) take the form

$$\begin{aligned} \partial x(t) &= (A + F\Lambda_1 L^{\mathrm{T}})x(t) + Bu(t) + F\Lambda_2\widehat{v}(t),\\ z(t) &= Cx(t) + Du(t), \end{aligned} \tag{4.2}$$

where $\Lambda_1 = \operatorname{diag}(\alpha_1,\dots,\alpha_{n_y})$, $\Lambda_2 = \operatorname{diag}(\beta_1 - \alpha_1,\dots,\beta_{n_y} - \alpha_{n_y})$, and

$$\widehat{v} = \widehat{\varphi}(y,t), \quad y = L^{\mathrm{T}}x, \tag{4.3}$$

while the functions $\widehat{\varphi}_i(y_i,t)$ satisfy the constraints (2.3) for $\alpha_i = 0$ and $\beta_i = 1$, $i = 1,\dots,n_y$, i.e., belong to the sector $[0,1]$.

We show that a Lyapunov function ensuring the absolute stability of the Lurie system with the guaranteed value of the quadratic performance index can be found by solving the corresponding worst-case disturbance problem in the linear system; for details, see [11]. Namely, for the Lurie system

$$\begin{aligned} \partial x(t) &= \mathcal{A}x(t) + \mathcal{F}v(t),\\ z(t) &= Cx(t), \end{aligned} \tag{4.4}$$

$$v(t) = \varphi(y(t),t), \quad y(t) = L^{\mathrm{T}}x(t), \tag{4.5}$$

with a stable matrix $\mathcal{A}$ and a vector function $\varphi(y,t)$ whose components lie in the sector $[0,1]$, we have the following result.

Lemma 4.1. *Assume that along the trajectories of the linear discrete- or continuous-time system* (4.4), *a function* $V(x) = x^{\mathrm{T}}Yx$ *with* $0 < Y = Y^{\mathrm{T}} < \gamma^2 R^{-1}$ *satisfies the inequality*

$$\triangle V + |z|^2 - v^{\mathrm{T}}\Gamma^{-1}(v - L^{\mathrm{T}}x) < 0, \quad \dot{V} + |z|^2 - v^{\mathrm{T}}\Gamma^{-1}(v - L^{\mathrm{T}}x) < 0, \tag{4.6}$$

where $\Gamma = \operatorname{diag}(\gamma_1,\dots,\gamma_{n_y}) > 0$, *for all* x *and* v $(|x|^2 + |v|^2 \neq 0)$. *Then the function* $V(x)$ *ensures the absolute stability of the Lurie system* (4.4), (4.5) *and, in addition,* $\|z\|^2 < \gamma^2 x_0^{\mathrm{T}} R^{-1} x_0$.

Remark 1. With the change of variables $\widehat{v} = \Gamma^{-1/2}(v - \frac{1}{2}L^{\mathrm{T}}x)$ and the performance output $\widehat{z} = \operatorname{col}(C, \frac{1}{2}\Gamma^{-1/2}L^{\mathrm{T}})x$, equations (4.4) become

$$\begin{aligned} \partial x(t) &= \left(\mathcal{A} + \frac{1}{2}\mathcal{F}L^{\mathrm{T}}\right)x(t) + \mathcal{F}\Gamma^{1/2}\widehat{v}(t),\\ \widehat{z}(t) &= \begin{pmatrix} C \\ \frac{1}{2}\Gamma^{-1/2}L^{\mathrm{T}} \end{pmatrix} x(t), \end{aligned} \tag{4.7}$$

where $\mathcal{A} + \frac{1}{2}\mathcal{F}L^{\mathrm{T}}$ is a Hurwitz matrix, and inequality (4.6) turns into $\dot{V} + |\widehat{z}|^2 - |\widehat{v}|^2 < 0$. Due to $Y < \gamma^2 R^{-1}$, this inequality is equivalent to the condition

$$\sup_{x_0,\,\widehat{v}} \frac{\|\widehat{z}\|^2}{x_0^{\mathrm{T}}\gamma^2 R^{-1}x_0 + \|\widehat{v}\|^2} < 1.$$

This means that the generalized H_∞ norm from the input $\widehat{v}$ to the output $\widehat{z}$ of system (4.7) with the weight matrix $\gamma^{-2}R$ is smaller than 1. As is easily checked, in the case of one nonlinearity ($z \equiv 0$), the resulting frequency condition $\|H\|_\infty < 1$ is equivalent to the circle criterion for absolute stability [12].

The next auxiliary statement characterizes the generalized H_∞ norm of a linear stable system

$$\begin{gathered} \partial x(t) = \mathcal{A}x(t) + \mathcal{B}v(t), \\ z(t) = \mathcal{C}x(t) \end{gathered} \tag{4.8}$$

in terms of the dual system.

Lemma 4.2 [3]. *The generalized H_∞ norm of system* (4.8) *with a weight matrix $\mathcal{R} > 0$ is smaller than* 1 *iff there exists a positive definite quadratic form $V_d(x_d) = x_d^{\mathrm{T}} P x_d$ with $P > \mathcal{R}$ such that*

$$\triangle V_d + |z_d|^2 - |v_d|^2 < 0, \quad \dot{V}_d + |z_d|^2 - |v_d|^2 < 0 \tag{4.9}$$

along the trajectories of the dual system

$$\begin{gathered} \partial x_d(t) = \mathcal{A}^{\mathrm{T}} x_d(t) + \mathcal{C}^{\mathrm{T}} v_d(t), \\ z_d(t) = \mathcal{B}^{\mathrm{T}} x_d(t) \end{gathered} \tag{4.10}$$

for all x_d and v_d ($|x_d|^2 + |v_d|^2 \neq 0$).

Remark 2. The matrices of the quadratic forms $V(x) = x^{\mathrm{T}} Y x$ and $V_d(x_d) = x_d^{\mathrm{T}} P x_d$ of the primal and dual systems are related by $P = Y^{-1}$.

Summarizing the above auxiliary statements and remarks, we arrive at the following result.

Theorem 4.1. *The closed-loop Lurie system* (2.1)–(2.3) *with given matrices A, B, and F and the state-feedback controller $u = \Theta x$ is absolutely stable and the performance index* (2.4) *is bounded above by a given constant γ^2 if the generalized H_∞ norm from the input v to the output z of the linear system*

$$\begin{gathered} \partial x(t) = (A + B\Theta + F\Lambda L^{\mathrm{T}})x(t) + F\Lambda_2 \Gamma^{1/2} v(t), \\ z(t) = \begin{pmatrix} C + D\Theta \\ \frac{1}{2}\Gamma^{-1/2} L^{\mathrm{T}} \end{pmatrix} x(t), \end{gathered} \tag{4.11}$$

with the weight matrix $\gamma^{-2}R$ and $\Lambda = \Lambda_1 + \frac{1}{2}\Lambda_2$ is smaller than 1.

Corollary 4.1. *In view of Lemma 4.2, the closed-loop Lurie system* (2.1)–(2.3) *is absolutely stable and the upper bound* (2.4) *holds if there exists a function $V(x_d) = x_d^{\mathrm{T}} P x_d$ with $P = P^{\mathrm{T}} > \gamma^{-2}R$ satisfying inequality* (4.9) *along the trajectories of the linear system*

$$\begin{gathered} \partial x_d(t) = (A + B\Theta + F\Lambda L^{\mathrm{T}})^{\mathrm{T}} x_d(t) + \begin{pmatrix} C + D\Theta \\ \frac{1}{2}\Gamma^{-1/2} L^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} v_d(t), \\ z_d(t) = \Gamma^{1/2} \Lambda_2 F^{\mathrm{T}} x_d(t). \end{gathered} \tag{4.12}$$

For system (4.12), inequality (4.9) can be written as an LMI, and we get another result.

Corollary 4.2. *System* (2.1)–(2.3) *with given matrices* A, B, *and* F *is absolutely stable and the upper bound* (2.4) *holds under the control law* $u = \Theta x$, *where* $\Theta = QP^{-1}$ *and* $P = P^{\mathrm{T}} > 0$, Q, $\Gamma = \operatorname{diag}(\gamma_1, \ldots, \gamma_{n_y}) > 0$, *and* $\gamma^2 > 0$ *satisfy the following LMIs:*

—in the discrete-time case,

$$\begin{pmatrix} -P & \star & \star & \star \\ AP + BQ + F\Lambda L^{\mathrm{T}}P & -P + F\Lambda_2\Gamma\Lambda_2F^{\mathrm{T}} & \star & \star \\ CP + DQ & 0 & -I & \star \\ \dfrac{1}{2}L^{\mathrm{T}}P & 0 & 0 & -\Gamma \end{pmatrix} < 0, \tag{4.13}$$

$$\begin{pmatrix} P & \star \\ I & \gamma^2R^{-1} \end{pmatrix} > 0;$$

—in the continuous-time case,

$$\begin{pmatrix} AP + PA^{\mathrm{T}} + BQ + Q^{\mathrm{T}}B^{\mathrm{T}} + F\Lambda L^{\mathrm{T}}P + PL\Lambda F^{\mathrm{T}} + F\Lambda_2\Gamma\Lambda_2F^{\mathrm{T}} & \star & \star \\ CP + DQ & -I & \star \\ \dfrac{1}{2}L^{\mathrm{T}}P & 0 & -\Gamma \end{pmatrix} < 0, \tag{4.14}$$

$$\begin{pmatrix} P & \star \\ I & \gamma^2R^{-1} \end{pmatrix} > 0.$$

5. THE DESIGN PROCEDURE OF OPTIMAL ABSOLUTELY STABILIZING CONTROLLERS

Now we write the equations of the closed-loop unknown system (2.1)–(2.3) in the form

$$\begin{aligned} \partial x(t) &= (A + B\Theta + F\Lambda_1L^{\mathrm{T}})x(t) + F\Lambda_2\widehat{v}(t), \\ z(t) &= (C + D\Theta)x(t), \end{aligned} \tag{5.1}$$

$$\widehat{v} = \widehat{\varphi}(y, t), \quad y = L^{\mathrm{T}}x, \tag{5.2}$$

where the components of the vector function $\widehat{\varphi}(y, t)$ are given by (4.1) and belong to the sector $[0, 1]$. In the theorem below, the parameters of the linear state-feedback controllers ensuring the absolute stability of the unknown nonlinear Lurie system and the guaranteed value of the performance index are expressed in terms of experimental data and a priori information.

Theorem 5.1. *The Lurie system* (2.1)–(2.3) *with the state-feedback controller* $u = \Theta x$ *is absolutely stable and the performance index* (3.16) *is bounded above,* $J(\Theta) < \gamma^2$, *if* $\Theta = QP^{-1}$, *where* $P = P^{\mathrm{T}} > 0$, Q, $\Gamma = \operatorname{diag}(\gamma_1, \ldots, \gamma_{n_y}) > 0$, $\gamma^2 > 0$, $\mu_1 \geqslant 0$, *and* $\mu_2 \geqslant 0$ *satisfy the following LMIs:*

—in the discrete-time case,

$$
\begin{pmatrix}
-P & \star & \star & \star \\
\begin{pmatrix} P \\ Q \\ \Lambda L^{\mathrm{T}} P \end{pmatrix} & \widehat{\Lambda}_2 \Gamma \widehat{\Lambda}_2^{\mathrm{T}} - \sum\limits_{k=1}^{2} \mu_k \Psi_{11}^{(k)} & \star & \star \\
0 & -\sum\limits_{k=1}^{2} \mu_k \Psi_{21}^{(k)} & -P - \sum\limits_{k=1}^{2} \mu_k \Psi_{22}^{(k)} & \star \\
\begin{pmatrix} C & D & 0 \\ 0 & 0 & \dfrac{1}{2} I \end{pmatrix} \begin{pmatrix} P \\ Q \\ L^{\mathrm{T}} P \end{pmatrix} & 0 & 0 & -\begin{pmatrix} I & \star \\ 0 & \Gamma \end{pmatrix}
\end{pmatrix} < 0,
\tag{5.3}
$$

$$
\begin{pmatrix} P & \star \\ I & \gamma^2 R^{-1} \end{pmatrix} > 0;
$$

—in the continuous-time case,

$$
\begin{pmatrix}
\widehat{\Lambda}_2 \Gamma \widehat{\Lambda}_2^{\mathrm{T}} - \sum\limits_{k=1}^{2} \mu_k \Psi_{11}^{(k)} & \star & \star \\
\begin{pmatrix} P \\ Q \\ \Lambda L^{\mathrm{T}} P \end{pmatrix}^{\mathrm{T}} - \sum\limits_{k=1}^{2} \mu_k \Psi_{21}^{(k)} & -\sum\limits_{k=1}^{2} \mu_k \Psi_{22}^{(k)} & \star \\
0 & \begin{pmatrix} C & D & 0 \\ 0 & 0 & \dfrac{1}{2} I \end{pmatrix} \begin{pmatrix} P \\ Q \\ L^{\mathrm{T}} P \end{pmatrix} & -\begin{pmatrix} I & \star \\ 0 & \Gamma \end{pmatrix}
\end{pmatrix} < 0,
\tag{5.4}
$$

$$
\begin{pmatrix} P & \star \\ I & \gamma^2 R^{-1} \end{pmatrix} > 0.
$$

Here, $\widehat{\Lambda}_2 = \mathrm{col}(0, 0, \Lambda_2)$, *and* $\Psi_{ij}^{(k)}$ *are the corresponding blocks of the matrices* $\Psi^{(1)}$ *and* $\Psi^{(2)}$ *given by* (3.7) *and* (3.15), *respectively.*

Proof of Theorem 5.1. By Theorem 4.1, system (5.1), (5.2) is absolutely stable and $J(\Theta) < \gamma^2$ if the generalized H_∞ norm from the input v to the output z of the linear system (4.11) with the weight matrix $\gamma^{-2} R$ is smaller than 1. In turn, this condition holds iff the generalized H_∞ norm of the dual system (4.12) is smaller than 1 (Lemma 4.2). Using the notations introduced above, we represent equations (4.12) as

$$
\begin{aligned}
\partial x_d(t) &= \begin{pmatrix} I \\ \Theta \\ \Lambda L^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \left[\Delta^{\mathrm{T}} x_d(t) + \begin{pmatrix} C^{\mathrm{T}} & 0 \\ D^{\mathrm{T}} & 0 \\ 0 & \frac{1}{2} \Lambda^{-1} \Gamma^{-1/2} \end{pmatrix} v_d(t) \right], \\
z_d(t) &= \Gamma^{1/2} \widehat{\Lambda}_2^{\mathrm{T}} \Delta^{\mathrm{T}} x_d(t).
\end{aligned}
\tag{5.5}
$$

Consider an augmented system with the additional artificial input $w_\Delta(t) \in L_2$ and output $z_\Delta(t)$ described by the equations

$$\partial x_a(t) = \begin{pmatrix} I \\ \Theta \\ \Lambda L^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \left[w_\Delta(t) + \begin{pmatrix} C^{\mathrm{T}} & 0 \\ D^{\mathrm{T}} & 0 \\ 0 & \frac{1}{2}\Lambda^{-1}\Gamma^{-1/2} \end{pmatrix} v_a(t) \right], \tag{5.6}$$

$$z_a(t) = \Gamma^{1/2}\widehat{\Lambda}_2^{\mathrm{T}} w_\Delta(t), \quad z_\Delta(t) = x_a(t).$$

Note that for $w_\Delta(t) = \Delta^{\mathrm{T}} z_\Delta(t)$, equations (5.6) coincide with (5.5). Suppose that for all $t \geqslant 0$, the additional input and output signals in system (5.6) satisfy the two inequalities

$$\begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix}^{\mathrm{T}} \Psi^{(1)} \begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix} \leqslant 0, \quad \begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix}^{\mathrm{T}} \Psi^{(2)} \begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix} \leqslant 0, \tag{5.7}$$

where the matrices $\Psi^{(1)}$ and $\Psi^{(2)}$ are given by (3.7) and (3.15). Let $\mathbf{W}_{\boldsymbol{\Delta}}$ denote the set of all such signals $w_\Delta(t)$. Given $w_\Delta(t) = \Delta^{\mathrm{T}} z_\Delta(t)$, for all $\Delta \in \boldsymbol{\Delta}$, from (3.6) and (3.14) it follows that

$$\begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix}^{\mathrm{T}} \Psi^{(1)} \begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix} = z_\Delta^{\mathrm{T}}(t) \begin{pmatrix} \Delta^{\mathrm{T}} \\ I \end{pmatrix}^{\mathrm{T}} \Psi^{(1)} \begin{pmatrix} \Delta^{\mathrm{T}} \\ I \end{pmatrix} z_\Delta(t) \leqslant 0,$$

$$\begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix}^{\mathrm{T}} \Psi^{(2)} \begin{pmatrix} w_\Delta(t) \\ z_\Delta(t) \end{pmatrix} = z_\Delta^{\mathrm{T}}(t) \begin{pmatrix} \Delta^{\mathrm{T}} \\ I \end{pmatrix}^{\mathrm{T}} \Psi^{(2)} \begin{pmatrix} \Delta^{\mathrm{T}} \\ I \end{pmatrix} z_\Delta(t) \leqslant 0.$$

Thus, $w_\Delta(t) = \Delta^{\mathrm{T}} z_\Delta(t) \in \mathbf{W}_{\boldsymbol{\Delta}}$ and, consequently, for $\Delta \in \boldsymbol{\Delta}$ system (5.5) is "immersed" in the augmented system (5.6), (5.7).

The proof below is for the continuous-time case: in the discrete-time one, it can be repeated by analogy. We establish conditions for the existence of a positive definite quadratic function $V(x_a) = x_a^{\mathrm{T}} P x_a$ with $P > \gamma^{-2} R$ that satisfies, for all x_a and v_a ($|x_a|^2 + |v_a|^2 \neq 0$), the inequality

$$\dot{V} + |z_a|^2 - |v_a|^2 < 0 \tag{5.8}$$

along the trajectories of the augmented system (5.6) for all $w_\Delta(t)$ with (5.7).

By the S-procedure, a sufficient condition is the existence of a function $V(x_a) = x_a^{\mathrm{T}} P x_a$ with $P > \gamma^{-2} R$ that satisfies, for all x_a, v_a, and w_Δ ($|x_a|^2 + |v_a|^2 + |w_\Delta|^2 \neq 0$) and some $\mu_1, \mu_2 \geqslant 0$, the inequality

$$\dot{V} + |z_a|^2 - |v_a|^2 - \sum_{k=1}^{2} \mu_k \begin{pmatrix} w_\Delta \\ z_\Delta \end{pmatrix}^{\mathrm{T}} \Psi^{(k)} \begin{pmatrix} w_\Delta \\ z_\Delta \end{pmatrix} < 0 \tag{5.9}$$

along the trajectories of the augmented system (5.6).

This inequality reduces to an inequality for a quadratic form in the variables w_Δ, x_a, and v_a with the matrix

$$\begin{pmatrix} \widehat{\Lambda}_2 \Gamma \widehat{\Lambda}_2^{\mathrm{T}} - \sum\limits_{k=1}^{2} \mu_k \Psi_{11}^{(k)} & \star & \star \\ \begin{pmatrix} P \\ Q \\ \Lambda L^{\mathrm{T}} P \end{pmatrix}^{\mathrm{T}} - \sum\limits_{k=1}^{2} \mu_k \Psi_{21}^{(k)} & -\sum\limits_{k=1}^{2} \mu_k \Psi_{22}^{(k)} & \star \\ 0 & \begin{pmatrix} I & 0 \\ 0 & \Gamma^{-1/2} \end{pmatrix} \begin{pmatrix} C & D & 0 \\ 0 & 0 & \dfrac{1}{2}\Lambda^{-1} \end{pmatrix} \begin{pmatrix} P \\ Q \\ \Lambda L^{\mathrm{T}} P \end{pmatrix} & -I \end{pmatrix}.$$

Multiplying this matrix on the left and right by diag $\left(I, I, \begin{pmatrix} I & 0 \\ 0 & \Gamma^{1/2} \end{pmatrix}\right)$ gives the matrix in the left-hand side of (5.4). For $\Delta \in \mathbf{\Delta}$, system (5.5) is immersed into the augmented system (5.6), (5.7). Therefore, we have the inequality $\dot{V} + |z_d|^2 - |v_d|^2 < 0$ for system (5.5) for all $\Delta \in \mathbf{\Delta}$, and its generalized H_∞ norm with the weight matrix $\gamma^{-2}R$ is smaller than 1. The proof of this theorem is complete.

Remark 3. According to the lossless S-procedure under two quadratic constraints (Theorem 4.1 in [13]), if $\mu_1\Psi^{(1)} + \mu_2\Psi^{(2)} > 0$ for some μ_1 and μ_2 (this LMI can be directly solved with respect to μ_1 and μ_2 after forming the particular matrix $\Psi^{(1)}$), then the corresponding inequality (5.9) is a sufficient and also necessary condition for the existence of the above function $V_a(x_a) = x_a^{\mathrm{T}} P x_a$ for the augmented system.

Remark 4. In the control design procedure based on only experimental data or only a priori information, due to the lossless S-procedure with one constraint, the conditions of Theorem 5.1 are sufficient and also necessary to fulfill inequality (5.8) along the trajectories of the augmented system (5.6), (5.7).

We denote by γ_*, γ_a, and γ_p the minimum upper bounds for the performance index $J(\Theta)$ that can be achieved using the control laws designed from experimental data and a priori information, only from a priori information, and only from experimental data, respectively, by Theorem 5.1. These bounds will be called guaranteed. The minimum values of γ for which inequalities (5.4) are solvable under $\mu_k \geqslant 0$, $k = 1, 2$, do not exceed, first, the minimum values of γ under $\mu_1 \equiv 0$ and $\mu_2 \geqslant 0$ and, second, the minimum values of γ under $\mu_1 \geqslant 0$ and $\mu_2 \equiv 0$. Therefore, Theorem 5.1 directly leads to the inequality

$$\gamma_* \leqslant \min\{\gamma_a, \gamma_p\},$$

which explains the advantage of control laws based on both a priori information and experimental data over those based on only a priori information or only experimental data. On the one hand, given rough a priori information (i.e., when the radius ρ of the matrix sphere in (3.13) is large enough and, accordingly, γ_a takes a high value), the index γ_* may turn out to be small if the measurement noise is not very significant (i.e., if the matrix ellipsoid $\mathbf{\Delta_p}$ is small). On the other hand, if the measurement noise turns out to be significant and, accordingly, γ_p is large (furthermore, if the information matrix is singular and the matrix ellipse $\mathbf{\Delta_p}$ is unbounded), then γ_* can nevertheless become small due to the smallness of the radii of the matrix sphere when using the a priori information. These conclusions will be confirmed by the simulation results in Section 6.

6. AN ILLUSTRATIVE EXAMPLE: A NONLINEAR OSCILLATOR

Let us design an absolutely stabilizing control law for the discrete-time model

$$x(t+1) = \begin{pmatrix} 1 & h \\ 0 & 1 - \delta h \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ h \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ -h\omega^2 \end{pmatrix} \varphi(y(t)) + \begin{pmatrix} 0 \\ h \end{pmatrix} w(t),$$

$$y(t) = (1\ 0)x(t), \quad z = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0.1 \end{pmatrix} u$$

of the nonlinear system

$$\ddot{\psi} + \delta\dot{\psi} + \omega^2\varphi(\psi) = u + w,$$

where $x = \mathrm{col}(\psi, \dot{\psi})$ and the nonlinearity $\varphi(\psi)$ satisfies condition (2.3) for $\alpha = -2/3\pi$ and $\beta = 1$. When implementing the control design procedure for a continuous-time system, it is necessary to calculate the derivatives, which entails an additional disturbance with bounds difficult to estimate in

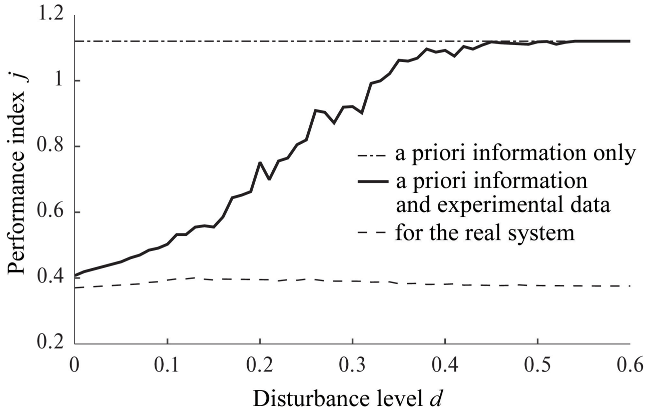

Fig. 2. The guaranteed values of the performance index under the robust controller based on experimental data and a priori information and its values under the same controller for the real system as functions of the disturbance level.

advance. In this sense, the discrete-time one is more preferable. In the experiment, it was assumed that the real system is a nonlinear oscillator with $\varphi(\psi) = \sin\psi$, a damping coefficient of $\delta = 0.1$, and a frequency of $\omega^2 = 1$; for the nominal system, $\delta_* = 0$ and $\omega_*^2 = 0.8$. The step $h = 0.2$, the weight matrix $R = 0.1I$, and the a priori uncertainty radius $\rho = 0.05$ were chosen. For each value of the disturbance level d, ten measurements were carried out, i.e., $N = 10$. In the experiment, the initial conditions and control were selected random in the interval $[-1, 1]$ and the disturbance $w(t)$ random in the interval $[-d, d]$. Inequalities (5.3) were solved using the CVX package, with "a slight departure from zero" to solve strict inequalities.

The figures demonstrate the simulation results obtained by averaging over 20 independent experiments. The following conclusions can be drawn from Fig. 2. As the disturbance level grows, the guaranteed value γ_* of the performance index calculated from the experimental data and a priori information increases, while remaining much smaller (under relatively small disturbance levels) than its counterpart γ_a obtained based on only the a priori information. The explanation is that the set $\mathbf{\Delta_p}$ of all systems consistent with the experimental data expands when increasing the disturbance level. Starting from some disturbance level (in the current example, approximately $d = 0.5$), the set $\mathbf{\Delta_p}$ includes the set $\mathbf{\Delta_a}$ of all systems separated based on the a priori information; therefore, with further increase of the disturbance level, γ_* stops growing and $\gamma_* = \gamma_a$. The dotted curve in Fig. 2 corresponds to the values of the performance index for the real system (if it were known) under the obtained robust controller based on the experimental data and a priori information under different disturbance levels. According to the experiments, this value weakly depends on the disturbance level and is close enough to the optimal value for the known system, i.e., $\gamma^2 \approx 0.39$.

The optimal controller and the corresponding value of the performance index for the real system (if it were known), computed by solving the LMIs (4.13), are as follows:

$$u = -6.58x_1 - 4.59x_2, \quad \gamma^2 = 0.36.$$

The robust controller and the corresponding value of the performance index, computed only from the a priori information using the LMIs (5.3) with $\mu_1 = 0$, are as follows:

$$u = -6.45x_1 - 5.38x_2, \quad \gamma_a^2 = 1.12.$$

The robust controller and the corresponding value of the performance index computed from the a priori information and experimental data in one of the experiments with the disturbance level $d = 0.1$ using the LMIs (5.3) with $\mu_1 \geqslant 0$ and $\mu_2 \geqslant 0$ are as follows:

$$u = -9.35x_1 - 6.61x_2, \quad \gamma_*^2 = 0.52.$$

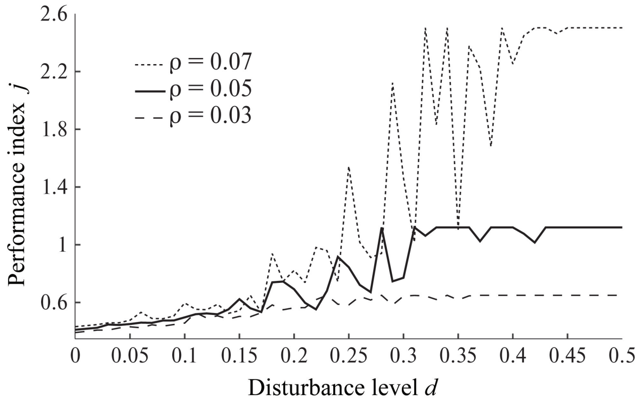

Fig. 3. The guaranteed values of the performance index under the robust controller based on experimental data and a priori information with different radii of matrix spheres as functions of the disturbance level.

As shown by the experiment, if the robust control law is built only from experimental data without a priori information (i.e., it is calculated using the LMIs (5.3) with $\mu_2 = 0$), then very large guaranteed values of the performance index are observed even under relatively small disturbance levels. The explanation is that even for small random disturbances the ellipsoid $\mathbf{\Delta_p}$ can be large enough or even degenerate (i.e., unbounded). Thus, considering a priori information has a regularizing effect on the robust control design procedure based on experimental data, even in the case when there is no unique robust controller on the entire set $\mathbf{\Delta_a}$ of systems identified from a priori information.

Figure 3 illustrates how increasing the radius of the matrix sphere in the a priori information affects the guaranteed value of the performance index under the robust controller obtained from the a priori information and experimental data.

7. CONCLUSIONS

In this paper, we have developed a method for designing an absolutely stabilizing control law for unknown Lurie systems that ensures the guaranteed value of the integral quadratic performance index characterizing the closed-loop system transients under uncertain initial conditions. The experimental data are not subject to the persistent excitation condition of the system, which is necessary for its identifiability. The resulting LMIs for calculating the feedback parameters serve to find the control laws based on only a priori information, only experimental data, and both a priori information and experimental data. The simulation results of experiments with a nonlinear oscillator have confirmed the advantage of control laws designed from experimental data and a priori information over those obtained using only experimental data or only a priori information.

FUNDING

This work was supported by the Russian Science Foundation, project no. 24-11-20023.

APPENDIX

Proof of Lemma 3.1. We write inequality (3.5) as

$$\Delta\widehat{X}\widehat{X}^{\mathrm{T}}\Delta^{\mathrm{T}} - X_+\widehat{X}^{\mathrm{T}}\Delta^{\mathrm{T}} - \Delta\widehat{X}X_+^{\mathrm{T}} + X_+X_+^{\mathrm{T}} - \Omega \leqslant 0.$$

With the change of variables, it becomes

$$\widehat{\Delta}^{(1)}\Sigma^2\widehat{\Delta}^{(1)\mathrm{T}} - X_+\widehat{X}^{(1)\mathrm{T}}\widehat{\Delta}^{(1)\mathrm{T}} - \widehat{\Delta}^{(1)}\widehat{X}^{(1)}X_+^{\mathrm{T}} + X_+X_+^{\mathrm{T}} - \Omega \leqslant 0.$$

Completing the square yields

$$[\widehat{\Delta}^{(1)} - X_+\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}]\Sigma^2[\widehat{\Delta}^{(1)} - X_+\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}]^{\mathrm{T}} \leqslant \Gamma,$$

where Γ is given by (3.11). Due to the expression for X_+ (3.9) and $\widehat{X}^{(1)}\widehat{X}^{(1)\mathrm{T}} = \Sigma^2$, we obtain $\Gamma = \Omega + W(\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}\widehat{X}^{(1)} - I)W^{\mathrm{T}}$. In view of (3.4), it follows that $\Gamma \geqslant 0$. Consider the matrix norm of the residual, i.e., the function $\operatorname{tr}(X_+ - \widehat{\Delta}^{(1)}\widehat{X}^{(1)})^{\mathrm{T}}(X_+ - \widehat{\Delta}^{(1)}\widehat{X}^{(1)})$. Equating its gradient with respect to $\widehat{\Delta}^{(1)}$ to zero, $-2X_+\widehat{X}^{(1)\mathrm{T}} + 2\widehat{\Delta}^{(1)}\widehat{X}^{(1)}\widehat{X}^{(1)\mathrm{T}} = 0$, we finally get the least-squares estimate $\Delta_{LS}^{(1)}$ of the unknown matrix $\Delta_{real}^{(1)}$ in (3.9) in the form $\widehat{\Delta}_{LS}^{(1)} = X_+\widehat{X}^{(1)\mathrm{T}}\Sigma^{-2}$.

Proof of Lemma 4.1. Due to $\varphi_i(y_i, t)[\varphi_i(y_i, t) - y_i] \leqslant 0$, letting $v = \varphi(y, t)$ in (4.6) yields the inequality $\triangle V + |z|^2 < 0$ along the trajectories of system (4.4), (4.5). In other words, $\dot{V} + |z|^2 < 0$ and, consequently, $\lim_{t\to\infty} x(t) = 0$. Since $Y < \gamma^2 R^{-1}$, after summation and integration we finally arrive at $\|z\|^2 < \gamma^2 x_0^{\mathrm{T}} R^{-1} x_0$.

REFERENCES

1. Lurie, A.I., *Nekotorye nelineinye zadachi teorii avtomaticheskogo regulirovaniya* (Some Nonlinear Problems in the Theory of Automatic Regulation), Moscow: Gostekhizdat, 1951.

2. Kogan, M.M. and Stepanov, A.V., Design of Suboptimal Robust Controllers Based on A Priori and Experimental Data, *Autom. Remote Control*, 2023, vol. 84, no. 8, pp. 918–932.

3. Kogan, M.M. and Stepanov, A.V., Design of Generalized H_∞-suboptimal Controllers Based on Experimental and A Priori Data, *Autom. Remote Control*, 2024, vol. 85, no. 1, pp. 1–14.

4. Kogan, M.M. and Stepanov, A.V., How to Improve Robust Control of a Linear Time-Varying System by Using Experimental Data, *Autom. Remote Control*, 2024, vol. 85, no. 6, pp. 636–654.

5. Petersen, I.R. and Tempo, R., Robust Control of Uncertain Systems: Classical Results and Recent Developments, *Automatica*, 2014, vol. 50, no. 5, pp. 1315–1335.

6. De Persis, C. and Tesi, P., Formulas for Data-Driven Control: Stabilization, Optimality and Robustness, *IEEE Trans. Automat. Control*, 2020, vol. 65, no. 3, pp. 909–924.

7. Waarde, H.J., Eising, J., Trentelman, H.L., and Camlibel, M.K., Data Informativity: a New Perspective on Data-Driven Analysis and Control, *IEEE Trans. Automat. Control*, 2020, vol. 65, no. 11, pp. 4753–4768.

8. Berberich, J., Scherer, C.W., and Allgower, F., Combining Prior Knowledge and Data for Robust Controller Design, *IEEE Trans. Automat. Control*, 2023, vol. 68, no. 8, pp. 4618–4633.

9. Luppi, A., De Persis, C., and Tesi, P., On Data-Driven Stabilization of Systems with Quadratic Nonlinearities, *arXiv: 2103.15631v1 [eess.SY]*, March 29, 2021.

10. Horn, R.A. and Johnson, Ch.R., *Matrix Analysis*, Cambridge University Press, 1990.

11. Kogan, M.M., Minimax Approach to the Synthesis of Absolutely Stabilizing Controllers for Nonlinear Lurie Systems, *Autom. Remote Control*, 1999, vol. 60, no. 5, pp. 667–677.

12. Gelig, A.Kh., Leonov, G.A., and Yakubovich, V.A., *Ustoichivost' nelineinykh sistem s needinstvennym sostoyaniem ravnovesiya* (Stability of Nonlinear Systems with Nonunique Equilibrium), Moscow: Nauka, 1978.

13. Polyak, B.T., Convexity of Quadratic Transformations and Its Use in Control and Optimization, *J. Optim. Theory Appl.*, 1998, vol. 99, no. 3, pp. 553–583.

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

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 2, pp. 113–133.

Towards Computation of Surface Area of Schur Stability Domain

M. V. Kukoverov

Moscow Integrated Power Company, Moscow, Russia
e-mail: gedevan_space@mail.ru

Received February 7, 2024
Revised November 11, 2024
Accepted November 25, 2024

Abstract—The paper considers the subset of the Schur stability domain, namely, the set of parameters for which the roots of polynomials of degree n lie in the complex unit disc and are real numbers. The method for computation of the hypersurface area of this defined set in n-dimensional space is obtained. The maximum surface area of this set is reached for $n = 3$, i.e., 3-dimensional set has maximum surface area.

Keywords: multiple integrals, Schur stability domain, autoregression, stationary autoregressive processes, unit root processes

DOI: 10.31857/S0005117925020025

1. INTRODUCTION

A significant part of recent research was devoted to the study of Schur stability domain and its boundary [1]. Though the question of determining the size of this domain's boundary, i.e. domain's hypersurface area, is still open.

We study polinomials of the form

$$x^n - \alpha_1 x^{n-1} - \ldots - \alpha_{n-1} x - \alpha_n = 0, \tag{1}$$

with parameters $\alpha_i \in R,\ i = 1, \ldots, n$.

Each polinomial corresponds to a single point $(\alpha_1, \ldots, \alpha_n)$ in n-dimensional Euclidean space R^n.

Schur stability domain is defined as a set in R^n, for which roots of corresponding polinomials lie in unit disc on the complex plane. We denote this set as D_n, as in [2]. That paper provides formula for computation of the volume $V(D_n)$.

We denote as $\mathcal{E}_n$, as in [3], the subset of D_n where roots of polinomials are real numbers with zero imaginary part. That paper provides formula for computation of the volume $V(\mathcal{E}_n)$ with a multiple integral, that is related to the Selberg type integral.

Volumes of stability domains [2, 3]

n	$V(D_n)$	$V(\mathcal{E}_n)$	$V(D_n\backslash\mathcal{E}_n)$	$\dfrac{V(\mathcal{E}_n)}{V(D_n)}$	$\dfrac{V(D_n\backslash\mathcal{E}_n)}{V(D_n)}$
"1"	"2"	"3"	"4"	"5"	"6"
1	2	2	0	1	0
2	4	1.333	2.667	0.3333	0.6667
3	5.333	0.355	4.978	0.0667	0.9333
4	7.111	0.041	7.070	0.0057	0.9943
5	7.585	0.002	7.583	0.0003	0.9997

Volumes $V(D_n)$ and $V(\mathcal{E}_n)$ reach maximum with $n = 6$ and $n = 1$ respectively and are infinitesimal as $n \to \infty$.[1]

The boundary ∂D_n of domain D_n is the union of two hyperplanes and one hypersurface [1, 4]. The hyperplanes correspond to roots -1 and 1.

The boundary $\partial\mathcal{E}_n$ of domain $\mathcal{E}_n$ consists of two hyperplanes, that correspond to hyperplanes of ∂D_n, and one hypersurface inside of D_n.

This paper presents results of computation of hypersurface area $S(\partial\mathcal{E}_n)$ of the boundary $\partial\mathcal{E}_n$ of the domain $\mathcal{E}_n$.

2. COMPUTATION OF $S(\partial\mathcal{E}_2)$

Consider polinom of degree 2:

$$x^2 - \alpha_1 x - \alpha_2 = 0. \tag{2}$$

Domain D_2 is the set of parameters:

$$\begin{cases} -2 < \alpha_1 < 2 \\ -1 < \alpha_2 < 1 - |\alpha_1|. \end{cases}$$

Domain $\mathcal{E}_2$ is the set of parameters:

$$\begin{cases} -2 < \alpha_1 < 2 \\ -\alpha_1^2/4 < \alpha_2 < 1 - |\alpha_1|. \end{cases}$$

From econometric point of view the picture shows stationarity domains of parameters that correspond to stationary autoregressive processes of the second order, while the outer boundary corresponds to unit root processes and the outer region corresponds to explosive processes.

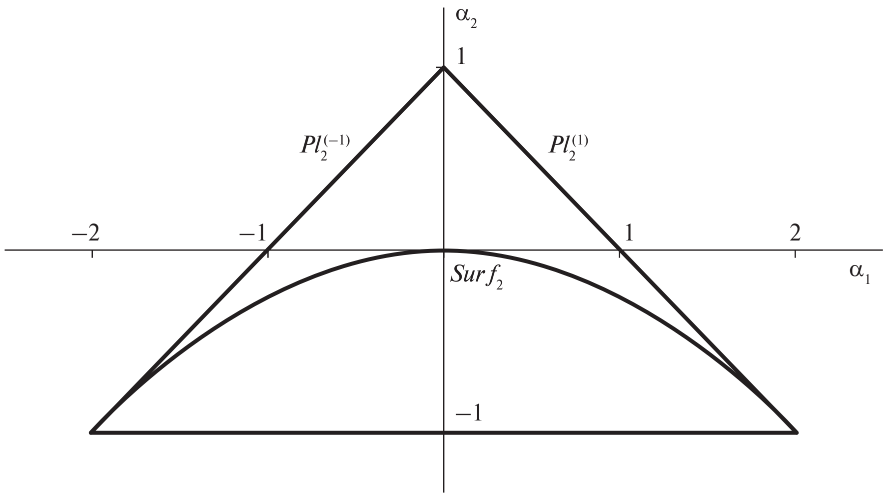

Fig. 1. Domains D_2 and $\mathcal{E}_2$. Source: [5].

[1] Though comparing values of not equal dimensions, such as length, area, volume, etc. may seem strange enough, sometimes it may also be curious. For instance, see study of these issues for hyperball and hypersphere:
https://mathworld.wolfram.com/Hypersphere.html
https://mathworld.wolfram.com/Ball.html

The boundary $\partial\mathcal{E}_2$ consists of parabola section $Surf_2$ and two line segments $Pl_2^{(1)}$ and $Pl_2^{(-1)}$, that correspond to roots 1 and -1. I.e. polinom (2) with parameters from $Pl_2^{(1)}$ has root, equal to 1, and polinom (2) with parameters from $Pl_2^{(-1)}$ has root, equal to -1.

The boundary length $S(\partial\mathcal{E}_2)$ can be computed as a sum of lengths:

$$S(\partial\mathcal{E}_2) = S(Pl_2^{(1)}) + S(Pl_2^{(-1)}) + S(Surf_2) \approx 2\sqrt{2} + 2\sqrt{2} + 4.591 \approx 10.248.$$

3. COMPUTATION OF $S(\partial\mathcal{E}_3)$

Consider polinom of degree 3:

$$x^3 - \alpha_1 x^2 - \alpha_2 x - \alpha_3 = 0. \tag{3}$$

According to [6], domain D_3 is a set of parameters:

$$\begin{cases} -1 < \alpha_3 < 1 \\ -3 < \alpha_1 < 3 \\ \alpha_2 < 1 - |\alpha_1 + \alpha_3| \\ \alpha_2 > -1 + \alpha_3^2 - \alpha_1\alpha_3. \end{cases}$$

Domain $\mathcal{E}_3$ is a set of parameters:

$$\begin{cases} -1 < \alpha_3 < 1 \\ -3 < \alpha_1 < 3 \\ \alpha_2 < 1 - |\alpha_1 + \alpha_3| \\ \alpha_1^2\alpha_2^2 + 4\alpha_2^3 - 4\alpha_1^3\alpha_3 - 27\alpha_3^2 - 18\alpha_1\alpha_2\alpha_3 > 0. \end{cases}$$

The boundary $\partial\mathcal{E}_3$ consists of the surface $Surf_3$ and two plane sections $Pl_3^{(1)}$ and $Pl_3^{(-1)}$, that correspond to roots 1 and -1. I.e. polinom (3) with parameters from section $Pl_3^{(1)}$ has a root, equal to 1, and polinom (3) with parameters from section $Pl_3^{(-1)}$ has a root, equal to -1. These surfaces can be parameterized applying Vieta's formulas.

Parametric form of $Pl_3^{(1)}$:

$$\begin{cases} \alpha_1 = 1 + x_2 + x_3 \\ \alpha_2 = -x_2 - x_3 - x_2x_3 \\ \alpha_3 = x_2x_3, \end{cases}$$

where roots of polinomial (3) $x_1 = 1$, $x_2 \in [-1, 1]$, $x_3 \in [-1, x_2]$.

Parametric form of $Pl_3^{(-1)}$:

$$\begin{cases} \alpha_1 = -1 + x_2 + x_3 \\ \alpha_2 = x_2 + x_3 - x_2x_3 \\ \alpha_3 = -x_2x_3, \end{cases}$$

where roots of polinomial (3) $x_1 = -1$, $x_2 \in [-1, 1]$, $x_3 \in [-1, x_2]$.

Parametric form of $Surf_3$:

$$\begin{cases} \alpha_1 = 2t + x_3 \\ \alpha_2 = -t^2 - 2tx_3 \\ \alpha_3 = t^2x_3, \end{cases} \tag{4}$$

where roots of polinomial (3) $x_1 = x_2 = t$, $t \in [-1, 1]$, $x_3 \in [-1, 1]$.

The boundary area $S(\partial\mathcal{E}_3)$ can be computed as a sum of surface areas:

$$S(\partial\mathcal{E}_3) = S(Pl_3^{(1)}) + S(Pl_3^{(-1)}) + S(Surf_3) \approx \sqrt{3}\times\frac{4}{3} + \sqrt{3}\times\frac{4}{3} + 6.759 \approx 11.378. \tag{5}$$

4. COMPUTATION OF $S(\partial\mathcal{E}_n)$

Consider polinom of degree n in the form (1).

The boundary $\partial\mathcal{E}_n$ of domain $\mathcal{E}_n$ consists of hypersurface $Surf_n$ and two hyperplanes sections $Pl_n^{(1)}$ and $Pl_n^{(-1)}$, that correspond to roots 1 and -1, i.e., polinom (1) with parameters from section $Pl_n^{(1)}$ has a root, equal to 1, and polinom (1) with parameters from section $Pl_n^{(-1)}$ has a root, equal to -1.

Statements about hypersurface areas of hyperplanes sections $Pl_n^{(1)}$ and $Pl_n^{(-1)}$ as well as of hypersurface $Surf_n$ we formulate as propositions.

Proposition 1. *Hypersurface area of hyperplanes sections $Pl_n^{(1)}$ and $Pl_n^{(-1)}$, $n \geqslant 2$, can be computed as*

$$S(Pl_n^{(1)}) = S(Pl_n^{(-1)}) = \sqrt{n}\times 2^{\frac{n(n-1)}{2}} \prod_{k=1}^{n-1} \frac{\{(k-1)!\}^2}{(2k-1)!}. \tag{6}$$

Proposition 2. *Hypersurface area of hypersurface $Surf_n$, $n \geqslant 3$, can be computed as*

$$S(Surf_n) = 2 \int\limits_{\substack{x_1\in(-1,1)\\ -1\leqslant x_2\leqslant\ldots\leqslant x_{n-1}\leqslant 1}} \cdots \int \frac{\sqrt{1-x_1^{2n}}}{\sqrt{1-x_1^2}} \prod_{1\leqslant i<j\leqslant n-1} |x_i - x_j|\; dx_1 \ldots dx_{n-1}. \tag{7}$$

In particular, for $n = 3$

$$S(Surf_3) = 2 \iint\limits_{\substack{-1<x_1<1\\ -1\leqslant x_2\leqslant 1}} \frac{\sqrt{1-x_1^6}}{\sqrt{1-x_1^2}}\, |x_1 - x_2|\; dx_1 dx_2 \approx 6.759.$$

The boundary area $S(\partial\mathcal{E}_n)$ can be computed as a sum of hypersurface areas:

$$S(\partial\mathcal{E}_n) = S(Pl_n^{(1)}) + S(Pl_n^{(-1)}) + S(Surf_n). \tag{8}$$

Next proposition presents the main result of the paper.

Proposition 3. *Hypersurface area $S(\partial\mathcal{E}_n)$ is infinitesimal with $n\to\infty$. The maximum value is reached for $n = 3$.*

5. CONCLUSION

The paper presents the results of computation of hypersurface area $S(\partial\mathcal{E}_n)$ of the boundary of the domain $\mathcal{E}_n$ in n-dimensional space, that is a subset of Schur stability domain D_n.

As a topic for further research we suggest the computation of hypersurface area $S(\partial D_n)$ of the boundary of the domain D_n. As already mentioned, boundary ∂D_n consists of two hyperplanes sections $Pl_n^{(1)}$ and $Pl_n^{(-1)}$, that correspond to roots 1 and -1, and one hypersurface, that is denoted as $\mathfrak{S}_n$. Computation of hypersurface area $S(\mathfrak{S}_n)$ for arbitrary n seems to be a more difficult task, comparing with the result (7) in Proposition 2.

APPENDIX

A.1. PROOF OF PROPOSITION 1

With standard methods one can check, that (6) holds for $n = 2, 3$. On the basis of (6) one can obtain

$$S(Pl_2^{(1)}) = 2\sqrt{2},$$
$$S(Pl_3^{(1)}) = \sqrt{3}\,\frac{4}{3},$$

that agrees with the values, computed in Sections 2 and 3.

Now we will prove (6) for arbitrary n.

The hyperplane section $Pl_n^{(1)}$ can be parameterized:

$$\begin{cases} \alpha_1 = 1 + \zeta_1 \\ \alpha_2 = -(\zeta_1 + \zeta_2) \\ \alpha_3 = \zeta_2 + \zeta_3 \\ \ldots \\ \alpha_k = (-1)^{k-1}(\zeta_{k-1} + \zeta_k) \\ \ldots \\ \alpha_{n-2} = (-1)^{n-3}(\zeta_{n-3} + \zeta_{n-2}) \\ \alpha_{n-1} = (-1)^{n-2}(\zeta_{n-2} + \zeta_{n-1}) \\ \alpha_n = (-1)^{n-1}\zeta_{n-1}, \end{cases} \tag{A.1}$$

where $\zeta_i,\ i = 1, \ldots, n-1$ – elementary symmetric polinomial of degree i of $(n-1)$ variables $x_2, \ldots, x_n$, $-1 \leqslant x_2 \leqslant \ldots \leqslant x_n \leqslant 1$.

Parametric form (A.1) is derived using Vieta's formulas and taking $x_1 = 1$. Ordering of parameters $x_2, \ldots, x_n$ is caused by the fact, that parametric form (A.1) is symmetric with respect to the parameters permutations.

By definition

$$S(Pl_n^{(1)}) = \underset{-1\leqslant x_2\leqslant\ldots\leqslant x_n\leqslant 1}{\int\cdots\int} \|H_n\| dx_2 \ldots dx_n, \tag{A.2}$$

where H_n – determinant:

$$H_n = \begin{vmatrix} \mathrm{j}_1 & \dfrac{\partial\alpha_1}{\partial x_2} & \dfrac{\partial\alpha_1}{\partial x_3} & \cdots & \dfrac{\partial\alpha_1}{\partial x_n} \\ \mathrm{j}_2 & \dfrac{\partial\alpha_2}{\partial x_2} & \dfrac{\partial\alpha_2}{\partial x_3} & \cdots & \dfrac{\partial\alpha_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathrm{j}_n & \dfrac{\partial\alpha_n}{\partial x_2} & \dfrac{\partial\alpha_n}{\partial x_3} & \cdots & \dfrac{\partial\alpha_n}{\partial x_n} \end{vmatrix}, \tag{A.3}$$

where $\{\mathrm{j}_1, \ldots, \mathrm{j}_n\}$ – orthonormal basis in R^n.

For ζ_i from (A.1) we have

$$\frac{\partial\zeta_i}{\partial x_r} = \zeta_{i-1}^{(r)},$$

where $\zeta_j^{(r)}$– elementary symmetric polinomial of degree j of $n-2$ variables $x_2, \ldots, x_{r-1}, x_{r+1}, \ldots, x_n$.

Without loss of generality let $n = 2k+1$. Then on the basis of (A.3) and (A.1) one can obtain

$$H_n = \begin{vmatrix} \mathfrak{j}_1 & 1 & 1 & \cdots & 1 \\ \mathfrak{j}_2 & -1-\zeta_1^{(2)} & -1-\zeta_1^{(3)} & \cdots & -1-\zeta_1^{(n)} \\ \mathfrak{j}_3 & \zeta_1^{(2)}+\zeta_2^{(2)} & \zeta_1^{(3)}+\zeta_2^{(3)} & \cdots & \zeta_1^{(n)}+\zeta_2^{(n)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{j}_{n-1} & -\zeta_{n-3}^{(2)}-\zeta_{n-2}^{(2)} & -\zeta_{n-3}^{(3)}-\zeta_{n-2}^{(3)} & \cdots & -\zeta_{n-3}^{(n)}-\zeta_{n-2}^{(n)} \\ \mathfrak{j}_n & \zeta_{n-2}^{(2)} & \zeta_{n-2}^{(3)} & \cdots & \zeta_{n-2}^{(n)} \end{vmatrix}. \tag{A.4}$$

Consider $H_n^{(n)}$ – cofactor of $\mathfrak{j}_n$ in expansion of H_n along the first column:

$$H_n^{(n)} = \begin{vmatrix} 1 & 1 & \cdots & 1 \\ -1-\zeta_1^{(2)} & -1-\zeta_1^{(3)} & \cdots & -1-\zeta_1^{(n)} \\ \zeta_1^{(2)}+\zeta_2^{(2)} & \zeta_1^{(3)}+\zeta_2^{(3)} & \cdots & \zeta_1^{(n)}+\zeta_2^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ -\zeta_{n-3}^{(2)}-\zeta_{n-2}^{(2)} & -\zeta_{n-3}^{(3)}-\zeta_{n-2}^{(3)} & \cdots & -\zeta_{n-3}^{(n)}-\zeta_{n-2}^{(n)} \end{vmatrix}.$$

With standard transformations of rows $H_n^{(n)}$ can be transformed to the form

$$H_n^{(n)} = \begin{vmatrix} 1 & 1 & \cdots & 1 \\ -\zeta_1^{(2)} & -\zeta_1^{(3)} & \cdots & -\zeta_1^{(n)} \\ \zeta_2^{(2)} & \zeta_2^{(3)} & \cdots & \zeta_2^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ -\zeta_{n-2}^{(2)} & -\zeta_{n-2}^{(3)} & \cdots & -\zeta_{n-2}^{(n)} \end{vmatrix}$$

While considering hyperplane section $Pl_n^{(1)}$ with $\sum\limits_{i=1}^{n} \alpha_i = 1$ in (A.1), we have $\sum\limits_{i=1}^{n} \frac{\partial \alpha_i}{\partial x_j} = 0$ for all $j = 2, \ldots, n$.

Thus, all cofactors in expansion of H_n along the first column are equal in absolute values, hence it is sufficient to evaluate one of them, for instance $H_n^{(n)}$.

Then on the basis of (A.2) and [3] one can derive

$$S(Pl_n^{(1)}) = \sqrt{n} \underset{-1\leqslant x_2 \leqslant \ldots \leqslant x_n \leqslant 1}{\int \cdots \int} |H_n^{(n)}| dx_2 \ldots dx_n = \sqrt{n}\, V(\mathcal{E}_{n-1}), \tag{A.5}$$

where $V(\mathcal{E}_{n-1})$ – volume of domain $\mathcal{E}_{n-1}$.

To complete the proof for $Pl_n^{(1)}$ it remains to take into account the result from [3], that

$$V(\mathcal{E}_n) = 2^{\frac{n(n+1)}{2}} \prod_{k=1}^{n} \frac{\{(k-1)!\}^2}{(2k-1)!}.$$

The proof for $Pl_n^{(-1)}$ is obtained in the same way.

The proof of Proposition 1 is finished.

A.2. PROOF OF PROPOSITION 2

We will prove the proposition by induction.

Let $n = 3$. Surface $Surf_3$ is parameterized in (4). One can calculate its area $S(Surf_3)$ in the following way:

$$S(Surf_3) = \iint\limits_{\substack{-1\leqslant t\leqslant 1\\ -1\leqslant x_3\leqslant 1}} \sqrt{\left(\frac{D(\alpha_1,\alpha_2)}{D(t,x_3)}\right)^2 + \left(\frac{D(\alpha_1,\alpha_3)}{D(t,x_3)}\right)^2 + \left(\frac{D(\alpha_2,\alpha_3)}{D(t,x_3)}\right)^2}\, dt\, dx_3,$$

where

$$\frac{D(\alpha_1,\alpha_2)}{D(t,x_3)} = \begin{vmatrix} \dfrac{\partial\alpha_1}{\partial t} & \dfrac{\partial\alpha_1}{\partial x_3} \\ \dfrac{\partial\alpha_2}{\partial t} & \dfrac{\partial\alpha_2}{\partial x_3} \end{vmatrix} \simeq 2(t-x_3), \quad \frac{D(\alpha_1,\alpha_3)}{D(t,x_3)} = \begin{vmatrix} \dfrac{\partial\alpha_1}{\partial t} & \dfrac{\partial\alpha_1}{\partial x_3} \\ \dfrac{\partial\alpha_3}{\partial t} & \dfrac{\partial\alpha_3}{\partial x_3} \end{vmatrix} \simeq 2t(t-x_3),$$

$$\frac{D(\alpha_2,\alpha_3)}{D(t,x_3)} = \begin{vmatrix} \dfrac{\partial\alpha_2}{\partial t} & \dfrac{\partial\alpha_2}{\partial x_3} \\ \dfrac{\partial\alpha_3}{\partial t} & \dfrac{\partial\alpha_3}{\partial x_3} \end{vmatrix} \simeq 2t^2(t-x_3).$$

The sign "$\simeq$" means being equal in absolute values. Its usage is convenient and valid, because corresponding values will be squared at the right moment.

Thus,

$$S(Surf_3) = \iint\limits_{\substack{-1\leqslant t\leqslant 1\\ -1\leqslant x_3\leqslant 1}} \|A_3\|\, dt\, dx_3, \tag{A.6}$$

where

$$\|A_3\| = 2\sqrt{1+t^2+t^4}\ |t-x_3|. \tag{A.7}$$

The induction base is (A.6) for $S(Surf_3)$.

Now consider hypersurface $Surf_n$ for arbitrary n, that is parameterized as:

$$\begin{cases} \alpha_1 = 2t + \sigma_1 \\ \alpha_2 = -(t^2 + 2t\sigma_1 + \sigma_2) \\ \alpha_3 = t^2\sigma_1 + 2t\sigma_2 + \sigma_3 \\ \dots \\ \alpha_k = (-1)^{k-1}(t^2\sigma_{k-2} + 2t\sigma_{k-1} + \sigma_k) \\ \dots \\ \alpha_{n-2} = (-1)^{n-3}(t^2\sigma_{n-4} + 2t\sigma_{n-3} + \sigma_{n-2}) \\ \alpha_{n-1} = (-1)^{n-2}(t^2\sigma_{n-3} + 2t\sigma_{n-2}) \\ \alpha_n = (-1)^{n-1}t^2\sigma_{n-2}, \end{cases} \tag{A.8}$$

where σ_i, $i = 1,\dots,n-2$, – elementary symmetric polinomial of degree i of $(n-2)$ variables $x_3,\dots,x_n$, $t \in [-1,1]$, $-1 \leqslant x_3 \leqslant \dots \leqslant x_n \leqslant 1$.

Parametric form (A.8) is derived on the basis of Vieta's formulas after equating $x_1 = x_2 = t$. Ordering of parameters $x_3,\dots,x_n$ is caused by the fact, that parametric form (A.8) is symmetric with respect to the parameters permutations.

The induction assumption is that the proposition is true for hypersurface area $S(Surf_n)$. I.e. we assume, that

$$S(Surf_n) = \underset{\substack{t\in[-1,1] \\ -1\leqslant x_3\leqslant\ldots\leqslant x_n\leqslant 1}}{\int\cdots\int} \|A_n\| dt\, dx_3 \ldots dx_n, \tag{A.9}$$

where

$$\|A_n\| = 2\sqrt{1+t^2+\ldots+t^{2(n-1)}}\left(\prod_{3\leqslant i\leqslant n}|t-x_i|\right)\left(\prod_{3\leqslant i<j\leqslant n}|x_i-x_j|\right). \tag{A.10}$$

One can note, that formulas (A.9) and (7) are equivalent, but for current purposes (A.9) is more convenient.

Now on the basis of (A.10) we will prove the proposition for hypersurface area $S(Surf_{n+1})$.

Let $y = x_{n+1}$. Then hypersurface $Surf_{n+1}$ is parameterized in the following way:

$$\begin{cases} \alpha_1 = 2t + \sigma_1 + y \\ \alpha_2 = -(t^2 + 2t(\sigma_1 + y) + \sigma_2 + y\sigma_1) \\ \alpha_3 = t^2(\sigma_1 + y) + 2t(\sigma_2 + y\sigma_1) + \sigma_3 + y\sigma_2 \\ \alpha_4 = -(t^2(\sigma_2 + y\sigma_1) + 2t(\sigma_3 + y\sigma_2) + \sigma_4 + y\sigma_3) \\ \ldots \\ \alpha_k = (-1)^{k-1}(t^2(\sigma_{k-2} + y\sigma_{k-3}) + 2t(\sigma_{k-1} + y\sigma_{k-2}) + \sigma_k + y\sigma_{k-1}) \\ \ldots \\ \alpha_{n-1} = (-1)^{n-2}(t^2(\sigma_{n-3} + y\sigma_{n-4}) + 2t(\sigma_{n-2} + y\sigma_{n-3}) + y\sigma_{n-2}) \\ \alpha_n = (-1)^{n-1}(t^2(\sigma_{n-2} + y\sigma_{n-3}) + 2ty\sigma_{n-2}) \\ \alpha_{n+1} = (-1)^n t^2 y\sigma_{n-2}, \end{cases} \tag{A.11}$$

where σ_i, $i = 1, \ldots, n-2$, were defined in (A.8), $t \in [-1, 1]$, $-1 \leqslant x_3 \leqslant \ldots \leqslant x_n \leqslant y \leqslant 1$.

Parametric form (A.11) is derived on the basis of Vieta's formulas after equating $x_1 = x_2 = t$. Ordering of parameters $x_3, \ldots, x_n, y$ is caused by the fact, that parametric form (A.11) is symmetric with respect to the parameters permutations.

By definition

$$S(Surf_{n+1}) = \underset{\substack{t\in[-1,1] \\ -1\leqslant x_3\leqslant\ldots\leqslant x_n\leqslant y\leqslant 1}}{\int\cdots\int} \|A_{n+1}\| dt\, dx_3 \ldots dx_n\, dy, \tag{A.12}$$

where A_{n+1} – determinant:

$$A_{n+1} = \begin{vmatrix} \mathrm{j}_1 & \dfrac{\partial\alpha_1}{\partial t} & \dfrac{\partial\alpha_1}{\partial x_3} & \cdots & \dfrac{\partial\alpha_1}{\partial x_n} & \dfrac{\partial\alpha_1}{\partial y} \\ \mathrm{j}_2 & \dfrac{\partial\alpha_2}{\partial t} & \dfrac{\partial\alpha_2}{\partial x_3} & \cdots & \dfrac{\partial\alpha_2}{\partial x_n} & \dfrac{\partial\alpha_2}{\partial y} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{j}_n & \dfrac{\partial\alpha_n}{\partial t} & \dfrac{\partial\alpha_n}{\partial x_3} & \cdots & \dfrac{\partial\alpha_n}{\partial x_n} & \dfrac{\partial\alpha_n}{\partial y} \\ \mathrm{j}_{n+1} & \dfrac{\partial\alpha_{n+1}}{\partial t} & \dfrac{\partial\alpha_{n+1}}{\partial x_3} & \cdots & \dfrac{\partial\alpha_{n+1}}{\partial x_n} & \dfrac{\partial\alpha_{n+1}}{\partial y} \end{vmatrix}, \tag{A.13}$$

$\{\mathrm{j}_1, \ldots, \mathrm{j}_{n+1}\}$ – orthonormal basis in R^{n+1}.

Consider A_{n+1} as a vector function of y:

$$A_{n+1} = G(y) = \sum_{j=1}^{n+1} \mathrm{j}_j \times g_j(y) = (g_1(y), \ldots, g_{n+1}(y)). \tag{A.14}$$

Taking into account (A.11) one can see, that $g_j(y)$, $j = 1 \ldots n+1$, are polinomials of y of degree at most $n-1$.

We argue, that $G(y) = 0$ for $y = t$ and $y = x_i$, $i = 3, \ldots, n$.

From (A.11) one can see, that for $k = 1, \ldots, n+1$

$$\alpha_k = (-1)^{k-1}(t^2(\sigma_{k-2} + y\sigma_{k-3}) + 2t(\sigma_{k-1} + y\sigma_{k-2}) + \sigma_k + y\sigma_{k-1}),$$

where σ_i were defined earlier for $i = 1, \ldots, n-2$, $\sigma_0 = 1$, $\sigma_m = 0$ for $m < 0$ and $m > n-2$.

Then

$$\frac{\partial \alpha_k}{\partial y} = (-1)^{k-1}(t^2\sigma_{k-3} + 2t\sigma_{k-2} + \sigma_{k-1}),$$

$$\frac{\partial \alpha_k}{\partial t} = (-1)^{k-1}2(t(\sigma_{k-2} + y\sigma_{k-3}) + \sigma_{k-1} + y\sigma_{k-2}),$$

$$\frac{\partial \alpha_k}{\partial x_r} = (-1)^{k-1}\left(t^2\left(\frac{\partial \sigma_{k-2}}{\partial x_r} + y\frac{\partial \sigma_{k-3}}{\partial x_r}\right) + 2t\left(\frac{\partial \sigma_{k-1}}{\partial x_r} + y\frac{\partial \sigma_{k-2}}{\partial x_r}\right) + \frac{\partial \sigma_k}{\partial x_r} + y\frac{\partial \sigma_{k-1}}{\partial x_r}\right),$$

where $r = n-3, \ldots, n$.

For $\frac{\partial \sigma_i}{\partial x_r}$ holds equality

$$\frac{\partial \sigma_i}{\partial x_r} = \sigma_{i-1}^{(r)},$$

where $\sigma_j^{(r)}$– elementary symmetric polinomial of degree j of $n-3$ variables $x_3, \ldots, x_{r-1}, x_{r+1}, \ldots, x_n$.

Then

$$\frac{\partial \alpha_k}{\partial x_r} = (-1)^{k-1}\left(t^2(\sigma_{k-3}^{(r)} + y\sigma_{k-4}^{(r)}) + 2t(\sigma_{k-2}^{(r)} + y\sigma_{k-3}^{(r)}) + \sigma_{k-1}^{(r)} + y\sigma_{k-2}^{(r)}\right). \tag{A.15}$$

For σ_i and $\sigma_i^{(r)}$ holds equality

$$\sigma_i = \sigma_i^{(r)} + x_r\sigma_{i-1}^{(r)}. \tag{A.16}$$

Then on the basis of (A.15) and (A.16) one can see, that $\forall k = 1, \ldots, n+1$ holds equality

$$\left.\frac{\partial \alpha_k}{\partial x_r}\right|_{y=x_r} = (-1)^{k-1}(t^2\sigma_{k-3} + 2t\sigma_{k-2} + \sigma_{k-1}) = \frac{\partial \alpha_k}{\partial y}.$$

It means, that for $y = x_r$ column r is equal to column $n+1$ in determinant A_{n+1} (A.13), $r = 3, \ldots, n$.

Moreover,

$$\left.\frac{\partial \alpha_k}{\partial t}\right|_{y=t} = (-1)^{k-1}(t^2\sigma_{k-3} + 2t\sigma_{k-2} + \sigma_{k-1}) = \frac{\partial \alpha_k}{\partial y}.$$

It means, that for $y = t$ column 2 is equal to column $n+1$ in determinant A_{n+1} (A.13).

Thus,

$$G(y) = K\ (t-y) \prod_{3\leqslant i\leqslant n} (x_i - y), \tag{A.17}$$

where K – vector in R^{n+1}.

Then

$$A_{n+1}\big|_{y=0} = G(0) = Ktx_3\cdots x_n, \tag{A.18}$$

where $A_{n+1}\big|_{y=0}$ – determinant A_{n+1} in (A.13) with $y = 0$.

Determinant $A_{n+1}\big|_{y=0}$ can be expanded along the last row, taking into account, that only the first and the last elements in that row will remain non-zero. Then

$$A_{n+1}\big|_{y=0} = \jmath_{n+1} B_n + A_n \left.\frac{\partial \alpha_{n+1}}{\partial y}\right|_{y=0}$$

and

$$\|\, A_{n+1}\big|_{y=0}\| = \sqrt{(B_n)^2 + \|A_n\|^2 (t^2 x_3\cdots x_n)^2}, \tag{A.19}$$

where $\|A_n\|$ was defined in (A.10), B_n – cofactor of $\jmath_{n+1}$ in expansion of $A_{n+1}\big|_{y=0}$.

Taking into account (A.11),

$$B_n \simeq$$

$$\begin{vmatrix} 2 & 1 & \cdots & 1 & 1 \\ 2t+2\sigma_1 & 2t+\sigma_1^{(3)} & \cdots & 2t+\sigma_1^{(n)} & 2t+\sigma_1 \\ 2t\sigma_1+2\sigma_2 & t^2+2t\sigma_1^{(3)}+\sigma_2^{(3)} & \cdots & t^2+2t\sigma_1^{(n)}+\sigma_2^{(n)} & t^2+2t\sigma_1+\sigma_2 \\ 2t\sigma_2+2\sigma_3 & t^2\sigma_1^{(3)}+2t\sigma_2^{(3)}+\sigma_3^{(3)} & \cdots & t^2\sigma_1^{(n)}+2t\sigma_2^{(n)}+\sigma_3^{(n)} & t^2\sigma_1+2t\sigma_2+\sigma_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 2t\sigma_{n-4}+2\sigma_{n-3} & t^2\sigma_{n-5}^{(3)}+2t\sigma_{n-4}^{(3)}+\sigma_{n-3}^{(3)} & \cdots & t^2\sigma_{n-5}^{(n)}+2t\sigma_{n-4}^{(n)}+\sigma_{n-3}^{(n)} & t^2\sigma_{n-5}+2t\sigma_{n-4}+\sigma_{n-3} \\ 2t\sigma_{n-3}+2\sigma_{n-2} & t^2\sigma_{n-4}^{(3)}+2t\sigma_{n-3}^{(3)} & \cdots & t^2\sigma_{n-4}^{(n)}+2t\sigma_{n-3}^{(n)} & t^2\sigma_{n-4}+2t\sigma_{n-3}+\sigma_{n-2} \\ 2t\sigma_{n-2} & t^2\sigma_{n-3}^{(3)} & \cdots & t^2\sigma_{n-3}^{(n)} & t^2\sigma_{n-3}+2t\sigma_{n-2} \end{vmatrix}. \tag{A.20}$$

Lemma 1.

$$B_n \simeq 2tx_3\cdots x_n \left(\prod_{3\leqslant i\leqslant n} |t-x_i|\right)\left(\prod_{3\leqslant i<j\leqslant n} |x_i - x_j|\right). \tag{A.21}$$

The proof of Lemma 1 is presented in the end of the proof of the proposition.

Now using (A.19), (A.10) and (A.21) in Lemma 1, one can derive

$$\|\, A_{n+1}\big|_{y=0}\| = 2|tx_3\cdots x_n| \left(\prod_{3\leqslant i\leqslant n} |t-x_i|\right)\left(\prod_{3\leqslant i<j\leqslant n} |x_i - x_j|\right)\sqrt{1+t^2(1+t^2+\ldots+t^{2(n-1)})}$$

and on the basis of (A.18) we have

$$\|K\| = \frac{\|\, A_{n+1}\big|_{y=0}\|}{|tx_3\cdots x_n|}. \tag{A.22}$$

Bringing back the notation $x_{n+1} = y$, on the basis of (A.17) and (A.22) one can obtain

$$\|G(x_{n+1})\| = 2\left(\prod_{3\leqslant i\leqslant n+1} |t - x_i|\right)\left(\prod_{3\leqslant i<j\leqslant n+1} |x_i - x_j|\right)\sqrt{1 + t^2 + \ldots + t^{2n}}. \tag{A.23}$$

And on the basis of (A.12), (A.14) and (A.23) one can obtain required induction step

$$S(Surf_{n+1}) \tag{A.24}$$

$$= 2 \underset{\substack{t\in[-1,1] \\ -1\leqslant x_3\leqslant\ldots\leqslant x_n\leqslant x_{n+1}\leqslant 1}}{\int\cdots\int} \sqrt{1+t^2+\ldots+t^{2n}}\left(\prod_{3\leqslant i\leqslant n+1} |t-x_i|\right)\left(\prod_{3\leqslant i<j\leqslant n+1} |x_i-x_j|\right) dt\, dx_3 \ldots dx_n\, dx_{n+1}.$$

A.2.1. Proof of Lemma 1

With standard computation methods one can check, that lemma is true for $n = 3, 4$.

$$B_3 \simeq \begin{vmatrix} 2 & 1 & 1 \\ 2t + 2x_3 & 2t & 2t + x_3 \\ 2tx_3 & t^2 & t^2 + 2tx_3 \end{vmatrix} \simeq 2tx_3(t - x_3),$$

$$B_4 \simeq \begin{vmatrix} 2 & 1 & 1 & 1 \\ 2t + 2(x_3 + x_4) & 2t + x_4 & 2t + x_3 & 2t + x_3 + x_4 \\ 2t(x_3 + x_4) + 2x_3x_4 & t^2 + 2tx_4 & t^2 + 2tx_3 & t^2 + 2t(x_3 + x_4) + x_3x_4 \\ 2tx_3x_4 & t^2x_4 & t^2x_3 & t^2(x_3 + x_4) + 2tx_3x_4 \end{vmatrix}$$
$$\simeq 2tx_3x_4(t - x_3)(t - x_4)(x_3 - x_4).$$

Now we will prove lemma for arbitrary n.

Meanwhile, one can find compact form of the idea of the proof in "II. Solution by R.J. Walker" [7], where authors prove similar statement.

First step.

On the basis of (A.16) make substitutions in (A.20) in rows $2, \ldots, n$ in columns $2, \ldots, n-1$. Obtain

$$B_n \simeq 2*$$
$$\begin{vmatrix} 1 & 1 & \cdots & 1 & 1 \\ t + \sigma_1 & 2t + \sigma_1 - x_3 & \cdots & 2t + \sigma_1 - x_n & 2t + \sigma_1 \\ t\sigma_1 + \sigma_2 & t^2 + 2t(\sigma_1 - x_3) + \sigma_2 - x_3\sigma_1^{(3)} & \cdots & t^2 + 2t(\sigma_1 - x_n) + \sigma_2 - x_n\sigma_1^{(n)} & t^2 + 2t\sigma_1 + \sigma_2 \\ t\sigma_2 + \sigma_3 & t^2(\sigma_1 - x_3) + 2t(\sigma_2 - x_3\sigma_1^{(3)}) + \sigma_3 - x_3\sigma_2^{(3)} & \cdots & t^2(\sigma_1 - x_n) + 2t(\sigma_2 - x_n\sigma_1^{(n)}) + \sigma_3 - x_n\sigma_2^{(n)} & t^2\sigma_1 + 2t\sigma_2 + \sigma_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t\sigma_{n-3} + \sigma_{n-2} & t^2(\sigma_{n-4} - x_3\sigma_{n-5}^{(3)}) + 2t(\sigma_{n-3} - x_3\sigma_{n-4}^{(3)}) & \cdots & t^2(\sigma_{n-4} - x_n\sigma_{n-5}^{(n)}) + 2t(\sigma_{n-3} - x_n\sigma_{n-4}^{(n)}) & t^2\sigma_{n-4} + 2t\sigma_{n-3} + \sigma_{n-2} \\ t\sigma_{n-2} & t^2(\sigma_{n-3} - x_3\sigma_{n-4}^{(3)}) & \cdots & t^2(\sigma_{n-3} - x_n\sigma_{n-4}^{(n)}) & t^2\sigma_{n-3} + 2t\sigma_{n-2} \end{vmatrix}.$$

From rows $j = 2, \ldots, n-1$ subtract the first row, multiplied by the last element of jth row $b_{j,n}$. From row n subtract the first row, multiplied by $t^2\sigma_{n-3}$. Multiply rows $j = 2, \ldots, n$ by -1.

Obtain

$$B_n \simeq 2*$$
$$\begin{vmatrix}
1 & 1 & \cdots & 1 & 1 \\
t & x_3 & \cdots & x_n & 0 \\
t^2 + t\sigma_1 & 2tx_3 + x_3\sigma_1^{(3)} & \cdots & 2tx_n + x_n\sigma_1^{(n)} & 0 \\
t^2\sigma_1 + t\sigma_2 & t^2x_3 + 2tx_3\sigma_1^{(3)} + x_3\sigma_2^{(3)} & \cdots & t^2x_n + 2tx_n\sigma_1^{(n)} + x_n\sigma_2^{(n)} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t^2\sigma_{n-5} + t\sigma_{n-4} & t^2x_3\sigma_{n-6}^{(3)} + 2tx_3\sigma_{n-5}^{(3)} + x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-6}^{(n)} + 2tx_n\sigma_{n-5}^{(n)} + x_n\sigma_{n-4}^{(n)} & 0 \\
t^2\sigma_{n-4} + t\sigma_{n-3} & t^2x_3\sigma_{n-5}^{(3)} + 2tx_3\sigma_{n-4}^{(3)} + \sigma_{n-2} & \cdots & t^2x_n\sigma_{n-5}^{(n)} + 2tx_n\sigma_{n-4}^{(n)} + \sigma_{n-2} & 0 \\
t^2\sigma_{n-3} - t\sigma_{n-2} & t^2x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-4}^{(n)} & -2t\sigma_{n-2}
\end{vmatrix}.$$

In row $n-1$ in columns $j = 2, \ldots, n-1$ apply identity $\sigma_{n-2} = x_{j+1}\sigma_{n-3}^{(j+1)}$. Obtain

$$B_n \simeq 2*$$
$$\begin{vmatrix}
1 & 1 & \cdots & 1 & 1 \\
t & x_3 & \cdots & x_n & 0 \\
t^2 + t\sigma_1 & 2tx_3 + x_3\sigma_1^{(3)} & \cdots & 2tx_n + x_n\sigma_1^{(n)} & 0 \\
t^2\sigma_1 + t\sigma_2 & t^2x_3 + 2tx_3\sigma_1^{(3)} + x_3\sigma_2^{(3)} & \cdots & t^2x_n + 2tx_n\sigma_1^{(n)} + x_n\sigma_2^{(n)} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t^2\sigma_{n-5} + t\sigma_{n-4} & t^2x_3\sigma_{n-6}^{(3)} + 2tx_3\sigma_{n-5}^{(3)} + x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-6}^{(n)} + 2tx_n\sigma_{n-5}^{(n)} + x_n\sigma_{n-4}^{(n)} & 0 \\
t^2\sigma_{n-4} + t\sigma_{n-3} & t^2x_3\sigma_{n-5}^{(3)} + 2tx_3\sigma_{n-4}^{(3)} + x_3\sigma_{n-3}^{(3)} & \cdots & t^2x_n\sigma_{n-5}^{(n)} + 2tx_n\sigma_{n-4}^{(n)} + x_n\sigma_{n-3}^{(n)} & 0 \\
t^2\sigma_{n-3} - t\sigma_{n-2} & t^2x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-4}^{(n)} & -2t\sigma_{n-2}
\end{vmatrix}.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j = 2, \ldots, n-1$ respectively. Obtain

$$B_n \simeq 2tx_3 \cdots x_n*$$
$$\begin{vmatrix}
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 1 \\
1 & 1 & \cdots & 1 & 0 \\
t + \sigma_1 & 2t + \sigma_1^{(3)} & \cdots & 2t + \sigma_1^{(n)} & 0 \\
t\sigma_1 + \sigma_2 & t^2 + 2t\sigma_1^{(3)} + \sigma_2^{(3)} & \cdots & t^2 + 2t\sigma_1^{(n)} + \sigma_2^{(n)} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t\sigma_{n-5} + \sigma_{n-4} & t^2\sigma_{n-6}^{(3)} + 2t\sigma_{n-5}^{(3)} + \sigma_{n-4}^{(3)} & \cdots & t^2\sigma_{n-6}^{(n)} + 2t\sigma_{n-5}^{(n)} + \sigma_{n-4}^{(n)} & 0 \\
t\sigma_{n-4} + \sigma_{n-3} & t^2\sigma_{n-5}^{(3)} + 2t\sigma_{n-4}^{(3)} + \sigma_{n-3}^{(3)} & \cdots & t^2\sigma_{n-5}^{(n)} + 2t\sigma_{n-4}^{(n)} + \sigma_{n-3}^{(n)} & 0 \\
t\sigma_{n-3} - \sigma_{n-2} & t^2\sigma_{n-4}^{(3)} & \cdots & t^2\sigma_{n-4}^{(n)} & (-1)^1 2t\sigma_{n-2}
\end{vmatrix}.$$

Second step.

On the basis of (A.16) make substitutions in rows $3,\ldots,n$ and columns $2,\ldots,n-1$. Obtain

$$B_n \simeq 2tx_3\cdots x_n *$$
$$\begin{vmatrix}
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 1\\
1 & 1 & \cdots & 1 & 0\\
t+\sigma_1 & 2t+\sigma_1-x_3 & \cdots & 2t+\sigma_1-x_n & 0\\
t\sigma_1+\sigma_2 & t^2+2t(\sigma_1-x_3)+\sigma_2-x_3\sigma_1^{(3)} & \cdots & t^2+2t(\sigma_1-x_n)+\sigma_2-x_n\sigma_1^{(n)} & 0\\
\vdots & \vdots & \ddots & \vdots & \vdots\\
t\sigma_{n-5}+\sigma_{n-4} & t^2(\sigma_{n-6}-x_3\sigma_{n-7}^{(3)})+2t(\sigma_{n-5}-x_3\sigma_{n-6}^{(3)})+\sigma_{n-4}-x_3\sigma_{n-5}^{(3)} & \cdots & t^2(\sigma_{n-6}-x_n\sigma_{n-7}^{(n)})+2t(\sigma_{n-5}-x_n\sigma_{n-6}^{(n)})+\sigma_{n-4}-x_n\sigma_{n-5}^{(n)} & 0\\
t\sigma_{n-4}+\sigma_{n-3} & t^2(\sigma_{n-5}-x_3\sigma_{n-6}^{(3)})+2t(\sigma_{n-4}-x_3\sigma_{n-5}^{(3)})+\sigma_{n-3}-x_3\sigma_{n-4}^{(3)} & \cdots & t^2(\sigma_{n-5}-x_n\sigma_{n-6}^{(n)})+2t(\sigma_{n-4}-x_n\sigma_{n-5}^{(n)})+\sigma_{n-3}-x_n\sigma_{n-4}^{(n)} & 0\\
t\sigma_{n-3}-\sigma_{n-2} & t^2(\sigma_{n-4}-x_3\sigma_{n-5}^{(3)}) & \cdots & t^2(\sigma_{n-4}-x_n\sigma_{n-5}^{(n)}) & -2t\sigma_{n-2}
\end{vmatrix}.$$

From row 3 subtract row 2, multiplied by $2t+\sigma_1$. From row 4 subtract row 2, multiplied by $t^2+2t\sigma_1+\sigma_2$. From rows $j=5,\ldots,n-1$ subtract row 2, multiplied by $t^2\sigma_{j-4}+2t\sigma_{j-3}+\sigma_{j-2}$. From row n subtract row 2, multiplied by $t^2\sigma_{n-4}-\sigma_{n-2}$. Multiply rows $j=3,\ldots,n$ by -1. Obtain

$$B_n \simeq 2tx_3\cdots x_n *$$
$$\begin{vmatrix}
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 1\\
1 & 1 & \cdots & 1 & 0\\
t & x_3 & \cdots & x_n & 0\\
t^2+t\sigma_1 & 2tx_3+x_3\sigma_1^{(3)} & \cdots & 2tx_n+x_n\sigma_1^{(n)} & 0\\
\vdots & \vdots & \ddots & \vdots & \vdots\\
t^2\sigma_{n-6}+t\sigma_{n-5} & t^2x_3\sigma_{n-7}^{(3)}+2tx_3\sigma_{n-6}^{(3)}+x_3\sigma_{n-5}^{(3)} & \cdots & t^2x_n\sigma_{n-7}^{(n)}+2tx_n\sigma_{n-6}^{(n)}+x_n\sigma_{n-5}^{(n)} & 0\\
t^2\sigma_{n-5}+t\sigma_{n-4} & t^2x_3\sigma_{n-6}^{(3)}+2tx_3\sigma_{n-5}^{(3)}+x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-6}^{(n)}+2tx_n\sigma_{n-5}^{(n)}+x_n\sigma_{n-4}^{(n)} & 0\\
t^2\sigma_{n-4}-t\sigma_{n-3} & t^2x_3\sigma_{n-5}^{(3)}-\sigma_{n-2} & \cdots & t^2x_n\sigma_{n-5}^{(n)}-\sigma_{n-2} & 2t\sigma_{n-2}
\end{vmatrix}.$$

In row n in columns $j=2,\ldots,n-1$ apply identity $\sigma_{n-2}=x_{j+1}\sigma_{n-3}^{(j+1)}$. Obtain

$$B_n \simeq 2tx_3\cdots x_n *$$
$$\begin{vmatrix}
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 1\\
1 & 1 & \cdots & 1 & 0\\
t & x_3 & \cdots & x_n & 0\\
t^2+t\sigma_1 & 2tx_3+x_3\sigma_1^{(3)} & \cdots & 2tx_n+x_n\sigma_1^{(n)} & 0\\
\vdots & \vdots & \ddots & \vdots & \vdots\\
t^2\sigma_{n-6}+t\sigma_{n-5} & t^2x_3\sigma_{n-7}^{(3)}+2tx_3\sigma_{n-6}^{(3)}+x_3\sigma_{n-5}^{(3)} & \cdots & t^2x_n\sigma_{n-7}^{(n)}+2tx_n\sigma_{n-6}^{(n)}+x_n\sigma_{n-5}^{(n)} & 0\\
t^2\sigma_{n-5}+t\sigma_{n-4} & t^2x_3\sigma_{n-6}^{(3)}+2tx_3\sigma_{n-5}^{(3)}+x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-6}^{(n)}+2tx_n\sigma_{n-5}^{(n)}+x_n\sigma_{n-4}^{(n)} & 0\\
t^2\sigma_{n-4}-t\sigma_{n-3} & t^2x_3\sigma_{n-5}^{(3)}-x_3\sigma_{n-3}^{(3)} & \cdots & t^2x_n\sigma_{n-5}^{(n)}-x_n\sigma_{n-3}^{(n)} & 2t\sigma_{n-2}
\end{vmatrix}.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j = 2, \ldots, n-1$ respectively. Obtain

$$B_n \simeq 2t^2 x_3^2 \cdots x_n^2 *$$
$$\begin{vmatrix} t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 1 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t+\sigma_1 & 2t+\sigma_1^{(3)} & \cdots & 2t+\sigma_1^{(n)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t\sigma_{n-6}+\sigma_{n-5} & t^2\sigma_{n-7}^{(3)}+2t\sigma_{n-6}^{(3)}+\sigma_{n-5}^{(3)} & \cdots & t^2\sigma_{n-7}^{(n)}+2t\sigma_{n-6}^{(n)}+\sigma_{n-5}^{(n)} & 0 \\ t\sigma_{n-5}+\sigma_{n-4} & t^2\sigma_{n-6}^{(3)}+2t\sigma_{n-5}^{(3)}+\sigma_{n-4}^{(3)} & \cdots & t^2\sigma_{n-6}^{(n)}+2t\sigma_{n-5}^{(n)}+\sigma_{n-4}^{(n)} & 0 \\ t\sigma_{n-4}-\sigma_{n-3} & t^2\sigma_{n-5}^{(3)}-\sigma_{n-3}^{(3)} & \cdots & t^2\sigma_{n-5}^{(n)}-\sigma_{n-3}^{(n)} & (-1)^2 2t\sigma_{n-2} \end{vmatrix}.$$

Third step.

On the basis of (A.16) make substitutions in rows $4, \ldots, n$ and columns $2, \ldots, n-1$. Obtain

$$B_n \simeq 2t^2 x_3^2 \cdots x_n^2 *$$
$$\begin{vmatrix} t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 1 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t+\sigma_1 & 2t+\sigma_1-x_3 & \cdots & 2t+\sigma_1-x_n & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t\sigma_{n-6}+\sigma_{n-5} & t^2(\sigma_{n-7}-x_3\sigma_{n-8}^{(3)})+2t(\sigma_{n-6}-x_3\sigma_{n-7}^{(3)})+\sigma_{n-5}-x_3\sigma_{n-6}^{(3)} & \cdots & t^2(\sigma_{n-7}-x_n\sigma_{n-8}^{(n)})+2t(\sigma_{n-6}-x_n\sigma_{n-7}^{(n)})+\sigma_{n-5}-x_n\sigma_{n-6}^{(n)} & 0 \\ t\sigma_{n-5}+\sigma_{n-4} & t^2(\sigma_{n-6}-x_3\sigma_{n-7}^{(3)})+2t(\sigma_{n-5}-x_3\sigma_{n-6}^{(3)})+\sigma_{n-4}-x_3\sigma_{n-5}^{(3)} & \cdots & t^2(\sigma_{n-6}-x_n\sigma_{n-7}^{(n)})+2t(\sigma_{n-5}-x_n\sigma_{n-6}^{(n)})+\sigma_{n-4}-x_n\sigma_{n-5}^{(n)} & 0 \\ t\sigma_{n-4}-\sigma_{n-3} & t^2(\sigma_{n-5}-x_3\sigma_{n-6}^{(3)})-(\sigma_{n-3}-x_3\sigma_{n-4}^{(3)}) & \cdots & t^2(\sigma_{n-5}-x_n\sigma_{n-6}^{(n)})-(\sigma_{n-3}-x_n\sigma_{n-4}^{(n)}) & 2t\sigma_{n-2} \end{vmatrix}.$$

From row 4 subtract row 3, multiplied by $2t+\sigma_1$. From row 5 subtract row 3, multiplied by $t^2+2t\sigma_1+\sigma_2$. From rows $j = 6, \ldots, n-1$ subtract row 3, multiplied by $t^2\sigma_{j-5}+2t\sigma_{j-4}+\sigma_{j-3}$. From row n subtract row 3, multiplied by $t^2\sigma_{n-5}-\sigma_{n-3}$. Multiply rows $j = 4, \ldots, n$ by -1. Obtain

$$B_n \simeq 2t^2 x_3^2 \cdots x_n^2 *$$
$$\begin{vmatrix} t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 1 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t & x_3 & \cdots & x_n & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^2\sigma_{n-7}+t\sigma_{n-6} & t^2x_3\sigma_{n-8}^{(3)}+2tx_3\sigma_{n-7}^{(3)}+x_3\sigma_{n-6}^{(3)} & \cdots & t^2x_n\sigma_{n-8}^{(n)}+2tx_n\sigma_{n-7}^{(n)}+x_n\sigma_{n-6}^{(n)} & 0 \\ t^2\sigma_{n-6}+t\sigma_{n-5} & t^2x_3\sigma_{n-7}^{(3)}+2tx_3\sigma_{n-6}^{(3)}+x_3\sigma_{n-5}^{(3)} & \cdots & t^2x_n\sigma_{n-7}^{(n)}+2tx_n\sigma_{n-6}^{(n)}+x_n\sigma_{n-5}^{(n)} & 0 \\ t^2\sigma_{n-5}-t\sigma_{n-4} & t^2x_3\sigma_{n-6}^{(3)}-x_3\sigma_{n-4}^{(3)} & \cdots & t^2x_n\sigma_{n-6}^{(n)}-x_n\sigma_{n-4}^{(n)} & -2t\sigma_{n-2} \end{vmatrix}.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j = 2, \ldots, n-1$ respectively. Obtain

$$B_n \simeq 2t^3 x_3^3 \cdots x_n^3 *$$
$$\begin{vmatrix} t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 1 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t\sigma_{n-7} + \sigma_{n-6} & t^2\sigma_{n-8}^{(3)} + 2t\sigma_{n-7}^{(3)} + \sigma_{n-6}^{(3)} & \cdots & t^2\sigma_{n-8}^{(n)} + 2t\sigma_{n-7}^{(n)} + \sigma_{n-6}^{(n)} & 0 \\ t\sigma_{n-6} + \sigma_{n-5} & t^2\sigma_{n-7}^{(3)} + 2t\sigma_{n-6}^{(3)} + \sigma_{n-5}^{(3)} & \cdots & t^2\sigma_{n-7}^{(n)} + 2t\sigma_{n-6}^{(n)} + \sigma_{n-5}^{(n)} & 0 \\ t\sigma_{n-5} - \sigma_{n-4} & t^2\sigma_{n-6}^{(3)} - \sigma_{n-4}^{(3)} & \cdots & t^2\sigma_{n-6}^{(n)} - \sigma_{n-4}^{(n)} & (-1)^3 2t\sigma_{n-2} \end{vmatrix}.$$

The third step is finished.

Now perform steps $4, \ldots, n-5$ in the same way, as step 3.

After finishing step $n-5$ obtain

$$B_n \simeq 2t^{n-5} x_3^{n-5} \cdots x_n^{n-5} *$$
$$\begin{vmatrix} t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 1 \\ t^{-n+6} & x_3^{-n+6} & \cdots & x_n^{-n+6} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t + \sigma_1 & 2t + \sigma_1^{(3)} & \cdots & 2t + \sigma_1^{(n)} & 0 \\ t\sigma_1 + \sigma_2 & t^2 + 2t\sigma_1^{(3)} + \sigma_2^{(3)} & \cdots & t^2 + 2t\sigma_1^{(n)} + \sigma_2^{(n)} & 0 \\ t\sigma_2 + \sigma_3 & t^2\sigma_1^{(3)} + 2t\sigma_2^{(3)} + \sigma_3^{(3)} & \cdots & t^2\sigma_1^{(n)} + 2t\sigma_2^{(n)} + \sigma_3^{(n)} & 0 \\ t\sigma_3 - \sigma_4 & t^2\sigma_2^{(3)} - \sigma_4^{(3)} & \cdots & t^2\sigma_2^{(n)} - \sigma_4^{(n)} & (-1)^{n-5} 2t\sigma_{n-2} \end{vmatrix}.$$

$(n-4)$th step.

On the basis of (A.16) make substitutions in rows $(n-3), \ldots, n$ in columns $2, \ldots, n-1$. Obtain

$$B_n \simeq 2t^{n-5} x_3^{n-5} \cdots x_n^{n-5} *$$
$$\begin{vmatrix} t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 1 \\ t^{-n+6} & x_3^{-n+6} & \cdots & x_n^{-n+6} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t + \sigma_1 & 2t + \sigma_1 - x_3 & \cdots & 2t + \sigma_1 - x_n & 0 \\ t\sigma_1 + \sigma_2 & t^2 + 2t(\sigma_1 - x_3) + \sigma_2 - x_3\sigma_1^{(3)} & \cdots & t^2 + 2t(\sigma_1 - x_n) + \sigma_2 - x_n\sigma_1^{(n)} & 0 \\ t\sigma_2 + \sigma_3 & t^2(\sigma_1 - x_3) + 2t(\sigma_2 - x_3\sigma_1^{(3)}) + \sigma_3 - x_3\sigma_2^3 & \cdots & t^2(\sigma_1 - x_n) + 2t(\sigma_2 - x_n\sigma_1^{(n)}) + \sigma_3 - x_n\sigma_2^{(n)} & 0 \\ t\sigma_3 - \sigma_4 & t^2(\sigma_2 - x_3\sigma_1^{(3)}) - (\sigma_4 - x_3\sigma_3^{(3)}) & \cdots & t^2(\sigma_2 - x_n\sigma_1^{(n)}) - (\sigma_4 - x_n\sigma_3^{(n)}) & (-1)^{n-5} 2t\sigma_{n-2} \end{vmatrix}.$$

From row $n-3$ subtract row $n-4$, multiplied by $2t+\sigma_1$. From row $n-2$ subtract row $n-4$, multiplied by $t^2+2t\sigma_1+\sigma_2$. From row $n-1$ subtract row $n-4$, multiplied by $t^2\sigma_1+2t\sigma_2+\sigma_3$. From row n subtract row $n-4$, multiplied by $t^2\sigma_2-\sigma_4$. Multiply rows $j=n-3,\ldots,n$ by -1. Obtain

$$B_n \simeq 2t^{n-5}x_3^{n-5}\cdots x_n^{n-5} *$$
$$\left|\begin{array}{ccccc}
t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 1 \\
t^{-n+6} & x_3^{-n+6} & \cdots & x_n^{-n+6} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\
1 & 1 & \cdots & 1 & 0 \\
t & x_3 & \cdots & x_n & 0 \\
t^2+t\sigma_1 & 2tx_3+x_3\sigma_1^{(3)} & \cdots & 2tx_n+x_n\sigma_1^{(n)} & 0 \\
t^2\sigma_1+t\sigma_2 & t^2x_3+2tx_3\sigma_1^{(3)}+x_3\sigma_2^3 & \cdots & t^2x_n+2tx_n\sigma_1^{(n)}+x_n\sigma_2^{(n)} & 0 \\
t^2\sigma_2-t\sigma_3 & t^2x_3\sigma_1^{(3)}-x_3\sigma_3^{(3)} & \cdots & t^2x_n\sigma_1^{(n)}-x_n\sigma_3^{(n)} & (-1)^{n-4}2t\sigma_{n-2}
\end{array}\right|.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j=2,\ldots,n-1$ respectively. Obtain

$$B_n \simeq 2t^{n-4}x_3^{n-4}\cdots x_n^{n-4} *$$
$$\left|\begin{array}{ccccc}
t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 1 \\
t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\
1 & 1 & \cdots & 1 & 0 \\
t+\sigma_1 & 2t+\sigma_1^{(3)} & \cdots & 2t+\sigma_1^{(n)} & 0 \\
t\sigma_1+\sigma_2 & t^2+2t\sigma_1^{(3)}+\sigma_2^{(3)} & \cdots & t^2+2t\sigma_1^{(n)}+\sigma_2^{(n)} & 0 \\
t\sigma_2-\sigma_3 & t^2\sigma_1^{(3)}-\sigma_3^{(3)} & \cdots & t^2\sigma_1^{(n)}-\sigma_3^{(n)} & (-1)^{n-4}2t\sigma_{n-2}
\end{array}\right|.$$

$(n-3)$th step.

On the basis of (A.16) make substitutions in rows $(n-2),\ldots,n$ in columns $2,\ldots,n-1$. Obtain

$$B_n \simeq 2t^{n-4}x_3^{n-4}\cdots x_n^{n-4} *$$
$$\left|\begin{array}{ccccc}
t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 1 \\
t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\
t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\
1 & 1 & \cdots & 1 & 0 \\
t+\sigma_1 & 2t+\sigma_1-x_3 & \cdots & 2t+\sigma_1-x_n & 0 \\
t\sigma_1+\sigma_2 & t^2+2t(\sigma_1-x_3)+\sigma_2-x_3\sigma_1^{(3)} & \cdots & t^2+2t(\sigma_1-x_n)+\sigma_2-x_n\sigma_1^{(n)} & 0 \\
t\sigma_2-\sigma_3 & t^2(\sigma_1-x_3)-(\sigma_3-x_3\sigma_2^{(3)}) & \cdots & t^2(\sigma_1-x_n)-(\sigma_3-x_n\sigma_2^{(n)}) & (-1)^{n-4}2t\sigma_{n-2}
\end{array}\right|.$$

From row $n-2$ subtract row $n-3$, multiplied by $2t+\sigma_1$. From row $n-1$ subtract row $n-3$, multiplied by $t^2+2t\sigma_1+\sigma_2$. From row n subtract row $n-3$, multiplied by $t^2\sigma_1-\sigma_3$. Multiply rows $j=n-2,n-1,n$ by -1. Obtain

$$B_n \simeq 2t^{n-4}x_3^{n-4}\cdots x_n^{n-4} *$$
$$\begin{vmatrix} t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 1 \\ t^{-n+5} & x_3^{-n+5} & \cdots & x_n^{-n+5} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t & x_3 & \cdots & x_n & 0 \\ t^2+t\sigma_1 & 2tx_3+x_3\sigma_1^{(3)} & \cdots & 2tx_n+x_n\sigma_1^{(n)} & 0 \\ t^2\sigma_1-t\sigma_2 & t^2x_3-x_3\sigma_2^{(3)} & \cdots & t^2x_n-x_n\sigma_2^{(n)} & (-1)^{n-3}2t\sigma_{n-2} \end{vmatrix}.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j=2,\ldots,n-1$ respectively. Obtain

$$B_n \simeq 2t^{n-3}x_3^{n-3}\cdots x_n^{n-3} *$$
$$\begin{vmatrix} t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 1 \\ t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t+\sigma_1 & 2t+\sigma_1^{(3)} & \cdots & 2t+\sigma_1^{(n)} & 0 \\ t\sigma_1-\sigma_2 & t^2-\sigma_2^{(3)} & \cdots & t^2-\sigma_2^{(n)} & (-1)^{n-3}2t\sigma_{n-2} \end{vmatrix}.$$

$(n-2)$th step.

On the basis of (A.16) make substitutions in rows $n-1,n$ in columns $2,\ldots,n-1$. Obtain

$$B_n \simeq 2t^{n-3}x_3^{n-3}\cdots x_n^{n-3} *$$
$$\begin{vmatrix} t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 1 \\ t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t+\sigma_1 & 2t+\sigma_1-x_3 & \cdots & 2t+\sigma_1-x_n & 0 \\ t\sigma_1-\sigma_2 & t^2-(\sigma_2-x_3\sigma_1^{(3)}) & \cdots & t^2-(\sigma_2-x_n\sigma_1^{(n)}) & (-1)^{n-3}2t\sigma_{n-2} \end{vmatrix}.$$

From row $n-1$ subtract row $n-2$, multiplied by $2t+\sigma_1$. From row n subtract row $n-2$, multiplied by $t^2-\sigma_2$. Multiply rows $j=n-1,n$ by -1. Obtain

$$B_n \simeq 2t^{n-3}x_3^{n-3}\cdots x_n^{n-3} *$$
$$\begin{vmatrix} t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 1 \\ t^{-n+4} & x_3^{-n+4} & \cdots & x_n^{-n+4} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t & x_3 & \cdots & x_n & 0 \\ t^2-t\sigma_1 & -x_3\sigma_1^{(3)} & \cdots & -x_n\sigma_1^{(n)} & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}.$$

Take factor t out of the first column, then take factors x_{j+1} out of columns $j=2,\ldots,n-1$ respectively. Obtain

$$B_n \simeq 2t^{n-2}x_3^{n-2}\cdots x_n^{n-2} *$$
$$\begin{vmatrix} t^{-n+2} & x_3^{-n+2} & \cdots & x_n^{-n+2} & 1 \\ t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-4} & x_3^{-4} & \cdots & x_n^{-4} & 0 \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t-\sigma_1 & -\sigma_1^{(3)} & \cdots & -\sigma_1^{(n)} & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}.$$

$(n-1)$th step.

On the basis of (A.16) make substitutions in the last row in columns $2,\ldots,n-1$. Obtain

$$B_n \simeq 2t^{n-2}x_3^{n-2}\cdots x_n^{n-2} *$$
$$\begin{vmatrix} t^{-n+2} & x_3^{-n+2} & \cdots & x_n^{-n+2} & 1 \\ t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-4} & x_3^{-4} & \cdots & x_n^{-4} & 0 \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t-\sigma_1 & x_3-\sigma_1 & \cdots & x_n-\sigma_1 & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}.$$

From row n subtract row $n-1$, multiplied by $-\sigma_1$. Obtain

$$B_n \simeq 2t^{n-2}x_3^{n-2}\cdots x_n^{n-2} *$$
$$\begin{vmatrix} t^{-n+2} & x_3^{-n+2} & \cdots & x_n^{-n+2} & 1 \\ t^{-n+3} & x_3^{-n+3} & \cdots & x_n^{-n+3} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-4} & x_3^{-4} & \cdots & x_n^{-4} & 0 \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & 0 \\ t & x_3 & \cdots & x_n & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}.$$

Take factor t out of the first column, take factors x_{j+1} out of columns $j = 2, \ldots, n-1$ respectively. Obtain

$$B_n \simeq 2t^{n-1}x_3^{n-1}\cdots x_n^{n-1} *$$
$$\begin{vmatrix} t^{-n+1} & x_3^{-n+1} & \cdots & x_n^{-n+1} & 1 \\ t^{-n+2} & x_3^{-n+2} & \cdots & x_n^{-n+2} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{-5} & x_3^{-5} & \cdots & x_n^{-5} & 0 \\ t^{-4} & x_3^{-4} & \cdots & x_n^{-4} & 0 \\ t^{-3} & x_3^{-3} & \cdots & x_n^{-3} & 0 \\ t^{-2} & x_3^{-2} & \cdots & x_n^{-2} & 0 \\ t^{-1} & x_3^{-1} & \cdots & x_n^{-1} & 0 \\ 1 & 1 & \cdots & 1 & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}.$$

Now we have finished step $(n-1)$, after which return factors, which were taken out, to the corresponding columns of determinant, and obtain

$$B_n \simeq 2\begin{vmatrix} 1 & 1 & \cdots & 1 & 1 \\ t & x_3 & \cdots & x_n & 0 \\ t^2 & x_3^2 & \cdots & x_n^2 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ t^{n-2} & x_3^{n-2} & \cdots & x_n^{n-2} & 0 \\ t^{n-1} & x_3^{n-1} & \cdots & x_n^{n-1} & (-1)^{n-2}2t\sigma_{n-2} \end{vmatrix}. \tag{A.25}$$

After expansion of determinant (A.25) along the last column obtain

$$B_n \simeq 2\Big((-1)^{n+1}tx_3\cdots x_nV(t, x_3, \ldots, x_n) + (-1)^{2n}(-1)^{n-2}2tx_3\cdots x_nV(t, x_3, \ldots, x_n)\Big), \tag{A.26}$$

where $V(t, x_3, \ldots, x_n)$ – Vandermonde determinant of variables $t, x_3, \ldots, x_n$.

Taking into account different parities of degrees of -1 in (A.26), obtain

$$B_n \simeq 2tx_3\cdots x_nV(t, x_3, \ldots, x_n).$$

Proofs of Lemma 1 and Proposition 2 are finished.

A.3. PROOF OF PROPOSITION 3

In [2] it is shown, that

$$\log_2 V(D_n) = -\left(\frac{n}{2}+\frac{1}{4}\right)\log_2 n + O(n).$$

Hence,

$$\ln V(D_n) = -\left(\frac{n}{2}+\frac{1}{4}\right)\ln n + O(n).$$

In [3] it is shown, that

$$\ln\left(\frac{V(\mathcal{E}_n)}{V(D_n)}\right) = -\frac{\ln 2}{2}n^2 + \frac{1}{8}\ln n + O(1)$$

and

$$V(\mathcal{E}_n) = \underset{-1\leqslant x_1\leqslant\ldots\leqslant x_n\leqslant 1}{\int\cdots\int}\left(\prod_{1\leqslant i<j\leqslant n}|x_i - x_j|\right)dx_1\ldots dx_n = 2^{\frac{n(n+1)}{2}}\prod_{k=1}^{n}\frac{\{(k-1)!\}^2}{(2k-1)!}. \tag{A.27}$$

Thus,

$$\ln V(\mathcal{E}_n) = -\frac{\ln 2}{2}n^2 - \left(\frac{n}{2}+\frac{1}{8}\right)\ln n + O(n). \tag{A.28}$$

On the basis of (A.5) from the proof of proposition 1 we have

$$S(Pl_n^{(1)}) = S(Pl_n^{(-1)}) = \sqrt{n}\,V(\mathcal{E}_{n-1}).$$

On the basis of (7) in Proposition 2 and (A.27) one can see, that

$$S(Surf_n) = 2\underset{\substack{x_1\in[-1,1]\\ -1\leqslant x_2\leqslant\ldots\leqslant x_{n-1}\leqslant 1}}{\int\cdots\int}\sqrt{1+x_1^2+\ldots+x_1^{2(n-1)}}\left(\prod_{1\leqslant i<j\leqslant n-1}|x_i - x_j|\right)dx_1\ldots dx_{n-1}$$

$$\leqslant 2\sqrt{n}\underset{\substack{x_1\in[-1,1]\\ -1\leqslant x_2\leqslant\ldots\leqslant x_{n-1}\leqslant 1}}{\int\cdots\int}\left(\prod_{1\leqslant i<j\leqslant n-1}|x_i - x_j|\right)dx_1\ldots dx_{n-1} = 2\sqrt{n}(n-1)V(\mathcal{E}_{n-1}).$$

Thus, on the basis of (8) one can obtain

$$S(\partial\mathcal{E}_n) \leqslant 2\sqrt{n}\,nV(\mathcal{E}_{n-1}). \tag{A.29}$$

On the basis of (A.28) and (A.29) one can obtain

$$\ln S(\partial\mathcal{E}_n) \leqslant -\frac{\ln 2}{2}n^2 - \frac{n}{2}\ln n + O(n).$$

For large enough n the value of $(-\frac{\ln 2}{2}n^2 - \frac{n}{2}\ln n)$ dominates over $O(n)$, therefore $\lim\limits_{n\to\infty}\ln S(\partial\mathcal{E}_n) = -\infty$, and therefore $\lim\limits_{n\to\infty} S(\partial\mathcal{E}_n) = 0$.

This finishes the proof of the first part of Proposition 3.

Now we will prove the second part of Proposition 3, that maximum value of $S(\partial\mathcal{E}_n)$ is reached for $n = 3$.

From (A.27) follows, that

$$V(\mathcal{E}_n) = V(\mathcal{E}_{n-1})\, 2^n \frac{\{(n-1)!\}^2}{(2n-1)!}, \qquad n \geqslant 2,$$
$$V(\mathcal{E}_1) = 2.$$

Then, taking into account (A.29), we obtain, that

$$S(\partial\mathcal{E}_n) \leqslant F(n), \qquad n \geqslant 2, \tag{A.30}$$

where

$$F(1) = 2,$$
$$F(n) = F(n-1) \times k(n), \qquad n \geqslant 2,$$
$$k(n) = 2^{n-1} \frac{n\sqrt{n}}{(n-1)\sqrt{n-1}} \frac{\{(n-2)!\}^2}{(2n-3)!}.$$

Now we will show, that $k(n) < 1$ for $n \geqslant 4$.

$$k(n) = 2^{n-1} \frac{n\sqrt{n}}{(n-1)\sqrt{n-1}} \frac{1 \times 2 \cdots (n-2)}{(n-1) \times n \cdots (2n-3)}$$
$$= \frac{2\sqrt{n}}{(n-1)\sqrt{n-1}} \times \frac{2 \times 1}{n-1} \times \frac{2 \times 2}{n+1} \times \frac{2 \times 3}{n+2} \cdots \frac{2 \times (n-2)}{2n-3}. \tag{A.31}$$

In (A.31) there are $(n-1)$ multipliers, and all of them are less than 1 for $n \geqslant n_0 = 4$.

It means, that $k(n) < 1$ and $F(n)$ in (A.30) is a decreasing function for $n \geqslant 4$. Therefore, for $n \geqslant 4$

$$S(\partial\mathcal{E}_n) \leqslant F(n) \leqslant F(4) = 2\sqrt{4} \times 4 \times V(\mathcal{E}_3) \approx 5.689.$$

Taking into account (5), the proof of the second part of Proposition 3 is finished.

REFERENCES

1. Dzhafarov, V., Bykkrolu, T., and Akyar, H., Stability Region for Discrete Time Systems and Its Boundary, *Trudy Instituta Matematiki i Mekhaniki UrO RAN*, 2021, vol. 27, no. 3, pp. 246–255.
2. Fam, A.T., The volume of the coefficient space stability domain of monic polynomials, *IEEE International Symposium on Circuits and Systems*, 1989, pp. 1780–1783.
3. Akiyama, S. and Petho, A., On the distribution of polynomials with bounded roots, I. Polynomials with real coefficients, *J. Math. Soc. Japan*, 2014, vol. 66, no. 3, pp. 927–949.
4. Fam, A.T. and Meditch, J.S., A canonical parameter space for linear systems design, *IEEE Transact. Autom. Control*, 1978, vol. 23, no. 3, pp. 454–458.
5. Zellner, A., *An Introduction to Bayesian Inference in Econometrics*, New York: John Wiley & Sons, 1996.
6. Farebrother, R.W., Simplified Samuelson Conditions for Cubic and Quartic Equations, *Manchester School Econom. Soc. Studies*, 1973, vol. 41, no. 4, pp. 396–406.
7. Wedderburn, J.H.M., Smiley, M.F., and Walker, R.J., Jacobian, Alternant, *Amer. Math. Monthly*, 1942, vol. 49, no. 10, pp. 694–696. https://www.jstor.org/stable/2302597

This paper was recommended for publication by A.I. Malikov, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 2, pp. 134–152.

STOCHASTIC SYSTEMS

Problems in the Theory of H^2/H_∞ Controllers for Linear Stochastic Multiplicative-Type Plants

M. E. Shaikin

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
e-mail: shaykin@ipu.ru

Received March 3, 2024
Revised September 29, 2024
Accepted October 2, 2024

Abstract—This paper considers H^2/H_∞ control problems for dynamic plants described by linear Itô stochastic equations with the drift and diffusion coefficients linearly dependent on the state vector, control input, and an exogenous disturbance. The controlled plant has two outputs, namely, the regulated z and the observed (noisy) y ones. The controller is optimized by the quadratic H^2 criterion under the boundedness condition for the induced norm of the operator $H_{z\upsilon}$ relating the exogenous disturbance υ to the regulated output $z : \|H_{z\upsilon}\|_\infty < \gamma$. The conditional H^2/H_∞ optimization problem is solved using differential game theory.

Keywords: H^2/H_∞ control theory, Itô diffusion equation, multiplicative stochastic system, induced operator norm, regulated output, output-feedback controller

DOI: 10.31857/S0005117925020037

1. INTRODUCTION

In this paper, we present some results on the theory of H^2/H_∞ controllers for time-varying plants described by Itô stochastic equations whose drift and diffusion coefficients depend linearly on the state x, control u, and exogenous disturbance υ vectors. These are Itô equations of the form

$$\begin{gathered} dx(t) = \varphi(t)dt + \Phi(t)dW(t), \quad \varphi = Ax + B_1u + B_2\upsilon, \\ \Phi\, dW = A_0xdw_0 + B_{01}udw_1 + B_{02}\upsilon dw_2. \end{gathered} \tag{1.1}$$

Such a specific structure of the drift and diffusion coefficients gives a reason to call equation (1.1) *multiplicative* in each of the three variables mentioned. By assumption, the stochastic processes $w_i(t)$, $i = 0, 1, 2$, are scalar, which is actually not a restriction. The dependence of the drift $\varphi(t)$ and diffusion $\Phi(t)dW(t)$ on the same disturbance $\upsilon(t)$ is not a restriction as well. Finally, the processes $w_i(t)$ are supposed to be statistically dependent, not necessarily with a unit intensity matrix.

The plant has two outputs, i.e., the regulated z and observed y ones:

$$\begin{cases} z(t) = C_1x(t) + D_{11}u(t) + D_{12}\upsilon(t) \\ y(t) = C_2x(t) + D_{21}u(t) + D_{22}\upsilon(t), \quad t \in [0, T]. \end{cases} \tag{1.2}$$

In the nonstationary case, the matrix coefficients depend on the time parameter t. We consider the cases of finite ($T < \infty$) and infinite ($T = \infty$) horizons $[0, T]$. Let x_0 denote the initial condition of equation (1.1).

In control theory, the presence of regulated and observed outputs is natural for practical problem statements. Already at the initial stage of optimization theory, it was necessary to distinguish between *LQR* problems (the design of linear *quadratic* regulators based on the full output z) and *LQG* problems (the design of linear quadratic *Gaussian* regulators based on the information contained in the partially observed output y). However, a well-known fact is that in the latter case with disturbances, it is not always possible to ensure the required *robustness* of the regulator. Generally speaking, the robustness property is one of the fundamental requirements for controller design in the modern H^2/H_∞ control theory and especially in the theory of the so-called *uncertain* systems. Note that in the presence of exogenous disturbances and endogenous perturbations, inherent to an uncertain plant, robustness is the most important issue, not only in controller design but also in *filtering*. However, filtering problems are not considered below, like control optimization and filtering problems for *discrete systems*. Here, it seems appropriate to draw the reader's attention to the similarity of many results of the deterministic control theory with some results of the stochastic H^2/H_∞ theory, especially those concerning plants with a state-feedback controller in Section 2. The main topic of this paper is the stochastic theory of robust control of systems (1.1)–(1.2).

The design of an H^2/H_∞ controller is a conditional (constrained) quadratic optimization problem under an upper bound $\gamma > 0$ imposed on the operator norm $\|H_{zv}\|_\infty$ of the operator H_{zv} with the domain and codomain defined by the functional spaces of all plant's input disturbances $v(\cdot)$ and regulated outputs $z(\cdot)$, respectively. The controller must stabilize the closed-loop system and ensure the boundedness condition $\|H_{zv}\|_\infty < \gamma$ for the induced norm. In the general H^2/H_∞ control theory, the description of such a class of controllers constitutes a subbranch designated by the theory of H_∞ controllers.

In the theory of time-invariant systems, it is conventional to consider a class of controllers of the form $u(t) = \mathcal{K}(t, x(\cdot)|_0^t)$, $t > 0$, where the function $\mathcal{K}$ is Borel measurable and Lipschitz continuous in the second argument. If $u(t) = Kx(t)$ and the matrix $A + B_1K$ is *stable* (in the Hurwitz sense), then one operates a transfer function relating v to z,

$$M(s) = (C_1 + D_1K)(sI - (A + B_1K))^{-1}B_2,$$

letting $\|H_{zv}\|_\infty = \sup_{\omega\in R} \sigma(M(j\omega)I)$, where $\sigma(M(s))$ is the largest singular number of the matrix $M(s)$. The number $\|H_{zv}\|_\infty$ defined in this way is also called the Hardy norm of the transfer function. The bound $\|H_{zv}\|_\infty < \gamma$ reflects the peculiarity of the conditional (suboptimal) H^2/H_∞ control theory in comparison with the theory of unconditional *LQR* optimization by the energy, in this case, optimality criterion.

In control theory, the concept of H_∞ control was pioneered by G. Zames; see his paper [1] published in 1981. The main results of the H^2/H_∞ control theory, first established for linear lime-invariant systems, e.g., [2, 3], were further generalized to time-varying ones and formulated in state-space terms [4, 5]. The most fruitful generalization of the H^2/H_∞ control theory was obtained within *differential game theory* [6]; subsequently, it became possible to investigate, from a uniform standpoint, time-varying deterministic and stochastic [7] systems defined on a finite horizon and systems with non-zero initial conditions for the state vector. Moreover, it became possible to study infinite-dimensional dynamic control systems and even some types of nonlinear systems. The detailed bibliography on these generalizations of the H^2/H_∞ control theory was provided in the monograph [8]; see the introduction to Section 3.2.

The transition from the theory of time-invariant systems to that of time-varying ones naturally required generalizing many concepts from the frequency-domain language to state-space terms. For example, as it turned out in several cases, the usual definition of a stable time-invariant system is convenient to be replaced by the concept of an exponentially stable system, an input-output stable system, or a system stable in some other suitable sense. Using any of these definitions, one can generalize the concepts of stabilizable, observable, etc., time-invariant system to the nonstationary case. For example, in the nonstationary case, a system $\dot{x} = A(t)x + B(t)u$ is said to be stabilizable if there exists a bounded matrix function $K(t)$ such that the system $\dot{x} = (A(t) + B(t)K(t))x$ is exponentially stable. Here is another example: given $\dot{x} = A(t)x + B(t)u$ and $y = C(t)x + D(t)v$, the pair $(A(t), C(t))$ is said to be detectable if there exists a bounded matrix function $L(t)$ such that the system $\dot{x} = (A(t) - L(t)C(t))x$ is exponentially stable. Note that in H_∞ control theory, the need for generalization also dictated the appearance of the concept of the induced H_∞ norm of an operator instead of the norm of the matrix transfer function $H_{zv}(s), Re\, s > 0$.

For a class of nondynamic causal controllers based on the system state vector $x(\cdot)$, the H^2/H_∞ control problem in the multiplicative case was solved in [9]. In other words, the controller was sought in the form $u(t) = K(t, x(\cdot)|_0^t)$, $t \geqslant 0$ and, more narrowly, in the form $u(t) = K(t)x(t)$. The controlled plant in [9] was supposed to be stochastic with the diffusion coefficient ΦdW, multiplicative by the state, control, and exogenous disturbance vectors; the case of a purely deterministic plant (with zero diffusion) was not excluded from the analysis. However, the case of simple linear diffusion of the form $(\Phi\, dW)(t) = B(t)dW(t)$ was not addressed therein. In this paper, the plant remains generally multiplicative, but the linear diffusion case is also considered. The assumptions $D_{12} = 0$, $D'_{11}C_1 = 0$, and $D'_{11}D_{11} = I$ were accepted regarding the matrix coefficients of the formula $z = C_1x + D_{11}u + D_{12}v$; in this paper, we relax these restrictions and designate D_{11} by D_1 and D_{22} by D_2 in (1.2).

There is an interesting connection between the stochastic theory of finite-dimensional multiplicative systems and the deterministic theory of matrix Lie algebras. The applications of Lie groups to ordinary differential equations are well known; for example, see [10]. But the mathematical apparatus of Lie group theory can also be used to find the fundamental matrices of stochastic multiplicative equations and integral representations for the solutions of such equations [11]. In this case, the fundamental matrix $\Phi(t, \tau)$ is a random function, and the general solution of the equation perturbed by the disturbance $h(t)$ is given by

$$x(t) = \Phi(t, t_0)x_0 + \int\limits_{t_0}^{t} \Phi(t, \tau) \circ dh(\tau).$$

In this formula, $\circ dh$ denotes the Fisk–Stratonovich differential; for example, see [12].

Now we provide a small list of publications thematically close to the problem of analyzing the systems of multiplicative type under consideration. The numerical approximation of the solution of the stochastic equation

$$dx_t = (Ax_t + f(x_t))dt + \sum_{i=1}^{n}(B_ix_t + g_i(x_t))dw_i, \quad x(0) = x_0 \in R^dm$$

with nonlinear functions $f, g_i : R^d \to R^d$ was considered in [13]; by assumption, the matrices $A, B_i \in R^{d\times d}$ take values in a matrix Lie algebra $\mathfrak{g}$ with commutator relations $[A, B] = 0$, $[B_i, B_j] = 0$ for all i, j. The analysis of numerical algorithms for finding the so-called exponential integrators [14] is an active area of research in both multiplicative equations and equations with additive noise [15, 16]. The mean-square stability of numerical methods for calculating exponential integrators was investigated in [17]. Group-theoretic methods are effective for numerical integration of stochastic partial differential equations as well [18]. Among the works by Russian researchers, we note the study of an infinite-dimensional stochastic multiplicative equation with operators A and B acting in a separable Hilbert space [19]. By assumption, the operator A induces here a semigroup of operators $S(t)$, $t > 0$, of the class C_0, which guarantees the well-posedness of the Cauchy problem for the unperturbed equation $\dot{X}_t = AX(t)$.

The remainder of this paper is organized as follows. Sections 2 and 3 summarize some known results of the H^2/H_∞ control theory with state-feedback and output-feedback controllers, respectively. Section 4 is devoted to the theory of an output-feedback controller for linear time-varying stochastic systems with linear Gaussian diffusion. This is the $LEQG$ problem of risk-sensitive control, which generalizes the usual LQG problem. Section 5 presents results concerning the control of stochastic time-varying systems with a state-feedback controller; Section 6, information on the theory of multiplicative systems with a dynamic output-feedback controller. In Section 7, we describe some elements of the theory of robust stochastic systems. Concluding remarks are given in Section 8.

2. ELEMENTS OF THE H^2/H_∞ CONTROL THEORY FOR PLANTS WITH STATE-FEEDBACK CONTROLLERS

The standard problem of the deterministic H^2/H_∞ control theory of both time-invariant and time-varying control systems is formulated as follows: for a given $\gamma > 0$, find a conditionally optimal controller in the class of all admissible controllers. A controller is said to be admissible if the closed-loop system with this controller is stable and satisfies the condition $\|H_{zv}\|_\infty < \gamma$. In the simple case of a time-invariant system without control,

$$\dot{x} = Ax + Bv, \quad z = Cx, \quad x(0) = 0,$$

the admissibility requirement takes the following form [20]: the matrix A is stable and the norm of the transfer function $M(s) = C(sI - A)^{-1}B$ is bounded: $\|M(s)\|_\infty < \gamma$. According to the Kalman–Yakubovich–Popov (KYP) lemma, also known as the Bounded Real (BR) lemma in the literature [21], the admissibility condition is equivalent to each of the following two statements: (i) There exists a matrix $\tilde{P} \succ 0$ such that $A'\tilde{P} + \tilde{P}A + \tilde{P}BB'\tilde{P} + C'C \prec 0$. (ii) The Riccati equation $A'P + PA + PBB'P + C'C = 0$ has a stabilizing solution $P \succeq 0$ (i.e., the matrix $A + BB'P$ is stable).

If conditions (i) and (ii) are equivalent, then $P \prec \tilde{P}$ and, therefor, $0 \preceq P \prec \tilde{P}$. Thus, checking the controller's admissibility is reduced to checking the existence (and some properties) of the solutions of Riccati inequalities and/or equations. In the case of linear time-varying systems, the algebraic Riccati equation makes room for a differential equation, i.e., the time derivative $\dot{P}$ is added to the left-hand side of the equation for P in (ii). Consider the *game-theoretic* formulation of the conditionally optimal H^2/H_∞ control problem in the class of admissible controllers. It is based on the following observation: for the closed-loop system with an initial condition $x(0) = 0$, the constraint $\|H_{zv}\|_\infty < \gamma$ is valid iff there exists a number $\varepsilon > 0$ such that, for all $v(\cdot) \in L_2[0, \infty)$,

$$J_\gamma(u, v) := \int_0^\infty (\gamma^2\|v(t)\|^2 - \|z(t)\|^2)dt \geqslant \gamma^2\varepsilon \int_0^\infty \|v(t)\|^2 dt. \tag{2.1}$$

In the dynamic game, the first player (the controller designer) seeks to minimize the loss $J_\gamma(u, v)$ by choosing the best control $u^*(\cdot)$, whereas the second player seeks to maximize it by choosing the least favorable (worst-case) exogenous disturbance $v^*(\cdot)$. In the case $u(t) = K(t, x(\cdot)|_0^t)$, the optimal value of the payoff functional $J_\gamma(u, v)$ at the saddle point is written as $\inf_u \sup_{v \in L_2[0,\infty)} J_\gamma(u, v)$. These are the basic concepts and notation of H_∞ control theory with state-feedback controllers for deterministic (time-invariant and time-varying) systems.

The algebraic Riccati equation for P and the differential equation with the derivative $\dot{P}$ on its left-hand side will be called associated with each other. The differential equation is interesting in the sense of solutions $P(\cdot)$ for which every matrix $P(t)$ is nonnegative definite, $P(t) \succeq 0$, $t \geqslant 0$, and those for which the matrix function $t \mapsto A(t) - BB'P(t)$ is exponentially stable. It is often useful to pass to new variables in the plant's equation and assign both associated Riccati equations, algebraic and differential, to this particular state equation in the new variables. As an illustration, we give a simple example of a deterministic control system of the form

$$\dot{x} = Ax + B_1 u + B_2 \upsilon, \quad z = C_1 x + D_1 u, \quad x(0) = 0. \tag{2.2}$$

Let $G := D_1' D_1 \succ 0$; then $D_1' z = D_1' C_1 x + D_1' D_1 u$, which implies $u = \bar{u} - G^{-1} D_1' C_1 x$, where $\bar{u} := G^{-1} D_1' z$ is the new control vector. Replacing $u(\cdot)$ with $\bar{u}(\cdot)$, we write system (2.2) as

$$\dot{x} = \tilde{A}x + B_1 \bar{u} + B_2 \upsilon, \quad z = \tilde{C}_1 x + D_1 \bar{u}, \tag{2.3}$$

where $\tilde{A} = A - B_1 G^{-1} D_1' C_1$ and $\tilde{C}_1 = (I - D_1 G^{-1} D_1') C_1$. The matrix K solves the original *minimax* controller problem iff $\tilde{K} := K + G^{-1} D_1' C_1$ solves the game-theoretic problem for (2.3) with the vector $\bar{u}(\cdot)$. The pair (A, B_1) is stabilizable iff the pair $(\tilde{A}, B_1)$ is such. The main result of the application of differential game theory to the design of an H_∞ state-feedback controller for a linear time-invariant plant is formulated below (Lemma 1). Consider the Riccati differential equation on an infinite horizon ($t \in [0, \infty)$) with the condition $X(T) = M$ for some matrix $M \succeq 0$ and some $T < \infty$:

$$\begin{gathered} \dot{X} + (A - B_1 G^{-1} D_1' C_1)' X + X(A - B_1 G^{-1} D_1' C_1) \\ + C_1' (I - D_1 G^{-1} D_1') C_1 + X(B_1 G^{-1} B_1' - \gamma^{-2} B_2 B_2') X = 0. \end{gathered} \tag{2.4}$$

We denote by $X_T(t)$ the solution of equation (2.4) corresponding to $M = 0$.

Lemma 1. *Let the pair $(\tilde{A}, \tilde{C}_1)$ be detectable. Then:*

(*i*) *For each fixed t, $X_T(t)$ is nondecreasing in T.*

(*ii*) *If there exists a solution $X \succeq 0$ of the algebraic equation with which equation* (2.4) *is associated, then there exists also a minimal such solution, denoted below by X^+. In addition, $X^+ \succeq X_T(t)$ for all $T \geqslant 0$. Moreover, if the pair $(\tilde{A}, \tilde{C}_1)$ is observable, then each solution $X \succeq 0$ of the algebraic equation is positive definite, $X \succ 0$.*

(*iii*) *If the solution $X^+ \succeq 0$ exists, then the controller*

$$u^*(t) = K^* x(t), \quad K^* = -G^{-1}(B_1' X + D_1' C_1) \tag{2.5}$$

ensures the equality

$$\sup_{\upsilon \in L_2[0,\infty)} J_\gamma(K^* x, \upsilon) = x_0' X^+ x_0. \tag{2.6}$$

For details, we refer to [22].

3. OUTPUT-FEEDBACK CONTROLLERS WITH THE DEPENDENCE ON EXOGENOUS DISTURBANCE

Now we present the results of [21, 23] concerning the design of an H^2/H_∞-controller based on the output $y(\cdot)$ for a plant described by

$$\dot{x} = Ax + B_1 u + B_2 \upsilon, \quad z = C_1 x + D_1 u, \quad y = C_2 x + D_2 \upsilon. \tag{3.1}$$

Here, υ is an exogenous disturbance and $D_2 \neq 0$. Consider a dynamic controller with the state vector x_c of the form

$$\dot{x}_c = A_c x_c + B_c y, \quad u = C_c x_c + D_c y. \tag{3.2}$$

This controller generates an augmented system with the state vector $\bar{x} := (x',\, x_c')'$. If the plant and the dynamic controller are time-invariant systems, then the augmented system is such as well, and its transfer function $\tilde{H}_{zv}(s)$ relating v to z has the form

$$\begin{gathered}\tilde{H}_{zv}(s) = \begin{pmatrix} C_1 + D_1 D_c C_2 & D_1 C_c \end{pmatrix} \\ \times \left(sI - \begin{pmatrix} A + B_1 D_c C_2 & B_1 C_c \\ B_c C_2 & A_c \end{pmatrix} \right)^{-1} \begin{pmatrix} B_2 + B_1 D_c D_2 \\ B_c D_2 \end{pmatrix} + D_1 D_c D_2. \end{gathered} \tag{3.3}$$

It generalizes the formula for the transfer function

$$H_{zv}^K(s) = (C_1 + D_1 K)(sI - A - B_1 K)^{-1} B_2 \tag{3.4}$$

of the closed-loop system, i.e., the plant (3.1) with the nondynamic controller $u(t) = K(t)x(t)$. The below presentation will also involve the concept of an injective mapping L of the output $y(\cdot)$, in some way dual to that of the mapping K defining the controller $u = Kx$. First, we restrict the analysis to the condition $D_c = 0$. Consider the system without control

$$\dot{x} = Ax + B_2 v + Ly, \quad z = C_1 x, \quad y = C_2 x + D_2 v. \tag{3.5}$$

The mapping L generates the transfer function

$$H_{zv}^L(s) = C_1 (sI - A - LC_2)^{-1} (B_2 + LD_2)$$

of system (3.5). As mentioned in the Introduction, the mapping L is closely related to the definition of a detectable system in the nonstationary case. The introduced concepts are sufficient to formulate solvability conditions for the design problem of an output-feedback H_∞ controller.

Lemma 2. *Assume that for the plant* (3.1) *with $D_c = 0$, there exists a controller of the form* (3.2) *such that the state matrix* $\begin{bmatrix} A & B_1 C_c \\ B_c C_2 & A_c \end{bmatrix}$ *of the augmented system is stable and the Hardy norm of the transfer function $\tilde{H}_{zv}(s)$ in* (3.3) *is bounded: $\|\tilde{H}_{zv}(s)\|_\infty < 1$. Then:*

1) There exist a matrix K of the state-feedback controller $u = Kx$ and a matrix $X \succ 0$ satisfying the Riccati equation (2.4); *in addition, the matrix $A + B_1 K$ is stable and the Hardy norm of the transfer function $H_{zv}^K(s)$ in* (3.4) *is bounded, i.e., $\|H_{zv}^K(s)\|_\infty < 1$.*

2) There exist a matrix L of the injective mapping of the output and a matrix $Y \succ 0$ such that

$$(A + LC_2)'Y + Y(A + LC_2) + YC_1'C_1 Y + (B_2 + LD_2)(B_2 + LD_2)' \prec 0;$$

in addition, the matrix $A + LC_2$ is stable and the Hardy norm of the transfer function $H_{zv}^L(s)$ is bounded, i.e., $\|H_{zv}^L(s)\|_\infty < 1$. The matrix Y satisfies the Riccati equation

$$\begin{gathered}(A - B_2 D_2' \Gamma^{-1} C_2)Y + Y(A - B_2 D_2' \Gamma^{-1} C_2)' + B_2(I - D_2' \Gamma^{-1} D_2) B_2' \\ -Y(C_2' \Gamma^{-1} C_2 - \gamma^{-2} C_1' C_1)Y = 0, \quad \Gamma := D_2 D_2' \succ 0. \end{gathered} \tag{3.6}$$

Note that this equation is obtained, based on the duality principle, from (2.4) by replacing the coefficients A', B_1', C_1', and D_1 with A, C_2, B_2, and D_2', respectively and, hence, replacing $G = D_1' D_1$ with $\Gamma = D_2 D_2'$. Under statements 1) and 2) of Lemma 2 and additional conditions (the detectability of the pairs $(A - B_1 G^{-1} D_1' C_1, (I - D_1 G^{-1} D_1')C_1)$, and (A, C_2) and the stabilizability of the pairs (A, B_1) and $(A - B_2 D_2' \Gamma^{-1} C_2, B_2(I - D_2' \Gamma^{-1} D_2)))$, it can be asserted [23, 24] that if the Riccati equations for X (2.4) and for Y (3.6) admit minimal solutions $X^+ \succeq 0$ and $Y^+ \succeq 0$, respectively, and $\rho(Y^+ X^+) < \gamma^2$, then the dynamic game with the system of equations (3.1) and the payoff functional $J_\gamma(\cdot,\cdot)$ (2.1) has the finite value $\inf_u \sup_{\upsilon\in L_2} J_\gamma(Kx,\upsilon)$. And the optimal minimizing controller is given by the equations [22]

$$\dot{\hat{x}} = A\hat{x} + B_1u + B_2\hat{\upsilon} + (I-\gamma^{-2}Y^+X^+)^{-1}(Y^+C_2' + B_2D_2')\Gamma^{-1}(y-\hat{y}),$$
$$u^* = -G^{-1}(B_1'X^+ + D_1'C_1)\hat{x},$$

where $\hat{\upsilon} = \gamma^{-2}B_2'X^+\hat{x}$ and $\hat{y} = C_2\hat{x} + D_2\hat{\upsilon}$.

Remark. The results provided by Lemma 2 are valid for $D_c = 0$. If $D_c \neq 0$, then block (11) in the state matrix A_{cl} is replaced by $A + B_1D_cC_2$, see (3.3); for $H_{z\upsilon}$ we have $H_{z\upsilon} = C_{cl}(sI - A_{cl})^{-1}B_{cl} + D_{cl}$, where $B_{cl} = \begin{bmatrix} B_2 + B_1D_cD_2 \\ B_cD_2 \end{bmatrix}$, $C_{cl} = [C_1 + D_1D_cC_2 \;\; D_1C_c]$, and $D_{cl} = D_1D_cD_2$. Interestingly, for $D_c \neq 0$, the optimal minimizing controller coincides with the above one in the case $D_c = 0$, see [21, 23].

4. OUTPUT-FEEDBACK H_∞ CONTROLLERS FOR TIME-VARYING PLANTS WITH GAUSSIAN DIFFUSION

Systems with Gaussian (linear) diffusion are the simplest in the class of linear stochastic systems. Let

$$dx(t) = (A(t)x(t) + B_1(t)u(t))dt + B_2(t)dW(t), \quad x(0) = x_0,$$
$$dy(t) = C_2(t)x(t)dt + D_2(t)dW(t), \quad y(0) = y_0, \quad 0 \leqslant t \leqslant T$$

($(x(0), y(0))$ and $W(t)$ are often supposed to be independent). If $(x(0), y(0))$ is a Gaussian vector, then $(x(t), y(t))$ is a Gaussian process. The theory of the controller based on the output $y(\cdot)$ will be constructed starting from its interpretation as the minimax theory of output-feedback *LQG* control [25]. In this theory [26, 27], the exponent of a quadratic functional is taken as an optimality criterion:

$$J_T(u) = 2\tau \log \mathsf{E} \exp \frac{1}{2\tau}\left[x'(T)Mx(T) + \int\limits_0^T F(x(t), u(t))dt\right],$$

where $F(x,u)$ is a quadratic form on the space of vector pairs (x,u). As usual, the problem is to find $\inf_{u(\cdot)} J_T(u(\cdot))$. This is a risk-sensitive control problem, designated as the *LEQG* problem; in the limit as $\tau \to \infty$, one obtains the standard *LQG* problem of risk-neutral control. For arbitrary τ, the controller has the form $u(t) = K\hat{x}(t)$, where $\hat{x}$ is the state vector of the filter estimating $x(t)$ from the measured output $y(\cdot)$.

We draw the reader's attention to the analogy between the results of the previous and current sections. However, note that Section 3 has presented mainly the results concerning the theory of time-invariant systems defined on an infinite horizon $0 \leqslant t < \infty$, whereas those below are related to time-invariant systems defined on $0 \leqslant t < T$, $T < \infty$. For this reason, the algebraic Riccati equations of Section 3 should be replaced by the Riccati differential equations in this section when comparing the results. Of course, a complete theory of systems of both types, deterministic and stochastic Gaussian, covers stationary and nonstationary plants.

The control problem in this section is solved under the following assumptions. First, the matrix of the quadratic form $F(x,u)$, written as $\begin{pmatrix} R(t) & \Upsilon(t) \\ \Upsilon(t)' & G(t) \end{pmatrix}$, satisfies the condition $R - \Upsilon G^{-1}\Upsilon' \succ 0$. If we define the regulated output $z = C_1x + D_1u$ and let $F(x,u) = z'z$ to clarify the analogy with the previous section results, then $R - \Upsilon G^{-1}\Upsilon' = C_1'C_1 - C_1'D_1G^{-1}D_1'C_1 \succ 0$. Suppose also that $M \succeq 0$ and $\tau = \gamma^2$. Second, we accept conditions (i)–(iii) below (see Lemma 3).

Lemma 3. *Assume that:*

(i) *The Riccati differential equation associated with the algebraic equation* (3.6), *with the initial condition* $Y(0) = Y_0$, *has a solution* $Y = Y'$ *such that* $Y(t) \succeq c_0 I$ *for some* $c_0 > 0$ *and all* $t \in [0, T]$.

(ii) *The Riccati differential equation associated with the algebraic equation* (2.4), *with the initial condition* $X(T) = M$, *has a solution* X *such that* $X(t) = X'(t) \succeq 0$ *for all* $t \in [0, T]$.

(iii) *For each* $t \in [0, T]$, $\rho(Y(t)X(t)) < \tau$, *meaning that the matrix* $I - \frac{1}{\tau}Y(t)X(t)$ *has only positive eigenvalues.*

Then the optimal control is provided by the state-feedback controller

$$u^*(t) = -G^{-1}(B_1'(t)X(t) + \Upsilon'(t))\hat{x}(t),$$

where the estimate $\hat{x}$ of the state vector x from the observed output $y(\cdot)$ is given by the *filter*

$$d\hat{x}(t) = \left(A - B_1G^{-1}\Upsilon' - \left(B_1G^{-1}B_1' - \frac{1}{\tau}B_2B_2'\right)X\right)\hat{x}(t)dt$$
$$+ \left(I - \frac{1}{\tau}YX\right)^{-1}(YC_2' + B_2D_2')\Gamma^{-1}\left(dy(t) - \left(C_2 + \frac{1}{\tau}D_2B_2'X\right)\hat{x}(t)\,dt\right).$$

Lemma 3 was established in [22, 28].

A new aspect of the theory is the need to answer the following question: What is the class of admissible controllers in this theory? It is required that in the notation $e(t) := x(t) - \hat{x}(t)$, $\epsilon(t) := e(t) + \frac{1}{\tau}YX\hat{x}(t)$, the process

$$\alpha(t) := B_2'Y\epsilon(t) - D_2'\Gamma^{-1}(C_2Y + D_2B_2')Y^{-1}e(t)$$

would generate the exponent

$$\zeta(t) = \exp\left\{\int_0^t \alpha'(s)dW(s) - \frac{1}{2}\int_0^t \|\alpha(s)\|^2 ds\right\},$$

which is known to be a martingale on $[0, T]$; see [29]. This condition holds at least for the linear controller [28]. Then it is easy to check the following fact: in the class of admissible controllers, the infimum of the optimality criterion is ensured by the linear controller.

Also, some comments are needed to generalize the results of this section to the case of *infinite-horizon* systems ($T = \infty$). The coefficients in the plant's equations at $T = \infty$ should be supposed to be independent of t, the condition $y(0) = 0$ should be imposed, and the cost functional should be defined as $J(u) := \lim_{T\to\infty} \frac{1}{T}J_T(u)$. Next, the Riccati differential equations should be replaced by the algebraic ones, and assumptions (i)–(iii) of Lemma 3 should be further specified by replacing the existence condition for a symmetric solution Y with two conditions:

(a) $R - \Upsilon G^{-1}\Upsilon' \succeq 0$.

(b) The pair of matrices $(A - B_1G^{-1}\Upsilon', R - \Upsilon G^{-1}\Upsilon')$ is detectable, and the pair $(A - B_2D_2'\Gamma^{-1}C_2, B_2(I - D_2'\Gamma^{-1}D_2))$ is stabilizable.

In addition, the designations X and Y in Lemma 3 should be replaced by the commonly used X_∞ and Y_∞, and the following assumptions should be made:

(i)$'$ The equation for Y admits a minimal solution $Y_\infty \succ 0$.

(ii)$'$ The equation for X admits a minimal solution $X_\infty \succeq 0$.

Then it is proved that X_∞ and Y_∞ can be obtained from $X = X_T$ and $Y = Y_T$, respectively, by passing to the limit as $T \to \infty$. This result constitutes the next lemma.

Lemma 4. *Under the above conditions, the following statements are true:*

(*i*) *For* $M = 0$, *the differential equation associated with* (2.4) *has a solution* $X_T(t) \succeq 0$ *such that* $X_T(t) \to X_\infty$ *as* $T \to \infty$.

(*ii*) *If* $Y_\infty \succeq Y_0 \succ 0$, *then for* $M = 0$, *the differential equation associated with* (3.6) *has a solution* $Y(t; Y_0)$ *such that* $Y(t; Y_0) \to Y_\infty$ *as* $t \to \infty$.

(*iii*) *For each* $T > 0$ *and* $t \in [0, T]$, *the matrices* $Y(t; Y_0)$ *and* $X_T(t; M)$ *satisfy the inequality* $\rho(Y(t)X(t)) < \tau$.

See [22, 28].

Lemma 4 is proved as follows. First, we check that for $M = 0$, the solution $X_T(t) \succeq 0$ of equation (2.4) converges to X_∞ as $T \to \infty$. To verify statement (*ii*) of Lemma 4, we consider the system

$$\dot{\eta} = A'\eta + C_2'\nu + R^{1/2}\omega, \quad \eta(0) = \eta_0, \quad \zeta = B_2'\eta + D_2'\nu,$$

where $\eta \in R^n$ is the state vector, $\nu \in R^l$ is the control input, $\omega \in R^q$ is a disturbance, and $\zeta \in R^p$ is the regulated output, and define the functionals

$$\begin{aligned} \bar{J}_\tau^{T,Y_0}(\nu, \omega) &:= \eta(T)'Y_0\eta(T) + \int_0^T (\|\zeta(t\|^2 - \tau\|\omega(t)\|^2)dt, \\ \bar{J}_\tau(\nu, \omega) &:= \int_0^\infty (\|\zeta(t)\|^2 - \tau\|\omega(t)\|^2)\, dt,\ \omega \in L_2[0, T]. \end{aligned} \tag{4.1}$$

Applying then differential game theory, we conclude that $\inf_\nu \sup_\omega \bar{J}_\tau(\nu, \omega) < \infty$ on an infinite horizon and there exists the limit Y_∞ for equation (3.6) with the solutions Y_T for each $T < \infty$. Thus, equation (3.6) has a minimal solution $Y_\infty \succ 0$, and by Lemma 1 of Section 2,

$$\inf_\nu \sup_\omega \bar{J}_\tau(\nu, \omega) = \eta_0' Y_\infty \eta_0. \tag{4.2}$$

Finally, since the variables X^+ and Y_∞ are dual, the formula $\nu^* = -\Gamma^{-1}(D_2B_2' + C_2Y_\infty)\eta$ is dual to formula (2.5), written as $u^* = K^*x(t)$, where $K^* = -G^{-1}(D_1'C_1 + B_1'X^+)$. Similarly, formulas (2.6) and (4.2) are dual as well. However, the inequality $Y_\infty \succeq Y_0$ has no analog in Lemma 1. But formula (4.1) and $Y_0 \preceq Y_\infty$ obviously imply $\bar{J}_\tau^{T,Y_0} \preceq \bar{J}_\tau^{T,Y_\infty}$ for $\nu = \nu^*$, and we ultimately arrive at (4.2). See [22, 28].

5. STATE-FEEDBACK H_∞ CONTROLLERS FOR UNCERTAIN STOCHASTIC SYSTEMS

This section considers a general multiplicative stochastic system with a state-feedback controller. The controlled plant is described by equations (1.1) and (1.2); the controller, by the formula $u(t) = Kx(t)$; the closed-loop system state, by the equation

$$dx = ((A + B_1K)x + B_2v)dt + A_0xdw_0 + B_{01}Kxdw_1 + B_{02}vdw_2. \tag{5.1}$$

System (5.1) is uncertain due to the dependence of its dynamics on the unknown parameter K to be found. The coefficient at x in the diffusion component of this equation, equal to $A_0dw_0 + B_{01}Kdw_1$, can be written as

$$([A_0\ 0] + [0\ B_{01}]K)dW_1, \quad dW_1 := \begin{pmatrix} dw_0 \\ dw_1 \end{pmatrix}.$$

Thereby, we replace the three Wiener processes w_0, w_1, and w_2 in (5.1) with the two processes W_1 and w_2. In other words, keeping the same notation for the matrix coefficients, it is possible to let $w_0 = w_1$ in (5.1):

$$dx = ((A + B_1K)x + B_2v)dt + (A_0 + B_{01}K)xdw_1 + B_{02}vdw_2. \tag{5.2}$$

Further, without loss of generality, we take

$$z(t) = C_1x(t) + D_1u(t), \quad y(t) = C_2x(t) + D_2v(t)$$

instead of the pair of equations (1.2). For $t \geqslant s$, where s is the initial time instant, the closed-loop system

$$dx = (A + B_1K)xdt + (A_0 + B_{01}K)xdw_1, \quad x(s) = h, \tag{5.3}$$

obtained from (5.2) with $v(t) \equiv 0$, will be called nominal. This system is time-varying even under constant matrix coefficients in (5.2), and its stability should be understood, e.g., as exponential stability in mean square if the solution $x(t)$ is desired to be an element of the space $L_2(s,\infty)$ of all functions for which $\int_s^\infty \|x(t)\|^2dt < \infty$. In the nonstationary case, one should find a suitable analog of the boundedness $\|H_{zv}\|_\infty < \gamma$ of the induced norm of the operator H_{zv}. The following condition is a suitable stochastic analog of norm boundedness: there exists a constant $\epsilon > 0$ such that for $x(s) = 0$,

$$\mathsf{E}\int_s^\infty \left(\|(C_1 + D_1K)x(t)\|^2 - \gamma^2\|v(t)\|^2\right)dt \leqslant -\epsilon\gamma^2\mathsf{E}\int_s^\infty \|v(t)\|^2dt \tag{5.4}$$

for each $v \in L_2(s,\infty)$. Note that the exponential stability of the nominal system is a *sufficient* condition for equation (5.2) to have solutions belonging to $L_2(s,\infty)$ for any $v \in L_2(s,\infty)$.

Now we formulate the stochastic H_∞ controller design problem solved in this section: *for system* (5.2)*, find a matrix K such that the nominal system* (5.3) *is exponentially stable in mean square, and the closed-loop system* (5.2) *satisfies the requirement* (5.4) *of the H_∞ boundedness of the norm of the transfer operator H_{zv}.* As in Section 2, the most fruitful approach for the stochastic case is to apply the theory of linear-quadratic differential games associated with the Itô multiplicative equation and the functional $J(u,v) = \int_s^\infty \mathsf{E}(z'(t)z(t) - \gamma^2\|v(t)\|^2)dt$. Here, the bridge between the theory of H_∞ controllers and game theory is the *stochastic BR* lemma; see the presentation below, following the monograph [8].

Suppose that the system $dx(t) = Ax(t)dt + A_0x(t)dw_1(t)$, $x(s) = h$, corresponding to the choice $u(\cdot) \equiv 0$ and $v(\cdot) \equiv 0$ in equation (5.2), is exponentially stable (and hence the matrix A is stable) and there exists a constant ϵ_2 such that $J_2(0,v) \leqslant -\epsilon_2\int_0^\infty \|v(t)\|^2dt$ for all $v \in L_2(0,\infty)$. Then the following stochastic BR lemma is true.

Lemma 5. *Part 1* (*existence*):

(*a*) *For each $s \geqslant 0$ and $h \in L_2(\Omega,\mathcal{F}_s,P)$, there exists a unique disturbance $v_2^s(\cdot) \in L_2(s,\infty)$ such that $J(0,v_2^s(\cdot)) = \sup_{v\in L_2(s,\infty)} J(0,v)$.*

(*b*) *There exists a matrix $X_2 \succeq 0$ such that $\sup_{v\in L_2(s,\infty)} J(0,v) = \mathsf{E}\langle h, X_2h\rangle$.*

(*c*) *For any $T > 0$, given $X_{2T}(T) = 0$, there exists a unique solution $X_{2T}(\cdot) \succeq 0$ of the generalized Riccati equation*

$$\dot{X}_{2T} + A'X_{2T} + X_{2T}A + A_0'X_{2T} + R + X_{2T}B_2(I - B_{02}'X_{2T}B_{02})^{-1}B_2'X_{2T} = 0.$$

Part 2 (minimax solution):

(*d*) *The worst-case disturbance* $\upsilon_{2T}^s(\cdot)$ *maximizing the functional*

$$J_T(u,\upsilon)=\int\limits_s^T \mathsf{E}(z'(t)z(t)-\gamma^2\|\upsilon(t)\|^2)dt,\quad s\leqslant T<\infty,$$

has the form

$$\upsilon_{2T}^s=(I-B_{02}'X_{2T}(t)B_{02})^{-1}B_2'X_{2T}(t)x_{2T}^s(t),$$

where $x_{2T}^s(\cdot)$ *is the optimal trajectory representing the solution of the closed-loop system*

$$\begin{aligned}dx(t)=(A+B_2(I-B_{02}'X_{2T}(t)B_{02})^{-1}B_2'X_{2T}(t))x(t)dt+A_0x(t)dw_1(t)\\+B_{02}(I-B_{02}'X_{2T}(t)B_{02})^{-1}B_2'X_{2T}(t)x(t)dw_2(t),\quad x(s)=h.\end{aligned}$$

(*e*) *The matrix* X_2 *in item* (*b*) *is also the minimal solution of the* generalized *Riccati equation*

$$A'X_2+X_2A+A_0'X_2A_0+R+X_2B_2(I-B_{02}'X_2B_{02})^{-1}B_2'X_2=0,$$

with the property $I-B_{02}'X_2(t)B_{02}\succ 0$.

We formulate the main theorem of Section 5 for the case of a multiplicative system on an infinite horizon.

Theorem 1. *Under the assumptions of the stochastic BR lemma, let* $\|C_1x+D_1u\|^2>\epsilon_1\|u\|^2$ *for some* $\epsilon_1>0$ *and all* $x\in R^n$, $u\in R^m$. *Then:*

(*a*) *For* $h\in L_2(\Omega,\mathcal{F}_s,P)$ *and all* $s\geqslant 0$, *there exists a unique minimax pair for the functional* $J(u,\upsilon)$.

(*b*) *There exists a unique solution* $X\succeq 0$ *of the algebraic Riccati equation*

$$\begin{aligned}A'X+XA+A_0'XA_0+R+XB_2(I-B_{02}'XB_{02})^{-1}B_2'X\\-(XB_1+A_0'XB_{01}+Q)(G+B_{01}'XB_{01})^{-1}(XB_1+A_0'XB_{01}+Q)'=0\end{aligned}\tag{5.5}$$

such that $I-B_{02}'XB_{02}\succ 0$ *and* $V=\mathsf{E}\langle h,Xh\rangle$.

(*c*) *The minimax pair is given by* $u=F_1x$, $\upsilon=F_2x$, *where*

$$\begin{aligned}F_1&=-(G+B_{01}'XB_{01})^{-1}(XB_1+A_0'XB_{01}+Q)',\\F_2&=(I-B_{02}'XB_{02})^{-1}B_2'X).\end{aligned}\tag{5.6}$$

(*d*) *The closed-loop stochastic system*

$$dx=(A+B_1F_1+B_2F_2)xdt+(A_0+B_{01}F_1)xdw_1(t)+B_{02}F_2xdw_2(t)\tag{5.7}$$

with $x(s)=h$ *is exponentially stable in mean square.*

It seems interesting to compare the conclusions in Theorem 1 with the results of solving the same problem on a finite horizon [30]. Let

$$J_T(u,\upsilon)=\int\limits_s^T \mathsf{E}(z'(t)z(t)-\gamma^2\|\upsilon(t)\|^2)dt,\quad s\leqslant T<\infty.$$

Consider the standard stochastic control problem $\inf\limits_{u\in L_2[s,T]} J_T(u,0)$. The solution $X_{1T} \succeq 0$ of the generalized Riccati equation

$$\begin{aligned}&\dot X_{1T} + A'X_{1T} + X_{1T}A + A_0'X_{1T}A_0 + R - (X_{1T}B_1 + A_0'X_{1T}B_{01} + Q)\\&\times (G + B_{01}'X_{1T}B_{01})^{-1}(X_{1T}B_1 + A_0'X_{1T}B_{01} + Q)' = 0, \quad X_{1T}(T) = 0,\end{aligned} \tag{5.8}$$

is associated with this problem. The equation has a unique solution $X_{1T} \succeq 0$ determining the optimal controller

$$u_{1T} = -(G + B_{01}'X_{1T}B_{01})^{-1}(B_1'X_{1T} + B_{01}'X_{1T}A_0 + Q')x.$$

The solutions X_{1T} and X_{2T} of both Riccati differential equations allow solving the minimax control problem with the general criterion $J_T(u,\upsilon)$ on a *finite horizon*. The following result was established in [8].

Theorem 2. *Let $F(x,u) > \epsilon_1\|u\|^2$ and $J(0,\upsilon) \leqslant -\epsilon_2\int_0^\infty \|\upsilon(t)\|^2dt$ for all $\upsilon \in L_2(0,\infty)$. Then the game-theoretic minimax problem with the criterion $J_T(u,\upsilon)$ has a saddle point $(u_T(\cdot),\upsilon_T(\cdot))$. The Riccati differential equation*

$$\begin{aligned}&\dot X_T + A'X_T + X_TA + A_0'X_TA_0 + R - (X_TB_1 + A_0'X_TB_{01} + Q)(G + B_{01}'X_TB_{01})^{-1}\\&\times(X_TB_1 + A_0'X_TB_{01} + Q)' + X_TB_2(I - B_{02}'X_TB_{02})^{-1}B_2'X_T = 0, \quad X_T(T) = 0\end{aligned} \tag{5.9}$$

has a unique solution $X_T \succeq 0$. The saddle point of the game is given by $u_T = F_{1T}x$, $\upsilon_T = F_{2T}x$, where

$$\begin{aligned}F_{1T} &= -(G + B_{01}'X_TB_{01})^{-1}(B_1'X_T + B_{01}'X_TA_0 + Q'),\\F_{2T} &= (I - B_{02}'X_TB_{02})^{-1}B_2'X_T.\end{aligned}$$

The proof of this theorem is quite complicated, it uses results from several works [31–33].

It is interesting to compare the Riccati equations (5.6) and (5.5): the last term on the right-hand side of (5.6) vanishes in (5.5). Of course, this is because equation (5.5) is associated with the criterion $J_T(u,0)$ whereas equation (5.6) with the criterion $J_T(u,\upsilon)$. The appearance of the term $X_TB_2(I - B_{02}'X_TB_{02})^{-1}B_2'X_T)^{-1}$ seems quite natural in the problem of maximizing the functional $J_T(u,\upsilon)$ with respect to the second argument.

Also, it is interesting to represent the criterion $J_T(u,\upsilon)$ through the functions u_T and υ_T determining the saddle point $(u_T(\cdot),\upsilon_T(\cdot))$. Suppose that a solution X_T of equation (5.6) exists on the closed interval $[T-\alpha,T]$, and let s be a point of this interval. Then, applying Itô's formula to the quadratic form $x(t)'X_Tx(t)$, where $x(t)$ satisfies equation (1.1) under the constraint $w_0 = w_1$, we obtain

$$\begin{aligned}J_T(u,\upsilon) = \mathsf{E}\int_s^T \|(I - B_{02}'X_TB_{02})^{\frac12}(\upsilon(t) - F_{2T}x(t))\|^2dt\\+ \mathsf{E}\int_s^T \|(G + B_{01}'X_TB_{01})^{\frac12}(u(t) - F_{1T}x(t))\|^2dt.\end{aligned}$$

Therefore, the saddle point satisfies the condition

$$J_T(u_T,\upsilon) \leqslant J_T(u_T,\upsilon_T) = \mathsf{E}h'X_T(s)h \leqslant J_T(u,\upsilon_T).$$

For a sufficiently small parameter α, this condition is used to prove the existence of a solution of the Riccati differential equation (5.6) on a finite interval $t \in [s,T]$.

6. OUTPUT-FEEDBACK H_∞ CONTROLLERS FOR STOCHASTIC SYSTEMS WITH MULTIPLICATIVE DISTURBANCES

Consider an Itô stochastic differential equation of the form

$$\Sigma : \begin{cases} dx(t) = (Ax(t) + B_1 \upsilon(t) + B_2 u(t))dt + A_0 x(t) dw_0(t) \\ \qquad\qquad + B_{01}\upsilon(t) dw_1(t) + B_{02} u(t) dw_2(t), \ x_0 \in \mathrm{R}^n \\ z(t) = C_1 x(t) + D_{11}\upsilon(t) + D_{12} u(t) \\ y(t) = C_2 x(t) + D_{21}\upsilon(t), \quad t \in [0, T]. \end{cases} \tag{6.1}$$

The controller in the feedback loop is a deterministic *dynamic* system with a state vector $\hat{x}$, described by the equations

$$d\hat{x}(t) = A_k \hat{x}(t) dt + B_k\, y(t) dt, \quad u(t) = C_k \hat{x}(t) + D_k\, y(t), \tag{6.2}$$

where all matrix coefficients are to be determined.

We denote by $\bar{x}$ the state vector of the closed-loop system and let $\bar{x}' := (x', \hat{x}')$. The stochastic equation for $\bar{x}$ is obtained by some cumbersome, albeit elementary, calculations. As is easily verified, the functions $t \mapsto \bar{x}(t),\ t \mapsto z(t)$ must satisfy the equations

$$\Sigma_2 : \begin{cases} d\bar{x}(t) = A_{cl}\bar{x}(t)\, dt + B_{cl}\upsilon(t)\, dt + A^0_{cl}\bar{x}(t) dw_0(t) \\ \qquad\qquad + B^0_{cl}\upsilon(t) dw_1(t) + A^1_{cl}\bar{x}(t)\, dw_2(t) + B^1_{cl}\upsilon(t)\, dw_2(t) \\ z(t) = C_{cl}\bar{x}(t) + D_{cl}\upsilon(t), \quad t \in [0, T]. \end{cases} \tag{6.3}$$

The formulas expressing the coefficients of the equations of the closed-loop system Σ_2 are directly derived; for example, see [34].

The main result of the theory of the dynamic output-feedback controller is provided immediately below.

Theorem 3. *For system* (6.1) *and* $\gamma > 0$, *the following statements are equivalent:*

(*i*) *There exists a controller* (6.2) *such that the corresponding closed-loop system* (6.3) *is internally stable and* $\|H_{z\upsilon}\|_\infty < \gamma$.

(*ii*) *There exists a matrix function* $P : t \mapsto P(t) \prec 0$ *such that* $\mathcal{M}(\gamma, P) \succ 0$ *for all* $t \in [0, T]$.

See [7, Theorem 3.3]. Here, $\mathcal{M}(\gamma, P)$ is the block matrix of dimensions 2×2 for a quadratic form in the space of vector pairs $\binom{\bar{x}}{\upsilon}$.

For the further presentation, it is convenient to write the condition $\mathcal{M}(\gamma, P) \succ 0$ in equivalent form, replacing the block diagonal matrix $\mathrm{diag}\{\mathcal{M}(\gamma, P), I\}$ with a block matrix $T\mathcal{N}(\gamma, P)T' \succ 0$ of dimensions 3×3, where

$$\mathcal{N}(\gamma, P) = \begin{pmatrix} PA_{cl} + A'_{cl}P + S_{11} & PB_{cl} + S_{12} & C'_{cl} \\ B'_{cl}P + S_{21} & \gamma^2 I + S_{22} & D'_{cl} \\ C_{cl} & D_{cl} & I \end{pmatrix}, \quad T = \begin{pmatrix} I & O & -C'_{cl} \\ O & I & -D'_{cl} \\ O & O & I \end{pmatrix}.$$

The matrices S_{ij} were found in [34]. For the blocks (submatrices) N_{ij} of the nonnegative definite matrix $\mathcal{N}(\gamma, P)$, the formulas presented therein are

$$N_{11} = P(A^0 + B^I M_k C^I) + (A^0 + B^I M_k C^I)'P + S_{11},$$
$$N_{12} = P(B^0 + B^I M_k D^0_{21} + S_{12}), \quad N_{13} = (C^0 + D^0_{12} M_k C^I)',$$
$$N_{22} = \gamma^2 I + S_{22}, \quad N_{23} = (D_{11} + D^0_{12} M_k D^0_{21})', \quad N_{33} = I,$$

where $M_k = \begin{pmatrix} A_k & B_k \\ C_k & D_k \end{pmatrix}$ is the parameter matrix of the dynamic controller. The matrices A_{cl}, B_{cl}, C_{cl}, and D_{cl} are expressed affinely through the matrix M_k :

$$A_{cl} = A^0 + B^I M_k C^I, \quad B_{cl} = B^0 + B^I M_k D^0_{21},$$
$$C_{cl} = C^0 + D^0_{12} M_k C^I, \quad D_{cl} = D_{11} + D^0_{12} M_k D^0_{21},$$

with some matrix coefficients [30]. We represent the matrix $\mathcal{N}(\gamma, P)$ as the sum of a matrix $\mathcal{H}$, independent of M_k, and two matrices that linearly depend on M_k. As a result, $\mathcal{H}+\mathcal{Q}'M_k'\mathcal{R}+\mathcal{R}'M_k\mathcal{Q}$ with some matrices $\mathcal{Q}$ and $\mathcal{R}$ and a nonnegative definite matrix $\mathcal{H}$. According to the stochastic generalization [7] of the *projection lemma* from the theory of linear matrix inequalities (LMIs, see [35]), the *LMI*

$$\mathcal{H} + \mathcal{Q}'M_k'\mathcal{R} + \mathcal{R}'M_k\mathcal{Q} \succ O$$

has a solution M_k iff the matrix $\mathcal{H}$ is positive definite on the null subspaces $\ker\mathcal{Q}$ and $\ker\mathcal{R}$ of the matrices $\mathcal{Q}$ and $\mathcal{R}$, respectively.

The projection lemma gives a necessary and sufficient condition for $\|H_{zv}\|_\infty < \gamma$. The condition is formulated in terms of LMIs instead of matrix differential equations. This lemma settles the issue about the admissibility conditions of the controller M_k and, moreover, allows calculating the parameters A_k, B_k, C_k, and D_k of the controller if they are unknown [7].

7. STOCHASTIC ROBUST ANALYSIS OF THE SYSTEM WITH PARAMETRIC PERTURBATION

Let

$$dx(t) = Ax(t)dt + A_0x(t)dw_1(t), \quad 0 < t < T, \tag{7.1}$$

be a nominal stochastic system and

$$dx(t) = (A + B\Delta C)x(t)dt + A_0x(t)dw_1(t) + B_0\Delta Cx(t)dw_2(t) \tag{7.2}$$

be its *stochastic* disturbance with a simultaneous *parametric* perturbation of the matrix parameter A. In system (7.2), the perturbing parameter is an arbitrary matrix Δ from the set $\mathcal{D} = R^{l\times q}$ of all matrices of dimensions $l \times q$. Being time-varying, the nominal system (7.1) is assumed to be stable in the following sense: there exists a constant $c > 0$ such that $\mathsf{E}\int_0^\infty \|x(t)\|^2 dt \leqslant c\|x^0\|^2$, where $x(\cdot) = x(\cdot, x^0)$ is the trajectory of equation (7.1) starting at $x^0 \in R^n$. The Wiener processes w_1 and w_2 are independent, the perturbing force is a state-multiplicative process, and the uncertainty of system (7.2) is measured by the norm $\|\Delta\|$ of the matrix Δ.

Assume that $\rho > 0$ is a small number. For small $\|\Delta\| < \rho$, system (7.2) is close to the unperturbed one (7.1) and is probably stable as well. What is the value $\rho_{\max}$ of the parameter ρ under which the stability of system (7.2) is ensured for each Δ from the set $\mathcal{D} = \{\Delta : \|\Delta\| < \rho_{\max}\}$? It is natural to call the number $\rho_{\max}$ the *robust* stability radius of system (7.1) with respect to uncertainties $\Delta \in \mathcal{D}$. Accordingly, we introduce the following definition: the number $r_{\mathcal{D}} = \inf\{\|\Delta\| :$ system (7.2) with the uncertainty Δ is unstable$\}$ is called the robust stability radius of system (7.1).

Below, to relate robust stability to H_∞ control analysis (based on the stochastic BR lemma), let us address the standard plant of the bilinear stochastic H_∞ control theory:

$$\begin{cases} dx(t) = Ax(t)dt + Bv(t)dt + A_0x(t)dw_1(t) + B_0v(t)dw_2(t) \\ z(t) = Cx(t). \end{cases} \tag{7.3}$$

Interpreting υ as a control action, we consider a block with the input z and output $\upsilon = \Delta z$ in the feedback loop. Then the closed-loop system becomes the perturbed system (7.2). In H_∞ control theory, the operator associated with this system is a disturbance operator $L : \upsilon \mapsto z$, specifying the effect of an exogenous disturbance (here, control) υ on the output z. The operator L acts from the space of functions $\upsilon(\cdot)$ into the space of functions $z(\cdot) = Cx(\cdot)$, where $x(\cdot) = x(\cdot, \upsilon, x^0)|_{x^0=0}$. Thus, $L : \upsilon(\cdot) \mapsto Cx(\cdot, \upsilon, 0)$.

The above definition of the robust stability radius is generalized to nonstationary and nonlinear uncertainties. For each $t \in R_+$, let $\Delta(t, \cdot)$ be a linearly bounded mapping $R^q \to R^l$, i.e., $\|\Delta(t, y)\| \leqslant K\|y\|$ for some $K > 0$ and all $t \in R_+$ and $y \in R^q$. In addition, assume that this mapping is Lipschitz bounded: for any $T > 0$ there exists a constant $L(T)$ such that $\|\Delta(t, y_1) - \Delta(t, y_2)\| \leqslant L(T)\|y_1 - y_2\|$ for all $y_1, y_2 \in R^q$ and all $t \in [0, T]$. The uncertainty Δ of this kind is nonlinear and nonstationary. Its value is found as the smallest K in the definition of linear boundedness, and the equation associated with such uncertainty has the form

$$dx(t) = (Ax(t) + B\Delta(t, Cx(t)))dt + A_0x(t)dw_1(t) + B_0\Delta(t, Cx(t))dw_2(t). \tag{7.4}$$

Let $\mathcal{D}_{tn}$ be the set of all such uncertainties. We denote by $x_\Delta(\cdot, x^0)$ the solution of equation (7.4), which is assumed to be unique in the class of random functions $L^2([0, T]; L^2(\Omega, R^n))$, and call system (7.4) stable if its solutions satisfy the condition $\int_0^\infty \mathsf{E}\|x_\Delta(t, x^0)\|^2 dt \leqslant c\|x^0\|^2$ for some constant c. In H_∞ control theory, it is required to ensure the inequality $\|L_{z\upsilon}\| < \gamma$; in robust systems theory, the inequality $\|\Delta_{\upsilon z}\| < \rho$. We choose γ as small as possible and ρ as large as possible.

Due to the linear and Lipschitz boundedness conditions of the function Δ, equation (7.4) has a unique solution $x_\Delta(\cdot, x^0)$, which is a stochastic process with bounded second moments [36]. In integral form, equation (7.4) is written as

$$x_\Delta(t) = x^0 + \int_0^t (Ax_\Delta(s) + B\upsilon_\Delta(s))ds + \int_0^t [A_0x_\Delta(s)\ B_0\upsilon_\Delta(s)]dw(s), \quad t \in [0, T],$$

for each $T > 0$, where $\upsilon_\Delta(\cdot) = \Delta(\cdot, Cx_\Delta(\cdot)), w(s) = [w_1(s),\, w_2(s)]'$. Let $\|\Delta\| < \|L_{z\upsilon}\|^{-1}$, then there exists a number $\gamma > \|L_{z\upsilon}\|$ such that $\gamma\|\Delta\| < 1$. In H_∞ control theory, the functional

$$J_T(x^0, \upsilon) = \int_0^T \mathsf{E}[\gamma^2\|\upsilon(t)\|^2 - \|z(t)\|^2]dt$$

is given by the well-known formula

$$J_T(x^0, \upsilon) = \langle x^0, P(0)x^0\rangle - \mathsf{E}\langle x(T), P(T)x(T)\rangle + \int_0^T \mathsf{E}\left\langle \begin{pmatrix} x(t) \\ \upsilon(t) \end{pmatrix}, M(P(t)) \begin{pmatrix} x(t) \\ \upsilon(t) \end{pmatrix} \right\rangle dt. \tag{7.5}$$

According to the stochastic BR lemma, for any $\gamma > 0$ the following statements are equivalent:

(i) There exists a matrix $P \prec 0$ such that $M(P) \succ 0$.

(ii) The equation for $x(\cdot)$ is internally stable, with $\|L_{z\upsilon}\| < \gamma$.

The relation $M(P) \succ 0$ implies $M(P) \succeq \delta^2 I$ for some number $\delta > 0$. Calculating $J_T(x^0, \upsilon)$ (7.5) for $x(\cdot) = x_\Delta(\cdot, x^0)$ and $\upsilon(\cdot) = \upsilon_\Delta(\cdot)$ yields

$$J_T(x^0, \upsilon_\Delta) = \int_0^T \mathsf{E}[\gamma^2\|\Delta(t, Cx_\Delta(t))\|^2 - \mathsf{E}\|Cx_\Delta(t)\|^2]dt. \tag{7.6}$$

To proceed, we formulate an important result on the stability of the uncertain system (7.4).

Theorem 4. *Assume that system* (7.1) *with* $\Delta = 0$ *is stable and the uncertainty* $\Delta, \Delta \neq 0$, *satisfies the condition* $\|\Delta\| = \sup\{\|\Delta(t,y)\| \diagup \|y\| : t > 0, y \in R^q, y \neq 0\} < \|L\|^{-1}$, *where* $L = L_{z\upsilon}$. *Then the perturbed* (*uncertain*) *system* (7.4) *is stable. In particular,* $r_{\mathcal{D}_{tn}} \geqslant \|L\|^{-1}$.

Indeed, since

$$\gamma^2\|\Delta(t, Cx_\Delta(t)\|^2 \leqslant \gamma^2\|\Delta\|^2\|Cx_\Delta(t)\|^2 \leqslant \gamma^2\|Cx_\Delta(t)\|^2$$

by the condition $\|L\|\|\Delta\| < \gamma\|\Delta\| < 1$, from (7.6) it follows that $J_T(x^0, \upsilon_\Delta) \leqslant 0$. By analogy, letting $x(\cdot) = x_\Delta(\cdot, x^0)$, $\upsilon(\cdot) = \upsilon_\Delta(\cdot)$, and $M(P) \succeq \delta^2 I$, we estimate the integral in (7.5) as

$$\int_0^T \delta^2 \mathsf{E}\{\|x_\Delta(t)\|^2 + \|\upsilon_\Delta(t)\|^2\}dt \geqslant \int_0^T \delta^2 \mathsf{E}\|x_\Delta(t)\|^2 dt.$$

Thus, for $x(\cdot) = x_\Delta(\cdot, x^0)$ and $\upsilon(\cdot) = \upsilon_\Delta(\cdot)$, formula (7.5) leads to the inequality

$$\mathsf{E}\langle x_\Delta(T), (-P)x_\Delta(T)\rangle \leqslant \langle x_0, (-P)x_0\rangle - \int_0^T \delta^2 \mathsf{E}\|x_\Delta(t)\|^2 dt$$

for any $T > 0$. The left-hand side of this inequality is positive due to $-P \succ 0$; therefore, $\int_0^T \delta^2 \mathsf{E}\|x_\Delta(t)\|^2 dt \leqslant \|P\|\|x^0\|^2/\delta^2$. This proves the stability of system (7.4).

As is easily verified, the perturbed closed-loop system obtained by substituting $\upsilon = \Delta z$ into equation (7.3) can be reduced to

$$d\bar{x}(t) = (A_{cl} + B_{cl}\Delta C_{cl})\bar{x}(t)dt + \left(A^0_{cl}dw_1(t) + B^0_{cl}\Delta C_{cl}dw_2(t)\right)\bar{x}(t),$$

where $\bar{x} := (x, \hat{x})$. Its coefficients with the *cl* subscript are calculated straightforwardly. In combination with Theorems 3 and 4, this result gives the following fact.

Lemma 6. *Let* γ_{opt} *be the infimum of those* $\gamma \geqslant 0$ *for which there exists a dynamic controller ensuring the internal stability of the closed-loop system* (7.4) *and the condition* $\|L_{cl}\| < \gamma$. *Then for any* $\gamma > \gamma_{opt}$, *there exists a controller* Δ *such that the corresponding closed-loop system* (7.4) *has a robust stability radius* $r_{\mathcal{D}}(A_{cl}, B_{cl}, C_{cl}, A^0_{cl}, B^0_{cl}) > \gamma^{-1}$.

For details, we refer to [7].

8. CONCLUSIONS

The topic of this paper has been the H^2/H_∞ branch of general control theory for technical systems (plants). This branch deals with, first, the problems of controller *analysis* and, second, the problems of controller *design*, i.e., optimization in the class of *admissible* controllers identified at the analysis stage. Note that H_∞ control theory solves the problem of rejecting an exogenous disturbance acting on a plant in the closed loop with an admissible controller. An admissible controller ensures that, first, the closed-loop system (plant + controller) is stable and, second, the H_∞ norm of the transfer operator relating an exogenous disturbance υ to a regulated output z is below a threshold $\gamma > 0$, set a priori by the designer: $\|H_{z\upsilon}\|_\infty < \gamma$. The controller is defined as a functional mapping $K : y \mapsto u$ of an observed output y (measured by a noisy sensor) into a control action u. In this problem statement, the system must have two inputs υ and u as well as two outputs z and y. Further, the system must be stochastic, given by an Itô stochastic equation whose diffusion term is not arbitrary but of a partial (multiplicative) structure; however, it is not the same as the structure in the linear Itô equation. The stochastic nature of the system is due to that of, first, the exogenous disturbance and /or observed output and, second, the nominal (unperturbed) system with zero exogenous disturbance. The standard model of a stochastic system in H_∞ control theory is a perturbed system. And the nominal system is perturbed by two stochastic forces: the force $B_1 v dt$ (which may be deterministic) and the stochastic force $B_0 v dw_2$. The presence of both types of disturbances is an essential point of the most complete generalization of H_∞ control theory. This is the case for the H_∞ controller design problem in the statistical H^2/H_∞ control theory.

On the other hand, robust control theory deals with uncertain systems, their robust stability, and calculation of the stability radius of closed-loop systems. The question arises: What is the relation between the theories of controllers for stochastic systems in robust control and H_∞ control theory? A detailed answer to this question has been provided in Section 7 of the paper. As it turned out, in H_∞ control theory, of interest are the operator $H_{zv} : v \mapsto z$ and its induced norm; in robust control theory, the operator $\Delta : z \mapsto v$ and its norm $\|\Delta\|$, which plays a crucial role when determining the stability radius of an uncertain system. Various aspects of the linear theory of stochastic robust systems were reflected in the monograph [37].

The theory of multiplicative stochastic systems presented in the review is some generalization of the linear theory since the diffusion term in the state equation is taken here as multiplicative instead of linear. An unconventional approach to the theory of Gaussian systems has been described in Section 4. Also, note interesting results of the stochastic theory of controllers based on the observed output y, also discussed in the review. Here, we have to pass to an augmented system with the state vector $\bar{x} = (x, \hat{x})$, where $\hat{x}$ is the controller's state vector calculated from the output y. After that, it seems interesting to compare the theory of such controllers with those of deterministic ones and controllers based on the state vector x.

Finally, let us emphasize possible lines to develop further the stochastic H^2/H_∞ control theory and its generalizations. In this context, we mention works on control theory for time-invariant systems with bounded spectral characteristics [38], some classes of nonlinear systems [39], both robust [40] and nonrobust, and systems with non-Gaussian uncertainties [41] and incomplete information about the state vector [42, 43]. Research into the control theory of discrete systems continues as well.

REFERENCES

1. Zames, G., Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms, and Approximate Inverses, *IEEE Trans. Automat. Control*, 1981, vol. AC-26, pp. 301–320.

2. Doyle, J., Glover, K., Khargonekar, P., and Francis, B., State Space Solutions to Standard H_2 and H^∞ Control Problems, *IEEE Trans. Automat. Control*, 1989, vol. AC-34, pp. 831–847.

3. Francis, B.A., *A Course in H_∞ Control Theory*, Lecture Notes in Control and Information Sciences, vol. 88. New York: Springer-Verlag, 1987.

4. Doyle, J., Zhou, K., Glover, K., and Bodenheimer, B., Mixed H^2 and H_∞ Performance Objectives II: Optimal Control, *IEEE Trans. Automat. Control*, 1994, vol. 39, pp. 1575–1587.

5. Glover, K. and Doyle, J., State-Space Formulae for All Stabilizing Controllers That Satisfy an H_∞ Norm Bound and Relations to Risk Sensitivity, *Syst. Contr. Lett.*, 1988, vol. 11, pp. 167–172.

6. Limebeer, D.J.N., Anderson, B.D.O., Khargonekar, P., and Green, M., A Game Theoretic Approach to H_∞ Control for Time-Varying Systems, *SIAM J. Control Optim.*, 1992, vol. 30, pp. 262–283.

7. Hinrichsen, D. and Pritchard, A.J., Stochastic H_∞, *SIAM J. Control Optim.*, 1998, vol. 36, no. 5, pp. 1504–1538.

8. Petersen, I.R., Ugrinovsky, V.A., and Savkin, F.V., *Robust Control Design Using H_∞-Methods*, London: Springer, 2006.

9. Shaikin, M.E., Multiplicative Stochastic Systems: Optimization and Analysis, *Differ. Equations*, 2017, vol. 53, no. 3, pp. 1–16.

10. Ovsyannikov, L.V., *Gruppovoi analiz differentsial'nykh uravnenii* (Group Analysis of Differential Equations), Moscow: Nauka, 1978.

11. Shaikin, M.E., Resolvents of the Ito Differential Equations Multiplicative with Respect to the State Vector, *Autom. Remote Control*, 2023, vol. 84, no. 8, pp. 958–971.

12. Ikeda, N. and Watanabe, S., *Stochastic Differential Equations and Diffusion Processes*, 2nd ed., Amsterdam–Oxford–New York: North-Holland, 1989.

13. Erdogan, U. and Lord, G.J., A New Class of Exponential Integrators for Stochastic Differential Equations with Multiplicative Noise, *arXiv:1608.07096v2*, 2016.

14. Hochbruck, M. and Ostermann, A., Exponential Integrators, *Acta Numerica*, 2010, no. 19, pp. 209–286.

15. Mora, C.M., Weak Exponential Schemes for Stochastic Differential Equations with Additive Noise, *IMA J. Numer. Anal.*, 2005, vol. 25, no. 3, pp. 486–506.

16. Jimenez, J.C. and Carbonell, F., Convergence Rate of Weak Local Linearization Schemes for Stochastic Differential Equations with Additive Noise, *J. Comput. Appl. Math.*, 2015, vol. 279, pp. 106–122.

17. Komori, Y. and Burrage, K., A Stochastic Exponential Euler Scheme for Simulation of Stiff Biochemical Reaction Systems, *BIT*, 2014, vol. 54, no. 4, pp. 1067–1085.

18. Lord, G.J. and Tambue, A., Stochastic Exponential Integrators for the Finite Element Discretization of SPDEs for Multiplicative and Additive Noise, *IMA J. Numer. Anal.*, 2013, vol. 33, no. 2, pp. 515–543.

19. Melnikova, I.V. and Alshanskiy, M.A., Stochastic Equations with an Unbounded Operator Coefficient and Multiplicative Noise, *Siberian Math. Journal*, 2017, vol. 58, no. 6, pp. 1052–1066.

20. Green, M. and Limebeer, D.J.N., *Linear Robust Control*, Englewood Cliffs: Prentice-Hall, 1995.

21. Petersen, I.R., Anderson, B.D.O., and Jonckheere, E.A., A First Principles Solution to the Non-singular H^∞ Control Problem, *Int. J. Robust Nonlin. Control*, 1991, vol. 1, no. 3, pp. 171–185.

22. Basar, T. and Bernhard, P., *H^∞-optimal Control and Related Minimax Design Problems: a Dynamic Game Approach*, Boston: Birkhäuser, 1995.

23. Sampei, M., Mita, T., and Nakamichi, M., An Algebraic Approach to H^∞ Output Feedback Control Problems, *Syst. Control Lett.*, 1990, vol. 14, pp. 13–24.

24. Bernstein, D.S. and Haddad, W.M., Robust Stability and Performance Analysis for State-Space Systems via Quadratic Lyapunov Bounds, *SIAM J. Matrix Anal.*, 1990, vol. 11, no. 2, pp. 239–271.

25. Runolfsson, T., The Equivalence between Infinite-Horizon Optimal Control of Stochastic Systems with Exponential-of-Integral Performance Index and Stochastic Differential Games, *IEEE Trans. Automat. Control.*, 1994, vol. 39, no. 8, pp. 1551–1563.

26. Jacobson, D.H., Optimal Stochastic Linear Systems with Exponential Performance Criteria and Their Relation to Deterministic Differential Games, *IEEE Transact. Autom. Control*, 1973, vol. 18, no. 2, pp. 124–131.

27. Bensoussan, A. and van Schuppen, J.H., Optimal Control of Partially Observable Stochastic Systems with an Exponential-of-Integral Performance Index, *SIAM J. Control. Optim.*, 1985, vol. 23, pp. 599–613.

28. Pan, Z. and Basar, T., Model Simplification and Optimal Control of Stochastic Singularly Perturbed Systems under Exponentiated Quadratic Cost, *SIAM J. Control. Optim.*, 1996, vol. 34, no. 5, pp. 1734–1766.

29. Girsanov, I.V., On Transforming a Certain Class of Stochastic Processes by Absolutely Continuous Substitution of Measures, *Theory of Probability & Its Applications*, 1960, vol. 5, no. 3, pp. 285–301.

30. Zhou, K., Doyle, J., and Glover, J., *Robust and Optimal Control*, Upper Saddle River: Prentice-Hall, 1996.

31. Ugrinovskii, V.A. and Petersen, I.R., Absolute Stabilization and Minimax Optimal Control of Uncertain Systems with Stochastic Uncertainty, *SIAM J. Control Optim.*, 1999, vol. 37, no. 4, pp. 1089–1122.

32. Ichikawa, A., Quadratic Games and H_∞-Type Problems for Time Varying Systems, *Int. J. Contr.*, 1991, vol. 54, no. 5, pp. 1249–1271.

33. Bensoussan, A., *Stochastic Control of Partially Observable Systems*, Cambridge: Cambridge University Press, 1992.

34. Shaikin, M.E., Output Dynamic Controller Analysis for Stochastic Systems of Multiplicative Type, *Autom. Remote Control*, 2022, vol. 83, no. 4, pp. 343–354.

35. Gahinet, P. and Apkarian, P., A Linear Matrix Inequality Approach to H^∞ Control, *Int. J. Robust Nonlin. Control.*, 1994, vol. 4, pp. 421–448.

36. Krylov, N.V., *Introduction to the Theory of Diffusion Processes*, Providence: AMS, 1995.

37. Dragan, V., Morozan, T., and Stoica, A.M., *Mathematical Methods in Robust Control of Linear Stochastic Systems*, Mathematical Concepts and Methods in Science and Engineering, Springer, 2006.

38. Ma, P., Zhu, Z., and Sheng, L., Static Output Feedback H^2/H_∞ Control with Spectrum Constraints for Stochastic Systems, *Syst. Sci. Control Eng.*, 2018, vol. 6, no. 3, pp. 118–125.

39. Paulson, J.A. and Mesbah, A., An Efficient Method for Stochastic Optimal Control with Joint Chance Constraints for Nonlinear Systems, *Int. J. Robust Nonlin. Control*, 2019, vol. 29, no. 15, pp. 5017–5037.

40. Lefebvre, T., De Belie, F., and Crevecoeur, G., A Framework for Robust Quadratic Optimal Control, *Opt. Control Appl. Methods*, 2020, vol. 41, no. 3, pp. 833–848.

41. Wan, Y., Shen, D.E., Lusia, S., Findeisen, R., and Braatz, R.D., Polynomial Chaos-Based H^2 Output-Feedback Control of Systems with Probabilistic Parameter Uncertainties, *Automatica*, 2021, vol. 131, art. no. 109743.

42. Panteleev, A.V. and Yakovleva, A.A., Synthesis of H-Infinity Controllers in a Finite Time Interval, *Modelling and Data Analysis*, 2021, vol. 11, no. 1, pp. 5–19.

43. Wang, M., Meng, Q., Shen, Y., and Shi, P., Stochastic H^2/H_∞-Control for Mean-Field Stochastic Differential Systems with (x, u, v)-Dependent Noise, *J. Optim. Theory Appl.*, 2019, vol. 197, no. 3, pp. 1024–1060.

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. 2, pp. 153–164.

Approaches to Optimizing Guidance Methods to High-Speed Intensively Maneuvering Targets. Part III. Guidance Optimization Considering the Dynamic Properties of Interceptors

V. S. Verba[*,a], V. I. Merkulov[*,b], and A. R. Ilchuk[,c]**

[*] *JSC Concern Vega, Moscow, Russia*
[**] *LLC NPO Istok, Fryazino, Russia*
e-mail: [a] *vvs.msk@gmail.com,* [b] *mvipost41@gmail.com,* [c] *arilchuk@istokmw.ru*

Received March 20, 2024
Revised July 8, 2024
Accepted September 2, 2024

Abstract—Procedures for designing guidance methods to an object with a complex maneuvering law are considered. These procedures are adapted to the interceptor's dynamic properties, implemented by transforming the composition of information support and using the predicted spatial position of the target. Modeling results are presented.

Keywords: guidance method, control optimization, information support transformation, target position prediction

DOI: 10.31857/S0005117925020044

1. INTRODUCTION

According to an analysis of conventional guidance methods [1–4], with proportional guidance and predicted meeting point guidance as the most common ones, they were designed autonomously without considering the dynamic properties of aircraft interceptors for a non-maneuvering target.

When implementing interception systems for non-maneuvering and weakly maneuvering targets, such an approach imposes no significant constraints on the properties of aircraft interceptors (except for the maximum allowable overload). However, in the case of intensively maneuvering targets, this approach does not ensure the desired intercept performance, requiring an appropriate improvement of the maneuvering properties of an aircraft interceptor by adapting them to the complex nature of changes in the target's relative and absolute motion coordinates.

In turn, this leads to appreciable time and material costs and hence cannot be a universal technique due to the wide range of possible target maneuvers.

Therefore, there is a high demand for developing optimization methods considering the dynamic properties of interceptors.

The topicality of this problem grows with expanding the class of high-speed aircraft with complex maneuvering laws, including those with changes in the absolute value and signs of the derivatives of angular coordinates [4]. In this regard, it is necessary to develop optimization methods for generating control laws considering the dynamic properties of interceptors [5, 6, 8, 9].

In general, the interceptor's delay can be considered in various ways:

- by weighting the control error coefficients in accordance with the carrier's time constant and the gain of its control signals, which is characteristic of classical optimization methods in the statistical theory of optimal control (STOC) [5, 6];
- by describing the mismatch between the dynamic properties of the target and interceptor directly in the guidance law [5, 6];
- by forming guidance laws with a nonlinear (cubic) dependence on control error combinations that indirectly include the carrier's delay [5, 6];
- by transforming the input signals acting on the carrier to optimize its particular type and expand the number of corrections;
- by forming a guidance law based on predicting the target position for a time interval determined by the carrier's delay.

Note that the first three ways were discussed in detail in [5, 6].

This paper aims to consider optimization methods based on transforming input signals and using the predicted spatial position of the target.

2. OPTIMIZATION OF GUIDANCE METHODS BASED ON INPUT SIGNAL TRANSFORMATION

Here, the core consists in artificially increasing the number of corrective input signals that compensate for delay, to a greater or lesser extent, for an interceptor with particular flight characteristics.

Such a framework requires changes only in information support algorithms, without affecting the aircraft interceptor.

Note that the required transformation of input signals for guidance system optimization can be obtained by using STOC algorithms [1, 3].

Consider the simplest form of this approach, based on local optimization [1]. For a given system (interceptor) described by

$$\dot{\mathbf{x}}_{\text{int}}(t) = \mathbf{F}_{\text{int}}\mathbf{x}_{\text{int}}(t) + \mathbf{B}_{\text{int}}\mathbf{u}(t) + \xi_{\text{int}}(t), \tag{2.1}$$

intended to track a multidimensional process (target)

$$\dot{\mathbf{x}}_{\text{tar}}(t) = \mathbf{F}_{\text{tar}}\mathbf{x}_{\text{tar}}(t) + \xi_{\text{tar}}(t) \tag{2.2}$$

under available measurements

$$\mathbf{z}(t) = \mathbf{H}\mathbf{x}(t) + \xi_{\text{z}}(t), \tag{2.3}$$

the approach yields the optimal control law

$$\mathbf{u}(t) = \mathbf{K}^{-1}\mathbf{B}_{\text{int}}\mathbf{Q}[\hat{\mathbf{x}}_{\text{tar}}(t) - \hat{\mathbf{x}}_{\text{int}}(t)] \tag{2.4}$$

in terms of the minimum value of the performance functional

$$I = M\left\{[\mathbf{x}_{\text{tar}}(t) - \mathbf{x}_{\text{int}}(t)]^{\text{T}}\mathbf{Q}[\mathbf{x}_{\text{tar}}(t) - \mathbf{x}_{\text{int}}(t)] + \int\limits_0^t \mathbf{u}^{\text{T}}(t)\mathbf{K}\mathbf{u}(t)dt\right\}. \tag{2.5}$$

Here, $\mathbf{x}_{\text{int}}$ and $\mathbf{x}_{\text{tar}}$ are the n-dimensional vectors of interceptor and target states, respectively; $\mathbf{F}_{\text{int}}$ and $\mathbf{F}_{\text{tar}}$ are the internal connection matrices of the processes (2.1) and (2.2); $\mathbf{u}$ is the r-dimensional control vector ($r \leqslant n$); $\mathbf{B}_{\text{int}}$ is the control efficiency matrix; $\mathbf{z}$ is the m-dimensional measurement vector ($m \leqslant 2n$); $\mathbf{x} = \left[\mathbf{x}_{\text{tar}}^{\text{T}}\ \mathbf{x}_{\text{int}}^{\text{T}}\right]^{\text{T}}$ is the compound vector; $\mathbf{H}$ is the connection matrix of $\mathbf{z}$

and $\mathbf{x}$; ξ_{int}, ξ_{tar}, and ξ_{mea} are the noise vectors of the states (2.1), (2.2) and measurements (2.3); $\mathbf{Q}$ and $\mathbf{K}$ are **nonnegative definite** penalty matrices for control accuracy and control costs, respectively; t denotes the current control time; $M\{\bullet\}$ is the mathematical expectation given available measurements (2.3); finally, $\hat{\mathbf{x}}_{\text{int}}$ and $\hat{\mathbf{x}}_{\text{tar}}$ are the vectors of the optimal estimates of the processes (2.1) and (2.2), respectively.

In general, the perfection of any control system (its ability to respond to input action changes) is conditioned by the number of feedback loops and the presence of corrective signals. In this case, the number of possible feedback loops is determined by the dimension of the given part (2.1), and the number of corrective signals depends on the dimension of the used input vector (2.2).

In Section 2, we assess the possibility of developing an aircraft guidance method to intensively maneuvering targets based on input signal transformation.

2.1. Control Law Design

In mathematical terms, the design problem is formulated as follows.

Consider an interceptor described by the simplified aircraft model [6]

$$\begin{aligned} &\dot{\varphi}_{\text{int}} = \omega_{\text{int}}, \quad \varphi_{\text{int}}(0) = \varphi_{\text{giv}}, \\ &\dot{\omega}_{\text{int}} = -\frac{1}{T_{\text{int}}}\omega_{\text{int}} + \frac{b}{T_{\text{int}}}u, \quad \omega_{\text{int}}(0) = \omega_{\text{giv}}, \end{aligned} \tag{2.6}$$

intended for a target with a maneuvering law

$$\varphi_{\text{tar}} = f(t), \quad \varphi_{\text{tar}}(0) = \varphi_{\text{tar0}}, \tag{2.7}$$

where $f(t)$ is a smooth, n_{tar} times differentiable function ($n_{\text{tar}} > 2$), and available measurements

$$\varphi_{\text{mea}} = \varphi_{\text{tar}} + \xi_{\text{mea}}. \tag{2.8}$$

For this interceptor, it is required to find a control law u optimizing the performance functional (2.5) and ensuring capture of the target with a complex maneuvering trajectory with changing the sign of the derivative φ_{tar} (2.7) with a miss h not exceeding an admissible threshold h_{adm}.

The expressions (2.6)–(2.8) have the following notation: φ_{tar} and φ_{int} are the target's line-of-sight (LoS) angle and the interceptor's flight direction angle, respectively, with initial values φ_{tar0} and φ_{giv}; ω_{tar} and ω_{int} are the angle rates of the target's LoS and the interceptor's flight direction, respectively, with initial values ω_{tar0} and ω_{giv}; b and T_{int} are the interceptor's gain and time constant, which characterize its dynamic properties; finally, u is the control signal. In formulas (2.6)–(2.8) and below, the time dependence of all variables is omitted for simplicity.

The geometry of the target's and interceptor's angles in the vertical plane is shown in Fig. 1.

In this figure, O_{tar} is the target's location point; $\mathbf{V}_{\text{tar}}$ and $\mathbf{V}_{\text{int}}$ are the velocity vectors of the target and interceptor; φ_{tar} is the target's LoS angle; finally, φ_{int} is the interceptor's flight direction angle.

To realize a complex trajectory of an intercepted object [4], high derivatives of the LoS angle φ_{tar} must be included in its model (2.7).

For the sake of definiteness, we represent the input action (2.7) as the fourth-order vector:

$$\begin{aligned} &\dot{\varphi}_{\text{tar}} = \omega_{\text{tar}}, \quad \varphi_{\text{tar}}(0) = \varphi_{\text{tar0}}, \\ &\dot{\omega}_{\text{tar}} = a_{\text{tar}}, \quad \omega_{\text{tar}}(0) = \omega_{\text{tar0}}, \\ &\dot{a}_{\text{tar}} = \ddot{\omega}_{\text{tar}} = j_{\text{tar}}, \quad a_{\text{tar}}(0) = a_{\text{tar0}}, \\ &\dot{j}_{\text{tar}} = \dddot{\omega}_{\text{tar}} = 0, \quad j_{\text{tar}}(0) = j_{\text{tar0}}, \end{aligned} \tag{2.9}$$

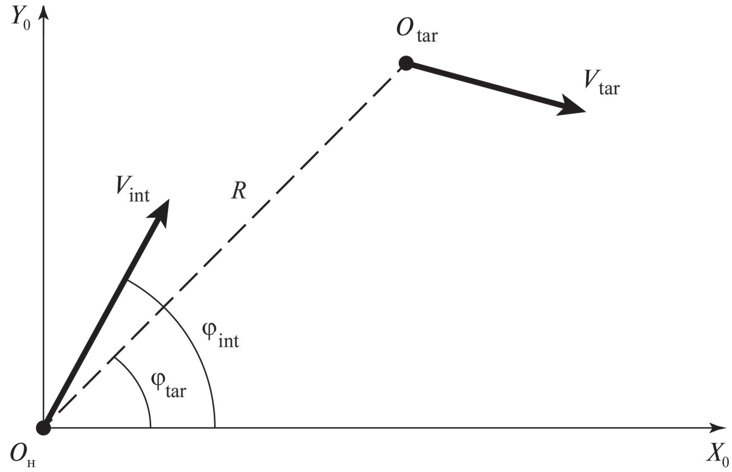

Fig. 1. The mutual arrangement of the target and interceptor in the Cartesian frame.

supplementing the model of the given part (2.6) with the two equations

$$\begin{aligned} \dot{a}_{\text{int}} &= 0, \quad a_{\text{int}}(0) = 0, \\ \dot{j}_{\text{int}} &= 0, \quad j_{\text{int}}(0) = 0. \end{aligned} \tag{2.10}$$

Associating (2.6), (2.10) and (2.9) with (2.1) and (2.2), as applied to (2.5), we obtain

$$\begin{gathered} \mathbf{x}_{\text{tar}} = \begin{bmatrix} \varphi_{\text{tar}} \\ \omega_{\text{tar}} \\ a_{\text{tar}} \\ j_{\text{tar}} \end{bmatrix}, \quad \mathbf{x}_{\text{int}} = \begin{bmatrix} \varphi_{\text{int}} \\ \omega_{\text{int}} \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{u} = u, \quad \mathbf{B}_{\text{int}} = \begin{bmatrix} 0 \\ b/T_{\text{int}} \\ 0 \\ 0 \end{bmatrix}, \\ \mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{bmatrix}, \quad \mathbf{K} = K_{\text{int}}, \end{gathered} \tag{2.11}$$

where the dimensions of the penalty matrix $\mathbf{Q}$ are extended due to increasing the dimension of the vectors $\mathbf{x}_{\text{int}}$ and $\mathbf{x}_{\text{tar}}$.

Note that the interceptor model (2.6), (2.10) does not correspond to the more complex model of the input action (2.9). Without additional measures, it will execute this action non-optimally.

Using the expressions (2.11) in (2.4), we have

$$\begin{gathered} u = K_{\text{int}}^{-1} \begin{bmatrix} 0 & \dfrac{b}{T_{\text{int}}} & 0 & 0 \end{bmatrix} \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{bmatrix} \begin{bmatrix} \hat{\varphi}_{\text{tar}} - \hat{\varphi}_{\text{int}} \\ \hat{\omega}_{\text{tar}} - \hat{\omega}_{\text{int}} \\ \hat{\dot{\omega}}_{\text{tar}} - 0 \\ \hat{\ddot{\omega}}_{\text{tar}} - 0 \end{bmatrix}, \\ u_1 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}(\hat{\varphi}_{\text{tar}} - \hat{\varphi}_{\text{int}}) + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}(\hat{\omega}_{\text{tar}} - \hat{\omega}_{\text{int}}) + \frac{bq_{23}}{K_{\text{int}}T_{\text{int}}}\hat{\dot{\omega}}_{\text{tar}} + \frac{bq_{24}}{K_{\text{int}}T_{\text{int}}}\hat{\ddot{\omega}}_{\text{tar}}. \end{gathered} \tag{2.12}$$

Direct analysis of (2.12) leads to the following conclusions.

The transformed control signal applied to the interceptor differs from the typical one derived from (2.6) [6]: it contains additional terms with the derivatives of the angle rate, $\hat{\dot{\omega}}_{\text{tar}}$ and $\hat{\ddot{\omega}}_{\text{tar}}$. The weight coefficients of the control errors are determined by the parameters (b/T_{int}) of the given part (2.6), the constraints on the control signal value (K_{int}), and the significance of the individual terms for the entire control signal ($q_{21}, q_{22}, q_{23}, q_{23}$, and q_{24}).

The control signal is represented by a set of terms considering negative feedback loops (their number is determined by the dimension of the model of the given part (2.6)) and corrections (their number depends on the complexity of representing the input actions).

To implement the desired control law, it is necessary to have the estimates $\hat{\varphi}_{\text{tar}}$, $\hat{\omega}_{\text{tar}}$, $\hat{\varphi}_{\text{int}}$, $\hat{\omega}_{\text{int}}$, $\hat{\dot{\omega}}_{\text{tar}}$, and $\hat{\ddot{\omega}}_{\text{tar}}$, which can be generated by different rules [5, 9, 10]. This issue somewhat complicates the tracking system [6] but requires no modifications of the interceptor (2.6), ensuring its optimality in terms of the minimum value of the performance functional (2.5).

Since the interceptor's control law incorporates the higher derivatives $\hat{\dot{\omega}}_{\text{tar}}$ and $\hat{\ddot{\omega}}_{\text{tar}}$, it can track targets with more complex maneuvering laws.

2.2. Analysis of the Effectiveness of the Control Law

The objective of this subsection is to assess the capabilities of the designed control law and its simplified modifications to intercept intensively maneuvering targets.

The effectiveness of the control law (2.12) was analyzed by simulating, in a forward hemisphere, the guidance procedure of the interceptor (2.6) to a target with a complex maneuvering trajectory with changing the sign of the derivatives of its angular coordinates under the perfectly accurate estimates of all the state coordinates φ_{tar}, ω_{tar}, φ_{int}, ω_{int}, $\dot{\omega}_{\text{tar}}$, and $\ddot{\omega}_{\text{tar}}$ used in (2.6).

During the simulation process, three modifications of control laws (with different degrees of interceptor's adaptation to the input actions) were compared for (2.6) as follows:

$$u_1 \text{ given by (2.12)},$$

$$u_2 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}(\hat{\varphi}_{\text{tar}} - \hat{\varphi}_{\text{int}}) + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}(\hat{\omega}_{\text{tar}} - \hat{\omega}_{\text{int}}) + \frac{bq_{23}}{K_{\text{int}}T_{\text{int}}}\hat{\dot{\omega}}_{\text{tar}}, \tag{2.13}$$

$$u_3 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}(\hat{\varphi}_{\text{tar}} - \hat{\varphi}_{\text{int}}) + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}(\hat{\omega}_{\text{tar}} - \hat{\omega}_{\text{int}}). \tag{2.14}$$

The effectiveness of guidance was evaluated by the current miss rate $h = R^2\omega_{\text{tar}}/V_{\text{o}}$, where R and V_{o} are the target's range and relative speed, respectively, under different time constants T_{int}.

Figures 2 and 3 show the trajectories of the target (T) and interceptor (I) controlled by the laws (2.12), (2.13), (2.14) (in one of the modifications) as well as the corresponding final and current miss rates. The dots indicate the positions of the target and interceptors at the same time instants.

According to the figures, the classical guidance method (2.14) does not ensure the interception of the intensively maneuvering target whereas the laws (2.12), (2.13) implement high-accuracy interception.

Based on the simulation results, we draw several conclusions:

- For the target's flight trajectory under analysis, in order to improve guidance accuracy, including $\dot{\omega}_{\text{tar}}$ in the control law (2.12) is more important than including $\ddot{\omega}_{\text{tar}}$.
- Using more inertial interceptors ($T_{\text{int2}} = 2T_{\text{int1}}$) slightly increases misses for the control laws (2.12), (2.13) and significantly increases misses for the control law (2.14).
- Similar results are observed when increasing the target's speed ($V_{\text{tar2}} = 2V_{\text{tar1}}$).

The simulation confirmed the effectiveness of the proposed guidance system optimization method based on transforming the input action in a wide range of target's speeds and interceptor's delays.

By manipulating the original models (2.7), (2.9), the penalty matrices **Q** and **K** in (2.5), (2.11), and the nature of their variation in time, one can obtain a large set of guidance methods that satisfy, to a greater or lesser extent, the requirements listed in part I of the study; for details, see [7].

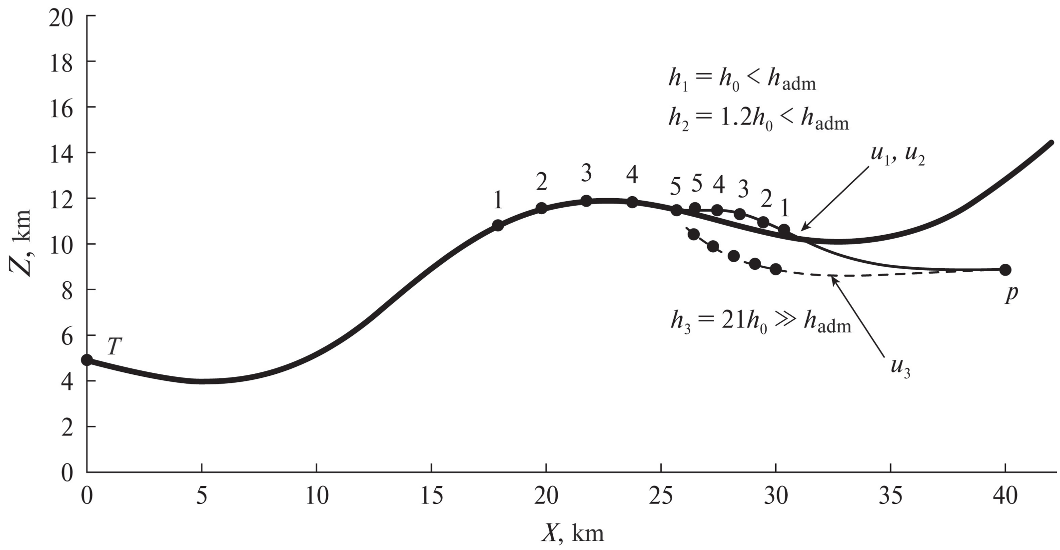

Fig. 2. The trajectories of the target and interceptors.

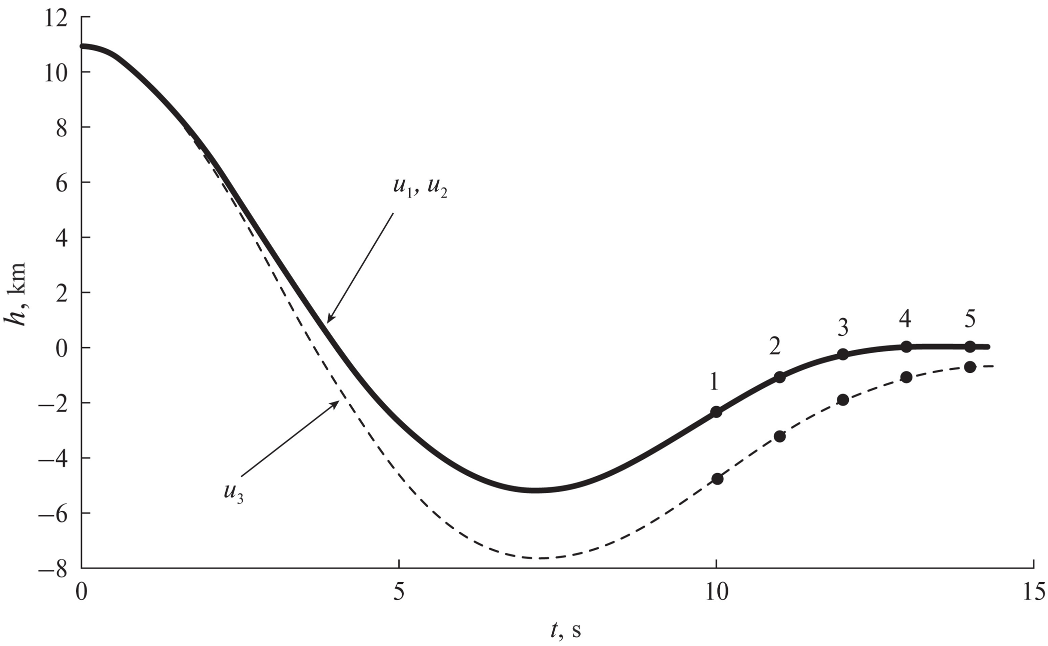

Fig. 3. Current miss rate depending on time.

As a disadvantage of this optimization method, we note the need to estimate the derivatives of the angular rate, which represents a nontrivial problem. One solution was proposed in [6].

3. GUIDANCE OPTIMIZATION BASED ON PREDICTING THE SPATIAL POSITION OF THE TARGET

In Section 2, we have discussed an optimization approach based on adapting the guidance method to a particular interceptor type, which significantly reduces the cost of developing an HSA interception system. However, another approach is also possible: adapt an interceptor to a particular guidance method without changing its dynamic properties.

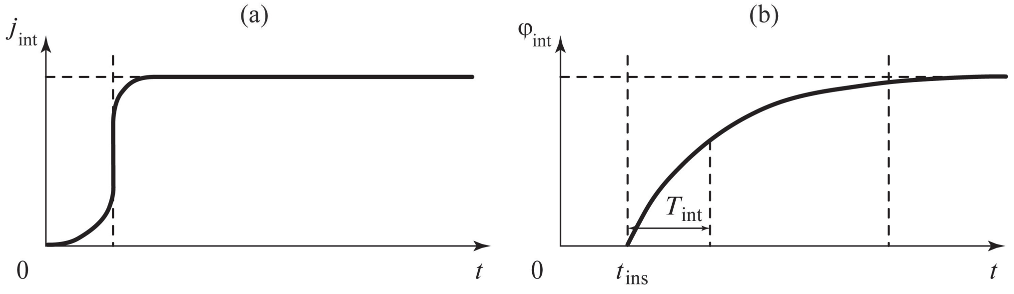

Fig. 4. (a) The input action depending on time and (b) the response to this action.

Here, the core is to compensate, by a certain method, for the delay of the interceptor's response to the control action caused by its inertia and insensitivity zone.

In Fig. 4, these features are qualitatively illustrated by the presence of the insensitivity zone t_{ins}, which determines the delay of the interceptor's response to a small control signal (see Fig. 4a), and the time constant T_{int} (see Fig. 4b) in the process of changing the interceptor's flight direction under the action of the control signal j_{int}.

The delay in the interceptor's response to a control action can be compensated for in various ways. One method is based on including the mismatch between the dynamic properties of the target and interceptor in the control law [6]. However, it is effective only at short distances to the target. In the second, control errors are formed not by the target's current position but by its prediction for a known delay in the interceptor's execution of control signals. This method can be used throughout the entire guidance procedure.

Both approaches can be (quite simply) realized within the local STOC under measured disturbances [5].

Consider the general case of this framework. For a given system (interceptor) described by

$$\dot{\mathbf{x}}_{\text{int}}(t) = \mathbf{F}_{\text{int}}\mathbf{x}_{\text{int}}(t) + \mathbf{B}_{\text{int}}\mathbf{u}(t) + \mathbf{s}_{\text{int}}(t) + \xi_{\text{int}}(t), \quad \mathbf{x}_{\text{int}}(0) = \mathbf{x}_{\text{int}0}, \tag{3.1}$$

intended to track a multidimensional process (target)

$$\dot{\mathbf{x}}_{\text{tar}}(t) = \mathbf{F}_{\text{tar}}\mathbf{x}_{\text{tar}}(t) + \xi_{\text{tar}}(t), \quad \mathbf{x}_{\text{tar}}(0) = \mathbf{x}_{\text{int}0}, \tag{3.2}$$

under available measurements

$$\mathbf{z}(t) = \mathbf{H}\left[\mathbf{x}_{\text{tar}}^{\mathrm{T}}(t) \quad \mathbf{x}_{\text{int}}^{\mathrm{T}}(t)\right]^{\mathrm{T}} + \xi_{\text{z}}(t), \tag{3.3}$$

the local STOC yields the optimal control law

$$\mathbf{u}(t) = \mathbf{K}^{-1}\mathbf{B}_{\text{int}}^{\mathrm{T}}\left[Q\left(\hat{\mathbf{x}}_{\text{tar}}(t) - \hat{\mathbf{x}}_{\text{int}}(t)\right) - \mathbf{G}\hat{\mathbf{s}}_{\text{int}}(t)\right] \tag{3.4}$$

in terms of the minimum value of the performance functional

$$I = M\left\{\Delta\mathbf{x}^{\mathrm{T}}(t)\mathbf{Q}\Delta\mathbf{x}(t) + 2\Delta\mathbf{x}^{\mathrm{T}}(t)\mathbf{G}\mathbf{s}_{\text{int}}(t) + \mathbf{s}_{\text{int}}^{\mathrm{T}}(t)\mathbf{Q}\mathbf{s}_{\text{int}}(t) + \int_0^t \mathbf{u}^{\mathrm{T}}(t)\mathbf{K}\mathbf{u}(t)dt\right\}, \tag{3.5}$$

$$\Delta\mathbf{x}(t) = \mathbf{x}_{\text{tar}}(t) - \mathbf{x}_{\text{int}}(t). \tag{3.6}$$

Here, $\mathbf{s}_{\text{int}}$ and $\hat{\mathbf{s}}_{\text{int}}$ are the n-dimensional vectors of measured disturbances and their optimal estimates; $\mathbf{G}$ is a nonnegative definite penalty matrix of the disturbances.

The objective of Section 3 is to assess the possibility of using the apparatus (3.1)–(3.6) to develop HSA interception methods.

3.1. Control Law Design

In mathematical terms, the design problem is formulated as follows. For a system described by

$$\begin{aligned} &\dot{\varphi}_{\text{int}}(t) = \omega_{\text{int}}(t), && \varphi_{\text{int}}(0) = \varphi_{\text{int}0}, \\ &\dot{\omega}_{\text{int}}(t) = -\frac{1}{T_{\text{int}}}\omega_{\text{int}}(t) + \frac{b}{T_{\text{int}}}j_{\text{int}}(t), && \omega_{\text{int}}(0) = \omega_{\text{int}0}, \end{aligned} \tag{3.7}$$

intended to intercept an HSA by predicting its spatial position for a time T_{pre} :

$$\begin{aligned} &\dot{\varphi}_{\text{tar}}(t+T_{\text{pre}}) = \omega_{\text{tar}}(t) + \dot{\omega}_{\text{tar}}(t)T_{\text{pre}}, \qquad \varphi_{\text{tar}}(0) = \varphi_{\text{tar}}(T_{\text{pre}}), \\ &\dot{\omega}_{\text{tar}}(t+T_{\text{pre}}) = -\frac{2\dot{R}(t)}{R(t)}\omega_{\text{tar}}(t) + \frac{1}{R(t)}(j_{\text{tar}}(t) - j_{\text{int}}(t)) + \ddot{\omega}_{\text{tar}}(t)T_{\text{pre}}, \\ &\qquad\qquad \omega_{\text{tar}}(0) = \omega_{\text{tar}}(T_{\text{pre}}), \end{aligned} \tag{3.8}$$

it is required to find a control signal j_{int} optimizing the performance functional

$$\begin{aligned} I = M\Bigg\{ &\begin{bmatrix} \Delta\varphi \\ \Delta\omega \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} \Delta\varphi \\ \Delta\omega \end{bmatrix} + 2\begin{bmatrix} \Delta\varphi \\ \Delta\omega \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \\ &\times \begin{bmatrix} \dot{\omega}_{\text{tar}}T_{\text{pre}} \\ \left(\dfrac{1}{T_{\text{int}}} - \dfrac{2\dot{R}}{R}\right)\omega_{\text{tar}} + \ddot{\omega}_{\text{tar}}T_{\text{pre}} \end{bmatrix} + \begin{bmatrix} \dot{\omega}_{\text{tar}}T_{\text{pre}} \\ \left(\dfrac{1}{T_{\text{int}}} - \dfrac{2\dot{R}}{R}\right)\omega_{\text{tar}} + \ddot{\omega}_{\text{tar}}T_{\text{pre}} \end{bmatrix}^{\mathrm{T}} \\ &\times \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} \dot{\omega}_{\text{tar}}T_{\text{pre}} \\ \left(\dfrac{1}{T_{\text{int}}} - \dfrac{2\dot{R}}{R}\right)\omega_{\text{tar}} + \ddot{\omega}_{\text{tar}}T_{\text{pre}} \end{bmatrix} + \int_0^t j_{\text{int}}^2 K_{\text{int}}\, dt \Bigg\}. \end{aligned} \tag{3.9}$$

The geometry of the target's and interceptor's angles is shown in Fig. 3.1.

In formula (3.9) and below, the time dependence of all variables is omitted for simplicity.

Using (3.7) and (3.8) in (3.6), we derive the system of equations

$$\begin{aligned} &\Delta\dot{\varphi} = \dot{\varphi}_{\text{tar}} - \dot{\varphi}_{\text{int}} = \Delta\omega + \dot{\omega}_{\text{tar}}T_{\text{pre}}, \\ &\Delta\dot{\omega} = \dot{\omega}_{\text{tar}} - \dot{\omega}_{\text{int}} = -\frac{1}{T_{\text{int}}}\Delta\omega - \frac{b}{T_{\text{int}}}j_{\text{int}} + \frac{1}{T_{\text{int}}}\omega_{\text{tar}} + \dot{\omega}_{\text{tar}} + \ddot{\omega}_{\text{tar}}T_{\text{pre}}. \end{aligned} \tag{3.10}$$

Therefore, the mismatch between the dynamic properties of the target and interceptor (acting as a measured disturbance) is given by the vector [5]

$$\mathbf{s}_{\text{int}} = \begin{bmatrix} \dot{\omega}_{\text{tar}}T_{\text{pre}} \\ \left(\dfrac{1}{T_{\text{int}}} - \dfrac{2\dot{R}}{R}\right)\omega_{\text{tar}} + \ddot{\omega}_{\text{tar}}T_{\text{pre}} \end{bmatrix}. \tag{3.11}$$

Using (3.10) and (3.11) in (3.4), we obtain

$$\begin{aligned} &j_{\text{int}} = \frac{1}{K_{\text{int}}}\begin{bmatrix} 0 & \dfrac{b}{T_{\text{int}}} \end{bmatrix} \left\{ \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} \Delta\hat{\varphi} \\ \Delta\hat{\omega} \end{bmatrix} - \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \left[\begin{matrix} \hat{\dot{\omega}}_{\text{tar}}T_{\text{pre}} \\ \left(\dfrac{1}{T_{\text{int}}} - \dfrac{2\hat{\dot{R}}}{\hat{R}}\right)\hat{\omega}_{\text{tar}} + \hat{\ddot{\omega}}_{\text{tar}}T_{\text{pre}} \end{matrix} \right] \right\}, \\ &j_{\text{int}} = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\varphi} + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\omega} + \frac{bg_{22}}{K_{\text{int}}T_{\text{int}}}\left(\frac{1}{T_{\text{int}}} - \frac{2\hat{\dot{R}}}{\hat{R}}\right)\hat{\omega}_{\text{tar}} \\ &\qquad + \frac{bg_{21}}{K_{\text{int}}T_{\text{int}}}\hat{\dot{\omega}}_{\text{tar}}T_{\text{pre}} + \frac{bg_{22}}{K_{\text{int}}T_{\text{int}}}\hat{\ddot{\omega}}_{\text{tar}}T_{\text{pre}}. \end{aligned} \tag{3.12}$$

Direct analysis of (3.12) leads to the following conclusions.

1. If $V_{\text{int}} > V_{\text{tar}}$, this method is all-sector and all-altitude. All-sector interception is provided by considering the signs of angular errors, the LoS and angle rate, and its derivatives. All-altitude interception is provided by using transverse acceleration as a control signal (instead of rudders, whose effectiveness depends on air density (altitude)).

2. Including the derivatives $\dot{\omega}_{\text{tar}}$ and $\ddot{\omega}_{\text{tar}}$ of the LoS angle rate in the control law allows guiding to intensively maneuvering targets.

3. The significance of the last three terms grows with approaching the target and increasing the forecast time.

4. To implement this guidance law, one needs optimal estimates of the range, approach speed, relative bearing, LoS angle rate, and its derivatives [5, 10, 11].

3.2. Analysis of the Effectiveness of the Control Law

The objective of this subsection is to assess the capabilities of the control law (3.12) and its simplified modifications to intercept intensively maneuvering targets.

The effectiveness of this law was analyzed by simulating, in a forward hemisphere, the guidance procedure of the interceptor (3.7) to the target (3.8) with a complex maneuvering trajectory with changing the sign of the derivatives of its angular coordinates under the perfectly accurate estimates of all the state coordinates and $V_{\text{int}} < V_{\text{tar}}$.

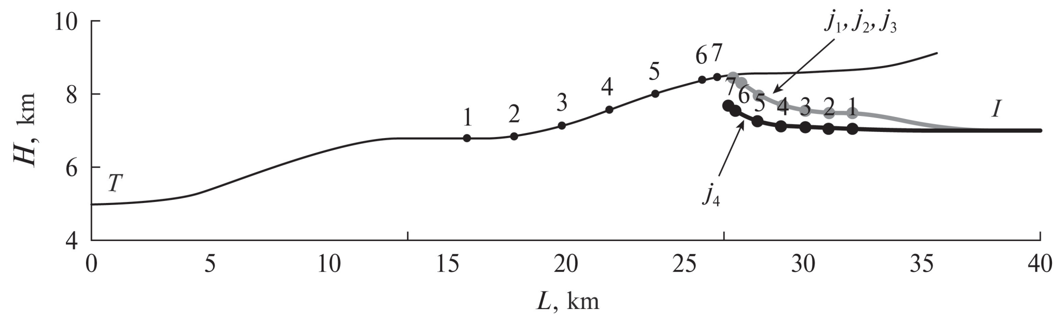

Fig. 5. The trajectories of the target and interceptors with different guidance laws.

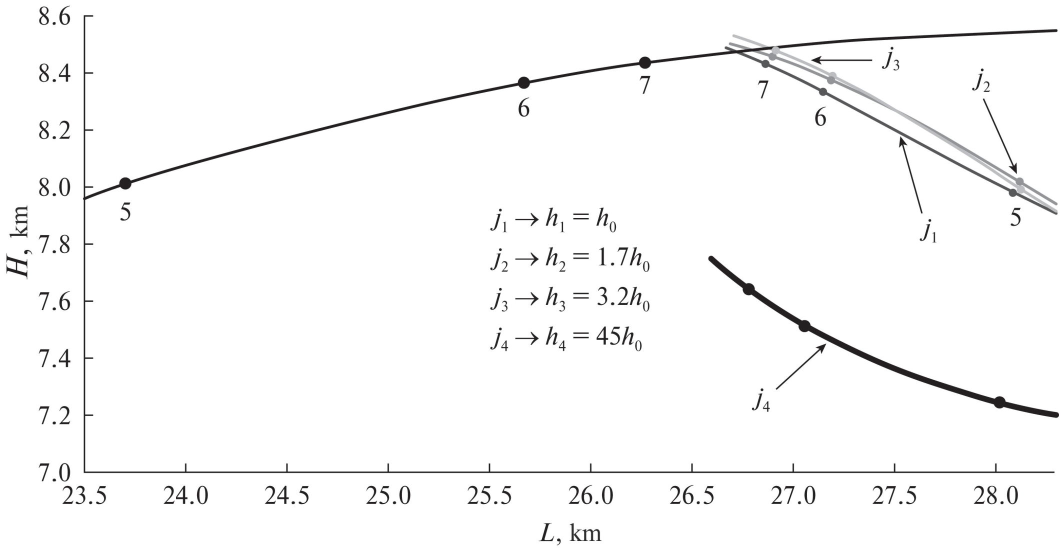

Fig. 6. The final section of the target's and interceptor's trajectories.

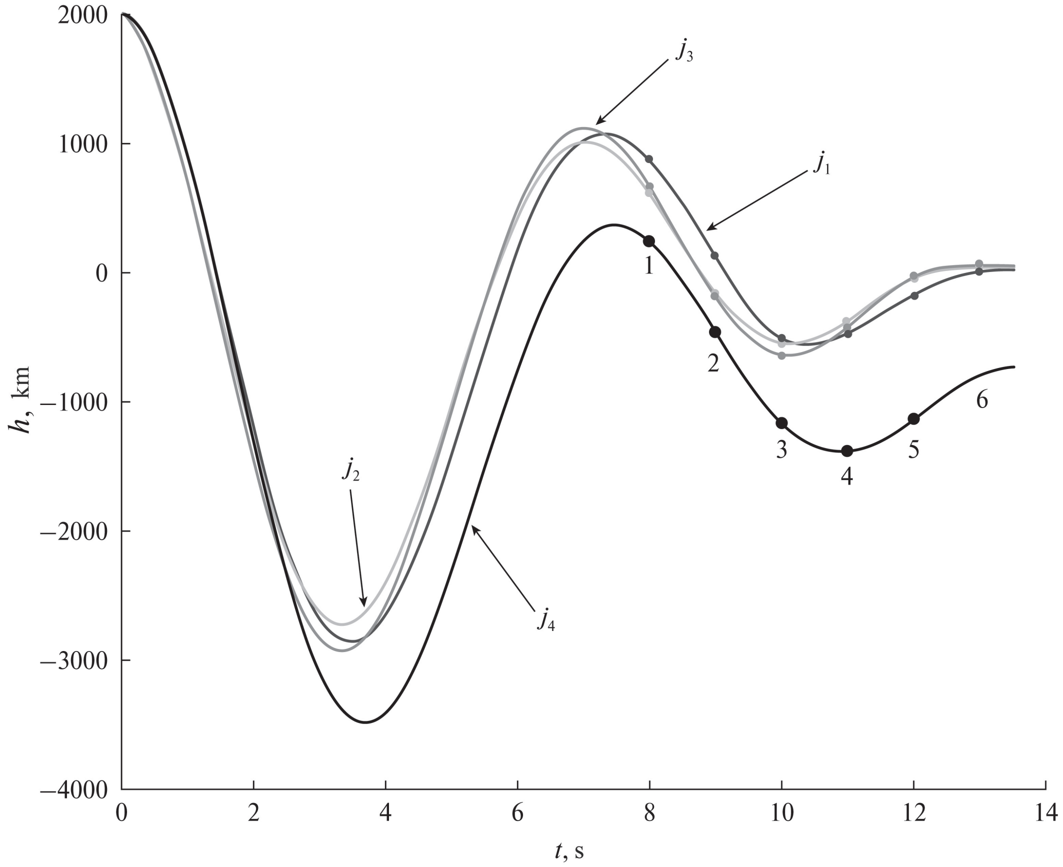

Fig. 7. The current miss rates of interceptors.

During the simulation process, four modifications of control laws were compared as follows:

$$j_1 \text{ given by (3.12)},$$

$$j_2 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\varphi} + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\omega} + \frac{bg_{22}}{K_{\text{int}}T_{\text{int}}}\left(\frac{1}{T_{\text{int}}} - \frac{2\hat{\dot{R}}}{\hat{R}}\right)\hat{\omega}_{\text{tar}} + \frac{bg_{21}}{K_{\text{int}}T_{\text{int}}}\hat{\dot{\omega}}_{\text{tar}}T_{\text{pre}},$$

$$j_3 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\varphi} + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\omega} + \frac{bg_{22}}{K_{\text{int}}T_{\text{int}}}\left(\frac{1}{T_{\text{int}}} - \frac{2\hat{\dot{R}}}{\hat{R}}\right)\hat{\omega}_{\text{tar}},$$

$$j_4 = \frac{bq_{21}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\varphi} + \frac{bq_{22}}{K_{\text{int}}T_{\text{int}}}\Delta\hat{\omega}.$$

The effectiveness of guidance was evaluated by the current and final miss rates and the control signal value under different time constants T_{int} and prediction times T_{pre}.

Figures 5 and 6 show the trajectories of the target (T) and interceptors (I) with the control laws j_1–j_4; points 1–5 indicates their current positions. Figures 7 and 8 present the corresponding miss rates and accelerations.

Based on the simulation results, we draw several conclusions:

1. The control laws with the signals j_1–j_3 realize a sufficiently high accuracy, which decreases with excluding individual terms. The conventional guidance method with the signal j_4 does not ensure the interception of an intensively maneuvering HSA (with changing the sign of the derivatives of its angular coordinates).

2. Guidance accuracy deteriorates when increasing the interceptor's delay.

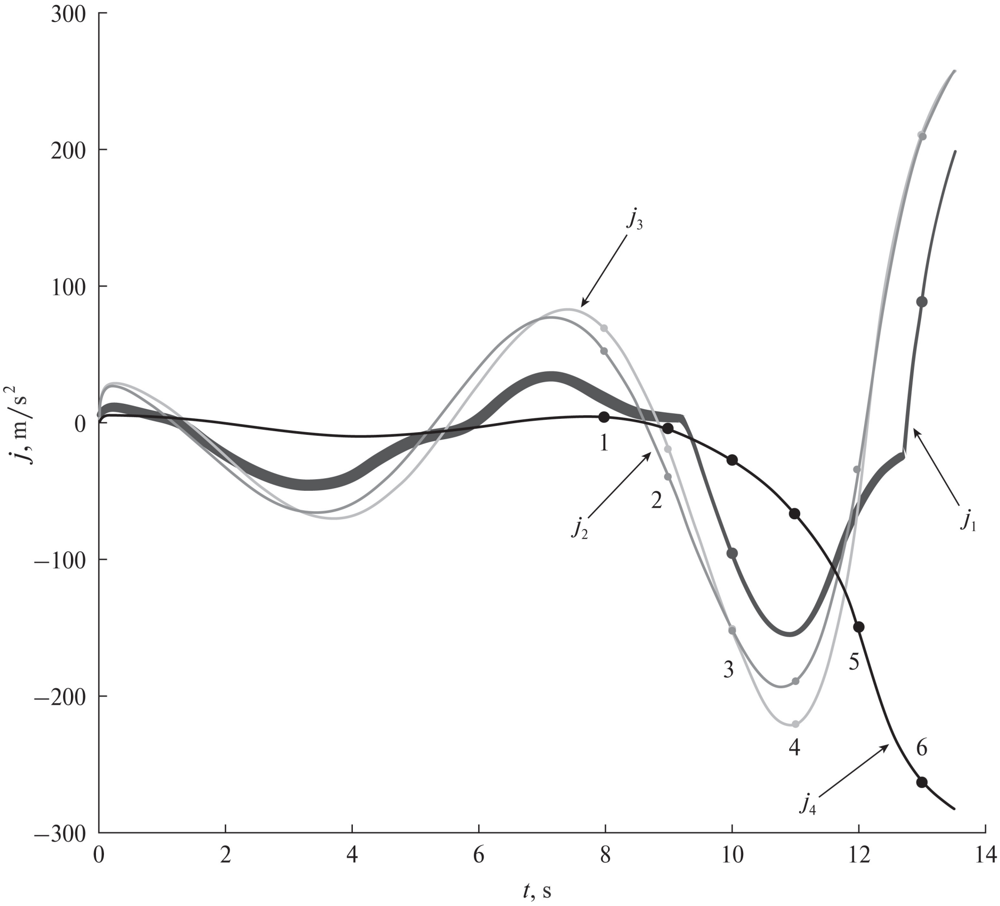

Fig. 8. The graphs of control signals.

3. Increasing the prediction time to $T_{\text{pre}} = 2T_{\text{int}}$ improves guidance accuracy.

4. Guidance is performed within the realizable transverse accelerations.

As a disadvantage, we note the complexity of information support due to the need to estimate the derivatives of the LoS angle rate.

4. CONCLUSIONS

The material presented in this paper leads to the following findings.

The mathematical apparatus of local STOC optimization can be used to develop guidance methods to HSA considering the carrier's dynamic properties. The problem can be solved by at least two approaches.

The first is based on transforming input actions. It provides adaptation of the guidance method and its information support to a particular carrier type.

Within the second approach, the control signal is formed by predicting the target's position for a time determined by the carrier's inertia (instead of the current state of the interceptor and target). Both approaches do not require redesigning the carrier to improve its maneuverability, which significantly reduces the cost of developing an interceptor system.

Both approaches ensure the interception of an intensively maneuvering HSA by including the derivatives of the angle rate in guidance methods. This will require complicating somehow the angular measurement channel of the onboard radar system.

REFERENCES

1. *Aviatsionnye sistemy radioupravleniya* (Aircraft Radio Control Systems), Merkulov, V.I., Ed., Moscow: Zhukovsky Air Force Engineering Academy, 2008.
2. Girard, A.R. and Kabamba, P.T., Proportional Navigation: Optimal Homing and Optimal Evasion, *SIAM Review*, 2015, vol. 57, no. 4, pp. 611–624.
3. An, JY., Lee, CH., and Tahk, MJ., A Collision Geometry-Based Guidance Law for Course-Correction-Projectile, *Int. J. Aeronaut. Space Sci.*, 2019, vol. 20, no. 2, pp. 442–458.
4. Verba, V.S., Merkulov, V.I., Zakomoldin, D.V., and Likhachev, V.P., Problems of Interception of High Speed Aircraft Maneuvering According to Complex Laws. Part 2. Analysis of Flight Paths of Foreign Aircraft, *Achievements of Modern Radioelectronics*, 2024, vol. 78, no. 4, pp. 5–14.
5. Merkulov, V.I. and Verba, V.S., *Sintez i analiz aviatsionnykh radioelektronnykh sistem upravleniya. Kniga 1* (Design and Analysis of Aircraft Radioelectronic Control Systems. Book 1), Moscow: Radiotekhnika, 2023.
6. Merkulov, V.I. and Verba, V.S., *Sintez i analiz aviatsionnykh radioelektronnykh sistem upravleniya. Kniga 2* (Design and Analysis of Aircraft Radioelectronic Control Systems. Book 2), Moscow: Radiotekhnika, 2023.
7. Verba, V.S. and Merkulov, V.I., Approaches to Optimizing Guidance Methods to High-Speed Intensively Maneuvering Targets. Part I. Justifying Requirements for Ways to Optimize Guidance Methods, *Autom. Remote Control*, 2024, vol. 85, no. 11, pp. 1108–1112.
8. Galyaev, A.A., Lysenko, P.V., and Rubinovich, E.Y., Optimal Stochastic Control in the Interception Problem of a Randomly Tracking Vehicle, *Mathematics*, 2021, vol. 9, no. 19, art. no. 2386.
9. Su, W., Li, K., and Chen, L., Coverage-Based Cooperative Guidance Strategy against Highly Maneuvering Target, *Aerospace Science and Technology*, 2017, vol. 71, pp. 147–155.
10. Song, F., Y. Li, Y., Cheng, W., et al., An Improved Kalman Filter Based on Long Short-Memory Recurrent Neural Network for Nonlinear Radar Target Tracking, *Wireless Communications and Mobile Computing*, 2022, no. 7, pp. 1–10.
11. Prokhorov, M.B. and Saul'ev, V.K., The Kalman–Bucy Method of Optimal Filtering and Its Generalizations, *Journal of Soviet Mathematics*, 1979, vol. 12, no. 3, pp. 354–380.

This paper was recommended for publication by A.A. Galyaev, a member of the Editorial Board

ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 2, pp. 165–190.

Visual Servoing for Deformable Objects with Pre-Planned Trajectory-Guided Geometric Primitives

D. D. Yasser*,a, S. S. Farah*,b, and M. M. Abed,c**

*LAT, Laboratoire d'Automatique de Tlemcen, Universit de Tlemcen, Tlemcen, 13000, Algeria
**GBM, Laboratoire de Genie Biomedical, Universit de Tlemcen, Tlemcen, 13000, Algeria
e-mail: [a] derrar.yasser@gmail.com, [b] saidifarah.mimouni@gmail.com, [c] abed.malti@gmail.com

Received February 13, 2024
Revised August 15, 2023
Accepted August 26, 2024

Abstract—This paper presents a novel technique for the rigid alignment of a perspective camera to the deformations of a non-rigid object, using an adaptive visual servoing approach. Unlike existing methods, the proposed approach does not rely on any parametric or model-based priors regarding the deformation. Assuming the availability of a pre-planned camera trajectory observing the non-rigid object, the method aims to align this trajectory during the execution phase, utilizing only the most relevant landmarks as prior information. Importantly, the approach does not depend on any parametric or non-parametric models of the underlying deformation physics. Instead, it is formulated as a tracking problem embedded within an optimal visual control framework. This tracking process involves the visual servoing of geometric features from the deformable object, bridging the gap between the planning and execution stages. The optimal visual control is defined using a weighted least-squares criterion, which minimizes the distance between the reference features and those observed in real-time. The weights are time-dependent, smooth functions that encode the relevance of the visible object features. The experimental results demonstrate the method?s ability to adapt to various pre-planned trajectories and different types of deformations, without requiring prior knowledge, while also exhibiting resilience to noise in the detection of image features.

Keywords: visual servoing, deforming objects, motion planning, trajectory tracking

DOI: 10.31857/S0005117925020057

1. INTRODUCTION

Visual servoing refers to the use of computer vision data, acquired from one or more cameras, to control their movement. Unlike vision-based tracking problems with rigid objects, for which a certain maturity has been reached, tracking by visual servoing on non-rigid objects is still a challenging problem. It has aroused much interest in recent years in the computer vision and robotics communities [1]. Many potential applications are targeted, in fields such as augmented reality, medical imaging, robotic manipulation, by handling a huge variety of objects: tissues, paper, rubber, viscous fluids, cables, food, organs, etc [2].

One characteristic of deformable object tracking is that the object shape changes while being tracked. Recent methods proposed a hybrid visual servoing decoupling the translational velocity in Z-axis and three rotational velocities. Such method has improved the performance of classical HVS (Hybrid Visual Servoing) in both image-plane and task space [3].

However, decoupling camera degrees of freedom put restrictions on the correction of camera motion and can lead to non-smooth tracking of the object's deformation. Tracking the geometry of deformable objects such as rope and cloth is difficult due to the continuous nature of the object

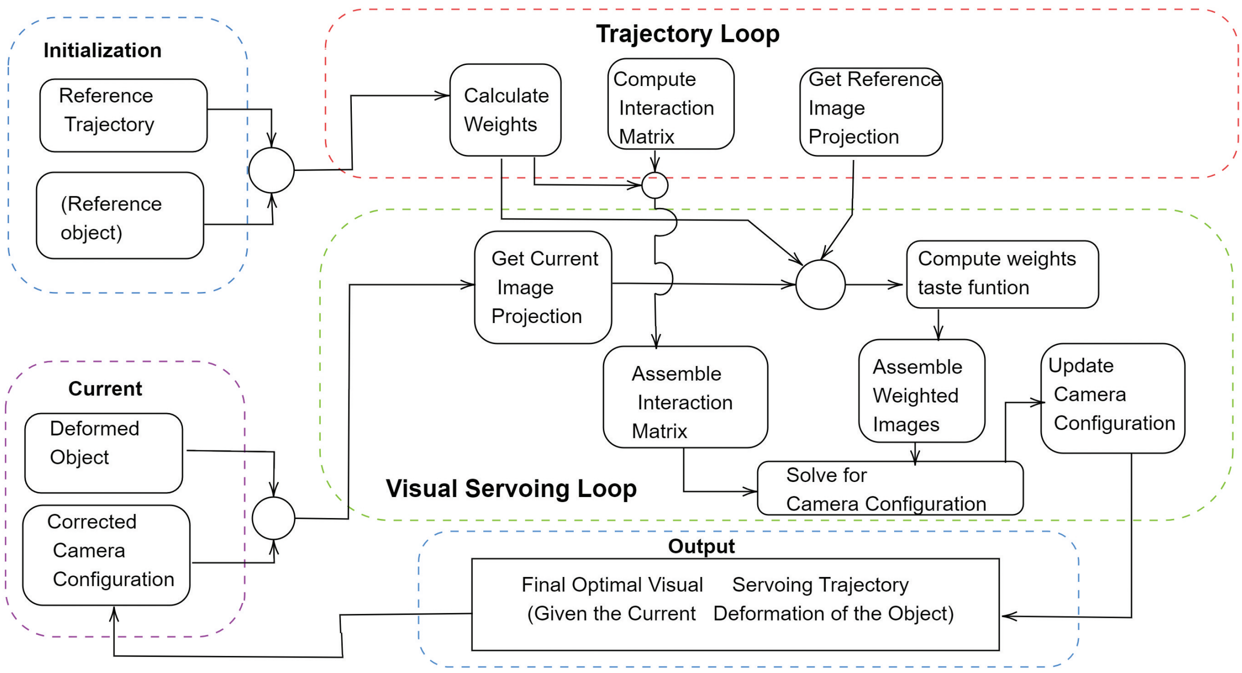

Fig. 1. Flowchart of the triangle-mesh-based visual servoing algorithm for deformable objects.

(i.e. an infinite number of degrees of freedom) [4, 5]. Most current method on tracking by visual servoing requires both the knowledge of physical properties of the object and real-time monitoring of the deformations. Several studies have focused on the monitoring of deformations in real time but not simultaneously with the control of the deformation [6]. These methods are highly parametric and require specific fine tuning that is strongly dependent on the physics of deformations. In this work, we propose a robust approach to track object's deformation by an optimal visual servoing. We assume having a pre-planned reference camera trajectory viewing the targeted object. We assume that this object is described within a mesh composed of set of triangular geometric primitives. We do not assume any prior knowledge or information on the type or physics of deformation. We used the liver as a reference deformable object. The addressed problem is stated as follows: Let us consider a camera following a reference trajectory and viewing a liver. If the latter is deformed, then how should one at best adapt the trajectory of the former. In the context of robotized laparoscopy, this is a standard problem and is usually put at the responsibility of an operator who manipulates the camera (the laparoscope) with a joystick according to the surgeon recommendation. This task can be fully automatized given the resolution of two main issues: First, an appropriate liver tracking which allows one to follow $2D$ features on the liver. Second, an automation criterion that allows one to adapt to deformations and modify the camera trajectory accordingly.

This paper focuses on proposing a novel strategy to tackle the second problem.

This diagram 1 illustrates the flow of a triangle-mesh-based visual servoing algorithm used for controlling a camera's motion when observing a deformable object. The process begins by initializing the reference trajectory and mesh data, followed by iterating through each point on the trajectory. For each point, the algorithm processes each triangle in the object's mesh, calculating key parameters such as weights and interaction matrices. A visual servoing loop then iteratively adjusts the camera's position and orientation to minimize the error between the current image of the deformed object and the reference image, ultimately producing an optimized trajectory that accounts for the object's deformation.

The contributions of our work can be summarized as follows

(1) Define a camera-based motion planning to compensate non-rigid object deformations with rigid camera motion. This definition allows us to formalize the viewing interaction between the deformable object and the camera with a set of camera-triangle primitives. These triangles are the geometric elements of the object's mesh.
(2) Associate a continuous weight function to every pair of camera-triangle primitive. The continuity is along the camera-trajectory and allows us to ensure smooth tracking.
(3) Solve the visual servoing and tracking task in an optimal visual control framework. This proposed method does not assume any prior on the object's deformation. The aforementioned weight functions encode the relevance of the camera-triangle primitives. The most important target related primitives have higher weights and allows us to find the optimal camera motion to compensate non-rigid deformations.
(4) Experimental validation demonstrates the robustness of the approach regarding the noise in image detection, the variability in trajectories and type of deformations.

This paper is organized as follows. Section 2 presents the related state-of-the-art works. Section 3 introduces the fundamental tools for our modeling. Section 4 describes the visual servoing with optimal visual control and tracking. Section 5 presents the experimental results and comments. Section 6 concludes the paper and draws a sketch of future works.

2. RELATED WORK

Tracking and visual servoing on non rigid objects are recent open problems. Many approaches propose to use physics-based or data-based mechanical models embedded into the classic algorithms. These methods are highly parametric and are fine-tuned for every specific type of deformable objects. In this paper, we propose a generic approach that enables rigid camera visual servoing to fit as much as possible visual features of objects subjected to non-rigid deformation. In this section, we review the literature on visual servoing, visual tracking and image alignment.

2.1. Visual Servoing on Rigid Objects

Classic visual servoing is a method of controlling the movement of a robot using real-time feedback from vision sensors [7–9]. Visual servoing control has two basic approaches: Position-based visual servoing and image-based visual servoing [10, 11]. In this paper, we use the position-based visual control since we assume tracking $2D$ features on the target deformable object. The relationship between the camera and the object is represented by an interaction matrix. This matrix can be determined in the image space by points, lines or ellipses and moments [12, 13]. In this work, point-based interaction matrix is used. It is used as an elementary building block for triangle-based interaction matrices. The triangles being the primitives that compose the surface of the deformable shape. Being an old problem, the literature of visual servoing on rigid objects is extensive and cannot be covered in this paragraph. We cover here, some main results of these works. Janabi et al. [14] used a reduced set of holes, circles and wedges as characteristics for their availability in many industrial parts and for their easy and robust extraction. Chaumette [15] used the moments calculated by image segmentation to determine the analytical shape of the interaction matrix. Molnar et al. [16] offered a marker-based visual servo technique for automated camera positioning in the case of Robot-Assist minimally invasive surgery. Their approach is to keep the followed surgical instrument within the limits of the camera image. This is done by constantly adjusting the endoscopic camera manipulation pose instead of keeping the instrument in the center of the image. In addition, the interaction matrix can be determined by a hybrid visual servoing which combines image-based visual servoing and position-based visual servoing. Iheb [17] proposed a real-time and inversion-free control strategy on the basis of sampling-based model predictive control algorithm for both image-based and position-based visual techniques. They considered system constraints and uncertainties parameters associated with the robot and the camera measurement. In this paper, we do not consider model uncertainties and we do not use a predictive control scheme. Instead, we use an optimal visual control framework to compensate for both noise and deformations. We propose to use a continuous weight functions on the triangle primitive to account for their relevance while tracking the reference trajectory.

2.2. Visual Servoing on Non-Rigid Objects

Robotic manipulation of non-rigid objects is a difficult task due to the deformations that occur which add several degrees of freedom to the initial problem with rigid objects. Among deformations we can cite shearing, scaling, stretching, torsion, compression, etc. Human organs are relevant cases of non-rigid objects with a variety of shapes and deforming behaviors [18]. Using such objects with visual servoing approaches gives solutions to automate problems in medical and surgery contexts. For instance, augmented reality to overlay MRI (Magnetic Resonance Imaging) images on live laparoscopic's stream [2], robotized laparoscopy to guide the laparoscope, etc. Hu et al. [19] presented a controller based on a deep neural network to control the position and shape of deformable objects with unknown deformation properties. They treated the nonlinear properties of the deformable objects and used a multilayer neural network to model the mapping function. In this work, we do not use any prior model on the deformation of the organ. We use a a set of continuous weight functions that allows us to account for relevant regions to focus on. Jia et al. [20] presented a histogram of oriented wrinkles to describe the variation in shape of a highly deformable object. The characteristics of the deformable object are calculated by applying Gabor filters and extracting the high and low frequency components. They precomputed the visual feedback using an offline training phase that stores a correspondence between these visual characteristics and the speed of the end effector. Our approach uses an optimal visual control setup which offers in-line camera correction without requiring to acquire data or train neural networks. Hu et al. [21] designed a servo algorithm that can learn a nonlinear deformation function along with the manipulation process. They used Gaussian process regression to model and learn the deformation parameters of a soft object. In this paper, we propose a generic approach that enables rigid camera visual servoing to adapt as much as possible visual features of objects subjected to non-rigid deformation. The proposed method uses a triangular mesh representation of the visualized object. It computes a relative camera rigid repositioning which fit the most relevant triangular primitives without fully ignoring the remainders.

2.3. Visual Tracking of Non-Rigid Objects

Tracking deformation has been studied in several works [22–25]. Chen et al. [26] presented a two-step object tracking method. They used the kernel-based method to efficiently locate the object under complex conditions with camera movement. For precision, they improved the tracking accuracy by using the contour-based method to follow the object's contour precisely after locating the target. Cao et al. [27] assumed that the extension of an object would deform from a reference by moving certain control points of the latter towards those of the old one. In our method, we do not consider any model of the deformed object. Our motion compensation is based on relevant image feature differences. Joo et al. [28] presented a generative body deformation model which has the capacity to express the movement of every principal part of the object. Royer et al [29] presented a novel approach for tracking a deformable anatomical target within 3D ultrasound volumes. This method is able to estimate deformations caused by the physiological motions of the patient. The displacements of moving structures are estimated from an intensity-based approach combined with a physically-based model and has therefore the advantage to be less sensitive to the image noise. Our method does not process a 3D reconstruction of the object and focuses only on the best move of the camera to compensate deformations and ultimately noise. Royer et al [30] extended their former work by proposing a 3D tracking method which is regularized by a statistical motion model obtained from biomechanical modeling. However, their method requires manually identifying certain points of the target object (here a prostate) on each ultrasound frame in order to drive the model. Kajihara et al. [31] developed a system that tracks the trajectory of a line drawn on a flexible object. They estimated the tracking performance with speed and precision in the presence of many uncertainties based on dynamic compensation. In the proposed approach, there is no pattern that is drawn on the target object. Instead, a set of weight functions are pre-defined on pre-selected regions of the deformable objects. These regions are defined within a set of object's mesh triangles.

2.4. Image Alignment

Image alignment is relative to the alignment of two or more images of the same scene on a common spatial axis. It plays an important role in computer vision and graphics, such as structure from motion, 3D reconstruction, motion tracking and recovery [32, 33]. The problem we are addressing can be seen as an image alignment of a deformable scene. Indeed, the camera motion compensation that is calculated by our method can also be used to align the current image with the reference image from the pre-defined trajectory. Shingo et al. [34] proposed a technique to align an end-edge of flexible sheet object to a specified line segment in 3D space. They made online estimation for a relative pose between the end of the edge and the robot hand in the visual servoing control laws. Our method relies on feature points instead of line features. Xi et al. [35] focused on parametric and non parametric alignment method which have complementary strength. They proposed a feature based parametric alignment using one or more homographies followed by non-parametric fine pixel wise alignment. Our method is non-parametric regarding to the deformation of the object. Dong et al. [36] proposed the spectral-spatial weighted kernel manifold embedded distribution alignment method for the classification of remote sensing images. They used a filter to express the mean spectrum of the neighboring pixel samples from each pixel sample. In this work, we consider the 2D feature points as being known. Mathiassen et al. [37] proposed a visual servoing method to move the ultrasound probe, using a robot to align the image plane of the probe with the needle. The method segments the needle and updates a set of visual features based on a model of the needle. A state machine is used to keep track of the alignment process, and different visual features are used to control the probe in the different states. In this paper, we consider a perspective camera model. The proposed method can be applied to ultrasound sensors however it will be necessary to modify the projection model that is used in this paper. Mura et al. [38] presented a vision-based haptic feedback system with the aim to assist the movement of an endoscopic device during capsule endoscopy guidance. It allows the user to control the movement of the capsule along the generated path. The haptic module also helps the operator by transforming the 3D maps and the relative paths into a guiding virtual force. Measuring the current relative distance between the user input and the maps boundaries, the haptic guidance module will check if the user is moving away or toward the colonic walls and will generate a feedback force with the aim to assist the operator during the navigation procedure. The user will also sense an attractive virtual feedback force toward the generated path that will help the user in the navigation. Our method can be applied to feed the haptic module with the necessary correction. Indeed, instead of compensating the trajectory of the camera, one correct the trajectory of the capsule by controlling the haptic joystick.

3. VISUAL PRIMITIVES AND INTERACTION MODELING

3.1. Definitions

3.1.1. Triangle. Let us consider a triangle T as a basic geometric primitive composed of 3 points in the 3D workspace. Let $L = \mathbf{R}^9$ be the configuration space of this primitive. We denote by

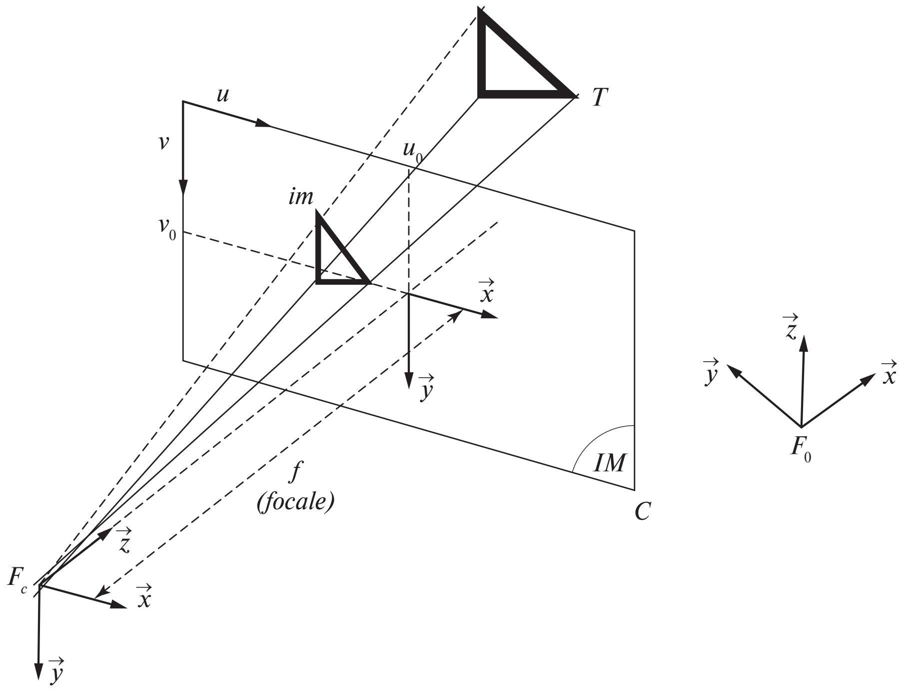

Fig. 2. Perspective projection of a triangle on the image plane of a camera.

$l = (X_1, X_2, X_3, Y_1, Y_2, Y_3, Z_1, Z_2, Z_3)^\top$ the configuration of T in L. Where $(X_i, Y_i, Z_i)^\top$, $1 \leqslant i \leqslant 3$ being the coordinates of the three vertices composing triangle T. Index i represents the number of the vertex in the considered triangle.

3.1.2. Camera. Let us consider C as a camera that maps one or several triangular primitives to a 2D feature in the image space. Let $\mathcal{C} = \mathbf{SE}(3)$ be the configuration space of this sensor. We denote by c the configuration of the camera in the space $\mathcal{C}$.

3.1.3. Sensing a triangle. The perception of a triangle T by a camera C can be characterized by a continuous mapping:

$$\Pi : \mathcal{C} \times \mathcal{T} \to I_{c,l}, \tag{1}$$

$$(c, l) \mapsto \Pi(c, l). \tag{2}$$

That associates to each configuration of the camera and the triangle a feature in the image space $I_{c,l} \subset \mathbf{R}^6$. The image space being composed of the projection of the triplet of points describing the triangle of interest (see Fig. 2). Remark that Π is usually defined over a subset of $\mathcal{C} \times \mathcal{T}$ corresponding to configurations pairs of the camera and the triangle that must be in the field of view of the camera. In the case of a perspective camera, Π can be written as

$$\Pi(c, l) = \Big(x_1 = X_1/Z_1,\ \ x_2 = X_2/Z_2,\ \ x_3 = X_3/Z_3,\ \ y_1 = Y_1/Z_1,\ \ y_2 = Y_2/Z_2,\ \ y_3 = Y_3/Z_3\Big)^\top. \tag{3}$$

3.2. Localization and Visual Servoing

3.2.1. Localization Equation. Let us consider a camera at a configuration $c \in \mathcal{C}$. Let us consider m triangles $T_1, \ldots, T_m$ of known configurations $l_1, \ldots, l_m \in L$, where each of them is visible from a camera C. Therefore, each pair (C, T_i) of camera-triangle gives rise to a localization equation system:

$$\Pi(c, l_i) = im_i. \tag{4}$$

Where $im_i \in I_{c,l_i}$ is the projection of T_i in the image space. im_i is measured and can thus be supposed as known. It can be denoted as $im_i = (x_1, x_2, x_3, y_1, y_2, y_3)^\top$. The knowledge of c and l_i is addressed in the next section.

3.2.2. Visual Servoing About a Reference Configuration. If the camera is expected to follow a reference trajectory viewing a reference object that can deform, then the result of equation (4) is expected to be in the neighborhood of a reference configuration of the camera and a reference configuration of the object. If we neglect the remainder of the linearization, then equation (4) can be linearized as follows:

$$\frac{\partial \Pi}{\partial c}(c_0, l_{0i})(c - c_0) + \frac{\partial \Pi}{\partial l}(c_0, l_{0i})(l_i - l_{0i}) = im_i - im_{0i}. \tag{5}$$

Where c_0 is the reference configuration of the camera on the pre-defined trajectory. l_{0i} is the expected reference configuration of the viewed triangle with reference features at im_{0i} in the image space I_{c,l_i}. At execution time, both of these reference configurations can be different from runtime configurations for two main reasons: Camera drift and object deformation. These consideration are valid if one assume that the localization is ran at high frequency [39] (faster than both the camera control loop and the dynamic of object's deformation). im_i and im_{0i}, respectively, represent the image of triangle T_i in sensor C and the expected image, i.e., the image that would be seen from configuration c_0 if there was no camera drift and no object deformation.

$\frac{\partial \Pi}{\partial c}$ is the Jacobian matrix of order 6 representing the variation of the image with respect to the variation of the camera configuration. This Jacobian matrix derives from the geometric properties of the camera and triangle. It represents the so called interaction matrix in the classic visual servoing terminology. $\frac{\partial \Pi}{\partial l}$ is the Jacobian matrix of order 6×9 representing the variation of the image with respect to the variation of the triangle configuration. This Jacobian is the zero-matrix in the case of non-deforming objects. Most of the previous works have modeled this matrix either with physics-based equations of deformations or with data-driven machine learning algorithms [40–42]. In this work, we use an optimal visual control framework to embed the control loop of the visual servoing algorithm. This approach allows us to account for the deformation behavior without having to numerically calculate or regress the deformations. Let us first focus on the analytical formulation of the interaction matrix. It is well known that for a single point at 3D corrdinates (X, Y, Z) perceived at 2D coordinates (x, y), the interaction matrix that relates variations of the camera pose to variations of the 2D projected coordinates is written as [15]:

$$L(x, y, Z) = \begin{bmatrix} L_u(x, y, Z) \\ L_v(x, y, Z) \end{bmatrix} = \begin{bmatrix} \frac{1}{Z} & 0 & \frac{-y}{Z} & -xy & 1 + x^2 & -y \\ 0 & \frac{1}{Z} & \frac{-y}{Z} & -(1 + y^2) & xy & x \end{bmatrix}. \tag{6}$$

The camera intrinsics are assumed to be known and undone from pixel coordinates. If we consider a triangle T at reference coordinates $l_0 = (X_1^0, X_2^0, X_3^0, Y_1^0, Y_2^0, Y_3^0, Z_1^0, Z_2^0, Z_3^0)^\top$ (see for instance Fig. 2), we represent the associated interaction matrix as

$$\frac{\partial \Pi}{\partial c}(c_0, l_{0i}) = \begin{bmatrix} L_u(x_1^{0i}, y_1^{0i}, Z_1^{0i}) \\ L_u(x_2^{0i}, y_2^{0i}, Z_2^{0i}) \\ L_u(x_3^{0i}, y_3^{0i}, Z_3^{0i}) \\ L_v(x_1^{0i}, y_1^{0i}, Z_1^{0i}) \\ L_v(x_2^{0i}, y_2^{0i}, Z_2^{0i}) \\ L_v(x_3^{0i}, y_3^{0i}, Z_3^{0i}) \end{bmatrix}. \tag{7}$$

Where $im^0 = (x_1^0, x_2^0, x_3^0, y_1^0, y_2^0, y_3^0, z_1^0, z_2^0, z_3^0)^\top$ is the reference projection of the triangle onto the camera plane (focal length and camera center are considered as known and undone.). We invite the reader to refer to appendix A.1 for more details on the definition of c and $\frac{\partial \Pi}{\partial c}(c_0, l_{0i})$.

3.2.3. Weighted Triangle-based Visual Servoing on Rigid Objects. If we consider the object as being rigid, then $l_i = l_{0i}$ for all the m triangles and equation (5) simplifies to:

$$\frac{\partial \Pi}{\partial c}(c_0, l_{0i})(c - c_0) = im_i - im_{0i}. \tag{8}$$

Classic visual servoing approaches on rigid objects consist in solving for the camera pose c such an equation for the whole set of triangles. This equation defines and solves the local task that is to fit the current triangle's view to the expected one. We can further assign real positive weights to the object's triangles that are relevant to the current task. The above equation turns to be

$$w_i \frac{\partial \Pi}{\partial c}(c_0, l_{0i})(c - c_0) = w_i(im_i - im_{0i}). \tag{9}$$

Here w_i is the assigned weight to triangle L_i. Considering the whole set of triangles composing the objects with their associated set of weights, we build the following system of equations

$$W(c - c_0) = IM - IM_0. \tag{10}$$

Where

$$W = \begin{pmatrix} w_0 \dfrac{\partial \Pi}{\partial c}(c, l_0) \\ \vdots \\ w_m \dfrac{\partial \Pi}{\partial C}(c, l_m) \end{pmatrix}, \tag{11}$$

$$IM = \begin{pmatrix} w_1 im_1 \\ \vdots \\ w_m im_m \end{pmatrix} \quad \text{and} \quad IM_0 = \begin{pmatrix} w_1 im_{01} \\ \vdots \\ w_m im_{0m} \end{pmatrix}. \tag{12}$$

To solve equation (10) for the camera pose in the rigid object case many consideration can be taken. For instance, the noise in the measurement of the triangle positions in the image, the inaccuracy of the 3D object shape that was scanned or modeled beforehand, etc. If the number of triangles is too large, the system of equations (10) has no exact solution and the servoing task is said to be over-constrained. In the contrary, if the system of equations (10) is underdetermined, there are an infinite number of possible camera poses that satisfies it. In this case, considerations like minimum norm solution can be taken to solve the system 10 as it was done in [39]. In the next section, we describe our third contribution which is embedding the weighted triangle-based visual servoing formalism in an optimal visual control framework for non-rigid objects. This embedding allows us to calculate the correction to impose to the camera motion while tracking a reference trajectory on a deforming object. The proposed approach does not consider any prior knowledge or parametric model that describes the deformation of the target object.

4. SERVOING ON DEFORMABLE OBJECTS WITH AN OPLIMAL VISUAL CONTROL

In 3D reconstruction of deformable objects from monocular views, the assumption of considering that any deformed shape lies at the minimal stretching/compressing energy has proven to be effective [43–46]. In our work, we do not aim to infer the 3D shape from the 2D view. We only need to compensate the deformation with a camera motion. Thus we define such motion as

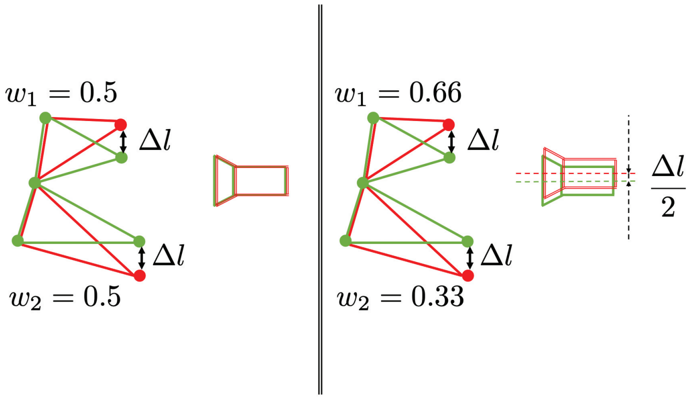

Fig. 3. Weight effect illustration. In this figure, we consider a reference camera in green viewing two reference triangles in green color. After deformation, the red vertices move in opposite directions by Δl along the vertical axis. This shift causes a non-rigid deformation of the two triangles. We consider here two scenarios: Left figure considers equal weights for both triangles. In this case, the correction with equation (14) does not produce any motion since the motion of the vertices compensates each other and the triangle have the same weight. Right figure considers the top triangle with a weight twice the value of the bottom one. In this case, equation (14) produces a vertical shift with half of Δl. If $w_2 = 0$ then the correction would be a vertical shift to the top within Δl.

being the one for which the stretching/compressing energy is minimal. Using equation (5), this statement comes to minimizing the deformation seen through the perspective projection, which can be formalized as follows

$$\hat{c} = \underset{\tilde{c}\in C}{\text{argmin}} \frac{1}{2}\sum_{i=1}^{m} \left\| \frac{\partial \Pi}{\partial l}(c_0, l_{0i})(l_i - l_{0i}) \right\|_2^2, \tag{13}$$
$$\text{s.t. } \frac{\partial \Pi}{\partial l}(c_0, l_{0i})(l_i - l_{0i}) + \frac{\partial \Pi}{\partial c}(c_0, l_{0i})(\tilde{c} - c_0) = (im_i - im_{0i}).$$

In our work, we aim at giving more importance to some spatial deformation than others. This goal is formalized through the association of real positive weight values to triangular primitives. Thus the above criterion can be rewritten as

$$\hat{c} = \underset{\tilde{c}\in C}{\text{argmin}} \frac{1}{2}\sum_{i=1}^{m} \left\| w_i \frac{\partial \Pi}{\partial l}(c_0, l_{0i})(l_i - l_{0i}) \right\|_2^2, \tag{14}$$
$$\text{s.t. } w_i\frac{\partial \Pi}{\partial l}(c_0, l_{0i})(l_i - l_{0i}) = -w_i\frac{\partial \Pi}{\partial c}(c_0, l_{0i})(\tilde{c} - c_0) + w_i(im_i - im_{0i}).$$

This criterion allows us to estimate a rigid camera motion $\hat{c}$ which minimizes the view between reference and current deformed shape. The weights allows us to give more importance to some regions above others as illustrated in Fig. 3. The following expression is used to compute the correction of the camera pose.

Proposition 1. *Let us consider a camera at some reference configuration $c_0 \in C$. The camera is viewing a deformed shape that is assumed to be at its minimal stretching/compressing energy. The rigid camera motion compensation $\hat{c}$, which satisfies criterion of equation (14), can be calculated as*

$$\hat{c} = c_0 + W^{+}(IM - IM_0), \text{ if } W \text{ is full column rank}, \tag{15}$$
$$\hat{c} = c_0 + W^{\top}(WW^{\top})^{-1}(IM - IM_0), \text{ else and if } W \text{ is full row rank}. \tag{16}$$

W^{+} being the pseudo-inverse matrix of W.

Proof. Taking into account equation (14) and using the linearisation equation (5), we write the optimal criterion as

$$\hat{c} = \underset{\tilde{c}\in C}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{m} \left\| w_i \frac{\partial \Pi}{\partial c}(c_0, l_{0i})(c - c_0) - w_i(im_i - im_{0i}) \right\|_2^2. \tag{17}$$

This is exactly the weighted sum of the squares of the Euclidean norms of the differences between each image that would be seen from $c \in C$ and the image seen at run time. This minimization can be rewritten in a concatenated formula as

$$\hat{c} = \underset{\tilde{c}\in C}{\operatorname{argmin}} \frac{1}{2} \|W(\tilde{c} - c_0) - (IM - IM_0)\|_2^2. \tag{18}$$

If W is full column rank, then solving for c this weighted least-squares problem gives the following solution

$$\hat{c} = c_0 + (W^\top W)^{-1} W^\top (IM - IM_0). \tag{19}$$

In our work, we consider that at every instant time there are sufficient visible triangles to compute the correction for the 6 degrees of freedom of the camera. For this purpose, we require at least 2 triangles to be visible to avoid ambiguous situations [45]. Given a mesh of N points, and a camera tracking full regions of this mesh, it is fair to consider this condition being verified in most of the time. In some edge cases where this might not be verified (for instance the camera is too close to the object), we setup a safe test for the number of visible triangles which allows us to compute the solution of minimum norm. If W is full row rank, then the camera correction is computed as the minimum norm of camera motion satisfying the following criterion [47]

$$\hat{c} = \underset{\tilde{c}\in C}{\operatorname{argmin}} \frac{1}{2} \|\tilde{c} - c_0\|_2 \text{ s.t. } W(\tilde{c} - c_0) = (IM - IM_0). \tag{20}$$

Given that W is full row rank, the solution of this under-determined least-squares problem is given as

$$\hat{c} = c_0 + W^\top (W W^\top)^{-1} (IM - IM_0). \tag{21}$$

In some isolated degenerate cases where W is neither full column rank nor full row rank, the camera is not corrected from the reference configuration and we keep $\hat{c} = c_0$.

The developments conducted in this section can be summarized as follows. Rigid compensation with camera motion of a nonrigid object motion consists of solving a system of equations that relate the configuration of the camera with the images of the object's triangles. If the system is over-constrained, the compensation consists of finding a configuration that minimizes a weighted sum of residues. If there is no deformation and the triangles are all rigid, the choice of weights will have no effect on the result (ignoring noise in vertex detection). If the object's triangles deform, the choice of weights will have an effect on the corrected configuration $\hat{c}$ as illustrated in Fig. 3. For this reason, we propose to use these weights as a tool for planning triangle mesh-based motions for deformable objects.

4.1. Triangle-Mesh-Based Motion for Visual Servoing on Deformable Objects

Definition 1. Let us consider a camera following a reference trajectory and viewing a deformable object meshed with m traingles $T_1, \ldots, T_m$. A triangle-mesh-based motion is composed of two main components:

(i) A reference trajectory

$$\gamma : [0,1] \to C : \; s \mapsto \gamma(s), \tag{22}$$

where $[0,1]$ is a normalized interval of the trajectory temporal parameter s.

(ii) m continuous positive real-valued functions $w_1, \ldots, w_m$:

$$w_i : [0,1] \to R^+ : s \mapsto w_j(s), \tag{23}$$

such that $w_i \;\; (s) = 0$ for any s such that T_i is not visible by camera C when the camera is at configuration $\gamma(s)$. Continuity of w_i is required to avoid undesirable jumps during the correction of the camera trajectory [39].

The above triangle-mesh-based motion for the rigid motion compensation of the camera pose consists in correcting the current configuration of the camera within the closed-loop control task by computing the correction from Proposition 1 about the reference configuration $\gamma(s)$ and with values of the weights $w_i(s)$ for abscissa s along the trajectory. The correction formula can be written for a given abscissa s as

1) If $W(s)$ is full column rank

$$\hat{c}(s) = \gamma(s) + (W(s)^\top W(s))^{-1} W(s)^\top (IM(s) - IM_0(s)). \tag{24}$$

2) If $W(s)$ is not full column rank but is full row rank

$$\hat{c}(s) = \gamma(s) + W(s)^\top (W(s)\, W(s)^\top)^{-1} (IM(s) - IM_0(s)). \tag{25}$$

3) Else

$$\hat{c}(s) = \gamma(s). \tag{26}$$

Where $W(s)$, $IM(s)$ and $IM_0(s)$ are extensions of the notations in equations (11) and (12) when the reference camera configuration follows a reference trajectory $c_0(s) = \gamma(s),\; s \in [0,1]$. The case 3 of equation (26) is an edge case and almost never occurs. It is used as a safety case if ever W is neither full column rank nor full row rank. The above formulas of corrected trajectory $\hat{c}(s)$ from equations (24)–(26) can be interpreted as the one satisfying an optimal visual alignment and represents the minimum of the integral along the whole trajectory of the difference between reference view and current deformed view as follows

$$J = \int_0^1 \frac{1}{2} \| W(c(s) - \gamma(s)) - (IM(s) - IM_0(s)) \|_2^2 \; ds. \tag{27}$$

In order to have smooth corrections it is necessary to have continuons weights [39]. This is the case if for instance the weights are proportional to the 2D area of the viewed triangle in the image plane (see the experimental Section A.2 for details on their usage and their computation). If the object is an organ and the task consists in compensating the deformations viewed by a laparoscope, then ideally the weights can be set manually by a technician before surgery. Indeed, a surgeon with the help of technician can point the main target area of an organ to be kept steady when seen by the laparoscope. The technician can then set higher weights to those area compared to others. Such semi-automatic process can be done at the surgery planning.

5. PROPOSED ALGORITHM AND IMPLEMENTATION DETAILS

Algorithm 1 Triangle-Mesh-Based Motion for Visual Servoing on Deformable Object

Require: Reference trajectory $\gamma : [0, 1] \to C : s \mapsto \gamma(s)$;
Require: Deformable object meshed with m triangles $T_1, \ldots, T_m$;
Require: step_size for sampling reference trajectory;
Require: max_iter for maximum number of iterations;
Require: ϵ_{im} stopping error on image difference;
1: **Initialization:**
2: Load reference trajectory γ for the camera;
3: Load data mesh of the reference object;
4: Load data mesh of the same object in its deformed state;
5: **Trajectory Loop:**
6: **for** $s \in [0, 1]$ **with** step_size **do**
7: Get current camera configuration $c(s) \leftarrow \gamma(s)$;
8: **Triangle-Mesh Loop:**
9: **for** each triangle i from 1 to m **do**
10: Get current weight w_i by applying formulas in equations (B14) and (B15);
11: Get current interaction matrix by applying formula of equation (7);
12: Get the reference image projection im_i^0 of the triangular primitive;
13: Compute the current task function of the triangle using equation (10);
14: **end for**
15: $error \leftarrow \epsilon_{im} + 1$;
16: $iter \leftarrow 0$;
17: **Visual Servoing Loop:**
18: **while** $iter < max\_iter$ **or** $error > \epsilon_{im}$ **do**
19: Get the current image projection primitive im_i from the current camera pose;
20: Assemble object weighted interaction matrix W using equation (11);
21: Assemble reference and current object weighted image IM, IM_0 using equation (12);
22: Solve the optimal servoing function using equations (24), (25), (26);
23: Use the obtained $\hat{c}(s)$ to update the current camera configuration;
24: $iter \leftarrow iter + 1$;
25: $error \leftarrow \left\| W\left(\hat{c}(s) - c(s)\right) - (IM - IM_0)\right\|_2$;
26: **end while**
27: **end for**
28: **Output:**
29: Final optimal visual servoing trajectory given the current deformation of the object;

The visual servoing Algorithm 1 described in this section uses a deformable object's mesh to achieve precise camera positioning relative to the object, even as the object deforms. The key steps are:

(1) The trajectory loop (lines 5:27) iterates through the reference camera trajectory, and for each camera pose, the algorithm computes:
   (a) for each triangle, using the reference object, the weights, the reference image projection, and the interaction matrix;
   (b) an overall weighted interaction matrix and reference and current object weighted images;
(2) The visual servoing loop (lines 17:26) then iteratively solves for the optimal camera pose update that minimizes the difference between the reference and current object weighted images, updating the camera pose accordingly.

This approach allows the visual servoing to adapt to deformations of the object, using the mesh representation to correct the camera trajectory accordingly.

This algorithm was implemented using Matlab version a2015b. It was run on a laptop computer with an Intel Core (TM) i5-4200U CPU processor and 6 GB of RAM. The object model used was a triangle mesh with 10 000 vertices and 20 000 faces. The reference trajectory was sampled at a step_size of 10 Hz. The maximum number of iterations, max_iter, was set to 100, and the stopping error, ϵ_{im}, was set to 0.01 pixels. The time complexity is of $O(N \times m^4 \text{max\_iter})$ order and the space complexity is of $O(m^2)$ order.

The overall algorithm runs at an average speed of 40 Hz, enabling real-time visual servoing of the deformable object.

This algorithm was tested across various scenarios, including multiple object types, reference trajectories, deformation modes, and levels of image noise. The following experimental section describes the results obtained in these diverse settings.

6. RESULTS ON SIMULATED DATA

This section shows several representative examples of our verification tests with simulated data using two deformable objects: a planar object that is deformed to a bump-like shape and a model of human liver that is deformed on its left and right lobes. We ran experiments on both linear and nonlinear deformations. We tested our method on two geometric trajectories: Linear and circular. To validate the robustness of our approach, we simulates noise detection on the 2D mesh vertices. We used a perspective camera with a focal length of 1500. We ran our simulations on a Core (TM) i5-4200U CPU processor and 6 GB of RAM and Matlab2015a. In the shown figures, distances are in meters, translation speeds of the camera is in meter per second and rotation speeds are in radian per seconds. The errors in image space are in pixels.

6.1. Results on a Planar Object and Linear Geometric Path

Figure 4 represents the target object in 3D space and its projection on the camera. The object is composed of 200 triangles as shown in Fig. 5. During deformation, the camera trajectory is used as a reference constraint to ensure that the target features remain within and adjust to the expected

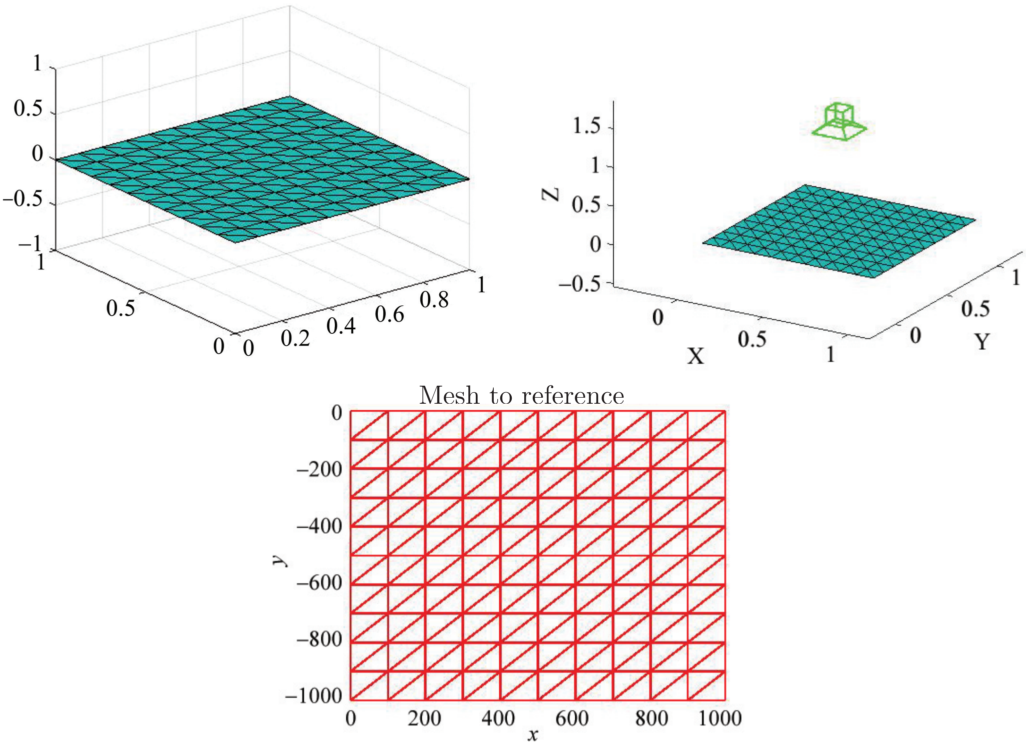

Fig. 4. Top right: The reference object is a planar triangular 3D mesh. Top left: The reference object and the camera configuration in 3D space. Bottom: The viewed planar mesh in the camera plane.

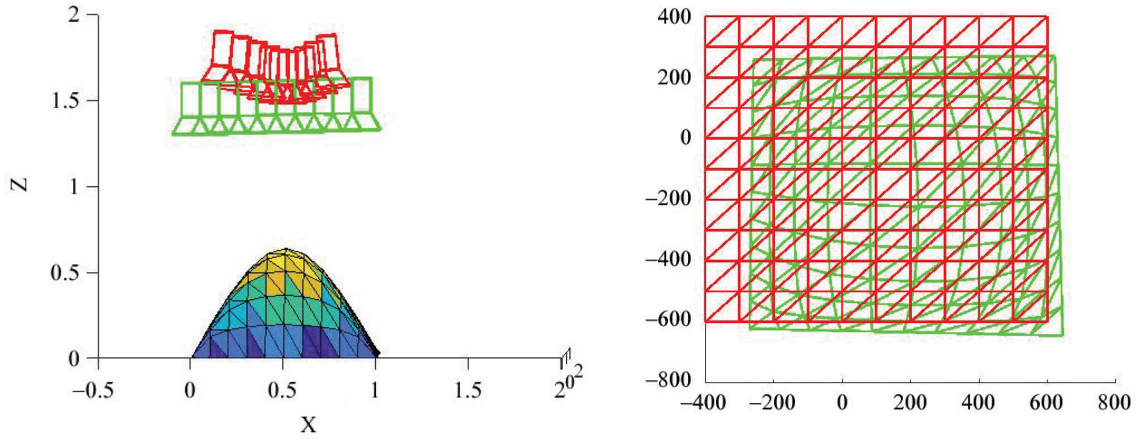

Fig. 5. Left: Deformed reference object with reference (green) and deformed (red) camera trajectories. The reference camera follows a linear planned trajectory. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory. Here all weights are equal to one. Right: Reference viewed object (red) and deformed viewed object (green). The corrected trajectory tries to fit as much as possible the reference view of the object.

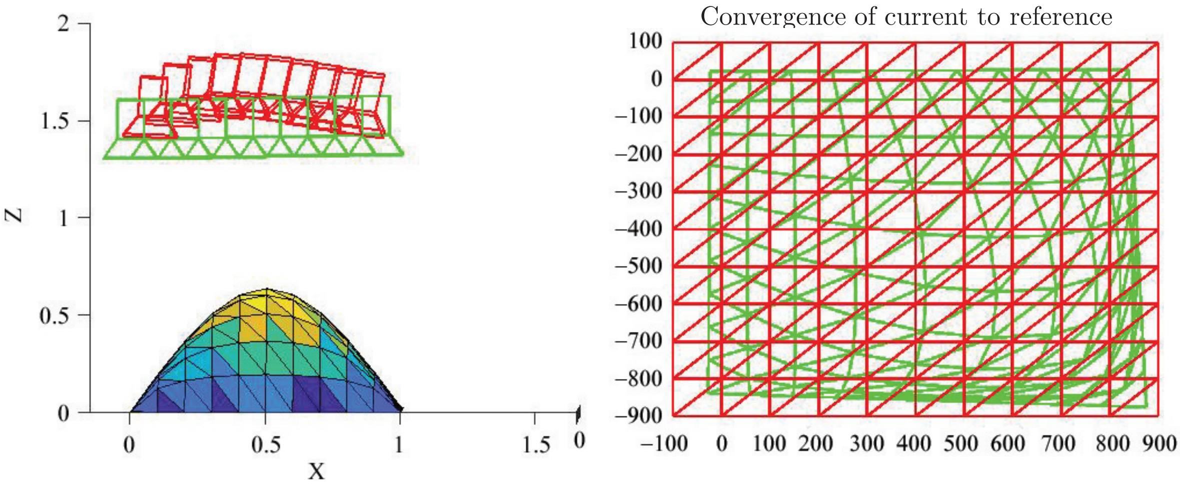

Fig. 6. Left: The deformed 3D plane object with the camera which follows a linear planned trajectory. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory. Here all weights are equal to one except the weights associated to the triangles on the top hill (yellow triangles). They are set to 4. Right: Reference viewed object (red) and deformed viewed object (green). The corrected trajectory tries to fit as much as possible the reference view of the object.

view dynamically. In Figs. 6, we control movement of the camera which depends on weights along the path. In Figs. 5 and 6, yellow, orange, green and blue regions represent respectively the heatmap of deformation going from high to low deformations. In this experiment we allocated continuous constant weights. The weights of yellow region are the highest with continuous constant weights equal to 4. The remaining regions have continuous constant weights equal to 1. There is no upper or lower limit to the choice of weights. Only relative bigger/smaller values matter to modulate more/less tracking focus on a particular area. Regions which are relevant at run time have bigger weight values to enforce the tracking task to better follow those deformed parts.

6.2. Results on a Simulated Model of a Human Liver

The Liver is the largest organ in the human body. It belongs to the digestive system and provides many vital functions to the body. In our work, we use the liver mesh to test the performance of

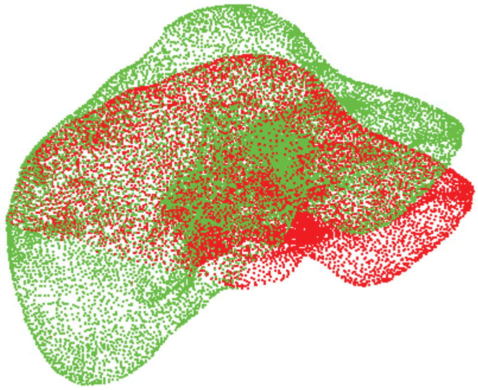

Fig. 7. Liver before and after nonlinear elastic deformation. Red Liver represents the organ at rest. Green Liver represents the organ after nonlinear deformation.

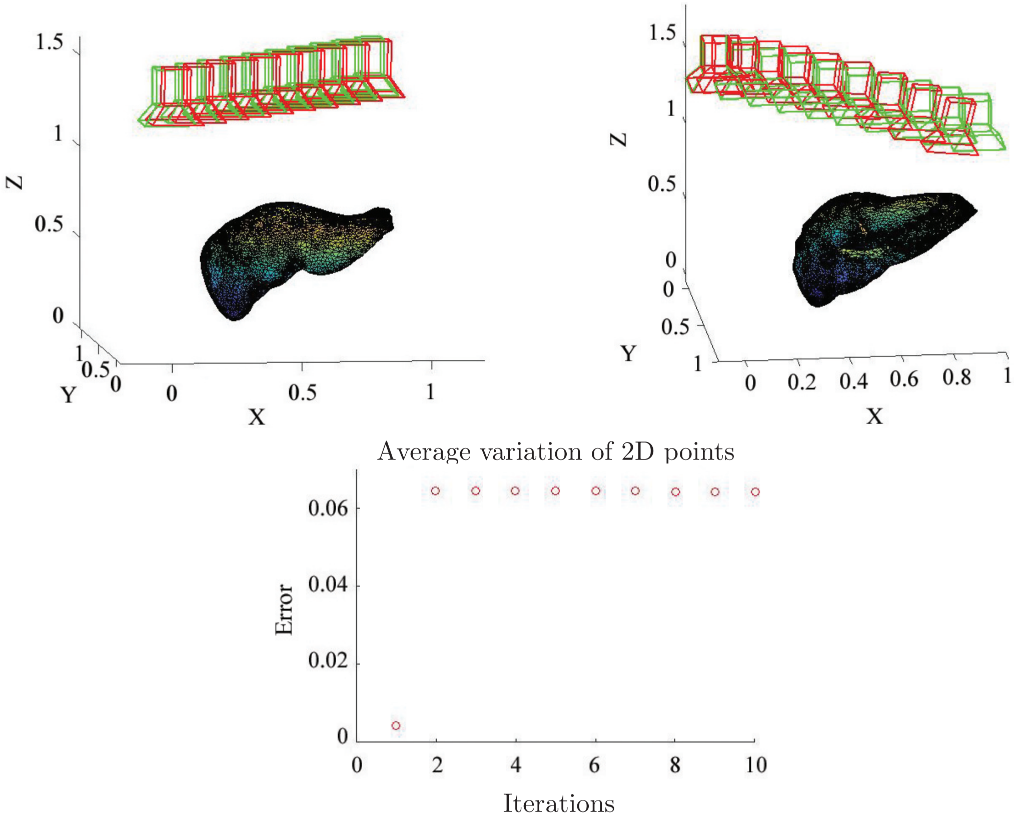

Fig. 8. Liver after a linear elastic deformation and camera follows a reference straight trajectory. Results of test (1) are shown. Left shows The reference trajectory. Right shows a side view of the trajectory after correction. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory. Bottom shows the variation of pixel points during the correction of the camera pose. Green cameras is the reference trajectory.

our method. To deform the liver we used a Mooney-Rivilin hyper-elastic physical model and used the approach proposed by Saidi et al. work [48] to speed-up the calculation of the nonlinear elastic deformations. The liver mesh contains 130 382 triangle and 12 226 vertex (see Fig. 7 for an illustration). We ran experiments on two types of deformations: Linear and nonlinear. For the same mesh of the liver, we moved a selected deformed part of 825 by 6 mm in the z-axis, which is considered as a linear deformation. In a second deformation, we moved the same 825 vertices by 22 mm, which is considered as a non-linear deformation.

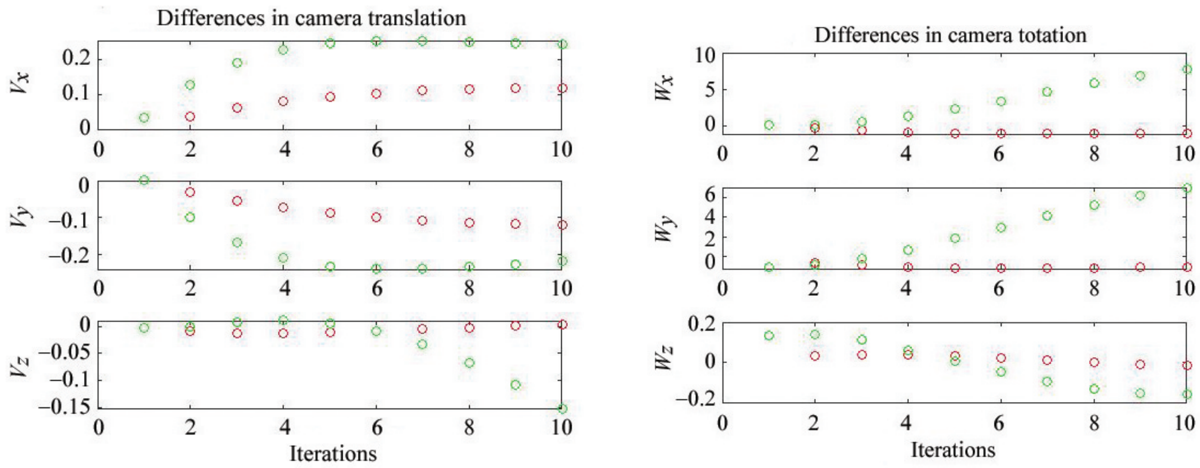

Fig. 9. Analysis of the performance of the camera speed when tracking a deformed object with linear deformation. Left figure represents camera speed in translation. Right figure represents camera speed in rotation. Red dots represent camera speed following strategy of test (1). Green dots represent camera speed following strategy of test (2). ω_y and ω_z are similar for both tests.

6.2.1. Results with linear deformation of the liver. For the linear deformation, we used a command to control the movement of the camera which was following a reference horizontal straight line trajectory as shown in Fig. 8. We ran two tests: Test (1) where all the weights are constants and equal to 1 for all the triangles. Test (2) where the weights vary according to the amount of viewed deformation. This amount is computed as the ratio of the area of current deformed triangle above the area of the reference visible triangle (see Section A.2 for more details). The areas are computed on the visible triangles in the 2D image (reference and current deformed). This strategy allows us to give more importance to deformed regions in comparison to other areas less or non-deformed. The obtained result shows us that the camera has followed the deformation of the liver without losing the imposed trajectory (test (2)). In test (1), the trajectory of the camera has changed barely and uniformly since the weights of all triangles were equal. In test (2) the camera has updated significantly its trajectory around the deformed area while keeping a trajectory close to the reference one far from the area of deformation. Figure 9 shows the performance of the camera control analysis during the dynamic tracking. It displays the translation and rotation of the camera during the tracking of the liver deformation. The rotation speeds ω_y and ω_z are similar for both tests. The translation amounts in y and z axes are also almost similar. The noticeable difference is in translation and rotation along x-axis which corresponds to the major axis of deformation.

6.2.2. Results with nonlinear deformation of the liver. In this experiment we ran the same configuration of test (1) and test (2) setups as previously. Here we consider a nonlinear deformation with a maximum of 6 mm displacement of the left liver lobe as shown in Fig. 7. The trajectory of the camera is a straight horizontal line represented by green, and the corrected view represented by red color. Figure 10-left shows a corrected camera trajectory in red which was obtained with all the weights being set to 1. Figure 10-right shows a corrected camera trajectory in red which was obtained with the weights of the most deformed regions are equal to 5 and the least deformed or non-deformed has a weight of 1. The curvature of the corrected trajectory is more important in the case of test (2) (right figure). The nonlinear deformation of the triangles in the image is kept as close as possible to their areas in the reference image. The weight functions allow us to take the most relevant deformed triangles from the liver and adjust the amount of correction accordingly. We notice through the Fig. 10 on the right that the camera path is most affected. This is consistent with the nonlinear deformed triangles.

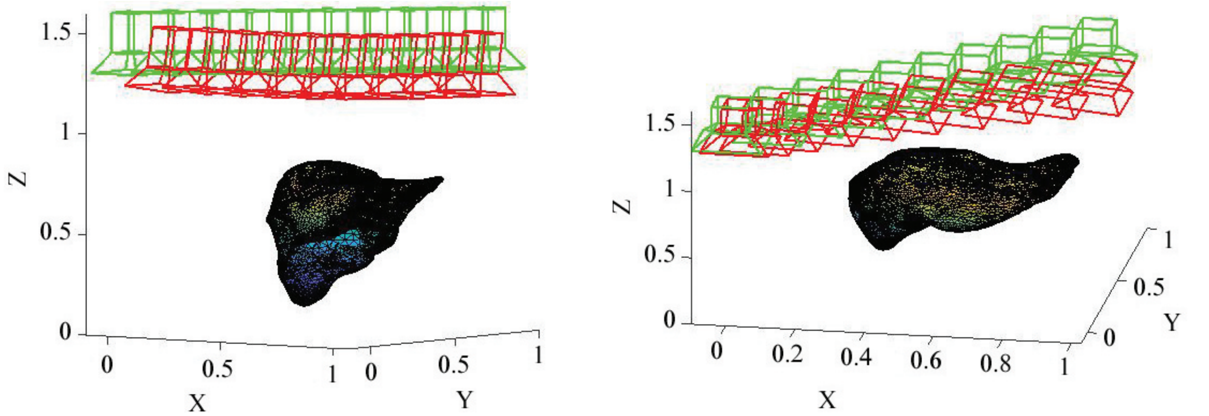

Fig. 10. Liver after a nonlinear elastic deformation and camera follows a reference straight trajectory. Results of test (2) are shown. Left shows side view view. Right shows another side view. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory.

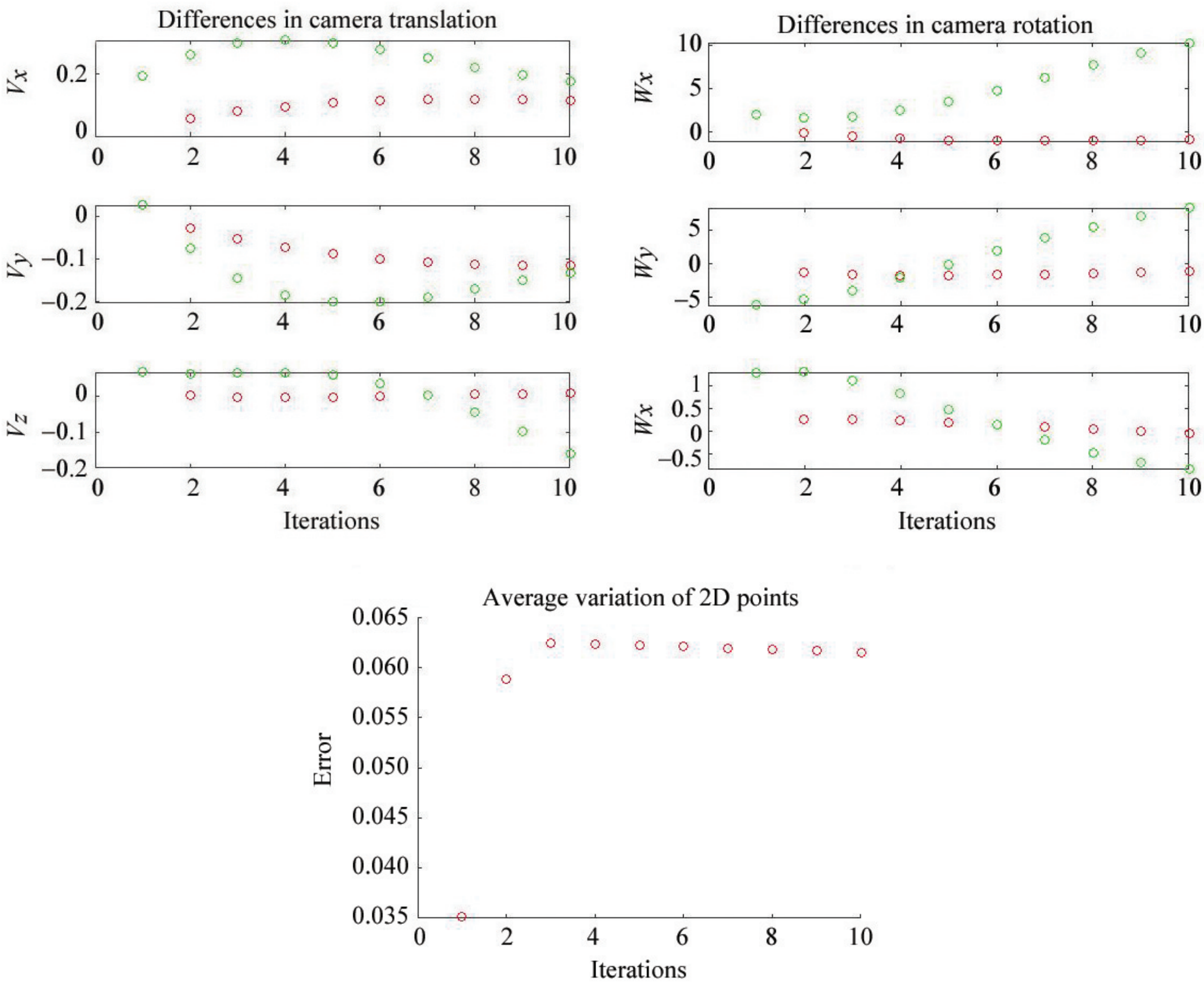

Fig. 11. Analysis of the performance of the camera speed when tracking a deformed object with nonlinear deformation. Left figure represents camera speed in translation. Right figure represents camera speed in rotation. Red dots represent camera speed following strategy of test (1). Green dots represent camera speed following strategy of test (2).

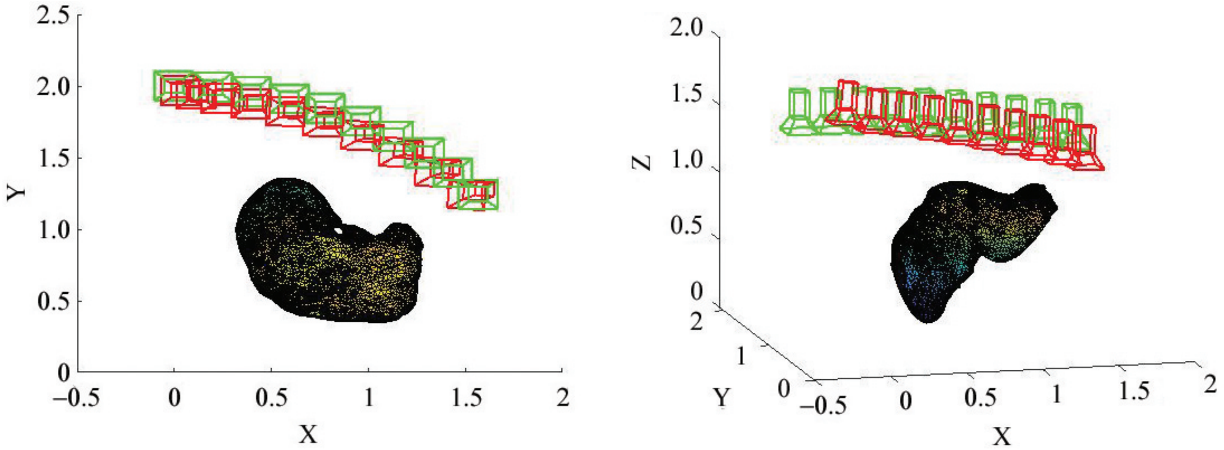

Fig. 12. Liver after a nonlinear elastic deformation and camera follows a reference curvilinear trajectory. Results of test (1) are shown. Left shows top view. Right shows a side view. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory.

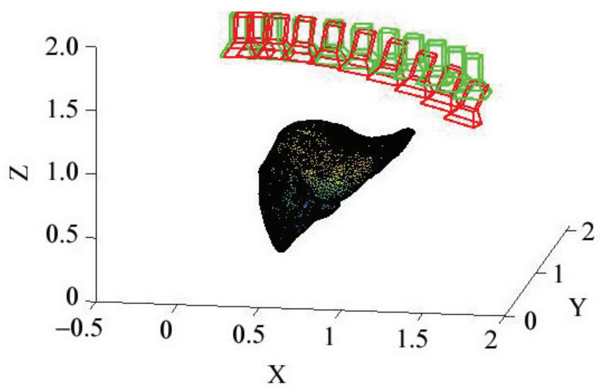

Fig. 13. Liver after a nonlinear elastic deformation and camera follows a reference curvilinear trajectory. Results of test (2) are shown. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory.

We can observe that the convergence speed is quite fast for the approach (after 10 iterations). The result of the computation of the correction of the camera configuration is displayed in Fig. 11.

6.2.3. Results with nonlinear deformation and curvilinear reference trajectory. For a non-linear trajectory, we performed a test on a curvilinear reference trajectory plane with the following parameters: $\theta = (0 : 0.01 : \frac{\pi}{2})$, $c_x = R * \cos(\theta)$, $c_y = R * \cos(\theta)$, $c_z = 1$. The angle of the camera is kept constant during this trajectory pointing down to the negative z-axis. We used this reference camera configuration in equation (22). We tested our control strategy with the two test cited in the Section 6.2.1 (test (1) and test (2)). For test (1), we observed that the camera followed the trajectory as is showed in the Fig. 12 in the right. For test (2), we made a test with the same control strategy and curvilinear trajectory but we changed the weight of the triangles where we gave an importance for the triangles that deform a constant value of weight between 1 and 5 as is showed in the Fig. 13. The result obtained shows us that test (2) is better than the test (1). In test (1), the camera followed the trajectory but the deformation was not well captured. Test 2 succeeded in following the trajectory with a precise deformation capture. In addition, the result of the speed translation and rotation of the camera tracking was better in test (2) than the test (1) as it is shown in Figs. 17. Also, we noted that even in the pixel variation there is a difference which test (1) has a variation of 0.06 to 0.1. and test (2) has a variation of 0 to 0.5.

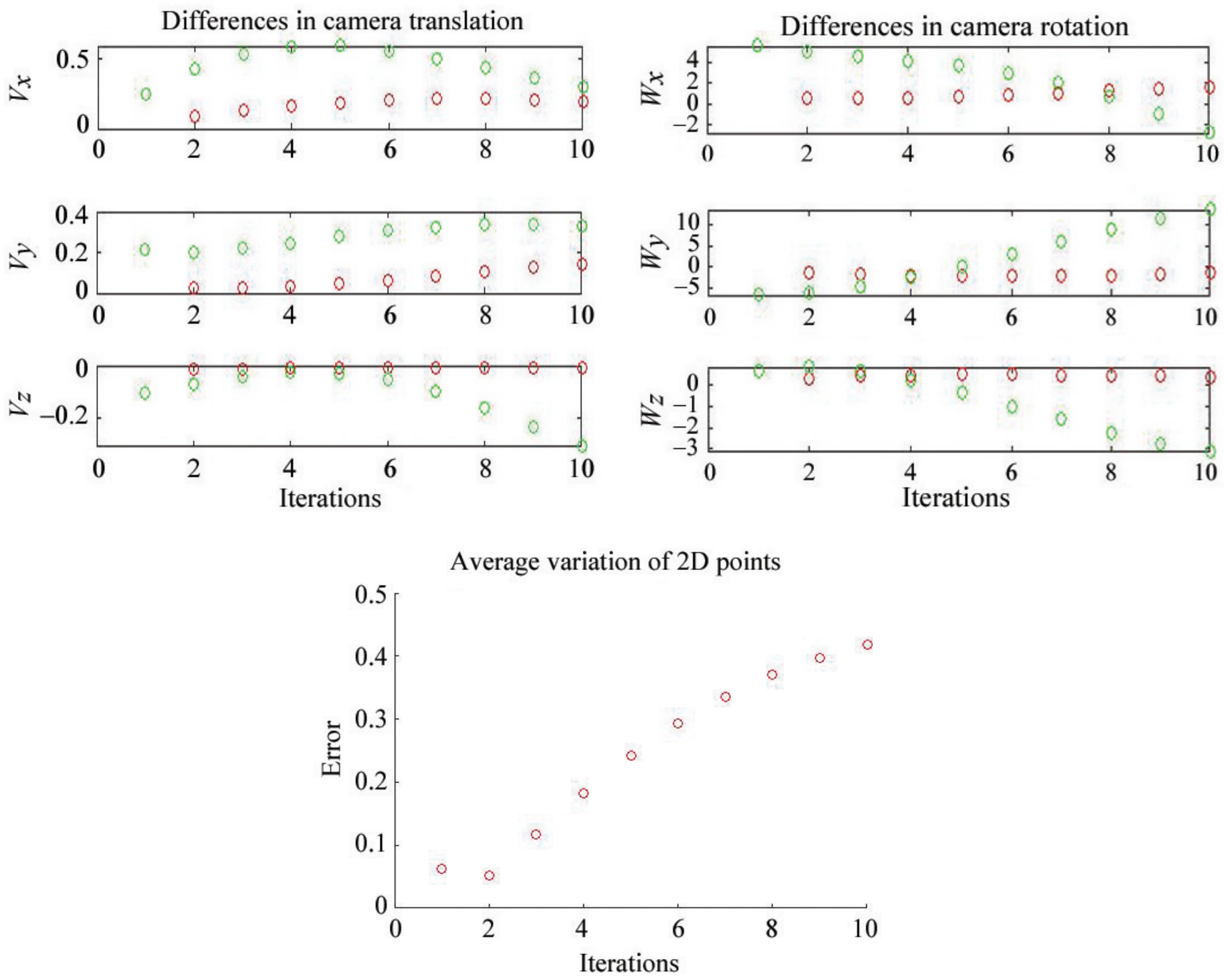

Fig. 14. Analysis of the performance of the camera speed when tracking a deformed object with nonlinear deformation and following a curvilinear trajectory. Left figure represents camera speed in translation. Right figure represents camera speed in rotation. Red dots represent camera speed following strategy of test (1). Green dots represent camera speed following strategy of test (2).

6.3. Robustness to noise in 2D vertex detection

Our method aims at making the camera motion control more robust. Therefore, we tried to mimic the real image measurements perturbations with a Gaussian random noise added to equation (4). This noise is centered with a standard deviation of 1 pixel. It is added to the coordinates of the image im_i, $1 \leqslant i \leqslant m$. We performed a test on a nonlinear deformation with a comparison of both experimental setups (test (1) and test (2)). We observed that for test (1), the corrected trajectory was not as satisfactory when compared to the same setup in the noise-free case (see Fig. 15 for an illustration). For test (2), the result was more consistent with the one obtained in the noise-free case as can be seen in Fig. 16. So we can say that allocating appropriate weights to relevant deformed regions is not influenced by the perturbations while tracking and control the camera trajectory. Figure 17 show the differences speed between the cameras translation, rotation and pixel point variation. The green color is for test (1) and the red color is for test (2). We noticed that test (1) is slow in translation and rotation compared to test (2).

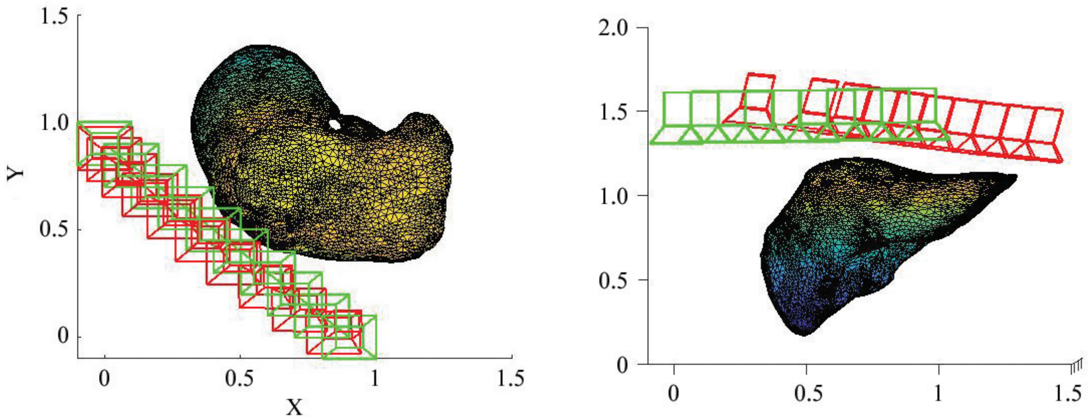

Fig. 15. Liver after a nonlinear elastic deformation and camera follows a reference curvilinear trajectory. Results of test (1) with noisy data are shown. Left shows top view. Right shows a side view. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory.

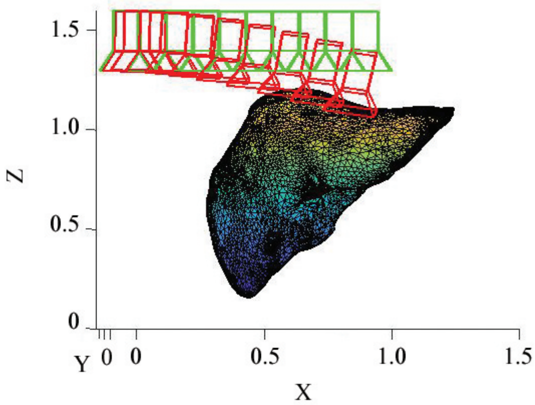

Fig. 16. Liver after a nonlinear elastic deformation and camera follows a reference curvilinear trajectory. Results of test (2) with noisy data are shown. Left shows top view. Right shows a side view. Green cameras is the reference trajectory. Red Cameras is the corrected trajectory.

6.4. Discussions and comments

The importance of the weights is clearly shown in Figs. 5 and 6. On one hand, we can see on Fig. 6-left how the reference trajectory is bent mostly when it comes close to the most deformed area. This behavior is obviously not observed when all the weights are equal to each other. On the other hand, we can see how the view of the deformed region is closer to the reference view in Fig. 6-right when compared to Fig. 5-right. The importance of the optimal visual control framework and its robustness can be seen in the results obtained in Figs. 8 to 17. These figure emphasize the reliability of the proposed approach to the type of deformation (linear/nonlinear), the type of trajectories (geometrically linear/nonlinear) and the noise in 2D detection. We have shown that at every scenario of these experiments, the proposed method allows us to compute a rigid camera correction. Given that the most relevant triangles of the deformable area are given high weight values, the reference trajectory is bent to adapt mostly to those areas no matter are the aforementioned categories.

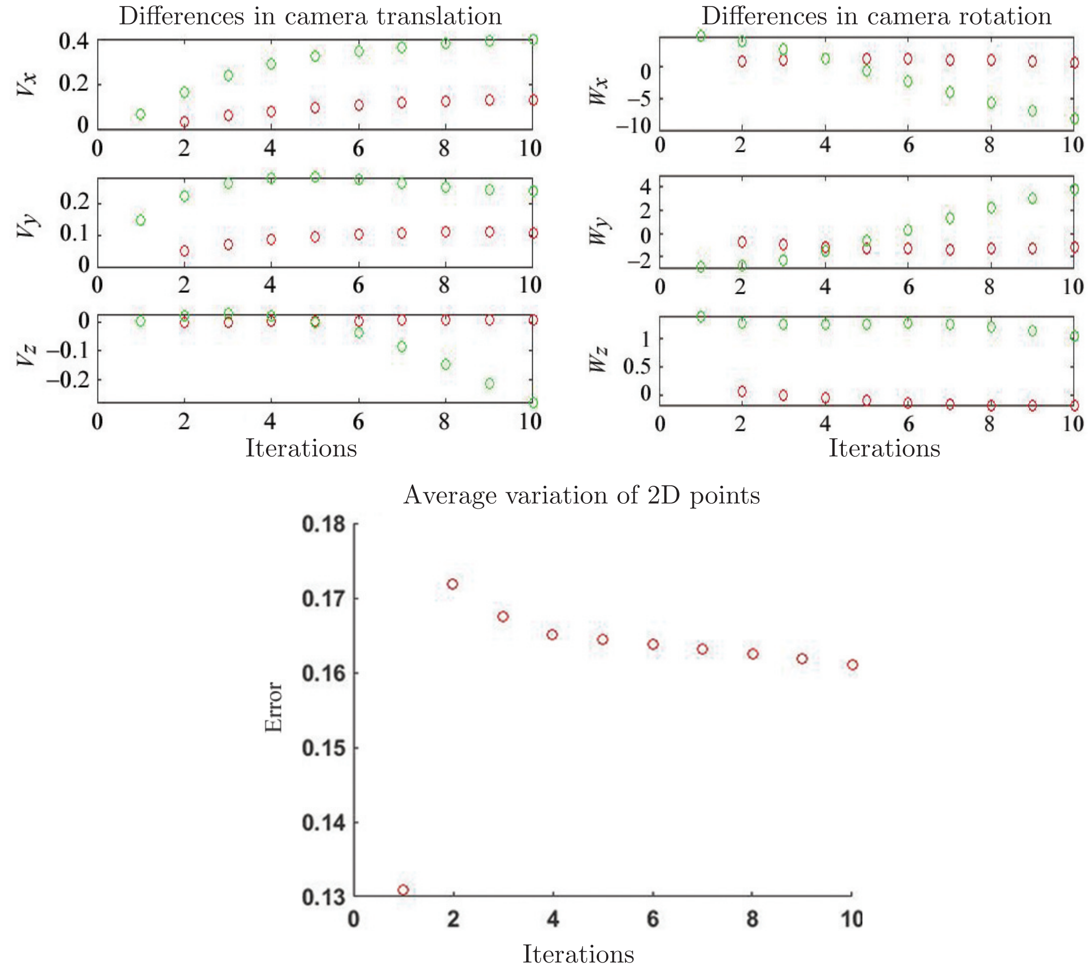

Fig. 17. Analysis of the performance of the camera speed when tracking a deformed object with nonlinear deformation and noisy data. Left figure represents camera speed in translation. Right figure represents camera speed in rotation. Red dots represent camera speed following strategy of test (1). Green dots represent camera speed following strategy of test (2).

7. CONCLUSION

This paper presents a nonparameteric approach to visually compensate nonrigid object deformation with rigid camera motion. This compensation is processed while the camera is following a reference trajectory thanks to continuous weight functions. These weights ensure to focus on the most relevant deforming areas while stabilizing the view at run-time according to the reference view. We embed our compensation strategy in an optimal visual control setup. We experimentally proved that our approach is robust to linear/nonlinear deformations, to linear/nonlinear geometric paths and to noise in 2D detection. We emphasized the fact that our method do not require information on the type of object, deformation or trajectory.

This work allows us to setup strong formal basis for a prospective real application in a laparo-surgery context. The surgeon can plan the intervention with the help of a technician giving large weights to relevant area of the target organ and describing a reference geometric path for the laparoscope. During surgery, the motion compensation can be done by a robot holding the laparoscope using our method. Based on state-of-the art approaches of 2D landmark detection in laparoscopy, we will implement a realtime algorithm to detect and build 2D mesh with vertices. This will allows us to fully automate the processed trajectory tracking and visual servoing.

APPENDIX

A.1. COMPUTATION OF THE INTERACTION MATRIX

Given a 3D vertex with coordinates $(X, Y, Z)^\top$ in the camera frame then its 2D perspective projection onto the camera plane is given as

$$x = X/Z, \tag{A.1}$$

$$y = Y/Z. \tag{A.2}$$

Where $im = (x, y)^\top$ is the image coordinate of the 3D point without considering the focal length and the principal point (in our work we consider a calibrated camera of known intrinsics). The time derivative of the above projection about a reference 3D vertex coordinate (X_0, Y_0, Z_0) is given as

$$\dot{x} = \dot{X}/Z - \dot{Z}x/Z, \tag{A.3}$$

$$\dot{y} = \dot{y}/Z - \dot{Z}y/Z. \tag{A.4}$$

Let us consider the translation velocity of the cameras as $v = (v_x, v_y, v_z)^\top$ and its rotation velocity as $\omega = (\omega_x, \omega_y, \omega_z)^\top$ such that the total vector of camera velocity is written as $\dot{c} = (v^\top, \omega^\top)^\top$. The formula relating the velocity of the 3D point to the velocities of the camera can written with the following formula from the classic work by [49]:

$$\dot{X} = -v_x - \omega_y Z + \omega_z Y, \tag{A.5}$$

$$\dot{Y} = -v_y - \omega_z X + \omega_x Z, \tag{A.6}$$

$$\dot{Z} = -v_z - \omega_x Y + \omega_y X. \tag{A.7}$$

Replacing the 3D velocity expression of equations (A.5)–(A.7) into the expressions of 2D velocity (A.3)–(A.4)

$$\dot{x} = -v_x/Z + xv_z/Z + +xy\omega_x - (1 + x^2)\omega_y + y\omega_z, \tag{A.8}$$

$$\dot{y} = -v_y/Z + yv_z/Z + (1 + y^2)\omega_x - xy\omega_y - x\omega_z. \tag{A.9}$$

Rearranging the terms and using the notation of the interaction matrix from equation (6), we can rewrite the above system as

$$(\dot{x}, \dot{y})^\top = L(x, y, Z)\dot{c}. \tag{A.10}$$

If we consider a high enough sampling period T (at least 30 fps), then the above formula can be discretized as follows

$$(x - x_0, y - y_0)^\top/T = L(x, y, Z)(c - c_0)/T. \tag{A.11}$$

Where c is represented by the translation position of the camera center and the Rodriguez vector of rotation

$$c = (t_x, t_y, t_z, \theta_x, \theta_y, \theta_z)^\top. \tag{A.12}$$

Where (t_x, t_y, t_z) are the 3D position coordinates and $\theta = \sqrt{\theta_x^2 + \theta_y^2 + \theta_z^2}$ is the angle of rotation and $(\theta_x/\theta, \theta_y/\theta, \theta_z/\theta)$ is the unit axis of rotation [50]. $c_0 = (t_x^0, t_y^0, t_z^0, \theta_x^0, \theta_y^0, \theta_z^0)^\top$ is a reference camera configuration close to c. Simplifying equation (A.11) by the sampling period on both sides gives

$$(x - x_0, y - y_0)^\top = L(x, y, Z)(c - c_0). \tag{A.13}$$

This mapping is known as the interaction matrix which relates close variations of the camera velocity to close variations of the 2D projected points. Let us consider the mapping Π that is defined in equation (3) which assumes the projection of a 3D triangle as a compound 3 vertices geometric primitive. The first order Taylor expansion of equation (4) about a reference camera configuration c_0 close to a given camera configuration c is provided in equation (5). The term $\frac{\partial \Pi}{\partial c}$ is the interaction matrix that relates the variation of the variation of camera configuration to the 2D projection of the 3 vertices composing the triangle. Thus the analytic expression of $\frac{\partial \Pi}{\partial c}$ is obtained by vertically concatenating the matrix provided in equation (A.10). Thus matrix $\frac{\partial \Pi}{\partial c}$ is 6 columns. It has 6 rows since every vertex' triangle gives 2 equations and every triangle has three of them.

A.2. EXAMPLE OF WEIGHT COMPUTATION WITH VISIBLE AREAS OF TRIANGLES

There is no optimal or exact way of computing the weight functions [39]. However, there many approaches that can be taken to build continuous weight functions. The simplest one is to make them constant as was done the protocol test (1) in our experiment. Another approach associate them to the amount of deformation to take into account deformed regions more than others. In this case, let us assume that we have one undeformed 3D triangle viewed by a camera along a reference trajectory $\gamma(s)$. The 2D projection of this triangle onto the reference camera is continuous along $\gamma(s)$ since it is a static object seen by a continuously moving camera (see Fig. 18). The 2D location of the projected triangle can be written as

$$im_0(s) = \left(x_1^0(s), x_2^0(s), x_3^0(s), y_1^0(s), y_2^0(s), y_3^0(s)\right)^\top . \tag{A.14}$$

The area of the projected triangle along $\gamma(s)$ is also continuous and can be computed as every s as

$$a_0(s) = \frac{1}{2}\det\begin{pmatrix} x_1^0(s) & y_1^0(s) & 1 \\ x_2^0(s) & y_2^0(s) & 1 \\ x_3^0(s) & y_3^0(s) & 1 \end{pmatrix}. \tag{A.15}$$

Where det() denotes the determinant of squared matrices. If we consider $a(s)$ as the area of the projected triangle onto the camera at runtime, then the formula of computing the weight function

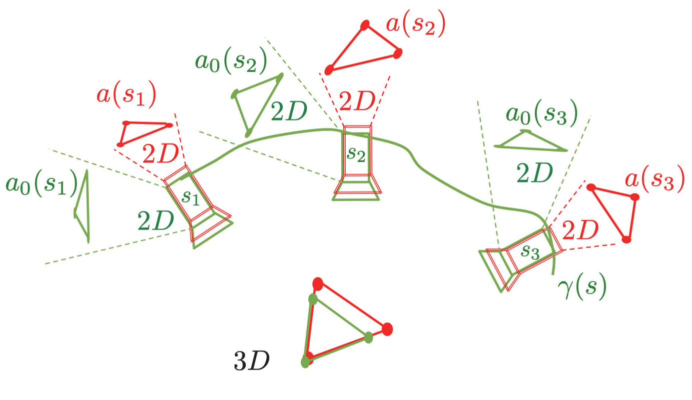

Fig. 18. Example of weight computation based on the area of 2D triangles. In this figure, we consider a reference camera in green viewing one reference triangle in green along a reference trajectory $\gamma(s)$, $0 \leqslant s \leqslant 1$. After deformation, the new deformed triangle is showed in 3D space with red color. The corrected camera along the trajectory is displayed in red color. We picked-up three samples of the trajectory s_1, s_2 and s_3 to depict three couple of views of both the reference triangle from reference camera and deformed triangle from corrected camera pose. As can be seen in the 2D image planes of both cameras, the area of the viewed triangles change continuously along the trajectory. Here the weight is computed as $w(s) = a_0(s)\mathrm{abs}(a_0(s) - a(s))/(1 + a_0(s))$.

can be computed as

$$w(s) = a_0(s)\frac{\mathrm{abs}(a_0(s) - a(s))}{(1 + a_0(s))}. \tag{A.16}$$

Where abs() stands for the absolute value of real numbers. The above defined $w(s)$ is continuous as composed of continuous functions ($a(s)$ and $a_0(s)$). It is positive at every s for $0 \leqslant s \leqslant 1$. When the triangle is not visible in the reference view, its area $a_0(s)$ is null, which makes its weight also zero. The more the triangle is deformed the bigger is $\mathrm{abs}(a_0(s) - a(s))$ which allows us to account more for deformed regions.

REFERENCES

1. Petit, A., Lippiello, V., Fontanelli, G.A., and Siciliano, B., Tracking elastic deformable objects with an RGB-d sensor for a pizza chef robot, *Robotics and Autonomous Systems*, 2017, vol. 88, pp. 187–201. https://doi.org/10.1016/j.robot.2016.08.023. Accessed 2021-06-17.
2. Haouchine, N., Dequidt, J., Peterlik, I., Kerrien, E., Berger, M.-O., and Cotin, S., Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery, *IEEE ISMAR*, 2013, pp. 199–208. https://doi.org/10.1109/ISMAR.2013.6671780
3. Rastegarpanah, A., Aflakian, A., and Stolkin, R., Optimized hybrid decoupled visual servoing with supervised learning, *Proc. Inst. Mech. Eng., Part I*, 2021, vol. 236, pp. 09596518211028379. https://doi.org/10.1177/09596518211028379. Accessed 2021-07-21.
4. Chi, C. and Berenson, D., Occlusion-robust deformable object tracking without physics simulation, *IEEE/RSJ IROS*, Macau, China, 2019, pp. 6443–6450. https://arxiv.org/abs/2101.007332101.00733. https://doi.org/10.1109/IROS40897.2019.8967827. Accessed 2021-06-29.
5. Lagneau, R., Shape Control of Deformable Objects by Adaptive Visual Servoing, *INSA de Rennes*, 2020. https://tel.archives-ouvertes.fr/tel-03087518. Accessed 2021-06-17.
6. Lagneau, R., Krupa, A., and Marchal, M., Active Deformation through Visual Servoing of Soft Objects, *IEEE ICRA*, Paris, France, 2020, pp. 8978–8984. https://doi.org/10.1109/ICRA40945.2020.9197506
7. Chaumette, F. and Ikeuchi, K. (ed.), Visual Servoing, *Computer Vision: A Reference Guide*, Springer, Boston, MA, 2014, pp. 869–874. https://doi.org/10.1007/978-0-387-31439-6_281
8. Tahri, O. and Chaumette, F., Image moments: generic descriptors for decoupled image-based visual servo, *IEEE ICRA*, 2004, vol. 2, pp. 1185–11902. https://doi.org/10.1109/ROBOT.2004.1307985
9. Agravante, D.J., Claudio, G., Spindler, F., and Chaumette, F., Visual servoing in an optimization framework for the whole-body control of humanoid robots, *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 608–615. https://doi.org/10.1109/LRA.2016.2645512
10. Ren, X., Li, H., and Li, Y., Image-based visual servoing control of robot manipulators using hybrid algorithm with feature constraints. *IEEE Access*, 2020, vol. 8, pp. 223495–223508. https://doi.org/10.1109/ACCESS.2020.3042207
11. Wang, T., Wang, W., and Wei, F., An overview of control strategy and trajectory planning of visual servoing / Fei, M., Li, K., Yang, Z., Niu, Q., and Li, X. (eds.) *Recent Featured Applications of Artificial Intelligence Methods. LSMS 2020 and ICSEE 2020 Workshops*, Springer, Singapore, 2020, pp. 358–370.
12. Corke, P.I. and Hutchinson, S.A., A new partitioned approach to image-based visual servo control, *IEEE Transactions on Robotics and Automation*, 2001, vol. 17, no. 4, pp. 507–515. https://doi.org/10.1109/70.954764.
13. Gans, N.R., Corke, P.I., and Hutchinson, S.A., Performance Tests for Visual Servo Control Systems, with Application to Partitioned Approaches to Visual Servo Control, *The International Journal of Robotics Research*, 2003, vol. 22, no. 10–11, pp. 955–981. https://doi.org/10.1177/027836490302210011. Accessed 2021-03-07.
14. Janabi-Sharifi, F. and Wilson, W.J., Automatic selection of image features for visual servoing, *IEEE Transactions on Robotics and Automation*, vol. 13, no. 6, pp. 890–903. https://doi.org/10.1109/70.650168
15. Chaumette, F., Image moments: a general and useful set of features for visual servoing, *IEEE Transactions on Robotics*, 2004, vol. 20, no. 4, pp. 713–723. https://doi.org/10.1109/TRO.2004.829463

16. Molná, C., Nagy, T.D., Elek, R.N., and Haidegger, T., Visual servoing-based camera control for the da vinci surgical system, *2020 IEEE 18th International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, Serbia, 2020, pp. 107–112. https://doi.org/10.1109/SISY50555.2020.9217086

17. Mohamed, I., MPPI-VS: Sampling-Based Model Predictive Control Strategy for Constrained Image-Based and Position-Based Visual Servoing, 2021. https://doi.org/10.48550/arXiv.2104.04925

18. Lagneau, R., Krupa, A., and Marchal, M., Active deformation through visual servoing of soft objects, *2020 IEEE International Conference on Robotics and Automation (ICRA)* Paris, France, 2020, pp. 8978–8984. https://doi.org/10.1109/ICRA40945.2020.9197506

19. Hu, Z., Han, T., Sun, P., Pan, J., and Manocha, D., 3-d deformable object manipulation using deep neural networks, *IEEE Robotics and Automation Letters*, 2019, vol. 4, no. 4, pp. 4255–4261. https://doi.org/10.1109/LRA.2019.2930476

20. Jia, B., Hu, Z., Pan, J., and Manocha, D., Manipulating highly deformable materials using a visual feedback dictionary, *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, Brisbane, QLD, Australia, 2018, pp. 239–246. https://doi.org/10.1109/ICRA.2018.8461264

21. Hu, Z., Sun, P., and Pan, J., Three-dimensional deformable object manipulation using fast online gaussian process regression, *IEEE Robotics and Automation Letters*, 2018, vol. 3, no. 2, pp. 979–986. https://doi.org/10.1109/LRA.2018.2793339

22. Zhu, J., Vision-based robotic manipulation of deformable linear objects, *PhD thesis*, Université Montpellier, 2020.

23. Chen, Z., Li, S., Zhang, N., Hao, Y., and Zhang, X., Eye-to-hand robotic visual tracking based on template matching on fpgas, *IEEE Access*, 2019, vol. 7, pp. 88870–88880. https://doi.org/10.1109/ACCESS.2019.2926807

24. Staneva, V. and Younes, L., Modeling and estimation of shape deformation for topology-preserving object tracking, *SIAM Journal on Imaging Sciences*, 2014, vol. 7, no. 1, pp. 427–455. https://doi.org/10.1137/130919714

25. Hu, Y., Carter, T.J., Ahmed, H.U., Emberton, M., Allen, C., Hawkes, D.J., and Barratt, D.C., Modelling prostate motion for data fusion during image-guided interventions, *IEEE Transactions on Medical Imaging*, 2011, vol. 30, no. 11, pp. 1887–1900. https://doi.org/10.1109/TMI.2011.2158235

26. Chen, Q., Sun, Q.-S., Heng, P.A., and Xia, D.-S., Two-stage object tracking method based on kernel and active contour, *IEEE Transactions on Circuits and Systems for Video Technology*, 2010, vol. 20, no. 4, pp. 605–609. https://doi.org/10.1109/TCSVT.2010.2041819

27. Cao, X., Lan, J., and Rong Li, X., Extension-deformation approach to extended object tracking, *2016 19th International Conference on Information Fusion (FUSION)*, Germany, 2016, pp. 1185–1192.

28. Joo, H., Simon, T., and Sheikh, Y., Total capture: A 3d deformation model for tracking faces, hands, and bodies, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Proc.*, 2018.

29. Royer, L., Marchal, M., Le Bras, A., Dardenne, G., and Krupa, A., Real-time tracking of deformable target in 3d ultrasound images, *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2430–2435. https://doi.org/10.1109/ICRA.2015.7139523

30. Royer, L., Krupa, A., Dardenne, G., Bras, A.L., Marchand, E., and Marchal, M., Real-time target tracking of soft tissues in 3d ultrasound images based on robust visual information and mechanical simulation, *Medical Image Analysis*, 2017, vol. 35, pp. 582–598. https://doi.org/10.1016/j.media.2016.09.004

31. Kajihara, K., Huang, S., Bergström, N., Yamakawa, Y., and Ishikawa, M., Tracking of trajectory with dynamic deformation based on dynamic compensation concept, *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macau, Macao, 2017, pp. 1979–1984. https://doi.org/10.1109/ROBIO.2017.8324709

32. Zhou, H., Ma, J., Tan, C.C., Zhang, Y., and Ling, H., Cross-weather image alignment via latent generative model with intensity consistency, *IEEE Transactions on Image Processing*, 2020, vol. 29, pp. 5216–5228. https://doi.org/10.1109/TIP.2020.2980210

33. Toriya, H., Dewan, A., and Kitahara, I., Sar2opt: Image alignment between multi-modal images using generative adversarial networks, *IGARSS 2019 – 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 923–926. https://doi.org/10.1109/IGARSS.2019.8898605

34. Kagami, S., Omi, K., and Hashimoto, K., Alignment of a flexible sheet object with position-based and image-based visual servoing, *Advanced Robotics*, 2016, vol. 30, no. 15, pp. 965–978. https://doi.org/10.1080/01691864.2016.1183518

35. Shen, Xi, Darmon, F., Efros, A., and Aubry M., Ransac-flow: Generic two-stage image alignment, *Computer Vision-ECCV 2020*, 2020, vol. 12349, pp. 618–637. https://doi.org10.1007/978-3-030-58548-8_36

36. Dong, Y., Liang, T., Zhang, Y., and Du, B., Spectral spatial weighted kernel manifold embedded distribution alignment for remote sensing image classification, *IEEE Transactions on Cybernetics*, 2021, vol. 51, no. 6, pp. 3185–3197. https://doi.org/10.1109/TCYB.2020.3004263

37. Mathiassen, K., Glette, K., and Elle, O.J., Visual servoing of a medical ultrasound probe for needle insertion, *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp 3426–3433. https://doi.org/10.1109/ICRA.2016.7487520

38. Mura, M., Abu-Kheil, Y., Ciuti, G., Visentini-Scarzanella, M., Menciassi, A., Dario, P., Dias, J., and Seneviratne, L., Vision-based haptic feedback for capsule endoscopy navigation: a proof of concept, *Journal of Micro-Bio Robotics*, vol. 11, pp. 35–45. https://doi.org/10.1007/s12213-016-0090-2. Accessed 2021-03-23.

39. Malti, A., Tax, M., and Lamiraux, F., A general framework for planning landmark-based motions for mobile robots, *Advanced Robotics*, 2011, vol. 25, no. 11–12, pp. 1427–1450. https://doi.org/10.1163/016918611X579457

40. Sengupta, A., Krupa, A., and Marchand, E., Visual Tracking of Deforming Objects Using Physics-based Models, *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 14178—14184. https://doi.org/10.1109/ICRA48506.2021.9561401

41. Feng, X., Mei, W., and Hu, D., A Review of Visual Tracking with Deep Learning, *AIIE 2016 Proc.*, 2016, pp. 231–234. https://doi.org/10.2991/aiie-16.2016.54

42. Marvasti-Zadeh, S.M., Cheng, L., Ghanei-Yakhdan, H., and Kasaei, S., Deep Learning for Visual Tracking: A Comprehensive Survey, *IEEE Transactions on Intelligent Transportation Systems*, 2021, vol. 23, no. 5, pp. 3943–3968. https://doi.org/10.1109/TITS.2020.3046478

43. Malti, A., Hartley, R., Bartoli, A., and Kim, J.-H., Monocular template-based 3d reconstruction of extensible surfaces with local linear elasticity, *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, 2013, pp. 1522–1529, https://doi.org/10.1109/CVPR.2013.200

44. Malti, A., Bartoli, A., and Hartley, R., A linear least-squares solution to elastic shape-from-template, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1629–1637. https://doi.org/10.1109/CVPR.2015.7298771

45. Malti, A. and Herzet, C., Elastic shape-from-template with spatially sparse deforming forces, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 143–151. https://doi.org/10.1109/CVPR.2017.23

46. Casillas-Perez, D., Pizarro, D., Fuentes-Jimenez, D., Mazo, M., and Bartoli, A., Equiareal Shape-from-Template, *Journal of Mathematical Imaging and Vision*, 2019, vol. 61, no. 5, pp. 607–626.

47. Ben-Israel, A. and Greville, T.N.E., *Generalized Inverses Theory and Applications*, New York: Springer, 2003. Google-Books-ID: o_zXUXaqGU8C.

48. Saidi, F. and Malti, A., Fast and accurate nonlinear hyper-elastic deformation with a posteriori numerical verification of the convergence of solution: Application to the simulation of liver deformation, *Int. J. Numer. Meth. Biomed. Engng.*, 2021, 37:e3444. https://doi.org/10.1002/cnm.3444. Accessed 2021-09-28.

49. Chaumette, F. and Hutchinson, S., Visual servo control. i. basic approaches, *IEEE Robotics Automation Magazine*, vol. 13, no. 4, pp. 82–90. https://doi.org/10.1109/MRA.2006.250573

50. Murray, R.M., Li, Z., and Sastry, S.S., *A Mathematical Introduction to Robotic Manipulation*, Boca Raton, CRC Press, 1994. https://doi.org/10.1201/9781315136370

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