ISSN 2782-2427

CONTROL SCIENCES

5/2022

[1]Russian Academy of Sciences.

CONTROL SCIENCES
Scientific Technical Journal

6 issues per year
ISSN 2782-2427
Open access

Published since 2021

Original Russian Edition
Problemy Upravleniya
Published since 2003

FOUNDER AND PUBLISHER
V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences

Editorial address
65 Profsoyuznaya st., office 410,
Moscow 117997, Russia

☎/🖹 +7(495) 198-17-20, ext. 1410

✉ pu@ipu.ru

URL: http://controlsciences.org

Published: December 14, 2022

Registration certificate of
Эл № ФС 77-80482
of 17 February 2021
issued by the Federal Service for Supervision of Communications, Information Technology, and Mass Media

CONTROL SCIENCES

5.2022

CONTENTS

DOI: http://doi.org/10.25728/cs.2022.5.1

DESIGNING AN ADAPTIVE STABILIZING SYSTEM FOR AN UNMANNED AERIAL VEHICLE

N.A. Pervushina[1] and A.D. Frolova[2]

Zababakhin All-Russian Research Institute of Technical Physics (VNIITF),
Russian Federal Nuclear Center, Snezhinsk, Russia

[1]✉ p-n-a100678@yandex.ru, [2]✉ antonina.garishina@yandex.ru

Abstract. This paper presents a mathematical model of an efficient adaptive stabilizing system in the pitch channel of an unmanned aerial vehicle. The model is described by a functional block diagram and is based on a correction method proposed for onboard computers. Some structural modifications are suggested for the correction loop to improve the performance of the stabilizing system of the nonlinear dynamic item under control mode switching. The operation of the stabilizing system is simulated with the tuned parameters of the correction loop under fixed gains of the main loop. The new structure of the correction loop in the stabilizing system demonstrates high efficiency in the operation modes of the vehicle. Due to the proposed design procedure, the stabilizing system with the new structure of the correction loop is constructed several times faster compared with the classical method of fixed factors.

Keywords: unmanned aerial vehicle, pitch channel, stabilizing system, mathematical modeling, design, the efficiency of a stabilizing system.

INTRODUCTION

The performance and operability of a stabilizing system (SS) of an unmanned aerial vehicle (UAV) in a domain corresponding to its admissible application conditions are determined by the aerodynamic properties of the guided item and the chosen structure of the control signal. The traditional solution of the SS design problem with constructing a linearized model of the UAV for each flight mode and approximating the resulting coefficients depending on the dynamic head by the method of fixed factors ensures system operation in the entire range of flight modes [1, 2] but requires computing cost. Therefore, finding the most efficient ways of solving this problem is topical. The efficiency of a system or process is usually understood as the ratio of the result achieved and the resources used.

This paper considers airplane-type guided UAVs with the normal aerodynamic scheme, a large elongation wing, and differential rudders. The object of study is the longitudinal control channel (pitch channel) in the UAV stabilizing system. The goal of study is to develop a mathematical model of an effective adaptive stabilizing system in the pitch channel. The effectiveness of the SS will be understood as system operation in switching modes with the highest possible performance in the entire range of flight modes with the minimum computing cost of the design procedure. A stabilizing system with these properties will be called efficient.

There are ways to improve the performance of the SS by increasing its speed. A term corresponding to some additional impact on the controls is introduced in the control signal structure of the pitch channel of the UAV. In [3–7], the control signal in the pitch channel was formed by an overload discrepancy, an integral of discrepancy, a signal proportional to the pitching rate (damper), i.e., an analog of a proportional-integral-differential (PID) controller, and an additional balancing signal or a signal similar to the pilot force applied to the stick. As shown in [9], when the aerial vehicle reaches a given altitude, the automatic thrust force control is implemented by a similar law. In [10], the control signal applied to the elevators was also formed with an additional term proportional to the angle of attack. A similar technique was used to improve the performance of the overload stabilization loop in the coefficient adaptation algorithm [11]. These examples of control signals in the pitch channel allow improving the quality of transients (increasing the speed). However, for the SS to operate in the entire range of admissible modes, we still need a specific set of system gains for each nominal point of this range.

Modern UAVs are designed to operate in various, particularly extreme, flight modes. The most difficult and dangerous modes are the critical ones in terms of the angle of attack. If the critical angles of attack are exceeded, the control efficiency of the UAV decreases. The paper [12] proposed a two-loop stabilizing system in the pitch channel where the auxiliary loop limits the angle of attack through an algebraic selector: it changes the system structure in accordance with the channel switching logic. This approach yields a well-damped system, but significantly restricting the angle of attack increases the transient time.

A mathematically justified method for designing linear systems with the minimum settling time was presented in [13]. The time-optimal solution was obtained using the theory of optimal controllers.

Fuzzy controllers are often employed to stabilize dynamic plants in modern control systems. This class of controllers has low sensitivity to changes in the plant's parameters and is characterized by high speed and accurate positioning [14, 15]. According to computer simulations [16], fuzzy controllers used for stabilizing the UAV in the pitch channel demonstrate high speed. However, such controllers are difficult to describe due to a rule base developed for the input parameters.

Currently, there are many examples of fuzzy controllers improving the performance of control systems [17–21]. Fuzzy controllers do increase the speed of systems, but they require additional tuning depending on the operation mode and under control mode switching.

In this paper, we construct a mathematical model of an efficient SS based on the approach described in the monograph [2]. This approach requires no parameter changing and involves the method of fixed factors. The authors [2] proposed a fast reduction of the control error using an additional signal of an appropriate sign when the stabilization error exceeds a certain threshold.

The results presented below were obtained by computer modeling. The flight of an aerial vehicle in the Earth's atmosphere is described by a system of nonlinear ordinary differential equations with coefficients that depend on free stream parameters.

The following problems are sequentially solved in this paper:

– Mathematical models of the guided item and SS are developed.

– The mathematical model of the SS is designed, and the correction loop parameters are tuned.

– Some structural modifications are suggested for the correction loop to improve the performance of the SS. The operation of the SS with the new correction loop is tested.

1. PROBLEM STATEMENT

To develop a mathematical model of an efficient adaptive stabilizing system, we choose the classical structure of the SS in the pitch channel [19] with the control signal

$$\sigma_{\text{elev}} = K_i \int_{t_0}^{t} \Delta n_y dt + K_n \Delta n_y - K_{\omega_z} \omega_z \qquad (1)$$

and the following notations: $\Delta n_y = n_{y\,\text{giv}} - n_y$ is the discrepancy value, where $n_{y\,\text{giv}}$ is a given value of the normal overload and n_y is the SS output; σ_{elev} is the elevator control signal, in deg; K_i, K_n, and $K\omega_z$ are known gains of appropriate dimension; finally, ω_z is the pitching rate, in deg/s.

We form the additional control signal σ_{add} using a functional analog of the pulse correction scheme that can be implemented in digital onboard systems [2]. Consider a scheme consisting of an integrator and an aperiodic link that are connected in parallel with the main stabilization loop when the stabilization error exceeds a given threshold. Connecting the scheme to the stabilization error signal Δn_y and the pitching rate signal ω_z reduces the overshoot and the rate of oscillation due to a small error threshold when increasing the system speed.

Figure 1 shows the block diagram of the pitch channel stabilizing system with the control signal (1) when connecting the correction loop as recommended in [2].

Figure 1 has the following notations: σ_{elev} is the equivalent elevator control signal, in deg; δ_{elev} is the equivalent elevator angle, in deg; δ_{add} is the additional control signal during the correction, in deg; σ_{thres} is the threshold control signal, in deg; K_n, K_ω, K_1, K_2, and K_3 are the appropriate-dimension gains of the auxiliary loop of the stabilizing system; T is the time constant of the aperiodic link in the correction loop, in s; sign(σ_{thres}) is the sign function [20]; finally, p is the Laplace transform variable.

The guided item is described by a nonlinear mathematical model in the aircraft-linked coordinate system [1, 19]:

$$\begin{cases} \dfrac{d\omega_z}{dt} = \dfrac{m_z qSL}{I_{zz}} \times \dfrac{180}{\pi}, \\ \dfrac{d\alpha}{dt} = \left(\dfrac{g\cos\vartheta}{V} - \dfrac{c_y qS}{mV} \right) \dfrac{180}{\pi} + \omega_z, \\ \dfrac{d\vartheta}{dt} = \omega_z, \end{cases} \quad (2)$$

where α is the angle of attack, in deg; ϑ is the pitch angle, in deg; $m_z = m_z(\mathrm{M}, \alpha, \delta_{elev}, ...)$ is the aerodynamical coefficient of the pitching moment, which nonlinearly depends on the free stream parameters, the elevator angle, etc., a dimensionless quantity; $c_y = c_y(\mathrm{M}, \alpha, \delta_{elev}, ...)$ is the aerodynamical lifting force coefficient, which nonlinearly depends on the free stream parameters, the elevator angle, etc., a dimensionless quantity; M is the Mach number; q is the dynamic head, in Pa; S is the midship area, in m^2; L is the characteristic linear dimension of the aerial vehicle, in m; m is the aerial vehicle mass, in kg; V is the airspeed of the aerial vehicle, in m/s; finally, g = 9.80665 m/s^2 is free fall acceleration.

In this problem, the actuator of the elevator has the linear mathematical model

$$T_A \frac{d\delta_{elev}}{dt} + \delta_{elev} = \sigma_{elev}, \quad (3)$$

where T_A = 0.025 s is the time constant of the actuator.

For example, consider five nominal points of the aerial vehicle trajectories (Table 1).

Table 1

Nominal points of trajectories

Parameters, units of measurement	Nominal points				
	1	2	3	4	5
q, kPa	5.5	10.5	20	30.5	44
V, m/s	100	160	200	250	270

The performance requirements for the stabilizing system in the entire range of nominal points of the aerial vehicle trajectories are as follows:

• The time t_{set} to execute a given control step excitation (5% of the steady-state value) must be about 1.5 s.

• The overshoot σ in the transients at the SS output must be minimum, not exceeding 30%.

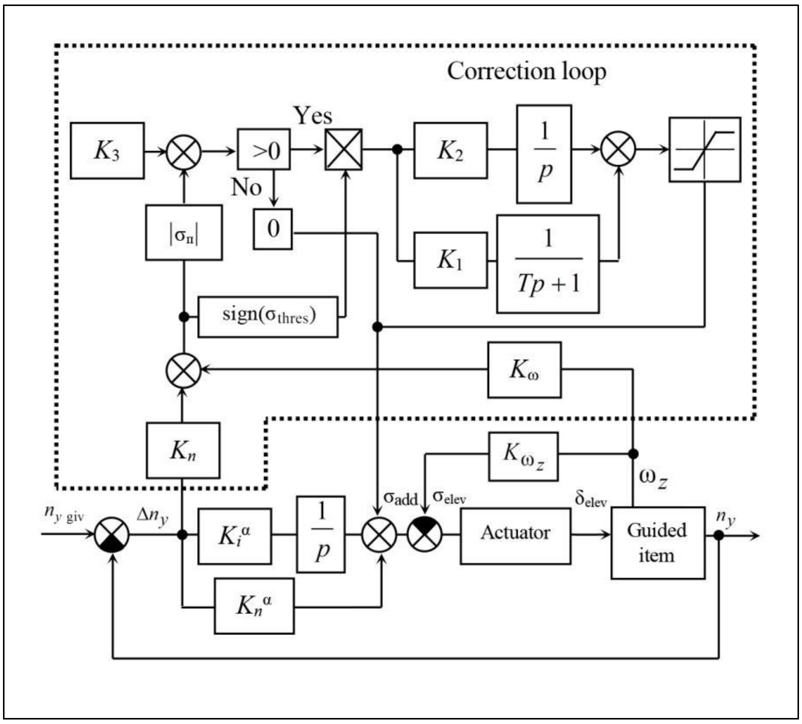


Fig. 1. The block diagram of the SS in the pitch channel with the correction loop.

• The maximum value of the control signal σ_{elev} in the SS must not exceed ±20°.

• At the design stage, the gain margin L_{mar} must be at least 10 dB.

• At the design stage, the phase margin φ_{mar} must be at least 30°.

2. DESIGNING THE MAIN LOOP OF THE STABILIZING SYSTEM

The system of differential equations (2) is linearized in the neighborhood of each nominal point; see Table 1. For linearization, we simplify the aerodynamical coefficients model:

$$c_y = c_{y_0} + c_y^\alpha \alpha + c_y^{\delta_{elev}} \delta_{elev},$$

$$m_z = m_z^\alpha \alpha + m_z^{\delta_{elev}} \delta_{elev}, \quad (4)$$

with the following notations: c_{y_0} is the aerodynamical coefficient c_y for α = 0 and δ_{elev} = 0; c_y^α is the derivative of the coefficient c_y with respect to the angle of attack, in deg^{-1}; $c_y^{\delta}{}_{elev}$ is the derivative of the coefficient c_y with respect to the elevator angle, in deg^{-1}; m_z^α is the derivative of the coefficient m_z with respect to the angle of attack, in deg^{-1}; finally, $m_z^{\delta}{}_{elev}$ is the derivative of the coefficient m_z with respect to the elevator angle, in deg^{-1}.

The relationship between the overload and the angle of attack [21] is described by

$$n_y = \left(c_y^{\alpha} Sq / mg\right)\alpha. \quad (5)$$

For $\triangle\vartheta \approx 0$, the linearized system of differential equations has the form

$$\begin{cases} \dfrac{d\omega_z}{dt} = m_z^{\alpha} \dfrac{qSL\times 180}{I_z \pi}\alpha + m_z^{\delta_{\text{elev}}} \dfrac{qSL\times 180}{I_z \pi}\delta_{\text{elev}}, \\ \dfrac{d\alpha}{dt} = -c_y^{\alpha} \dfrac{qS\times 180}{mV\pi}\alpha - c_y^{\delta_{\text{elev}}} \dfrac{qS\times 180}{mV\pi}\delta_{\text{elev}} + \omega_z. \end{cases} \quad (6)$$

Under the above requirements, we designed the main loop of the SS in the pitch channel (Fig. 1) using the method of amplitude-log responses [20] for the guided item (6). The resulting gains K_i, K_n, and $K\omega_z$ of the main loop as well as the values of the performance indicators and stability margins are combined in Table 2.

The transients in the main loop of the SS with the linearized guided item (4) are shown in Fig. 2. In the graphs of Figs. 2 and 3, the results for point 1 are indicated by a black bold line; for point 2, by a black thin line; for point 3, by a black dashed line; for point 4, by a gray bold line; for point 5, by a gray dashed line.

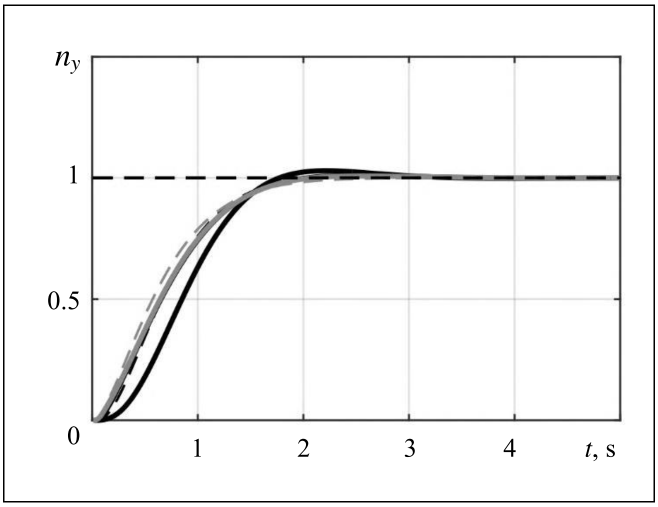


Fig. 2. Transients in the main loop of the SS in the pitch channel.

Table 2

The main loop of the SS

Nominal points	SS gains			Performance indicators		Stability margins	
	K_i	K_n	$K\omega_z$	t_{set}, s	σ, %	L_{mar}, dB	φ_{mar}, °
1	– 10.86	– 0.098	– 0.239	1.50	1.4	14	66
2	– 6.488	– 0.889	– 0.290	1.55	1.0	31	71
3	– 2.796	– 0.051	– 0.175	1.50	0.1	21	71
4	– 2.289	– 0.176	– 0.259	1.55	0.5	36	73
5	– 1.702	– 0.092	– 0.178	1.48	0.1	35	74

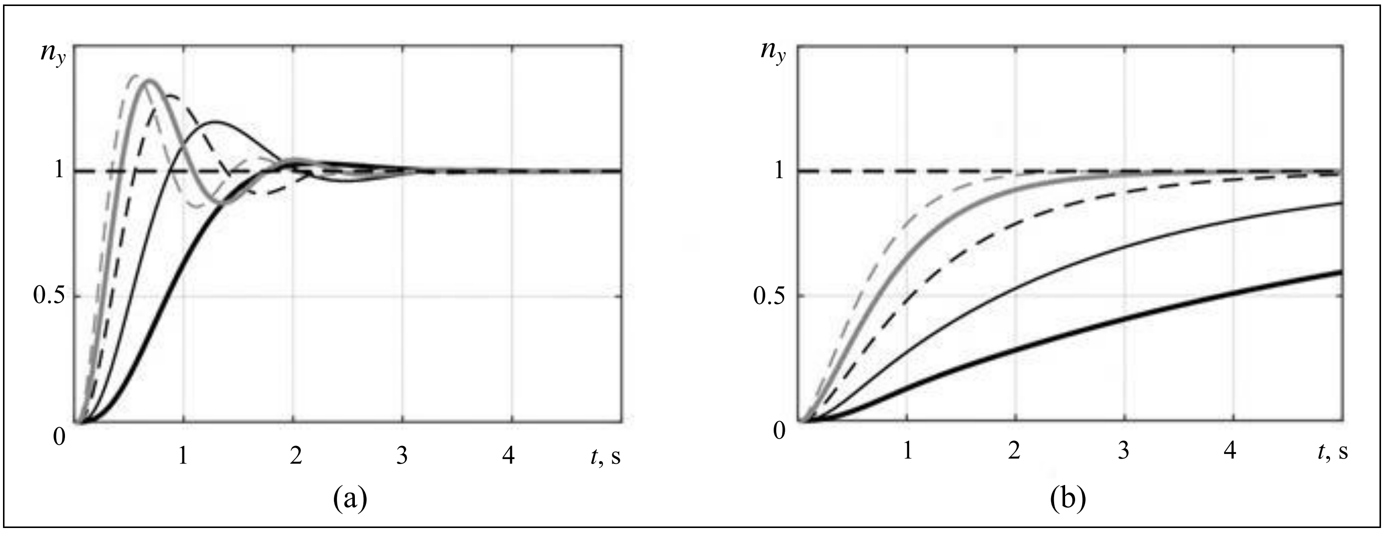


Fig. 3. Transients in the main loop of the SS in the pitch channel: (a) the gains K_i = – 10.86, K_n = – 0.098, and $K\omega_z$ = – 0.239 (point 1) and (b) the gains K_i = – 1.702, K_n = – 0.092, and $K\omega_z$ = – 0.178 (point 5).

This design procedure of the main loop of the SS in the pitch channel at each nominal point of the UAV trajectory illustrates the method of fixed factors. Further, the resulting coefficients are traditionally approximated depending on the dynamic head: $K_i(q)$, $K_n(q)$, $K\omega_z(q)$. The result is an adaptive SS.

This design procedure is computationally intensive due to considering each nominal point. Indeed, the SS gains vary significantly; see Table 2. For instance, the coefficient $K_i = -10.86$ for point 1 and $K_i = -1.702$ for point 5 differ almost tenfold.

As an example, Figs. 3a and 3b show the simulated operation of the SS with the linearized guided item (3) and the gains corresponding to the extreme points (1 or 5).

At nominal points with a small dynamic head (point 1), a large control action is required for stabilization. In other modes, however, this control action turns out excessive (Fig. 3a). For instance, when simulating the operation of the SS at nominal point 5, the overshoot reaches 40%, which is unacceptable. At nominal points with a high dynamic head (point 5), such a control action becomes unnecessary. In this case, when simulating the operation of the SS at nominal point 1, the transient time increases manifold; therefore, the control action is insufficient (Fig. 3b).

We use a correction loop (Fig. 1) to increase the efficiency of the SS (stabilize the pitch angle in the entire range of nominal points using only one set of gains).

3. TUNING THE CORRECTION LOOP

The recommendations provided in [2], semi-empirical in nature, can underlie a tuning procedure for the parameters of the correction loop (the auxiliary loop) depending on the dynamic properties of the UAV under consideration. According to these recommendations, the time constant T of the aperiodic link in the correction loop (Fig. 1) is related to the coefficient K_2 at the integrator. More precisely, the smaller this constant is, the greater value the coefficient will take. A larger value of the coefficient increases the speed of discrepancy processing but can quickly bring the scheme to saturation, so the choice of the time constant T is determined by the performance requirements for a particular system. In an initial approximation, the time constant T should be taken 10–15 times greater than the time constant T_A (3) of the actuator. We restrict the control signal of the correction loop to ±10° (50% of the maximum value of the control signal according to the SS requirements). The other parameters of the correction loop are selected depending on the properties of the guided item and actuator and the operation modes of the SS.

3.1. Tuning under the Single Step Excitation

The parameters of the auxiliary control loop were tuned under a typical single step excitation of the form

$$n_{y\ \text{giv}} = 1(t) = \begin{cases} 0 \text{ for } t < 0, \\ 1 \text{ for } t \geq 0. \end{cases} \quad (7)$$

Let the main loop of the SS be designed for the nominal point with the smallest dynamic head (point 1). In this case, the auxiliary loop is intended to reduce the control action: the other nominal points have a greater value of the dynamic head, and this control action will be excessive for them (Fig. 3a).

In accordance with the above recommendations, the correction loop was configured to work the influence (7). The values of the loop parameters after tuning are shown in Table 3.

Table 3

Correction loop parameters

Parameters	T	K_1	K_2	K_3	K_n	K_ω
Values	0.25	0.1	10	0.5	0.1	– 0.25

In the scheme of Fig. 1, the signal σ_{add} enters the main loop of the SS with the minus sign. The simulated operation of the SS with the additional loop is shown in Fig. 4 (bold line).

According to the simulation results, the gains of the main loop obtained for point 1 work with the correction loop in the system with the guided item's mathematical model for point 5 and the input (5).

Now we test the SS in the case when the input amplitude differs from 1. The simulation results are presented Figs. 5a and 5b. The graphs indicate that the correction result depends on the input value of the SS. However, this feature is not stipulated by the correction loop scheme selected in the paper (Fig. 1).

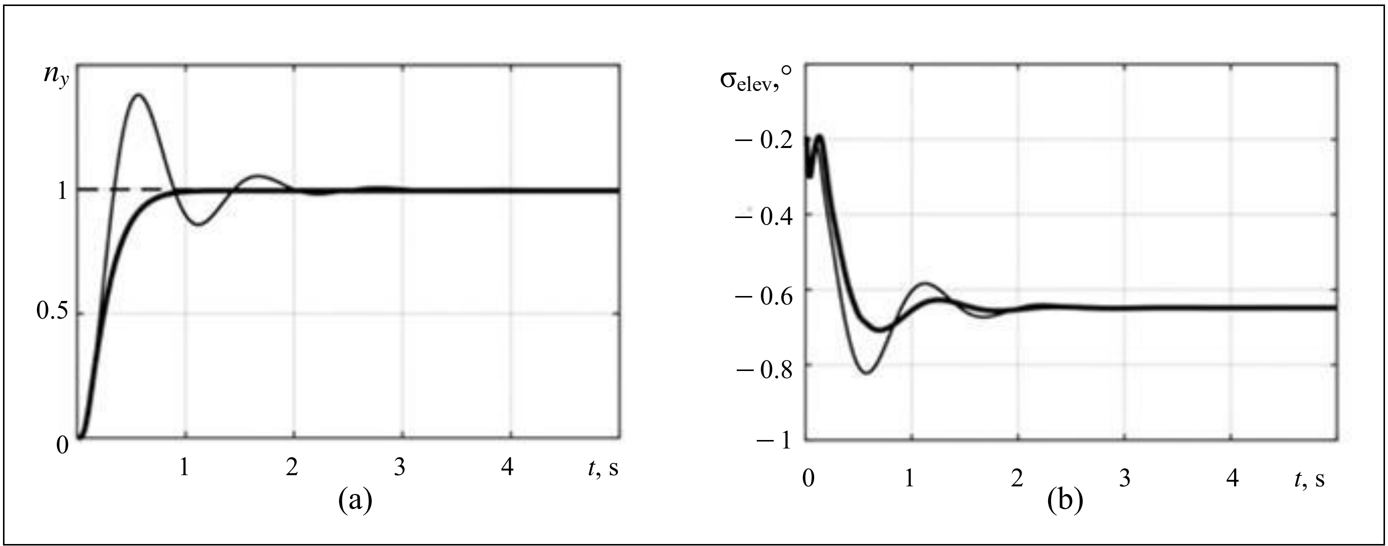


Fig. 4. Parameters of the SS in the pitch channel before and after correction: (a) n_y and (b) σ_{elev}.

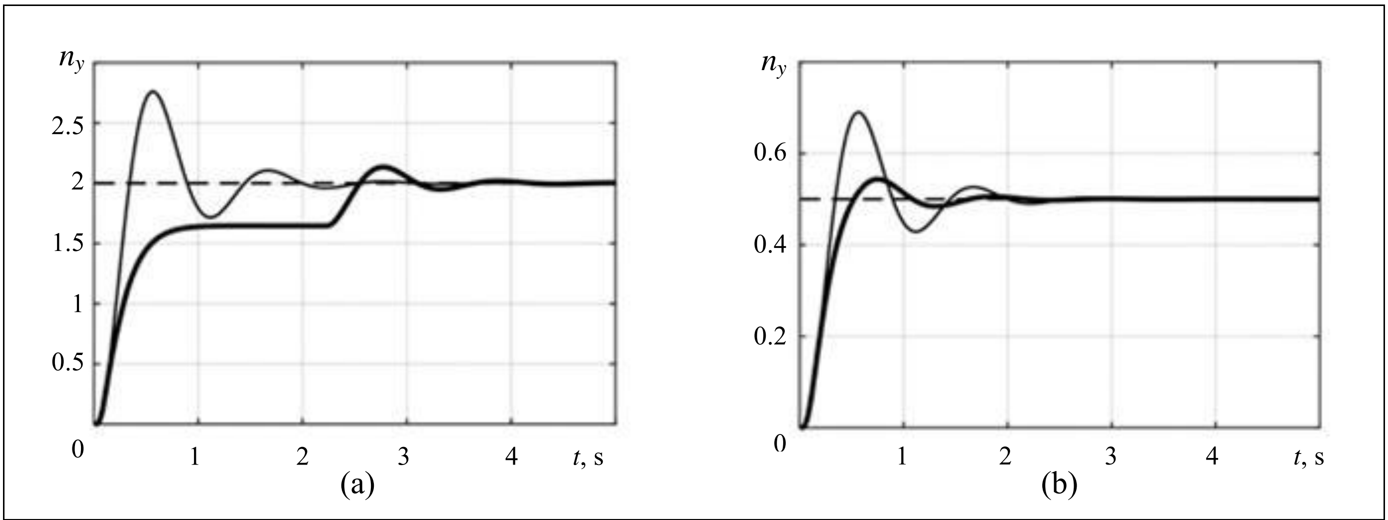


Fig. 5. Output processes of the SS in the pitch channel: (a) $n_{y\ giv} = 2\cdot 1(t)$ and (b) $n_{y\ giv} = 0.5\cdot 1(t)$.

3.2. Structural Modifications in the Correction Loop

In view of the aforesaid, we suggest considering $n_{y\ giv}$ as an input signal of the correction loop of the SS. This structural modification of the auxiliary loop is shown in Fig. 6.

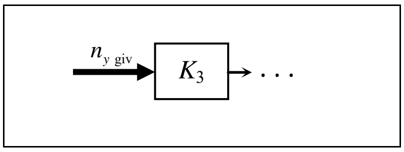


Fig. 6. The suggested modification in the correction loop scheme.

3.3. Simulation of the Stabilizing System at Each Nominal Point

Let us simulate the operation of the modified SS (Fig. 6) at the nominal points (Table 1). The simulation results are demonstrated in Fig. 7 except for point 5 (loop tuning; see Fig. 4).

According to these graphs, the additional control signal σ_{add} worsens the transients when passing from one nominal point to another with a decrease in the dynamic head (i.e., when the main loop of the SS approaches its initial settings). In other words, the SS loop with the gains obtained at the design stage (for

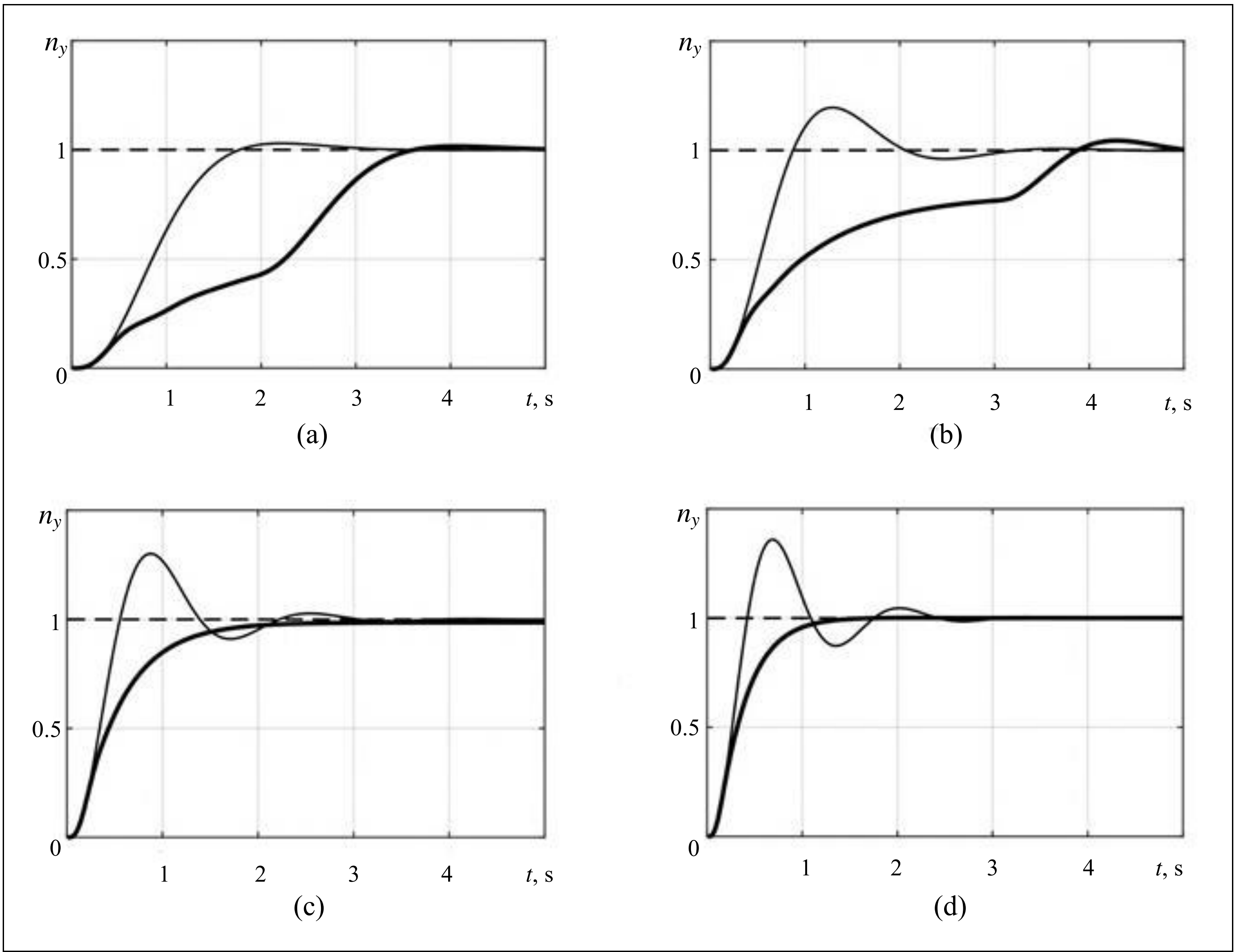


Fig. 7. Output processes of the modified SS in the pitch channel: (a) point 1 (q = 5.5 kPa), (b) point 2 (q = 10.5 kPa), (c) point 3 (q = 20 kPa), and (d) point 4 (q = 30.5 kPa).

point 1) needs no additional correction. For the other points (2, 3, and 4), it is necessary to gradually increase the correction effect and, hence, the signal σ_{add}.

3.4 Tuning the Control Signal σ_{add}

To regulate the degree of correction, we suggest a dimensionless coefficient K that depends on the dynamic head q as follows: if q = 44 kPa (max), then K = 1; if q = 5.5 kPa (min), then K = 0. Let the coefficient K change linearly between the nodal points. The linear approximation of K is given by

$$K(q) = 0.026q - 0.143, \quad (8)$$

where q is the dynamic head, in kPa.

Considering (8), the signal σ_{add} takes the form

$$\sigma_{add} = K(q)\,\sigma^*_{add}, \quad (9)$$

where σ^*_{add} is the control signal of the correction loop before the tuning procedure (8).

Figure 8 shows the simulation results of the SS with the control signal (9) for points 1 and 5. These graphs are the counterparts of the ones presented in Fig. 7.

3.5. Simulation of the Stabilizing System with the Nonlinear Guided Item

Next, we simulate the operation of the SS under the following conditions:

1) The guided item in the pitch channel is described by the system of nonlinear differential equations (2).

2) The aerodynamical coefficients model is:

2.1) linear (4),

2.2) nonlinear.

3) The control signal in the main loop has the structure (1).

4) The control signal in the auxiliary loop has the structure shown in Fig. 1.

5) The actuator model is described by the differential equation (3), written in difference form as

$$\delta_{\mathrm{elev}_i} = a\sigma_{\mathrm{elev}_{i-1}} + b\delta_{\mathrm{elev}_{i-1}},$$

where $a = h / T_{\mathrm{A}}$ and $b = e^{-h/T_{\mathrm{A}}}$.

6) The relation between the overload and the angle of attack is described by formula (5).

7) The system of equations (2) is integrated using the fourth-order Runge–Kutta method with an integration step h, a conventional approach in the numerical modeling of aircraft flight dynamics.

Figure 9 shows the simulation results under conditions 1)–7) for the linear aerodynamical coefficients model with $h = 0.01$ s.

In Fig. 9, the results for point 1 are indicated by a black bold line; for point 2, by a black thin line; for point 3, by a black dotted line; for point 4, by a gray bold line; for point 5, by a gray thin line.

The values of the integral performance criteria of the processes in Fig. 9 are combined for comparison in Table 4. The values of the integrated square error (ISE) and the integrated weighted absolute error (IWAE) decreased significantly during the execution of the single step excitation. For instance, for the extreme nominal point 5, the IWAE value decreased by 4.5 times.

The performance of transients in the SS was also assessed by the settling time t_{set} and the overshoot σ. The estimated values of these indicators are given in Table 5.

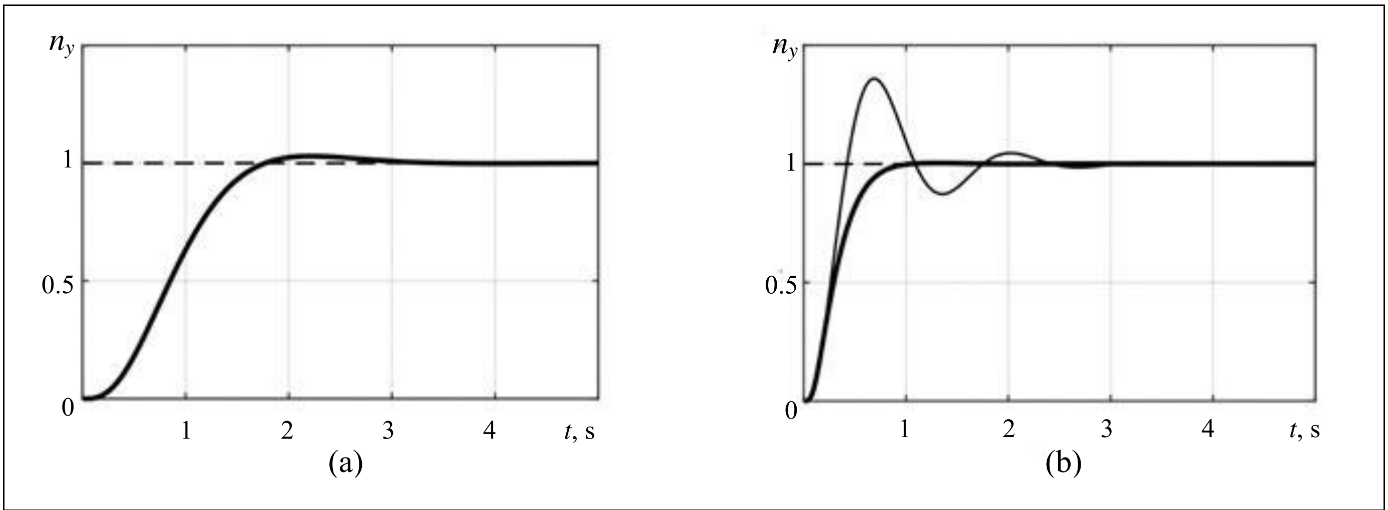


Fig. 8. Output processes of the SS with the tuned control signal σ_{add}: (a) point 1 (q = 5.5 kPa) and (b) point 4 (q = 30.5 kPa).

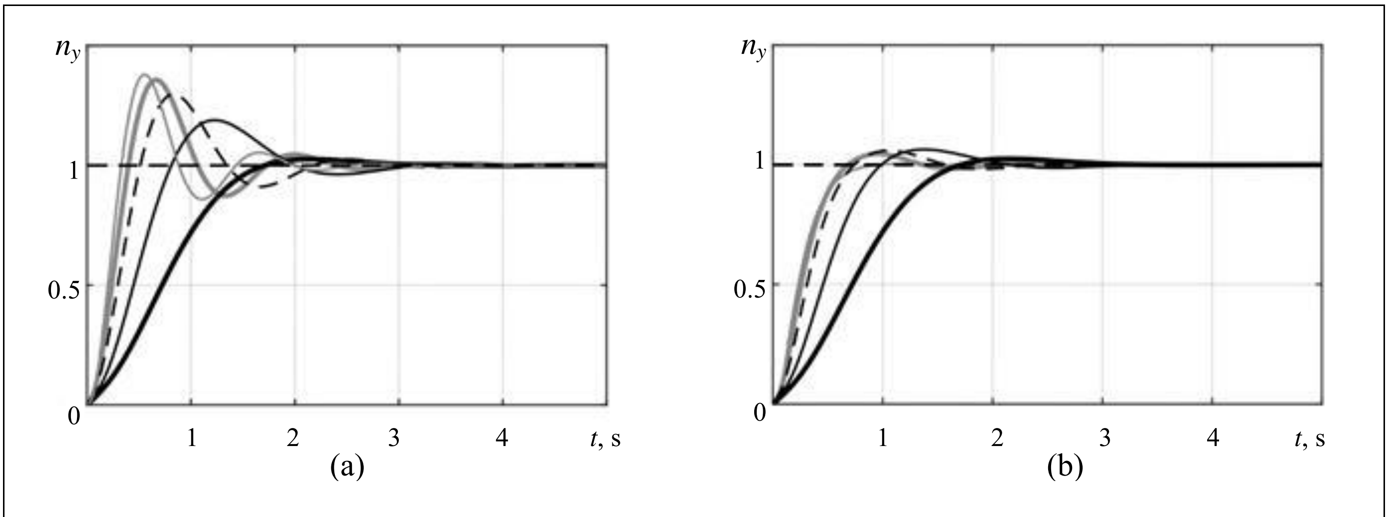


Fig. 9. Output processes of the SS: (a) before correction and (b) after correction.

Table 4

Integral performance criteria

Nominal point	Without correction		With correction	
	$\text{ISE} = \int_0^T \left(n_{y\text{ giv}} - n_y\right)^2 dt$	$\text{IWAE} = \int_0^T t\left\|n_{y\text{ giv}} - n_y\right\| dt$	ISE	IWAE
1	0.479	0.409	0.479	0.409
2	0.329	0.414	0.312	0.236
3	0.248	0.338	0.213	0.124
4	0.218	0.289	0.178	0.082
5	0.191	0.220	0.160	0.047

Table 5

Performance indicators of transients

Nominal point	Indicators			
	Without correction		With correction	
	t_{set}, s	σ, %	t_{set}, s	σ, %
1	2.1	6	2.1	6
2	2.4	24	1.6	9
3	1.9	34	1.2	7
4	2	40	0.9	5
5	1.7	42	0.6	0

For instance, for nominal point 5, the time t_{set} decreased by 3 times and the overshoot σ decreased by 42% during the execution of the single step excitation.

Figure 10 presents the simulation results for the nonlinear aerodynamical coefficients model. The range of the dynamic head q on the simulated trajectory was from 25 to 6.5 kPa. The gains of the control signal (1) in the main loop were fixed and corresponded to nominal point 1: $K_i = -10.86$, $K_n = -0.098$, and $K\omega_z = -0.239$. The gains of the auxiliary loop were taken from Table 3.

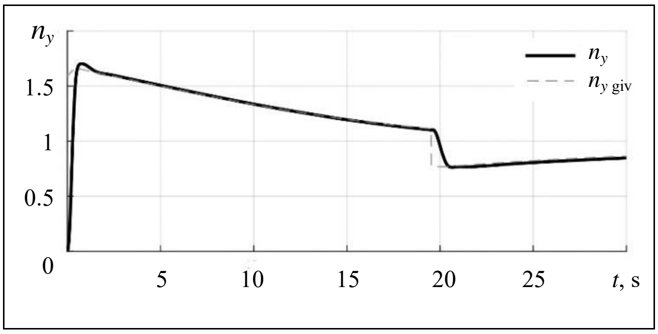


Fig. 10. The output process of the SS with the correction loop for the nonlinear aerodynamical coefficients model.

For comparison, Fig. 11 shows the simulated operation of the SS without the correction loop with the gains calculated depending on the dynamic head q according to Tables 1 and 2. Figure 12 demonstrates the deviations of the parameter n_y from the given values $n_{y\text{ giv}}$ for the two types of stabilizing systems.

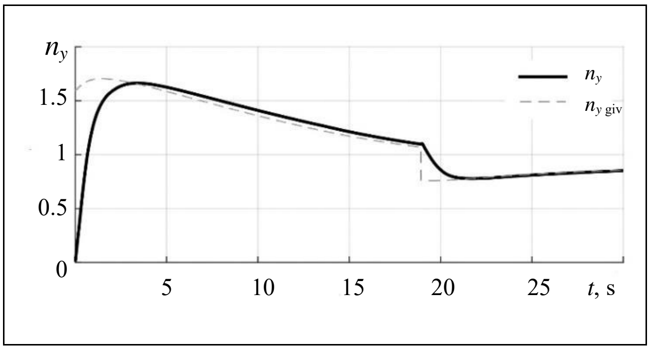


Fig. 11. The output process of the SS without the correction loop for the nonlinear aerodynamical coefficients model.

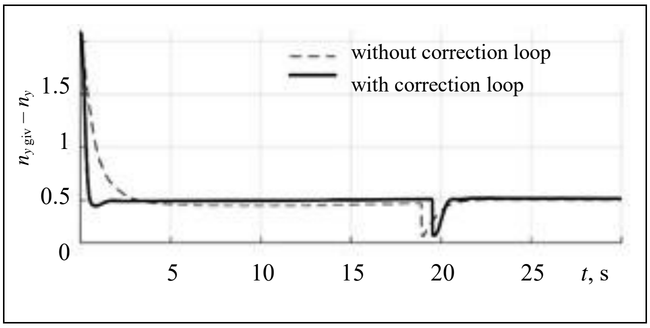


Fig. 12. The deviations of the parameter n_y from the given values $n_{y\text{ giv}}$ for the nonlinear aerodynamical coefficients model.

As follows from Fig. 12, the developed adaptive stabilizing system rapidly reduced the error. These results indicate the efficiency of the developed SS, the correct design procedure of the main loop, and the correct parametric tuning of the auxiliary loop. The proposed procedure works for the angles of attack within the range of ±24°. The absolute value 24° of the angle of attack is the limit for the UAVs under consideration.

Figure 13 presents the new block diagram of the pitch channel correction scheme based on the refinements and modifications suggested in this paper (shown in bold).

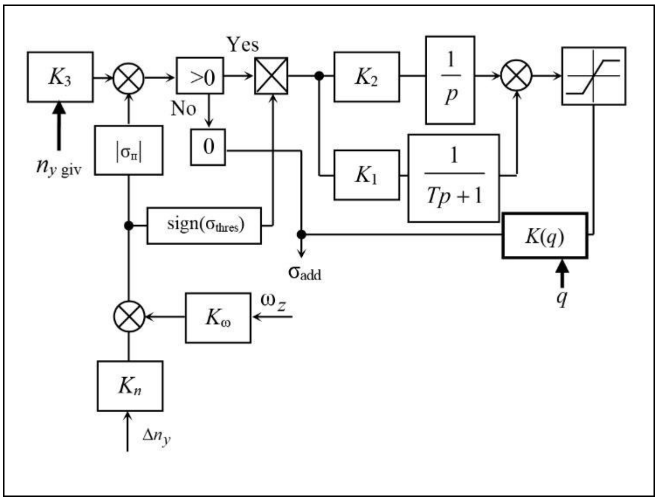


Fig. 13: The block diagram of the correction loop of the SS in the pitch channel.

CONCLUSIONS

This paper has presented a mathematical model of an efficient adaptive stabilizing system (SS) in the pitch channel of an unmanned aerial vehicle (UAV). The model is based on the correction method proposed for onboard computers.

The linearized mathematical model of the UAV has been simulated under different single step excitations. According to the simulation results, the performance of the SS worsens if the amplitude of the input excitation differs from the one used for tuning the correction loop. Therefore, the following structural modifications have been proposed for the auxiliary loop:

– treating a given excitation as an input signal,

– introducing a coefficient that depends on the dynamic head to regulate the degree of correction.

Computer simulations have been carried out with integration by the fourth-order Runge–Kutta method, and the values of integral performance criteria and performance indicators have been calculated. Based on these results, the stabilizing system with the new correction loop demonstrates a good quality of stabilization: no overshoot and high settling times of up to 0.6 s.

The mathematical model of the SS has been implemented in the program code. According to the simulation results, due to the proposed design procedure, the stabilizing system with the new structure of the correction loop is constructed several times faster compared with the classical method of fixed factors: the classical method is applied only for one nominal point (with the minimum dynamic head) from the set of admissible flight modes of the UAV. The constant coefficients of the correction loop are then tuned to ensure the operation of the SS in the entire range of flight modes (up to the maximum dynamic head), and the tuning procedure requires insignificant time compared with the main loop design procedure.

The design approach proposed above is recommended for developing digital adaptive stabilizing systems in the pitch channel of unmanned aerial vehicles to improve the quality of stabilization in a set of their operation modes while reducing the associated computing cost.

REFERENCES

1. Lebedev, A.A. and Karabanov, V.A., *Dinamika sistem upravleniya bespilotnymi letatel'nymi apparatami* (Dynamics of Control Systems for Unmanned Aerial Vehicles), Moscow: Mashinostroenie, 1965. (In Russian.)
2. Obnosov, B.V., Voronov, E.M., Mikrin, E.A., et al., *Stabilizatsiya, navedenie, gruppovoe upravlenie i sistemnoe modelirovanie bespilotnykh letatel'nykh apparatov. Sovremennye podkhody i metody* (Stabilization, Guidance, Group Control, and System Simulation of Unmanned Aerial Vehicles. Modern Approaches and Methods), Voronov, E.M., Mikrin, E.A., and Obnosov, V.M., Eds., Moscow: Bauman Moscow State Technical University, 2018. (In Russian.)
3. Onuora, A.E., Mbaocha, C.C., Eze, P.C., and Uchegbu, V.C., Unnamed Aerial Vehicle Pitch Optimization for Fast Response of Elevator Control System, *International Journal of Scientific Engineering and Science*, 2018, vol. 2, pp. 16–18.
4. Rodalski, T., Nowak, D., Walek, L., and Rzonca, D., Control System for Aircraft Take-off and Landing Based on Modified PID Controllers, *MATEC Web of Conferences*, 2019, vol. 252, art. no. 06008. DOI: https://doi.org/10.1051/matecconf/201925206008.
5. Sakharchuk, D.A., Sivashko, A.B., Kruglikov, S.V., et al., Directions of Creation and Development of Pilotless Aviation Complexes, *Science Intensive Technologies*, 2014, vol. 15, no. 5, pp. 22–26. (In Russian.)
6. Zhivov, Yu.G. and Poedinok, A.M., Adaptive Control System for Aircraft Longitudinal Motion, *TsAGI Science Journal,* 2012, vol. 43, no. 5, pp. 683–695.
7. Vassilyev, S.N., *Problemy upravleniya slozhnymi dinamicheskimi ob'ektami aviatsionnoi i kosmicheskoi tekhniki* (Control Problems for Complex Dynamic Aerospace Objects), Moscow: Mashinostroenie, 2015. (In Russian.)
8. Grebenkin, V.A., Developing Autoflare Algorithms for Landing with Interceptor Control, *Trudy MIEA. Navigats. Upravlen. Letat. Appar.*, 2013, no. 6, pp. 2–17. (In Russian.)
9. Kulikov, L.I., Synthesis of a Flight Control Algorithm for an Unmanned Parachute-Type Aerial Vehicle in the Barraging Mode at a Given Altitude, *Trudy 9-oi Nauchno-prakticheskoi konferentsii "Perspektivnye sistemy i zadachi upravleniya" i 4-oi molodezhnoi shkoly-seminara "Upravlenie i obrabotka informatsii v tekhnicheskikh sistemakh"* (Proceedings of the 9th Scientific and Practical Conference "Promising Systems and Problems of Control") and the 4th Youth School–Seminar "Control and Information Processing in Technical Systems"), Taganrog, 2014, pp. 226–227. (In Russian.)
10. Popova, I.V., Zemskov, A.V., Lestev, A.M., and Pestova, K.S., Control Algorithms for Gliding Unmanned Aerial Vehicles of

the Air-Surface Class, *Trudy 12-go Vserossiiskogo soveshchaniya po problemam upravleniya* (Proceedings of the 12th All-Russian Meeting on Control Problems), Moscow, Trapeznikov Institute of Control Sciences, 2014, pp. 3681–3689. (In Russian.)

11. Bystrov, D.A., Development and Research of Adaptation and Stabilization Algorithms in the System of an Aircraft, *Modern Science: Actual Problems of Theory and Practice. Natural and Technical Sciences*, 2014, no. 5/6, pp. 3–10. (In Russian.)
12. Abdullina, E.Yu. and Efanov, V.N., Synthesis of Pitch Angle Control System with Angle of Attack Limiting Channel, *Russian Aeronautics*, 2020, vol. 63, pp. 25–32.
13. Lovchakov, V.I., Synthesis of Linear Control Systems with Maximum Speed and Given Overshoot, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2020, vol. 21, no. 9, pp. 499–510. (In Russian.)
14. Piegat, A., *Fuzzy Modeling and Control*, Heidelberg: Physica-Verlag, 2001.
15. Vasil'ev, V.I. and Il'yasov, B.G., *Intellektual'nye sistemy upravleniya. Teoriya i praktika* (Intelligent Control Systems: Theory and Practice), Moscow: Radiotekhnika, 2009. (In Russian.)
16. Pervushina, N.A. and Khakimova, A.N., Development of Mathematical Models of Fuzzy Controllers Set by Genetic Algorithm to Stabilize Dynamic Object, *Control Sciences*, 2020, no. 4, pp. 3–14. (In Russian.)
17. Burakov, M.V. and Yakovets, O.B., Fuzzy Logic Control over a Power Gyroscopic System, *Journal of Instrument Engineering*, 2015, vol. 58, no. 10, pp. 804–808 (In Russian.)
18. Lysenko, L.N., Kyong, N.D., and Chyong, F.V., Motion Modeling for a Remotely Piloted Aerial Vehicle with a Modified Fuzzy Controller in the Flight Control Loop, *Polyot*, 2013, no. 2, pp. 24–30. (In Russian.)
19. Ulianov, G.N., Ivanov, S.A., and Vladyko, A.G., The Model of Controlling the Unmanned Aerial Vehicle with a Fuzzy Logic Controller, *Information and Control Systems*, 2012, no. 4, pp. 70–73. (In Russian.)
20. Matveev, E.N. and Glinchikov, V.A., Fuzzy Logic Conclusion in the Control System UAV, *Journal of Siberian Federal University. Engineering & Technologies*, 2011, no. 4, pp. 79–91. (In Russian.)
21. Sarabakha, A. and Kayacan, E., Type-2 Fuzzy Logic Controllers Made Even Simpler: From Design to Deployment for UAVs, *IEEE Transactions on Industrial Electronics*, 2018, vol. 65, no. 6, pp. 5069–5077.
22. Mikhalev, I.A., Okoemov, B.N., and Chikulaev, M.S., *Sistemy avtomaticheskogo upravleniya samoletom* (Aircraft Automatic Control Systems), Moscow: Mashinostroenie, 1987. (In Russian.)
23. Besekerskii, V.A. and Popov, E.P., *Teoriya sistem avtomaticheskogo regulirovaniya* (Theory of Automatic Control Systems), 4th ed., St. Petersburg: Professiya, 2007. (In Russian.)
24. Ostoslavskii, I.V. and Strazheva, I.V., *Dinamika poleta. Traektorii letatel'nykh apparatov* (Flight Dynamics. Aircraft Trajectories), 2nd ed., Moscow: Mashinostroenie, 1969. (In Russian.)

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

Received December 21, 2021, and revised August 17, 2022.
Accepted September 27, 2022.

Author information

Pervushina, Natal'ya Aleksandrovna. Cand. Sci. (Phys.–Math.), Zababakhin All-Russian Research Institute of Technical Physics (VNIITF), Russian Federal Nuclear Center, Snezhinsk, Russia
✉ p-n-a100678@yandex.ru

Frolova, Antonina Dmitrievna. Research Engineer, Zababakhin All-Russian Research Institute of Technical Physics (VNIITF), Russian Federal Nuclear Center, Snezhinsk, Russia
✉ antonina.garishina@yandex.ru

Cite this paper

Pervushina, N.A. and Frolova, A.D., Designing an Adaptive Stabilizing System for an Unmanned Aerial Vehicle. *Control Sciences* **5,** 2–12 (2022). http://doi.org/10.25728/cs.2022.5.1

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com

DOI: http://doi.org/10.25728/cs.2022.5.2

AN ANISOTROPY-BASED BOUNDEDNESS CRITERION FOR TIME-INVARIANT SYSTEMS WITH MULTIPLICATIVE NOISES

A. V. Yurchenkov

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia

✉ alexander.yurchenkov@yandex.ru

Abstract. This paper presents an anisotropy-based analysis of linear time-invariant systems with multiplicative noises. The system dynamics are described in the state space. The external disturbance belongs to the set of stationary sequences of random vectors with bounded mean anisotropy. The multiplicative noises are centered and have unit variance; the external disturbance and noises are mutually independent. We derive a boundedness criterion for the anisotropic norm in terms of Riccati-like inequalities using the bounded real lemma of the anisotropy-based theory. With a special change of variables, we reduce the analysis problem to a convex optimization problem with additional constraints. The existence of the latter's solution implies the bounded anisotropic norm of the system with multiplicative noises, and the minimal upper bound of the anisotropic norm can be obtained by solving this convex optimization problem.

Keywords: anisotropy-based theory, anisotropic norm, multiplicative noises, time-invariant systems, bounded real lemma.

INTRODUCTION

The attenuation of external disturbances is still one of the most topical problems in control theory [1, 2]. First appeared in the 1950s, when the growing complexity of technical systems required high accuracy as one priority, this research area has gradually formed an entire branch in modern control theory with many applications for different systems; for example, see [3–5]. In the problems of motion along a given trajectory, the control object is often subjected to disturbances whose stochastic characteristics significantly affect the choice of the control law. Some ways to reject external disturbances of bounded energy were considered in [6, 7]. Ensuring optimal control, the approach presented therein still suffers from a drawback: the resulting controllers have a high dimension. The technical implementation peculiarities of optimal control laws for continuous-time systems with bounded disturbances were analyzed in [8].

Note that such control problems were solved not only in the case of bounded disturbances. For example, in $\mathcal{H}_2$ control theory, random disturbances with known stochastic characteristics were considered; for $\mathcal{H}_\infty$ control laws, square integrable and square summable disturbances were selected for continuous-time and discrete-time systems, respectively, [9]. The choice of an appropriate optimality criterion largely depends on the type of disturbances: $\mathcal{H}_\infty$-optimal controllers have an increased conservatism due to the assumption on the worst-case input of the system and give far from optimal results under weakly colored disturbances; in contrast, $\mathcal{H}_2$-optimal control laws are oriented to no uncertainty in the stochastic parameters of Gaussian disturbances.

Despite the mixed $\mathcal{H}_2 / \mathcal{H}_\infty$ control statement proposed to eliminate the drawbacks of each disturbance control method mentioned, where different types of impacts on the system are separated by channels [10, 11], a stochastic approach to $\mathcal{H}_\infty$-optimization was also developed in [12–14]. This approach was introduced by I.G. Vladimirov and was called *the anisotropy-based (control) theory* of stochastic filtering and control. The *anisotropy* of a random vector is a measure of uncertainty for the distribution function of this vector. Due to this concept, the conservatism inherent in $\mathcal{H}_\infty$ control theory was reduced. *Mean anisotropy* was defined for stationary Gaussian sequences of random vectors. A performance criterion—*the anisotropic norm*—was chosen as a stochastic $\mathcal{H}_\infty$ norm of the system. Within the anisotropy-based theory, filtering and control problems (analysis and design) were solved for linear time-invariant and time-varying deterministic models. The analysis problem with random matrices in the object's state-space description was first posed in [15]; subsequently, systems with multiplicative noises were considered. Such descriptions of dynamics are typical of mechanical systems, financial models, chemical reactions [16, 17], and network systems [18, 19], arising interest in studying systems with multiplicative noises.

Within the anisotropy-based theory, the first works on control design for a system with multiplicative noises were estimative in nature: the anisotropic norm was majorized (an upper bound was constructed), and a control method for the upper bound was proposed [20]. The paper [21] considered the analysis problem, but an exact method for calculating the anisotropic norm was developed in [22] based on the approach presented in [15]. With the analysis problem solved, it became possible to construct an estimate in the case of measurement dropout correction [23] and an estimate based on a sensor network [24]. In the case of using a sensor network, one possible way to improve the efficiency of estimation is to adjust the information exchange scheme of the sensors; for details, see [25]. The above results refer to time-varying systems; for the class of time-invariant systems, the analysis problem was solved in [26]. Based on those results, below we reduce the anisotropy-based analysis problem to systems of matrix inequalities with convex constraints.

The remainder of this paper is organized as follows. Section 1 gives a brief introduction to the anisotropy-based theory. The problem under consideration is stated in Section 2. We present the main result of the paper in Section 3. Section 4 is devoted to numerical simulation.

1. PRELIMINARIES

This section provides only the basic definitions of the anisotropy-based theory for discrete time-varying systems. A more complete description can be found in [27, 28].

1.1. Mean Anisotropy and Anisotropic Norm

The mean anisotropy of a sequence of random vectors was defined in [13]. The anisotropy of a random m-dimensional vector $w \in \mathbb{R}^m$ with a probability density function (PDF) $f(x)$ is given by

$$\mathcal{A}(w) = \min_{\lambda>0} \mathcal{D}(f \,\|\, p_\lambda),$$

where the reference probability distribution $p_\lambda(x)$ is centered Gaussian with the scalar covariance matrix λI_m, i.e.,

$$p_\lambda(x) = (2\pi\lambda)^{-m/2} \exp\left(-\frac{\|x\|^2}{2\lambda}\right),$$

and $\mathcal{D}(f \,\|\, p_\lambda)$ denotes the Kullback–Leibler divergence (differential entropy) of the PDF f with respect to p_λ, i.e.,

$$\mathcal{D}(f \,\|\, p_\lambda) = E\left[\ln \frac{f}{p_\lambda}\right],$$

where $E[\cdot]$ stands for the expectation operator.

The mean anisotropy of a sequence of random vectors $W = \{w_k\}$ is the time-averaged anisotropy of an infinitely growing fragment of the sequence

$$\bar{\mathcal{A}}(W) = \lim_{N\to\infty} \frac{\mathcal{A}(W_{0:N-1})}{N},$$

where $W_{0:N-1} = \left(w_0^{\mathrm{T}}, \ldots, w_{N-1}^{\mathrm{T}}\right)^{\mathrm{T}}$ is the extended vector. The definition and properties of mean anisotropy were discussed in detail in [28].

Consider a linear system F with input $W \in \mathbb{L}_2^m$ and output $Z \in \mathbb{L}_2^p$ sequences. If the sequence W is obtained by a linear filter G from a white-noise Gaussian sequence V, then each random vector w_j of the former sequence can be written as

$$w_j = \sum_{k=0}^{\infty} g_k v_{j-k}, \quad j \in \mathbb{Z},$$

where $g_k \in \mathbb{R}^{m\times m}$, $k \ge 0$, denotes the impulse function. The generating filter G and its transfer function $G(z)$ have the relation

$$G(z) = \sum_{k=0}^{\infty} g_k z^k$$

for $|z|<1$, $z\in\mathbb{C}$. The finite $\mathcal{H}_2$-norm $\|G\|_2$ of the transfer function $G(z)$ can be calculated as

$$\|G\|_2=\left(\sum_{k=0}^{\infty}\mathrm{tr}\left(g_k g_k^{\mathrm{T}}\right)\right)^{1/2}.$$

We denote by $F(z)$ the transfer function of a linear system F with a finite $\mathcal{H}_\infty$-norm

$$\|F\|_\infty=\sup_{|z|<1}\bar{\sigma}\left(F(z)\right)=\operatorname{ess}\sup_{-\pi\le\omega\le\pi}\bar{\sigma}\left(\hat{F}(\omega)\right),$$

where $\bar{\sigma}(\cdot)$ is the maximum singular value of a corresponding matrix and $\hat{F}(\omega)=\lim_{\rho\to 1-}F\left(\rho e^{i\omega}\right)$.

In the anisotropy-based theory, the set of linear filters that generate sequences with a bounded mean anisotropy is denoted by

$$\mathcal{G}_a=\left\{G\in\mathcal{H}_2^{m\times m}:\ W=GV,\ \bar{\mathcal{A}}\left(W\right)\le a\right\},$$

where $\mathcal{H}_2^{m\times m}$ stands for the Hardy space of complex-valued matrix functions analytic inside the unit circle and $V=\left\{v_k\right\}_{k\in\mathbb{Z}}$ is a centered Gaussian sequence with the unit covariance matrix [12]. The anisotropic norm of a linear time-invariant system F with the input W generated by a filter G has the form

$$|||F|||_a=\sup\left\{\frac{\|FG\|_2}{\|G\|_2}:G\in\mathcal{G}_a\right\}.$$

For a causal system $F\in\mathcal{H}_\infty^{p\times m}$ satisfying the condition $\frac{\|F\|_2}{\sqrt{m}}<\|F\|_\infty$, the anisotropic norm always takes an intermediate value:

$$\frac{1}{\sqrt{m}}\|F\|_2=\lim_{a\to 0}|||F|||_a\le|||F|||_a\le\lim_{a\to\infty}|||F|||_a=\|F\|_\infty.$$

Due to this property, the anisotropy-based theory generalizes $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control theories: in the limiting cases (when mean anisotropy is zero or tends to infinity), we obtain either the scaled $\mathcal{H}_2$ norm or $\mathcal{H}_\infty$ norm of the linear time-invariant system. If mean anisotropy takes intermediate values, the anisotropic norm can be called a compromise between these norms.

For disturbing sequences with nonzero mean anisotropy, the anisotropic norm is a stochastic analog of the $\mathcal{H}_\infty$ norm; hence, an information criterion on the non-uniform distribution of the external disturbance can be used to reduce the conservatism of the classical $\mathcal{H}_\infty$ norm calculated for the “worst” case.

1.2. Calculation of the Anisotropic Norm

Consider the state-space description of the system:

$$F:\begin{cases}x_{k+1}=Ax_k+Bw_k,\\ z_k=Cx_k+Dw_k,\end{cases}\tag{1}$$

where $x_k\in\mathbb{R}^{n_x}$, $w_k\in\mathbb{R}^{m_w}$, and $z_k\in\mathbb{R}^{p_z}$ are the state vector, the external disturbance, and the system output, respectively. The system matrices A, B, C, and D are constant and have compatible dimensions. The system is stable if the spectral radius of the matrix A satisfies the inequality $\rho(A)<1$. The external disturbance is a colored sequence obtained by a generating filter from a white-noise sequence V. The state w_k of the filter G is a linear combination of the state vector of system (1) and the corresponding element of the Gaussian sequence V:

$$w_k=Lx_k+\Sigma^{1/2}v_k,$$

where $\Sigma\in\mathbb{R}^{m_w\times m_w}$ is a symmetric positive definite matrix and $L\in\mathbb{R}^{m_w\times n_x}$ is a matrix ensuring the asymptotic stability of ($A+BL$). There exists a parameter $q\in\left[0,\|F\|_\infty^{-2}\right)$ that relates via a special equation the mean anisotropy a and the anisotropic norm of the linear time-invariant system to the solutions of the Riccati and Lyapunov equations expressed through the state-space matrices [14]:

$$|||F|||_a=\left(\frac{1}{q}\left(1-\frac{m_w}{\mathrm{tr}\left(LPL^{\mathrm{T}}+\Sigma\right)}\right)\right)^2.$$

Moreover, the generating filter G ensures the mean anisotropy

$$a=-\frac{1}{2}\ln\det\left(\frac{m_w\Sigma}{\mathrm{tr}\left(LPL^{\mathrm{T}}+\Sigma\right)}\right),\tag{2}$$

where Σ is the covariance matrix, $P\in\mathbb{R}^{n_x\times n_x}$ is the solution of the Lyapunov equation

$$P=\left(A+BL\right)P\left(A+BL\right)^{\mathrm{T}}+B\Sigma B^{\mathrm{T}},\tag{3}$$

and the parameters $q, L,$ and Σ parameters are related to the solution of the Riccati equation:

$$R = A^{\mathrm{T}}RA + qC^{\mathrm{T}}C + L^{\mathrm{T}}\Sigma^{-1}L,$$
$$\Sigma = \left(I_{m_w} - qD^{\mathrm{T}}D - B^{\mathrm{T}}RB\right)^{-1}, \quad (4)$$
$$L = \Sigma\left(B^{\mathrm{T}}RA + qD^{\mathrm{T}}C\right).$$

Analysis issues in the anisotropy-based theory were described in detail in [13, 14]. The concepts mentioned above refer to linear time-invariant systems only, just one class of models considered in this theory.

1.3. The Suboptimal Problem

The system of coupled matrix equations (2)–(4) is nonlinear, which complicates numerical solution. In the anisotropy-based theory, optimal problems are therefore often replaced by suboptimal ones, for which an efficient numerical solution method has been developed. This method involves convex optimization to find an upper bound γ on the anisotropic norm $|||F|||_a$ of system (1). See the papers [29, 30] for numerical methods for solving suboptimal problems in the anisotropy-based theory.

The anisotropic norm of the linear system (1) is bounded above by a given threshold γ if the inequalities

$$\eta - \left(\exp(-2a)\det\Xi\right)^{1/m_w} < \gamma^2, \quad (5)$$

$$\begin{bmatrix} \Xi - \eta I_{m_w} & * & * \\ B & -\Theta & * \\ D & 0 & -I_{p_z} \end{bmatrix} \prec 0, \quad (6)$$

$$\begin{bmatrix} -\Theta & * & * & * \\ 0 & -\eta I_{m_w} & * & * \\ A & B & -\Theta & * \\ C & D & 0 & -I_{p_z} \end{bmatrix} \prec 0 \quad (7)$$

have positive definite solutions $\Xi \in \mathbb{R}^{m_w \times m_w}$ and $\Theta \in \mathbb{R}^{p_z \times p_z}$ with a parameter $\eta > 0$. (The symbol $*$ indicates symmetric blocks with respect to the main diagonal.) The sufficient conditions (5)–(7) of anisotropic norm boundedness can be obtained from equations (2)–(4) by passing to inequalities using the Schur complement lemma, the appropriate changes of variables, and the properties of solutions of the Riccati equations and inequalities [29]. Note that inequalities like (6) and (7) are understood in the sense of positive or negative definiteness.

The linear matrix inequalities (LMIs) (6) and (7) are obtained by congruent transformations: after applying the Schur complement lemma, these inequalities should be multiplied by the matrices $\mathrm{blockdiag}\left(I_{m_w}, \Theta, I_{p_z}\right)$ and $\mathrm{blockdiag}\left(I_{p_z}, I_{m_w}, \Theta, I_{p_z}\right)$ on the left and right, respectively.

Remark 1. This method of passing to matrix inequalities is not the only way to obtain a suboptimal solution based on the original optimal problem. Using the change of variables $\Theta^{-1} = \Pi$, we can introduce the inequality

$$\begin{bmatrix} \Theta & I_{p_z} \\ I_{p_z} & \Pi \end{bmatrix} \succ 0 \quad (8)$$

to eliminate the nonlinearity in inequalities (6) and (7) and use the algorithm for calculating the mutually inverse (reciprocal) matrices Θ and Π [31, 32].

The corresponding optimization problem has the form

$$\gamma^2 \underset{\Theta,\Xi,\eta,\gamma^2}{\to} \min$$

subject to the constraints (5)–(8). The minimum value γ^2 can be found using standard optimization procedures in applied software packages.

2. PROBLEM STATEMENT

Consider a linear discrete time-invariant system F with the state-space realization (1), where $w_k \in \mathbb{R}^{m_w}$ is a disturbance with a given upper bound a on its mean anisotropy. Let the free dynamics matrix A be represented as a linear combination of known matrices with random coefficients:

$$A = A_0 + \sum_{i=1}^{n} \xi_{i,k} A_i, \quad (9)$$

where the random variables $\xi_{i,k}$, $i = \{1, \ldots, n\}$, have zero mean and unit covariance. The existence of the first two moments of these variables is sufficient to apply the anisotropy-based theory methods. The matrices A_i, $i = \{0, \ldots, n\}, B, C,$ and D are known and have compatible dimensions. An additional condition, an analog of the Hurwitz property in the classical case of discrete time-invariant systems, has the form

$$\lim_{k\to\infty}\rho\left(\left(E\left[A^k\right]\right)^{\frac{1}{k}}\right)<1, \quad (10)$$

where $\rho(\cdot)$ is the spectral radius.

The problem is to find a condition on matrices of system (1), (9) under which its anisotropic norm will not exceed a given threshold γ :

$$|||F|||_a \le \gamma.$$

3. THE MAIN RESULT

In the general case, all matrices of system (1) may contain multiplicative noises, but we will focus on the problem statement above: for such models, one application is sensor systems with random dropouts in which the closed loop system contains multiplicative noises only in the matrix A.

The following lemma will serve for deriving a boundedness condition for the anisotropic norm of system (1) with the free dynamics matrix (9).

Lemma [26]. *The anisotropic norm* $|||F|||_a$ *of system* (1) *with the additional conditions* (9) *and* (10) *is bounded above by a positive number* γ *if there exist positive definite matrices* $R_1, R_2 \in \mathbb{R}^{n_x\times n_x}$ *and a parameter* $q\in\left[0, \|F\|_\infty^{-2}\right)$ *satisfying the system of modified Riccati-like equations*

$$\begin{gathered}R_1=\sum_{i=0}^{n}A_i^{\mathrm T}R_1A_i+qC^{\mathrm T}C,\\ R_2=A_0^{\mathrm T}R_2A_0+L^{\mathrm T}S^{-1}L,\\ S=\left(I_{m_w}-qD^{\mathrm T}D-B^{\mathrm T}R_1B-B^{\mathrm T}R_2B\right)^{-1},\\ L=S\left(qD^{\mathrm T}C+B^{\mathrm T}R_1A_0+B^{\mathrm T}R_2A_0\right)\end{gathered} \quad (11)$$

and the special inequality

$$-\frac{1}{2}\ln\det\left(\left(1-q\gamma^2\right)S\right)\ge a, \quad (12)$$

where a > 0 *is the mean anisotropy bound for the input sequence of random vectors* $\{w_k\}$.

This lemma is a modified analog of the bounded real lemma for time-invariant systems within the anisotropy-based theory [33]. Formulas (11) and (12) contain nonlinearities, which may complicate finding the solution. Therefore, it is necessary to reduce the equations to LMIs with an additional convex constraint. Before formulating this result as a theorem, we prove another assertion.

Theorem 1. *Let the mean anisotropy of the disturbance* $\{w_k\}$ *of system* (1) *with the additional condition* (10) *be bounded above by a number* $a\ge 0$. *If the inequality*

$$\begin{gathered}\tilde R\succ\sum_{i=0}^{n}A_i^{\mathrm T}RA_i+qC^{\mathrm T}C+L^{\mathrm T}S^{-1}L,\\ S=\left(I_{m_w}-qD^{\mathrm T}D-B^{\mathrm T}\tilde RB\right)^{-1},\\ L=S\left(B^{\mathrm T}\tilde RA_0+qD^{\mathrm T}C\right),\end{gathered} \quad (13)$$

jointly with the special inequality

$$-\frac{1}{2}\ln\det\left(\left(1-q\gamma^2\right)S\right)\ge a$$

has a solution $\tilde R\succ 0$, $S\succ 0$, $q\in\left[0, \|F\|_\infty^{-2}\right)$, *then the anisotropic norm of system* (1) *with* (9) *is bounded above by* $\gamma>0$.

P r o o f of Theorem 1. We introduce a new matrix variable of the form

$$R=R_1+R_2.$$

It satisfies an equation similar to the Riccati equation

$$\begin{gathered}R=\sum_{i=0}^{n}A_i^{\mathrm T}RA_i+qC^{\mathrm T}C+L^{\mathrm T}S^{-1}L-\sum_{i=1}^{n}A_i^{\mathrm T}R_2A_i,\\ S=\left(I_{m_w}-qD^{\mathrm T}D-B^{\mathrm T}RB\right)^{-1},\\ L=S\left(qD^{\mathrm T}C+B^{\mathrm T}RA_0\right),\end{gathered}$$

obtained using formula (11) and the variable R. According to the properties of the solutions of Riccati equations and inequalities [34], there exists a matrix $\tilde R=\tilde R^{\mathrm T}\succ 0$ satisfying inequality (13). ♦

Theorem 1 provides sufficient boundedness conditions for the anisotropic norm of the system with multiplicative noises. However, their verification is difficult due to the nonlinearity contained in formulas (12) and (13). The next result expresses a boundedness condition for the anisotropic norm in terms of LMIs with a convex constraint.

Theorem 2. *Consider the system with multiplicative noises* (1) *and the additional conditions* (9) *and* (10) *and let the mean anisotropy of the external disturbance be bounded above by a given number* $a\ge 0$. *The anisotropic norm of the system will not exceed a given threshold* γ,

$$|||F|||_a\le\gamma,$$

if the inequalities

$$\begin{bmatrix} \sum_{i=0}^{n} A_i^{\mathrm{T}} R A_i - R + C^{\mathrm{T}} C & * \\ B^{\mathrm{T}} R A_0 + D^{\mathrm{T}} C & \eta I_{m_w} - D^{\mathrm{T}} D - B^{\mathrm{T}} R B \end{bmatrix} \prec 0, \quad (14)$$

$$\begin{bmatrix} \eta I_{m_w} - \Psi - D^{\mathrm{T}} D & * \\ RB & R \end{bmatrix} \succ 0, \quad (15)$$

$$\ln\det\Psi \geq 2a + \ln\left(\eta - \gamma^2\right) \quad (16)$$

have solutions $R = R^{\mathrm{T}} \succ 0$, $\Psi = \Psi^{\mathrm{T}} \succ 0$, *and* $\eta > 0$.

Proof of Theorem 2. We choose $\eta = q^{-1}$ as a new variable. With the change $R = q^{-1}\tilde{R}$, inequality (14) can be obtained from inequality (13) by applying the Schur complement lemma. Next, we introduce a matrix Θ satisfying the relation $0 \prec \Theta \prec S^{-1}$. Then the matrix $\Psi = q^{-1}\Theta$ will satisfy inequality (15) after applying the Schur complement lemma. The convex constraint (16) is the special inequality (14) written in terms of the new variables. ♦

Obviously, system (14)–(16) is convex in the variable γ^2. Hence, we can formulate the convex optimization problem

$$\gamma^2 \underset{R,\Psi,\eta,\gamma^2}{\rightarrow} \min$$

under the existence of solutions of the LMIs (14) and (15) with the convex constraint (16). This convex optimization problem can be solved using standard semidefinite programming tools.

4. NUMERICAL SIMULATION

As an illustrative example, we consider a two-mass oscillating system described in [35]. The system was closed by a standard linear-quadratic controller and discretized. Its state-space implementation has the form

$$x_{k+1} = \left(A_0 + \xi_1 A_1 + \xi_2 A_2\right) x_k + B w_k,$$
$$z_k = C x_k + D w_k,$$

where the mean anisotropy of the external disturbance (a sequence of random vectors $\{w_k\}$) is bounded above by a given number a and the random variables ξ_1, ξ_2 are centered and have unit variance. The numerical matrices are known:

$$A_0 = \begin{bmatrix} 0.9918 & 0.0444 & 0.0031 & -0.0043 \\ -0.3177 & 0.7829 & 0.1190 & -0.1651 \\ 0.0012 & 0.0000 & 0.9988 & 0.0500 \\ 0.0498 & 0.0012 & -0.0499 & 0.9987 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} 0.0992 & 0.0044 & 0.0003 & -0.0004 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0001 & 0.0000 & 0.1492 & 0.0075 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix},$$

$$C = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 1.0000 & 0.0000 \\ -6.0545 & -4.7020 & 1.5823 & -2.7857 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0001 \\ -0.0043 \\ 0.0012 \\ 0.0507 \end{bmatrix}, \quad D = 0.$$

The table below combines the upper bounds (thresholds) γ of the anisotropic norm $|||F|||_a$ calculated for different mean anisotropies a of the external disturbance.

Note that the $\mathcal{H}_\infty$ norm of the system is 3.3244, i.e., the anisotropic estimator provides a much better quality of estimation in terms of the root-mean-square gain.

The anisotropic norm threshold depending on the mean anisotropy

The mean anisotropy a	0.0	0.01	0.05	0.10	0.20	0.50	1.00	1.50	2.00	3.00
The anisotropic norm threshold γ	0.3035	0.3048	0.3124	0.3211	0.3363	0.3655	0.3737	0.4299	0.9739	2.9180

5. CONCLUSIONS

This paper has presented an anisotropy-based analysis of linear discrete time-invariant systems with multiplicative noises. The bounded real lemma and a special change of variables have been adopted to establish a boundedness condition for the anisotropic norm of the system in terms of state-space matrices. Moreover, the upper bound on the anisotropic norm can be numerically minimized by standard semidefinite programming tools. As an illustrative example, the upper bound has been calculated for the anisotropic norm of an oscillating system. As demonstrated above, anisotropy-based estimation can significantly improve the quality of estimation under a priori information (the bounded mean anisotropy of the external disturbance), especially in the cases of weakly colored disturbances.

REFERENCES

1. Wang, L. and Guan, R.P., Disturbance Rejection and Reference Tracking via Observer Design, in *State Feedback Control and Kalman Filtering with MATLAB/Simulink Tutorials*, IEEE, 2022, pp. 195–251.
2. Kondratenko, Yu.P., Kuntsevich, V.M., Chikrii, A.A., and Gubarev, V.F., 5 Inverse Model Approach to Disturbance Rejection Problem, in *Advanced Control Systems: Theory and Applications*, River Publishers, 2021, pp. 129–166.
3. Gómez-León, B.C., Aguilar-Orduña, M.A., and Sira-Ramírez, H.J., A Disturbance Observer Based Control Scheme Using an Active Disturbance Rejection Controller: An Underactuated Moving Crane Example, *18th Int. Conf. on Electrical Engineering, Computing Science and Automatic Control (CCE)*, Mexico City, Mexico, 2021, pp. 1–6.
4. Mylvaganam, T. and Sassano, M., Disturbance Attenuation by Measurement Feedback in Nonlinear Systems via Immersion and Algebraic Conditions, *IEEE Transactions on Automatic Control*, 2019, vol. 65, no. 2, pp. 854–860.
5. Lu, P., Sandy, T., and Buchli, J., Nonlinear Disturbance Attenuation Control of Hydraulic Robotics, *2018 IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)*, Kuala Lumpur, Malaysia, 2018, pp. 1451–1458.
6. Yakubovich, Ye.D. Solving One Problem of Optimal Control for a Discrete Linear System, *Automation and Remote Control*, 1975, vol. 36, no. 9, pp. 1447–1453.
7. Vidyasagar, M., Optimal Rejection of Persistent Bounded Disturbances, *IEEE Trans. Automat. Control*, 1986, vol. 31, pp. 527–535.
8. Nazin, S.A., Polyak, B.T., and Topunov, M.V., Rejection of Bounded Exogenous Disturbances by the Method of Invariant Ellipsoids, *Automation and Remote Control*, 2007, vol. 68, no. 3, pp. 467–486.
9. Doyle, J.C., Glover, K., Khargonekar, P.P., and Francis, B.A., State-Space Solutions to Standard H_2 and H_∞ Control Problems, *IEEE Trans. Automat. Control*, 1989, vol. 34, no. 8, pp. 831–847.
10. Doyle, J.C., Zhou, K, and Bodenheimer, B., Optimal Control with Mixed H_2 and H_∞ Performance Objectives, *Proc. of the American Control Conference*, Pittsburg, 1989, pp. 2065–2070.
11. Steinbuch, M. and Bosgra, O.H., Necessary Conditions for Static and Fixed Order Dynamic Mixed H_2 / H_∞ Optimal Control, *Proc. of the American Control Conference*, Boston, 1991, pp. 1137–1142.
12. Semyonov, A.V., Vladimirov, I.G., and Kurdyukov, A.P., Stochastic Approach to H_∞ Optimization, *Proc. of the 33rd IEEE Conf. Decision and Control*, Lake Buena Vista, 1994, vol. 3, pp. 2249–2250.
13. Vladimirov, I.G., Kurdjukov, A.P., and Semyonov, A.V., Anisotropy of Signals and the Entropy of Linear Stationary Systems, *Doklady Math.,* 1995, vol. 51, pp. 388–390.
14. Vladimirov, I.G., Kurdjukov, A.P., Semyonov, A.V. State-Space Solution to Anisotropy-based Stochastic H_∞-optimization Problem, *Proc. of the 13th IFAC World Congress*, San Francisco, 1996, pp. 427–432.
15. Kustov, A.Yu., State-Space Formulas for Anisotropic Norm of Linear Discrete Time Varying Stochastic Systems, *Proc. of the 15th Int. Conf. on Electrical Eng. Science and Aut. Control*, Mexico City, Mexico, 2018, pp. 1–6.
16. Gershon, E., Shaked, U., and Yaesh, I., H_∞ Control and Filtering of Discrete-Time Stochastic Systems with Multiplicative Noise, *Automatica*, 2001, vol. 37, pp. 409–417.
17. Gershon, E., Shaked, U., and Yaesh, I., *H_∞ Control and Estimation of State-multiplicative Linear Systems*, Lecture Notes in Control and Information Sciences, vol. 318, Springer-Verlag, 2005.
18. Shen, B., Wang, Z., and Hung, Y.S., Distributed H_∞-consensus Filtering in Sensor Networks with Multiple Missing Measurements: The Finite-Horizon Case, *Automatica*, 2010, vol. 46, pp. 1682–1688.
19. Liu, S., Zhao, G., He, Y., and Gao, C., Decentralized Moving Horizon Estimation for Networked Navigation System with Packet Dropouts, *39th Chinese Control Conference (CCC)*, Shenyang, China, 2020, pp. 3381–3384.
20. Yurchenkov, A.V., Anisotropy-Based Controller Design for Linear Discrete-Time Systems with Multiplicative Noise, *Journal of Computer and Systems Sciences Int.,* 2018, vol. 57, no. 6, pp. 864–873.
21. Yurchenkov, A.V., Kustov, A.Yu, and Kurdyukov, A.P., Anisotropy-based Bounded Real Lemma for Discrete-time Systems with Multiplicative Noise, *Doklady Mathematics*, 2016, vol. 93, no. 2, pp. 238–240.
22. Belov, I.R., Yurchenkov, A.V., and Kustov, A.Yu., Anisotropy-Based Bounded Real Lemma for Multiplicative Noise Systems: the Finite Horizon Case, *Proc. of the 27th Med. Conf. on Control and Automation*, Akko, 2019, pp. 148–152.
23. Kustov, A.Yu. and Yurchenkov, A.V., Finite-horizon Anisotropy-based Estimation with Packet Dropouts, *IFAC-PapersOnLine*, 2020, vol. 53, no. 2, pp. 4516–4520.
24. Kustov, A.Yu. and Yurchenkov, A.V., Finite-horizon Anisotropic Estimator Design in Sensor Networks, *Proc. of the 59th IEEE Conf. on Decision and Control*, Jeju: IEEE, 2020, pp. 4330–433.
25. Yurchenkov, A.V. and Kustov, A.Yu., Sensor Network Adjusting Based on Anisotropic Criterion, *Proc. of the 2021*

9th Int. Conf. on Systems and Control, Caen, 2021, pp. 268–273.
26. Kustov, A.Yu., Timin, V.N., and Yurchenkov, A.V., Boundedness Condition for Anisotropic Norm of Linear Discrete Time-invariant Systems with Multiplicative Noise, *Journal of Physics: Conference Series*, 2021, vol. 1864, pp. 1–5.
27. Diamond, P., Vladimirov, I.G., Kurdyukov, A.P., and Semenov, A.V., Anisotropy-based Performance Analysis of Linear Discrete Time Invariant Control Systems, *Int. Journ. of Control*, 2001, vol. 74, no. 1, pp. 28–42.
28. Diamond, P., Kloeden, P., and Vladimirov, I.G., Mean Anisotropy of Homogeneous Gaussian Random Fields and Anisotropic Norms of Linear Translation-Invariant Operators on Multidimensional Integer Lattices, *Journ. of Applied Mathematics and Stochastic Analysis*, 2003, vol. 16, no. 3, pp. 209–231.
29. Tchaikovsky, M.M., Kurdyukov, A.P., and Timin, V.N., Strict Anisotropic Norm Bounded Real Lemma in Terms of Inequalities, *Proc. of the 18th IFAC World Congr.,* Milan, 2011, pp. 2332–2337.
30. Tchaikovsky, M.M., Static Output Feedback Anisotropic Controller Design by LMI-based Approach: General and Special Cases, *Proc. of the American Control Conference*, Montreal, 2012, pp. 5208–5213.
31. Iwasaki, T. and Skelton, R.E., The XY-centering Algorithm for the Dual LMI Problem: A New Approach to Fixed Order Design, *Int. Journ. of Control*, 1995, vol. 62, pp. 1257–1272.
32. Apkarian, P. and Tuan, H.D., Concave Programming in Control Theory, *Journ. of Global Optimization*, 1999, vol. 15, pp. 343–370.
33. Maximov, E.A., Kurdyukov, A.P., and Vladimirov, I.G., Anisotropic Norm Bounded Real Lemma for Linear Discrete Time Varying Systems, *Proc. of the 18th IFAC World Congr.,* Milan, 2011, pp. 4701–4706.
34. de Souza, C.E., On Stabilizing Properties of Solutions of the Riccati Difference Equation, *IEEE Trans. on Automatic Control*, 1989, vol. 34, pp. 1313–1316.
35. Chilali, M. and Gahinet, P., H_∞ Design with Pole Placement Constraints: An LMI Approach, *IEEE Trans. on Automatic Control*, 1996, vol. 41, no. 3, pp. 358–367.

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

Received July 14, 2022, and revised October 30, 2022.
Accepted November 9, 2022.

Author information

Yurchenkov, Alexander Viktorovich. Cand. Sci. (Phys.–Math.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ alexander.yurchenkov@yandex.ru

Cite this paper

Yurchenkov, A.V., An Anisotropy-Based Boundedness Criterion for Time-Invariant Systems with Multiplicative Noises. *Control Sciences* **5,** 13–20 (2022). http://doi.org/10.25728/cs.2022.5.2

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com

DOI: http://doi.org/10.25728/cs.2022.5.3

A MATHEMATICAL FORMULATION OF CONTROL PROBLEMS ON COGNITIVE MODELS[1]

O.I. Gorbaneva[1], A.D.Murzin[2], and G.A. Ougolnitsky[3]

Southern Federal University, Rostov-on-Don, Russia

[1]✉ oigorbaneva@sfedu.ru, [2]✉ admurzin@sfedu.ru, [3]✉ gaugolnickiy@sfedu.ru

Abstract. This paper considers cognitive modeling methods under different types of control. Relevant publications are briefly surveyed. The cognitive model is formally described as a simulation model based on a directed graph (signed or weighted digraph). Mathematical formulations of the optimal, conflict, and hierarchical control problems are proposed for cognitive models in the case of pulse processes and in the general case as well. The methodology is applied to the predator-prey model and the aggregative model of a national economy. The methodological assumptions are detailed for another control problem on a cognitive map (the optimal management of a university). In this model, a university determines the number of commercial places and the price of commercial education. The model is identified on real data for the three largest universities of the Rostov region (the Russian Federation). Some conclusions and recommendations are formulated based on model analysis.

Keywords: control problems, discrete dynamic models, cognitive modeling, control in social and economic systems.

INTRODUCTION

The cognitive modeling of complex systems is an extensive research area that has been actively developing for several decades. In a broad sense, cognitive modeling is understood as the use of various artificial intelligence (AI) models, e.g., neural networks [1]. A particular decision methodology for weakly structured systems based on cognitive maps was proposed by R. Axelrod [2]. A cognitive map is a (signed or weighted) digraph with the vertices corresponding to system components and the arcs to the relations between them. The paper [3] was among the first publications on the subject; the approach was described in detail in the monograph [4]. Fuzzy cognitive map models were surveyed in [5–7].

The methodology of cognitive modeling as the simulation modeling of complex systems based on cognitive maps was developed by N.A. Abramova, Z.K. Avdeeva, et al. (Trapeznikov Institute of Control Sciences RAS) and G.V. Gorelova et al. (Southern Federal University) [8–15].

The paper [8] provided a detailed analysis of the verification problem of cognitive models and an illustrative example of a particular model from this point of view. Some prospects for the development of this research area were outlined in [9, 13].

S.V. Kovriga, E.K. Kornoushenko, and V.I. Maximov (Trapeznikov Institute of Control Systems RAS) considered cognitive modeling as a structure-and-goal analysis tool with application to the development problems of Russian regions as well as stated and solved control problems; see [16–19].

The experience of cognitive modeling of regional socio-economic processes was presented in [20]. The influence of a regional higher education system on the innovative development of the region was studied in [21]. Modern approaches to cognitive modeling were described in [22–24].

Therefore, the works [8–19] pointed to the managerial aspect of cognitive modeling. As we believe, this aspect should be crucial due to the activity of complex

[1] This work was supported by Southern Federal University within the project "The Digital Atlas of Political and Socio-economic Threats and Risks to the Development of Russia's Southern Border Area: The National and Regional Context (The Digital South)," no. SP-14-22-06.

socio-economic and other weakly structured systems [25–30]. It is reasonable to distinguish dynamic models of optimal control with a single subject [31, 32], conflict control with several competing subjects [33], and hierarchical control with an ordered set of control subjects [34]. The author's approach to solving complex dynamic control problems using simulation modeling was described in [35]. The existence of a cognitive map in the form of a digraph relates cognitive modeling with network control models [36, 37]. Here, the paper [38] by D.A. Novikov is of great interest: he analyzed the feasibility of combining cognitive and game-theoretic approaches, classified cognitive games, and gave an example of a linear impulse cognitive game.

This paper has the following contributions:

- The formal description of the cognitive model as a simulation model based on a weighted digraph (cognitive map) is refined.
- Mathematical formulations of optimal, conflict, and hierarchical control problems are developed for cognitive models.
- The methodological assumptions are detailed for another control problem on a cognitive model: the optimal management of a university.

The remainder of this paper is organized as follows. Section 1 describes the basic cognitive model. In Section 2, we consider control problems of different types. Section 3 is devoted to a particular control problem on a cognitive map (the optimal management of a university). Some conclusions and recommendations are formulated in the Conclusions based on model analysis.

1. THE BASIC COGNITIVE MODEL

The basic cognitive model involves a digraph (cognitive map) in which each vertex and each arc are assigned some real value (as a function of time) and some constant weight (real number), respectively. In particular, the most common ones are signed digraphs, in which the arc weights take values ± 1. Digraph vertices represent the elements of a complex system under study, and digraph arcs represent the connections between them. Each element has some quantitative characteristic that can change over time, whereas each connection has a constant quantitative characteristic.

The basic cognitive model serves to describe and forecast the dynamics of vertex values, which are determined by their initial values and the structure of their connections with arc weights. Forecasting uses several rules reflecting different hypotheses about the dynamics of vertex values.

1.1. The Set of Vertices and Their Values

The vertices of the basic cognitive model represent elements of the system under study. Depending on the nature of the system, these can be employees or divisions of an organization, firms or corporations, biological populations, social groups, countries or their regions, etc. The list of vertices (the cognitive model variables) includes only the system elements with a principal role in the goals of study. This list reflects a compromise between the desire to consider as many system indicators as possible and the real possibilities of study. Formally, the list of vertices is a finite set $V=\{u_1,..., u_n\}$, where n denotes the number of vertices.

Each vertex $u_i \in V$ is assigned a real value x_i (a function of discrete time), i.e., $x_i:\{0,1,...,T\} \to \mathbb{R}$. Thus, $x_i(t)$ is the value of the vertex u_i at a time instant t. Of course, the scalarity hypothesis of the value x_i is strongly restrictive because, in reality, each element of the system has several indicators. However, it considerably simplifies the study, still yielding quite interpretable models. The vector $x(t)=(x_1(t),..., x_n(t))$ fully characterizes the system state at each time instant t.

Finally, it is important to determine the initial values of all vertices, $x_0=(x_{10},..., x_{n0})$ (the initial state of the system). This is done based on the available literature, consulting, expertise, etc.

1.2. The Set of Arcs and Their Weights

The arcs of the cognitive model reflect cause-and-effect relations between elements of the system under study. If an arc is positive, increasing the value of the input vertex leads to increasing the value of the output vertex, i.e., the connection is direct. If an arc is negative, increasing the value of the input vertex leads to decreasing the value of the output vertex, i.e., the connection is inverse. The weight of an arc shows the strength of the corresponding connection (the increase/decrease coefficient).

Note that the total number of possible connections between n vertices equals C_n^2 (the number of combinations). The value C_n^2 grows fast with the parameter n, so only the most significant connections should be considered when constructing a cognitive map.

Sometimes the sign of an arc (and even more so its weight) is difficult to determine unambiguously. For example, the plus sign shows a proportional dependence of the price of a bus ticket on the trip length.

However, it is also possible to use the minus sign, which encourages long trips by public transport instead of private cars. In this case, both scenarios should be considered to compare their effect on the system.

Generally speaking, the identification problem is crucial for cognitive models. Researchers often distinguish between structural identification (selection of the sets of vertices and arcs) and numerical identification (selection of the initial values of vertices and the initial weights of arcs). Unfortunately, the identification problem (especially the structural one) is extremely difficult to formalize, which causes inevitable errors in the expertise-based solution [8].

1.3. Value Change Rules for Vertices

It is reasonable to treat cognitive modeling as simulation modeling by cognitive maps. The basic simulation model has the following form:

$$x_j(t+1) = x_j(t) + f(x(t)), \quad x_j(0) = x_{j0}, \quad j = 1,\dots, n. \tag{1}$$

Formula (1) describes the value conservation law for the vertex u_j as the balance relation. A particular cognitive model is specified by the function f. The so-called impulse process (step-function) [4]

$$x_j(t+1) = x_j(t) + \sum_{i=1}^{n} a_{ij} p_i(t), x_j(0) = x_{j0}, j = 1,\dots,n, \tag{2}$$

is the most widespread rule; here, the difference

$$p_i(t+1) = x_i(t+1) - x_i(t) \tag{3}$$

denotes the impulse in the vertex u_i at the time instant t. Due to formula (3), the rule (2) can be written as

$$p_j(t+1) = \sum_{i=1}^{n} a_{ij} p_i(t), x_j(0) = x_{j0}, j = 1,\dots,n. \tag{4}$$

In the vector form,

$$p(t+1) = A^{\mathrm{T}} p(t), \ t = 0, 1, \dots, \tag{5}$$

where $p(t) = (p_1(t),\dots, p_n(t))^{\mathrm{T}}$. Then it is easy to show by induction that

$$p(t) = A^{\mathrm{T}} p(0), \ t = 0, 1, \dots \tag{6}$$

To emphasize the role of a cognitive map (V, A) defining the system structure, we represent the rule (2) as

$$x_j(t+1) = x_j(t) + \sum_{i \in I(j)} a_{ij} p_i(t), \quad x_j(0) = x_{j0}, \ j = 1,\dots, n, \tag{7}$$

where $I(j)$ denotes the set of all vertices with outgoing arcs to vertex j. The expressions (2) and (7) are equivalent since $a_{ij} = 0$ in the case of no incoming arc (u_i, u_j).

Thus, knowing the weight matrix A and the initial impulse vector $p(0)$, we can forecast the values of all impulses $p(t)$ for any time instant t; knowing the initial value vector x_0, we can calculate the values of all vertices for any time instant t using formula (3), i.e., completely solve the forecasting problem [4].

Thus, the basic cognitive model is described by

$$\langle V, A, x_0, f \rangle \tag{8}$$

with the following notations: $V = \{u_1,\dots, u_n\}$ is a finite set of vertices; $A = \| a_{ij} \|, i = 1,\dots, n, j = 1,\dots, n,$ is an adjacency matrix (if $a_{ij} \neq 0,$ we have arc (u_i, u_j) with the weight a_{ij}); $x_0 = (x_{10},\dots, x_{n0})$ is the vector of initial vertex values. Here, the function f specifies the value change rule, which has the general form (1). When considering an impulse process, we assume that the initial impulse vector $p_0 = (p_{10},\dots,p_{n0})$ is given.

2. CONTROL PROBLEMS

The basic cognitive model assumes that the values of vertices change only due to the natural dynamics (1), e.g., those of the impulse process (2). In this case, the change of values of all vertices (the dynamics of the system state) on the entire forecasting period is completely determined by the weight matrix and the initial distribution of impulses and values. If the initial values of impulses can be set arbitrarily, they perform the control function. However, real systems often undergo some external impact. If this impact is purposeful, the control itself and its optimization problem arise in the model. With a certain degree of conditionality, depending on the set of control subjects and its structure, we will distinguish among three types of control: optimal, conflict, and hierarchical.

2.1. Optimal Control

In this problem statement, there is one control subject influencing the dynamic system (1) with an optimality criterion

$$J = \sum_{t=1}^{T} \delta^t g(x(t), u(t)) + \delta^T G(x(T)) \to \max \tag{9}$$

under control constraints

$$u(t) \in U(t), \ t = 1,\dots, T. \tag{10}$$

Here $\delta \in (0, 1]$ is the discount factor, and $g(\cdot)$ and $G(\cdot)$ are the instantaneous and terminal goal functions. (If $T = \infty$, the term $G(x(T))$ disappears.) The control variable u can be an open-loop $u(t)$ or closed-loop $u(t, x(t))$ strategy. Also, we can introduce additional constraints of the form

$$x(t) \in X^*, t = 1,...,T, \tag{11}$$

or

$$x(T) \in X^*, \tag{12}$$

known as the viability (homeostasis) conditions in the theory of sustainable management of active systems [30]. The strong form (11) (the weak form (12)) means that the state variable of the controlled dynamic system is within a given domain X^* at any time instant (at the terminal time instant, respectively). These conditions can be treated as the goal of control.

With the control impact, the dynamics equation (1) becomes

$$x_j(t+1) = x_j(t) + f(x(t), u(t)),$$
$$x_j(0) = x_{j0},\ j = 1,..., n, \tag{13}$$

yielding the optimal control model (9), (10), (13) with the state-space constraints (11) or (12).

The control variable (i.e., the function $f(x, u)$ in (13)) can be incorporated into the cognitive model in different ways. Let the vertex set V of the cognitive map be augmented by a control variable v and the arc set A be augmented by arcs (v, u_i) with some weights b_j. If $b_j \neq 0,$ then the vertex u_j is controlled.

Then the controlled impulse process takes the form

$$x_j(t+1) = x_j(t) + b_j h_j(u_j(t)) +$$
$$\sum_{i=1}^{n} a_{ij} p_i(t),\ x_j(0) = x_{j0},\ j = 1,..., n, \tag{14}$$

where the control function $h_j(u_j)$ is, e.g., $h_j(u_j) = u_j^{p_j}$ with $p_j > 0$.

2.2. Conflict Control

In this problem statement, there are several control subjects influencing, simultaneously and independently, the dynamic system (1) with optimality criteria

$$J_k = \sum_{t=1}^{T} \delta^t g_k(x(t), u(t)) +$$
$$\delta^T G_k(x(T)) \to \max,\ k = 1,..., m, \tag{15}$$

under control constraints (10), $u(t) = (u_1(t),..., u_m(t)),$ where m is the number of control subjects. By assumption, Nash equilibrium [33] is the solution of the differential game (10), (13), and (15) with the state-space constraints (11) or (12).

In this case, the vertex set V of the cognitive map is augmented by control vertices $v_1,..., v_m,$ whereas the arc set A is augmented by arcs (v_k, u_i) with weights b_{kj}, $k = 1,..., m,\ j = 1,..., n$. If $b_{kj} \neq 0,$ then the vertex u_j is controlled by the vertex v_k.

The conflict-controlled impulse process takes the form

$$x_j(t+1) = x_j(t) + \sum_{k=1}^{m} b_{kj} h_{kj}(u_{kj}(t)) + \sum_{i=1}^{n} a_{ij} p_i(t),$$
$$x_j(0) = x_{j0},\ j = 1,..., n. \tag{16}$$

According to [38], cognitive games are classified by several features. In the proposed approach, we fix the following features: nonlinear games, common knowledge, no uncertainty, discrete time, the dependence payoff functions on the actions of all players and the trajectory (closed-loop strategies), a finite horizon, individual constraints, the choice of decisions at each time instant, simultaneous (in the next subsection, sequential) decision making, and no coalitions.

2.3. Hierarchical Control

In this case, the set of control subjects has a hierarchical structure and includes several influence agents and one coordinating center (the Principal). The Principal makes the first move by selecting a control impact

$$u_0(t) \in U_0(t),\ t = 1,..., T, \tag{17}$$

and reporting it to all influence agents. The Principal's optimality criterion has the form

$$J_0 = \sum_{t=1}^{T} \delta^t g_0(x(t), u(t)) + \delta^T G_0(x(T)) \to \max. \tag{18}$$

Knowing the value u_0, the influence agents choose, simultaneously and independently, their control impacts

$$u_k(t) \in U_k(t),\ t = 1,..., T. \tag{19}$$

The influence agents are guided by their optimality criteria (15), $u(t) = (u_0(t), u_1(t),..., u_m(t)),$ where m denotes the number of influence agents. Let the optimal response of influence agents to the Principal's control be one of the Nash equilibria in the agents' game. By assumption, Stackelberg equilibrium [33] is the solution of the hierarchical game (13), (15), (17)–(19) with the state-space constraints (11) or (12).

In this case, the vertex set V of the cognitive map is augmented by control vertices $v_0, v_1,..., v_m,$ whereas the arc set A is augmented by arcs (v_k, u_i) with some weights $b_{kj}, k = 0, 1,..., m, j = 1,..., n$. If $b_{kj} \neq 0,$ then the vertex u_j is controlled by the vertex v_k.

The hierarchically controlled impulse process takes the form

$$x_j(t+1) = x_j(t) + \sum_{k=0}^{m} b_{kj} h_{kj}(u_{kj}(t)) + \sum_{i=1}^{n} a_{ij} p_i(t),$$
$$x_j(0) = x_{j0},\ j = 1,..., n. \tag{20}$$

Thus, in almost all application-relevant cases, the basic cognitive model (8) is supplemented with the problems of optimal, conflict, or hierarchical control. Then the goal of study is to forecast the dynamics of the controlled system under different influence scenarios and optimize control in some sense.

Briefly, a cognitive model can be defined as a simulation model of a complex system whose structure is specified by a (signed or weighted) digraph and determines the dynamics of the controlled system state under various purposeful control impacts and external factors.

For impulse processes with any control type, simulation scenarios include two components, namely, the initial impulse distribution $p_0 = (p_{10}, ..., p_{n0})$ and the control trajectory, which has the following form:

- $\{u_j(t),\ j = 1, ..., n,\ t = 0, 1, ..., T-1\}$ for optimal control,
- $\{u_{kj}(t),\ k = 1, ..., m,\ j = 1, ..., n,\ t = 0, 1, ..., T-1\}$ for conflict control,
- $\{u_{kj}(t),\ k = 0, 1, ..., m,\ j = 1, ..., n,\ t = 0, 1, ..., T-1\}$ for hierarchical control.

For simulations, we apply the method of qualitatively representative scenarios [35].

Example 1. The predator–prey model.

The cognitive map of this model is shown in Fig. 1.

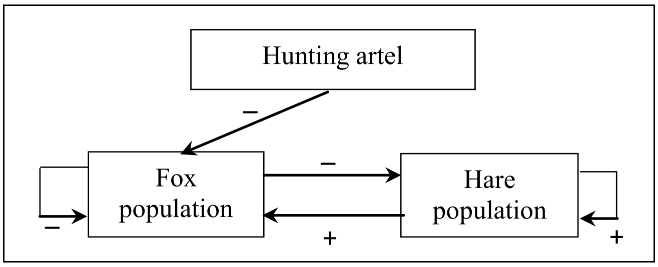


Fig. 1. The cognitive map of the predator–prey model.

The model based on this cognitive map is

$$x_F(t+1) = [1 - s(t)]x_F(t) - a_F x_F(t) + b_F x_F(t) x_H(t),\ x_F(0) = x_{F0}; \quad (21)$$

$$x_H(t+1) = x_F(t) + a_H x_H(t) - b_H x_F(t) x_H(t),\ x_H(0) = x_{H0}. \quad (22)$$

Here $x_F(t)$ and $x_H(t)$ denote the numbers of foxes and hare (the predator and prey, respectively) in year *t*; $a_F > 0$ and $a_H > 0$ are the reproduction rates of the fox and hare populations, respectively; $b_F > 0$ and $b_H > 0$ are the trophic interaction rates of the fox and hare populations, respectively; x_{F0} and x_{H0} are the initial numbers of foxes and hare, respectively; finally, $s(t)$ is the shooting rate of foxes in year *t*.

For the fox population, the optimal exploitation problem has the form

$$J = \sum_{t=1}^{T} [cs(t)x_F(t) - ds^2(t)] \to \max,\ 0 \le s(t) \le 1, \quad (23)$$

$$x_F(T) \ge x_F^*,\ x_H(T) \ge x_H^*, \quad (24)$$

where $c > 0$ is the specific hunting utility, $d > 0$ is the hunting cost coefficient, and x_F^* and x_H^* are the critical numbers of foxes and hare, respectively.

The optimal exploitation problem (23) can be easily generalized to the case of competing hunting artels. In fact, the sustainable development condition (24) is external to hunting, and an environmental body should be introduced to influence hunters. This approach leads to a hierarchical control problem. ♦

Example 2. The Ramsey–Solow aggregate model of a national economy.

The cognitive map of this model is shown in Fig. 2.

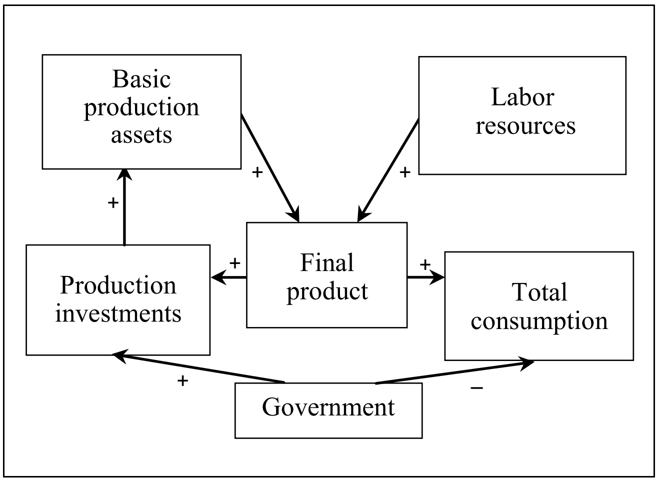


Fig. 2. The cognitive map of the aggregate national economy model.

The model based on this map has the form

$$K(t+1) = (1-\mu)K(t) + I(t),\ K(0) = K_0; \quad (25)$$

$$L(t+1) = (1+\eta)L(t),\ L(0) = L_0; \quad (26)$$

$$Y(t+1) = AK^{\alpha}(t+1)L^{1-\alpha}(t+1),\ 0 \le \alpha \le 1; \quad (27)$$

$$I(t+1) = s(t+1)Y(t+1),\quad C(t+1) = (1 - s(t+1))Y(t+1); \quad (28)$$

$$0 \le s(t+1) \le 1,\ t = 0, 1, ..., T-1. \quad (29)$$

Here $K(t)$ is the amount of basic production assets; $L(t)$ is the amount of labor resources; $Y(t)$ is the final product of the economy; $I(t)$ is the amount of production investments; $C(t)$ is the consumption level; $s(t)$ is the share of investments in the final product (all these parameters, in year *t*); $\mu > 0$ is the amortization rate of production assets; $\eta > 0$ is the reproduction rate of labor resources; finally, K_0 and L_0 are the initial values of the variables K and L, respectively.

The optimal control problem has the form

$$J = \sum_{t=1}^{T} c(t) \to \max, \; 0 \le s(t) \le 1, \quad (30)$$

$$K(T) \ge K^*, \; L(T) \ge L^*, \quad (31)$$

where $c(t) = C(t)/L(t)$ is the specific consumption (per one employee) and K^* and L^* are the target values of the indicators. In this problem, the sustainable development condition (31) can be treated as the government's goal. ♦

3. THE COGNITIVE OPTIMAL MANAGEMENT MODEL OF A UNIVERSITY

As a detailed example, we consider the cognitive optimal management model of a separate university.

The university enrolls in M specialties. Students can study in the university on state-funded (budgetary) or commercial places. Budgetary places in the university are allocated by the government: their number does not directly depend on the university management. The number of commercial places for particular specialties can be set by the university independently. Also, the university determines the price of commercial education for each specialty and bears the costs of educating a given number of students (on commercial and budgetary places). Some commercial places provided by a university can remain unclaimed by applicants. We assume that the demand for commercial places in a specialty is directly proportional to the future wage of a graduate and inversely proportional to the price of commercial education. Also, the more graduates of a specialty are employed, the higher attractiveness it will have for applicants. For a chosen planning horizon T, we obtain an optimization model of the form:

$$J = \sum_{t=1}^{T} \sum_{j=1}^{M} \left(a_j^C x_j^C(t) - c_j \left(x_j(t) \right)^2 \right) \to \max, \quad (32)$$

$$x_j^C(t) \ge 0, a_j^C(t) \ge 0; \quad (33)$$

$$x_j(t+1) = x_j^B(t+1) + \min\{x_j^C(t+1), (\gamma_j - a_j^C(t)/4)^{\alpha_j} y_j(t)\}, \; x_j(0) = x_{j0}; \quad (34)$$

$$y_j(t+1) = (1-\kappa_j) x_j(t), \; y_j(0) = y_{j0}, \; j = 1,\ldots, M, \; t = 1,\ldots, T-1. \quad (35)$$

Here, M is the number of specialties; $x_j^B(t)$ is the number of budgetary places for the j-th specialty in year t (the exogenous variable); $x_j^C(t)$ is the number of commercial places for the j-th specialty in year t (the first control variable); $x_j(t)$ is the total number of places for the jth specialty in year t; $a_{ij}^C(t)$ is the price of commercial education for the j-th specialty in year t (the second control variable); γ_j is the influence coefficient of potential employment on applying to the j-th specialty; κ_i is the share of unemployed graduates with the jth specialty; c_j is the education cost coefficient for the jth specialty depending on the total number of students in year t; $y_j(t)$ is the number of employed graduates with the j-th specialty in year t; finally, α_j is the elasticity of demand for commercial places in the jth specialty. The model is reflected in a cognitive map (Fig. 3).

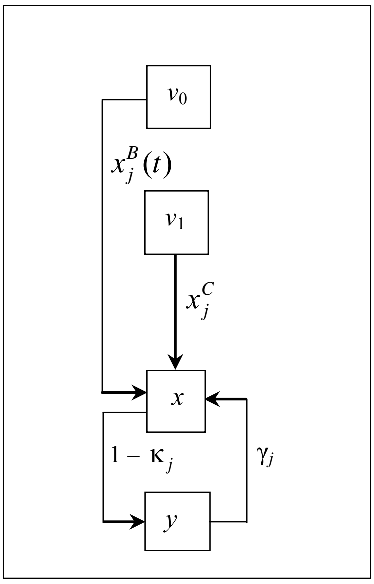


Fig. 3. The cognitive map of the university management model.

The demand of applicants for commercial places in the jth specialty (see the function (34)) is given by

$$(\gamma_j - a_j^C(t)/4)^{\alpha_j} y_j(t).$$

The expression in parentheses can be explained as follows. By some estimates [39], long-term investments in higher education in Russia have an average payback period of 10 years: the price of commercial education is covered by future wages in a specialty at the rate of at least 10% annually. According to the Tabiturient portal (https://tabiturient.ru/vuzcost/), the average price of higher education in Russia in 2022 is 174 533 rubles per year (about 17 453 rubles per month). As discovered by the analytical center of Synergy University (https://ria.ru/20210914/zarplata-1749940260.html), the average desired wage is about 50–70 thousand rubles per month, which provides a sufficient level of comfortable life. Thus, for an effective return on investment in higher education in Russia, the future wage of a graduate should be at least three times higher than the monthly expenditures on education.

Let us introduce two simplifying assumptions:

- All budgetary places allocated are filled.
- There is no expulsion, i.e., all university applicants become graduates.

Thus, at each time instant, a university is informed by the government about the number of budgetary places allocated for each specialty. This is important for universities since education costs depend on the total number of students. After the university receives information on budgetary places, it decides on the maximum possible enrollment of commercial students and determines the price of commercial education in each specialty.

We investigate the model by computer simulation [40]. The uncontrolled parameters of the model are identified, and then different control scenarios are analyzed. Each scenario consists in specifying:

- the vector of exogenous variables

$$\{x_{ij}^B(t)\},\ \ j=1,..., M, t=0,..., T-1,$$

- the vector of control variables

$$\{a_j^C(t), x_j^C(t)\},\ \ j=1,..., M, t=0,..., T-1.$$

Six enlarged groups of specialties were taken for the study as the most important ones: pedagogy, medicine, economics, engineering, construction, and agriculture. Then the key universities of the Rostov region with the corresponding specialties were selected: Rostov State Medical University (RostSMU), Don State Technical University (DSTU), and Southern Federal University (SFedU).

RostSMU is a specialized regional university with 11 faculties (medicine, pharmacology, and psychology). DSTU is a regional supporting multidisciplinary university of the Rostov region with 24 faculties (engineering, agriculture, and social and humanitarian specialties). SFedU is the largest research and educational center in the South of Russia, which includes 29 structural units (natural sciences, engineering, and social and humanitarian specialties).

Table 1 shows the specialties of these universities.

Table 1

The specialties of universities in the Rostov region

Specialties	RostSMU	DSTU	SFedU
Pedagogy		+	+
Medicine	+		
Economics		+	+
Engineering		+	+
Construction		+	+
Agriculture		+	

Let us describe the parameter identification procedure. The components of the vector

$$\left(\{\gamma_j\}_{j=1}^M, \{\kappa_j\}_{j=1}^M, \{c_j\}_{j=1}^M, \{x_{j0}\}_{j=1}^M, \{y_{j0}\}_{j=1}^M, \{\alpha_j\}_{j=1}^M\right),$$

which form the uncontrolled parameters of the model, were to be identified. Consider them in detail.

The parameter γ_j is the influence coefficient of potential employment on applying to the *j*th specialty. As this parameter, we took the average wage of the corresponding profession in the Rostov region. Note that its value does not depend on a particular university. The data were provided by the territorial body of the Federal State Statistics Service in the Rostov region (Rostovstat; see https://rostov.gks.ru). For each industry and specialty, the average wages were calculated for several years. The data for 2020 were taken as the parameter γ_j.

Table 2

The influence coefficient of potential employment on applying to specialties

Specialties	Parameter	Value, in roubles
Pedagogy	γ_1	28550
Medicine	γ_2	35849
Economics	γ_3	35000
Engineering	γ_4	53000
Construction	γ_5	47000
Agriculture	γ_6	23726

The parameter x_{j0} is the number of graduates for the *j*-th specialty in the initial year of the planning horizon. The data were taken from public documents (self-evaluation reports, enrollment orders, and enrollment statistics by year) on the official portals of RostSMU, DSTU, and SFedU. The number of graduates and enrolled students for 2020 was considered (Table 3).

The parameter κ_j is the share of unemployed graduates with the *j*th specialty. This parameter was calculated based on public documents on the official portals of the universities. The share of unemployed graduates was determined by subtracting that of employed graduates from 1. For the calculations, this parameter was set equal to 1 for the universities without appropriate specialties. The resulting values are presented in Table 4. The parameter y_{j0} is the number of employed graduates with the *j*th specialty in the initial year of the planning horizon. It was calculated (see Table 5) through the parameters x_{j0} and κ_j by the formula

$$y_{j0} = \left(1-\kappa_j\right)x_{j0}.$$

Table 3

The number of graduates in the initial year

Specialties	Parameter	RostSMU	DSTU	SFedU
Pedagogy	x_{10}	–	40	666
Medicine	x_{20}	165	–	–
Economics	x_{30}	–	319	296
Engineering	x_{40}	–	675	895
Construction	x_{50}	–	586	150
Agriculture	x_{60}	–	138	–

Table 4

The share of unemployed graduates

Specialties	Parameter	RostSMU	DSTU	SFedU
Pedagogy	κ_1	–	0.45	0.45
Medicine	κ_2	0.16	–	–
Economics	κ_3	–	0.45	0.18
Engineering	κ_4	–	0.45	0.18
Construction	κ_5	–	0.45	0.07
Agriculture	κ_6	–	0.45	–

Table 5

The number of employed graduates in the initial year

Specialties	Parameter	RostSMU	DSTU	SFedU
Pedagogy	y_{10}	–	22	366
Medicine	y_{20}	139	–	–
Economics	y_{30}	–	171	213
Engineering	y_{40}	–	371	644
Construction	y_{50}	–	323	144
Agriculture	y_{60}	–	76	–

The parameter c_j is the education cost coefficient for the *j*th specialty depending on the total number of students in year *t*. It is directly related to the prime cost of tutoring in this specialty. This parameter was assigned through expertise as 80% of the price of commercial education available from public sources: the official portals of SFedU (https://sfedu.ru), DSTU (https://donstu.ru), and RostSMU (http://rostgmu.ru). See Table 6 below.

The values in Table 6 are not the values of the parameter c_j. Assuming quadratic costs, the value c_j is given by

$$c_j = \frac{c}{x_{j0}},$$

where c denotes the prime cost of tutoring.

Table 6

The prime cost of tutoring, in roubles

Specialties	RostSMU	DSTU	SFedU
Pedagogy	–	86 000	88 000
Medicine	125 000	–	–
Economics	–	86 000	107 000
Engineering	–	100 000	104 000
Construction	–	100 000	113 000
Agriculture	–	100 000	–

The resulting values of the parameter c_j are combined in Table 7. For the calculations, this parameter was set equal to almost infinity for the universities without appropriate specialties.

The parameter α_j is the elasticity of demand for commercial places in the *j*th specialty. It characterizes demand variations under changing the future wage or the price of commercial education. The data were taken from https://iq.hse.ru/news/177671083.html (IQ: Research and Education Website, National Research University Higher School of Economics). The cited source indicates the relative variation *Pov* under increasing the demand of applicants for a specialty with a 40% increase in graduate wages.

Therefore, we calculated this parameter by the formula

$$\alpha_j = \log_{1.4}\left(1 + \frac{Pov}{100}\right);$$

see Table 8.

Even at the identification stage, we arrive at the following conclusion: it is unprofitable for applicants to study medicine and agriculture on commercial places. Really, the expression $\gamma_j - a_j^C(t)/4$ (the basis for calculating the demand) is negative even at the prime cost of tutoring. For agriculture, it can be explained by low wages; in the case of medicine, the reason is the high prime cost of tutoring. Engineering and construction attract applicants for commercial places with high future wages. Economics and pedagogy lie at the borderline: the future wages are commensurate with the price of commercial education.

For the prices $a_j^C(t) > 4\gamma_j$, there is no demand for commercial education: see medicine and agriculture as examples. Therefore, the problem for RostSMU has a trivial solution and will not be considered below.

In view of (34), the university need not enroll commercial students above the demand $(\gamma_j - a_j^C(t)/4)^{\alpha_j} y_j(t)$. (Although universities incur no losses from the excessive commercial enrollment.) When increasing the number of students, the costs grow faster than the income (32). Hence, there exists a finite optimal number of commercial students for the university: for this number, the goal function (32) achieves maximum. The university should enroll precisely this number of commercial students.

The optimal control problem for the university is solved in two stages.

- For each specialty, it is required to determine the maximum number of commercial students $\{x_j^C(t)\}$, $j = 1,\dots, M$, $t = 0,\dots, T-1$, profitable for the university considering the goal function (32) and the demand for commercial places (34).
- For each specialty, it is required to select the maximum price of commercial education $\{a_j^C(t)\}$, $j = 1,\dots, M$, $t = 0,\dots, T-1$, that maximizes the function (32). The price is determined as a markup to the prime cost of tutoring.

If the university does not receive any budgetary places, the model takes the following form:

$$J = \sum_{t=1}^{T}\sum_{j=1}^{M}\left(a_j^C x_j(t) - c_j\left(x_j(t)\right)^2\right) \to \max,$$

$$x_j^C(t) \ge 0,\ a_j^C(t) \ge 0;$$

$$x_j(t+1) = \min\{x_j^C(t+1), (\gamma_j - a_j^C(t)/4)^{\alpha_j} y_j(t)\},$$

$$x_j(0) = x_{j0};$$

$$y_j(t+1) = (1-\kappa_j)x_j(t),\ y_j(0) = y_{j0},$$

$$j = 1,\dots, M,\ t = 1,\dots, T-1.$$

The calculation results for DSTU and SFedU are shown in Tables 9 and 10, respectively.

Table 7

The education cost coefficient

Specialties	Parameter	RostSMU	DSTU	SFedU
Pedagogy	c_1	–	2150	132
Medicine	c_2	758	–	–
Economics	c_3	–	270	361
Engineering	c_4	–	148	116
Construction	c_5	–	171	753
Agriculture	c_6	–	725	–

Table 8

The elasticity of demand for commercial places

Specialties	*Pov*	α_j
Pedagogy	76	$\alpha_1 = \log_{1.4} 1.76 = 1.68$
Medicine	131	$\alpha_2 = \log_{1.4} 2.31 = 2.48$
Economics	77	$\alpha_3 = \log_{1.4} 1.77 = 1.70$
Engineering	42	$\alpha_4 = \log_{1.4} 1.42 = 1.04$
Construction	51	$\alpha_5 = \log_{1.4} 1.51 = 1.22$
Agriculture	41	$\alpha_6 = \log_{1.4} 1.41 = 1.02$

Table 9

Calculation results for DSTU (no budgetary places)

Specialties*	Academic year					
	First		Second		Third	
	Commercial enrollment	Markup, %	Commercial enrollment	Markup, %	Commercial enrollment	Markup, %
Pedagogy	100	3	38	3	95	3
Engineering	56	30	63	30	5	30
Construction	67	19	65	19	8	19
Economics	130	9	85	9	35	9
* There is no information about agriculture due to no demand.						

Table 10

Calculation results for SFedU (no budgetary places)

Specialties*	Academic year					
	First		Second		Third	
	Commercial enrollment	Markup, %	Commercial enrollment	Markup, %	Commercial enrollment	Markup, %
Pedagogy	250	11	250	11	250	11
Engineering	86	18	98	18	9	18
Construction	28	1	27	1	5	1
Economics	250	8	250	8	250	8
* There is no information about agriculture due to no demand.						

At SFedU, admission to pedagogy and economics is limited by the university's capacity and does not exceed 250 places. The demand for commercial places for these specialties is above 250. Note that generally, the demand for commercial places decreases over time due to the rational search for budgetary ones. Consequently, universities have to reduce the price of commercial education. The new values are presented in Table 11.

According to the calculation results, commercial enrollment is not profitable for the university: under a high price of education, there is no demand for commercial places; under a low price, the university suffers losses due to the quadratic costs.

Table 11

Additional budgetary places for DSTU

Specialty	$x_j^B(t)$
Pedagogy	1988
Economics	816
Engineering	459
Construction	331
Agriculture	331

Note that this study covers only the first level of higher education (bachelor's degree). In the case of master's and other postgraduate programs, all qualitative conclusions will remain valid, whereas quantitative conclusions will rest on the identification of model parameters. The demand for postgraduate commercial places can be ensured under the following conditions:

- The employer must clearly understand the increased qualifications of a graduate with a master's degree (and realize the potential utility of attracting him or her to a higher wage) compared to an employee with a bachelor's degree. The wage differential should be motivating when deciding to invest in a master's degree.

In other words, the wage jump must satisfy the condition

$$\gamma_j^{\text{mast}} - \gamma_j^{\text{bach}} > \frac{a_j^C(t)}{4}.$$

There should be a greater market demand for employees with higher qualifications (master's degree), who have competencies lacking in bachelor's degree and a greater potential utility for employers. This assertion is confirmed especially during the systemic economic recession (a reduced supply of jobs and an increased level of unemployment).

CONCLUSIONS

This paper has presented a cognitive modeling methodology under different types of control (optimal, conflict, and hierarchical).

A cognitive optimization model of a university has been studied. The analysis allows drawing several conclusions as follows. Within the model, if a university is allocated many budgetary places in a specialty, it has low demand for commercial enrollment. In this case, commercial places in excess of the budgetary ones incur losses. (The model does not allow the university's refusal from the allocated budgetary places.) A large number of commercial places remains economically justified only in case of no budgetary enrollment (SFedU). Allocating a small number of budgetary places has economic advantages: the university provides the maximum of commercial places within the resource potential and gains higher income.

The university and the government are recommended to choose the priority of enrollment (budgetary or commercial). If budgetary places are the priority, the control problem will have another statement, yielding other conclusions. Changes are also possible in the case of considering master's degree programs and university rankings.

It seems very promising to combine cognitive simulation modeling with the mathematical apparatus of network games, which has been intensively developed recently [41].

As we believe, there are no fundamental theoretical limitations on the applicability of this methodology. It is possible to simulate any dynamic active systems of any complexity. However, there may arise technical limitations related to the model dimension and the need to collect and process relevant data. Following standard practice in applied systems analysis, we have to compromise between the desired accuracy and the capabilities of study.

The model adequacy should be assessed primarily through the meaningful analysis of the results based on expert opinions. Of course, it is possible to apply traditional methods of applied statistics (e.g., hypothesis testing). Their usefulness, however, seems limited due to obvious difficulties when satisfying formal requirements for the available data. In addition, when modeling complex socio-economic systems, the conclusions and recommendations are mainly qualitative in nature. They should be validated through expertise.

The opinion of the expert community is even more useful when interpreting the results. We have not yet considered in detail the well-known methods to aggregate and process expert information [42]. This is one subject for further research.

REFERENCES

1. *Cognitive Modeling*, Polk, T.A. and Seifert, C.M., Eds., MIT Press, 2002.
2. Axelrod, R., *The Structure of Decision: Cognitive Maps of Political Elites*, Princeton: University Press, 1976.
3. Maruyama, M., The Second Cybernetics: Deviation-Amplifying Mutual Causal Processes, *American Scientist*, 1963, vol. 51, no. 2, pp. 164–179.
4. Roberts, F., *Discrete Mathematical Models with Applications to Social, Biological and Ecological Problems*, Pearson, 1976.
5. Papageorgiou, E., Review Study on Fuzzy Cognitive Maps and Their Applications during the Last Decade, in *Business Process Management*, SCI444, Glykas, M., Ed., Springer, 2013, pp. 281–298.
6. Groumpos, P., Modelling and Analyzing Manufacturing Systems Using Advanced Methods of Fuzzy Cognitive Maps, *J. of Computational Intelligence and Electronic Systems*, 2014, vol. 3, no. 2, pp. 143–150.
7. Benjamín, G., Nápoles, G., Falcon, R., et al., A Review on Methods and Software for Fuzzy Cognitive Maps, *Artificial Intelligence Review*, 2017. DOI: 10.1007/s10462-017- 9575-1.
8. Abramova, N.A., Expert Verification in Formal Cognitive Map Application. Approaches and Practices, *Large-scale Systems Control*, 2010, no. 30-1, pp. 371–410. (In Russian.)
9. Abramova, N.A., On the Prospects of the Modern Paradigm of Cognitive Modeling, *Proceedings of the 13th All-Russian Meeting on Control Problems (AMCP-2019)*, Trapeznikov Institute of Control Sciences RAS, 2019, pp. 1858–1863. (In Russian.)

10. Abramova, N.A. and Avdeeva, Z.K., Cognitive Analysis and Management of Situation Development: Problems of Methodology, Theory and Practice, *Control Sciences*, 2008, no. 3, pp. 85–87. (In Russian.)
11. Avdeeva, Z.K. and Kovriga, S.V., On the Statement of Control Problems of the Situation with Many Active Stakeholders with Use of Cognitive Maps, *Large-Scale Systems Control*, 2017, no. 68, pp. 74–99. (In Russian.)
12. Avdeeva, Z.K. and Kovriga S.V., Some Principles and Approaches to Construction of Group Cognitive Maps, *Large-Scale Systems Control*, 2014, no. 52, pp. 37–68. (In Russian.)
13. Gorelova, G.V., Cognitive Modeling of Complex Systems: State and Prospects, *Trudy 25-oi Mezhdunarodnoi nauchnoi i uchebno-prakticheskoi konferentsii "Sistemnyi analiz v proektorovanii i upravlenii"* (Proceedings of the 25th International Scientific and Practical Conference "System Analysis in Design and Management"), St. Petersburg, 2021, pp. 224–248. (In Russian.)
14. Gorelova, G.V., Cognitive Approach to Simulation of Large Systems, *Izvestiya SFedU. Engineering Sciences*, 2013, no. 103 (40), pp. 239–250. (In Russian.)
15. Gorelova, G.V., Zakharova, E.N., and Radchenko, S.A., *Issledovanie slabostrukturirovannykh problem sotsial'no-ekonomicheskikh sistem: kognitivnyi podkhod* (Study of Weakly Structured Problems of Socio-economic Systems: A Cognitive Approach), Rostov-on-Don: Rostov State University, 2006. (In Russian.)
16. Kornoushenko, E.K. and Maksimov, V.I., Situation Control Using Structural Properties of Its Cognitive Map, in *Trudy Instituta problem upravleniya im. V.A. Trapeznikova RAN* (Proceedings of Trapeznikov Institute of Control Problems RAS), Moscow, 2000, vol. 11, pp. 85–90. (In Russian.)
17. Kovriga, S.V. and Maksimov, V.I., A Cognitive Modeling Technology for the Purposeful Development of Russian Regions, in *Trudy Instituta problem upravleniya im. V.A. Trapeznikova RAN* (Proceedings of Trapeznikov Institute of Control Problems RAS), Moscow, 2000, vol. 11, pp. 91–103. (In Russian.)
18. Maksimov, V.I., Structure and Goal Analysis of the Development of Socio-economic Situations, *Control Sciences*, 2005, no. 3, pp. 30–38. (In Russian.)
19. Maksimov, V.I. and Kovriga, S.V., Application of Structure and Goal Analysis of the Development of Socio-economic Situations, *Control Sciences*, 2005, no. 3, pp. 39–44. (In Russian.)
20. Arzhenovskiy, I.V. and Dakhin, A.V., Cognitive Regionology: The Experience of Modeling Regional Socio-economic Processes, *Russian Journal of Regional Studies*, 2020, vol. 28, no. 3, pp. 470–489. (In Russian.)
21. Makarova, E.L. and Firsova, A.A., Cognitive Modeling the Impact of a Regional System of Higher Education in the Innovative Development of the Region, *Izv. Saratov University. Economics. Management. Law*, 2015, vol. 5, no. 4, pp. 411–416. (In Russian.)
22. Arzy, S. and Kaplan, R., Transforming Social Perspectives with Cognitive Maps, *Social Cognitive and Affective Neuroscience*, 2022, vol. 17, iss. 10, pp. 939–955.
23. Napoles, G., Grau, I., Conception, L., et al., Modeling Implicit Bias with Fuzzy Cognitive Maps, *Neurocomputing*, 2022, vol. 481, pp. 33–45.
24. Whittington, J.C.R., McCaffary, D., Bakermans, J.J.W., and Behrens, T.E.J., How to Build a Cognitive Map: Insights from Models of the Hippocampal Formation, *arXiv: 2202.01682 [q-bio.NC]*. https://doi.org/10.48550/arXiv.2202.01682
25. Burkov, V.N. and Novikov, D.A., *Teoriya aktivnykh sistem: sostoyanie i perspektivy* (Active Systems Theory: State and Prospects), Moscow: Sinteg, 1999. (In Russian.)
26. Novikov, D.A., *Theory of Control in Organizations*, New York: Nova Science Publishers, 2013.
27. Gorelik, V.A. and Kononenko, A.F., *Teoretiko-igrovye modeli prinyatiya reshenii v ekologo-ekonomicheskikh sistemakh* (Game-Theoretic Models of Decision-Making in Ecological-Economic Systems), Moscow: Radio i Svyaz', 1982. (In Russian.)
28. Gorelik, V.A., Gorelov, M.A., and Kononenko, A.F., *Analiz konfliktnykh situatsii v sistemakh upravleniya* (Conflict Situation Analysis in Control Systems), Moscow: Radio i Svyaz', 1991. (In Russian.)
29. Laffont, J.-J. and Martimort, D., *The Theory of Incentives: The Principal-Agent Model*, Princeton University Press, 2002.
30. Ougolnitsky, G.A., *Upravlenie ustoichivym razvitiem aktivnykh system* (Sustainable Management of Active Systems), Rostov-on-Don: Southern Federal University, 2016. (In Russian.)
31. Pontryagin, L.S., Boltyansky, V.G., Gamkrelidze, R.V., and Mischenko, E.F., *Mathematical Theory of Optimal Processes*, New York–London: Interscience Publishers, John Wiley & Sons, 1962.
32. *Optimal Control of Nonlinear Processes (with Applications to Drugs, Corruption, and Terror)*, Grass, D., Caulkins, J.P., Feichtinger, G., Tragler, G., and Behrens, D.A., Eds., Springer, 2008.
33. *Differential Games in Economics and Management Science*, Dockner, E., Jorgensen, S., Long, N.V., and Sorger, G., Eds., Cambridge: Cambridge University Press, 2000.
34. Gorelov, M.A. and Kononenko, A.F., Dynamic Models of Conflicts. III. Hierarchical Games, *Automation and Remote Control*, 2015, vol. 76, no. 2, pp. 264–277.
35. Ougolnitsky, G.A. and Usov, A.B., Computer Simulations as a Solution Method for Differential Games, in *Computer Simulations: Advances in Research and Applications*, Pfeffer, M.D. and Bachmaier, E., Eds., New York: Nova Science Publishers, 2018, pp. 63–106.
36. Gubanov, D.A., Novikov, D.A., and Chkhartishvili, A.G., *Social Networks: Models of Information Influence, Control and Confrontation*, Cham: Springer, 2019.
37. Belov, M.V., and Novikov, D.A., Network Active Systems: Planning and Stimulation Models, *Control Sciences*, 2018, no. 1, pp. 47–57. (In Russian.)
38. Novikov, D.A., Cognitive Games: A Linear Step-Function Model, *Control Sciences*, 2008, no. 3, pp. 14–22. (In Russian.)
39. Antonenko, V.V. and Karaulova, N.M., On Payback Period of Investment in Education, *Finance and Credit*, 2015, no. 40 (644), pp. 10–21. (In Russian.)
40. Gorbaneva, O., Murzin, A., and Ougolnitsky, G., Control of Socio-economic Systems Based on Cognitive Modeling. Certificate of state registration of the computer program No. 2022611893 dated February 4, 2022. (In Russian.)
41. Novikov, D.A., Games and Networks, *Automation and Remote Control*, 2014, vol. 75, no. 6, pp. 1145–1154.
42. Pavlov, A.N. and Sokolov, B.V., *Metody obrabotki ekspertnoi informatsii* (Processing Methods for Expert Information), St. Petersburg: State University of Aerospace Instrumentation, 2005. (In Russian.)

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

Received August 8, 2022, and revised November 15, 2022. Accepted November 15, 2022.

Author information

Gorbaneva, Olga Ivanovna. Dr. Sci. (Eng.), Southern Federal University, Rostov-on-Don, Russia
✉ oigorbaneva@sfedu.ru

Murzin, Anton Dmitrievich. Dr. Sci. (Eng.), Southern Federal University, Rostov-on-Don, Russia
✉ admurzin@sfedu.ru

Ougolnitsky, Gennady Ivanovich. Dr. Sci. (Phys.–Math.), Southern Federal University, Rostov-on-Don, Russia
✉ gaugolnickiy@sfedu.ru

Cite this paper

Gorbaneva, O.I., Murzin, A.D., Ougolnitsky, G.A., A Mathematical Formulation of Control Problems on Cognitive Models. *Control Sciences* **5,** 21–33 (2022). http://doi.org/10.25728/cs.2022.5.3

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com

DOI: http://doi.org/10.25728/cs.2022.5.4

APPLICATION OF SEMIOTIC MODELS TO DECISION-MAKING

A. A. Kulinich

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia

✉ alexkul@rambler.ru

Abstract. This paper introduces an approach to building decision support systems based on a semiotic domain model and natural language processing methods. The knowledge base of this model is a text corpus of linguistic information obtained from the Internet. The text corpus is relevant to the subject domain in which the subjective semiotic model of the situation is constructed. A method for solving the inverse problem in a semiotic system is proposed. The obtained solutions are interpreted in the subject domain using a semantic calculator. The semantic calculator extracts generic relations from the text corpus based on lexico-syntactic patterns and determines the frequency of joint occurrence of words in the solution based on the distributive analysis of the text corpus. The generalized structures of monitoring and decision-making subsystems with the semiotic model of the situation and natural language processing methods are described. A software layout of the decision-making subsystem is developed. The effectiveness of this approach is demonstrated by experiments.

Keywords: decision-making, semiotic system, subjective model, natural language processing, distributive analysis.

INTRODUCTION

Currently, decision support methods in complex socio-economic and political systems under uncertainty can be classified as follows. The first class of methods (Data Mining) obtains general trends of a subject domain by extracting knowledge from data represented in numerical scales.

Another class of methods directly extracts expert knowledge (the best solution in choice models or situation forecasting models according to experts' views). These methods use the subjective preferences of experts, their assessments and knowledge of the general trends of the subject domain, etc. However, in this case, there are difficulties in constructing a mathematical model of the object and measuring its parameters. Under uncertainty, such an expert model conceptually simulates and qualitatively reflects the main trends of the situation. In such conditions, the model of the situation is difficult to verify; therefore, the simulation results are difficult to interpret in terms of the subject domain and are unreliable.

Decision-making methods using linguistic information about a controlled object were investigated within situation management [1]. Here, the natural language description of an object is represented in a restricted natural language through core structures, which include language elements and various relations between them. Such a description is called the object's state, and management is possible if there exists a natural language description of the control action for some target state. In situation management, it is necessary to enumerate all possible states of the controlled object and assign a control action in the natural language to each state. For complex objects, this problem becomes difficult to solve and requires much expert work.

The ideas of situation management were further developed within applied semiotics [2]. Here, the model of an object is constructed using sign-symbols. A sign-symbol was defined by German logician G. Frege as a triplet consisting of a name, a sense, and a sign meaning [3]. A sign symbol relates the knowledge of an expert (name and sense) with an object of the real world (sign meaning).

In [4], a sign was defined as a quadruple: a name, an image (percept), a meaning, and a personal sense. Here, the mathematical model of a sign, the operators of binding all its elements, and the operations on different sets of sign elements were defined. The authors placed emphasis on the recognition of perceptual images in the form of the connected sign picture of the world; this picture determines the behavior of a subject based on its experience (personal meaning).

In [2], the model of a semiotic system known as Pospelov's semiotic square was introduced. Pospelov's semiotic square includes the following elements: a metasign that defines the name of a semiotic system (the set of sign symbols); a syntax that defines the rules of building a sign system; semantics that define the basic properties of the semiotic system; pragmatics that define the basic actions performed within this semiotic system.

The main aspects of semiotic systems (syntax, semantics, and pragmatics) were formulated in the classical works of logicians Ch. Peirce [5] and J. Morris [6].

The semiotic approach has the following applications: information systems design [7]; computer systems design (*computer semiotics*) [8]; system interface representation in different but equivalent sign systems (*algebraic semiotics*) [9]; conceptual modeling in databases [10] (the extended entity-relationship model with frame algebra and data images).

The semiotic approach was adopted to solve complex strategic problems in power engineering and other critical infrastructures; for details, see [11, 12].

This paper considers the construction of semiotic decision support systems under uncertainty. For this purpose, we combine a qualitative semiotic model of the situation and technologies for obtaining relevant information from the Internet with various natural language processing methods. The qualitative subjective semiotic model is used as a pattern for obtaining relevant information from the Internet. The following problems are considered: situation monitoring and forecasting; decision support to manage the situation, including the interpretation of solutions and search for their precedents on the Internet.

1. THE ARCHITECTURE OF A SEMIOTIC DECISION SUPPORT SYSTEM

The architecture of control systems based on applied semiotics [1, 2, 13] was formed as part of research on situation management systems for complex objects.

The architecture of a semiotic decision support system is oriented to work with objects (situations) described in a natural language. It includes the following subsystems (Fig. 1):

- an input language interpreter, which translates unstructured linguistic information about the control object in a natural language into the internal language of the system;
- an analyzer, which preliminarily classifies the current situation into situations requiring (and not requiring) control;
- a classifier, which generalizes and reduces the current situation to one or more classes of typical situations from a knowledge base to apply one-step control actions;

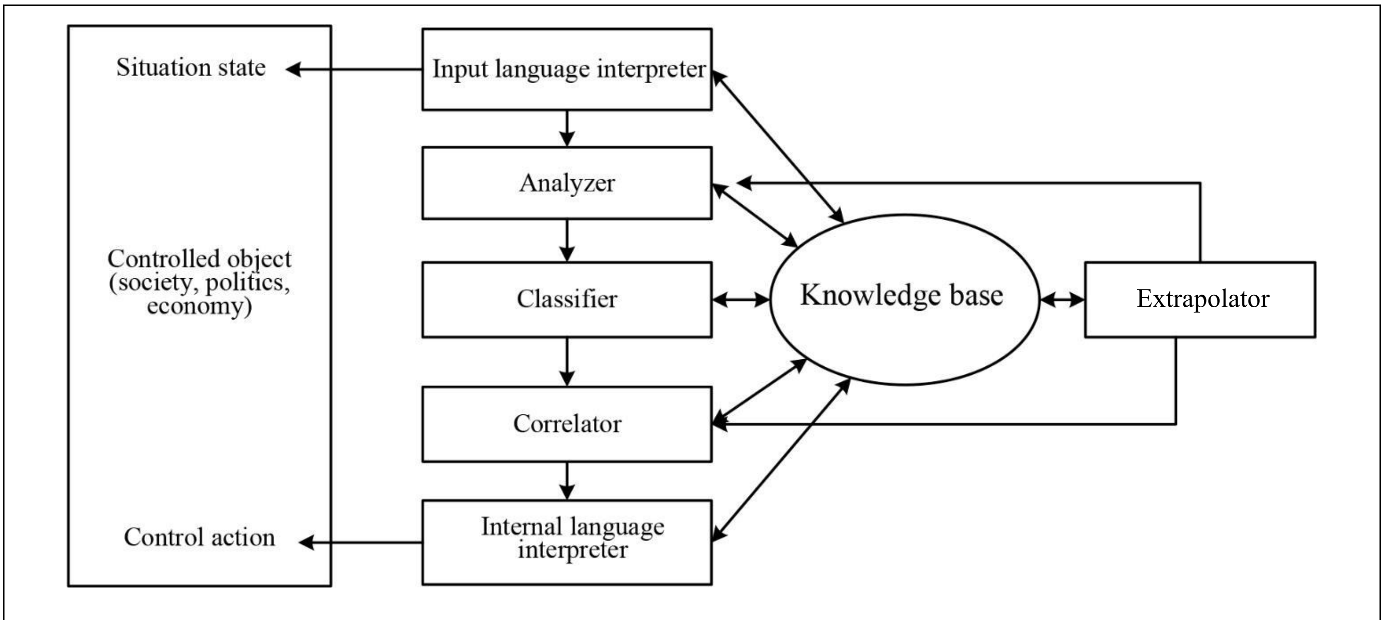


Fig. 1. The architecture of a semiotic system [13].

- a correlator, which forms adequate control actions for the controlled object in the current situations and the current world model from the knowledge base of the system;
- an extrapolator, which assesses the obtained solution based on situation forecasting in the current world model and possibly corrects it (by adjusting the world model in the “analyzer–classifier–correlator” cycle;
- an internal language interpreter, which presents the resulting solution in the natural language.

The main element of this architecture is a knowledge base, which includes a set of typical situations and models of possible worlds. The sequential transfer of the results from the interpreter to the analyzer, classifier, and then the correlator and extrapolator forms an operation cycle of the semiotic architecture.

In management systems built on the principles of applied semiotics, mathematical modeling of situations alternates with control design within fixed formal world models represented in the knowledge base. Semiotic modeling restructures formal world models based on knowledge about the subject domain and real situations that arise during the operation of the object and control system [13].

In this semiotic architecture, the models and possible worlds for decision-making represented in the knowledge base are closed with respect to knowledge of the problem domain.

Under uncertainty, the semiotic system must be open, i.e., it must have the capability to supplement knowledge. Therefore, the main distinctive feature of the semiotic decision support system considered here is that it uses information from the Internet as a knowledge base.

International and Russian standards provide different definitions of information. For example, in information technology, information is knowledge about facts, events, things, ideas, and concepts that has a particular meaning in a definite context (ISO/IEC 2382:2015); in information processing, information is any data, in electronic or any other form, to be processed by information and decision-making systems. Federal law no. 149-FZ “On Information, Information Technology, and Information Protection” interprets information as “knowledge (messages, data) regardless of its form of presentation.”

Data are defined as the presentation of information in a formalized way suitable for communication, interpretation, or processing (ISO/IEC 2382-1:2015).

Depending on the form of presentation, there exist structured and unstructured data. *Structured data* are organized according to a predefined set of rules. A predefined set of rules that regulates the basis of structured data must be clearly established and made public; it can be used to manage data structuring (ISO/IEC 20546:2019(ru), 3.1.35). Examples of structured data are data from database tables or manually or automatically marked-up text. During the markup procedure, certain labels (tags) are assigned to text words. As a result, information can be presented in tabular form and processed.

Unstructured data are characterized by the absence of any structure other than that at the record or file level. An example of unstructured data is free text (ISO/IEC 20546:2019(ru), 3.1.37). Formally, free text has a syntactic structure and conveys a particular meaning. However, text has to be preprocessed with natural language processing and intelligent analysis methods to retrieve information from it.

Note that the architecture described in [13] is abstract. The main ideas were formulated therein: how such a semiotic system and its subsystems can function. Unfortunately, this architecture is impossible to implement. Below, we propose a possible implementation of a semiotic decision support system under uncertainty. The proposed architecture includes a semiotic model, a subsystem to monitor the situation, and a subsystem to generate solution alternatives and explain them. The internal language is set by a subjective semiotic model built by an expert. The input language interpreter is the state monitoring subsystem. This subsystem obtains free text from the Internet and represents it in the internal language of the semiotic system. The decision-making subsystem is based on solving the inverse problem in the semiotic system. In addition, this subsystem implements an interpreter of the internal language. The interpreter translates the solutions obtained in the semiotic system into natural language, naming the class of solutions and explaining the context in which the name is used (this sentence is free text).

Under uncertainty, we propose using unstructured Internet data (free text) as a knowledge base, which is pre-structured by natural language processing methods.

2. THE SEMIOTIC MODEL OF THE SITUATION

The semiotic model of a subject domain proposed in [14] serves as a qualitative subjective model of the situation. The semiotic model of the situation [14] is a subjective qualitative model that represents expert’s knowledge of dynamic systems. Its elements are G. Frege’s sign model [3] connecting the real world (denotation) with mental representations about the world (knowledge) in the form of a sign name (symbol) and meaning, defining its main features (properties). The semiotic model describes the situation in three aspects: syntactic, semantic, and pragmatic. Syntax is responsible for representing the relations between signs describing the reality. Semantics studies the relationship between signs and their significations in the real world. Pragmatics is responsible for the relationship between signs and their users. In the case under consideration, the sign system user is the decision maker (DM).

The syntactic model. When constructing a syntactic model, we employ a logical-linguistic representation for the main elements of the system under study [15]. In logical-linguistic models, the basic elements, relations between them, and their possible states (values) are represented in a natural language. The syntactic model describes the main parts of the system as a set of their names $D = \{d_i\}$, $i = 1,..., M$. The "part–whole" relation is defined on this set: $\Theta \subseteq D \times D$. For each constituent part d_i of the simulated situation, the names of its parameters form the set $F_i = \{f_{ij}\}$, where j is the parameter number of the ith part. The value set $Z_i = \{Z_{ij}\}$ of each parameter is defined as an ordered set of linguistic values, i.e., $Z_{ij} = \{z_{ij1},..., z_{ijq}\}$, where $z_{ijq+1} \succ z_{ijq}$, $q = 0,..., n - 1$. The vector $Z(t) = (z_{1je},..., z_{njq})$ with the values of all situation parameters at a time instant t is called the state of the situation.

The logical-linguistic model is intended to represent the dynamics of all situation parameters and solve the inverse problem (find the parameter values for changing the current state of the situation to the target state). To model the dynamics, we have to determine the cause-and-effect (causal) relations between the parameters and their strength.

The strength of a causality is defined in a natural language from the set of possible values, e.g., RF = {"*Heavily strengthens*," "*Strengthens*," "*Slightly strengthens*"}. The strength of a causality defines a binary relation between the sets of possible parameter values. For example, RF("*Heavily strengthens*") $\subseteq Z_{1j} \times Z_{2u}$ means that the jth parameter of the first subsystem (cause) is related to the uth parameter of the second subsystem (effect). The strength of influence is defined by the pairs of values from the set $Z_{1j} \times Z_{2u}$. As an example, we take the "*Heavily strengthens*" relation; for this relation, the pairs of values $(z_{1j2}, z_{2u3};..., z_{1jn}, z_{2um})$ mean that changing the value of the jth parameter of the first subsystem to the second element (z_{1j2}) of the value set Z_{1j} (cause) will increase the value of the uth parameter of the second subsystem to the third element (z_{2u3}) of the set Z_{2u}, etc.

In the syntactic model, the causal relations between different parameters are determined through expertise as a relation W on the value sets of all parameters. Consider this relation in the form of logical-linguistic equations for the system with f_i parameters ($i = 1,..., n$) and their value sets Z_i. To forecast the situation, we write a causal relation W as a mapping [15]

$$W: Z(t) \rightarrow Z(t + 1), \tag{1}$$

where $Z(t) \in \underset{i}{\times} Z_i$, $Z(t) = (z_{1e},..., z_{nq})$ is the state vector of the situation, and $\underset{i}{\times} Z_i$ denotes the set of all possible state vectors ($i = 1,..., n$).

Situation forecasts in the model specified by logical-linguistic equations are often calculated using the theory of fuzzy sets and systems [16]. In this case, membership functions and fuzzy causal relations have to be defined for all parameters. Note that for fuzzy forecasting, all theoretical issues have been settled. However, the procedure of constructing the membership functions and defining the fuzzy relations requires much expert work.

In [17], B. Kosko proposed a fuzzy causal algebra without the need to construct membership functions: it suffices to obtain ordered linguistic values of the strength of causal relations. His idea is to calculate the influence of one parameter on another through chains of causal relations. The strength of influence of a chain is determined by the minimum of all strengths in the chain, and the aggregate strength of all chains is determined as the maximum of all strengths for the parameter of these chains.

In what follows, we forecast the situation using the method proposed in the paper [18].

We construct the linguistic scale of a parameter $Z_{ij} = \{z_{ij1},..., z_{ijq}\}$ as a mapping into a numerical set $X_{ij} = \{x_{ij1},..., x_{ijq}\}$ whose elements are defined on a segment of the numerical axis [0, 1], i.e., $x_{ij1},..., x_{ijq} \in [0, 1]$. The points of the numerical axis form an ordered set X_{ij} of numerical points ordered the same way as the linguistic values: if $z_{ijq+1} \succ z_{ijq}$, then $x_{ijq+1} \succ x_{ijq}$. Thus, a mapping $\varphi: Z_{ij} \rightarrow X_{ij}$ is defined. The inverse mapping $\varphi^{-1}: X_{ij} \rightarrow Z_{ij}$ allows interpreting any value $x_{ijq} \in [0, 1]$ into a linguistic value $z_{ijq} \in Z_{ij}\ \forall q$.

In this case, the strength of a causality can be treated as a real-valued gain. The values of the cause and effect parameters have the relationship $x_{ijq} = w_{ijpt}^{*}\, x_{pto}$, where $x_{ijq} \in X_{ij}$ is the value of the effect parameter and $x_{pto} \in X_{pt}$ is the value of the cause parameter. The gain is $w_{ijpt}^{*} = 1$ for the "Strengthens" relation, $w_{ijpt}^{*} > 1$ for the "*Heavily strengthens*," and $w_{ijpt}^{*} < 1$ for the "*Slightly strengthens*." The issues of determining the strength of causalities (gains) were described in detail in [18].

The linear relationship between the values of the cause and effect parameters is justified under uncertainty with subjective assessments of the strength of the causality.

The situation forecast in the numerical system is calculated from the finite-difference equation

$$X(t+1) = W^{*}\circ X(t),$$

where: $X(t)$ and $X(t+1)$ denote the situation parameters vectors at sequential time instants; $W^{*} = |\ w_{ij}^{*}\ |_{n\times n}$ is the gain matrix; finally, ° specifies a rule for calculating the forecast values.

We calculate the forecast vector $X(t+1)$ by aggregating the max-product values (multiplication and taking the maximum). Therefore, the ith element of the forecast vector is given by the following rule:

$$x_i\ (t+1) = \max_{r=1,\dots,n} x_r\ (t)\ w_{ir}\quad \forall i.$$

The forecast vector can be written in the linguistic form: $Z(t+1) = \varphi^{-1}\ (X(t+1))$.

Thus, the syntactic model is defined by the quadruple

$$\langle F, Z, W, Z(t)\rangle, \qquad (2)$$

where: F is the set of parameters; Z is the set of parameter value sets; W is a causal relation on the set of parameter values; finally, $Z(t)$ is the state (the vector of all parameter values).

The semantic model. The semantic model of a subject domain describes possible states of the syntactic model (2) as a partially ordered set of the named classes of states.

Such a representation is based on interpreting the space of possible states of the dynamic system (1), $SS = \times_i\ Z_i$, as a *semantic space*.

In the feature semantic space, the situation states are represented as notions. Real situations (states-denotations) are defined by the names and value vectors of the attributes characterizing their content (meaning). In semantic spaces, situations with close parameter values form classes of states, and certain relations are defined between different classes ("class–subclass" or "genus–species"). In other words, a notional structure is defined.

The paper [19] proposed a method for structuring the state space SS of the dynamic system (1) into nested domains of possible states: $SS(d^H) \subset SS$. These domains have artificial names d^H determining the class of states of system (1), i.e., $SS(d^H) \Leftrightarrow d^H$, $H = 0,\dots, 3^N$, where N is the number of parameters.

For a system with two parameters (features), this method is illustrated in Figs. 2–5 below.

In particular, Fig. 2 shows an example of the semantic space for a situation with two features $F = \{f_1, f_2\}$ and value sets $Z_1 = \{Z_{11},\dots, Z_{1n}\}$ and $Z_2 = \{Z_{21},\dots, Z_{2m}\}$, respectively. The initial state $Z(0)$ of the situation is indicated by a point with the coordinates $Z(0) = (Z_{1q}, Z_{2s})$ in the space $SS = Z_1 \times Z_2$.

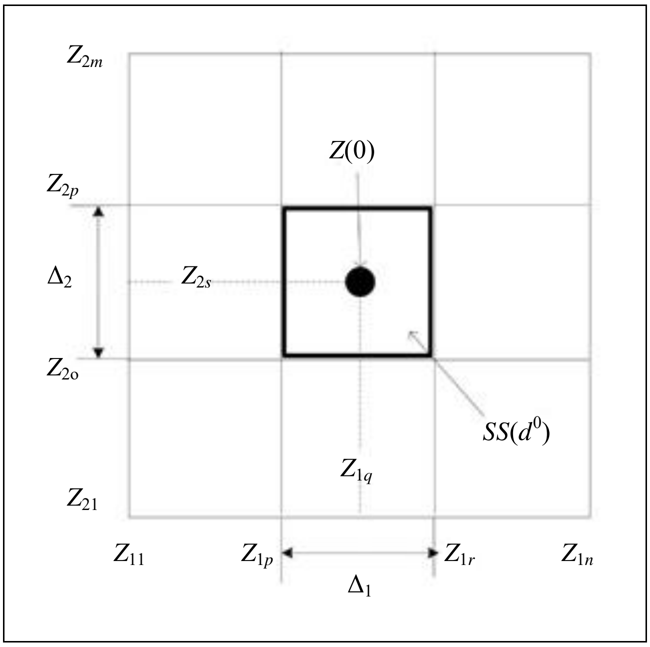


Fig. 2. The domain of the basic class of states.

The neighborhoods of the initial state point by features 1 and 2, $\Delta_1 = \{Z_{1p},\dots, Z_{1r}\}$ and $\Delta_2 = \{Z_{2o},\dots, Z_{2p}\}$, respectively, are assigned through expertise. They are called the initial state tolerance intervals by features 1 and 2. The semantic space domain $SS(d^0) \subseteq SS$ obtained by the direct product of all tolerance intervals (by all features of the state $Z(0)$) defines the basic class of states: $SS(d^0) = \Delta_1 \times \Delta_2$. Any state $Z(t)$ of system (1) from the basic class $SS(d^0)$, i.e., $Z(t) \in SS(d^0)$, has the name d^0. In other words, the basic class defines the class of indistinguishable, equivalent states.

Figure 3 demonstrates the semantic space domains $SS(d^1)$, $SS(d^2)$, $SS(d^3)$, and $SS(d^4)$. For example, the domain $SS(d^3)$ is defined as $SS(d^3) = \{Z_{2o},\dots, Z_{2p}\} \times \{Z_{11},\dots, Z_{1r}\} = \Delta_2 \times \Delta_3$. Since the domains $SS(d^1)$, $SS(d^2)$, $SS(d^3)$, and $SS(d^4)$ include the domain of the basic class of states $SS(d^0)$, they are said to generalize this class. Moreover, the domains $SS(d^1)$ and $SS(d^3)$ generalize the domain $SS(d^0)$ by feature 2 whereas $SS(d^2)$ and $SS(d^4)$ by feature 1. The domains $SS(d^1)$, $SS(d^2)$, $SS(d^3)$, and $SS(d^4)$ have the corresponding names (those of the state classes d^1, d^2, d^3, and d^4, respectively). In the following, we operate the names of the state classes.

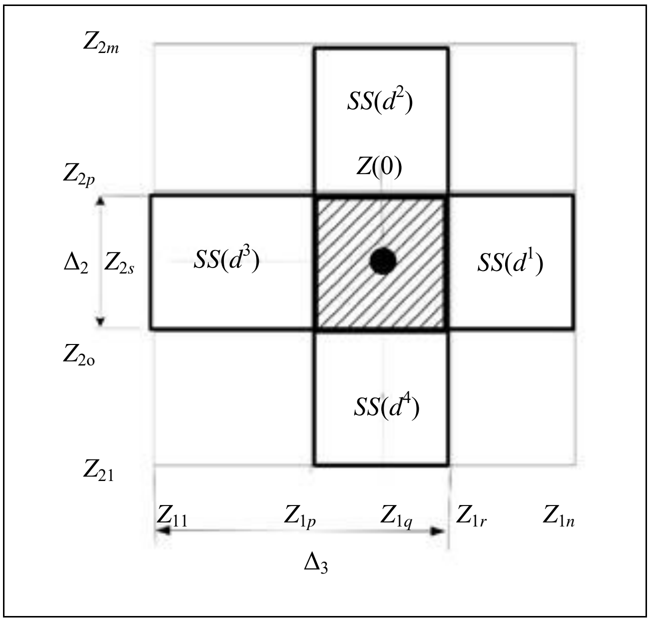


Fig. 3. The domains of generalized state classes by features 1 and 2.

Figure 4 shows the semantic space domains $SS(d^5)$, $SS(d^6)$, $SS(d^7)$, and $SS(d^8)$. For example, the domain $SS(d^5)$ is defined as $SS(d^5) = \{Z_{2o},..., Z_{2m}\} \times \{Z_{1r},..., Z_{1n}\} = \Delta_4 \times \Delta_3$. These domains include the basic class domain and generalized domains by one of the features. For example, $SS(d^5)$ includes the domains $SS(d^1)$ and $SS(d^2)$ and the basic class domain $SS(d^0)$. Due to such nesting, these domains generalize generalized domains by one feature and the basic state class by two features. These domains are denoted by the names d^5, d^6, d^7, and d^8.

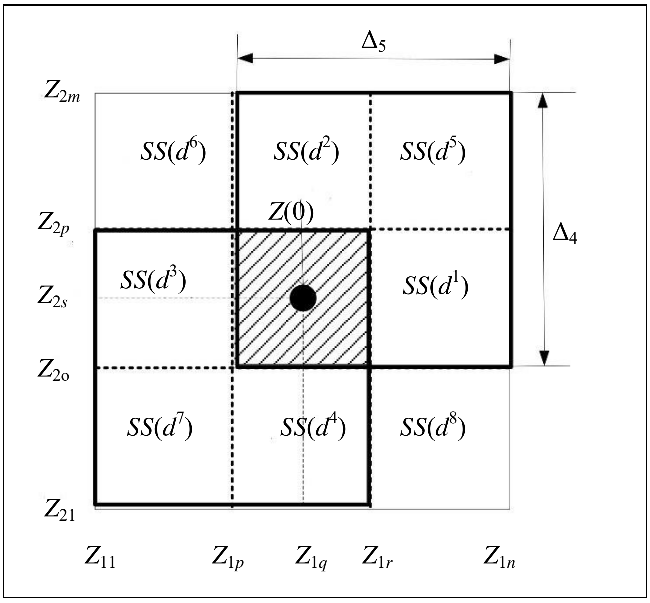


Fig. 4. The domains of generalized state classes by two features.

As shown in [16], the names d^H form a partially ordered set $\{d^H\}$ of the names of state classes $CF = (\{d^H\}, \leq)$ by the nesting $SS(d^H)$ of the state domains. This set is called a qualitative *conceptual framework*, which defines a qualitative ontology of the subject domain with the syntactic model (2).

A Hasse diagram in Fig. 5 defines a partial order of state class names. The first level of class names generalizes the basic class, whereas the second level generalizes the first-level names. Such a qualitative ontology determines a notional system of an ill-defined subject domain.

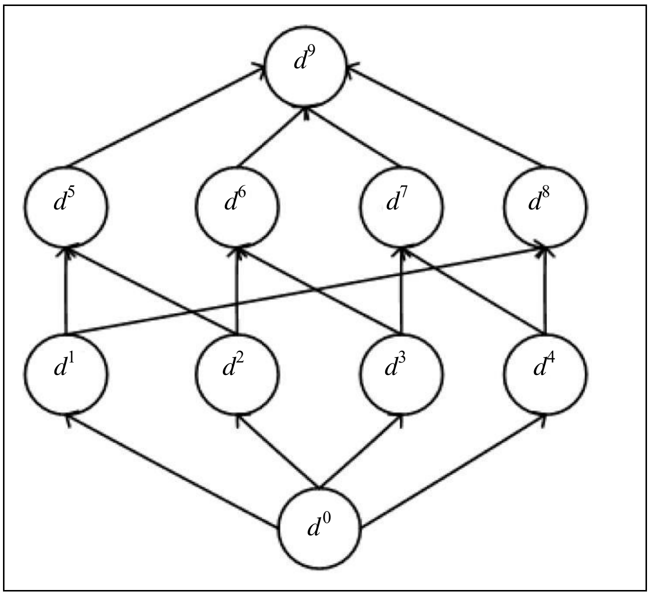


Fig. 5. The conceptual framework of a subject domain.

Thus, the semantics of the syntactic model (2) is defined by the qualitative conceptual framework

$$CF = (\{d^H\}, \leq), \tag{3}$$

where $d^H \Leftrightarrow SS(d^H)$ are the names of state classes that uniquely define the semantic space domains.

Note that at different time instants, the syntactic model states $Z(t)$ may belong to different domains $SS(d^H)$ and, therefore, have different names d^H, $Z(t) \in SS(d^H)$.

Also, we emphasize that the conceptual framework (3) is an idealized semantic model of the subject domain. Here, only the name d^0 of the basic notion (class) is explicitly defined: for all other state classes, we have artificial names d^H and the corresponding semantic domains $SS(d^H)$ they define.

Under uncertainty, when it is impossible to construct an ontology of the subject domain, the conceptual framework defines "reference points" of possible state classes. These points have the "class–subclass" relationship with the basic class. The real (not artificial) names of state classes correspond to the solutions of the inverse problem. We will find them by unstructured data processing (free text processing) methods on the Internet.

The pragmatic model. In semiotics, pragmatics is responsible for the relations between signs and their users. The pragmatic model assesses the utility of the system state for the DM. Assessment is based on determining the values of the expert's preference coefficients α_j with respect to the parameter values in the state vector ($Z(t)$).

The state estimate $O(Z(t))$ is given by the linear convolution

$$O(Z(t)) = \sum_j \alpha_j x_j(t),\ j = 1,..., n, \quad (4)$$

$$\sum_j a_j = 1,$$

where $x_j(t) \in [0, 1]$ is a mapping φ of the linguistic parameter values $z_j(t)$ on a segment of the numerical axis [0, 1], i.e., φ: $z_j(t) \rightarrow x_j(t) \in [0, 1]$, $z_j(t) \in Z(t)$, $O(Z(t)) \in [0, 1]$.

The three models have common parameters; therefore, a change in the state of one model will cause a change in the states of the others.

The general statement of the decision problem in the semiotic system is as follows. For the semiotic description (2)–(4) of a complex system, it is necessary to find a new description in the syntactic and semantic models with a better pragmatic assessment O^* compared to the existing one O. A solution method for this problem was proposed in [20]. The method is based on solving the inverse problem in the semiotic model by defining a target in the pragmatic model, solving the inverse problem in this model, and sequentially transferring the solution results from the pragmatic model to the syntactic model and then to the semantic model.

The inverse problem in the syntactic model is solved using methods for solving such problems for systems with qualitatively defined or fuzzy parameters. The methods were considered in [21, 22].

3. THE METHOD FOR SOLVING THE INVERSE PROBLEM

Consider the general algorithm for solving *the inverse problem*. The relation $W = |w_{ij}|_{n \times n}$ between the situation features and the target vector $G = (g_1, g_2,..., g_n)$ of their values are given. The problem is to find the set of input control vectors $\Omega = \{U_k\}$ such that $U_k \circ W = G\ \forall k$ for all $U_k = (u_{k1}, u_{k2},..., u_{kn})$. This problem is solved in a numerical system, i.e., the elements of the relation W, the target vector $g_i \in G$ and the vector of control actions $u_{ij} \in U$ are defined as real numbers: $w_{ij} \in$ R, $g_i \in [0, 1]$, and $u_{ki} \in [0, 1]$.

Recall that the situation is forecasted using the max-product composition. Therefore, we consider algorithms for solving inverse problems for the max-product composition rule. In this case, it is required to find an inverse mapping for the max-product composition.

An iterative algorithm developed in [22] yields a set of solutions of the inverse problem in the form $\Omega = \{U_{\max}, U_{\min}\}$: one maximum solution $U_{\max}$ and a set of minimum solutions $U_{\min} = \{U_1, U_2,..., U_q\}$, where $U_{\max}, U_1,..., U_q$ are the value vectors of the situation parameters. The iterative algorithm includes the following steps.

1. Find the vector $U_{\max} = (u_{1\max},..., u_{n\max})$ of the maximum solution $U_{\max} = \min(W\alpha' G^{\mathrm{T}})$, where

$$w_{ij}\,\alpha\, g_j = \begin{cases} 1 \text{ if } g_j \geq w_{ij}, \\ \dfrac{g_j}{w_{ij}} \text{ otherwise.} \end{cases}$$

The ith element of the vector $U_{\max}$ is given by $u_{i\max} = \min_{r=1}^{n} w_{ir}\alpha\, g_r$. When determining the vector $U_{\max}$ using the max-product composition, conventional matrix multiplication is replaced by the operation α and conventional matrix summation by the minimum operation.

2. Find the set of minimum solutions

$$U_{\min} = \{ \max \Phi[((W\beta G^{\mathrm{T}})\gamma(U_{\max})^{\mathrm{T}})]\}.$$

2.1. Here, the operation β is defined as follows:

$$U_\beta = w_{ij}\,\beta\, g_j = \begin{cases} 0 \text{ if } g_j > w_{ij} \text{ or } g_j = w_{ij} = 0, \\ \dfrac{g_j}{w_{ij}} \text{ otherwise.} \end{cases}$$

The ith element of the row vector $U_\beta = (u_{\beta 1},..., u_{\beta n})$ is given by $u_{\beta i} = \max_{r=1}^{n} w_{ir}\alpha\, g_r$.

2.2. The operation γ for the matrices U_β and $U_{\max}^{\mathrm{T}}$ is defined as follows:

$$U_\gamma = u_{\beta i}\,\gamma\, u_{i\max} = \begin{cases} 0 \text{ if } u_{i\max} \neq u_\beta, \\ u_{i\max} \text{ if } u_{i\max} = u_\beta. \end{cases}$$

The elements of the matrix $U_\gamma = |u_{\gamma ij}|_{n\times n}$ are given by

$$u_{\gamma ij} = (u_{i\,max}\,\gamma\, u_{\beta j})_{\substack{j=1,...,n,\\ i=1,...,n}}.$$

2.3. The operation $\Phi(U_\gamma)$ forms a set of matrices $\Phi(U_\gamma) = \{\phi(U_{\gamma k})\}$ from U_γ by the following rule:

- Each column of the matrices $\phi(U_{\gamma k})$ contains only one non-zero element, and all other elements are zeroed. Hence, the sum of elements in each column of the matrix $\phi(U_{\gamma k})$ equals its non-zero element.
- Any matrix $\phi(U_{\gamma k}) \in \Phi(U_\gamma)$ contains a unique combination of nonzero column elements of the original matrix $U_{\gamma k}$.

2.4. The operation $\max(\Phi(U_\gamma))$ forms all minimum solutions of the inverse problem by taking the maximum over the rows of each matrix $\phi(U_{\gamma k}) = |u_{\gamma kij}|$. The minimum solutions are given by $U_k = \max\limits_{i=1,\ldots,n} \phi(U_{\gamma k})$, $U_k = (u_{\gamma k1},\ldots, u_{\gamma kn})$. Thus, the algorithm generates the column vectors $(U_1,\ldots, U_k)$ of the solutions.

Applying the inverse mapping φ^{-1} to the elements of the solution vectors, we obtain the vector of linguistic values for the solution of the inverse problem, i.e., $Z^*_k = \varphi^{-1}(U_k) = (z_{k1},\ldots, z_{kn})\ \forall z_{ki} \in Z_i$.

All solutions of the inverse problem are represented as points in the structured semantic space CF(3). Then the solutions (the points in the semantic space) fall into different domains $SS(d^H)$ characterizing state classes with different names d^H. Thus, the formal solution of the inverse problem gives a set of names for solution classes, which are structured by the nesting of the domains $SS(d^H)$ corresponding to these names in the form of a qualitative ontology (the conceptual framework of solutions).

The formal names of solution classes are given by the mathematical symbols d^H. In the paper [20], the solution classes in a subject domain were named using methods based on classification [23] and categorization [24] processes studied by psychologists. In particular, a method for determining the compound name of a new class was proposed: This method supplements the name of the basic class d^0 with an estimate of a distinctive feature or features. We explain this method below.

The semantic model defines the basic class domain $SS(d^0) = \times_i \Delta_i$, where $\Delta_i = z_{i0} \pm \varepsilon_i$ is the tolerance interval for the ith feature, $z_{i0} \in Z_i$ is the feature value in the initial state, and $\pm\varepsilon_i$ specifies the tolerance interval limits for the feature f_i. The solution of the inverse problem is the vector $Z_k = (z_{k1},\ldots, z_{kn})$, $z_{ki} \in Z_i$. The solution of the inverse problem in the semantic model (3) will be written as the vector $A_k = (a_{k1},\ldots, a_{kn})$, $a_{ki} \in \{-1, 0, 1\}$, where $a_{ki} = -1$ if $u_{ki} < z_{i0} - \varepsilon_i$, $a_i = 1$ if $u_{ki} > z_{i0} + \varepsilon_i$, and $a_i = 0$ if $u_{ki} \in z_{i0} \pm \varepsilon_i$.

The component $a_{ki} \in \{1, 0, -1\}$ in the solution vector $A_k = (a_{k1},\ldots, a_{kn})$ qualitatively assesses the value of the ith parameter (f_i) in the inverse problem solution: if $a_{ki} = 1$ ($a_{ki} = -1$), the parameter has a large value (a small value, respectively). This vector can be represented as a vector with the linguistic assessments $L_k = (l_{k1},\ldots, l_{kn})$, where l_{ki} = "*Large*" if $a_{ki} = 1$, and l_{ki} = "*Small*" if $a_{ki} = -1$. Then the solution class has the compound artificial name

$$d_k^H = d^0 \mathop{\&}_{i\,(a_{ki}\neq 0)} l_{ki}.$$

For example, consider a basic class with the name d^0 = "*Inflation.*" For this class, possible compound names of new classes by the feature "*Inflation rate*" are d_k^1 = "*Inflation high*" or d_k^2 = "*Inflation low.*"

For each solution from the set $Z_k\ \forall k$, this compound artificial naming method gives a solution expressed in a restricted natural language: the name of the basic solution class d^0 and the assessments (l_{ki}) of the feature values differing from the feature values of the basic class.

Psychology suggests another naming method for solution classes based on the psychological theory of prototypes (categories) [24]. In this theory, a prototypical name is determined by the name of the most characteristic and frequently used name of a representative of this class. Note that a prototype has a sociocultural context. This means that a prototypical name may have different meanings in different social or cultural communities. The meaning in G. Frege's definition [3] is information about an object (in other words, the set of its properties).

In the case under consideration, the prototypical solution class with the name d_k^H is the words (sign symbols) often used with the compound name of the solution class d_k^H in the context of the subject domain.

Semiotic models have a peculiarity: under uncertainty and incomplete knowledge, they represent the set of alternative syntactic models of the situation as a partially ordered set containing the names of state classes of the semantic model (the conceptual framework of the subject domain).

Subjective qualitative semiotic models suffer from the following drawbacks: they are difficult to verify, and the subjective interpretations of modeling results are multiple. The problem essence and some remedies were described in [25, 26]. In this paper, we apply natural language processing methods for a relevant text corpus from the Internet to support the interpretation of solutions in the subject domain.

4. SITUATION MONITORING

Monitoring refers to the process of observing and recording the state of some object or situation. In the case of a technical object with measurable parameters in numerical scales, the monitoring process can be often implemented without difficulties. The parameters of social and political situations are represented in linguistic scales, which reflect opinions, points of view, and the behavior of social groups or individual politicians. For such situations, the program implementation of monitoring is much more complicated.

Comprehensive linguistic analysis (*Knowledge Acquisition*) involving the morphological, syntactic, and semantic analysis of the text yields a semantic network of subject domain concepts. Due to the theoretical difficulties of natural language processing, knowledge acquisition is an unreasonable approach to determining the state of the situation.

The situation monitoring subsystem can be treated as an interpreter of the input language in the semiotic architecture of a decision support system [13].

Figure 6 shows the generalized structure of a situation monitoring subsystem with textual information processing technologies based on the subjective semiotic model.

It includes two main subsystems:

– a subsystem for acquiring and processing unstructured data from the Internet,

– a subsystem of the semiotic situation model.

Situation monitoring is based on the subjective semiotic situation model (2)–(4). The parameters of this model are used as parameters in the information retrieval subsystem for obtaining unstructured data about the current state of the situation from the Internet. The current state means the state of the situation at the observation instant, $Z(t)$. The current state may differ from the initial one $Z(0)$. Unstructured data from news lines are analyzed.

In this subsystem, the situation parameter names of the semiotic model, $f_i \in F$, and their linguistic values Z_i are used for constructing a pattern base $\langle\{f_i\}, \{Z_i\}\rangle$. The situation parameter names $f_i \in F$ and the basic class name d^0 are used when forming a query for the Information Retrieval subsystem. The textual information about the parameter values (the current state of the situation) is extracted from the retrieved information.

To extract information from text, we will apply the technology presented in [27–29]. This technology is oriented to the semiotic model of the situation and uses the following parameters of the syntactic model: $\{f_i\}$ (the set of parameter names) and $\{Z_i\}$ (the set of their possible values). The method constructs patterns on a text corpus of a subject domain. During the pattern construction procedure, a reference element is defined in the sentences of the subject domain (the name of the semiotic system parameter f_i. In the sentence to the left or right of the reference element, an expert uses a text markup program to determine the text values of this parameter.

For example, for the parameter "*Social tension*," possible text values are as follows: "*Single picket*," "*Mass rally*," etc. An expert assigns possible linguistic assessments to these text values: "*Very much grows*," "*Strongly grows*," "*Grows*," "*Slightly grows*," "*Does not change*," "*Slightly decreases*," "*Decreases*," "*Strongly decreases*," and "*Very much decreases*." For example, the text value "*Single picket*" in the current state may be assessed as "*Weakly grows*" whereas the text value "*Mass rally*" as "*Strongly grows*." In this technology, the value scale of each parameter Z_i also has a linguistic assessment scale (see the previously listed assessments). Any textual assessment identified

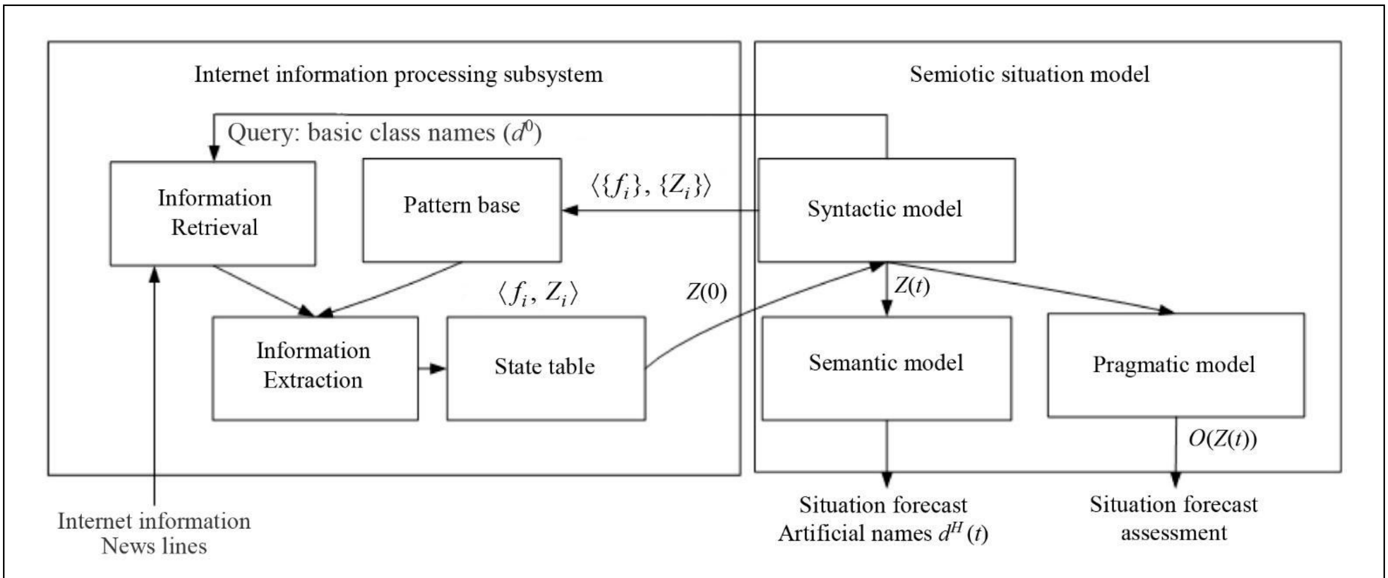


Fig. 6. The generalized structure of the situation monitoring subsystem.

by the expert in a text corpus is given the linguistic value of the parameter scale Z_i if their assessments coincide. Thus, a text pattern is formed: it consists of the parameter name, the text value in the text corpus, and the corresponding linguistic value of the parameter scale.

The quality of the information extraction method depends on the number of constructed patterns. But constructing each pattern requires much expert work. Each text pattern is unique and can be identified only in a particular text. To reduce the amount of expert work and improve the quality of identifying factographic information, the authors proposed an intelligent thesaurus revealing the synonyms of possible text values of the parameters.

The final pattern is formed through the conceptual assembly of a text pattern considering the synonyms of text values and the concepts of the subject domain ontology (the "general–particular" relationship). The pattern base contains conceptual patterns, which are employed to identify classes of similar factographic information considering synonymy and the "part–whole" relationships in the ontology.

Thus, the information extraction subsystem produces the situation parameter vector by extracting the linguistic parameter values: $Z^* = (z^*_{1h},..., z^*_{nq})$, $z^*_{ij} \in Z_i$. This vector is normalized, φ: $Z^* \to X^*$, $X^* = (x^*_{1h},..., x^*_{nq})$, $x^*_{ij} \in [0, 1]$, and then passed to the situation state table.

If the newly obtained state Z^* differs from the current one $Z(0)$, the semiotic model will forecast the situation. In the syntactic model, the forecast is the value vector $Z(n)$; in the semantic model, the state class name ($d^H \in CF$); in the pragmatic model, the new state assessment $O(Z^*)$.

The main elements of the monitoring subsystem are two technologies: Information Retrieval and Information Extraction. These technologies are studied by many researchers and engineers; different methods and algorithms were proposed for their implementation. With the quality estimations of these technologies available in the literature, we can understand and assess the effectiveness of the proposed monitoring approach based on the semiotic model.

The information retrieval technology is described as a set of Internet search services to get information from the Internet based on queries. Queries include parameters of the semiotic model (the parameter names and basic class names).

The quality of this technology (the completeness and relevance of the retrieved information) is provided by the developers of the corresponding services, and the results can be used due to libraries for different programming languages. Hence, the technology is applicable to end-user software development for specific tasks.

This technology extracts information from the text with a pattern contained in the pattern base; patterns include the semiotic model parameters. This quality of this subsystem is satisfactory for structured data with an explicitly identifiable pattern.

However, in the case of unstructured data (no pattern), this system works only after solving specific linguistic tasks: defining named entities, settling the coreference referential identity, and constructing relationships and scenarios. All these tasks are complex: for example, even an approximate solution of the coreference referential identity problem is possible only in some subject domains with an available knowledge base [29]. According to the presentations at the Message Understanding Conference (MUC-6, 1995), the best solutions of the coreference referential identity problem reached 59% of completeness under 72% of accuracy. Human performance in this case was estimated at 80% [30]. These figures are considered some quality limit of Information Extraction when analyzing unstructured data, reflecting the natural language properties. Further quality improvement of this subsystem for unstructured data incurs considerable costs [30].

Methods for extracting generic relations from text to supplement the taxonomies, thesauri, and ontologies of subject domains are of interest. Several international conferences and competitions [31–33] were organized on hyperonym extraction algorithms for the automatic or automated enrichment of the existing taxonomies of English and other Western European languages.

At the Dialogue 2020 conference (Moscow, 2020) the task was set to extract hyperonyms for the automated enrichment of RuWordNet, a Russian-language thesaurus [34]. The task was to find hyperonyms for a target word (noun or verb) based on text corpus analysis [35]. The developers proposed combined methods with calculating the co-occurrence vectors of the target word and the set of words from the text corpus [35]. The set of candidate hyperonyms from the co-occurrence vector is selected using different techniques (word weighting based on heuristics, closeness estimation for the text corpus word vectors and the vectors of known taxonomies and thesauri marked manually).

Different dictionaries (e.g., Wiktionary), lexical templates, and pre-trained multilingual neural networks (R-BERT) [36] are used to extract hyperonyms as well.

Nowadays, there are many commercial systems implementing Information Extraction. Most of them preliminarily prepare and structure text corpora and extract numerical information about the values of some parameters.

We note GATE (General Architecture for Text Engineering), a modular natural language processing system developed at the Department of Computer Science, the University of Sheffield [37]. ANNIE (ANearly-NewIEsystem) [38], an information extraction system, was developed based on GATE's architecture.

Presently, it seems reasonable to apply the information extraction technology together with the subjective semiotic model in situation monitoring to preliminarily structure text corpora and construct patterns for identifying situation dynamics. This approach eliminates a considerable part of the routine work of the analyst in situation monitoring.

5. DECISION-MAKING IN THE SEMIOTIC SYSTEM

The monitoring subsystem assesses the situation forecast, outputting $O(Z(n))$. If $O(Z(n))$ is worse than the current assessment $O(Z(0))$, the decision-making problem arises. In [20], decision-making was reduced to solving the inverse problem in the semiotic model. In this case, the target vector $G = (g_1, g_2,..., g_n)$ is set (Section 3). Its elements contain parameter values that will improve the pragmatic assessment of the situation from the expert's point of view, i.e., $O(G) > O(Z(n))$. Solving the inverse problem with a given target in the syntactic model yields the set of solutions U_{max}, U_1,..., U_k. These solutions are the control actions (solution alternatives) for achieving the target and, consequently, improving the pragmatic assessment. These solutions are described in the semantic model by the compound names of the solution classes d_k^H. The compound names are represented in the internal formal language of the expert (the developer of the subjective expert model). Under uncertainty, the expert's reasoning, justification, and choice of an acceptable solution using the subjective model are possible within his knowledge, which may be incomplete and contradictory. In this case, an external knowledge base is needed to support the expert's work, e.g., unstructured data (free text) from the Internet.

The decision-making subsystem based on the semiotic model is intended to find the interpretations of artificial compound names of solution classes on the Internet and explain them.

The generalized structure of the decision-making subsystem with textual information processing technologies based on the subjective semiotic model is shown in Fig. 7.

The decision-making system includes the following main subsystems:

– the semiotic model subsystem,
– the unstructured Internet data processing subsystem,
– the alternative solution subsystem (generation and explanation).

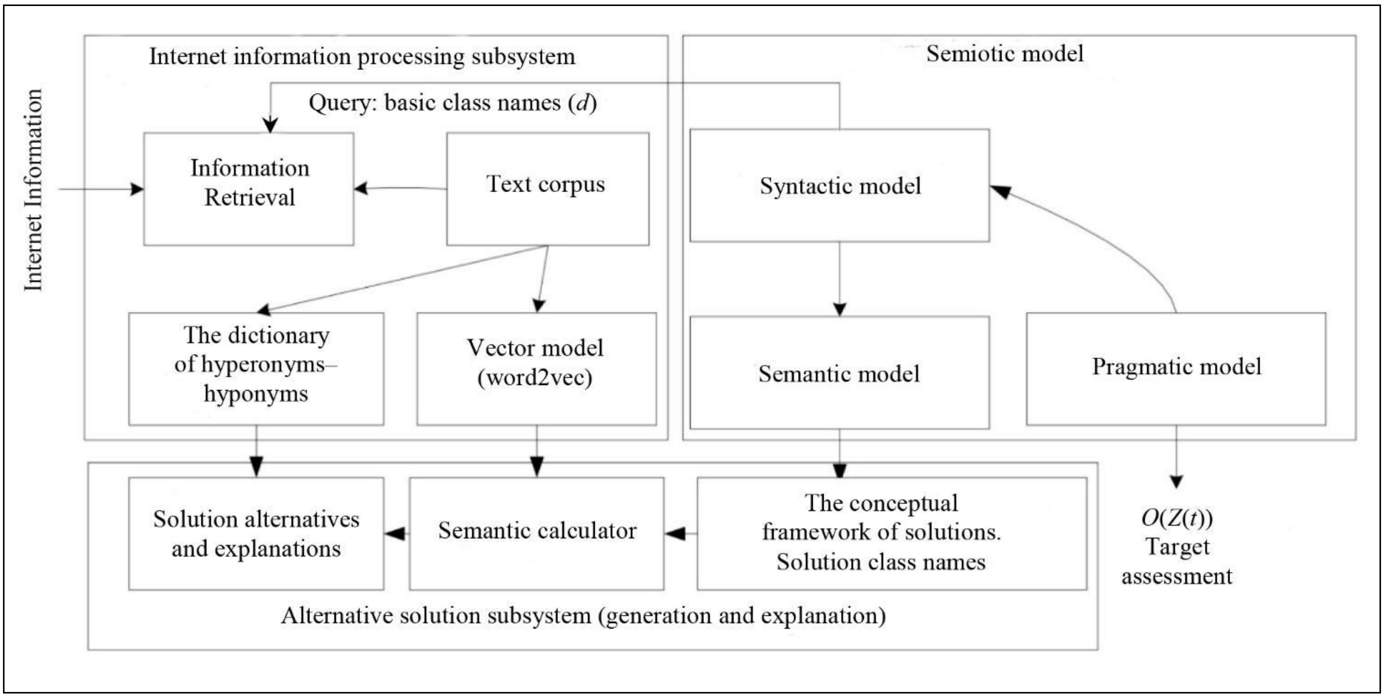


Fig. 7. The generalized structure of the decision-making subsystem.

5.1. The Unstructured Internet Data Processing Subsystem

This subsystem includes the following blocks: information retrieval on the Internet; text corpus; the dictionary of "hyperonyms–hyponyms"; the vector model.

Text corpus. The information retrieval block is to find as much information relevant to the query as possible. An information retrieval query on the Internet includes the names of the basic concepts of the semiotic model. As a result, we obtain a text corpus relevant to the name of the basic concept of the semiotic model. However, the text corpus needs normalization considering the syntactic relationships of the generic relations to build the word2vec vector model [39]. For this purpose, a syntactic window separates the nouns in the text corpus and reduces them to the normal form (nominative case, singular).

The vector model. The vector model of a text corpus is based on the distributive analysis of texts. According to [40], distributive analysis is a method to study languages depending on the environment (distribution) of individual linguistic units in the text without information about their lexical or grammatical meaning. In distributive analysis, each word (word combination) in some text is represented as a vector of words used jointly with this word in a given context. Each pair of words in this vector is characterized by the frequency of their co-occurrence in this context. Under the hypothesis formulated in [41], the linguistic units occurring in similar contexts have the close vectors of jointly used words.

Formally, this technology can be represented as follows. Consider a given text corpus, i.e., a set of sentences characterizing a subject domain. In distributive analysis, syntactic relationships in sentences are ignored. The subject domain is characterized by the word set of all sentences, $Tp = \{v_{gh}\}$, where g is the sentence number and h is the word number in the sentence (text corpus). The words without repetitions are defined on the set of all words as a word subset $V \subseteq Tp$, called a subject domain dictionary. It has the form $V = \{v_r\}$, where $r = 1,..., q$ are the word numbers in the dictionary. The joint usage of dictionary words in a given context is defined as the relation

$$R_{w2v}: V \times V \rightarrow r_{ij},$$

where $r_{ij} \in [0, 1]$ characterizes the co-occurrence of the words v_i and v_j in the subject domain under consideration.

For each dictionary word $v_r \in V$, the vector $R_p = (v_1/r_{p1},..., v_q/r_{pq})$, $p = 1,..., q$, characterizes its co-occurrence r_{pi} with other words $(v_1,..., v_q)$ of the considered subject domain (the so-called context vector). A slash in the context vector separates a dictionary word and its co-occurrence with other words.

Currently, the mapping R_{w2v} is constructed by the machine learning of an artificial neural network [39] in which the training sample is a text corpus Tp.

The word2vec technology introduces operations with word vectors to define new vectors determining the joint usage frequency of individual words from a dictionary V with other words of a subject domain. Operations in the word2vec technology can be represented as a mapping

$$\text{w2v}:((\bullet)(v_1,..., v_p)) \rightarrow R^*_w,$$

where the resulting vector R^*_w characterizes the joint usage frequency of words $(v_1,..., v_n)$ with other words of a subject domain.

Here $(\bullet)$ are the operations defined in word2vec as positive() and negative(). The resulting vector R^*_w of the positive() operation characterizes the joint usage frequency of the argument-specified words with other words of a subject domain. The resulting vector R^*_w of the negative() operation characterizes the frequency of subject domain words that are not used with the argument-specified words.

The operation of such vector models can be illustrated by Google's browser. When typing a word in the search line, the system shows a word vector frequently used with this word; adding one more word, we get another hint (a word vector frequently used with two words), etc. All typed words are arguments of the positive() operation. The word2vec technology allows searching for words that are not used jointly (the argument of the negative() operation). This operation also outputs a word vector.

Context vectors in this technology include words and their co-occurrence with other words of a subject domain. In this paper, words are the names of signs (G. Frege [3]) that denote a real object or situation and determine its properties (the meaning of the object or situation). In G. Frege's definition, the meaning is information about an object (i.e., a set of its properties in the word usage context). Given a word denoting an object, we can determine its properties (meaning).

In other words, the word2vec technology defines the function

$$\text{w2v}(\text{positive}(v_t,..., v_s); \text{negative}(v_q,..., v_n)) = R^*_w, \quad (5)$$

where $(v_t,..., v_s) \in$ V are the argument-specified words of the positive() operation and $(v_q,..., v_n) \in$ V are the argument-specified words of the negative() operation.

Recall that the name of a solution class is a prototype name, which is determined by the frequency of use to denote a category. The context vector contains information about the co-occurrence of words. Therefore, we assume that words with large co-occurrence values with the argument-specified words of function (5) can be prototype names.

The dictionary of hyperonyms-hyponyms. This dictionary reflects the generic relations extracted from a text corpus of a subject domain. They are extracted using lexico-syntactic patterns [42]. A lexico-syntactic pattern is a structural pattern of a linguistic construction reflecting its lexical and surface-syntactic properties. In the general case, a pattern defines a sequence of linguistic construction elements and sets grammatical agreement conditions for them.

In the scientific literature, there are many works devoted to extracting generic relations from English and Russian texts, assessing the identification quality of relations, and constructing and debugging patterns and their applications. According to the authors, the patterns proposed in [43] allow extracting 78.5% of generic relations contained in a text.

The patterns [43] were adopted to develop algorithms for extracting generic relations from a typical text of a subject domain. The morphological characteristics of the words in the analyzed sentences were used to develop the algorithms implementing the patterns. In particular, hyperonyms and hyponyms were considered to be nouns with the same animate characteristic. The patterns involved the agreement rule of case endings for the hyperonyms and hyponyms in a sentence.

The rules based on morphological analysis can improve the quality of extracting generic relations in sentences.

In distributive analysis, the words on the left and right of a given word are equivalent since the syntactic relationships between them are excluded. To eliminate this drawback of distributive analysis, it was proposed to use separate dictionaries for model words and contexts [44, 45]. We include the context explaining the generic relations in the dictionary of hyperonyms–hyponyms.

Lexico-syntactic patterns allow forming the dictionary of hyperonyms–hyponyms of a subject domain and the context dictionary associated with these words [46]. In the cited work, the context dictionary was extended to the explanatory dictionary, which contains sentences with generic relations extracted using patterns. Adding the context to the dictionary of hyperonyms–Hyponyms helps to select a hyperonym as a prototype, focusing on its usage context in the subject domain.

This is how the dictionary of hyperonyms–hyponyms and sentences describing generic relations is formed. Thus, the corresponding structure is described by the triplet

$$\langle \text{HYPER}, \text{HYPO}, \text{Context} \rangle. \qquad (6)$$

HYPER and HYPO are the sets of hyponyms and hyperonyms, respectively, of a subject domain reduced to the normal form; Context is the set of sentences with defined generic relations.

Note that the generic relations included in the dictionary reflect the structure of knowledge about the subject domain. As a matter of fact, they are elements of the subject domain ontology. In the semantic model, we have introduced a qualitative ontology of an ill-defined subject domain as an idealized conceptual structure with artificial names (the conceptual framework (3)). The identified hyperonyms can replace the artificial names of the state class of the conceptual framework, and hyponyms can serve as the name of this class. By extracting generic relations, we try to identify possible names of the state classes of the idealized conceptual framework obtained from the text corpus of the subject domain.

5.2. The Alternative Solution Subsystem (Generation and Explanation)

The subsystem for generating solution alternatives and their explanations includes a conceptual framework of solutions, a semantic calculator, and a block for identifying and explaining alternatives for solution class names.

The conceptual framework of solutions. The formal solution of the inverse problem in the semiotic model gives a set of names of solution classes structured as a qualitative ontology (a conceptual framework of solutions). These solutions are sign symbols with names and content and are expressed in an internal language of the semiotic model. They must be interpreted in the subject domain under consideration.

The solution of the inverse problem in the semantic model (3) is written as the vector $A_k = (a_{k1},..., a_{kn})$, where $a_{ki} \in \{1, 0, –1\}$, where a_{ki} qualitatively assesses the value of the ith parameter (f_i) in the solution: a_{ki} = 1 if the parameter has a large value and a_{ki} = –1 otherwise.

For example, the solution vector (1, 0, 0) means that the value of the first feature in the inverse problem solution is significantly larger than its counterpart in the basic notion d^0, whereas the other two features remain the same. This vector defines a domain of the semantic space $SS(d_k^H)$ and correspondingly the name of a solution class. All solution class names (d_k^H) can be represented as a partially ordered set by the nestings of the semantic space domains they define, (d_k^H, ≤). As demonstrated above, all solution classes have compound names. These names will be employed below to search a text corpus for prototype names using a semantic calculator.

The semantic calculator. It is intended to determine the joint usage vector for words included in the compound name of the inverse problem solution class obtained in the semiotic model (on the one part) and the words of a text corpus of the subject domain (on the other part). The semantic calculator is based on the trained distributive model of the subject domain. Its operation is described by function (5), where the arguments are the artificial names of the inverse problem solution classes. The compound name of a solution contains the basic notion name and qualitative assessments of the dynamics of different parameters: "*Large*," "*Small*", or their synonyms.

In the semantic calculator, the inverse problem solution $A_k = (a_{k1},..., a_{kn})$ in the semantic model is written as

$$\text{w2v (positive}(d^0, f_t|a_{kt} = 1,..., f_s|a_{ks} = 1);$$
$$\text{negative}(f_q|a_{kq} = -1,..., f_n|fa_{kn} = -1)) = R^*_w,$$

where f_t and f_s are the names of the model parameters for which the element a_{kt} = 1 is included in the argument of the positive() operation, and f_q and f_n are the names of the model parameters for which the element a_{kq} = –1 is included in the argument of the negative() operation. The basic class name (d^0) is also added to the argument of the positive() operation.

The calculator yields the word vector R^*_w = (v_i/r_{i1},..., v_n/r_{in}), which orders the joint usage of words (r_{ij}), the model parameters determined in the inverse problem solution, and all words of a text corpus of the subject domain included in the dictionary $v_i \in V$. As stated above, words with a high frequency of occurrence can be regarded as name prototypes for a solution class.

Alternatives names of solution classes and their explanation. Solutions in the semiotic system are possible names of solution classes. The hyperonyms extracted from a text corpus can be the names of solution classes since they define the elements of the ontology of the subject domain. Therefore, we find the intersection of all hyperonyms from the word dictionary (6) in the solution vector R^*_w to obtain alternative solution classes. Let the possible names of solution classes be written as

$$\langle (V \cap \text{HYPER}); \text{Context}\rangle,$$

where the intersection of the set $V = \{v_i\} \in R^*_w$ and the set of hyperonyms from the dictionary (5) gives the set of solution class names, and Context (the sentence text) helps to choose the desired name.

6. AN EXAMPLE AND EXPERIMENTS

The proposed semiotic system was experimental tested for the decision support subsystem [46]. The semiotic model of a sociopolitical situation was developed. The following elements were defined in the syntactic model: "*Power*" (d_1^0), "*Population*" (d_2^0), "*Economy*" (d_3^0), and "*Oligarchs*" (d_4^0) as the basic notions; the features of these notions, $f_i \in F$; the possible values $Z_i \in Z$ of the features and a causal network *W*.

The basic notion "*Oligarchs*" was assigned the features "*The level of discontent*" and "*The level of patriotism*." For a given target $O(Z(t))$, when solving the inverse problem, the feature "*The level of discontent*" was increased for "*Oligarchs*." Thus, the new notion d_4^1 ("*Oligarchs*" with a high value of "*The level of discontent*") was obtained. In the semantic model, this solution is formally represented by the vector A_4 = (1, 0) and denoted by the artificial name d_4^1 = "*Discontented oligarchs*."

It is required to interpret this solution in the subject domain.

For this purpose, a program layout was developed in Python3.

WebScraper was developed to extract relevant information from the Internet and build the text corpus. Information from 150 URLs (sites) was read; in addition, the text corpus was supplemented with the book [47] devoted to Russia's oligarchs. The Google library, googlesearch, was used as a search engine with the following parameters: the name of the basic notion and the number of links to the retrieved web pages. The syntactic analysis of the html code of the retrieved web pages (parsing) was carried out using BeautifulSoup, a Python3 library.

Lexico-syntactic patterns were developed and debugged to build the dictionary of hyperonyms–hyponyms. When constructing the patterns, the morphological analysis of the Russian text was performed using Pymorphy2 [48].

The vector model of the text corpus was obtained using word2vec. The word2vec model was trained with the following parameters: the training model—skipgram, the training window—5, training iterations––10, the aggregation method— softmax, the word occurrence threshold—3, and the word vector dimension—150.

The dictionary and vector model were stored in a SQLite-3 database.

SQL queries to the SQLite-3 database were developed to generate solution alternatives and their explanations. They return the names of solution classes with comments.

Example. To interpret the inverse problem solution d_4^1= "*Discontented oligarchs*", we normalized it and substituted the result in the semantic calculator:

$$\text{w2v(positive(“}\textit{Oligarchs},\text{” “}\textit{Discontent}\text{”))} = R^*_w.$$

The trained word2vec model yielded the word vector R^*_w reflecting the joint usage frequency of the words "*Oligarchs*" and "*Discontent*":

R^*_w = (*Harm/0.904; Fact/0.885; Respondent/0.873; Expert/0.872; Annexation/0.866; Regret/0.863; Position/0.852; Factor/0.844; Trend/0.833; Claim/0.830; Distrust/0.817; Effectiveness/0.813; Advantage/0.808; Character/0.806; Request/0.805; Reason/0.805;...*)

Then we obtained possible names of the solution classes by intersecting the word vector R^*_w with the hyperonyms of the subject domain:

($W \cap$ HYPER) = (*Harm/0.904; Fact/0.885; Regret/0.863; Position/0.852; Claim/0.830; Distrust/0.817; Character/0.806;...;*).

Clearly, in the word vector, "*Harm*" is the closest word to the inverse problem solution with the name "*Discontented oligarchs*": the co-occurrence is 0.904. Therefore, a possible new class of solutions is the one named "*Harmful oligarchs*."

The context of the candidate hyperonym was analyzed to select the solution class names through expertise. For example, the following context was found for the word "*Distrust*" with a co-occurrence of 0.817: "Between this category of business and the conditional "collective Putin" there has been a steady mutual distrust: the former has always feared the seizure of property, whereas the "collective Putin" has feared disloyalty."[1] ♦

Thus, the proposed semiotic architecture of the decision support system allows getting alternative names of the solution classes and choosing a solution based on relevant text analysis.

Note that the text corpus of the example included about 30 000 sentences describing different aspects of the subject domain. With the described approach, the possible names of solution classes were determined by analyzing a much smaller number of sentences. This illustrates the effectiveness of the method: the routine analytical work of an expert is reduced, and his intellectual productivity is improved accordingly.

[1] https://carnegie.ru/commentary/76115 (Accessed February 24, 2022.)

CONCLUSIONS

This paper has proposed a semiotic decision support system in complex dynamic systems under uncertainty. The support is based on extracting, processing, and structuring information from the Internet and a relevant semiotic model of the situation. This model includes three parametrically interconnected submodels: syntactic, semantic, and pragmatic. The inverse problem solution in the semiotic model has been represented as a qualitative ontology of solution classes (the conceptual framework of solutions). Methods for determining and interpreting solution class names extract relevant information from the Internet. Lexico-semantic patterns in a text corpus of a subject domain serve to define the dictionary of generic relations (hyperonyms–hyponyms) and the contextual dictionary. Distributive semantics methods (word2vec) have been applied to construct a semantic calculator. This calculator determines the meanings of the solution class names in the conceptual framework.

Experimental testing of the proposed architecture has shown its effectiveness. Further experimental research will aim at improving the quality of the proposed approach by increasing the volume of text corpora of a subject domain, using the free dictionaries of hyperonyms–hyponyms, and performing the additional semantic analysis of sentences containing solution vector words with a high co-occurrence with the compound name of the solution class.

REFERENCES

1. Pospelov, D.A., *Situatsionnoe upravlenie: teoriya i praktika* (Situation Management: Theory and Practice), Moscow: Nauka, 1986. (In Russian.)
2. Pospelov, D.A. and Osipov, G.S., Applied Semiotics, *Novosti Iskusstvennogo Intellekta*, 1999, no. 1, pp. 9–35. (In Russian.)
3. Frege, G., On Sense and Nominatum, *Readings in Philosophical Analysis*, 1949, pp. 85–102.
4. Osipov, G.S., Chudova, N.V., Panov, A.I., and Kuznetsova, Yu.M., *Znakovaya kartina mira sub"ekta povedeniya* (The Sign Picture of the World of the Behavioral Subject), Moscow: Fizmatlit, 2018. (In Russian.)
5. Ch.S. Peirce: What is Sign?, *Vestnik Tomsk. Gos. Univ. Filosofiya. Sotsiologiya. Politologiya*, 2009, no. 3(7), pp. 88–95. (In Russian.)
6. Morris, Ch.W., *Foundations of the Theory of Signs*, The University of Chicago Press, 1938.
7. Barron, T.M., Chiang Roger, H.L., and Storey, V.C., A Semiotics Framework for Information Systems Classification and Development, *Decision Support Systems*, 1999, no. 25, pp. 1–17.
8. Andersen, P.B., *A Theory of Computer Semiotics: Semiotic Approaches to Construction and Assessment of Computer Systems*, Cambridge University Press, 2006.
9. Goguen, J., An Introduction to Algebraic Semiotics, with Applications to User Interface Design, in *Computation for Meta-*

phor, Analogy and Agents, Nehaniv, Ch., Ed., Springer Lecture Notes in Artificial Intelligence, 1999, vol. 1562, pp. 242–291.
10. Furtado, A.L., Casanova, M.A., and Barbosa, S.D.J., A Semiotic Approach to Conceptual Modelling, in *ER 2014: Conceptual Modeling*, Yu, E., Dobbie, G., Jarke, M., and Purao, S., Eds., Lecture Notes in Computer Science, 2014, vol. 8824, pp. 1–12.
11. Massel, L.V. and Massel, A.G., Situation Management and Semantic Modeling in Power Engineering, *Proceedings of the 4th International Conference on "Open Semantic Technology for Intelligent Systems" (OSTIS-2014)*, Minsk, 2014, pp. 111–116.
12. Massel, L.V., Creation Problems of Intelligent Systems of Semiotic Type for Strategic Contingency Management in Critical Infrastructures, *Information and Mathematical Technologies in Science and Management*, 2016, no. 1, pp. 7–27. (In Russian.)
13. Ehrlikh, A.I., Applied Semiotics and Management of Complex Objects, *Software & Systems*, 1997, no. 3, pp. 3–11. (In Russian.)
14. Kulinich, A.A., Semiotic Approach in Modeling and Decision-making in Ill-defined Complex Situations, *2018 Eleventh International Conference "Management of Large-Scale System Development" (MLSD-2018)*, Moscow, October 1–3, 2018, Trapeznikov Institute of Control Sciences RAS, pp. 1–5. DOI: 10.1109/MLSD.2018.8551835.
15. Pospelov, D.A., *Logiko-lingvisticheskie modeli v sistemakh upravleniya* (Logic-Linguistic Models in Control Systems), Moscow: Energoizdat, 1981. (In Russian.)
16. Borisov, V.V., Kruglov, V.V., and Fedulov, A.S., *Nechetkie modeli i seti* (Fuzzy Models and Networks), 2nd ed., Moscow: Goryachaya Liniya-Telekom, 2012. (In Russian.)
17. Kosko, B., Fuzzy Cognitive Maps, *International Journal of Man-Machine Studies*, 1986, vol. 24, iss. 1, pp. 65–75.
18. Kulinich, A.A., Cognitive Maps Modelling Computer Systems (Approaches and Methods), *Control Sciences*, 2010, no. 3, pp. 2–16. (In Russian.)
19. Kulinich, A.A., Conceptual Frameworks of Ontologies for Ill-structured Problem Domains, *Scientific and Technical Information Processing,* 2015, vol. 42, pp. 411–419.
20. Kulinich, A.A., Solution of the Inverse Problem in Semiotic Systems under Uncertainty, *Trudy 9-oi Mezhdunarodnoi nauchno-prakticheskoi konferentsii "Integrirovannye modeli i myagkie vychisleniya v iskusstvennom intellekte"* (Proceedings of the 9th International Scientific and Practical Conference "Integrated Models and Soft Computing in Artificial Intelligence"), Kolomna, 2019, vol. 2, pp. 572–582. (In Russian.)
21. Pedrycz, W., Fuzzy Models and Relational Equations, *Math. Modeling*, 1987, no. 9, pp. 427–434.
22. Pedrycz, W., Algorithms for Solving Fuzzy Relational Equations in a Probabilistic Setting, *Fuzzy Sets and Systems*, 1990, no. 38, pp. 313–327.
23. Vygotskii, L.S., *Myshlenie i rech'* (Thinking and Speaking), Moscow: Pedagogika, 1982. (In Russian.)
24. Rosch, E., Principles of Categorization, in *Cognition and Categorization*, Rosch, E. and Lloyd, B.D., New York: Lawrence Erlbaum Association, 1978.
25. Abramova, N.A., Expert Verification in Formal Cognitive Map Application. Approaches and Practices, *Large-Scale Systems Control*, 2010, no. 30.1, pp. 371–410. (In Russian.)
26. Abramova, N.A., Interdisciplinary Approach to Verification in Decision-Making with Formal Methods, in *Handbook on Psychology of Decision-Making: New Research. Series: Psychology of Emotions, Motivations and Actions*, Hauppauge, USA: Nova Science Publishers, 2012, pp. 89–111.
27. Rubashkin, V.Sh., Ontology – Problems and Solutions. The Designer, *Proceedings of International Conference on Computational Linguistics and Intellectual Technologies (Dialogue-2007)*, Bekasovo, 2007, pp. 481–485. (In Russian.)
28. Pivovarova, L.M., Factographic Text Analysis in a Decision Support System, *Vestnik of St. Petersburg State University. Language and Literature*, 2010, no. 4, pp. 190–197. (In Russian.)
29. Rubashkin, V.Sh., *Predstavlenie i analiz smysla v intellektual'nykh informatsionnykh sistemakh* (Representation and Analysis of Meaning in Intelligent Information Systems), Moscow: Nauka, 1989. (In Russian.)
30. Grishman, R., Information Extraction: Techniques and Challenges, in *Information Extraction*, Pazienza, M.T., Ed., Lecture Notes in Artificial Intelligence, Rome: Springer-Verlag, 1997, vol. 1299, pp. 10–27.
31. Camacho-Collados, J., Delli Bovi, C., Espinosa-Anke, L., et al. SemEval-2018 Task 9: Hypernym Discovery, *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, Stroudsburg, 2018, pp. 712–724.
32. Jurgens, D. and Pilehvar, M.T., SemEval-2016 Task 14: Semantic Taxonomy Enrichment, *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, San Diego, California, 2016, pp. 1092–1102.
33. Bernier-Colborne, G. and Barriere, C., CRIM at SemEval-2018 Task 9: A Hybrid Approach to Hypernym Discovery, *Proceedings of the 12th International Workshop on Semantic Evaluation*, New Orleans, Louisiana, 2018, pp. 725–731.
34. Loukachevitch, N.V. and Dobrov, B.V., RuThes Linguistic Ontology vs. Russian Wordnets, *Proceedings of the 7th Global Wordnet Conference*, Tartu, 2014, pp. 154–162.
35. Nikishina, I., Logacheva, V., Panchenko, A., and Loukachevitch, N., RUSSE'2020: Findings of the First Taxonomy Enrichment Task for the Russian Language, *Proceedings of International Conference on Computational Linguistics and Intellectual Technologies (Dialogue 2020)*, Moscow, 2020, pp. 579–595. DOI: 10.28995/2075-7182-2020-19-579-595.
36. Tikhomirov, M.M., Loukachevitch, N.V., and Parkhomenko, E.A., Combined Approach to Hypernym Detection for Thesaurus Enrichment, *Proceedings of International Conference on Computational Linguistics and Intellectual Technologies (Dialogue 2020)*, Moscow, 2020, pp. 736–746. DOI: 10.28995/2075-7182-2020-19-736-746.
37. Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V., GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, 2002.
38. *GATE's ANNIE System*. URL: http://gate.ac.uk/ie/annie.html. (Accessed February 24, 2022.)
39. Mikolov, T., Sutskever, I., View, M., et al., Distributed Representations of Words and Phrases and Their Compositionality, in *Advances in Neural Information Processing Systems: the 27th Annual Conference on Neural Information Processing Systems*. Proceedings of a meeting held December 5–8, 2013, Lake Tahoe, Nevada, 2013, pp. 3111–3119.
40. *Lingvisticheskii entsiklopedicheskii slovar'* (The Linguistic Encyclopedic Dictionary), Yartsev, V.N, Ed., Moscow: Sovetskaya Entsiklopediya, 1990. (In Russian.)

41. Sahlgren, M., The Distributional Hypothesis, *Italian Journal of Linguistics*, 2008, vol. 20, no. 1, pp. 33–53.
42. Bol'shakova, E.I., Vasil'eva, N.E., and Morozov, S.S., Lexico-syntactic Patterns for Automatic Analysis of Scientific and Technical Texts, *Trudy 10-oi Natsional'noi konferentsii po iskusstvennomu intellektu (KII-2006)* (Proceedings of the 10th National Conference on Artificial Intelligence (AIC-2006)), Moscow: Fizmatlit, 2006, vol. 2, pp. 506–524 (In Russian.)
43. Sabirova, K. and Lukanin, A., Automatic Extraction of Hypernyms and Hyponyms from Russian Texts, *CEUR Workshop Proceedings*, 2014, vol. 1197, pp. 35–40.
44. Levy, O. and Goldberg, Y., Dependency-Based Word Embeddings, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Short Papers, Baltimore, Maryland, 2014, pp. 302–308.
45. Trofimov, I.V. and Suleymanova, E.A., A Dependency-based Distributional Semantic Model for Identifying Taxonomic Similarity, *Program Systems: Theory and Applications*, 2018, vol. 9, no. 4(39), pp. 443–460. (In Russian.)
46. Kulinich, A., Semiotic Models in Monitoring and Decision Support Systems, in *RCAI 2021: Artificial Intelligence*, Kovalev, S.M., Kuznetsov, S.O., and Panov, A.I., Eds., Lecture Notes in Computer Science, Cham: Springer, 2021, vol. 12948, pp. 303–316.
47. Hoffman, D.E., *The Oligarchs: Wealth and Power in the New Russia*, New York: PublicAffairs, 2011.
48. Korobov, M., Morphological Analyzer and Generator for Russian and Ukrainian Languages, *Analysis of Images, Social Networks and Texts*, 2015, pp. 320–332.

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

Received March 1, 2022, and revised November 1, 2022.
Accepted November 22, 2022.

Author information

Kulinich, Alexander Alexeevich. Cand. Sci. (Eng.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ alexkul@rambler.ru

Cite this paper

Kulinich, A.A., Application of Semiotic Models to Decision-Making. *Control Sciences* **5,** 34–50 (2022). http://doi.org/10.25728/cs.2022.5.4

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com

DOI: http://doi.org/10.25728/cs.2022.5.5

DIFFERENTIATION AND INTEGRATION IN FUNCTIONAL VOXEL MODELING

A.V. Tolok[1] and N.B. Tolok[2]

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia

[1]✉ tolok_61@mail.ru, [2]✉ nat_tolok@mail.ru

Abstract. This paper presents a simple method for generating the partial derivatives of a multidimensional function using functional voxel models (FV-models). The general principle of constructing, differentiating, and integrating an FV-model is considered for two-dimensional functions. Integration is understood as obtaining local geometrical characteristics for the antiderivative of a local function with solving the Cauchy problem when finally constructing the FV-model. The direct and inverse differentiation algorithm involves the basic properties of the local geometrical characteristics of functional voxel modeling and the inherent linear approximation principle of the codomain of the algebraic function. Simple computer calculations of this algorithm yield an FV-model suitable for any further algebraic operations. An illustrative example of constructing a functional voxel model of a complex two-dimensional algebraic function is provided. Functional voxel models of partial derivatives are obtained based on this model. These models and the boundary condition at a given point are used to obtain an initial FV-model of a complex algebraic function. The approach is applicable to algebraic functions defined on the domain of various dimensions.

Keywords: functional voxel model, local geometrical characteristics, local function, partial derivative, antiderivative.

INTRODUCTION

Differential calculus is still topical: it underlies almost all theoretical mechanics and mathematical physics as well as control theory. Nowadays, there exists a developed mathematical apparatus based on formal partial differentiation and derivation of integrand expressions to solve the inverse problem. Rather complex solutions are formed using the tables of known antiderivatives for various-type simple expressions and integration rules. Many attempts were undertaken to automate this process and obtain equations for further calculations [1–6]. In this case, the computer acts as a calculator without acquiring any "intelligent skills." The main problem is the inapplicability of such approaches to complex differentiable functions with peaks and discontinuities. Such functions arise in *R*-functional modeling and actively participate in the analytical modeling of geometric models to describe different objects and continuous processes. Among some examples, we mention a function describing a rectangular or polygonal zero contour, etc. For example, the following expression describes the positive domain of a rectangle with sides *a* and *b*:

$$a^2 + b^2 - x^2 - y^2 - \sqrt{\left(a^2 - x^2\right)^2 + \left(b^2 - y^2\right)^2} \geq 0. \quad (1)$$

Numerical methods based on discrete calculus have much contributed to automating the process of differentiation (the method of differences) and integration (the method of trapezoids, etc.). The problem grows sharply when increasing the dimensionality, especially with respect to the automation of expressions. In numerical methods, all arguments of a function become constants, except for the argument of differentiation, and the required order of the derivative is achieved by successive differentiation. However, numerical methods have an obvious disadvantage: their result is the value of the derivative at a point, not its algebraic function, which is required for solving the inverse integration problem [7–9].

Thus, the approaches discussed above cannot generally provide an automated solution of the direct and inverse differentiation problems.

We consider a developing computer method called functional voxel modeling (FVM). This method is intended for the discrete computer representation of continuous functions on a given multidimensional domain. It involves local geometrical characteristics (LGCs) on a given domain of an algebraic function. FVM was described in detail in [10, 11]. This method is based on the computer representation of the domain of local functions that replace the given domain of an algebraic function at each point. In contrast to numerical methods, the result at a point is not a numerical value but a function of simple linear form. For a two-dimensional complex algebraic function as one example, we consider the fundamental principle of obtaining the domain of local linear functions in a functional voxel computer model (FV-model) and the main differential operations performed to obtain the derivatives and the antiderivative.

1. THE FUNCTIONAL VOXEL MODELING OF AN ALGEBRAIC FUNCTION

As an illustrative example of obtaining an FV model, we consider the smooth function

$$u = x\sin\left(\pi\frac{y}{k}\right) + y^2\cos\left(\pi\frac{x}{k}\right), \quad (2)$$

on the domain [–1, 1] × [–1, 1] in the space xOy, where the coefficient k takes any value, e.g., 0.5.

This example of a continuous and smooth function provides a mathematical solution of partial derivatives for comparing FV-models.

We apply a regular rectangular grid with a cell spacing of 0.02 to the domain of the function. Let the nodes be numbered as in Fig. 1 to form a group of nodes of the triangular grid segment.

For the given coordinates of the three points, the determinant-based equation of the plane has the form

$$\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} = ax + by + cz + d = 0,$$

where

$$a = y_1(z_2 - z_3) - y_2(z_1 - z_3) + y_3(z_1 - z_2),$$

$$b = -(x_1(z_2 - z_3) - x_2(z_1 - z_3) + x_3(z_1 - z_2)),$$

$$c = x_1(y_2 - y_3) - x_2(y_1 - y_3) + x_3(y_1 - y_2),$$

$$d = -(x_1(y_2z_3 - y_3z_2) - x_2(y_1z_3 - y_3z_1) + x_3(y_1z_2 - y_2z_1)). \quad (3)$$

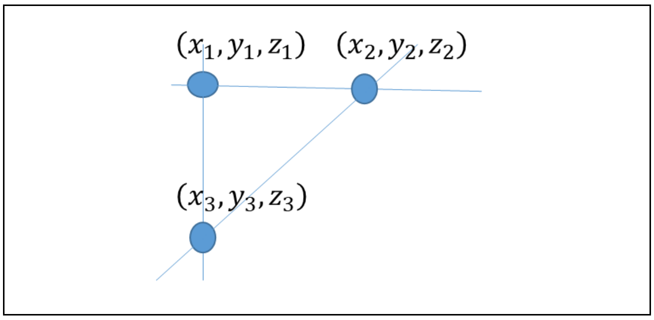


Fig. 1. The nodes of an approximation grid.

Next, we normalize the coefficients by the length of the four-dimensional gradient vector:

$$n_1 = \frac{a}{\sqrt{a^2+b^2+c^2+d^2}},$$

$$n_2 = \frac{b}{\sqrt{a^2+b^2+c^2+d^2}},$$

$$n_3 = \frac{c}{\sqrt{a^2+b^2+c^2+d^2}},$$

$$n_4 = \frac{d}{\sqrt{a^2+b^2+c^2+d^2}}.$$

Let the color gradation values of the monochrome palette P be associated with the values of the normal components (LGCs) as follows:

$$M_i = \frac{P(1+n_i)}{2}, \ \left(P = 256,\ i = \overline{1,4}\right).$$

Figure 2 shows the M-images (image-models) of the FV-model color mapping for the corresponding domain of the local geometrical characteristics of function (2).

At each point of the domain, this data representation allows automatically producing a local function that duplicates function (1) but has the simplest possible form:

$$n_1x + n_2y + n_3z + n_4 = 0. \quad (4)$$

To illustrate the next steps of differentiation, we model the M-images for the partial derivatives of function (2) expressed traditionally:

$$\frac{\partial u}{\partial x} = \sin\left(\pi\frac{y}{a}\right) - y^2\frac{\pi}{a}\sin\left(\pi\frac{x}{a}\right), \quad (5)$$

$$\frac{\partial u}{\partial y} = x\frac{\pi}{a}\cos\left(\pi\frac{y}{a}\right) + 2y\cos\left(\pi\frac{x}{a}\right). \quad (6)$$

Figures 3 and 4 demonstrate the FV-models for equations (5) and (6), respectively. Each M-image visualizes the changes in the local geometrical characteristics forming the local function for each point.

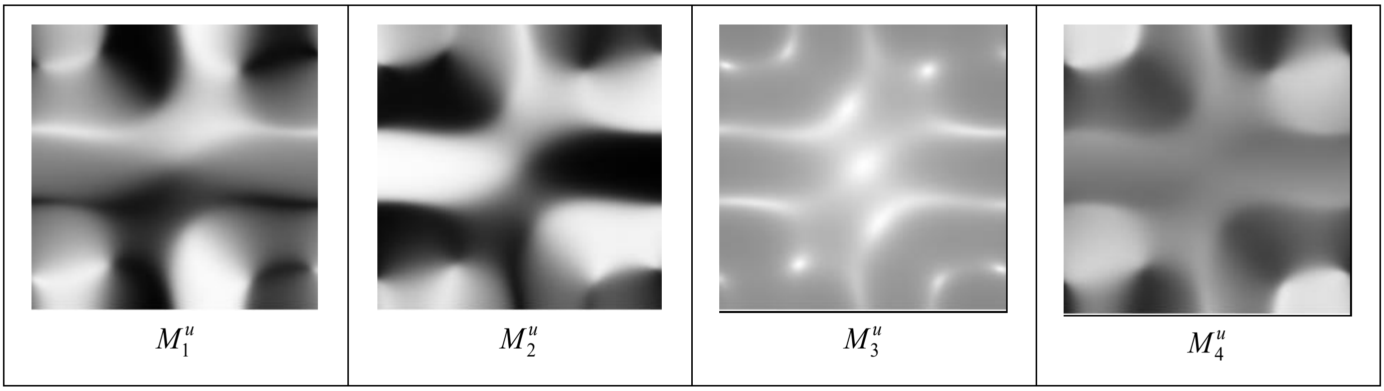


Fig. 2. The basic *M*-images of the function *u*.

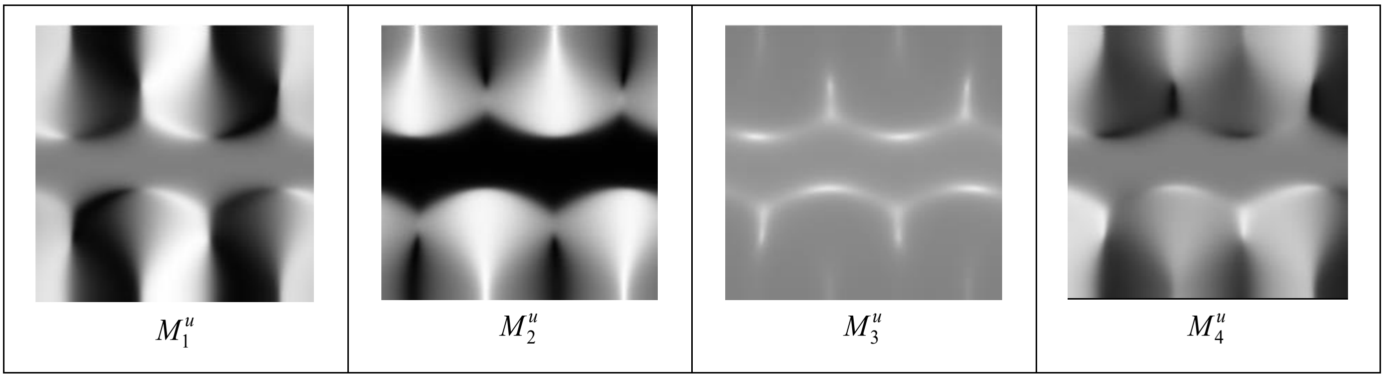


Fig. 3. The basic *M*-images of the function $\partial u / \partial x$.

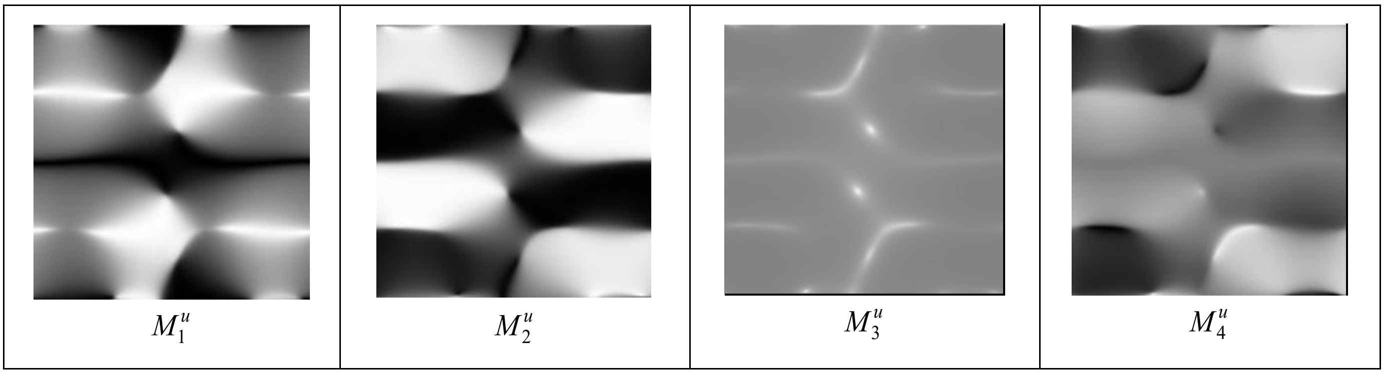


Fig. 4. The basic *M*-images of the function $\partial u / \partial y$.

2. PARTIAL DIFFERENTIATION OF FV-MODELS

We model the partial derivative along the axis Ox using the above algorithm and the relation

$$\frac{\partial u}{\partial x} = \frac{a}{c} = \frac{n_1}{n_3}, \tag{7}$$

where n_1 and n_3 are the coefficients of equation (4).

Having such values for each point of the domain with the *M*-images M_1^u and M_3^u (Fig. 2), we obtain a similar approximation scheme (Fig. 5).

For the three points, the determinant-based equation of the plane has the form

$$\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & \left(\dfrac{n_1}{n_3}\right)_1 & 1 \\ x_2 & y_2 & \left(\dfrac{n_1}{n_3}\right)_2 & 1 \\ x_3 & y_3 & \left(\dfrac{n_1}{n_3}\right)_3 & 1 \end{vmatrix} = ax + by + cz + d = 0.$$

Figure 6 shows the M-images of the color mapping for the corresponding domain of the local geometrical characteristics of (7). Obviously, these M-images visually coincide with the ones in Fig. 3.

By analogy, we can obtain M-images for the derivative along the axis Oy (Fig. 7), where

$$\frac{\partial u}{\partial y} = \frac{b}{c} = \frac{n_2}{n_3}. \tag{8}$$

Consider an example of modeling the second derivative along the axis Ox. For this purpose, we differentiate function (3) using its FV-model (Fig. 6) and the local differentiation algorithm proposed above.

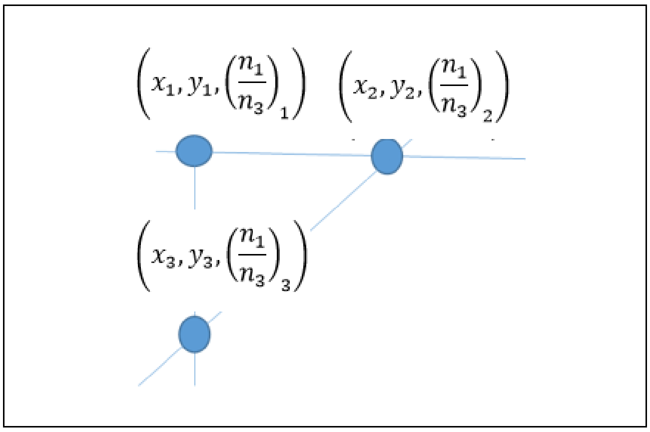


Fig. 5. The nodes of an approximation grid.

The resulting FV-models of the derivatives

$$\frac{\partial^2 u}{\partial x^2} \text{ and } \frac{\partial^2 u}{\partial x \partial y}$$

are presented in Fig. 8.

Thus, the FVM approach to differential images allows obtaining derivatives of different order without much difficulty.

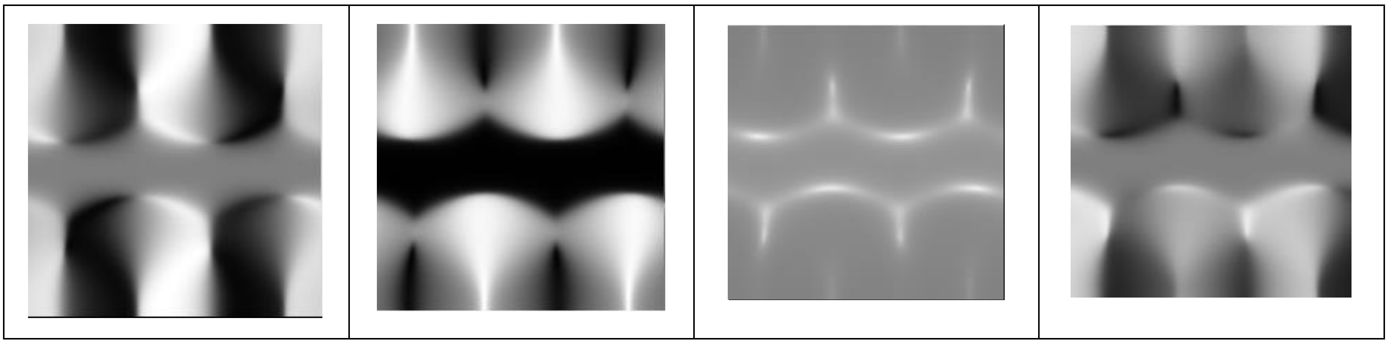


Fig. 6. The basic *M*-images of the function n_1 / n_3.

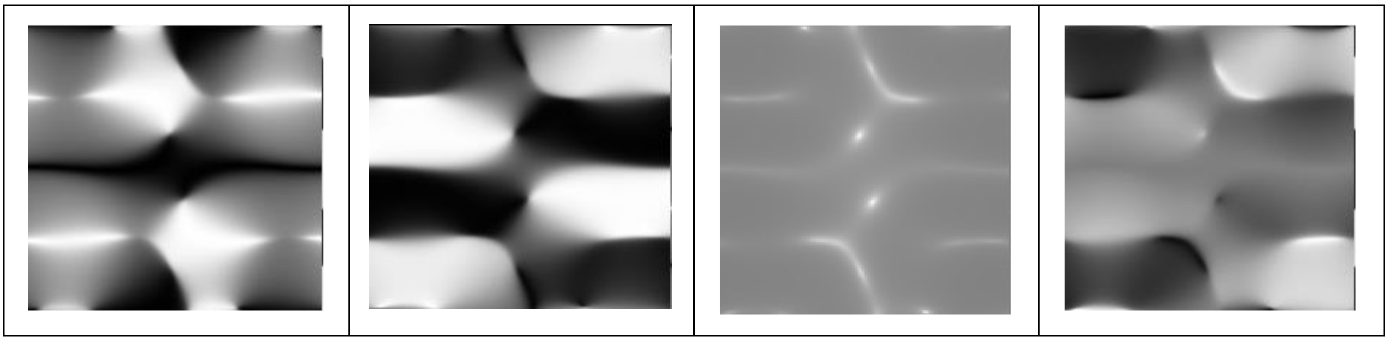
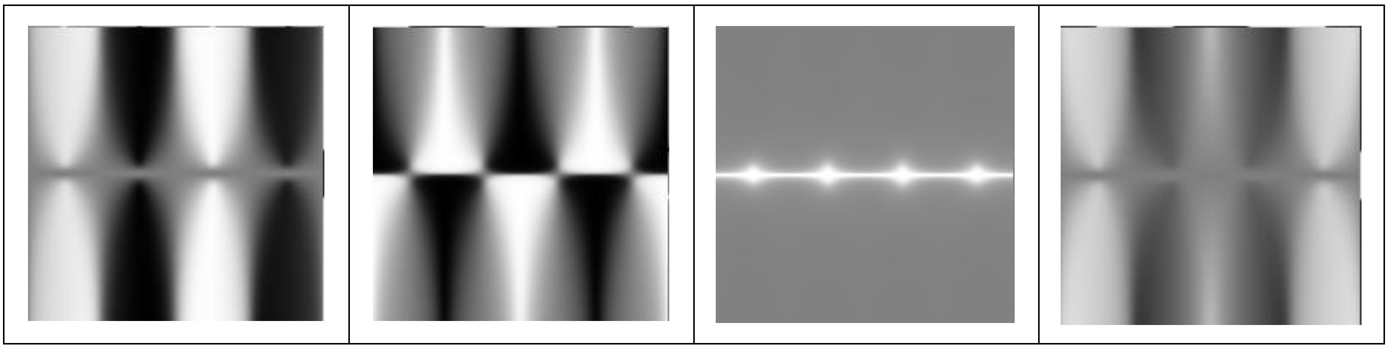
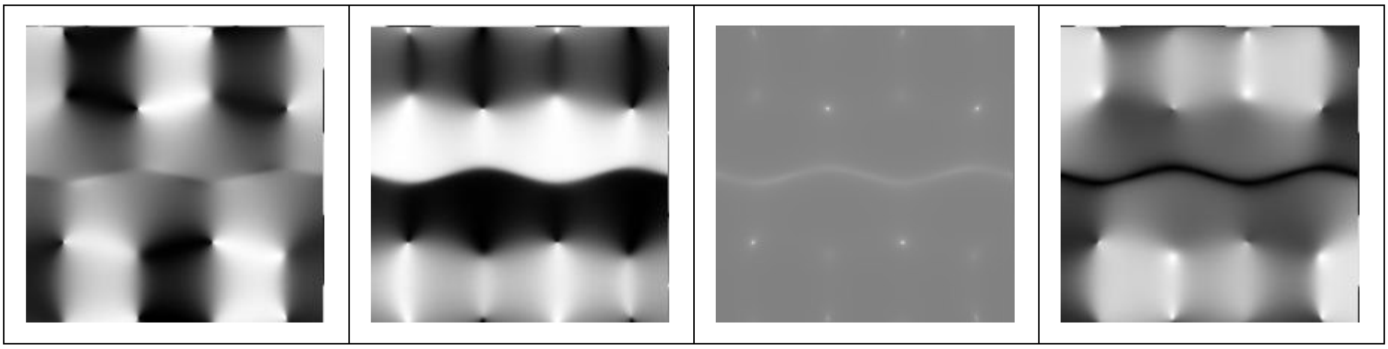


Fig. 7. The basic *M*-images of the function n_2 / n_3.



(a)



(b)

Fig. 8. The basic *M*-images of functions: (a) $\partial^2 u / \partial x^2$ and (b) $\partial^2 u / \partial x \partial y$.

3. INTEGRATION OF FV-MODELS

Consider the inverse differentiation problem: finding the antiderivative (integration). Let us refer to formulas (7) and (8), i.e., the equations

$$\frac{\partial u}{\partial x} = \frac{a}{c}, \ \frac{\partial u}{\partial y} = \frac{b}{c}.$$

Note that the coefficient c in the denominators is the doubled area of the triangle with the vertices $(x_1,\ y_1), (x_2,\ y_2)$, and $(x_3,\ y_3)$ in the plane xOy, i.e., is calculated by formula (3).

The coordinates of the nodes of the approximation grid are known or can be easily determined for the given domain of the function and the sizes of the *M*-images. Hence, we can calculate the coefficient c and then the coefficients a and b by the formulas

$$a = \frac{\partial u}{\partial x} c, \ b = \frac{\partial u}{\partial y} c.$$

This leads to an indefinite local integral at the point (x_i, y_j):

$$ax + by + cz = 0.$$

To find the antiderivative, assume that $z_1 = f(x_1,\ y_1)$ is known; in other words, we calculate

$$z_1 = x_1 \sin\left(\pi \frac{y_1}{0.5}\right) + y_1^{\ 2} \cos\left(\pi \frac{x_1}{0.5}\right).$$

Then

$$d = -ax_1 - by_1 - cz_1.$$

Completing the definition of the coefficients of the local equation, we obtain the corresponding values for the other nodes of the approximation grid segment (Fig. 9):

$$z_2 = -\frac{a}{c} x_2 - \frac{b}{c} y_2 - \frac{d}{c},$$

$$z_3 = -\frac{a}{c} x_3 - \frac{b}{c} y_3 - \frac{d}{c}.$$

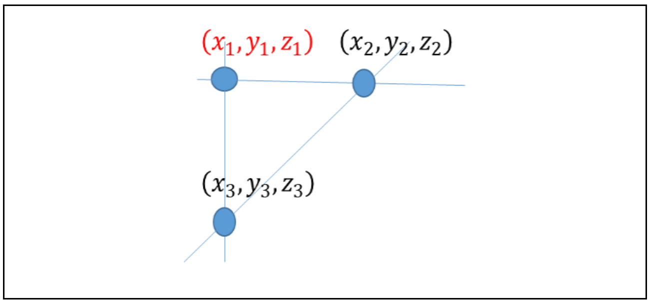


Fig. 9. The nodes of an approximation grid.

Now we apply the local integration algorithm to the second derivative. As expected, the resulting *M*-images should be as similar as possible to the *M*-images of the FV-model of the first derivative (Fig. 6). The initial *M*-images are the *M*-images obtained for the derivatives $\frac{\partial^2 u}{\partial x^2}$ and $\frac{\partial^2 u}{\partial x \partial y}$, presented in Figs. 7 and 8, respectively. The result of the local integration algorithm is shown in Fig. 10. These *M*-images visually coincide with the ones in Fig. 6.

The error in the resulting images is due to the loss of accuracy when passing to integer values of the palette. In many cases, this error is insignificant since the values differ by the third decimal place.

Applying the local integration algorithm to the *M*-images of the first derivative yields the result in Fig. 11. It is quite comparable to the *M*-images of the original function u; see the FV-model in Fig. 2.

Consider an example of differentiating function (1) using the proposed approach. Figure 12 demonstrates the *M*-images of the FV-model for the expression

$$u = a^2 + b^2 - x^2 - y^2 - \sqrt{\left(a^2 - x^2\right)^2 + \left(b^2 - y^2\right)^2}$$

with $a = 0.5$ and $b = 1$ on the domain $[-1, 1] \times [-1, 1]$.

The *M*-images of the partial derivative along the axis Ox are shown in Fig. 13.

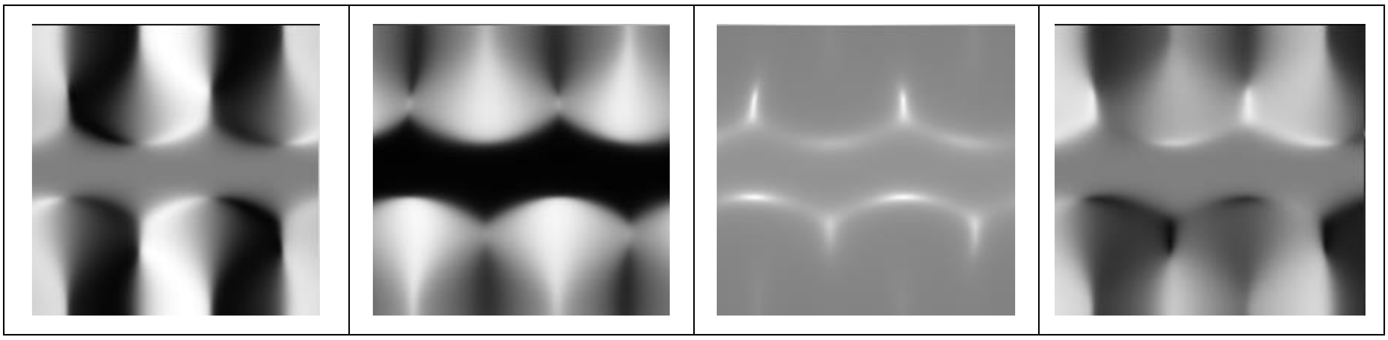


Fig. 10. The basic *M*-images of the second derivative of the function *u*.

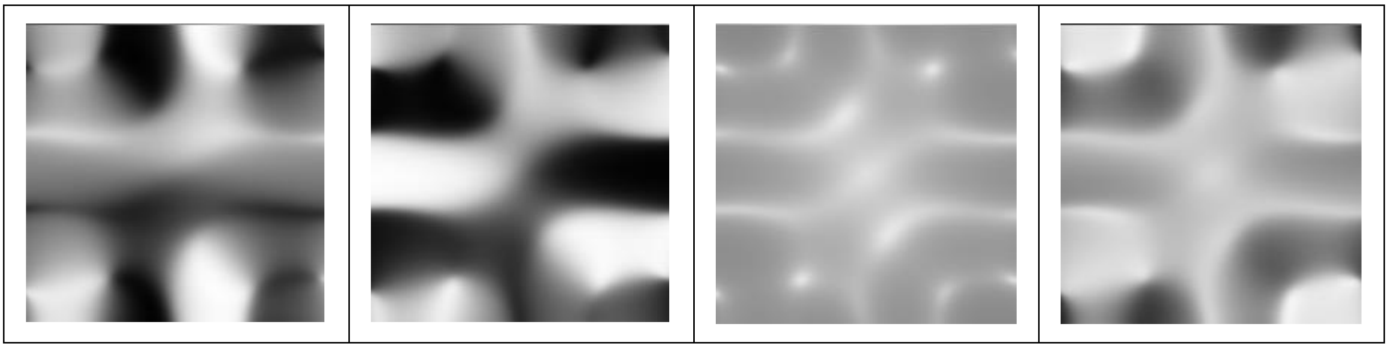


Fig. 11. The basic *M*-images of the integral of the second derivative of the function *u*.

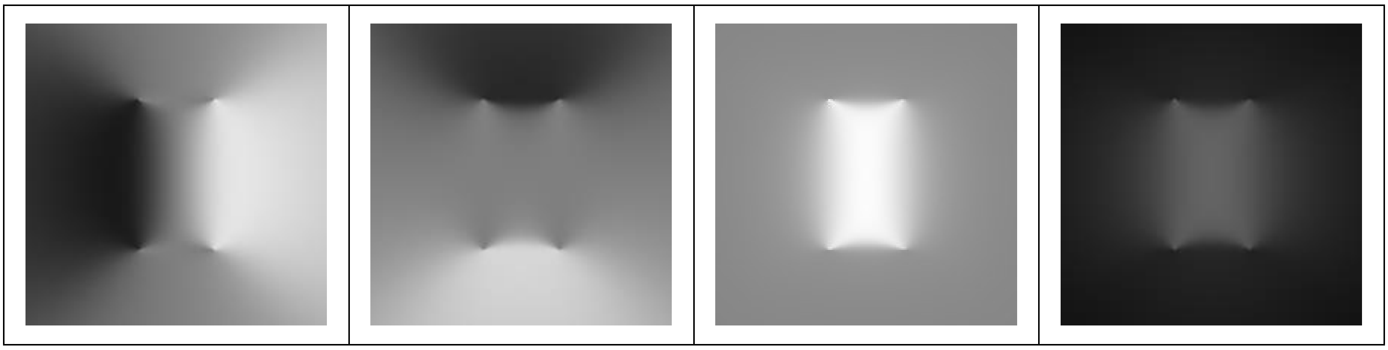


Fig. 12. The basic *M*-images of the integral of the second derivative of the function *u*.

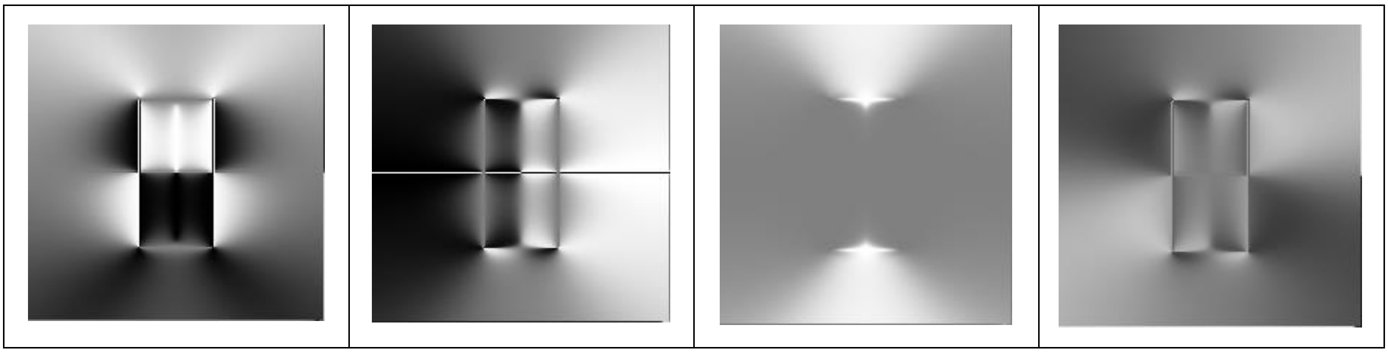


Fig. 13. The basic *M*-images of the first derivative of the function n_1/n_3.

CONCLUSIONS

This paper has presented a tool for automating the differentiation and integration of wide-range complex algebraic functions based on functional voxel modeling. Due to linear approximation, the proposed approach allows differentiating and integrating a wide class of undifferentiated functions that arise in *R*-functional modeling. Despite a visual error in the result, this approach is robust and ensures a solution even if the function has no mathematical form. An example of two-dimensional functions has been provided to demonstrate and visually compare the results. Note that the algorithm is easily transferable to any dimension.

REFERENCES

1. Rystsov, D., Automatic Differentiation, *Abnormal Programming, Mathematics*, Habr, 2009. http://habrahabr.ru/post/63055/ (In Russian.)
2. Neidinger, R.D., Introduction to Automatic Differentiation and MATLAB Object-Oriented Programming, *SIAM Review*, 2010, vol. 52, no. 3, pp. 545–563. DOI:10.1137/080743627.
3. Baydin, A.G., Pearlmutter, B.A., Radul, A.A., and Siskind, J., Automatic Differentiation in Machine Learning: a Survey, *Journal of Machine Learning Research*, 2018, vol. 18, art. no. 153, pp. 1–43.
4. Rall, L.B., *Automatic Differentiation: Techniques and Applications*, Lecture Notes in Computer Science, vol. 120, Berlin–Heidelberg–New York, Springer-Verlag, 1981. ISBN 3-540-10861-0.
5. Conal, E., The Simple Essence of Automatic Differentiation, *Proc. ACM Program. Lang.*, 2018, vol. 2, iss. ICFP, art. no. 70, pp. 1–29. DOI: https://doi.org/10.1145/3236765.
6. Liao, H., Pearlmutter, B.A., Potluru, V.K., and Woodruff, D.P., Automatic Differentiation of Sketched Regression, *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics* (*AISTATS*), Palermo, Italy, 2020, PMLR, vol. 108, pp. 4367–4376.
7. Bakhvalov, N.S., Zhidkov, N.P., and Kobel'kov, G.M., *Chislennye metody* (Numerical Methods), 3rd ed., Moscow: BINOM. Laboratoriya znanii, 2004. (In Russian.)
8. Berezin, I.S. and Zhidkov, N.P., *Metody vychislenii. T. I* (Computing Methods. Vol. I), 2nd ed. (In Russian.)
9. Mysovskikh, I.P., *Lektsii po metodam vychislenii* (Lectures on Computing Methods), Moscow: Nauka, 1982. (In Russian.)
10. Tolok, A.V., *Funktsional'no-voksel'nyi metod v komp'yuternom modelirovanii* (The Functional Voxel Method in Computer Modeling), Moscow: Fizmatlit, 2016. (In Russian.)
11. Tolok, A.V., *Lokal'naya komp'yuternaya geometriya* (Local Computer Geometry), Moscow: IPR-Media, 2022. (In Russian.)

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

Received July 18, 2022, and revised September 15, 2022.
Accepted October 3, 2022.

Author information

Tolok, Alexey Vyacheslavovich. Dr. Sci. (Eng.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ tolok_61@mail.ru

Tolok, Nataliya Borisovna. Cand. Sci. (Eng.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ nat_tolok@mail.ru

Cite this paper

Tolok, A.V. and Tolok, N.B., Differentiation and Integration in Functional Voxel Modeling. *Control Sciences* **5,** 51–57 (2022). http://doi.org/10.25728/cs.2022.5.5

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com

ERRATUM TO: THE SIMULTANEOUS START OF ACTIONS IN A DISTRIBUTED GROUP OF AUTOMATIC DEVICES: A DECENTRALIZED CONTROL METHOD WITH A SIGNAL REPEATER

G.G. Stetsyura

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia

✉ gstetsura@mail.ru

The paper "The Simultaneous Start of Actions in a Distributed Group of Automatic Devices: A Decentralized Control Method with a Signal Repeater" by G.G. Stetsyura was originally published in *Control Sciences*, 2022, no. 3, pp. 46–54.

In Section 4 of the original publication, the formula $D_j = C - t_j + a_j$ should read as follows:

$$D_j = 2(C - t_j) + a_j.$$

Author information

Stetsyura, Gennady Georgievich. Dr. Sci. (Eng.), Chief Researcher, Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ gstetsura@mail.ru

Cite this paper
Stetsyura, G.G. Erratum to: The Simultaneous Start of Actions in a Distributed Group of Automatic Devices: A Decentralized Control Method with a Signal Repeater, *Control Sciences* **5,** 58 (2022).

Translated into English by *Alexander Yu. Mazurov*,
Cand. Sci. (Phys.–Math.),
Trapeznikov Institute of Control Sciences,
Russian Academy of Sciences, Moscow, Russia
✉ alexander.mazurov08@gmail.com