ISSN 2782-2427

CONTROL SCIENCES

3/2025

[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

Editor-in-Chief
D.A. Novikov, RAS Academician

Deputy Editor-in-Chief
F.F. Pashchenko

Executive Editor-in-Chief
N.E. Maximova

Editor
L.V. Petrakova

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: July 29, 2025

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

3.2025

CONTENTS

TECHNICAL CONDITION MONITORING METHODS TO MANAGE THE REDUNDANCY OF SYSTEMS. PART II: Classical Models

V. N. Bukov*, A. M. Bronnikov**, A. S. Popov***, and V. A. Shurman****

*Airborne Aeronautical Systems, Moscow, Russia
**Bauman Moscow State Technical University, Moscow, Russia
***Zhukovsky–Gagarin Air Force Academy, Voronezh, Russia
****Institute of Aircraft Equipment, Zhukovsky, Russia

*✉ v_bukov@mail.ru, **✉ bronnikov_a_m@mail.ru, ***✉ saga30@yandex.ru, ****✉ shurman@niiao.ru

Abstract. Redundancy management of a technical system involves a monitoring procedure (control of the current state of its components) to reconfigure the system as needed. Consisting of four parts, this survey presents modern and newly developed technical condition monitoring methods for redundancy management. Part I was devoted to a general description of built-in control, voting schemes, and fidelity rules; control codes and program execution control methods were briefly covered. Part II of the survey considers fault diagnosis methods based on the classical modeling of the system diagnosed in the discrete time and frequency domains. The Chow–Willsky fault detection scheme, as well as the definition of a residual and its generation procedures in the diagnosis problem, are presented. The main model-based diagnosis methods using equation errors, observers, parity equations, and redundant variables are described. In conclusion, the robustness problem of diagnosis methods of the corresponding type is briefly discussed.

Keywords: technical condition monitoring, redundancy management, diagnosis, residual, analytical models of systems, equation error method, observers, Beard filter, fault detection via parity equations, redundant variable method, robustness of methods.

INTRODUCTION

Mathematical model-based fault diagnosis methods[1] detect a mismatch between the real measurements of input and output signals of a system (on the one hand) and the corresponding signals of its mathematical model (on the other), expressed by the so-called *residual*. These methods started to be actively developed in the 1970s. In particular, static or parallel redundancy schemes [1, 2] were applied, which can be obtained directly from measurements (hardware redundancy) or analytical relations (analytical redundancy). Such methods were reviewed in [3–5].

In part II of the survey, we highlight the methods commonly used in fault diagnosis systems of various-purpose complex technical systems. Their primary indisputable advantage is engineering transparency, i.e., the intuitive clarity of the processes and causal relations for experts in the corresponding technical field. The effectiveness of the methods under consideration depends on particular conditions of application, and their choice is determined by the following factors: the functional purpose of the device, its structural organization and technological features of manufacturing and operation, and the required reliability and fidelity indicators.

1. GENERAL CONDITIONS OF MODEL-BASED METHODS

Figure 1 shows the Chow–Willsky scheme [4], which generalizes most fault detection methods for a system with actuators, an object, and sensors.

[1] In this context, we mean only analytical modeling in the "classical" setting: the central link is the reproduction (with certain level of detail) of the operation laws of a system diagnosed; approaches with other types of models will be presented in parts III and IV of the survey.

The real-time fault detection scheme includes two main blocks:

• Residual generation: a signal called the residual is formed by using the input and output signals of a monitored process. This residual should be independent of the input and output signals and should take a value equal to/close to 0 in the absence of faults.

• Residual estimation: the value of the residual is analyzed, and the presence or absence of a fault is decided accordingly.

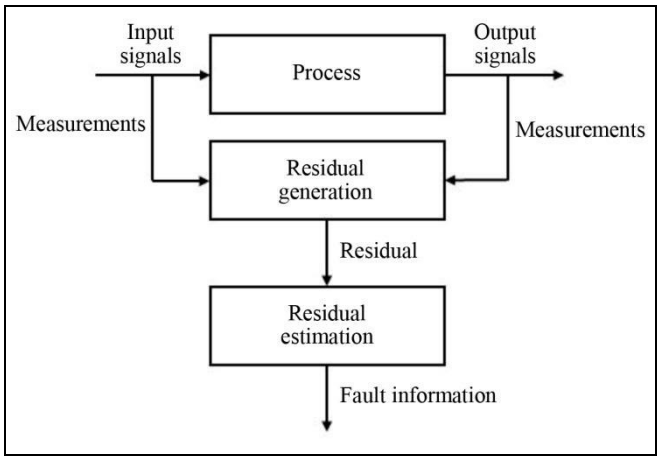

Fig. 1. The Chow–Willsky scheme.

Decision rules involve simple tolerance control or complex signal transformation methods, including statistical analysis.

In this case, the main attention of researchers is focused on residual generation since residual estimation procedures are largely universal.

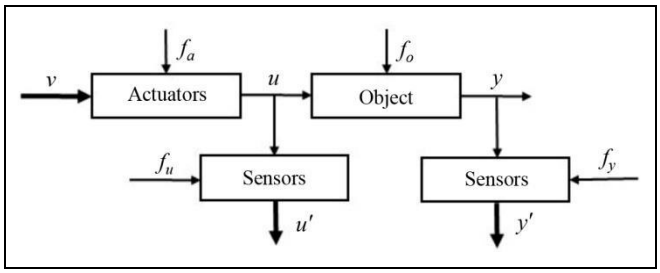

Fig. 2. The fault topology of a dynamic system.

Figure 2 explains the topology of faults affecting the system operation, with the following notation: v is an external input (disturbance); u is a control input applied to the object; y is the output signal of the object; u' and y' are measured signals (available for processing); f_{a}, f_{o}, f_u, and f_y are formalized representations of faults occurring in actuators, the object, and sensors of object's input and output signals, respectively. Only the signals v, u', and y' are known with some accuracy.

Despite the nonlinearity of most real systems monitored, modeling and identification methods for linear systems are often used to avoid the difficulties inherent in nonlinear models. In most cases, this is not a significant limitation since the dynamic system is monitored in the neighborhood of an expected operation mode, and deviations from the mode are described well by linear models.

1.1. Description of Analytical Models

According to the linearity hypothesis of a dynamic process, its behavior in discrete time $t = 0, 1, 2,\ldots$ is described by the state-space model [6–10]

$$x_{t+1} = Ax_t + Bu_t, \quad y_t = Cx_t, \tag{1}$$

where $x_t \in \Re^n$ is the state vector of the system; $u_t \in \Re^r$ is the vector of true input signals; $y_t \in \Re^m$ is the vector of true output signals; A, B, and C are the numerical matrices of system coefficients.

The effect of faults on the object is described by the model

$$x_{t+1} = Ax_t + Bu_t + f_{ab.t}, \quad y_t = Cx_t + f_{c.t}, \tag{2}$$

where $f_{ab.t} = \Delta Ax_t + \Delta Bu_t$ and $f_{c.t} = \Delta Cx_t$; ΔA, ΔB, and ΔC are the variations of the coefficient matrices due to the occurred faults. (The fault vector f_{o} is represented by two vectors $f_{ab.t}$ and $f_{c.t}$, associated with time instant t). Sometimes faults are treated differently: $f_{ab.t} = A\Delta x_t + B\Delta u_t$ and $f_{c.t} = \Delta y_t$, where Δx_t is an external disturbance for internal variables; Δu_t is actuator errors (faults); Δy_t is sensor errors (faults).

State-space descriptions provide general and mathematically rigorous tools for modeling the system and robustly generating uncertainties in both the deterministic and stochastic cases (measurements without noise and noisy measurements, respectively). Therefore, the system matrices *A*, *B*, and *C* in canonical form for models (1) and (2) can be obtained using multivariate identification procedures.

The input and output measurements are affected by faults according to the formulas

$$u'_t = u_t + f_{u.t}, \; y'_t = y_t + f_{y.t},$$

where u'_t and y'_t are the measured values of the input and output signals.

Typically, step and ramp (gradient) signals are used to model sudden and evolving faults, representing a displacement (a fixed change during one time

instant) and a drift (monotonic changes on a sequence of time instants), respectively.

Measurement noise affecting sensor output signals is often described by uncorrelated Gaussian processes.

The general model-based fault detection (FD) problem can be solved only with knowledge of the measured sequences u_t' and y_t'. In addition, a priori knowledge of the characteristics of the received signals u_t' and y_t' is widely used. Examples include the spectrum, dynamic range of the signal, and its variations. However, the need for a priori information about the signals monitored and the dependence of the signal characteristics on the unknown conditions of the system diagnosed are the main drawbacks of this class of methods.

Along with models (1) and (2), the operation of a system diagnosed can be described by frequency-domain models of the form

$$y_t = W_y^u(z)u_t + \Delta W_y^u(z)u_t, \qquad (3)$$

where z is the forward shift operator (one time instant ahead)[2]; $W_y^u(z)$ is the transfer (rational polynomial) matrix relating the input u_t to the system output y_t; $\Delta W_y^u(z)$ is the variation of the transfer matrix $W_y^u(z)$ due to faults.

As a rule, model (3) is applied under known frequency characteristics of faults and disturbances: information in frequency spectra can be used as criteria for fault detection.

1.2. Residual Generation

The general structure of a residual generator [11, 12] is demonstrated in Fig. 3. Here, z_t is an auxiliary signal (as a rule, meaningful for the developer, e.g., the state vector of the system or its parts); r_t is the residual signal; $W_1(u_t, y_t)$ and $W_2(z_t, y_t)$ are the main blocks of the generator.

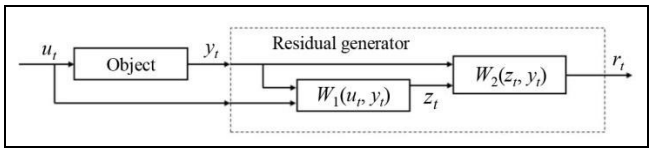

Fig. 3. The general structure of a residual generator.

Regardless of the method applied, residual generation is nothing but a linear mapping with the inputs consisting of the inputs and outputs of the process monitored. In the normal condition (no fault), the equality $r_t = 0$ holds.

The simplest residual generator is obtained when the transfer matrix W_1 coincides with model (3) of the object: $W_1(u_t, y_t) = W_y^u(z)$. In other words, this is a real process description yielded by a system identification procedure (e.g., an autoregressive exogenous model [13]).

The simplest, and most widespread, fault detection method is to compare the residual r_t or its some function $J(r_t)$ with a fixed threshold ε or a threshold function ε_t as follows:

$$J(r_t) = \begin{cases} \le \varepsilon_t & \text{for } f_t = 0 \\ > \varepsilon_t & \text{for } f_t \ne 0, \end{cases} \qquad (4)$$

where f_t is the generalized fault vector. Thus, if the residual exceeds a fixed threshold, a fault is declared.

Such a test is particularly effective with fixed thresholds ε if the object operates predominantly in a steady state. It responds after either a sudden large fault or a persistent, gradually increasing fault.

In most practical cases, process parameters are completely unknown or known with insufficient accuracy. Then they can be determined using parameter estimation methods by measuring the input and output signals u_t and y_t if the main structure of the model is known.

2. THE EQUATION ERROR METHOD

For a *Single Input, Single Output* (SISO) object, the discrete-time model of order n is written in the vector form

$$y_t = \Psi^{\mathrm{T}}\Theta,$$

where $\Theta = \begin{bmatrix} a_1 & \cdots & a_n & b_1 & \cdots & b_n \end{bmatrix}^{\mathrm{T}}$ denotes the vector of object's parameters; $\Psi = \begin{bmatrix} y_{t-1} & \cdots & u_{t-n} \end{bmatrix}^{\mathrm{T}}$ is the vector of discrete-time sample data at n consecutive time instants (measurement points).

The error signal is given by

$$e_t = y_t - \Psi^{\mathrm{T}}\hat{\Theta},$$

where $\hat{\Theta}$ is the vector of parameter estimates. Based on the equations

$$J(\hat{\Theta}) = \sum_t e_t^2 = e^{\mathrm{T}}e, \ \frac{dJ(\hat{\Theta})}{d\hat{\Theta}} = 0,$$

the least-squares estimates of the parameters are determined by the formula

$$\hat{\Theta}_t = (\Psi^{\mathrm{T}}\Psi)^{-1}\Psi^{\mathrm{T}}y_t. \qquad (5)$$

[2] Such an interpretation ignores the initial conditions of the model. An alternative is to treat the operator as the z-transform, thereby considering the initial conditions.

As described in [14–16], the estimates (5) can also be computed using the recurrent[3] least-squares algorithm in real time $t = 0, 1, 2,\ldots$ with respect to the estimates at time instant *t*:

$$\hat{\Theta}_{t+1} = \hat{\Theta}_t + \gamma_t \left(y_{t+1} - \Psi_{t+1}^{\mathrm{T}} \hat{\Theta}_t \right),$$

where

$$\gamma_t = (\Psi_{t+1}^{\mathrm{T}} P_t \Psi_{t+1} + 1)^{-1} P_t \Psi_{t+1},\ P_{t+1} = (I_{2n} - \gamma_t \Psi_{t+1}^{\mathrm{T}}) P_t,$$

and I_{2n} stands for an identity matrix of dimensions $2n \times 2n$. The initial conditions P_0 for the matrix P_t is either the identity matrix or a supposed covariance matrix of the initial errors of the estimates $\hat{\Theta}_0$.

It is possible to improve convergence using filtering methods. In particular, a Kalman filter can be applied for parameter estimation in the case of noisy measurements [17, 18].

Fault diagnosis is performed by the difference between the vector $\hat{\Theta}_t$ of parameter estimates and its "reference" values, corresponding either to the absence or presence of definite faults in the system. For this purpose, one may employ the approaches described earlier; see Section 5 in part I of the survey [19].

3. OBSERVER-BASED APPROACHES

The principal idea of observer/filter-based methods is to estimate the system outputs from measurements using Lüenberger observers (in the deterministic case) or Kalman filters (in the presence of random noises). The output estimation error (or its weighted value) is used as a residual.

Note that in the case of observer-based fault diagnosis, the outputs must be estimated and the estimation of the state vector is often not required [20]. In addition, the advantage of an observer is the flexible choice of its gains, which leads to a rich variety of fault diagnosis schemes [21–30].

In the linear dynamic model (1) with exactly known matrices A, B, and C, the variables are restored using the observer

$$\hat{x}_{t+1} = A\hat{x}_t + Bu_t + He_t,\ e_t = y_t - C\hat{x}_t, \qquad (6)$$

where $\hat{x}_t \in \Re^n$ denotes the state vector estimate; $e_t \in \Re^m$ is the output error. At the same time, the state error $e_t^x \in \Re^n$ satisfies the equations

$$e_t^x = x_t - \hat{x}_t, \quad e_{t+1}^x = (A - HC)e_t^x,$$

asymptotically vanishing if the matrix $A - HC$ is stable.

In the presence of disturbances and faults, we have the equations

$$x_{t+1} = Ax_t + Bu_t + Qv_t + L_1 f_t,\ y_t = Cx_t + Rw_t + L_2 f_t,$$

where v_t is the unmeasured vector of input disturbances; w_t is the unmeasured vector of output disturbances; f_t is the reduced vector of faults additively affecting the object's operation; Q and R are disturbance effect matrices; L_1 and L_2 are fault effect matrices.

Under the conditions $v_t = 0$ and $w_t = 0$, the state estimate satisfies the equations

$$e_{t+1}^x = (A - HC)e_t^x + L_1 f_t - HL_2 f_t,$$
$$e_t = Ce_t^x + L_2 f_t,$$

which describe the effect of the generalized faults f_t on the errors e_t^x and e_t, taken as residuals. The matrix H influences the properties of the residuals. The requirements for choosing H are stability and sensitivity to the disturbances v_t and w_t. If the signals are subjected to noises, a Kalman filter [17] should be used instead of classical observations.

If faults manifest themselves in the form of parameter variations ΔA or ΔB, the process behavior is described by

$$x_{t+1} = (A + \Delta A)x_t + (B + \Delta B)u_t,\ y_t = Cx_t,$$

and the estimation errors are written as

$$e_{t+1}^x = (A - HC)e_t^x + \Delta A x_t + \Delta B u_t,\ e_t = Ce_t^x.$$

In this case, the variations ΔA and ΔB are multiplicative errors, and the changes in errors depend on both parameter variations and changes in the input and state variables. Therefore, the effect of parameter variations on errors is less unambiguous than in the case of additive faults f_t. Particular solutions can be found in [14, 25].

For *Multiple Input*, *Multiple Output* (MIMO) processes, the following decomposition is applied: either a single observer excited by a single output [26], or a bank of observers excited by all outputs with hypothesis testing [27], or a bank of observers excited by either separate outputs [26] or all outputs except one [28, 29]. This approach allows detecting faults of different combinations of sensors.

In the MIMO case, the gain matrix *H* of the state observer in equation (6) is chosen so that fault signals $L_1 f_t$ change in a definite direction and fault signals $L_2 f_t$ in a definite plane [29, 31].

[3] In programming, the equivalent term "recursive" is used.

In the presence of directional residual vectors, the fault isolation problem is reduced to determining the nearest fault signature direction for the residual vector. The original form of a "fault detection filter" was proposed by R. Beard [31] and H. Jones [32] for generating directional residual vectors.

Subsequently, many simpler methods were proposed, including a "robust fault detection filter" [33]. It represents a class of Lüenberger observers with a special feedback gain matrix. Another possibility is to use output observers or the so-called generalized observers, e.g., observers with an unknown input with output reconstruction, if the state estimate $\hat{x}_t$ is not of primary interest.

4. BEARD FILTER

We will illustrate the possibilities and limitations of approaches using fault signature directions with the Beard filter [31].

Consider a linear dynamic system with discrete time $t = 1,\ 2,...$, described by the equation

$$x_{t+1} = Ax_t + Bu_t + B\begin{bmatrix}0 & \cdots & f_i & \cdots & 0\end{bmatrix}^{\mathrm{T}}, \quad x_{t=0} = x_0, \tag{7}$$

where $x_t \in \Re^n$ is the state vector; $u_t \in \Re^l$ is the control vector; $A \in \Re^{n\times n}$ and $B \in \Re^{n\times l}$ are matrices of constant coefficients; the additional term $B\begin{bmatrix}0 & \cdots & f_i & \cdots & 0\end{bmatrix}^{\mathrm{T}}$ corresponds to an unknown disturbance (external signal) f_i at the ith input of the system, $i = \overline{1, l}$, $l \le n$. By assumption, all system faults under consideration are reduced to a set of unknown external signals.

The Beard filter allows detecting m faults of the specified type and finding the corresponding column b_i of the matrix B.

The filter involves an auxiliary dynamic system of the form

$$w_{t+1} = Aw_t + Bu_t + K(x_t - w_t),\ w_{t=0} = w_0, \tag{8}$$

which repeats the structure of the diagnosed system (7) with the additional term $K(x_t - w_t)$. The coefficient matrix K is assigned in a special way.

Subtracting equation (8) from equation (7) gives

$$x_{t+1} - w_{t+1} = A(x_t - w_t) - K(x_t - w_t) + B(u_t - u_t) + B\begin{bmatrix}0\\ f_i\\ 0\end{bmatrix}. \tag{9}$$

By introducing the residual $r_t = x_t - w_t$, we write equation (9) as

$$r_{t+1} = (A - K)r_t + B\begin{bmatrix}0\\ f_i\\ 0\end{bmatrix}. \tag{10}$$

Note that the residual is independent of the control vector u_t and state x_t of the system, being determined only by the presence of an external signal f_i and the properties of the matrix $(A - K)$.

Let the coefficient matrix K be chosen so that

$$A - K = \alpha I_n, \tag{11}$$

where I_n denotes an identity matrix of dimensions $n \times n$; α is a real number with norm $|\alpha| < 1$. Then the expression (10) takes the form

$$r_{t+1} = \alpha r_t + B\begin{bmatrix}0\\ f_i\\ 0\end{bmatrix}, \quad r_{t=0} = r_0, \tag{12}$$

and the filter is stable. Due to this property and the diagonality of the matrix (11), the residual r_t gradually converges the vector

$$r_{\lim} = \frac{1}{1-\alpha}\begin{bmatrix}b_{1i} & \cdots & b_{ii} & \cdots & b_{ni}\end{bmatrix}^{\mathrm{T}} f_i.$$

The matrix B of equation (7) contains m columns, each defining a particular direction in the n-dimensional space.[4] Thus, the issue is to determine the columns of the matrix B on which the vector (12) is projected in the vector representation. All l external signals (faults) f_i can be monitored simultaneously.

The possibilities of the Beard filter are not exhausted by this. Similar calculations yield relations for the column-wise determination of variations in the elements of the matrix A in equation (7). In this case, it is necessary to consider the equation

$$x_{t+1} = Ax_t + Bu_t + A\begin{bmatrix}0 & \cdots & \mu_i & \cdots & 0\end{bmatrix}^{\mathrm{T}}. \tag{13}$$

Under the appropriate conditions, the residual converges to

$$r_{\lim} = \frac{1}{1-\alpha}\begin{bmatrix}a_{1i} & \cdots & a_{ii} & \cdots & a_{ni}\end{bmatrix}^{\mathrm{T}} \mu_i.$$

The remaining considerations are analogous to the above. All n external signals (faults) μ_i can be estimated simultaneously.

[4] The columns are supposed to be linearly independent.

The main properties of the Beard filter are as follows:

– Constant faults of the specified type are detected.

– Fault estimation requires time to complete the filter transients.

– When only part of the state vector is measured, $y_t = Cx_t$, where $y_t \in \Re^m$ and $C \in \Re^{m\times n}$, one can estimate only m faults described by m columns according to the schemes (6) and (12) from the set of all columns of the matrices A and B.

For further presentation, we recall some formulas from part I of the survey; see Section 5 in [19].

Let a syndrome be an m-dimensional vector $\Delta = \begin{bmatrix}\Delta_1 & \cdots & \Delta_m\end{bmatrix}^{\mathrm{T}}$, and let the possible faults of the object be formalized by an n-dimensional vector $F = \begin{bmatrix}f_1 & \cdots & f_n\end{bmatrix}^{\mathrm{T}}$. Then

$$\Delta = \Phi(Y, Z, F) = 0. \qquad (14)$$

For the syndrome Δ to respond to a single fault $f_i \neq 0$, it suffices to satisfy the nonzero sensitivity condition

$$\exists j : \frac{\partial \Phi_j}{\partial f_i} \neq 0. \qquad (15)$$

In the case of a multiple fault (when several faults f_i occur simultaneously), the additional condition

$$\begin{bmatrix} \partial\Phi_1 / \partial f_1 & \cdots & \partial\Phi_1 / \partial f_n \\ \vdots & \ddots & \vdots \\ \partial\Phi_m / \partial f_1 & \cdots & \partial\Phi_m / \partial f_n \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} \neq 0, \qquad (16)$$

(no mutual compensation (14) for the effects of these faults) must be valid.

The Beard filter is a special case of the approach with algebraic invariants, where redundancy is introduced artificially via the auxiliary system (8). The sensitivity condition (15) always holds for each time instant: the syndrome (14) is the residual r_{i+1} and , due to equations (12) and (13), $\partial r_{i+1} / \partial f_i = B$ and $\partial r_{i+1} / \partial \mu_i = A$. Condition (16) turns into the above property of a bounded number of detectable faults, as it prescribes the linear independence of the columns $\begin{bmatrix}\partial r_{1.i+1} / \partial f_k & \cdots & \partial r_{m.i+1} / \partial f_k\end{bmatrix}^{\mathrm{T}}$, making the corresponding matrix nonsingular and, as a necessary condition, the number of residuals $r_{j.i+1}$ equal to the number of detectable faults f_k.

5. FAULT DETECTION VIA PARITY EQUATIONS

The main idea of *fault detection via parity equations* is to check the consistency of different (usually input and output) measurements obtained at an object diagnosed and a control scheme [34].

As for observers, the model parameters and the structure of the process observed must be known a priori.

This fault detection approach is as follows. For a SISO object with a known transfer function $W(z) = b(z) / a(z)$, one uses, in parallel, a model of the form $W_m(z) = b_m(z) / a_m(z)$, where $a(z)$, $b(z)$, $a_m(z)$, and $b_m(z)$ are known polynomials of the operator z. According to the developer's decision, any scheme can be implemented as shown in Fig. 4.

The scheme in Fig. 4a corresponds to the residual of outputs; the scheme in Fig. 4c, to the residual of inputs; the scheme in Fig. 4b, to the intermediate solution. In these schemes, when the dynamic properties (parameters of polynomials) of the object and its model coincide, the residual r_t is 0. The presence of faults causing variations in the parameters of the object's transfer function generates a non-zero residual.

The above schemes have different sensitivities to different faults. Moreover, they can be augmented with different filters to achieve acceptable sensitivity or coarseness [34, 35]. Here, by default, the measured signals are equal to their true values: $u'_t = u_t$ and $y'_t = y_t$.

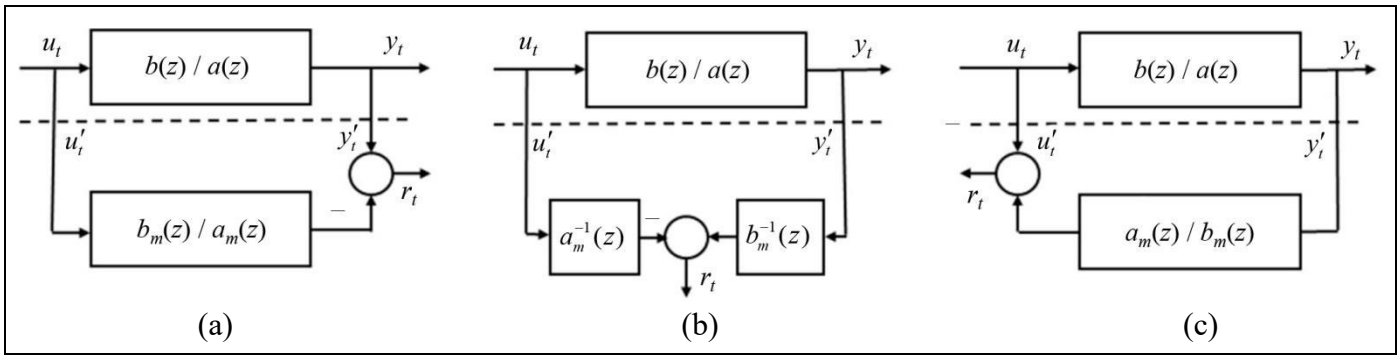

Fig. 4. Typical schemes of fault detection via parity equations: (a) by outputs, (b) by the intermediate signal, and (c) by inputs.

As shown in [36], fault detection via parity equations provides less design flexibility compared to observer-based methods without any constraints.

A disadvantage of this method is the generation of only one feature (residual) for an SISO object, which prevents from specifying the character and location of the fault: there is one equation, and a non-zero value of one residual indicates only the inconsistency of this equation. Nothing more can be extracted from this information. Various solutions were proposed to overcome this drawback [33, 37].

6. THE REDUNDANT VARIABLE METHOD

Pioneered by L.A. Mironovskii [38], this method is a most effective one. The method involves a dynamic system for variables $z \in \Re^k$, supplementing the model of an object diagnosed:

$$x_{t+1} = Ax_t + Bu_t, \qquad (17)$$

where $x_t \in \Re^n$, $u_t \in \Re^l$, and A and B are constant matrices.

The redundant variables are introduced by satisfying two conditions

$$Mz_t = 0,\ Tz_t = x_t, \qquad (18)$$

where $M \in \Re^{k\times(n+k)}$ and $T \in \Re^{n\times(n+k)}$ are known (given) constant matrices.[5] Substituting the expression (17) into conditions (18) and resolving for z_{t+1}, we obtain the following equation of order $(n+k)$:

$$z_{t+1} = \begin{bmatrix} T \\ M \end{bmatrix}^{-1} \begin{bmatrix} AT \\ 0 \end{bmatrix} Tz_t + \begin{bmatrix} T \\ M \end{bmatrix}^{-1} \begin{bmatrix} B \\ 0 \end{bmatrix} u_t.$$

It describes the behavior of the redundant system. This system, if stable, is equivalent to the original system (17) in terms of the output x_t and, at the same time, can be used to monitor its operation by the k-dimensional residual vector

$$r_t = Mz_t.$$

Various refinements of the redundant variable method are known [11, 39], both in terms of formulation and solution. They extend the original method to models with transfer functions and nonlinear models and minimize the redundancy of the system.

Like the Beard filter, the redundant variable method is a special case of the method with algebraic invariants. Therefore, the sensitivity conditions (15) and (16) apply to it as well.

[5] The rows of the matrices M and T are supposed to be linearly independent.

Let the possible faults of an object diagnosed be formalized by the equation

$$x_{t+1} = Ax_t + Bu_t + F,$$

where $F = \begin{bmatrix} f_1 & \cdots & f_n \end{bmatrix}^{\mathrm{T}}$ is the vector of unmeasured signals representing changes (faults) in the object. According to conditions (18), at the first time instant, for each formalized fault f_i we obtain

$$\frac{\partial r_{t+1}}{\partial f_i} = \begin{bmatrix} \partial r_{1.t+1} / \partial f_i \\ \vdots \\ \partial r_{k.t+1} / \partial f_i \end{bmatrix} = M \begin{bmatrix} T \\ M \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \left.\begin{matrix} 1_i \\ 0 \end{matrix}\right\}_{(n\times 1)} \\ 0_{k\times 1} \end{bmatrix}_{(n+k)\times 1}, \qquad (19)$$

where 1_i is the unit standing in the ith row of the last cofactor column. At each subsequent time instant $t+q,\ q = \overline{2,N}$, in view of formula (17), we have

$$\frac{\partial r_{t+q}}{\partial f_i} = \begin{bmatrix} \partial r_{1.t+q} / \partial f_i \\ \vdots \\ \partial r_{k.t+q} / \partial f_i \end{bmatrix} = \sum_{s=0}^{N-1} A^s M \begin{bmatrix} T \\ M \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \left.\begin{matrix} 1_i \\ 0 \end{matrix}\right\}_{(n\times 1)} \\ 0_{k\times 1} \end{bmatrix}_{(n+k)\times 1}, \qquad (20)$$

where A^s is the sth degree of the matrix A from formula (17).

In the presence of faults, non-zero values of the components (19) and (20) are not obvious and depend on the choice of the matrices T and M, which requires developer's art. For example, as the reader can easily verify, in the simplest case of a diagonal or codiagonal matrix $\begin{bmatrix} T^{\mathrm{T}} & M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, the sensitivity of all components of the residual r_{t+1} to any fault f_i has zero value.

7. ROBUSTNESS OF MODEL-BASED METHODS

The robustness of fault diagnosis algorithms to various uncertainties, including modeling errors and the active noise of meters, is a problem deserving special attention.

As a rule, in theoretical studies and practical applications, all uncertainties are generalized into additive disturbances affecting a system (an object, sensors, and actuators). Although the disturbance vector is unknown, by assumption, it can be estimated through identification procedures.

The objective of disturbance elimination approaches in a fault diagnosis system is to completely elimi-

nate the effect of disturbances on the residual, which may be generally impossible. There is a trade-off between sensitivity to errors and robustness to modeling uncertainty, so robust residual generation can be viewed as a multicriteria optimization problem [4, 40]. It consists of maximizing the effect of faults and minimizing the effect of uncertainty.

One active way to achieve robust fault detection [1] is based on an approximate representation of the modeling errors in model (1):

$$\Delta W_y^u(z)u_t \approx W_y^d(z)d_t,$$

where d_t is an unknown vector and $W_y^d(z)$ is the estimated transfer function. Robust fault detection algorithms can be obtained by using this approximate structure in the design of disturbance-eliminating residual generators.

Another approach, called passive robustness, involves a residual generator with an adaptive threshold. A simple example is a robust fault detection method with a threshold adaptor or selector [41]. This method requires no effort to develop a robust residual generator.

In this case, the residual generation uncertainty is represented as

$$r_t = H(z)\Delta W_y^u(z)u_t.$$

Under the assumption of small modeling errors,

$$\left\|\Delta W_y^u(z)\right\| \le \delta,$$

the adaptive threshold can be produced by a linear system of the form

$$\varepsilon_t = \delta H(z)u_t.$$

In this case, the threshold ε_t is no longer fixed but depends on the input signal u_t, thus being adaptive to the operating mode of the system. A fault is detected if

$$\|r_t\| > \|\varepsilon_t\|.$$

A more detailed approach to robust residual generation proceeds from a discrete object description in terms of transfer matrices:

$$y_t = \left(W_y^u(z) + \Delta W_y^u(z)\right)u_t + W_y^d(z)d_t + W_y^f(z)f_t, \quad (21)$$

where u_t and y_t are the input and output of the object; f_t is the vector of faults to be detected; d_t is the vector of disturbances; $\Delta W_y^u(z)$ is the representation error of the transfer matrix $W_y^u(z)$; $W_y^d(z)$ is the effect of the modeling disturbance; the matrices $\Delta W_y^u(z)$ and $W_y^d(z)$ together describe the modeling uncertainties.

The disturbance generator r_t in Fig. 4a is described by the equation

$$r_t = H_y(z)\left(y_t - y_t^m\right), \quad (22)$$

where y_t^m denotes the output of model (1) without faults, errors, and disturbances; $H_y(z)$ is the transfer matrix of residual processing. Substituting the expression (21) into equation (22) yields

$$\begin{aligned} r_t &= H_y(z)W_y^f(z)f_t \\ &+H_y(z)W_y^d(z)d_t + H_y(z)\Delta W_y^u(z)u_t. \end{aligned} \quad (23)$$

Extracting the first term in formula (23) against the background of the other two terms is a very difficult task. Therefore, robust residual generation is commonly reduced to satisfying the condition

$$H_y(z)W_y^d(z) = 0. \quad (24)$$

Here, various approaches are applied (e.g., the ones with observers, optimization, given structures, identification schemes, etc.). Often, such a problem cannot be completely solved, as sensitivity to faults is lost. However, compromise solutions are known.

Decoupling from disturbances can also be achieved using design methods in the frequency domain. An example is the robust detection of faults using standard H_∞ filtering.

For instance, when condition (24) fails, an approximate estimate can be obtained, e.g., in the form of the efficiency index [42]:

$$J_d = \frac{\left\|H_y(z)W_y^d(z)\right\|}{\left\|H_y(z)W_y^f(z)\right\|} \text{ or } J = \left\|\frac{\partial r}{\partial \varepsilon}\right\| \Big/ \left\|\frac{\partial r}{\partial f}\right\|.$$

More elegant and advanced H_∞ optimization methods are based on the algebraic Riccati equation [43]. Often, a slightly modified H_∞ filter is used to form the residual, i.e., the objective of design is to minimize the effect of disturbances and modeling errors on the estimation error and, consequently, on the residual. However, robust residual generation differs from robust estimation as not only disturbance attenuation is required: the residual must remain sensitive to faults while minimizing the effect of disturbances.

CONCLUSIONS

According to part II of the survey, fault diagnosis methods based on the classical modeling of an object monitored still attract many researchers. Fault diagnosis methods based on the identification approach, observers, and parity equations are actively applied.

A widely known solution is the Beard filter, representing a special case of the approach with algebraic invariants, where redundancy is artificially introduced via an auxiliary system. The redundant variable method is an effective approach to fault diagnosis. The robustness of fault diagnosis algorithms to various uncertainties, including modeling errors and the active noise of meters, deserves special attention.

In part III of the survey, we will analyze diagnosis methods based on neural networks, fuzzy and structural models, and models in the form of sets. Finally, part IV will be devoted to new approaches to fault diagnosis and combinations of different models and methods.

REFERENCES

1. Potter, I.E. and Suman, M.C., Thresholdless Redundancy Management with Array of Skewed Instruments: Technical Report, *AGARDOGRAPH. Integrity in Electronic Flight Control Systems*, 1977, no. 224, pp. 1511–1525.
2. Ivanov, N.V., Krasnov, I.A., Mozgalevskii, A.V., and Yudin, Yu.V., USSR Inventor's Certificate no. SU 475604 A1, *Byull. Izobret.*, 1975, no. 24.
3. Ray, A. and Luck, R., An Introduction to Sensor Signal Validation in Redundant Measurement Systems, *IEEE Contr. Syst. Mag.*, 1991, vol. 11, no. 2, pp. 44–49.
4. Chow, E.Y. and Willsky, A.S., Issues in the Development of a General Algorithm for Reliable Fault Detection, *Proceedings of the 19th Conf. on Decision & Control*, Albuquerque, NM, 1980.
5. *Osnovy tekhnicheskoi diagnostiki. T. 1: Model ob"ektov, metody i algoritmy diagnoza* (Fundamentals of Technical Diagnosis. Vol. 1: Models of Objects, Diagnosis Methods and Algorithms), Parkhomenko, P.P., Ed., Moscow: Energiya, 1976. (In Russian.)
6. Mozgalevskii, A.V., Technological Diagnostics. Continuous Objects (Survey), *Automation and Remote Control*, 1978, vol. 39, no. 1, pp. 118–136. (In Russian.)
7. Shumsky, A.E. and Zhirabok, A.N., *Metody i algoritmy diagnostirovaniya i otkazoustoichivogo upravleniya dinamicheskimi sistemami* (Methods and Algorithms for Diagnosis and Fault-Tolerant Control of Dynamic Systems), Vladivostok: Far-Eastern State Technical University, 2009. (In Russian.)
8. Kulida, E.L. and Lebedev, V.G., Modern Approaches to Prognostics and Health Management of an Aircraft Electromechanical Actuator, *Control Sciences*, 2024, no. 3, pp. 2–15.
9. *Diagnosis and Fault-Tolerant Control 1: Data-Driven and Model-Based Fault Diagnosis Techniques*, Puig, V. and Simani, S., Eds., Hoboken: John Wiley & Sons, 2021.
10. Mironovsky, L.A., Solov'eva, T.N., and Shintyakov, D.V., Fault Detection Optimization for Controllable Dynamic Systems, *Information and Control Systems*, 2019, no. 6, pp. 12–21. (In Russian.)
11. Mironovsky, L.A., Functional Diagnosis of Dynamic Systems, *Avtomatika i Telemekhanika*, 1980, no. 8, pp. 96–121. (In Russian.)
12. Basseville, M., Detecting Changes in Signals and Systems: A Survey, *Automatica*, 1988, vol. 24, no. 3, pp. 309–326.
13. Billings, S.A., *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*, Hoboken: Wiley, 2013.
14. *Issues of Fault Diagnosis for Dynamic Systems*, Patton, R.J., Frank, P.M., and Clark, R.N., Eds., London: Springer-Verlag, 2000.
15. *Fault Detection, Supervision and Safety for Technical Processes. Safeprocess'91*, Isermann, R. and Freyermuth, B., Eds., Oxford: Pergamon Press, 1992.
16. Chernodarov, A.V., *Kontrol', diagnostika i identifikatsiya aviatsionnykh priborov i izmeritel'nykh vychislitel'nykh kompleksov* (Control, Diagnosis, and Identification of Aircraft Instruments and Measuring Computing Systems), Moscow: Nauchtekhlitizdat, 2017. (In Russian.)
17. Jazwinski, A.H., *Stochastic Processes and Filtering Theory*, New York: Academic Press, 1970.
18. Totiev, D.A. and Vinichenko, S.N., Kalman Filter Application for Signal Processing from Measuring Devices, in *Sbornik nauchnykh trudov kafedry avtomatiki i promyshlennoi elektroniki* (Collection of Scientific Papers of the Department of Automation and Industrial Electronics), Moscow: The Kosygin State University of Russia, 2024, pp. 165–168. (In Russian.)
19. Bukov, V.N., Bronnikov, A.M., Popov, A.S., and Shurman, V.A., Technical Condition Monitoring Methods to Manage the Redundancy of Systems. Part I: Built-in Control and Partition into Classes, *Control Sciences*, 2025, no. 2, pp. 2–10.
20. Chen, J. and Patton, R.J., *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Norwell: Kluwer Academic, 1999.
21. Frank, P.M., Enhancement of Robustness on Observer-Based Fault Detection, *Int. J. Control*, 1994, vol. 59, no. 4, pp. 955–983.
22. Frank, P.M. and Ding, X., Survey of Robust Residual Generation and Evaluation Methods in Observer-Based Fault Detection System, *J. Process Control*, 1997, vol. 7, no. 6, pp. 403–424.
23. Liu, G.P. and Patton, R.J., *Eigenstructure Assignment for Control System Design*, Chichester: John Wiley & Sons, 1998.
24. Zhirabok, A.N., Shumsky, A.Y., and Suvorov, A.Y., Observer-Based Robust Fault Diagnosis: Logic-Dynamic Approach, Proceedings of the 10th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Reykjavik, Iceland, 2013, pp. 239–244.
25. Isermann, R., Estimation of Physical Parameters for Dynamic Processes with Application to an Industrial Robot, *Int. J. Control*, 1992, vol. 55, pp. 1287–1298.
26. Clark, R.N., Instrument Fault Detection, *IEEE Trans. Aero. & Electronic Systems*, 1978, vol. 14, no. 3, pp. 455–465.
27. Willsky, A.S., A Survey of Design Methods for Fault Detection in Dynamic Systems, *Automatica*, 1976, vol. 12, no. 6, pp. 601–611.
28. Wünnenberg, J. and Frank, P.M., Sensor Fault Detection Via Robust Observer, in *System Fault Diagnosis, Reliability, and Related Knowledge-Based Approaches*, Tzafestas, S., Singh, M., and Schmidt, G., Cham, Switzerland: Springer, 1987.
29. Frank, P.M., Advances in Observer-Based Fault Diagnosis, *Proceedings of TOOLDIAG'93 Conference*, Toulouse, France, 1993, pp. 817–836.
30. Zhirabok, A.N. and Zuev, A.V. Interval Observers for Estimating Unknown Inputs in Discrete Time-Invariant Dynamic Systems, *Automation and Remote Control*, 2024, vol. 85, no. 8, pp. 747–760.

31.Beard, R.V., Fault Accommodation in Linear Systems Through Self-Reorganisation, *Technical Report MVT-71-1*, Cambridge: Massachusetts Institute of Technology, Man Vehicle Lab., 1971.
32.Jones, H.L., Fault Detection in Linear Systems, *PhD Thesis*, Cambridge: Massachusetts Institute of Technology, Department of Aeronautics, 1973.
33.Chen, J., Patton, R.J., and Zhang, H.Y., Design of Unknown Input Observer and Robust Fault Detection Filters, *Int. J. Control*, 1996, vol. 63, no. 1, pp. 85–105.
34.Gertler, J., Generating Directional Residuals with Dynamic Parity Equations, *Proceedings of the IFAC/IMACS Symp. SAFEPROCESS'91*, Baden-Baden, 1991.
35.Patton, R.J. and Chen, J., A Review of Parity Space Approaches to Fault Diagnosis for Aerospace Systems, *AIAA J. of Guidance, Contr. & Dynamics*, 1994, vol. 17, no. 2, pp. 278–285.
36.*Fault Diagnosis in Dynamic Systems, Theory and Application*, Patton, R.J., Frank, P.M., and Clark, R.N., Eds., London: Prentice Hall, 1989.
37.Chow, E.Y. and Willsky, A.S., Analytical Redundancy and the Design of Robust Detection Systems, *IEEE Trans. Automatic Control*, 1984, vol. 29, no. 7, pp. 603–614.
38.Britov, G.S. and Mironovskii, L.A., The Redundant Variables Method to Check Links with Fractional-Rational Transfer Functions, in *Informatsionnye materialy NS po probleme "Kibernetika" AN SSSR* (Information Materials of the Scientific Council on the Cybernetics problem of the USSR Academy of Sciences), 1971, no. 3. (In Russian.)
39.Ignatyev, M.B. and Katermina, T.S., The Redundant Variables Method for Checking and Correction of Computing Processes in Real Time, *SPIIRAS Proceedings*, 2013, vol. 3, no. 26, pp. 234–252. (In Russian.)
40.Zhirabok, A.N. and Kucher, D.N., A Robustness Improvement Method in Fault Diagnosis, *Vologdinskie Chteniya*, 2008, no. 69, pp. 6–9. (In Russian.)
41.Emami-Naeini, A., Akhter, M., and Rock, M., Effect of Model Uncertainty on Fault Detection: The Threshold Selector, *IEEE Trans. on Automatic Control*, 1988, vol. 33, no. 2, pp. 1106–1115.
42.Frank, P.M., Ding, S.X., and Köpper-Seliger, B., Current Developments in the Theory of FDI, *SAFEPROCESS'00: Preprints of the IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, vol. 1, London: Elsevier, 2000.
43.Zhou, K., Doyle, J.C., and Glover, K., *Robust and Optimal Control*, Upper Saddle River: Prentice Hall, 1996.

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

Received November 10, 2024, and revised April 23, 2025. Accepted June 16, 2025.

Author information

Bukov, Valentin Nikolaevich. Dr. Sci. (Eng.), JSC Airborne Aeronautical Systems, Moscow, Russia
✉ v_bukov@mail.ru
ORCID iD: https://orcid.org/0000-0002-5194-8251

Bronnikov, Andrei Mikhailovich. Dr. Sci. (Eng.), Bauman Moscow State Technical University, Moscow, Russia
✉ bronnikov_a_m@mail.ru
ORCID iD: https://orcid.org/0009-0009-1216-3521

Popov, Aleksandr Sergeevich. Cand. Sci. (Eng.), Zhukovsky–Gagarin Air Force Academy, Voronezh, Russia
✉ saga30@yandex.ru

Shurman, Vladimir Aleksandrovich. JSC Research Institute of Aviation Equipment, Zhukovsky, Russia
✉ shurman@niiao.ru

Cite this paper
Bukov, V.N., Bronnikov, A.M., Popov, A.S., and Shurman, V.A., Technical Condition Monitoring Methods to Manage the Redundancy of Systems. Part II: Classical Models. *Control Sciences* **3,** 2–11 (2025).

Original Russian Text © Bukov, V.N., Bronnikov, A.M., Popov, A.S., Shurman, V.A., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 3–14.

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

ESTIMATING THE FUNDAMENTAL FREQUENCY OF A PROPELLER SHAFT–BLADE HARMONIC SERIES USING THE HILBERT TRANSFORM AND AUTOCORRELATION

O. V. Babikov* and V. G. Babikov**

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

*✉ babikov.ov@phystech.edu, **✉ babikov@ipu.ru

Abstract. The problem of estimating the fundamental frequency of a harmonic series arises in many areas of science and technology. For example, in vibration diagnosis, it is required to estimate the wear of bearings by the shift of the base of a harmonic series. In audio signal processing, this problem is associated with automatic instrument tuning. In speech synthesis, the fundamental frequency determines the pitch. In speech recognition, the frequency of the fundamental tone is an important information feature. In radio engineering, this problem is solved for the purpose of signal restoration, filtering, and decoding. In biomedical engineering, when analyzing a patient's ECG, EEG, voice, or breathing, pathologies such as arrhythmia are diagnosed by the fundamental frequency. In the detection and classification of sea vessels, the most significant information criterion is the base of a propeller shaft–blade harmonic series. This paper proposes new approaches to estimating the fundamental frequency in high noise conditions. To reduce errors, the idea is to use the method of periodograms, filtering, autocorrelation, and the Hilbert transform. Note that in high noise conditions, the estimate of the fundamental frequency of a harmonic series is significantly improved by selecting optimal parameters: the size of the time window, filtering parameters, the spectrum interval for autocorrelation, and the number of autocorrelations.

Keywords: fast Fourier transform, discrete Fourier transform, autocorrelation, Hilbert transform, fundamental frequency.

INTRODUCTION

When estimating the fundamental frequency, the original general problem is concretized considering particualr differences in the types of signals and noise and signal-noise mixture preprocessing methods. This paper addresses the problem of finding the base of a *propeller shaft–blade harmonic series* (PSBHS) generated by a sea vessel, although the methods proposed are also applicable in other engineering disciplines. Propellers are a main source of the primary hydroacoustic field of sea vessels. They generate vibrations at two key discrete frequencies: at the rotational frequency of the shaft (the shaft frequency) and the frequency representing the product of the shaft frequency and the number of propeller blades (the blade frequency) [1, 2]. For modern vessels, shaft frequencies commonly lie between 1 Hz and 6 Hz whereas blade frequencies between 6 Hz and 24 Hz [3].

Due to nonlinear effects during the emission of acoustic waves, a set of harmonics with multiple frequencies arises in the low-frequency spectrum of vessel noise. The amplitude of these harmonics, called discrete components, significantly exceeds the ambient noise level. A group of such discrete components located at multiple frequencies is called an acoustic harmonic series. If the source of these harmonics is a vessel propeller, the series is called a PSBHS.

Two types of discrete components can be distinguished in a PSBHS: the shaft and blade harmonic series. The first discrete component of the shaft harmonic series corresponds to the rotational frequency of the shaft, which is directly related to the vessel speed [2, 4]. The basic frequency of the blade

harmonic series is determined by the product of the shaft frequency and the number of propeller blades. Thus, by analyzing a PSBHS, it is possible to obtain valuable information about the vessel design, including the number of propeller blades, which is actively used in marine target recognition systems based on signal spectrum analysis [5].

Modern hydroacoustic systems analyze such signals with high accuracy. However, despite the well-developed algorithms of data processing, the final decision is mostly made by the operator [6]. The number of propeller blades is determined based on the spectrum parameters of the PSBHS presented to the operator.

Among the signal processing methods, note wavelet analysis, which allows detecting hydroacoustic signals in the form of a harmonic series and measuring the fundamental frequency of the shaft harmonic series [4]. Another approach is spectral analysis with the sequential extraction of particular discrete components and formation of the corresponding harmonic series [3].

For operator convenience, the results of narrowband frequency analysis are commonly presented in two forms: the spectrum graph, showing the location of signal harmonics, and the parameter table of detected discrete components and their characteristics. When analyzing the spectrum graph, the operator visually identifies the harmonics in order to distinguish among them the main shaft and blade frequencies. As a rule, the shaft frequency is the first discrete component, whereas the blade frequency is a subsequent one with maximum amplitude. In addition, the parameter table of the discrete components provides the operator with numerical values of the frequencies, increasing the accuracy of identification and reducing the probability of error.

Modern research aims at developing automated algorithms to decrease the dependence of analysis on the human factor. The conventional tool for analyzing acoustic noise is the *Fast Fourier Transform* (FFT), which extracts the main harmonic components of a signal. However, FFT has limited resolution, especially in low-signal and high-noise conditions. To overcome these limitations, the method of periodograms and autocorrelation analysis methods are used. For example, Welch's method reduces the scatter of power spectral density estimates [7]; MDVR (*Minimum Variance Distortionless Response*), MUSIC (*MUltiple SIgnal Classification*), and ESPRIT (*Estimation of Signal Parameters via Rotational Invariant Techniques*) provide superresolution [8].

Autocorrelation is effective in the analysis of weak signals and allows identifying regular patterns even under strong noise. Spectral envelope analysis is also actively used for extracting the characteristics of blade noise. An adaptive spectral envelope analysis technique introduced in [9] considers the effect of fluctuations in the spectrum.

In this paper, we propose several approaches to studying a mixture of a harmonic series and noise; when used together, they yield an estimate for the harmonic series base even under a small signal-to-noise ratio (SNR) for a single time window. The idea is to use the logarithm of the power spectrum of the signal under study. The envelope of the logarithm of the power spectrum is estimated by iterative averaging over three points. As shown below, this method is very close to the convolution of the spectrum and the Gaussian function, but three-point averaging allows estimating the harmonic series base with smaller errors. Next, the pseudospectrum (the difference between the logarithm of the power spectrum and its envelope) is analyzed.

The method of successive autocorrelations does not necessarily give a correct estimate for the harmonic series base. The result is influenced by the density of spectral lines, the value of the harmonic series base, and the noise level. To increase the sensitivity of this method, before applying autocorrelation, we "blur" the harmonic discrete samples a little by slightly smoothing the pseudospectrum.

After several successive autocorrelations, we calculate the so-called cepstral phase or saphe using the Hilbert transform (see the explanation in Section 1). The point is that after successive autocorrelations of the pseudospectrum, the result resembles a decreasing harmonic in the time domain, but the series itself relates to the frequency domain. Therefore, by applying the Hilbert transform to the autocorrelation of the pseudospectrum, we actually estimate the cepstral phase; when divided by the frequency to which this phase relates, the resulting value gives an estimate of the harmonic series base.

With the development of artificial intelligence methods, neural network models for classifying vessels based on their acoustic signals have become popular. *Convolutional Neural Networks* (CNNs) trained on noise spectrograms were considered in [10, 11]. Machine learning approaches improve the accuracy of determining propeller parameters but require large amounts of training data. Despite advances in automation, operator involvement remains

an important element of the analysis, especially in complex acoustic environments with possible interference and false responses.

DEMON (*Demodulation of Envelope Modulation On Noise*) is a special method for detecting modulations arising from the envelope of propeller cavitation. DEMON serves to extract the cavitation noise from the overall signal spectrum and determine the number of shafts, the rotational frequency of a shaft, and the number of propeller blades. In bioacoustics, this method is used to analyze the signals of whales, dolphins, and other animals. The limitation of this method is the need to select the noise band, which requires good skills of the hydroacoustic operator. Low interference immunity is another disadvantage of this method. The algorithm proposed below, utilizing three-point smoothing of the logarithm of the power spectrum, pseudospectrum pre-filtering, several successive autocorrelations, and the Hilbert transform, significantly improves interference immunity in PSBHS base estimation.

1. PROBLEM STATEMENT AND DETAILED DESCRIPTION OF FUNDAMENTAL FREQUENCY ESTIMATION STAGES

Now we formulate the main problem of this research. Let $\{x_k\}_{k=1}^{N}$ be a discrete series containing the sound recordings of sea vessel noise in a time window of size T (in seconds), with N samples in one window, $k \in [1, N]$, and $N = T f_s$, where f_s is the sampling frequency. It is required to develop an algorithm for estimating the PSBHS base f_0 by one time window and determine the optimal parameters of this algorithm.

The flowchart of the PSBHS base estimation algorithm is described in Fig. 1.

Periodogram construction

The periodogram is calculated using the *Discrete Fourier Transform* (DFT) for the signal x_k of length N:

$$X_j = \sum_{k=0}^{N-1} x_k e^{-i\frac{2\pi}{N}jk},$$

where X_j denotes the complex value of the spectrum at the jth frequency ($j = 0, 1, \ldots, N/2$), the frequency value in Hz is calculated as $f_j = j\frac{f_s}{N}$. Further, the algorithm deals only with a frequency range $f_{\min} \le f_j \le f_{\max}$. This limitation is due to, first, the absence of harmonic series components at frequencies above $f_{\max}$ and, second, the presence of strong noise at frequencies below $f_{\min}$. Hence, $N\frac{f_{\min}}{f_s} < j < N\frac{f_{\max}}{f_s}$.

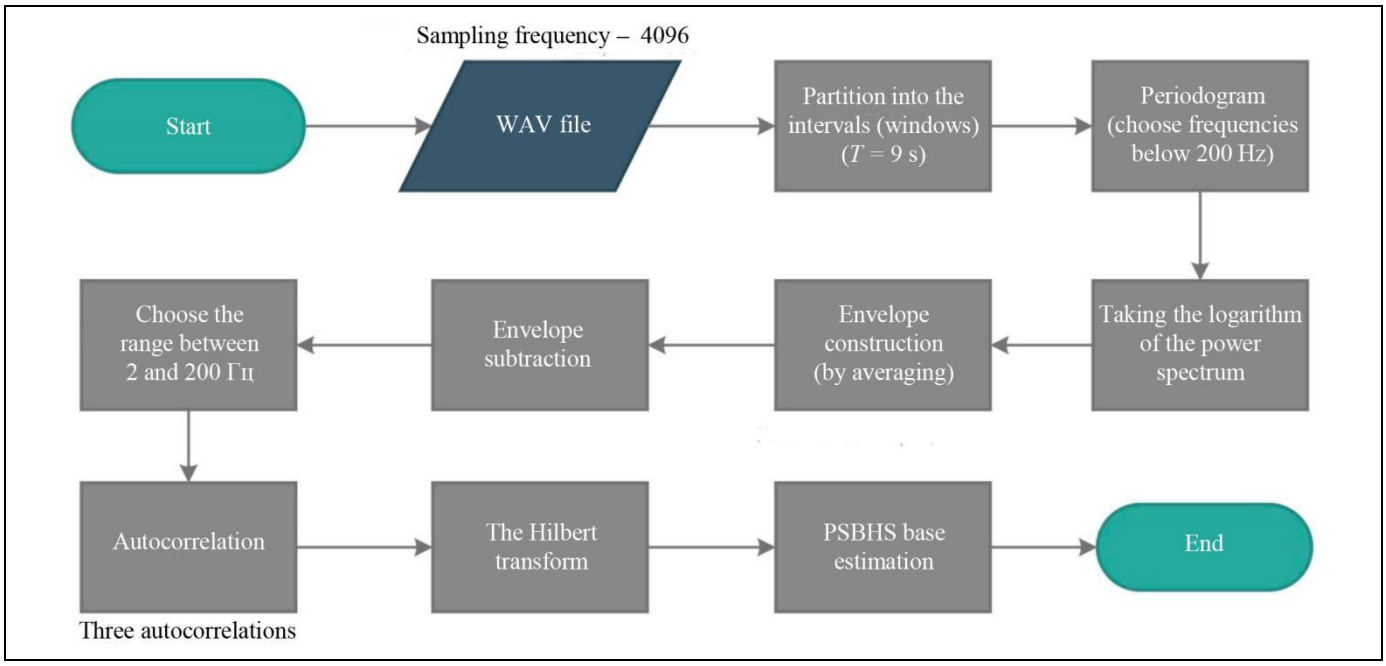

Fig. 1. The flowchart of the PSBHS base estimation algorithm.

Taking the logarithm of the periodogram

Taking the logarithm of the power spectrum of the signal eliminates sharp jumps associated with harmonics of the PSBHS, and the envelope can be separated by smoothing:

$$S_j^{(0)} = \ln |X_j|^2 .$$

Iterative spectrum smoothing

The spectrum envelope is constructed by averaging over three points in M iterations:

$$S_j^{(m)} = \frac{S_{j-1}^{(m-1)} + S_j^{(m-1)} + S_{j+1}^{(m-1)}}{3}, \qquad (1)$$

where j is the element number in the spectral series $\left(N\frac{f_{\min}}{f_s} < j < N\frac{f_{\max}}{f_s} \right)$; m is the iteration number $(m \in [1, M])$; M is the total number of iterations.

We clarify that the spectrum limits make some corrections to the expression (1):

$$S_j^{(m)} = \begin{cases} S_j^{(m-1)} \\ \quad \text{(for boundary points)} \\ \dfrac{S_{j-1}^{(m-1)} + S_j^{(m-1)} + S_{j+1}^{(m-1)}}{3} \\ \quad \text{(for inner points).} \end{cases} \qquad (2)$$

Definition 1. The iterative smoothing algorithm (2) of the series will be called the three-point smoothing of order M. ♦

The corrections in the final three-point smoothing algorithm (2) cover the effect of boundary points and insignificantly contribute to the spectrum smoothing results under $M \ll N$. Therefore, to simplify the proofs, the mathematical calculations below use the expression (1).

Note that M iterations in (1) are almost equivalent to weighting the original series elements by the Gaussian function with a standard deviation of $\sigma \approx \sqrt{\frac{2M}{3}} \approx 0.820467\sqrt{M}$. Examples for $M = 5$ and $M = 50$ are given in Fig. 2. (Also, see an example for M = 1,..., 4 in the table below.)

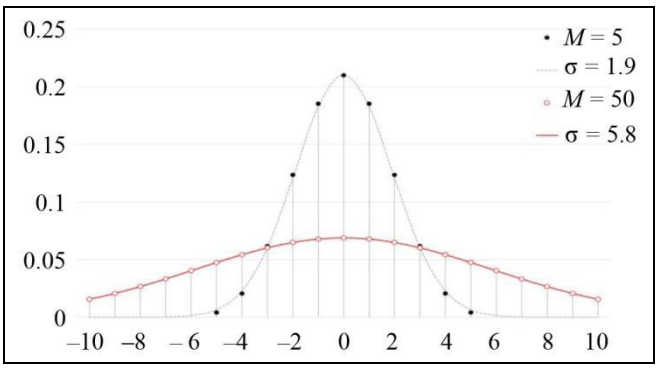

Fig. 2. The corresponding weights of the series elements obtained by formula (1) for M = 5 and M = 50 and the Gaussian function ($\sigma = \sqrt{2M/3}$): a visual comparison. The abscissa axis shows the deviations of the element number of the original series from j when estimating the weights.

Let us formulate the following result.

Lemma 1 (weight calculation). The M *successive iterations* (1) *generate the weights that can be approximately estimated using the Gaussian function with the parameter* $\sigma = \sqrt{\frac{2M}{3}}$ $(M >> 1)$:

$$S_j^{(M)} = \sum_{k=-M}^{M} \left[\frac{S_{j+k}^{(0)}}{\sqrt{4\pi M/3}} \exp\left(-\frac{k^2}{4M/3} \right) \right]. \qquad (3)$$

The weights of the three-point smoothing algorithm depending on the iteration number

m	$j-4$	$j-3$	$j-2$	$j-1$	j	$j+1$	$j+2$	$j+3$	$j+4$
1				$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$			
2			$\frac{1}{9}$	$\frac{2}{9}$	$\frac{3}{9}$	$\frac{2}{9}$	$\frac{1}{9}$		
3		$\frac{1}{27}$	$\frac{3}{27}$	$\frac{6}{27}$	$\frac{7}{27}$	$\frac{6}{27}$	$\frac{3}{27}$	$\frac{1}{27}$	
4	$\frac{1}{81}$	$\frac{4}{81}$	$\frac{10}{81}$	$\frac{16}{81}$	$\frac{19}{81}$	$\frac{16}{81}$	$\frac{10}{81}$	$\frac{4}{81}$	$\frac{1}{81}$
					...				

P r o o f. The fact that the weights obtained through successive iterations can be estimated by the Gaussian function is established technically. We show that in M iterations, the weights with the three-point averaging (1) are approximated by the Gaussian function with the parameter $\sigma = \sqrt{\frac{2M}{3}}$.

Let M be sufficiently large and consider iteration $M+1$; given that the weights in the three-point averaging algorithm at step M are approximated by the Gaussian function with the parameter $\sigma_M^2 = \alpha M$, for $j \ll M$ we have

$$\frac{1}{3\sigma\sqrt{2\pi}}\left(e^{-\frac{(j-1)^2}{2\sigma^2}} + e^{-\frac{(j)^2}{2\sigma^2}} + e^{-\frac{(j+1)^2}{2\sigma^2}}\right)$$

$$\approx \frac{1}{3\sigma\sqrt{2\pi}}\left(1-\frac{(j-1)^2}{2\sigma^2}+1-\frac{(j)^2}{2\sigma^2}+1-\frac{(j+1)^2}{2\sigma^2}\right)$$

$$= \frac{1}{\sigma\sqrt{2\pi}}\left(1-\frac{1}{2\sigma^2}\left(j^2+\frac{2}{3}\right)\right) \approx \frac{1}{\sigma\sqrt{2\pi}}\,\frac{1}{\left(1+\frac{1}{3\sigma^2}\right)}\,e^{-\frac{j^2}{2\sigma^2}}.$$

At step $M+1$, the coefficient at the exponent must be determined by the new parameter $\sigma_{M+1}^2 = \alpha(M+1)$. Therefore,

$$\sigma_{M+1}^2 = \alpha(M+1) = \alpha M + \alpha \approx \alpha M\left(1+\frac{1}{3\alpha M}\right)^2 \approx \alpha M + \frac{2}{3}.$$

It immediately follows that $\alpha = \frac{2}{3}$ and, accordingly, $\sigma = \sqrt{\frac{2M}{3}}$. ♦

We proceed to the next results.

Lemma 2 (spectrum smoothing by convolution). *The smoothing result with M successive iterations* (1) *applied to the logarithm of the power spectrum of the signal in a time window T can be approximately estimated by the convolution of this logarithm and the Gaussian function with the parameter* $\sigma = \frac{1}{T}\sqrt{\frac{2M}{3}}$ ($M >> 1$):

$$S_j^{(M)} = \sum_{k=-M}^{M}\left[S_{j+k}^{(0)}\sqrt{\frac{3}{\pi}}M^{-\frac{1}{2}}\exp\left(-\frac{3k^2T^2}{4M}\right)\right]. \quad (4)$$

The alternative form is

$$S^{(M)}(f) = S^{(0)}(f) * N(f,\sigma). \quad (5)$$

P r o o f. When increasing the time window size T, the density of the spectral lines grows whereas the standard deviation (the parameter of the Gaussian function used to estimate the weights, see formula (3) in Lemma 1) is reduced proportionally to T. Observing the normalization condition of the Gaussian function, we obtain the desired expression (4). ♦

Lemma 3 (spectrum smoothing by the cepstrum and Gaussian function). *The smoothing result with M successive iterations* (1) *applied to the logarithm of the power spectrum of the signal in a time window T can be approximately estimated by the Fourier transform of the product of its cepstrum, $K(\tau) = F^{-1}\left[S^{(0)}(f)\right]$, and the Gaussian function with the parameter* $\sigma^* = T\sqrt{\frac{3}{2M}}$ ($M >> 1$):

$$S^{(M)} = F\left[K(\tau)\cdot N(\tau,\sigma^*)\right].$$

P r o o f. Note that the expression (5) is the convolution of the logarithm of the signal power spectrum and the Gaussian function. By the well-known formula for the Fourier transform of the product of two functions ($F[f\cdot g] = F[f] * F[g]$), the above expression is nothing but the Fourier transform of the product of the inverse Fourier transform of the logarithm of the power spectrum (which is the cepstrum) and the inverse Fourier transform of the Gaussian function in the frequency domain with the parameter $\sigma = \frac{1}{T}\sqrt{\frac{2M}{3}}$ (which is also the Gaussian function in the time domain with the parameter $\sigma^* = T\sqrt{\frac{3}{2M}}$). Also, see an example in Fig. 3a. ♦

Lemma 4 (line spectrum extraction). *The line spectrum $\hat{S}(f) = S^{(0)}(f) - S^{(M)}(f)$ of the signal (defined as its pseudospectrum above) can be approximately estimated in the following ways:*

– via the convolution of the logarithm of its power spectrum with the difference between the delta function $\delta(f)$ and the Gaussian function:

$$\hat{S}(f) = S^{(0)}(f) * \left(\delta(f) - N(f,\sigma)\right),$$

where $\delta(f) \approx N(f,\sigma_\delta)$ and $\sigma_\delta \ll \sigma$, with σ_δ determining the accuracy of the line spectrum and σ the degree of smoothness of the spectrum envelope.

– via the Fourier transform of the product of the cepstrum of the original signal and the difference between the constant and the Gaussian function:

$$\hat{S}(f) = F\left[K(\tau)\cdot\left(\Psi - N(\tau,\sigma^*)\right)\right],$$

where the cepstrum is $K(\tau) = F^{-1}\left[S^{(0)}(f)\right]$ and $\Psi = F^{-1}\left[\delta(f)\right]$ ($\delta(f) \approx N(f,\sigma^\delta)$).

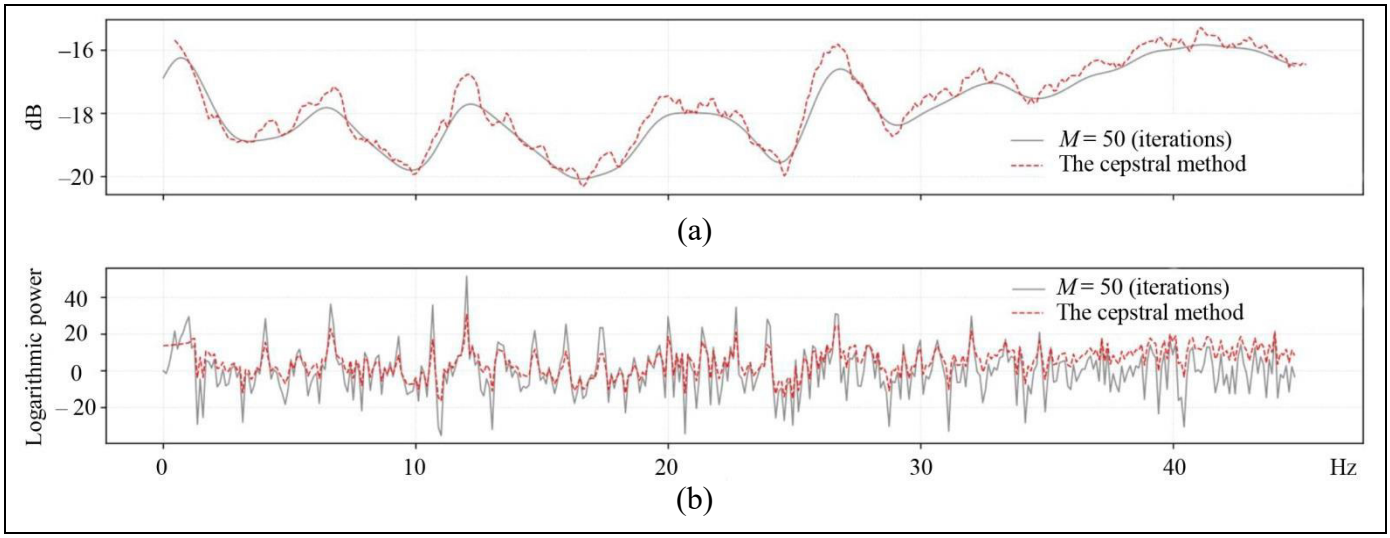

Fig. 3: (a) spectrum envelope estimation (Lemma 3) **and (b) centered logarithmic power spectrum** (Lemma 4). The estimates are obtained by two methods: iterative spectrum smoothing with envelope subtraction (solid lines) and envelope estimation with centered spectrum estimation via the cepstral method (dashed lines).

P r o o f. The desired result follows from Lemmas 2 and 3, the additivity of the Fourier transform, and the Fourier transform formula for the delta function:

$$\hat{S}(f) = S^{(0)}(f) - S^{(M)}(f) =$$

$$= S^{(0)}(f) - S^{(0)}(f) * N(f, \sigma)$$

$$= S^{(0)}(f) * (\delta(f) - N(f, \sigma)).$$

In turn, applying the Fourier transform formula for the product of two functions, we obtain

$$\hat{S}(f) = S^{(0)}(f) * (\delta(f) - N(f, \sigma))$$

$$= F\left[F^{-1}[S^{(0)}(f)] \cdot F^{-1}[(\delta(f) - N(f, \sigma))]\right]$$

$$= F\left[K(\tau) \cdot \left(\Psi - N(\tau, \sigma^*)\right)\right].$$

In numerical calculations with the discrete Fourier transform, an estimate of the delta function can be obtained from the Gaussian function with a small parameter $\sigma_\delta \ll \sigma$, with an appropriate normalization condition, the value $\Psi = F^{-1}\left[\delta(\tau)\right]$; see an example in Fig. 3b. (By the normalization condition, the sum of all weights obtained by the Gaussian function approximation must be 1.) ♦

Recall that cepstrum-related methods consider functions that can be viewed as spectra of logarithmic spectra. The concept of a cepstrum was introduced in 1963.

A power cepstrum is defined as "the power spectrum of a logarithmic power spectrum" [12]. Power cepstrum was proposed as a more efficient alternative to the autocorrelation function in detecting echoes in signals.

As the corresponding function describes the spectrum of a spectrum by definition, following a terminological analogy with "spectrum," A.S. Gol'din gave this function the name "cepstrum."[1]

However, the most important feature of cepstrum is not its representation as the spectrum of a spectrum but the associated logarithmic transformation of the original spectrum and the spectrum processed further. Note that the autocorrelation function, determined from the eigen power spectrum by the inverse Fourier transform, can also be treated as the "spectrum of a spectrum." In essence, the currently used notion of a cepstrum defines a power cepstrum as "the inverse Fourier transform of a logarithmic power spectrum." The difference between this definition and the definition of an autocorrelation function is only the logarithmic transformation of the original spectrum.

The application of power cepstra to study PSBHSs is based on the former's capability to detect spectrum periodicities, such as a series of uniformly distributed harmonics. From the standpoint of this application, an important advantage of cepstra is related to their small dependence on the propagation paths of signals, including the paths from sources to measurement points.

Trend subtraction and frequency range narrowing

Further, to separate the line spectrum, containing important information about a PSBHS, we subtract its envelope from the logarithm of the signal power

[1] Similarly, the terms "*quefrency*," "*rahmonic*," "*lifter*," "*gamnitude*," and "*saphe*" evolved from analogy with the conventional terms "*frequency*," "*harmonic*," "*filter*," "*magnitude*", and "*phase*." The terms related to "lifter" ("liftering," "liftered," etc.), indicating the filtering process in the cepstral domain, are occasionally used in the literature as well.

spectrum. In this case, the desired centered spectral series is given by

$$\hat{S}_j = \hat{S}(f_j) = S^{(0)}(f_j) - S^{(M)}(f_j) = S_j^{(0)} - S_j^{(M)}.$$

Then, an appropriate frequency range $f_j \in [f_{\min}, f_{\max}]$ is determined. Accordingly, the element number j of these frequencies satisfies the two-sided inequality $Tf_{\min} < j < Tf_{\max}$.

Successive autocorrelations

We apply several successive autocorrelations to the spectrum (precisely, to the centered logarithm of the spectral power) to smooth it further, improve pattern identification, and extract useful information from the spectrum more accurately.

Each autocorrelation calculation averages the information, removing noise components and short-term fluctuations. Performing several successive autocorrelations enhances this effect. Noise components often have a short period or random nature, and autocorrelation helps isolate regular and repeating elements. If the signal spectrum has a complex structure with several periodic components (contains several harmonic series), performing several autocorrelations will identify these components more accurately. With each autocorrelation step of the spectrum, more stable harmonic series are separated, whereas unstable ones and various noise are smoothed out. Therefore, several successive autocorrelations allow extracting the most stable component of propeller noise, i.e., the PSBHS base.

When analyzing the PSBHS, several successive autocorrelation steps for the logarithm of the power spectrum better isolate the low-frequency component (shaft frequency) against the background of the higher-frequency component (blade frequency). Note that when autocorrelation is applied to the spectrum, the low and high frequencies seem to change their places: the low-frequency shaft rotation is manifested by rapid peaks whereas the blade frequency by rarer peaks; after each autocorrelation, the shaft frequency is manifested more and more strongly.

The repeated autocorrelation of the spectrum is calculated for the smoothed result from its first autocorrelation, which gives additional smoothing. Since the noise components in the first autocorrelation are attenuated, averaging them again in the second autocorrelation even more damps both random bursts and non-harmonic components. However, there is a natural limitation: a very large number of autocorrelations will eventually smooth even useful information about stable, regular patterns that have long-term structure associated with the PSBHS base.

We define the autocorrelation function of the centered logarithmic power spectrum $\hat{S}_j$ of order $p = 0$ as follows:

$$C^{(0)}(f_k) = C_k^{(0)} = \sum_{j=Tf_{\min}}^{Tf_{\max}-k} \hat{S}_j \hat{S}_{j+k}.$$

Next, we define $C_k^{(p)}$ as the autocorrelation function of the centered logarithmic power spectrum $\hat{S}_j$ of order p:

$$C^{(p)}(f_k) = C_k^{(p)} = \sum_{j=Tf_{\min}}^{Tf_{\max}-k} C_j^{(p-1)} C_{j+k}^{(p-1)},$$

where $p \in [1, P]$ and $P + 1$ is the number of successive autocorrelations.

Hilbert transform

Modern methods of analytic signal theory [14] serve to extract (demodulate) an instantaneous amplitude (envelope), instantaneous phase, and instantaneous frequency from an oscillatory process. To obtain these instantaneous functions, one transforms an original process $x(t)$, defined on some interval, into the conjugate process $\hat{x}(t)$ using the *integral Hilbert transform* [5]:

$$\hat{x}(t) = \mathcal{H}\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau.$$

The analytic signal can be written as

$$x_a(t) = x(t) + i\hat{x}(t).$$

As is easily verified, the function $\sin \omega_0 t$ represents the Hilbert transform of the function $\cos \omega_0 t$. Therefore, the analytic signal corresponding to $\cos \omega_0 t$ is

$$x_a(t) = \cos \omega_0 t + i \sin \omega_0 t = \exp(i\omega_0 t).$$

It seems convenient to write a general analytic signal in exponential form as

$$x_a(t) = |x_a(t)| \exp\left[i\Phi(t)\right],$$

where

$$\begin{aligned} |x_a(t)| &= \left[x^2(t) + \hat{x}^2(t)\right]^{1/2}, \\ \Phi(t) &= \operatorname{arctg}\left[\hat{x}(t) / x(t)\right]. \end{aligned} \tag{6}$$

Now, by letting $\Phi(t) = \omega_0 t + \varphi(t)$, we have

$$x_a(t) = |x_a(t)| \exp\left[i\varphi(t)\right] \exp(i\omega_0 t) = \theta(t) \exp(i\omega_0 t).$$

The complex envelope $\theta(t)$ is obtained by removing the complex factor associated with the carrier from the analytic signal:

$$\theta(t) = x_a(t) \exp(-i\omega_0 t) = |x_a(t)| \exp\left[i\varphi(t)\right].$$

If $\theta(t)$ is a narrowband function relative to $\omega_0/2\pi$, it will have properties intuitively associated with the concept of envelope.

For an original signal in the frequency domain, the physical meaning of the integral Hilbert transform is the phase shift of all spectral components of this signal by $\pi/2$. The double Hilbert transform leads to the original process but with the opposite sign: it shifts the original signal by π.

We apply the Hilbert transform to $C^{(P)}(f_k)$ to obtain the *analytic autocorrelation*

$$\tilde{C}^{(P)}(f_k) = C^{(P)}(f_k) + i\mathcal{H}\{C^{(P)}(f_k)\}.$$

Here, the cepstral phase is defined as the argument of the analytic autocorrelation:

$$\varphi_k = \arg\tilde{C}^{(P)}(f_k) = \arg\left(C^{(P)}(f_k) + i\mathcal{H}\{C^{(P)}(f_k)\}\right),$$

$$\varphi_k = \operatorname{arctg}\left(\frac{\mathcal{H}\{C^{(P)}(f_k)\}}{C^{(P)}(f_k)}\right). \qquad (7)$$

Dwelling on these formulas, we introduce the following.

Definition 2. The *cepstral phase* is the value given by the expression (7). ♦

Let f_H denote the frequency for which the cepstral phase is calculated. Such a designation emphasizes that the saphe is calculated via the Hilbert transform.

Lemma 5 (the harmonic series base and the Hilbert transform). *Consider a given signal $x(t)$ with a non-constant amplitude and a frequency slowly changing during the entire observation period T. Then the average period $\overline{T_0}$ of signal oscillations over a time t can be estimated as*

$$\overline{T_0} = 2\pi t\left[\operatorname{arctg}\left(\frac{\mathcal{H}\{x(t)\}}{x(t)}\right)\right]^{-1},\ 0 < t < T.$$

The proof is technical and can be easily derived from the expression (6).

This lemma leads to the following result. Let the signal $x(t)$ in Lemma 5 be the correlogram $C^{(P)}$ of the logarithmic power spectrum of a signal-noise mixture containing a signal (a harmonic series with a base f_0) and some noise with a sufficiently high SNR value. Then the harmonic series base can be estimated as

$$f_0 = 2\pi f_H\left[\operatorname{arctg}\left(\frac{\mathcal{H}\{C^{(P)}(f_H)\}}{C^{(P)}(f_H)}\right)\right]^{-1}, \qquad (8)$$

where $f_{\min} < f_H < f_{\max}$.

Note that the frequency f_H should be assigned depending on a particular task and conditions. For example, under strong noise, the recommendation is to choose this frequency in the range $0.1f_{\max} < f_H < 0.2f_{\max}$ (the correlogram "breaks" at the right end due to the large noise component). On the other hand, $f_H \approx f_{\max}$ is recommended when estimating the PSBHS base with high accuracy in weak noise conditions. As is easily shown, the PSBHS base estimate (8) represents the averaging of the frequency differences of the autocorrelation peaks; in turn, this determines the estimation accuracy $T^{-1}\frac{f_0}{f_H}$, constituting thousandths of a hertz in practice. (For example, for a time window of $T = 10$ s, the PSBHS base is $f_0 \approx 1$ Hz, and the search range of the harmonic series modes is limited by the frequency $f_{\max} = 100$ Hz.)

In general, we underline that the harmonic series base estimation method (8) gives less error with increasing SNR, and relevant estimates can be obtained only above a certain threshold ($\mathrm{SNR} > \mathrm{SNR}_0$). Below, we demonstrate how this threshold can be reduced.

Figure 4 presents the results of a numerical experiment: $\Omega = (1 + \exp(-0.4364\cdot\mathrm{SNR} - 0.8545))^{-1}$ (the solid line, without smoothing) and $\Omega = (1 + \exp(-0.4212\cdot\mathrm{SNR} - 2.3633))^{-1}$ (the dashed line). Each point was constructed from 500 cases. Each case was obtained by generating a time series of length $N = 4096$ via summing a signal (a harmonic series) and white noise with the following parameters: $T = 1$ s (time window), $d = 15$ (the number of harmonic series samples, with the same power level), $f_0 = 60$ Hz (the frequency difference between samples), and SNR = -12 dB,…, 7 dB.

As discovered, the pre-smoothing of the pseudospectrum allows reducing the SNR thresholds from 0 to -5 dB. In Fig. 4a, the optimum is achieved by 40 iterations of the three-point smoothing of the pseudospectrum; for this experiment, it is equivalent to the convolution of the pseudospectrum and the Gaussian function ($\sigma \approx 5$ Hz). Figure 4b shows a comparative analysis of the error-free harmonic series base estimates without smoothing (the lower curve) and with smoothing in 40 iterations (the upper curve). As a rule, the pseudospectrum in real sound recordings is more or less "blurred," and additional smoothing is not always necessary.

Next, Fig. 5 provides the histograms of the harmonic series base estimates for different SNR values of the harmonic series and white noise. One

thousand (1000) signal-noise mixtures were generated to construct each histogram. The frequency difference between harmonics was set equal to 60 Hz. No pre-smoothing of the pseudospectrum was performed.

Figure 6 shows the histograms of the harmonic series base estimates for SNR = –3 dB. Five thousand (5000) signal-noise mixtures were generated to construct each histogram. The frequency difference

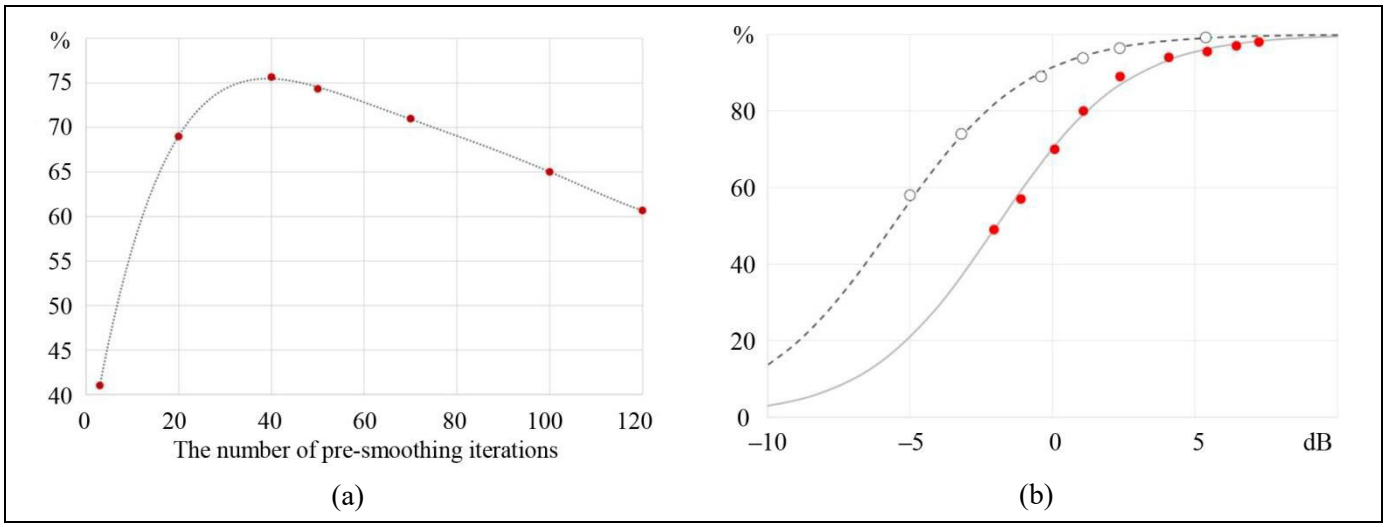

Fig. 4: (a) the share of error-free PSBHS base estimates Ω (SNR= –3 dB) depending on the pre-smoothing of the harmonic pseudospectrum and (b) a visual comparison of the dependences of error-free PSBHS base estimates without spectrum smoothing and with spectrum smoothing in 40 iterations.

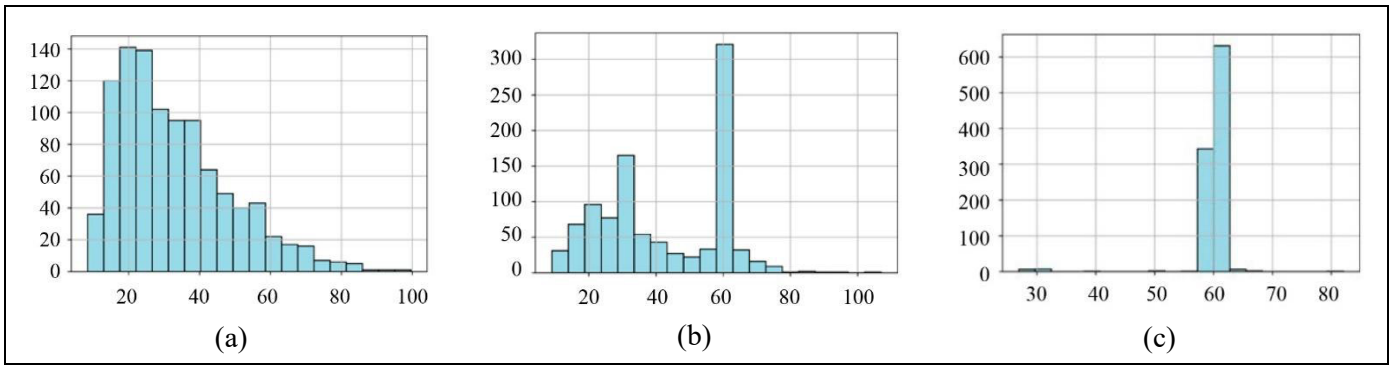

Fig. 5. The histograms of PSBHS bases for different SNR values (without smoothing): (a) SNR = – 12 dB, Ω = 1%, (b) SNR = – 3 dB, Ω = 40%, and (c) SNR = + 7 dB, Ω = 98%.

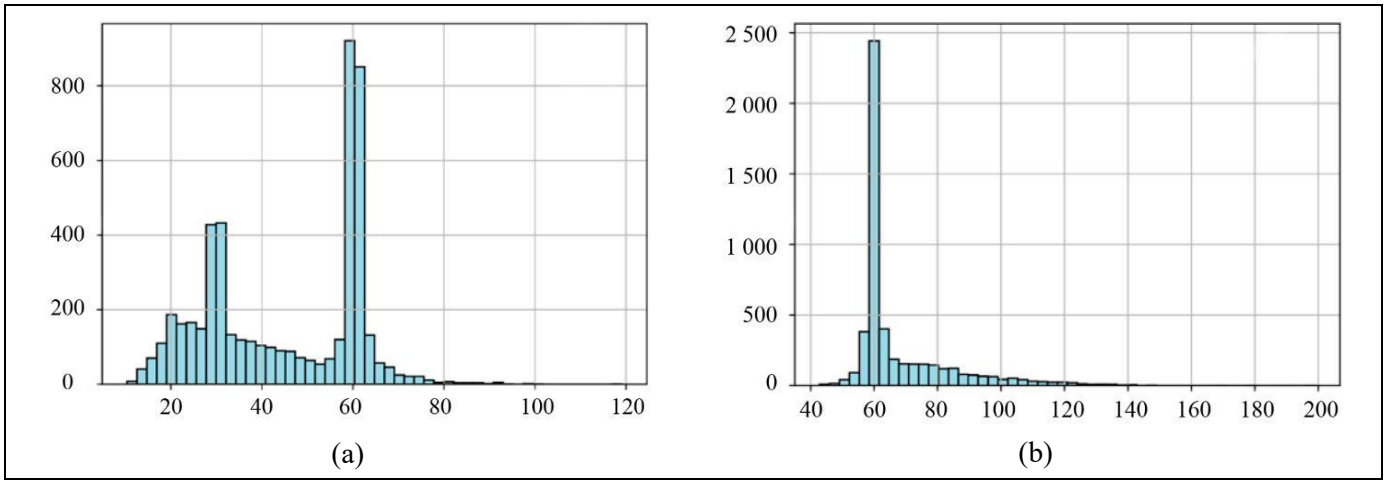

Fig. 6. The histograms of PSBHS bases for different smoothing methods, SNR = – 3 dB: (a) without smoothing, Ω = 40% and (b) with smoothing in 100 iterations, Ω = 65%.

between harmonics was set equal to 60 Hz. Figure 6a corresponds to the solution without smoothing; Fig. 6b, to the solution with the pre-smoothing (100 iterations over three points) of the spectrum of the signal-noise mixture.

2. THE PSBHS BASE ESTIMATION ALGORITHM: A BRIEF DESCRIPTION OF KEY STEPS

The PSBHS base estimation algorithm was tested on the real sound recordings of sea vessel noise, and the effectiveness of the iterative three-point smoothing method was validated for constructing the spectrum envelope. Also, the effectiveness of the PSBHS base estimation method with the convolution of the logarithm of the signal power spectrum and the difference of two Gaussian functions was validated.

Summarizing the outcomes, we write the main steps of the PSBHS base estimation algorithm using convolution:

1. Estimating the periodogram of the signal x_k of length N in a time window of size T:

$$X_j = \sum_{k=0}^{N-1} x_k e^{-i\frac{2\pi}{N}jk}.$$

2. Taking the logarithm of the power spectrum of this signal:

$$S_j^{(0)} = \ln |X_j|^2.$$

3. Extracting the line (centered) spectrum of the signal by convolution (M_0 blurring iterations for the centered spectrum and M spectrum smoothing iterations for envelope estimation, where $M_0 \ll M$ and $Tf_{\min} < j < Tf_{\max}$):

$$\hat{S}_k = S_j^{(0)} * \left[\frac{1}{\sqrt{M_0}} \exp\left(-\frac{j^2T^2}{M_0} \right) - \frac{1}{\sqrt{M}} \exp\left(-\frac{j^2T^2}{M} \right) \right].$$

Due to boundary effects, the mean value of the series $\hat{S}_k$ slightly differs from zero, and the mean value is subtracted from the obtained series before constructing the autocorrelation function.

4. Estimating the autocorrelation function of the centered spectrum:

$$C_k^{(0)} = \sum_{j=Tf_{\min}}^{Tf_{\max}-k} \hat{S}_j \hat{S}_{j+k},$$

$$C_k^{(p)} = \sum_{j=Tf_{\min}}^{Tf_{\max}-k} C_j^{(p-1)} C_{j+k}^{(p-1)},$$

where $p \in [1, P]$ and $P+1$ is the number of successive autocorrelations.

5. Estimating the PSBHS base:

$$f_0 = 2\pi f_H \left[\operatorname{arctg}\left(\frac{\mathcal{H}\{C^{(P)}(f_H)\}}{C^{(P)}(f_H)} \right) \right]^{-1},$$

where $f_{\min} < f_H < f_{\max}$.

3. OPTIMAL PARAMETERS OF THE PSBHS BASE ESTIMATION ALGORITHM

Consider a set of PSBHS bases obtained by processing the sound recordings of sea vessel noises using the above algorithm:

$$F_0^{(r)} = \{f_0^{(r)}\}_{i=1}^{Z_r},$$

where r is the WAV file number; $F_0^{(r)}$ is the set of PSBHS base estimates in each time window for the rth WAV file; Z_r is the total number of time windows in the rth WAV file; finally, Z_rT is the size of the sea vessel sound recording.

Among all WAV files, we take those satisfying the inequality $\sigma^{(r)} < \alpha T^{-1}$ ($\alpha \approx 10$), where $\sigma^{(r)}$ is the standard deviation of the PSBHS base estimates for the rth recording (i.e., the ones with an insignificant change of the PSBHS base estimates during the observation time). Erroneous estimates are such that

$$|f_0^{(r)} - \mu^{(r)}| > \alpha T^{-1},$$

where $\mu^{(r)}$ denotes the mean value of the PSBHS bases for the rth Wav file.

An optimal tuple of the parameters ($T, M, f_{\min}, f_{\max}, P,$ and f_H) of the PSBHS base estimation algorithm is selected according to the criterion

$$\sum_r \omega^{(r)} \underset{T,M,f_{\min},f_{\max},P,f_H}{\rightarrow} \min,$$

where $\omega^{(r)}$ is the number of errors for the rth WAV file.

The following optimal parameters of the algorithm were obtained on real data on cargo and passenger sea vessels in terms of this criterion:

- $T = 10$ s (the time window size),
- $M = 100$ (the number of smoothing iterations given $M_0 = 10$ pre-smoothing iterations to blur the pseudospectrum),
- ($f_{\min} = 2, f_{\max} = 200$) Hz (the autocorrelation interval),

• $P = 2$ ($P + 1 = 3$ is the number of successive autocorrelations),

• $f_H = 25$ Hz (the frequency for estimating the "cepstral phase").

CONCLUSIONS

This paper has presented new approaches to estimating the fundamental frequency of a harmonic series from a single time window in high noise conditions. The algorithm for determining the base of a propeller shaft–blade harmonic series has demonstrated stable performance under a signal-to-noise ratio exceeding –5 dB. As expected, in future works, the method will be improved to get a refined base estimate by continuous signal processing over multiple time windows as well as by using information characteristics [15]. As shown above, an effective solution can be obtained via a set of measures and correctly chosen values of the algorithm parameters.

In the course of this study, numerical experiments have been conducted on the sound recordings of sea vessel noises (in total, over 400 sound recordings of passenger ships, container ships, tankers, and tugs). According to the experiment results, the methods are effective.

In several cases, when the sea vessel noise is generated by several propellers, beats have been observed. Consequently, it is difficult to determine the shaft frequency from separate time windows (in these windows, the signals at the shaft frequencies arrive in counter-phase). Here, one faces the validity problem of PSBHS base estimation, which is also the objective of subsequent research.

Also, this work is part of a group of approaches to investigating the noise spectra of sea vessels. Note that the identification and utilization of additional information criteria in other spectrum ranges of sea vessel noises significantly improves the estimation quality of the PSBHS base as well.

Acknowledgments. *This work was supported by the Russian Science Foundation, project no. 23-19-00134.*

REFERENCES

1. Urick, R.J., *Principles of Underwater Sound*, McGraw-Hill, 1975.
2. Evtyutov, A.P. and Mit'ko, V.B., *Primery inzhenernykh raschetov v gidroakustike* (Examples of Engineering Calculations in Hydroacoustics), Leningrad: Sudostroenie, 1981. (In Russian.)
3. Kudryavtsev, A.A., Luginets, K.P., and Mashoshin, A.I., Amplitude Modulation of Underwater Noise Produced by Seagoing Vessels, *Acoustical Physics*, 2003, vol. 49, no. 2, pp. 184–188.
4. Malyi, V.V., Saprykin, V.A., Rokhmaniiko, A.Yu., et al., Patent RU 2464588 C1, *Byull. Izobr*., 2012, no. 29.
5. Burdik, V.S., *Analiz gidroakusticheskikh sistem* (Analysis of Hydroacoustical Systems), Leningrad: Sudostroenie, 1988. (In Russian.)
6. Evtyutov, A.P., Kolesnikov, A.E., Korepin, E.A., et al., *Spravochnik po gidroakustike* (Hydroacoustics Handbook), Leningrad: Sudostroenie, 1988. (In Russian.)
7. Astfalck, L.C., Sykulski, A.M., and Cripps, E.J., Debiasing Welch's Method for Spectral Density Estimation, *arXiv:2312.13643*, 2023, pp. 1–17. DOI: https://doi.org/10.48550/arXiv.2312.13643
8. Knichuta, E.V., Pachotin, V.A., Budnik, S.S., and Rzhanov, A.A., The Signal Parameter Estimation Task Solution in the Frequency Domain for the Pulse Signals, *Journal of the Russian Universities. Radioelectronics*, 2005, no. 2, pp. 19–29. (In Russian.)
9. Marapulets, Yu.V., Adaptive Spectral Analysis of the Amplitude Envelope of Marine Vessel Noise, *Bulletin of the Kamchatka State Technical University*, 2003, no. 2, pp. 52–60. (In Russian.)
10. Liu, D., Yang, H., Hou, W., and Wang, B., A Novel Underwater Acoustic Target Recognition Method Based on MFCC and RACNN, *Sensors*, 2024, vol. 24, no. 1, art. no. 273.
11. Doan, V.-S., Huynh-The, T., and Kim, D.-S., Underwater Acoustic Target Classification Based on Dense Convolutional Neural Network, *IEEE Geoscience and Remote Sensing Letters*, 2022, vol. 19, art no. 15009052020, pp. 1–5. DOI: 10.1109/LGRS.2020.3029584
12. Randall, R.B., *Frequency Analysis*, Naerum: Brüel & Kjaer, 1987.
13. Gol'din, A.S., *Vibratsiya rotornykh mashin* (Vibration of Rotary Machines), Moscow: Mashinostroenie, 2000. (In Russian.)
14. Genkin, M.D. and Sokolova, A.G., *Vibroakusticheskaya diagnostika mashin i mekhanizmov* (Vibroacoustical Diagnosis of Machines and Mechanisms), Moscow: Mashinostroenie, 1987. (In Russian.)
15. Galyaev, A.A., Babikov, V.G., Lysenko, P.V., and Berlin, L.M., A New Spectral Measure of Complexity and Its Capabilities for Detecting Signals in Noise, *Doklady Mathematics*, 2024, vol. 110, pp. 361–368.

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

Received April 15, 2025, and revised June 5, 2025.
Accepted June 25, 2025

Author information

Babikov, Oleg Vladimirovich. Mathematician, Trapeznikov Institute of Control Sciences, Russian Academy of Science, Moscow, Russia
✉ babikov.ov@phystech.edu

Babikov, Vladimir Georgievich. Cand. Sci. (Phys.–Math.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ babikov@ipu.ru

Cite this paper

Babikov, O.V., and Babikov, V.G., Estimating the Fundamental Frequency of a Propeller Shaft–Blade Harmonic Series Using the Hilbert Transform and Autocorrelation. *Control Sciences* **3,** 12–23 (2025).

Original Russian Text © Babikov, O.V., Babikov, V.G., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 58–73.

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

MODELS OF FATIGUE AND REST IN LEARNING. PART I: Extension of the General Iterative Learning Model

D. I. Grebenkov*, A. A. Kozlova**, D. V. Lemtyuzhnikova***, and D. A. Novikov****

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

*✉ grebenkov-d-i@mail.ru, **✉ sankamoro@mail.ru, ***✉ darabbt@gmail.com, ****✉ novikov@ipu.ru

Abstract. In this paper, classical iterative learning models are extended by including the factors of fatigue and rest. The existing approaches to model fatigue and rest from various fields—education, production, sports, and medicine—are analyzed, and the need to include these factors in the models is justified accordingly. Mathematical models are proposed to describe learning level dynamics during rest periods, considering the reduced efficiency of acquiring skills due to fatigue. Ten models of growing complexity are studied: from simple models without any fatigue effects to complex ones with the probabilities of skill acquisition and forgetting depending on time and rest periods. As is shown, the breaks of optimal duration allow improving the terminal learning level. In the model with fatigue, rest, and no forgetting, the optimal time to start rest is independent of the probability of skill acquisition as a function of time and lies at the middle of the experience acquisition interval. The models proposed are intended for predicting performance and optimizing training programs, production processes, and training cycles. This study emphasizes the need to consider biological and cognitive constraints when designing adaptive learning systems.

Keywords: experience, iterative learning, learning curve, mathematical modeling, fatigue, rest, skill acquisition, forgetting.

INTRODUCTION

Learning, the process and result of acquiring individual experience [1], underlies the adaptation of living and nonliving systems to changing conditions. In the context of mathematical modeling, learning is understood as a process during which a (biological, technical, or abstract-logical) system optimizes its actions to achieve a given goal. Of particular interest is *iterative learning* (IL), a type of learning based on the system's repetition of actions (trials and errors) to achieve a fixed goal under constant external conditions [2]. This process is the foundation for developing skills in humans, conditioned reflexes in animals, and adaptation algorithms in robotics and artificial intelligence.

Mathematical models of IL describe a sequence of learning levels (the so-called *learning curves*, LCs) [3] through systems of equations, graphs, or algorithms to reveal universal regularities. For example, an LC is the probability of mastering an activity component depending on time or the number of repetitions (iterations).

Based on experimental data, for most systems from humans to neural networks, these curves have a slowly asymptotic character: the rate of performance improvement decreases with time, and the curve tends to some limit [2]. Such behavior of LCs is often approximated by exponential functions, which confirms the "saturation" of the learning process.

Despite the universality of slowly asymptotic regularities, classical learning models neglect such factors as fatigue and rest [3]. To understand these limitations and justify model extensions, it is necessary to analyze the existing approaches to model learning in different contexts, from cognitive processes in education to motor skills in sports and manufacturing.

1. APPROACHES TO MODEL LEARNING AND REST: A REVIEW

Learning plays a key role in almost any human activity, from education to industrial manufacturing and from sports to robotic surgery: understanding the regularities of learning allows optimizing processes and

reducing risks. (An *activity* is a human's interaction with the surrounding world in which the former represents a *subject* with a purposeful influence on an *object* [4].) As demonstrated by current research, the modeling of learning requires considering not only individual characteristics but also contextual specifics, whether it is cognitive load in education or physical fatigue in sports or (and) manufacturing.

In **education**, learning is closely related to the management of cognitive resources. The effect of cognitive fatigue on standardized test scores was studied and a clear relationship between the time of day and school performance was revealed [5]. According to the analysis of two million tests of Danish schoolchildren presented therein, the scores deteriorate by 0.9% of the standard deviation every hour after the start of school. This decline is due to the accumulation of fatigue, which impairs concentration and reduces the ability to solve complex tasks. However, the introduction of 20–30 min breaks compensates for this effect and even improves the results by 1.7%, especially for low-performing schoolchildren. For those with learning difficulties, the authors recommended small breaks of 5–10 mins, which decrease cognitive overload and improve learning by 12% [5]. These data emphasize that the educational process should be designed considering biological rhythms. Interestingly, similar regularities are observed in sports and manufacturing, where short pauses help to maintain productivity [6].

The learning process of a second language was modeled in [7]; as shown, progress in language skills is not a smooth growth but a series of sharp transitions between steady states. For example, an optimal ratio of new to familiar material (approximately 30% to 70%) triggers a noticeable progress in mastering, whereas an imbalance makes the learning process "stuck" on a "plateau." This phenomenon is well known to educators: facing overly complex tasks, students often lose motivation, and routine exercises without novel elements also inhibit development [8]. Such conclusions agree with corpus studies, where the distribution of errors in the speech of language learners corresponds to power laws [9, 10].

In **professional activities**, especially in high-tech industries, learning is often associated with the problem of fatigue. The cognitive effects of fatigue in employees of engineering companies were studied in [11]; according to the results, after 8-h work, the accuracy of information retrieval in memory drops by 6%, and the time of switching between tasks increases by 120 ms. Moreover, 22% of employees unconsciously switch to less efficient strategies for solving their tasks, thereby saving cognitive resources. These findings are especially relevant for spheres with critical consequences, from air traffic control to medical diagnosis. According to the recommendations in [11], adaptive work schedules should be adopted by alternating periods of intensive workload with rest phases; in addition, simulators should be used for practicing skills under fatigue simulation conditions.

Mastering complex medical technologies such as robotic surgery is of particular interest. Learning curves for nurses working in operating rooms were analyzed in [12]; based on the results, achieving mastery requires an average of 8–11 procedures. However, individual differences are huge: some nurses needed only three operations while others up to 11. As it turned out, the most challenging steps are planning the placement of spinal screws and controlling the robotic arm, where errors are often associated with cognitive overload. To reduce training time, the authors of [12] proposed simulation training on 3D models of the spine, where key skills can be practiced without risk to patients. This approach reduces the level of stress and fatigue by 35%, directly affecting the speed of mastering the technology.

In **manufacturing**, the modeling of learning has become the basis for process optimization. The classical Wright's power curve [13], describing the decrease in task completion time with experience, is still the starting point. However, many tasks combining cognitive and motor components require more sophisticated approaches. The model presented in [14] divides task completion time into two components: cognitive (planning and decision-making) and motor (physical actions). For example, when assembling a mechanical device, the initial stages require active thinking and their time decreases faster, while motor skills improve gradually. Compared to traditional approaches, this model is 23% more accurate in predicting performance in the early stages of learning, which is crucial for production costing and planning.

The best practices for the modeling of learning in manufacturing were systematized in the meta-analysis [15] covering 115 datasets. For tasks involving time or cost reduction, the S-shaped and three-parameter hyperbolic models were found to be the most effective. They consider the "plateau" phase when further improvement becomes minimal. In the context of productivity growth, the leader is the three-parameter exponential model, which well describes processes with the saturation effect. The integration of these models into production management systems reduces inventory planning errors by 18%, preventing both excess inventory and downtime.

In **sports and rehabilitation**, learning is inseparable from fatigue management. According to the study [16], fatigue increases the risk of injury by 40% due to a decrease in proprioception (the ability to orient the

body in space). For example, in soccer, 67% of injuries occur in the last 20 minutes of a match when concentration drops and muscles lose elasticity. Objective methods (heart rate variability analysis) are combined with subjective questionnaires to monitor the condition of athletes. Heart rate variability analysis can predict readiness to load: a 15% decrease in heart rate variability correlates with an increased risk of injury. This data is used to personalize training by reducing intensity at the first signs of overfatigue.

The problem of rehabilitation after injuries or strokes deserves special attention. In the model proposed in [17], fatigue is considered as a balance between two components: the objective decrease in performance and the subjective perception of effort. For example, stroke patients often experience muscle weakness even with minimal load. Traditional training aimed at increasing strength can aggravate the condition when neglecting individual fatigue thresholds. The authors of [17] suggested using sensory feedback (e.g., real-time visualization of muscle activity) to help patients learn to distribute effort while avoiding overstrain.

In some works on **robotic systems and industrial design,** the learning process is described using biomechanical analysis. For example, differential equations of muscle fatigue dynamics were presented in [6]. Recovery from a 5 kg muscle load requires an average of 2.4 mins of rest; this data is used to design workstations on conveyors. With simulators integrating such models, ergonomics can be tested before production starts. In the aircraft industry, this approach reduced cumulative fatigue by 22% by optimizing tool angles and work surface heights.

Thus, modern research into the learning process demonstrates the absence of universal learning models with fatigue effects: each context requires consideration of its specifics. In education, the focus is now shifting to the management of cognitive load and periods; in manufacturing, to the separation of cognitive and motor components; in sports, to the balance between fatigue and recovery. The above approaches indicate an important role of fatigue and rest in experience acquisition dynamics. Hence, it is necessary to extend the existing classical learning models by including these factors. Let us take the model [3] as a basis for further extension.

2. FATIGUE AND REST IN EXPERIENCE ACQUISITION MODELS

Consider the following learning models. Let a learner (further called the agent) gain individual experience, i.e., master some type of activity through successive trials (iterative learning). Assume that the experience in each period is characterized by two possible states: "formed" or "not formed." In each period $t = 1, 2, ...$, if the experience has not been gained, experience acquisition occurs with a probability $0 \le w(t) < 1$, generally depending on time. In parallel, the experience is forgotten with a probability $0 \le u(t) < 1$, also generally depending on time. The value $q(t)$ of the agent's individual experience criterion belongs to [0, 1]. The value $q(t)$ is the probability that in period t the agent's experience will be gained and not forgotten. Both the dependence of the probability of experience acquisition on time (a decreasing function) and the dependence of the probability of forgetting on time (an increasing function) can reflect the effects of fatigue, exhaustion, etc. during education and (or) a productive activity.

Consider several learning models covering such processes as mastering, forgetting, fatigue, and rest, sequentially in ascending order of their complexity. The presence of certain processes in the models is reflected in the table below.

The processes and parameters of experience acquisition models

Models	Mastering	Forgetting	Fatigue	Rest
Model 1	Time-invariant	-	-	-
Model 2	-	Time-invariant	-	-
Model 3	Time-invariant	Time-invariant	-	-
Model 4	Time-varying	-	+	-
Model 5	Time-varying	Time-invariant	+	-
Model 6	Time-invariant	Time-varying	+	-
Model 7	Time-varying	-	+	+
Model 8	Time-varying	Time-invariant, considered only on rest intervals	+	+
Model 9	Time-varying	Time-invariant	+	+
Model 10	Time-invariant	Time-varying	+	+

Model 1: In the basic (simplest) learning model [3], there is no forgetting and the probability of expe-

rience acquisition w does not depend on time. Here, the learning level dynamics are described by

$$q(t+1)=q(t)+(1-q(t))w,\ t=0,\ 1,\ 2,\ldots, \qquad (1)$$

with a known initial value $q(0)$.

In continuous time, the difference equation (1) turns into the differential equation

$$\dot{q}=(1-q)w,$$

and its solution has the form

$$q(t)=1-(1-q(0))e^{-wt}. \qquad (2)$$

The learning curve (2) is non-decreasing and asymptotically tends to unity. Figure 1 shows the graph of the function (2) with $q(0)=0$.

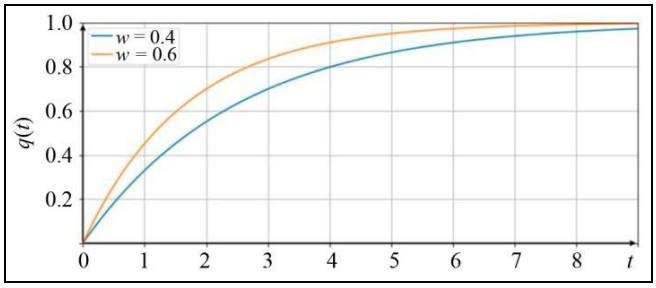

Fig. 1. The learning curve for Model 1: an illustrative example.

Model 2 (the forgetting model). Assume there is no experience acquisition but only time-invariant forgetting:

$$q(t+1)=q(t)-uq(t),$$

$$\dot{q}=-qu,$$

$$q(t)=q(0)\mathrm{e}^{-ut}. \qquad (3)$$

The graph of the function (3) with $q(0)=1$ is shown in Fig. 2.

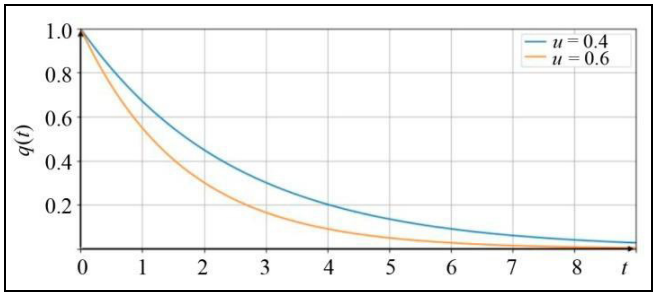

Fig. 2. The learning curve for Model 2: an illustrative example.

Model 3. Within Model 1, consider time-invariant forgetting (equivalently, time-invariant mastering within Model 2):

$$q(t+1)=q(t)+(1-q(t))w-uq(t),$$

$$\dot{q}=(1-q)w-qu, \qquad (4)$$

$$q(t)=\frac{w}{w+u}-\left(\frac{w}{w+u}-q(0)\right)\mathrm{e}^{-(w+u)t}. \qquad (5)$$

The learning curve (5) with $q(0)<\dfrac{w}{w+u}$ is non-decreasing and asymptotically tends to $\dfrac{w}{w+u}$. The graph of the function (5) with $q(0)=0$ and $w=0.6$ is shown in Fig. 3.

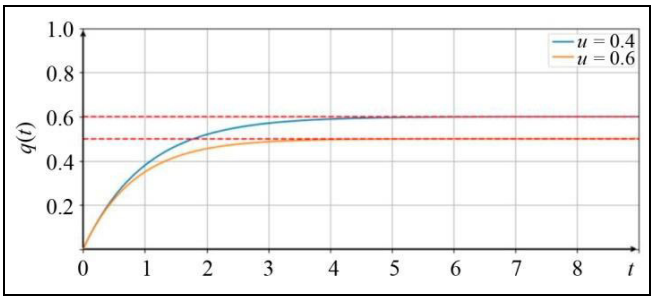

Fig. 3. The learning curve for Model 3: an illustrative example.

Model 4. Within Model 1, let the probability of experience acquisition depend on time and be positive: $w(t) > 0$. Assume that $w(\cdot)$ is a continuous non-increasing function, conditionally called the "*fatigue curve*". In a practical interpretation, this function reflects the agent's *fatigue* and/or *exhaustion* in the learning process. Then

$$q(t+1)=q(t)+w(t)(1-q(t)),$$

$$\dot{q}=(1-q)w(t),$$

$$q(t)=1-(1-q(0))e^{-W(t)}, \qquad (6)$$

where $W(t)=\int_0^t w(\xi)\,d\xi$. Obviously, the latter function has the following properties.

Lemma 1. *$W(\cdot)$ is a continuous, positive, monotonically increasing, and concave function of time such that $W(0) = 0$.*

The learning curve (6) is non-decreasing and asymptotically tends to 1. In the special case $w(t) = w$, Model 3 turns into Model 1.

For illustrations below, we choose $w(t)=w_0\mathrm{e}^{-\alpha t}$ and $W(t)=\dfrac{w_0}{\alpha}\left(1-\mathrm{e}^{-\alpha t}\right)$. The graph of the function (6) with $q(0)=0$ and $w_0=0.6$ is shown in Fig. 4.

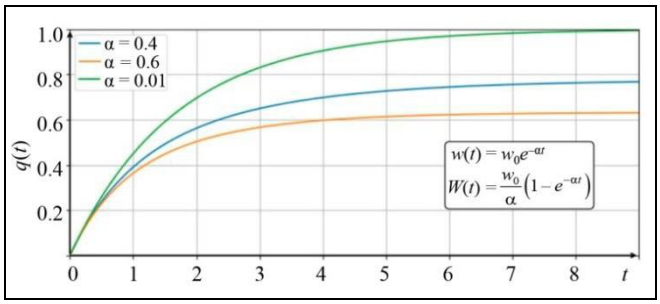

Fig. 4. The learning curve for Model 4: an illustrative example.

Model 5. Within Model 4, consider time-invariant forgetting: $u(t) = u$. Then

$$q(t+1) = q(t) + w(t)\big(1 - q(t)\big) - uq(t),$$

$$\dot{q} = (1-q)w(t) - uq,$$

$$q(t) = \mathrm{e}^{-(W(t)+ut)} \cdot \int_0^t w(\xi) \cdot \mathrm{e}^{(W(\xi)+u\xi)} \mathrm{d}\xi. \qquad (7)$$

The graph of the function (7) with $w_0 = 0.6$ and $u = 0.1$ is shown in Fig. 5. As can be observed, the learning curve in Model 5 has a maximum.

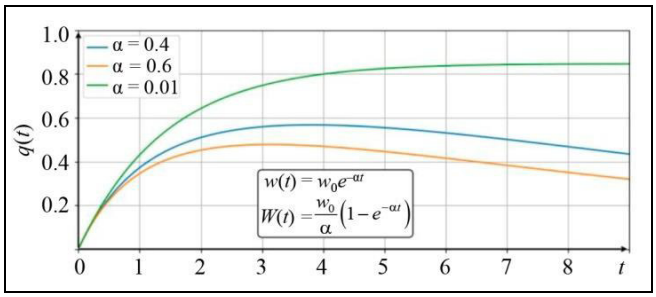

Fig. 5. The learning curve for Model 5: an illustrative example.

Model 6. Within Model 3, let the probability of forgetting be a continuous non-decreasing function $u(\cdot)$ of time. In this case, the solution of equation (4) is the learning curve

$$q(t) = \mathrm{e}^{-U(t)} \left(q(0) + w \int_0^t \mathrm{e}^{U(\xi)} d\xi \right), \qquad (8)$$

where $U(t) = \int_0^t (w + u(\xi)) d\xi$. For $U(t) = u$, the expression (8) becomes (5). The graph of the function (8) with $w = 0.6$, $q(0) = 0$, $u(t) = 1 - \mathrm{e}^{-\alpha t}$, and $U(t) = (w+1)t - \frac{1}{\alpha}\left(1 - \mathrm{e}^{-\alpha t}\right)$ is shown in Fig. 6.

Lemma 2. *$U(\cdot)$ is a continuous, positive, monotonically increasing, and convex function such that $U(0) = 0$.*

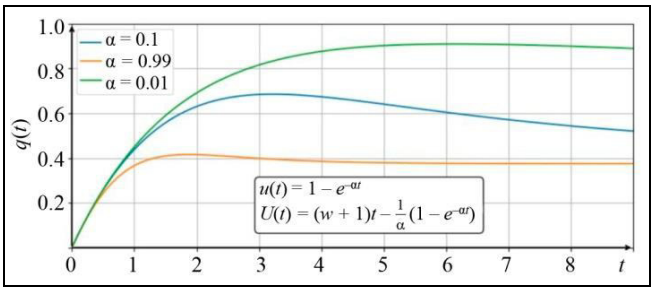

Fig. 6. The learning curve for Model 6: an illustrative example.

The graph of the function (8) with $w_0 = 0.6$ and $u_0 = 0.1$ is shown in Fig. 5.

The curve (8) can have a maximum at a point t^* satisfying the condition

$$\left(w + u\left(t^*\right)\right) \int_0^{t^*} \mathrm{e}^{u(\xi)} d\xi = \mathrm{e}^{u\left(t^*\right)}.$$

The value t^* can be interpreted as the maximum reasonable duration of learning.

Model 7. Within Model 4, let the agent have a rest interval $[\tau, \tau + \Delta]$, where $\tau > 0$ is the time to start a rest of duration $\Delta \geq 0$. Recall that there is no forgetting in the case under consideration: $q(\tau + \Delta) = q(\tau)$. The result of rest (an increase in the probability of experience acquisition) will be reflected by assuming that $\Delta \geq \Delta_0$, i.e., the duration of rest is sufficient for the agent to fully recover:

$$w(t) = w(t - \tau - \Delta),\ t \geq \tau + \Delta.$$

where the minimum necessary duration $\Delta_0 > 0$ is known. Then

$$q(t) = \begin{cases} 1 - \big(1 - q(0)\big)\mathrm{e}^{-W(t)}, & t \in [0, \tau], \\ 1 - \big(1 - q(0)\big)\mathrm{e}^{-W(\tau)}, & t \in [\tau, \tau + \Delta], \\ 1 - \Big[\big(1 - q(0)\big)\mathrm{e}^{-W(\tau)}\Big]\mathrm{e}^{-W(t-\tau-\Delta)}, & t \geq \tau + \Delta. \end{cases} \qquad (9)$$

For the sake of simplicity, we take the initial level of learning to be $q(0) = 0$. The graph of the function (9) with $w_0 = 0.6$ and $\alpha = 0.6$ is shown in Fig. 7.

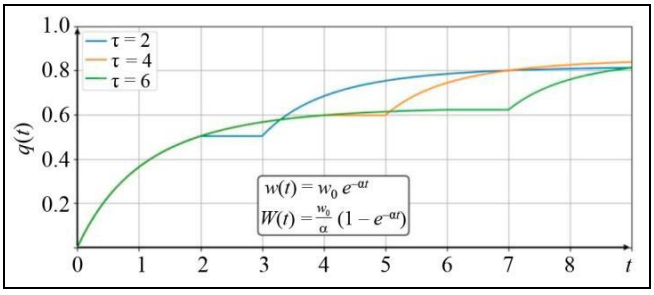

Fig. 7. The learning curve for Model 7: an illustrative example.

Two problems can be posed and solved by choosing the time to start rest and its duration.

The first problem is to maximize the terminal learning level on a planning horizon T:

$$1-e^{-W(\tau)-W(T-\tau-\Delta)}\to\max_{\tau\in(0,T-\Delta_0),\,\Delta\in[\Delta_0,T-\tau]}. \quad (10)$$

The second problem is to maximize the integral value of the learning level:

$$\int_0^{\tau}\left(1-e^{-W(\varsigma)}\right)d\varsigma+\int_{\tau+\Delta}^{T}\left(1-e^{-W(\tau)}e^{-W(\xi-\tau-\Delta)}\right)d\xi \to\max_{\tau\in(0,T-\Delta_0),\,\Delta\in[\Delta_0,T-\tau]}. \quad (11)$$

The integral criterion can be interpreted as the amount of agent's correct actions if learning occurs in productive activity [18].

Problem (10) can be written in the form

$$W(\tau)+W(T-\tau-\Delta)\to\max_{\tau\in(0,T-\Delta_0),\,\Delta\in[\Delta_0,T-\tau]}. \quad (12)$$

And problem (11) can be written in the form

$$\int_0^{\tau}e^{-W(\varsigma)}d\varsigma+e^{-W(\tau)}\int_0^{T-\tau-\Delta}e^{-W(\xi)}d\xi \to\min_{\tau\in(0,T-\Delta_0),\,\Delta\in[\Delta_0,T-\tau]}. \quad (13)$$

Due to Lemma 1 and the structure of the criteria (12) and (13), we arrive at the following result.

Proposition 1. *Within Model 7, the optimal rest duration* $\Delta^{(*)}$ *in both problems (maximizing the terminal and integral learning levels) is equal to the minimum necessary one:* $\Delta^{*}=\Delta_0$.

Based on Proposition 1, problems (12) and (13) are reduced to the following problems of selecting the optimal time to start rest:

$$W(\tau)+W(T-\tau-\Delta_0)\to\max_{\tau\in(0,T-\Delta_0)}. \quad (14)$$

and

$$\int_0^{\tau}e^{-W(\varsigma)}d\varsigma+e^{-W(\tau)}\int_0^{T-\tau-\Delta_0}e^{-W(\xi)}d\xi\to\min_{\tau\in(0,T-\Delta_0)}, \quad (15)$$

respectively.

The first-order optimality condition for the solution $\tau^{(*)}$ of problem (14) is $w(\tau^{*})=w(T-\tau^{*}-\Delta_0)$. By the continuity and monotonicity of the function $w(\cdot)$, we obtain

$$\tau^{*}=\frac{T-\Delta_0}{2}. \quad (16)$$

The second derivative of (14) at this point is negative, which can be easily verified.

Within Model 7, it is better to take a break near the middle of the experience acquisition interval in terms of maximizing the terminal learning level.

Proposition 2. *Within Model 7, the optimal time to start rest (in terms of maximizing the terminal learning level) is given by* (16) *and does not depend on the fatigue curve.*

The expression (16) yields the optimal time to start rest if a rest of duration Δ_0 must be taken. Meanwhile, is a rest break necessary? To answer, we should compare the value of the criterion (10) at the point τ^{*} with the maximum value of the learning level $1-e^{-W(T)}$ achieved without rest. Of course, the answer depends both on the properties of the function $w(\cdot)$ and on the values of T and Δ_0 and is provided below.

Proposition 3. *A rest break is reasonable in terms of maximizing the terminal learning level if*

$$W\left(\frac{T-\Delta_0}{2}\right)\ge\frac{W(T)}{2}. \quad (17)$$

Condition (17) *can be written as*

$$\int_0^{\frac{T-\Delta_0}{2}}w(\varsigma)d\varsigma\ge\int_{\frac{T-\Delta_0}{2}}^{T}w(\varsigma)d\varsigma. \quad (18)$$

Due to the decreasing integrand, inequality (18) naturally holds for sufficiently small Δ_0 and/or sufficiently large T. At the same time, for a fixed Δ_0, it is possible to find a value of T turning (18) into equality, i.e., the minimum planning horizon on which a rest of a given duration will still be reasonable. Conversely, for a fixed T, we can find a value of Δ_0 turning (18) into equality, i.e., the maximum duration of rest that will still be reasonable on a given planning horizon.

Now we pass to problem (15). The first-order optimality condition for its solution τ^{**} is given by

$$\int_0^{T-\tau^{**}-\Delta_0}e^{-W(\xi)}d\xi=\frac{1-e^{-W\left(T-\tau^{**}-\Delta_0\right)}}{w\left(\tau^{**}\right)}.$$

As is easily verified, the second derivative of (15) takes a negative value at this point.

Model 8. Within Model 7, let forgetting occur only during rest with a constant and time-invariant probability u. According to the expression (3) of Model 2, when the rest phase ends, the learning level will decrease to

$$q(t+\Delta_0)=q(\tau)e^{-u\Delta_0}.$$

Then

$$q(t) = \begin{cases} 1-e^{-W(t)}, \ t \in [0, \tau], \\ \left(1-e^{-W(\tau)}\right) e^{-u(t-\tau)}, \ t \in [\tau, \tau+\Delta_0], \\ 1-\left[1-\left(1-e^{-W(\tau)}\right)e^{-u\Delta_0}\right] e^{-W(t-\tau-\Delta_0)}, \ t \geq \tau+\Delta_0. \end{cases} \quad (19)$$

For Model 8, we consider the problem of maximizing the terminal learning level. The integral criterion is analyzed similar to Model 7:

$$1-\left[1-\left(1-e^{-W(\tau)}\right)e^{-u\Delta_0}\right] e^{-W(t-\tau-\Delta_0)} \to \max_{\tau \in (0, T-\Delta_0)},$$

which means that

$$\left[1-\left(1-e^{-W(\tau)}\right)e^{-u\Delta_0}\right] e^{-W(T-\tau-\Delta_0)} \to \min_{\tau \in (0, T-\Delta_0)}. \quad (20)$$

The first-order optimality condition for problem (20) has the form

$$e^{u\Delta_0} - 1 = \left(\frac{w(\tau^*)}{w(T-\tau^*-\Delta_0)} - 1\right) e^{-W(\tau^*)}.$$

The graph of the function (19) with $w_0 = 0.6$, $u_0 = 0.6$, and $\alpha = 0.6$ is shown in Fig. 8.

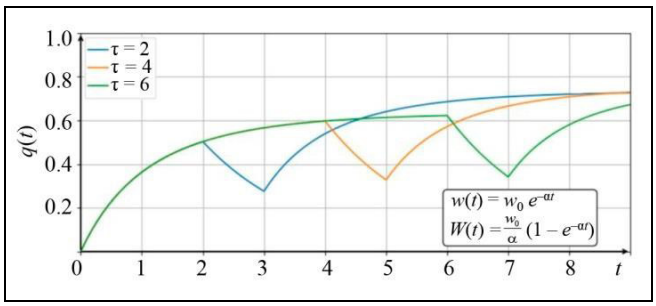

Fig. 8. The learning curve for Model 8: an illustrative example.

Model 9. Within Model 8, consider time-invariant forgetting during rest and learning as well. In this case, the section of $q(t)$ before rest will match formula (7) of Model 5. The corresponding learning curve may no longer be monotonic and, e.g., have a maximum.

Model 10. Within Model 9, let mastering be time-invariant whereas forgetting time-varying. In this case, the section of $q(t)$ before rest will match formula (8) of Model 6. Then the analog of the expression (19) for this model has the form

$$q(t) = \begin{cases} e^{-U(t)} \cdot \left(w \int_0^t e^{U(\xi)} d\xi + q(0) \right), \\ \qquad t \in [0, \tau], \\ e^{\hat{U}(t-\tau)} \cdot \left[e^{-U(\tau)} \cdot \left(w \int_0^\tau e^{U(\xi)} d\xi + q(0) \right) \right], \\ \qquad t \in [\tau, \tau+\Delta_0], \\ e^{-U(t-\tau-\Delta_0)} \cdot \left\{ w \int_0^{t-\tau-\Delta_0} e^{U(\xi)} d\xi \right. \\ \left. + e^{\hat{U}(t-\tau)} \cdot \left[e^{-U(\tau)} \cdot \left(w \int_0^\tau e^{U(\xi)} d\xi + q(0) \right) \right] \right\}, \\ \qquad t \geq \tau+\Delta_0, \end{cases}$$

where generally $\hat{U}(t) \neq U(t)$.

It seems unreasonable to study this model (and even more complex ones) in the analytical form further since the current results are cumbersome and yield no analytical conclusions about the regularities of experience acquisition. Numerical methods can be used instead. However, for this purpose, it is necessary to use particular functions as the probabilities of mastering and forgetting depending on time.

CONCLUSIONS

In this paper, we have extended the experience acquisition models from [3] by including the factors of fatigue and rest. An important result is the identification of optimal time intervals for breaks in terms of maximizing the terminal learning level: for problems described by Model 7, the optimal time to start rest does not depend on the time-varying probability of experience acquisition and is approximately at the middle of the experience acquisition interval. In more complex models with known learning and/or forgetting functions, it is also possible to identify the maximum reasonable learning time and the optimal time to start rest.

Thus, supplementing experience acquisition models with fatigue and rest processes increases their practical applicability, allowing one to predict performance degradation and optimize resources by calculating rest intervals.

Part II of this research will be devoted to datasets with experience acquisition dynamics and examples of the application of the above models to describe the results of well-known experimental studies.

REFERENCES

1. *Kratkii psikhologicheskii slovar'* (Brief Psychological Dictionary), Moscow: Politizdat, 1985. (In Russian.)
2. Novikov, D.A., *Zakonomernosti iterativnogo naucheniya* (Regularities of Iterative Learning), Moscow: Institute of Control Sciences RAS, 1998. (In Russian.)
3. Belov, M.V. and Novikov, D.A., Models of Experience, *Control Sciences*, 2021, no. 1, pp. 37–52. DOI: 10.25728/cs.2021.1.5.
4. Novikov, A.M. and Novikov, D.A., *Metodologiya* (Methodology), Moscow: Sinteg, 2007. (In Russian.)
5. Piovesan, M., Cognitive Fatigue Influences Students' Performance on Standardized Tests, *Proceedings of the National Academy of Sciences of the United States of America*, 2016, vol. 113, no. 10. DOI: 10.1073/pnas.1516947113
6. Bennis, F., A New Muscle Fatigue and Recovery Model and Its Ergonomics Application in Human Simulation, *Virtual and Physical Prototyping*, 2010, vol. 5, no. 3. DOI: 10.1080/17452759.2010.504056
7. van Geert, P., Some Thoughts on Dynamic Systems Modeling of L2 Learning, *Frontiers in Physics*, 2023, vol. 11. DOI: 10.3389/fphy.2023.1186136
8. Ryan, R.M. and Deci, E.L., Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being, *American Psychologist*, 2000, vol. 55, no. 1, pp. 68–78.
9. Ellis, N. and Larsen-Freeman, D., Constructing a Second Language: Analyses and Computational Simulations of the Emergence of Linguistic Constructions from Usage, *Lang. Learn.*, 2009, vol. 59 (suppl. l), pp. 90–125. DOI:10.1111/j.1467-9922.2009.00537.x
10. Rastelli, S., The Discontinuity Model: Statistical and Grammatical Learning in Adult Second-Language Acquisition, *Lang. Acquis.*, 2019, vol. 26, no. 4, pp. 387–415. DOI:10.1080/10489223.2019.1571594
11. Velichkovsky, B.B., Cognitive Effects of Mental Fatigue, *Lomonosov Psychology Journal*, 2019, no. 1, pp. 108–122. DOI: 10.11621/vsp.2019.01.108 (In Russian.)
12. Yao, Y., Wang, H., Zhang, Q., et al., Learning Curves for Itinerant Nurses to Master the Operation Skill of Ti-Robot Assisted Spinal Surgery Equipment by CUSUM Analysis: A Pilot Study, *PLoS ONE*, 2024, vol. 19, no. 3. DOI: 10.1371/journal.pone.0291147
13. Wright, T.P., Factors Affecting the Cost of Airplanes, *Journal of the Aeronautical Sciences*, 1936, vol. 3, no. 4, pp. 122–128.
14. Mohamad, Y.J. and Christoph, H.G., A Learning Curve for Tasks with Cognitive and Motor Elements, *Computers & Industrial Engineering*, 2013, vol. 64, no. 3, pp. 866–871. DOI: 10.1016/j.cie.2012.12.005
15. Eric, H.G., Christoph, H.G., and Sebastian, M., Production Economics and the Learning Curve: A Meta-Analysis, *International Journal of Production Economics*, vol. 170, part B, pp. 401–412. DOI: 10.1016/j.ijpe.2015.06.021
16. Bestwick-Stevenson, T., Toone, R., Neupert, E., et al., Assessment of Fatigue and Recovery in Sport: Narrative Review, *International Journal of Sports Medicine*, 2022, vol. 43, no. 14. DOI: 10.1055/a-1834-7177
17. Enoka, R.M. and Duchateau, J., Translating Fatigue to Human Performance, *Medicine & Science in Sports & Exercise*, 2016, vol. 48, no. 11, pp. 2228–2238. DOI: 10.1249/MSS.0000000000000929
18. Belov, M.V. and Novikov, D.A., *Models of Technologies*, Cham: Springer, 2020.

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

Received March 26, 2025, and revised April 29, 2025. Accepted May 5, 2025.

Author information

Grebenkov, Dmitry Igorevich. Mathematician, Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ grebenkov-d-i@mail.ru
ORCID iD: https://orcid.org/0009-0002-7085-5912

Kozlova, Anastasiia Andreevna. Engineer, Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ sankamoro@mail.ru
ORCID iD: https://orcid.org/0009-0005-6105-121X

Lemtyuzhnikova, Dar'ya Vladimirovna. Cand. Sci. (Phys.–Math.), Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ darabbt@gmail.com
ORCID iD: https://orcid.org/0000-0002-5311-5552

Novikov, Dmitry Aleksandrovich. Academician, Russian Academy of Sciences; Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
✉ novikov@ipu.ru
ORCID iD: https://orcid.org/0000-0002-9314-3304

Cite this paper

Grebenkov, D.I., Kozlova, A.A., Lemtyuzhnikova, D.V., and Novikov, D.A., Models of Fatigue and Rest in Learning. Part I: Extension of the General Iterative Learning Model. *Control Sciences* 3, 24–31 (2025).

Original Russian Text © Grebenkov, D.I., Kozlova, A.A., Lemtyuzhnikova, D.V., Novikov, D.A., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 28–37.

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

THE ESSENCE, ATTRIBUTES, AND DESCRIPTION PRINCIPLE OF ORGANIZATIONAL SYSTEMS

G. A. Ougolnitsky

Southern Federal University, Rostov-on-Don, Russia

✉ gaugolnickiy@sfedu.ru

Abstract. The concept and essence of an organizational system are discussed. A general characterization of this concept is provided within the theory of control in organizational systems. Several attributes are indicated to classify a given system as an organizational one (the description principle of organizational systems). A mathematical model of a general organizational system is described, with an illustrative example based on the Cournot duopoly. Extended examples of some classes of organizational systems specified using this principle are presented. Such classes include special-purpose organizational and technical systems, queuing organizational systems, and ecological-economic organizational systems. Typical representatives of each class are described within the general model proposed. This paper has a methodological focus and is intended to define a standard description of organizational systems.

Keywords: organizational systems, game-theoretic models, control in active systems.

INTRODUCTION

The topicality of organizational control problems is beyond doubt: the quality of this control determines labor productivity, economic growth, and thus government capabilities and the welfare of the population.

The Philosophical Encyclopedic Dictionary gives the following definition of *organization*: 1) internal orderliness, the consistency of interaction between more or less differentiated and autonomous parts of a whole, conditioned by its structure; 2) a totality of processes or actions leading to the formation and improvement of interrelations between the parts of a whole; 3) an association of people jointly implementing some program or goal and acting based on certain procedures and rules [1]. The totality of these procedures and rules is called a *mechanism of functioning*. The third meaning of the term "organization" is the definition of an *organizational system*. Methodology uses a similar definition [2]: *organization* is a complex activity with the goal of creating internal orderliness, the consistency of interaction of more or less differentiated and autonomous elements of the subject of this activity (in particular, by forming and maintaining interrelations with specified characteristics between these elements).

Formal models of organization control are provided by the theories of active systems and control in organizational systems (OSs) [3, 4], the information theory of hierarchical systems [5–8], contract theory and mechanism design [9], and the theory of sustainable management of active systems [10–12]. The possibilities of using artificial neural networks in the study of hierarchical organizational systems were shown in [13]. To a large extent, we agree with the author's opinion [13] that multistage hierarchical games of many persons represent the **LANGUAGE** of organizational systems control, although models of this control theory involve other mathematical constructs as well. According to the fundamental monograph on the theory of control in organizations (TCO) [4], the object of research is OSs, the subject of research is control mechanisms, and the main method of research is mathematical modeling. Following this approach, the main attention in [4] was paid to control problems and control mechanisms, namely, control of the staff and structure of OSs, institutional control of activity constraints and norms, motivational control of interests and preferences, informational control, and control of the order of functioning (the sequence of acquiring information and choosing strategies by agents). Finally, the OS model is defined by specifying its staff (a set of OS participants), structure (a set of various rela-

tions between OS participants), a set of feasible strategies of OS participants, their preferences, awareness, and the order of functioning.

Meanwhile, where is the watershed between the approaches of management and TCO? This issue remains unsettled. Another question deserving a clear answer is: what kind of system can be considered an organizational system? Indeed, many complex systems of the real world are not such.

This paper has the following contribution:

- The essence of organizational systems is described, and their mathematical formalization defining the TCO approach is proposed.
- Several attributes are indicated to classify a given system as an organizational one (the description principle of organizational systems).
- In accordance with this principle, some classes of organizational systems are identified and characterized in detail.

In Section 1, we describe the essence of organizational systems as well as their attributes and general mathematical model. In Section 2, the description principle of classes of organizational systems is considered. Section 3 gives extended examples of some classes of organizational systems obtained by applying this principle. The results and perspectives are discussed in the Conclusions.

1. THE ESSENCE AND ATTRIBUTES OF ORGANIZATIONAL SYSTEMS

The following definition was given in the remarkable book [14]: a social system is a dynamic set of autonomous individuals pursuing their goals in interaction with an environment. This definition is quite close to what should be understood by an OS. However, since any complex concept is a synthesis of many definitions, it seems more convenient to define an OS in a detailed way through its mathematical model. This model synthesizes the description from [3, 4] (see the Introduction) and the active network model from [15], with some essential refinements and additions. We suppose that an OS consists of a control (active) subsystem and a controlled subsystem. (Note that such a structure includes organizational and technical systems as well.) The OS model has the form

$$\left\langle N,\ A,\ X,\ I,\ U,\ S,\ F,\ J, R\right\rangle \qquad (1.1)$$

with the following notation:

$N = \{0,\ 1,\ldots,\ n\}$ is the set of active agents, which form the staff of the OS control subsystem. The number 0 is associated with a selected agent (Principal), representing a single agent for the time being.

$A = \{(i,\ j)\}$ is the set of different-type links between active agents $i, j \in N$. A directed graph $D = (N,\ A)$ defines the structure of OS links, which are defined by subordination relations as well as substance, energy, and information flows.

X is the state set of the controlled subsystem of the OS (a subset of some topological vector space).

$I = I_0 \times I_1 \times \ldots \times I_n$, where I_i is the information available to agent i about the OS and its environment. Since the deterministic model (1.1) is considered, in contrast to [16], this information contains the agent's beliefs about the actions of other agents and their payoff functions as well as about the controlled subsystem's state. All agents, including the Principal, strive to maximize their payoffs; in the presence of ambiguity, they are guided by the principle of guaranteed result [8].

$U = U_0 \times U_1 \times \ldots \times U_n$, where U_i is the set of feasible actions of agent i (a subset of some finite-dimensional space [8]).

$S = S_0 \times S_1 \times \ldots \times S_n$, where S_i is the set of feasible strategies of agent i. A feasible strategy $s_i \in S_i$ is a mapping $s_i : I_i \to U_i$ that determines the choice of a feasible action by agent i depending on the information available to him/her.

F is a rule of changing the states under the actions of active agents. It can be a system of algebraic, differential or difference equations, or an algorithm that explicitly defines the transitions between the controlled subsystem's states. This rule can be treated as a generalized operator acting in the state space.

$J = (J_0, J_1, \ldots, J_n)$ is the set of payoff functionals of active agents. A mapping $J_i : U \times X \to \mathbb{R}$ defines the payoff of agent i depending on the actions of all agents and the current state of the controlled subsystem. An efficiency criterion of the entire organization is the value of a Principal's payoff functional, which generally depends on the actions of all agents and the state of the controlled subsystem.

R is an order of functioning of the OS, which algorithmically determines the sequence of choosing strategies by active agents, the possible transmission of information to other agents, and changes in the state of the controlled subsystem.

Accordingly, we distinguish the following attributes of an OS.

1. *The activeness of agents*. Each agent has an individual payoff functional J_i and independently chooses a feasible strategy $s_i \in S_i$. Within the model, the optimization of the payoff functional completely determines the agent's interests and preferences. In

particular, agents may deliberately distort the information transmitted to other agents in their own interests (the so-called *manipulation* problem of decision procedures [17]). Other manifestations of activity include the *far-sighted* behavior of agents [18] and *reflexion* regarding their activity and the activity of other agents [16].

2. *Goal-setting*. An organizational system has a certain goal, which is established independently or set from the outside (e.g., by society or a superior organization). In the general case, this goal, expressed in the form of a constraint, consists at least in fulfilling the viability condition $X \subseteq X^*$ of the organization. (In other words, the values of all its essential indicators belong to a given range.) Technically, it is possible to incorporate the viability requirement into the Principal's payoff functional J_0 via a penalty function $\rho(X, X^*)$: the latter is 0 when the viability condition holds and takes an infinitely large value when it is violated. In the absence of any viability condition, the goal of the OS is only to maximize the functional J_0 without penalties.

3. *Organization*. An organizational system (an extended active system) is formed by a control subsystem consisting of active agents, including the Principal, and a controlled subsystem. A controlled subsystem does not contain active agents: it includes technical, economic, and other components controlled by active agents. The interaction between active agents is established by an order of functioning R and determines the dynamics of the controlled subsystem (the change of its state $x \in X$ over time by a rule F) and the payoffs J_i of agents.

2. THE DESCRIPTION PRINCIPLE OF ORGANIZATIONAL SYSTEMS. AN ILLUSTRATIVE EXAMPLE

Some system is called organizational if and only if it has all the three features of OSs listed above: the activeness of agents, goal-setting, and a control mechanism. Then it is possible to build model (1.1) of this OS and specify the values of its components. In particular cases, some components can be empty sets or take trivial values.

Consider the following simplified illustrative example, known as the Germeier game (the inverse Stackelberg game) Γ_2 in a Cournot duopoly [8]. Here, $N = \{0, 1\}$ in the expression (1.1). The organization consists of two active elements, namely, a Principal and an agent; $A = \{(0, 1)\}$. The Principal makes the first move by formulating the "rules of play" for the agent.

The Principal knows U_0, U_1, J_0, and J_1, will have information about the choice $u_1 \in U_1$, and is aware that the agent chooses an appropriate strategy by maximizing his/her own payoff. In other words, the Principal's strategy (control mechanism) is a mapping of the set U_1 into U_0. The agent knows U_1 and J_1. Given the agent's known strategy, the Principal seeks to maximize his/her own payoff J_0. Under an ambiguous choice of the agent, the Principal is guided by Germeier's generalized principle of guaranteed result [5, 6, 8]. Let

$$J_0 = (a - c_1 - u_1 - u_2)u_1,$$

$$J_1 = (a - c_2 - u_1 - u_2)u_2,$$

$$c_1 < c_2 < a,$$

where $U_0 = U_1 = [0, 1]$, $s_0 : U_1 \to U_0$, $s_1 \equiv u_1$. The model is static and contains no controlled subsystem, i.e., the set X and the rule F are not considered. The goal of the organization is to maximize the Principal's payoff J_0.

The Principal makes the first move by choosing the strategy

$$u_0(u_1) = \begin{cases} (a - c_1)/2, & u_1 = 0, \\ a - c_1 - u_1, & u_1 > 0, \end{cases}$$

and reports it to the agent. The agent's unique optimal response is $u_1 = 0$, resulting in $J_0 = (a - c_1)^2/4$, $J_1 = 0$ (otherwise, $J_1 < 0$). A possible practical interpretation of the Cournot duopoly is as follows: firm 0 offers firm 1 some terms, becoming the duopoly's leader. Say firm 0 can offer firm 1 a side payment for a higher price in a tender involving firm 0. This situation is more precisely described by the Bertrand duopoly [7], but Cournot's quantity competition is suitable as well.

3. EXTENDED EXAMPLES

Here are some examples to demonstrate the description approach to OSs. In each subsection below, we provide a general characterization of some class of OSs, with its further specification to particular OSs within the class.

3.1 Special-Purpose Organizational and Technical Systems

The first extended example of OSs is special-purpose organizational and technical systems (OTSs). Here, we consider surveillance and interception systems for opponent's aircraft and missiles. Such systems and their several examples were characterized in [19]. In the general case, a special-purpose OTS has the following components:

N is the group of surveillance agents. The Principal is the group commander. Note that active agents also represent the opponent's side. They are not included in the OS (being its environment); however, their actions can be considered in the OS model, particularly when determining the state of its controlled subsystem and describing other aspects, e.g., active confrontation.

A is the set of links between agents. They are determined by the need for operational interaction during surveillance and interception.

I_i is the information available to agent i about the actions of other agents, the Principal, and the opponent as well as about the state of the controlled system.

U_i is the set of feasible actions of agent i (position choice, targeting parameters of anti-aircraft mounts).

S_i is the set of strategies of agent i depending on the opponent's actions. Since these actions affect the system state, closed-loop strategies are generally used; they depend not only on time but also on the system state.

$x = (x_1,..., x_m) \in X$ is the system state vector (the coordinates and velocities of objects under surveillance).

F is the general rule of changing the system state. This rule is determined by the mechanical and geometrical conditions as well as the regularities of system operation.

J_i is the payoff functional of agent i. It is determined by the accuracy of target detection.

R is the order of decision-making in the special-purpose OTS. This order is determined by the rules of some Germeier game [8] in which the Principal acts as the leader whereas surveillance agents and opponent agents as followers with simultaneously chosen actions.

Let us take the triangulation measurement system (TMS) [19] as a particular example of special-purpose OTSs. In this case, the set N (agents) is formed by TMS operators, each associated with a separate measuring point. The set A is given by pairwise bilateral links between the agents and the Principal. Using these links, the agents transmit information about their actions and the state of the controlled system to the Principal and the Principal informs the agents of his/her strategies chosen. Each agent knows the set of his/her actions and the payoff functional, which define the set I_i. We consider the control problem in a static formulation, so the state vector x and the rule F are omitted.

Let $\{P_n\}_{n=1}^N = \{[x_n^p,\ y_n^p]\}_{n=1}^N$ be the measuring points of the TMS and $\{S_m\}_{m=1}^M = \{[x_m^s,\ y_m^s]\}_{m=1}^M$ be the points of a deception jamming system. Here, N is the total number of measuring points, M is the total number of jamming points, and x and y are the 2D coordinates of points. The superscripts p and s indicate measuring and jamming points, respectively; the subscripts n and m, the point numbers. The surveillance agent's problem can be written as follows:

$$J(u_p, u_s) = |\, N - K \,| \to \max_{u_p},$$

$$K < \lfloor N/2 \rfloor + 1,$$

$$\forall i, j \in \{1,...,N\} : \| P_i - P_j \| > B_{\min},$$

$$\forall i, j, k \in \{1,...,N\} : \frac{y_k^p - y_i^p}{y_j^p - y_i^p} \neq \frac{x_k^p - x_i^p}{x_j^p - x_i^p},$$

where K is the number of measuring points in the jamming zone; $u_p = [P_1...P_N]$ and $u_s = [\alpha_1, S_1..., \alpha_M, S_M]$; $\lfloor \cdot \rfloor$ denotes the integer part of numbers; $\|\cdot\|$ means the Euclidean norm; α_m is the angle of rotation for the jamming sector of point m. Here, the condition $K < \lfloor N/2 \rfloor + 1$ reflects the number of working measuring points necessary for normal TMS operation (more than half of their total number). The other two constraints express requirements for the TMS topology: $B_{\min}$ is the minimum allowable distance between TMS points; no three TMS points should lie on the same straight line. Thus, the set of feasible actions U_i has been defined; for the sake of simplicity, by assumption [19], the strategies S_i coincide with the actions.

The value of the goal function $J(u_p, u_s)$ represents the number of working measuring points. The surveillance agent maximizes this number under the above constraints. On the other hand, the opponent seeks to minimize it over u_s:

$$J(u_p, u_s) = |\, N - K \,| \to \min_{u_s},$$

$$K \geq \lfloor N/2 \rfloor + 1.$$

Thus, the payoff functionals $J = J_1 = -J_2$ define an antagonistic game between the surveillance agent and the opponent. According to the condition $K \geq \lfloor N/2 \rfloor + 1$, for the value minimizing the goal function, the number of TMS points jammed exceeds half of their total number. (Only in this case the TMS becomes inoperable.)

The decision order R is as follows: each agent chooses its feasible actions and reports them to the Principal. The Principal analyzes the whole situation and approves the actions proposed by the agents or corrects them by a command; after that, the set N can be treated as a single player. The opponent acts similarly, thereby determining the outcome (value) of the antagonistic game and the players' payoffs [19].

3.2 Queuing Organizational Systems

As the second extended example of an OS, we select queuing organizational systems (QOSs). This class of OSs was introduced in a series of papers [20–24]. Here:

N is the set of active agents who serve an incoming flow of requests. The Principal is the management of the service organization.

A is the set of links between agents arising in the service process.

I_i is the information available to agent i about the goals and capabilities of other agents (especially the Principal) and the principles of their decision-making.

U_i is the set of feasible actions of agent i related to the service process (the values of service intensity and quality).

The Principal assigns a service discipline and determines the capacity of service units (e.g., through expenditures for their purchase and maintenance), the staff, structure, and skill levels of agents, as well as the numerical parameters of administrative and economic mechanisms for managing agents.

S_i is the set of strategies of agent i. As in the case of special-purpose OTSs, the Principal's strategies are control mechanisms with feedback by agents' actions. However,

degenerate control mechanisms without feedback are also possible, e.g., normative regulations. The strategies of the other agents can either coincide with their actions (open-loop strategies) or depend on the state of the controlled system (closed-loop strategies) or on the actions of other agents (strategies with control feedback).

$x = (x_1,..., x_m) \in X$ is the state vector of the QOS, which includes as main variables the time interval between successively arriving requests and the service time of one request; m is the number of all state variables under consideration. These variables may differ for particular service units. The derived variables are the waiting time for a request and the idle time of the service units. It is also possible to introduce additional variables reflecting the specifics of a particular QOS.

F is a general rule of changing the QOS state. It is defined by the distribution laws of the above random variables and their parameters.

J_i is the payoff functional of agent i (the difference between his/her income and the cost of labor effort and skill development). The Principal's main goal in the QOS is to keep a balance between the waiting time of a request in the queue and the idle time of the service equipment, which determines the viability of the system.

R is the order of decision-making in the QOS. Here, the key role is played by the mechanisms of administrative and economic management of the Principal; knowing them, service agents make their decisions.

Now consider the "railway station–marine port" QOS [20–24]. Figure 3.1 shows the structural diagram of such a QOS. The control authority (Principal) is represented by the Ministry of Transport of the Russian Federation, regional ministries of transport, or another body capable of coordinating the work of given railway station and marine port, by order of the government or by voluntary agreement of economic entities. Active subsystems (agents) are the management and personnel of the railway station and marine ports, which define the set N. The set A is demonstrated in Fig. 3.1.

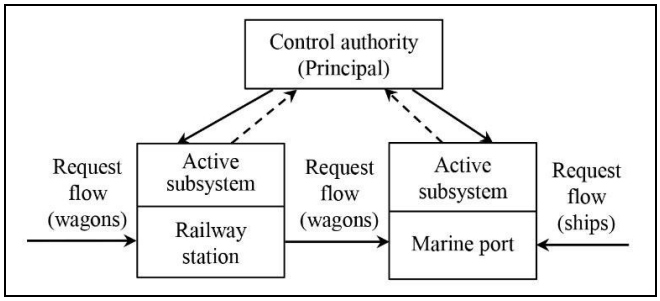

Fig. 3.1. The "railway station–marine port" QOS [24].

According to Fig. 3.1, this QOS contains two queuing subsystems: "wagon–port" (the left part of the figure) and "ship–port" (the right part of the figure). Let us begin with the "wagon–port" subsystem. Each wagon will be characterized by a set $w = \langle y, g, s, j, m, b, \omega \rangle$,

$y \in Y,\ g \in G,\ s \in S,\ j \in J,\ m \geq 0,\ b \in B,\ \omega \in \Omega,$

with the following notation: y is the wagon type from a set Y of different wagon types described by this model; g is the cargo kind from a set G of cargo kinds; s is the cargo type from a set S of cargo types; j is the batch number from a set J of batch numbers; m is the net weight of cargo in a wagon (if the wagon arrives for unloading) or the net weight of cargo required for loading (if an empty wagon arrives for loading); b is the attribute indicating the type of cargo operation with a given wagon (1—loading and 0—unloading, i.e., $B = \{0, 1\}$); ω is the train type of the wagon arrived, from a set Ω of train types under consideration.

The requests of the second ("ship–port") subsystem are the ships arriving at the port for cargo operations. Service units for these requests are berths with loading and unloading equipment.

We introduce the following set of characteristics for each ship:

$$\eta = \langle \tilde{G}, \tilde{S}, \tilde{J}, \Phi, l \rangle.$$

Here, $\tilde{G} \subset G$ is a set of transported cargo kinds; $\tilde{S} \subset S$ is a set of transported cargo types; $\tilde{J} \subset J$ is a set of cargo batch numbers; l is the ship length; $\Phi : \tilde{G} \times \tilde{S} \times \tilde{J} \to \mathbb{R}^+ \times B \times \mathbb{N}$ is a mapping that specifies the ship loading structure, i.e., for each triple (g, s, j) is assigns a triple (m, b, d), where $m \in \mathbb{R}^+$ is the weight of cargo of the gth kind and the sth type in the jth batch on the ship to be unloaded (loaded), $b \in B$ is the type of cargo operation (loading or unloading), and $d \in \mathbb{N}$ is the natural number specifying the order of cargo operation with the given cargo. The state variables x and their change rule F were described in detail in [20–23].

The dynamic hierarchical control model of the QOS has the following form [21–23]:

$$H(s_1, s_2) = \int_0^T \left[k_1 s_1(t) V(y_1(t)) + k_2 s_2(t) y_2(t) - d s_1^{\gamma}(t) - d s_2^{\gamma}(t) \right] dt, \tag{3.1}$$

$$C_1(y_1, s_1) = \int_0^T \left[c_1 y_1^{\frac{1}{\alpha}}(t) - k_1 s_1(t) V(y_1(t)) - c_3 x_1(t) p_1 - P \right] dt, \tag{3.2}$$

$$C_2(y_1, s_1) = \int_0^T \left[c_2 y_2^{\frac{1}{\beta}}(t) - k_2 s_2(t) y_2(t) - c_4 x_1(t) p_2 - k_3 Q(y_2(t)) \right] dt, \tag{3.3}$$

$$s_1 = (s_1(t_0), s_1(t_1),..., s_1(T)), \quad s_2 = (s_2(t_0), s_2(t_1),..., s_2(T)), \tag{3.4}$$

$$c_1 = (c_1(t_0), c_1(t_1),..., c_1(T)), \quad c_2 = (c_2(t_0), c_2(t_1),..., c_2(T)), \tag{3.5}$$

$$s_1^{\min}(t) \leq s_1(t) \leq s_1^{\max}(t), \quad y_1^{\min}(t) \leq y_1(t) \leq y_1^{\max}(t), \tag{3.6}$$

$$s_2^{\min}(t) \le s_2(t) \le s_2^{\max}(t), \quad y_2^{\min}(t) \le y_2(t) \le y_2^{\max}(t), \tag{3.7}$$

$$x_1(t) = x_1(t-1) + y_1^{\max}(t) - y_2(t), \tag{3.8}$$

$$x_2(t) = x_2(t-1) + y_1(t) - y_2(t-1); \tag{3.9}$$

$$V(y_1(t)) = \begin{cases} \dfrac{y_1(t)}{w_1}, & \mathrm{mod}(y_1, w_1) = 0 \\ \dfrac{y_1(t)}{w_1} + 1, & \mathrm{mod}(y_1, w_1) > 0, \end{cases} \tag{3.10}$$

$$Q(y_2(t)) = \begin{cases} \dfrac{y_2(t)}{w_2}, & \mathrm{mod}(y_2, w_2) = 0 \\ \dfrac{y_2(t)}{w_2} + 1, & \mathrm{mod}(y_2, w_2) > 0. \end{cases} \tag{3.11}$$

Here, H, and C_1, C_2 are the profit functions of the Principal and agents for time t; $s_1(t)$ is the Principal's control action for agent 1 (a railway transport company) at time t (i.e., the price of renting a docking place for railway customers); $s_2(t)$ is the Principal's control action for agent 2 (port) at time t (i.e., the port fee paid when transporting goods); $y_1(t)$ is the control action of agent 1 (i.e., the amount of goods sent from the company's warehouse to the port); $y_2(t)$ is the control action of agent 2 (i.e., the amount of products sent as delivery from the port warehouse); $x_i(t)$ is the amount of cargo not delivered to the final consumer by agent i; $c_i(t)$ is a penalty coefficient; p_1, p_2 are price coefficients; P is additional costs; α, β, γ are tuning parameters; finally, k_1, k_2, k_3 are coefficients for dimensionality matching [20–22].

Model (3.1)–(3.11) describes the elements J_i and U_i of the general model (1.1). The order of functioning R is as follows. The Principal selects its control actions $s_1(t)$, $s_2(t)$ and reports them to the agents. Knowing the Principal's control actions, the agents simultaneously and independently choose the values of their control actions $y_1(t)$, $y_2(t)$, respectively. Thus, the strategies of the agents and the Principal coincide with their actions (the dynamic Germeier game Γ_{1t}); the mutual awareness I_i of the players has been described as well.

3.3 Ecological-Economic Organizational Systems

The third extended example of a class of OSs is ecological-economic organizational systems (EEOSs). Control mechanisms for EEOSs with the specifics of their controlled subsystems were described in the monographs [25, 26] and papers [27–30]. In general, the components of EEOSs can be characterized as follows.

N is the set of agents (exploiters of natural resources) associated with an ecosystem forming the environmental basis of the ecological-economic system under consideration. These include enterprises of manufacturing, agriculture, recreation, transportation, and other economic sectors located in the territory of the ecosystem (or adjacent to its water area) as well as the local population. The Principal is the public administration of the territory.

A is the set of transport, economic, and administrative links between agents.

I_i is the information available to agent i about the actions and payoff functions of the other agents (especially the Principal).

U_i is the set of feasible actions of agent i. The Principal assigns permissible quotas for the exploitation of natural resources (fishing, logging, mining, etc.) and maximum permissible concentrations (emissions) of pollutants in the environment, as well as the numerical parameters of economic regulation mechanisms (taxes, fines, benefits, or subsidies). Economic agents choose production outputs, product prices, and the parameters of environmental protection measures.

S_i is the set of strategies of agent i. As a rule, the Principal's strategies are control mechanisms with feedback by the actions of agents. In particular, when considering opportunistic behavior, the control system has a feedback loop by the amount of agents' bribes, paid to the Principal, e.g., to relax environmental requirements or allocate additional resources. However, control mechanisms without feedback are also possible, such as legislative regulations. The strategies of other agents can either coincide with their actions (open-loop strategies) or depend on the state of the controlled system (closed-loop strategies) or on the actions of other agents (strategies with control feedback).

$x = (x_1, \ldots, x_m) \in X$ is the state vector of the ecological-economic system. It seems natural to divide its components into two subsets describing the ecological and economic subsystems of the EEOS, respectively.

F is a general rule of changing the EEOS state. In fact, it also consists of two parts: first, material balances and production functions describing economic activity; second, the rules of changing the values of ecological indicators. In turn, the second part is subdivided into the description of the natural dynamics of the environment and its anthropogenic change (pollution, exploitation of natural resources, environmental protection, etc.).

J_i is the payoff functional of agent i. Here, as active agents, we consider economic entities: their payoff is profit after the deduction of environmental protection costs and possible ecological penalties. The Principal has two criteria: economic growth in the territory and compliance with ecological requirements.

R is the order of decision-making in the EEOS. The main role here is played by the Principal's mechanisms of administrative and economic control, reported to the agents. Knowing these mechanisms, economic agents make their decisions.

Now we take an EEOS arising in fishery management [31]. Here, active agents are fishing enterprises and the Principal is an environmental authority. The links are limited by the Principal's impact on the agents. The latter maximize the goal functionals

$$J_i = \int_0^T e^{-\rho t}\{av_i(t)P(t) - s_i(t)M[P(t)-P^*]^2\}dt - e^{-\rho T}s_i(T)M[P(T)-P^*] \to \max \quad (3.12)$$

under their control constraints (the permissible actions of agents)

$$q_i(t) \le u_i(t) \le r_i,\ i = 1,..., N, \quad (3.13)$$

due to the bioresource dynamics equations

$$\frac{dP}{dt} = [\varepsilon - \beta P(t) + \alpha(\sum_i u_i(t))^\gamma - \sum_i \alpha_i (r_i - u_i(t))^{\gamma_i}]P(t), \quad (3.14)$$
$$P(0) = P_0,\ i = 1,..., N$$

(the rule F of changing the scalar state variable $x = P$). Here, P_0, $P(t)$ are the initial and current values of the bioresource (fish population biomass), and P^* is its ideal value fully satisfying the population viability requirements. When violating the viability condition, the agents are penalized with a coefficient $M >> 1$; a is the unit price of fish biomass; ε is the natural growth coefficient of the fish population; β is the self-limiting coefficient; ρ is the discount factor; finally, α, γ are tuning parameters of the model.

A simplified linear version of model (3.12)–(3.14) was also considered in [31].

By assumption, each agent allocates his/her resource between social and private interests. Therefore, his/her payoff is made up of two components, namely, the income from the private activity and the share of damage from the social evil that the agents jointly fight against. Agents (fishing enterprises) $i = 1,..., N$ maximize the income from fishing considering a possible penalty for violating the viability condition of the fish population. The viability condition (the Principal's control goal) is written as $\forall t\ P(t) = P^*$ or, in a weaker form, as $\forall t\ [P(t) - P^*]^2 \le \delta$.

The agent's control action $u_i(t)$ is the share of the resource r_i allocated to social needs. (Then $r_i - u_i(t)$ is the share of the resource allocated to the private activity.) In this model, $r_i - u_i(t)$ is the investment in increasing the fishing effort; then the share of fish caught by enterprise i is calculated as a function of the fishing effort: $v_i(t) = h_i(r_i - u_i(t))$. Without essential loss of generality, let $v_i(t) = k_i(r_i - u_i(t))^{p_i}$, $0 < p_i < 1$, where k_i, p_i are tuning parameters of the model. The value $u_i(t)$ is the allocations to improve sustainable fishery and fish farming.

Model (3.12)–(3.14) is an N-player differential game with the viability conditions incorporated into the goal functionals via the penalties $M[P(t) - P^*]^2$. The variables $s(t) = \{s_i(t)\}_{i=1}^N \in S$ are naturally interpreted as economic controls (impulsion) of the Principal as a top-level state control authority (e.g., the Fisheries Service) such that

$$0 \le s_i(t) \le 1,\ \sum_{i=1}^N s_i = 1;\ t \ge 0,\ i = 1,..., N. \quad (3.15)$$

The Principal can also use administrative control (compulsion) $q(t) = \{q_i(t)\}_{i=1}^N \in Q = \{0 \le q_i(t) \le r_i,\ t \ge 0\}$, selecting the values of the variables $q_i(t)$ from the condition

$$0 \le q_i(t) \le r_i,\ t \ge 0,\ i = 1,..., N. \quad (3.16)$$

The Principal's interests are supposed to be described by maximization of the following functional (the utilitarian social welfare function):

$$J = \sum_{i=1}^N J_i \to \max. \quad (3.17)$$

Then model (3.12)–(3.17) is a hierarchical differential game between the Principal and several active agents of the lower control level. We make several assumptions concerning the order R of such a game: all players use open-loop strategies; the Principal chooses the economic (3.15) or administrative (3.16) control actions (functions of time only or those of time and the agents' control actions); in particular, the Principal assigns penalties; under known Principal's strategies, the agents simultaneously and independently choose their actions, which leads to a Nash equilibrium in the normal-form game of the agents [31]. All players, including the Principal, know their strategy sets and payoff functionals; the awareness of players regarding the strategy sets and payoff functionals of other players is determined by the rules of some Germeier game [6–8].

CONCLUSIONS

Traditional management as a science and TCO have the same object, i.e., control of organizations. However, the subject of management is organizational-economic and socio-psychological relations and control methods, and the subject of TCO is mathematical models, information technologies, and methods of their application in control of organizations. We emphasize that mathematical models in TCO are used exclusively as tools for solving control problems (building and improving control mechanisms) and not as abstract formal-logical constructs of pure mathematics to be studied. By the way, in the classifier of the State Commission for Academic Degrees and Titles of Russia (VAK RF), the first direction corresponds to specialty 5.2.6 "Management (economic sciences)" whereas the second to specialty 2.3.4 "Control in organizational systems (engineering)."

This paper has proposed a general mathematical model describing an organizational system from the standpoint of TCO. The construction of such a model and its application to the analysis and control of a particular organization or class of organizations allows referring this study to TCO (the description principle

of OSs). Extended examples of some classes of OSs (special-purpose OSs, queuing OSs, and ecological-economic OSs) and their particular representatives have been provided and described in terms of the mathematical model proposed. Of course, the range of currently known control models of OSs (with active agents and a controlled subsystem of a certain nature) is much wider. In this context, we should mention production systems [32, 33], organizational and technical systems [33], project management [33], military operations [34], and others.

For the sake of simplicity, the model has been given in a deterministic formulation. The functioning of real OSs is inevitably accompanied by significant uncertain factors. In view of the problems addressed above, the consideration of uncertainty will only complicate the understanding. However, it undoubtedly forms the first important direction of further research.

The second natural direction of refining the model is to consider the bounded rationality of active agents (H. Simon, R. Heiner, R. Selten, D. Kahneman, A. Tversky, etc.).

The third direction is the study of OSs with multiple Principals (the case of distributed control) as well as multilevel OSs.

Note that the controlled subsystem of an OS can be of technical, biological, economic, or other nature. Considering the specifics of the controlled subsystem, determined by the characterization of its state, the rule of state change, and other elements of the model, is essential when designing and implementing control mechanisms for OSs. Despite the general character of the regularities of control processes and the structure of control mechanisms, in practice, it is impossible to control an OS successfully without understanding its technical and economic specifics. Therefore, the model description of the controlled subsystem dynamics and its viability conditions is crucial in TCO along with the consideration and coordination of interests of its active agents.

Acknowledgments. *This work was supported by the Russian Science Foundation, project no. 25-11-00094.*

REFERENCES

1. *Filosofskii entsiklopedicheskii slovar'* (Philosophical Encyclopedic Dictionary), Moscow: Sovetskaya entsiklopediya, 1983. (In Russian.)
2. Belov, M.V. and Novikov, D.A., *Methodology of Complex Activity: Foundations of Understanding and Modelling*, Cham: Springer, 2020.
3. Burkov, V.N. and Novikov, D.A., *Teoriya aktivnykh sistem: sostoyanie i perspektivy* (Theory of Active Systems: State-of-the-Art and Prospects), Moscow: SINTEG, 1999. (In Russian.)
4. Novikov, D., *Theory of Control in Organizations*, New York: Nova Science, 2013.
5. Germeier, Yu.B., *Vvedenie v teoriyu issledovaniya operatsii* (Introduction to the Theory of Operations Research), Moscow: Nauka, 1971. (In Russian.)
6. Germeier, Yu., *Non-antagonistic Games*, Dordrecht: D. Reidel Publishing, 1986.
7. Kukushkin, N.S. and Morozov, V.V., *Teoriya neantagonisticheskikh igr* (Theory of Nonantagonistic Games), Moscow: Moscow State University, 1984. (In Russian.)
8. Gorelik, V.A., Gorelov, M.A., and Kononenko, A.F., *Analiz konfliktnykh situatsii v sistemakh upravleniya* (Analysis of Conflict Situations in Control Systems), Moscow: Radio i Svyaz', 1991. (In Russian.)
9. Laffont, J.-J. and Martimort, D., *The Theory of Incentives: The Principal-Agent Model*, Princeton: Princeton University Press, 2002.
10. Ougolnitsky, G.A., *Upravlenie ustoichivym razvitiem aktivnykh sistem* (Sustainable Management of Active Systems), Rostov-on-Don: Southern Federal University, 2016. (In Russian.)
11. Ougolnitsky, G.A., Methodology and Applied Problems of the Sustainable Management in Active Systems, *Control Sciences*, 2019, no. 2, pp. 19–29. (In Russian.)
12. Ougolnitsky, G., Gorbaneva, O., Usov, A., et al., Theory of Sustainable Management in Active Systems, *Large-Scale Systems Control*, 2020, no. 84, pp. 89–113. (In Russian.)
13. Ereshko, F.I., Application of Hierarchical Games to Control Organizational Systems, *Trudy 14-go Vserossiiskogo soveshchaniya po problemam upravleniya* (Proceedings of the 14th All-Russian Meeting on Control Problems), Novikov, D.A., Ed., Moscow: Trapeznikov Institute of Control Sciences RAS, 2024, pp. 3314–3318. (In Russian.)
14. Slovokhotov, Yu.L., *Fizika obshchestva: Primenenie fizicheskikh modelei v opisanii obshchestvennykh yavlenii* (The Physics of Society: Application of Physical Models to the Description of Social Phenomena), Moscow: LENAND, 2024. (In Russian.)
15. Ougolnitsky, G. and Gorbaneva, O., Sustainable Management in Active Networks, *Mathematics*, 2022, vol. 10, no. 16, art. no. 2848.
16. Novikov, D.A. and Chkhartishvili, A.G., *Reflexion and Control: Mathematical Models*, Leiden: CRC Press, 2014.
17. Burkov, V.N., Enaleev, A.K., and Korgin, N.A., Incentive Compatibility and Strategy-Proofness of Mechanisms of Organizational Behavior Control: Retrospective, State of the Art, and Prospects of Theoretical Research, *Automation and Remote Control*, 2021, vol. 82, no. 7, pp. 1119–1143.
18. Shchepkin, A.V., Dynamic Active Systems with Far-sighted Elements. I. The Dynamic Model of an Active System, *Avtom. Telemekh.*, 1986, no. 10, pp. 89–94. (In Russian.)
19. Ugolnickiy, G.A. and Chepel, E.N., Game-Theoretic Approach to Managing the Composition and Structure of a Bearing-Only Measurement System in Conditions of A Priori Uncertainty, *Journal of Computer and Systems Sciences International*, 2024, vol. 63, no. 2, pp. 343–356.
20. Agiev, Kh.R., Model of Queuing in the System of Logistic Interaction of the Railway with the Seaport, *Engineering Journal of Don*, 2022, no. 6. URL: http://ivdon.ru/ru/magazine/archive/n6y2022/7733. (In Russian.)
21. Demenskiy, V.I., Malsagov, M.K., and Agiev, K.R., A Model of Interaction between Two Types of Transport in Multimodal

Transport, *Engineering Journal of Don*, 2023, no. 7. URL: http://ivdon.ru/ru/magazine/archive/n7y2023/8528. (In Russian.)
22. Agiev, Kh.R., Malsagov, M.Kh., Ougolnitsky, G.A., Dynamic Models of the Coordination of Interests of the Active Participants of Logistic Interactions. Part 1, *Sistemy Upravleniya i Informatsionnye Tekhnologii*, 2024, no. 1 (95), P. 31–36. (In Russian.)
23. Agiev, Kh.R., Malsagov, M.Kh., Ougolnitsky, G.A. Dynamic Models of the Coordination of Interests of the Active Participants of Logistic Interactions. Part 2, *Sistemy Upravleniya i Informatsionnye Tekhnologii*, 2024, no. 3 (97), pp. 30–34. (In Russian.)
24. Agiev, Kh.R., Queuing Organizational System "Railway - Marine Port," *Engineering Journal of Don*, 2025, no. 4. URL: http://www.ivdon.ru/ru/magazine/archive/n4y2025/10019. (In Russian.)
25. Burkov, V.N., Novikov, D.A., and Shchepkin, A.V., *Control Mechanisms for Ecological-Economic Systems*, Cham: Springer, 2015.
26. Ugol'nitskii, G.A., *Upravlenie ekologo-ekonomicheskimi sistemami* (Control of Ecological-Economic Systems), Moscow: Vuzovskaya Kniga, 1999. (In Russian.)
27. Mazalov, V.V., Rettieva, A.N. Asymmetry in a Cooperative Bioresource Management Problem, *Large-Scale Systems Control*, 2015, no. 55, pp. 280–325.
28. Rettieva, A., Cooperation Maintenance in Dynamic Discrete-Time Multicriteria Games with Application to Bioresource Management Problem, *J. of Computational and Applied Mathematics*, 2024, vol. 441, no. 8, art. no. 115699.
29. Grune, L., Kato, M., and Semmler, W., Solving Ecological Management Problems Using Dynamic Programming, *J. of Economic Behavior and Organization*, 2005, vol. 57, no. 4, pp. 448–473.
30. Fanokoa, P.S., Telahigue, I., and Zaccour, G., Buying Cooperation in an Asymmetric Environmental Differential Game, *J. of Economic Dynamics and Control*, 2011, vol. 35, no. 6, pp. 935–946.
31. Sukhinov, A.I, Ougolnitsky, G.A., and Usov, A.B., Methods of Solving the Theoretic Game Models for Coordinating Interests in Regulating the Fishery Industry, *Mathematical Models and Computer Simulations*, 2020, vol. 12, pp. 176–184.
32. Burkov, V.N., Goubko, M.V., Kondrat'ev, V.V., Korgin, N.A., and Novikov, D.A., *Mechanism Design and Management: Mathematical Methods for Smart Organizations*, New York: Nova Science, 2013.
33. Belov, M.V. and Novikov, D.A., *Optimal Enterprise: Structures, Processes and Mathematics of Knowledge, Technology and Human Capital*, Boca Raton: CRC Press, 2021.
34. *Modeli voennykh, boevykh i spetsial'nykh deistvii* (Models of Military, Combat, and Special Operations), Novikov, D.A., Ed., Moscow: Lenand, 2025. (In Russian.)

This paper was recommended for publication by RAS Academician D.A. Novikov, a member of the Editorial Board.

Received May 27, 2025, and revised June 19, 2025. Accepted June 24, 2025.

Author information

Ougolnitsky, Gennady Anatol'evich. Dr. Sci. (Phys.–Math.), Southern Federal University, Rostov-on-Don, Russia
✉ gaugolnickiy@sfedu.ru
ORCID iD: https://orcid.org/0000-0001-5085-5144

Cite this paper

Ougolnitsky, G.A., The Essence, Attributes, and Description Principle of Organizational Systems. *Control Sciences* **3,** 32–40 (2025).

Original Russian Text © Ougolnitsky, G.A., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 38–48.

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

COMPOSITIONS OF TWO CONSTANT-WEIGHT CODES WITH ORTHOGONAL COMBINATIONS OVER ALL BITS FOR SELF-CHECKING DISCRETE DEVICE DESIGN

D. V. Efanov

Peter the Great Saint Petersburg Polytechnic University, St. Petersburg, Russia
Russian University of Transport, Moscow, Russia

✉ TrES-4b@yandex.ru

Abstract. This paper proposes using compositions of two constant-weight codes with orthogonal combinations over all bits in the design of controllable and self-checking discrete devices. With such codes, computation control at the outputs of discrete devices can be implemented via the attribute of belonging of the codewords to a given constant-weight code and, moreover, via the attribute of belonging of each function describing the codeword bits to the class of self-dual Boolean functions. It is shown how to construct noninterference codes based on the composition of two constant-weight codes with orthogonal combinations over all bits. Explicit formulas are derived to determine the number of errors undetectable by compositions of constant-weight codes by their types (by the number of erroneous 0s and 1s in codewords) and multiplicities. The properties of the codes under consideration are briefly described. The structure of concurrent error-detection circuits is presented for discrete devices based on the composition of two constant-weight codes with orthogonal combinations over all bits and computation control via two diagnostic attributes. The use of such compositions can be effective in building highly reliable discrete devices on various components.

Keywords: controllable and self-checking devices; computation control via two diagnostic attributes; compositions of constant-weight codes; error detection at discrete device outputs.

INTRODUCTION

Constant-weight codes, also called "*r*-out-of-*n*" or *r/n* codes, are constructed by selecting, from a set of codewords with *n* bits (with a codeword length of *n*), those having the same weight *r*. The number of codewords in an *r/n* code is determined by the binomial coefficient C_n^r. These codes were described in [1] and found wide application in data processing and transmission [2, 3] as well as in the design of controllable and self-checking discrete devices [4–7]. The design theory of detectors and checkers of constant-weight codes is rather deeply developed [8–12]; special properties and characteristics of these codes were investigated in several works, e.g., in the paper [13]. Note also that constant-weight codes are closely related to the theory of combinatorial block design and Steiner systems [14, 15].

We focus the reader's attention on the application of constant-weight codes to build devices with fault detection [16, 17]. When designing such devices using constant-weight codes, the following advantages are utilized. First of all, since the codewords of these codes have the same weight, they detect any errors in the codewords except for multidirectional errors with even multiplicity containing the distortion group $\{0\rightarrow 1,\ 1\rightarrow 0\}$ (the so-called symmetrical errors). Constant-weight codes detect any non-symmetrical errors classified as unidirectional (associated with distortions of exclusively 0s or exclusively 1s) and asymmetrical (containing an unequal number of distortions of 0s and 1s). This feature is used in the design of self-checking devices by searching for groups of unidirectional-independent (UI) outputs [18, 19] or groups of unidirectional/asymmetrical-independent (UAI) outputs [20, 21] in a device and organizing computation control at these outputs by means of con-

stant-weight codes. An alternative is to modify the structure of a controlled device into one with a single group of UI outputs or UAI outputs [22]. Another fruitful property is that constant-weight codes do not detect a small number of errors with even multiplicity, which also contributes to covering a large number of real distortions at the outputs of controlled devices. The third important advantage of constant-weight codes over other codes is the simplicity of control equipment (checkers, detectors, and check logic blocks) and clear principles of ensuring self-checking [9].

Different approaches can be applied to build self-checking discrete devices using constant-weight codes. It is interesting to control computations using not only constant-weight codes but also the properties of Boolean functions describing the bits of their codewords, namely, the properties of self-dual Boolean functions. Such an organization of computation control significantly improves controllability indicators in the part of observability [23]. When building discrete devices with computation control via two diagnostic attributes, only $\frac{n}{2}/n$ codes can be applied [24]. In this paper, the idea is to design self-checking discrete devices using compositions of two constant-weight codes with orthogonal combinations over all bits, which significantly extends the design theory of self-checking discrete devices.

1. COMPOSITIONS OF TWO CONSTANT-WEIGHT CODES WITH ORTHOGONAL COMBINATIONS OVER ALL BITS

Among all constant-weight codes, it is possible to separate those with the following interesting property: the set of their codewords contains exclusively the pairs of orthogonal codewords over all bits. Such codes exist only for even n, and there is a single code of this type with the value $r = \frac{n}{2}$ for each even n. The whole variety of constant-weight codes can be illustrated on Pascal's triangle, where each number characterizes the cardinality of the codeword set of r/n codes; see a fragment of Pascal's triangle in Fig. 1. That is, Fig. 1a shows the triangle with highlighted numbers corresponding to the cardinalities of the codeword sets of constant-weight codes with the above property. For example, the constant-weight 1/2 code contains the two combinations {01, 10} and is used in information coding for error detection, e.g., in the spatial two-rail representation of signals [25]. The constant-weight 2/4 code contains the six combinations {0011, 0101, 0110, 1001, 1010, 1100} and is used in organizing concurrent error-detection (CED) circuits for the combinational components of discrete devices [26].

In [24], it was proposed to use (constant-weight) $\frac{n}{2}/n$ codes in the design of self-checking discrete devices with computation control via two diagnostic attributes, namely, the belonging of codewords to a given constant-weight code and the belonging of each function describing the bits of codewords to the class of self-dual Boolean functions. For this purpose, the following property must be provided in the design process of a CED circuit: on the sets of argument values orthogonal over all bits, it is required to form, in the CED circuit, orthogonal codewords over all bits for constant-weight codes.

The studies of the applicability of constant-weight codes to computation control at discrete device outputs via two diagnostic attributes have led to the task of determining the applicability of groups of other constant-weight codes to CED circuit organization.

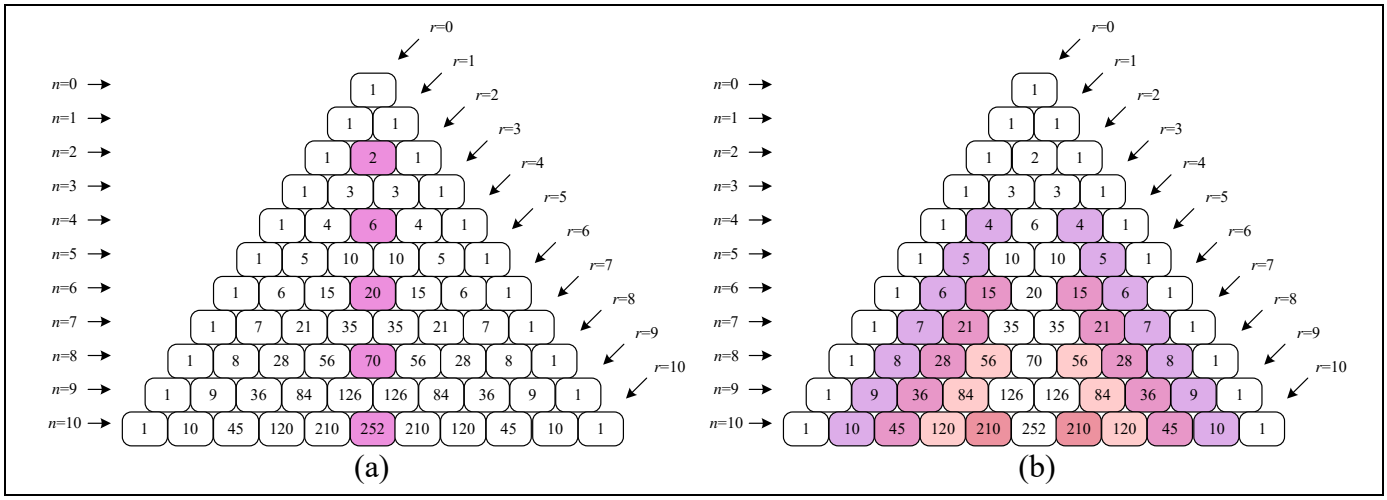

Fig. 1. Ways to select a) one and b) two constant-weight codes with orthogonal codewords over all bits on Pascal's triangle.

It would ensure higher variability in choosing CED circuit design methods and adjusting the key performance indicators of self-checking discrete device implementations.

Combining the codeword sets of constant-weight *r/n* and (*n* – *r*)*/n* codes, where $r \in \left\{1, 2, ..., \left\lfloor \frac{n}{2} \right\rfloor \right\}$ is the weight of a codeword, yields a composition of the pair of constant-weight codes with the desired property of their codewords. The binomial coefficients corresponding to such pairs of constant-weight codes are highlighted by the same color on Pascal's triangle in Fig. 1b. The non-highlighted ones are the codes with *r* = 0, the degenerate pairs (for which the weight is $r = \frac{n}{2}$ for even *n*), and the pairs of codes with the weights differing by 1.

Proposition 1. *A composition of pairs of r/n and (n – r)/n codes will detect any single errors in codewords if*

$$n - 2r \geq 2. \quad (1)$$

P r o o f. The codewords of an *r/n* code have weight *r*; the codewords of an (*n* – *r*)*/n* code, weight *n* – *r*. Clearly, *r/n* and (*n* – *r*)*/n* codes will detect any single errors distorting codewords belonging to the given code into those of the same code. Only single errors distorting codewords of an *r/n* code into those of an (*n* – *r*)*/n* code may be undetected. This is possible only if the weight of the codewords of an (*n* – *r*)*/n* code exceeds that of an *r/n* code by 1. In other words, $(n-r)-r=1$, which is equivalently written as $n-2r=1$. Hence, inequality (1) is valid for detecting any single errors. ♦

Thus, we obtain a whole group of modified codes based on pairs of constant-weight codes. They can be applied to design some controllable and self-checking discrete devices as well as to provide their diagnostic support.

2. CHARACTERISTICS OF ERROR DETECTION BY COMPOSITIONS OF TWO CONSTANT-WEIGHT CODES

We determine the number of errors that will not be detected by compositions of two constant-weight codes as well as characterize them.

Proposition 2. *The total number of codeword errors undetectable by a selected composition of pairs of r/n and (n – r)/n codes is given by*

$$N_{ND} = 2C_n^r \left(2C_n^r - 1\right). \quad (2)$$

P r o o f. The codeword set of an *r/n* code has cardinality C_n^r; the codeword set of an (*n* – *r*)*/n* code, cardinality $C_n^{n-r} = C_n^r$. Hence, the composition is formed by $2C_n^r$ codewords. An error will not be detected if and only if it distorts a codeword of a given composition of codes into a codeword belonging to codes from this composition. For each codeword, there are $2C_n^r - 1$ such distortions in total. Therefore, we arrive at the expression (2). ♦

Let (*r/n* + (*n* – *r*)*/n*) codes denote a composition of pairs of the corresponding constant-weight codes. For the first values of *n* as an example, they are given in Table 1. Here, the color indicates constant-weight $\frac{n}{2}/n$ codes, which also have pairs of orthogonal combinations over all bits in the codeword set.

Table 2 characterizes errors undetectable by (*r/n* + (*n* – *r*)*/n*) codes. For each code, two numbers are

Table 1

Compositions of pairs of constant-weight codes

n	*r*					
	1	2	3	4	5	6
4	1/4 + 3/4	2/4	–	–	–	–
5	1/5 + 4/5	–	–	–	–	–
6	1/6 + 5/6	2/6 + 4/6	3/6	–	–	–
7	1/7 + 6/7	2/7 + 5/7	–	–	–	–
8	1/8 + 7/8	2/8 + 6/8	3/8 + 5/8	4/8	–	–
9	1/9 + 8/9	2/9 + 7/9	3/9 + 6/9	–	–	–
10	1/10 + 9/10	2/10 + 8/10	3/10 + 7/10	4/10 + 6/10	5/10	–
11	1/11 + 10/11	2/11 + 9/11	3/11 + 8/11	4/11 + 7/11	–	–
12	1/12 + 11/12	2/12 + 10/12	3/12 + 9/12	4/12 + 8/12	5/12 + 7/12	6/12
13	1/13 + 12/13	2/13 + 11/13	3/13 + 10/13	4/13 + 9/13	5/13 + 8/13	–
14	1/14 + 13/14	2/14 + 12/14	3/14 + 11/14	4/14 + 10/14	5/14 + 9/14	6/14 + 8/14
15	1/15 + 14/15	2/15 + 13/15	3/15 + 12/15	4/15 + 11/15	5/15 + 10/15	6/15 + 9/15

Table 2

Characterization of errors undetectable by ($r/n + (n - r)/n$) codes

n	The number of undetectable errors and their share in the total number of errors in codewords					
	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$
4	56 23.33333%	30 12.5%	–	–	–	–
5	90 9.07258%	–	–	–	–	–
6	132 3.27381%	870 21.57738%	380 9.4246%	–	–	–
7	182 1.11959%	1722 10.59301%	–	–	–	–
8	240 0.36765%	3080 4.71814%	12 432 19.04412%	4830 7.3989%	–	–
9	306 0.11696%	5112 1.95389%	28 056 10.72346%	–	–	–
10	380 0.03628%	8010 0.76464%	57 360 5.47562%	175 980 16.79917%	63 252 6.03808%	–
11	462 0.01102%	11 990 0.286%	108 570 2.58978%	434 940 10.37484%	–	–
12	552 0.00329%	17 292 0.10309%	193 160 1.1516%	979 110 5.83738%	2 507 472 14.94935%	852 852 5.08464%
13	650 0.00097%	24 180 0.03604%	326 612 0.48675%	2 043 470 3.04538%	6 622 902 9.8701%	–
14	756 0.00028%	32 942 0.01227%	529 256 0.19718%	4 006 002 1.49244%	16 028 012 5.97126%	36 066 030 13.43646%
15	870 0.00008%	43 890 0.00409%	827 190 0.07704%	7 450 170 0.69387%	36 066 030 3.35901%	100 190 090 9.33121%

specified (top and bottom in cells), namely, the number of undetectable errors and the share of undetectable errors in the total number of errors in the codewords (γ_n, in %). Compositions of two constant-weight codes detect a significant number of errors. With increasing n for each fixed r, the share of undetectable errors gradually drops (Fig. 2).

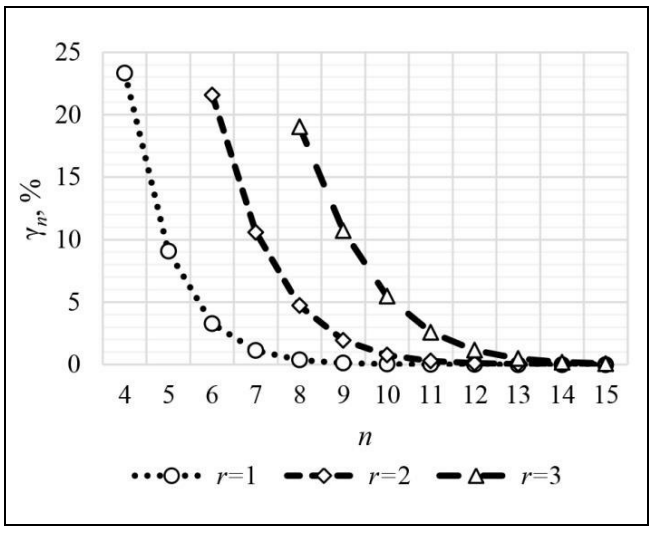

Fig. 2. The dependences of γ_n on n for different ($r/n + (n - r)/n$) codes.

Now we characterize the errors undetectable by different ($r/n + (n - r)/n$) codes by their types (unidirectional, symmetrical and asymmetrical) and multiplicities d.

First, consider ($1/n + (n - 1)/n$) codes.

Let us take the first (1/4 + 3/4) code and choose an arbitrary codeword of the 1/4 code, e.g., <0001>. It can be distorted into codewords belonging to the composition of these codes in $2C_4^1 - 1 = 7$ ways. The number of distortions of the codeword <0001> into the codewords belonging to the 1/4 code is $C_4^1 - 1 = 3$. Since the codeword weight is preserved, all these distortions will be double symmetrical distortions. The codeword <0001> can be distorted into codewords belonging to the 3/4 code in $C_4^3 = 4$ ways. Moreover, if after the distortion the single 1 in the codeword preserves its value, the error will be unidirectional. As the codewords of the 3/4 code have weight $n - r = 3$, there are $C_{4-1}^{3-1} = C_3^2 = 3$ possible ways of such distortions. The multiplicity of unidirectional undetectable errors will equal 2. The remaining single codeword <1110> of the 3/4 code is orthogonal over all bits to the codeword <0001>. The latter can be distorted into the former only under a quadruple asymmetrical error.

The same fact takes place for the other codewords of (1/4 + 3/4) codes.

Proceeding to the second (1/5 + 4/5) code, we choose an arbitrary codeword of the 1/5 code, e.g., <00001>. It can be distorted into codewords belonging to the composition of these codes in $2C_5^1 - 1 = 9$ ways. The number of distortions of the codeword <00001> into the codewords belonging to the 1/5 code is $C_5^1 - 1 = 4$. Due to preserving the codeword weight, all these distortions will be double symmetrical distortions. The codeword <00001> can be distorted into codewords belonging to the 4/5 code in $C_5^4 = 5$ ways. Moreover, if after the distortion the single 1 in the codeword preserves its value, the error will be unidirectional. As the codewords of the 4/5 code have weight $n - r = 4$, there are $C_{5-1}^{4-1} = C_4^3 = 4$ possible ways of such distortions. The multiplicity of unidirectional undetectable errors will equal 3. The remaining single codeword <11110> of the 4/5 code is orthogonal over all bits to the codeword <00001>. The latter can be distorted into the former only under a quintuple asymmetrical error. The other codewords of the (1/5 + 4/5) codes are considered by analogy.

Continuing the series of considerations, we arrive at the following results, valid for (1/n + (n – 1)/n) codes:

– For each codeword, there are n – 1 possible ways to distort it into a codeword belonging only to this code, characterized by a double symmetrical error.

– For each codeword, there are n – 1 possible ways to distort it into a codeword belonging to the second code in the composition, characterized by an (n – 2r) = (n – 2)-tuple unidirectional error.

– For each codeword, there is a single possible way to distort it into a codeword belonging to the second code in the composition, characterized by an n-tuple asymmetrical error.

There exist $2C_n^1 = 2n$ codewords in total; therefore, each of the above numbers, for each error type and multiplicity, should be multiplied by this value to obtain the total number of undetectable errors by type and multiplicity.

Let us represent the total number of undetectable errors as

$$N_{ND} = N_{\upsilon,d}\upsilon_d + N_{\sigma,d}\sigma_d + N_{\alpha,d}\alpha_d, \qquad (3)$$

with the following notation: $N_{\upsilon,d}$, $N_{\sigma,d}$, and $N_{\alpha,d}$ are the numbers of undetectable unidirectional, symmetrical, and asymmetrical errors, respectively, with multiplicity d; the symbols υ_d, σ_d, and α_d indicate the belonging of undetectable errors to the class of unidirectional, symmetrical, and asymmetrical errors, respectively, with multiplicity d.

Using the expression (3) and the above reasoning, we derive the following formula for undetectable errors for (1/n + (n – 1)/n) codes:

$$\begin{aligned} N_{ND}^{1/n+(n-1)/n} &= 2n\left((n-1)\upsilon_{n-2} + (n-1)\sigma_2 + 1\alpha_n\right) \\ &= 2n(n-1)\left(\upsilon_{n-2} + \sigma_2\right) + 2n\alpha_n. \end{aligned} \qquad (4)$$

For example, for (1/4 + 3/4) codes, formula (4) gives

$$\begin{aligned} N_{ND}^{1/4+3/4} &= 2\cdot 4\cdot(4-1)\left(\upsilon_2 + \sigma_2\right) + 2\cdot 4\alpha_4 \\ &= 24\upsilon_2 + 24\sigma_2 + 8\alpha_4. \end{aligned}$$

Now we generalize this result to the composition of (2/n + (n – 2)/n) codes.

As an example, let us take the (2/6 + 4/6) code and choose an arbitrary codeword of the 2/6 code, e.g., <000011>. There are $2C_6^2 - 1 = 29$ its possible distortions into codewords belonging to this composition. Among them, there are $C_6^2 - 1 = 14$ distortions into the codewords of the 2/6 code due to double symmetrical errors. Also, there are $C_6^2 = 15$ distortions into the codewords of the 4/6 code. Such errors have the following structure. There is a single distortion into the codeword of the 4/6 code orthogonal over all bits to the codeword <111100>, due to a sextuple asymmetrical error. There are $C_{6-2}^2 = C_4^2 = 6$ distortions under which all 1s in the codeword <000011> preserve their positions and two 0s are distorted. Such distortions are caused by double unidirectional errors. Also, there are distortions under which a single 1 preserves its value in the codeword <000011>, a single 1 is distorted, and three 0s are distorted. The number of possible distortions of 1s and 0s described above are C_2^1 and C_4^3, respectively. In total, we have $C_4^3 C_2^1 = 8$ quadruple asymmetrical distortions.

This error structure is inherent in all codewords in the composition under consideration.

It is easy to generalize the results to (2/n + (n – 2)/n) codes:

– For each codeword, there are $C_n^2 - 1$ possible ways to distort it into a codeword belonging only to this code, characterized by a double symmetrical error.

– For each codeword, there are C_{n-2}^2 possible ways to distort it into a codeword belonging to the second code in the composition, characterized by an (n – 2r) = (n – 4)-tuple unidirectional error.

– For each codeword, there is a single possible way to distort it into a codeword belonging to the second code in the composition, characterized by an n-tuple asymmetrical error.

There exist $2C_n^2$ codewords in total; therefore, each of the above numbers, for each error type and multi-

plicity, should be multiplied by this value to obtain the total number of undetectable errors by type and multiplicity:

$$N_{ND}^{2/n+(n-2)/n} = 2C_n^2\left(C_{n-2}^2\upsilon_2 + \left(C_n^2-1\right)\sigma_2 + 2C_{n-2}^{n-3}\alpha_4 + 1\alpha_n\right) = 2C_n^2C_{n-2}^2\upsilon_2 + 2C_n^2\left(C_n^2-1\right)\sigma_2 + 4C_n^2C_{n-2}^{n-3}\alpha_4 + 2C_n^2\alpha_n. \quad (5)$$

For example, for (2/6 + 4/6) codes, formula (5) gives

$$N_{ND}^{2/6+4/6} = 2C_6^2C_4^2\upsilon_2 + 2C_6^2\left(C_6^2-1\right)\sigma_2 + 4C_6^2C_4^3\alpha_4 + 2C_6^2\alpha_6 = 180\upsilon_2 + 420\sigma_2 + 240\alpha_4 + 30\alpha_6.$$

The generalization to $(r/n + (n - r)/n)$ codes leads to the following:

– For each codeword, there are $C_n^r - 1$ possible ways to distort it into a codeword belonging only to this code, characterized by a double symmetrical error.

– For each codeword, there are C_{n-r}^r possible ways to distort it into a codeword belonging to the second code in the composition, characterized by an $(n - 2r)$-tuple unidirectional error.

– For each codeword, there are

$$C_r^1C_{n-r}^{(n-r)-(r-1)} + C_r^2C_{n-r}^{(n-r)-(r-2)} + \ldots + C_r^{r-1}C_{n-r}^{n-r-(r-(r-1))} + C_r^rC_{n-r}^{n-r-(r-r)} = C_r^1C_{n-r}^{n-2r+1} + C_r^2C_{n-r}^{n-2r+2} + \ldots + C_r^{r-1}C_{n-r}^{n-r-1} + C_r^rC_{n-r}^{n-r} = \sum_{i=1}^{r} C_r^iC_{n-r}^{(n-r)-(r-i)}$$

possible ways to distort it into a codeword belonging to the second code in the composition, characterized by asymmetrical errors. Note that the first term is the number of asymmetrical errors of multiplicity $n - 2r + 2$; the second term, the number of asymmetrical errors of multiplicity $n - 2r + 4$; ....; the $(r - 1)$th term, the number of asymmetrical errors of multiplicity $((n - r) - (r - (r - 1)) + (r - 1)) = (n - 2)$; finally, the rth term, the number of asymmetrical errors of multiplicity n.

The total number of codewords in the composition considered is $2C_n^r$.

Thus, we arrive at the general formula for the number of errors of different types and multiplicities undetectable by $(r/n + (n - r)/n)$ codes:

$$N_{ND}^{r/n+(n-r)/n} = 2C_n^r\left(C_{n-r}^r\upsilon_{n-2r} + \left(C_n^r-1\right)\sigma_2 + C_r^1C_{n-r}^{n-2r+1}\alpha_{n-2r+2} + C_r^2C_{n-r}^{n-2r+2}\alpha_{n-2r+4} + \ldots + C_r^{r-1}C_{n-r}^{n-r-1}\alpha_{n-2} + C_r^rC_{n-r}^{n-r}\alpha_n\right) = 2C_n^rC_{n-r}^r\upsilon_{n-2r} + 2C_n^r\left(C_n^r-1\right)\sigma_2 + 2C_n^rC_r^1C_{n-r}^{n-2r+1}\alpha_{n-2r+2} + 2C_n^rC_r^2C_{n-r}^{n-2r+2}\alpha_{n-2r+4} + \ldots + 2C_n^rC_r^{r-1}C_{n-r}^{n-r-1}\alpha_{n-2} + 2C_n^rC_r^rC_{n-r}^{n-r}\alpha_n. \quad (6)$$

Formulas (4) and (5) are special cases of (6).

The error detection characteristics of compositions of constant-weight codes can be used in practice to build discrete devices with fault detection.

3. COMPOSITIONS OF AN EVEN NUMBER OF CONSTANT-WEIGHT CODES

For $n \geq 8$, compositions of more than two constant-weight codes can be constructed in which the codeword set will contain only pairs of codewords orthogonal over all bits. Such codes are constructed for even values of $n \geq 8$. Desired combinations can be formed by uniting the codeword sets of the four constant-weight codes highlighted on Pascal's triangle (Fig. 3).

Table 3 shows the composition of quadruples of constant-weight codes, and Table 4 characterizes the errors undetectable by them. For $n > 8$, compositions of quadruple constant-weight codes do not detect hundredths and thousandths of a percent of the total number of errors occurring in codewords with the corresponding codeword length.

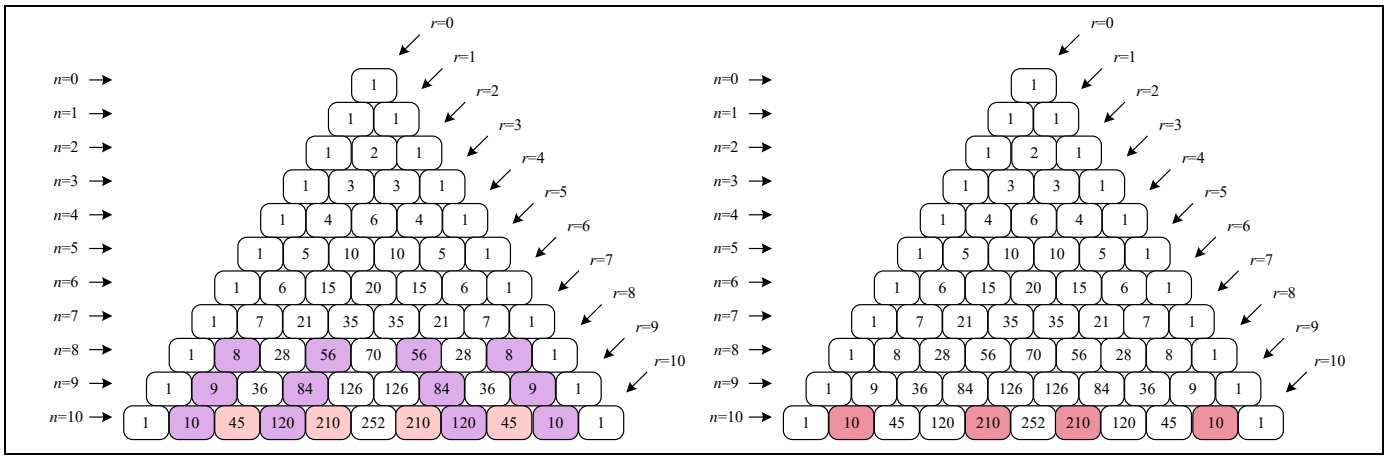

Fig. 3. Ways to select four constant-weight codes with orthogonal combinations over all bits on Pascal's triangle.

Table 3

Compositions of quadruples of constant-weight codes

n	r_1, r_2									
	r_1=1, r_2=3	r_1=1, r_2=4	r_1=1, r_2=5	r_1=1, r_2=6	r_1=2, r_2=4	r_1=2, r_2=5	r_1=2, r_2=6	r_1=3, r_2=5	r_1=3, r_2=6	r_1=4, r_2=6
8	1/8+7/8, 3/8+5/8	–	–	–	–	–	–	–	–	–
9	1/9+8/9, 3/9+6/9	–	–	–	–	–	–	–	–	–
10	1/10+9/10, 3/10+7/10	1/10+9/10, 4/10+6/10	–	–	2/10+8/10, 4/10+6/10	–	–	–	–	–
11	1/11+10/11, 3/11+8/11	1/11+10/11, 4/11+7/11	–	–	2/11+9/11, 4/11+7/11	–	–	–	–	–
12	1/12+11/12, 3/12+9/12	1/12+11/12, 4/12+8/12	1/12+11/12, 5/12+7/12	–	2/12+10/12, 4/12+8/12	2/12+10/12, 5/12+7/12	–	3/12+9/12, 5/12+7/12	–	–
13	1/13+12/13, 3/13+10/13	1/13+12/13, 4/13+9/13	1/13+12/13, 5/13+8/13	–	2/13+11/13, 4/13+9/13	2/13+11/13, 5/13+8/13	–	3/13+10/13, 5/13+8/13	–	–
14	1/14+13/14, 3/14+11/14	1/14+13/14, 4/14+10/14	1/14+13/14, 5/14+9/14	1/14+13/14, 6/14+8/14	2/14+12/14, 4/14+10/14	2/14+12/14, 5/14+9/14	2/14+12/14, 6/14+8/14	3/14+11/14, 6/14+8/14	3/14+11/14, 6/14+8/14	4/14+10/14, 6/14+8/14
15	1/15+14/15, 3/15+12/15	1/15+14/15, 4/15+11/15	1/15+14/15, 5/15+10/15	1/15+14/15, 6/15+9/15	2/15+13/15, 4/15+11/15	2/15+13/15, 5/15+10/15	2/15+13/15, 6/15+9/15	3/15+12/15, 6/15+9/15	3/15+12/15, 6/15+9/15	4/15+11/15, 6/15+9/15

Table 4

Characteristics of errors undetectable by compositions of quadruples of constant-weight codes

n	The number of undetectable errors and their share in the total number of errors in codewords									
	r_1=1, r_2=3	r_1=1, r_2=4	r_1=1, r_2=5	r_1=1, r_2=6	r_1=2, r_2=4	r_1=2, r_2=5	r_1=2, r_2=6	r_1=3, r_2=5	r_1=3, r_2=6	r_1=4, r_2=6
8	128 0.19608 %	–	–	–	–	–	–	–	–	–
9	186 0.07109 %	–	–	–	–	–	–	–	–	–
10	260 0.02482 %	440 0.042 %	–	–	510 0.04868 %	–	–	–	–	–
11	352 0.0084 %	682 0.01627 %	–	–	770 0.01837 %	–	–	–	–	–
12	464 0.00277 %	1014 0.00605 %	1608 0.00959 %	–	1122 0.00669 %	1716 0.01023 %	–	2024 0.01207 %	–	–
13	598 0.00089 %	1456 0.00217 %	2600 0.00387 %	–	1586 0.00236 %	2730 0.00407 %	–	3146 0.00469 %	–	–
14	756 0.00028 %	2030 0.00076 %	4032 0.0015 %	6034 0.00225 %	2184 0.00081 %	4186 0.00156 %	6188 0.00231 %	4732 0.00176 %	6734 0.00251 %	8008 0.00298 %
15	940 0.00009 %	2760 0.00026 %	6036 0.00056 %	10040 0.00094 %	2940 0.00027 %	6216 0.00058 %	10220 0.00095 %	6916 0.00064 %	10920 0.00102 %	12740 0.00119 %

The above considerations for determining the error detection characteristics of pairs of compositions of constant-weight codes by error types and multiplicities (see the previous section) can be repeated to characterize the errors undetectable by the compositions of quadruple constant-weight codes.

4. APPLICATION OF COMPOSITIONS OF TWO CONSTANT-WEIGHT CODES IN SELF-CHECKING DISCRETE DEVICE DESIGN

Figure 4 shows the structural diagram of a self-checking discrete device with a CED circuit based on an $(r/n + (n - r)/n)$ code. It uses two diagnostic attributes for computation control: the belonging of the codewords formed to a given $(r/n + (n - r)/n)$ code and the belonging of each function describing the check bits to the class of self-dual functions.

The object under diagnosis is a combinational discrete device $F(X)$ that calculates the values of Boolean functions $f_1(X)$, $f_2(X)$,…, $f_{n-1}(X)$, $f_n(X)$ when sets of argument values $<x_t\ x_{t-1}\ \ldots\ x_2\ x_1> = <X>$ are supplied to its inputs. The device $F(X)$ is augmented with a special CED circuit to control computation. In addition, the operation of this circuit is adjusted appropriately [27].

The outputs of the device $F(X)$ are connected to the inputs of a Boolean signal correction (BSC) block in the CED circuit. This block is intended to transform the output signals of the device $F(X)$ into signals $h_1(X)$, $h_2(X)$,…, $h_{n-1}(X)$, $h_n(X)$ on each set of argument values. The signal correction block is formed by a cascade of two-input XORs: their first inputs receive the signals $f_1(X)$, $f_2(X)$,…, $f_{n-1}(X)$, $f_n(X)$ whereas their second inputs the signals $g_1(X)$, $g_2(X)$,…, $g_{n-1}(X)$, $g_n(X)$ with the same subscripts from a block $G(X)$ calculating the Boolean correction functions:

$$h_i(X) = f_i(X) \oplus g_i(X), i = \overline{1, n}. \qquad (7)$$

The transformation (7) is performed on each set of argument values to endow the functions $h_1(X)$, $h_2(X)$,…, $h_{n-1}(X)$, $h_n(X)$ with the following two properties. First, the codeword $<h_n(X)\ h_{n-1}(X)\ \ldots\ h_2(X)\ h_1(X)>$ must belong to the $(r/n + (n - r)/n)$ code selected. Second, the values of each function $h_1(X)$, $h_2(X)$,…, $h_{n-1}(X)$, $h_n(X)$ on each set of argument values are generated so that the functions turn out to be self-dual. This requires only that orthogonal combinations over all bits belonging to the $(r/n + (n - r)/n)$ code be formed on orthogonal sets of argument values (see above). It is necessary to redefine the values of the vectors $<h_n(X)\ h_{n-1}(X)\ \ldots\ h_2(X)\ h_1(X)>$ on all sets of argument values so that all codewords belonging to the r/n code and $(n - r)/n$ code will be formed at least once. This feature requires the constraint

$$t \geq \lceil \log_2 2C_n^r \rceil.$$

The belonging of the codewords $<h_n(X)\ h_{n-1}(X)\ \ldots\ h_2(X)\ h_1(X)>$ to the $(r/n + (n - r)/n)$ code selected is verified by a totally self-checking checker (TSC). Such checkers can be designed in various ways, e.g., by separating two functions $z_1^0(X)$ and $z_1^1(X)$, each being the disjunction of conjunctions corresponding to the codewords of the r/n code and $(n - r)/n$ code, respectively. (By the way, they may appear in each of the functions, but only once!) It is possible to use bracketed forms to simplify

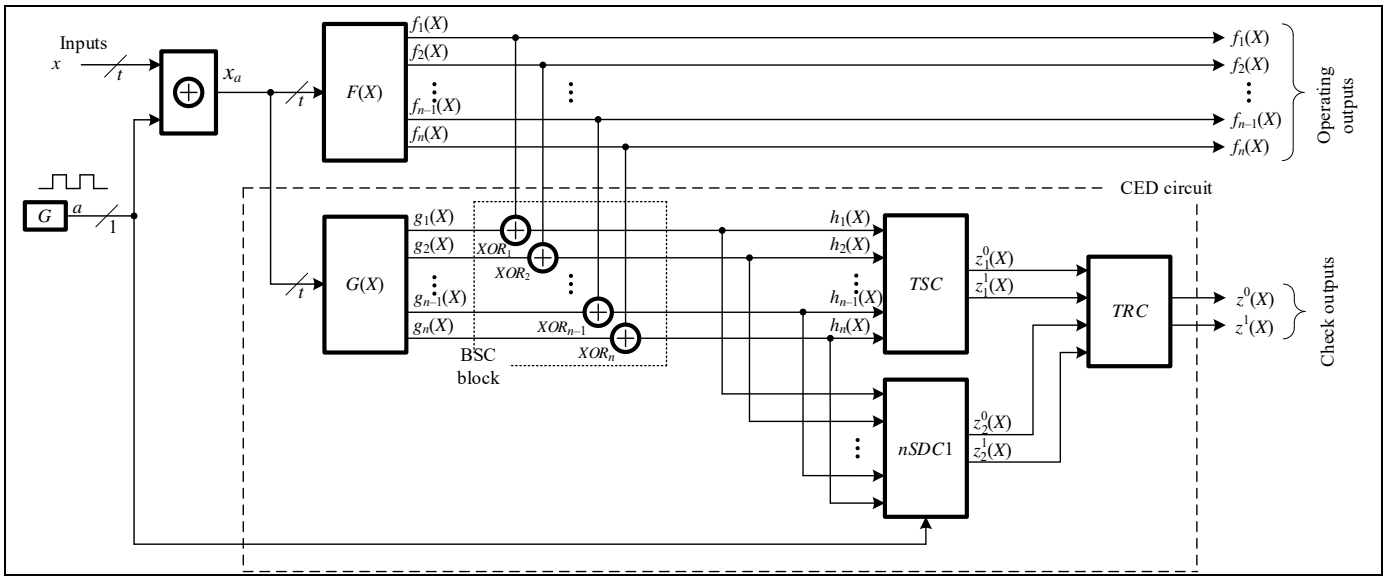

Fig. 4. The structural diagram of a self-checking discrete device.

checker structures. The design methods of constant-weight code checkers described in [9, 10] are not directly applicable to build checkers of their compositions.

The belonging of each function $h_1(X)$, $h_2(X)$,…, $h_{n-1}(X)$, $h_n(X)$ to the class of self-dual Boolean functions is checked using *nSDC*1, a device that receives n self-dual signals at its inputs and produces the values of two functions $z_2^0(X)$ and $z_2^1(X)$ at its outputs. Such a device can be implemented in two ways. The first solution is to install n elementary self-dual checkers (SDC) [27] and $n - 1$ elementary two-rail checkers (TRC) [28]. The second is based on the preliminary compression of n self-dual signals into one signal using three-input XORs (see the corresponding scheme in [29]) and control of the received signal using one SDC device.

The outputs $z_1^0(X)$ and $z_1^1(X)$, as well as $z_2^0(X)$ and $z_2^1(X)$, operate in the two-rail logic. Violation of the two-rail principle indicates a computational error, indirectly meaning the presence of faults in one of the blocks. The outputs $z_1^0(X)$ and $z_1^1(X)$, as well as $z_2^0(X)$ and $z_2^1(X)$, are connected to the inputs of one TRC block to produce a single reference signal $Z^0(X)$ and $Z^1(X)$.

According to the control principles for self-dual signals, it is necessary to compare sequentially the value on the direct combination supplied and on the orthogonal one over all bits, which actually reduces the performance twofold [30]. Then the first combination supplied is the operating one, and the second is the check combination. Thus, it is required to supply, to the inputs of the device, pairs of sets of argument values composed of operating combinations and check combinations orthogonal over all variables to the former. This is conveniently organized in the pulse operation mode of the structure shown in Fig. 4. The mode is implemented by installing a generator G of rectangular pulses with frequency $S = 2$, which produces the sequence a 0101...01 at its output. This sequence is supplied to the second inputs of two-input XORs, installed for each input $x_1, x_2,..., x_{t-1}, x_t$. The first inputs of XORs are fed with the values of the input variables themselves. With such a circuit, the Boolean zero signal is transformed into the sequence 0101...01 and the Boolean one signal into the sequence 1010...10. Thus, the sets of argument values arrive at the inputs of the block $F(X)$ in pairs, namely, the operating and check ones, the latter being orthogonal over all bits to the former. Also, the signal from the rectangular pulse generator is necessary for the operation of the checkers of self-dual signals (SDC) in the CED circuit.

Separate examples of CED circuit design by the above method are beyond the scope of this paper. Note that in practice, the use of $(r/n + (n - r)/n)$ codes in the CED circuit design according to Fig. 4 yields totally self-checking discrete devices with different performance indicators. First of all, it is important to cover errors at the outputs of the device $F(X)$, ensure the self-checking property of the CED circuit, and obtain the simplest CED circuit implementation in order to design less redundant structures compared to the duplication method. Given a high variability in obtaining the values of the functions $g_1(X)$, $g_2(X)$,…, $g_{n-1}(X)$, $g_n(X)$ on each set of argument values, it is possible to form sufficiently many structures and choose among them the simplest in terms of complexity. This indicator determines the device complexity [31]. For instance, the number of ways to redefine the function values essentially depends on the number t of inputs in the object under diagnosis $F(X)$. It suffices to redefine the function values on the first half of the sets, their number equals 2^{t-1}. On each of these sets, redefinition can be performed in $2C_n^r$ ways. On the second half of the sets of argument values, the function values are obtained based on the properties of self-dual functions: it is required to redefine the functions to opposite values on the sets orthogonal over all bits. In total, there are $2^t C_n^r$ ways to redefine the function values realized at the outputs of the block $G(X)$. Given many ways to build the blocks $G(X)$, one can choose the simplest implementations while ensuring the testability of CED circuit components. In many practical cases, it is possible to implement the block $G(X)$ simpler than $F(X)$, which significantly affects the hardware redundancy of CED circuits and allows building less redundant self-checking devices compared to duplication with improved controllability indicators [23]. Nevertheless, for some blocks $F(X)$, it is impossible to build a simpler implementation of this structure than in the case of duplication, which is a disadvantage of CED circuits with correction of all signals from the object under diagnosis [32]. According to studies, under certain conditions, the number of outputs of the block $G(X)$ can be reduced by using the same functions for the correction of different signals from the block $F(X)$. This approach works only in particular cases and has been underinvestigated so far. However, it can be considered promising for improving CED circuit design algorithms based on the structure in Fig. 4.

CONCLUSIONS

Using compositions of two constant-weight codes, r/n and $(n - r)/n$ with $r \in \left\{1, 2, ..., \left\lfloor \frac{n}{2} \right\rfloor\right\}$ and condition (1) allows significantly expanding the number of ways to organize controllable and self-checking discrete

devices with computation control via two diagnostic attributes in comparison with the application of only $\frac{n}{2}/n$ codes with even n. Compositions of constant-weight codes with orthogonal combinations over all bits can be separated out for $n \geq 4$. With increasing n, the number of codewords in such codes grows accordingly, and the number of ways to select exactly two constant-weight codes sequentially increases by 1 for each even n, starting from $n = 4$. For $n \geq 8$, it is possible to construct a composition of an even number of constant-weight codes with orthogonal combinations over all bits.

In a wide class of compositions of constant-weight codes with orthogonal combinations over all bits, an appropriate composition can be selected for a given number of outputs of the object under diagnosis and for a possible separation of groups of outputs with their smaller number to organize separate computation control subcircuits. The application of compositions of constant-weight codes with small n may be of interest due to simpler checkers and simpler conditions for ensuring their total self-checking property.

Further research can be connected with developing CED circuit design methods based on compositions of constant-weight codes with minimization of various quality indicators (structural redundancy, controllability, power consumption, etc.). Practical applications of these codes can be interesting, especially in the areas of science and technology with rarely changing input data. (Such systems are widespread and include, e.g., air defense systems, electrical interlocking systems in railway transport, control systems in the nuclear industry, etc.). In these systems, operational diagnosis and the readiness of devices are very topical tasks [33, 34].

The principles to extract compositions of constant-weight codes described in this paper, as well as their properties, may be in demand when building highly reliable discrete systems on modern and promising components [35, 36].

REFERENCES

1. Freiman, C.V., Optimal Error Detection Codes for Completely Asymmetric Binary Channels, *Information and Control*, 1962, vol. 5, no. 1, pp. 64–71. DOI: 10.1016/S0019-9958(62)90223-1
2. Tallini, L.G. and Bose, B., Design of Balanced and Constant Weight Codes for VLSI systems, *IEEE Transactions on Computers*, 1998, vol. 47, no. 5, pp. 556–572. DOI: 10.1109/12.677239
3. Butler, J.T. and Sasao, T., Index to Constant Weight Codeword Converter, *Proceedings of 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Seoul, South Korea, 2011. DOI: 10.1109/MWSCAS.2011.6026312
4. Samofalov, K.G., Romankevich, A.M., Valuiskii, V.N., et al., *Prikladnaya teoriya tsifrovykh avtomatov* (Applied Theory of Digital Automata), Samofalov, K.G., Ed., Kiev: Vishcha Shkola, 1987. (In Russian.)
5. Sogomonyan, E.S. and Slabakov, E.V., *Samoproveryaemye ustroistva i otkazoustoichivye sistemy* (Self-Checking Devices and Fault-Tolerant Systems), Moscow: Radio i Svyaz', 1989. (In Russian.)
6. Göessel, M., Ocheretny, V., Sogomonyan, E., and Marienfeld, D., *New Methods of Concurrent Checking*, 1st ed., Dordrecht: Springer Science+Business Media, 2008.
7. Das, D.K., Roy, S.S., Dmitiriev, A., et al. Constraint Don't Cares for Optimizing Designs for Concurrent Checking by 1-out-of-3 Codes, *Proceedings of the 10th International Workshops on Boolean Problems*, Freiberg, Germany, 2012, pp. 33–40.
8. Sapozhnikov, V.V. and Sapozhnikov, Vl.V., Self-Checking Checkers for Constant-weight codes, *Autom. Remote Control*, 1992, vol. 53, no. 3, pp. 321–348.
9. Sapozhnikov, V.V. and Sapozhnikov, Vl.V., *Samoproveryaemye diskretnye ustroistva* (Self-Checking Discrete Devices), St. Petersburg: Energoatomizdat, 1992. (In Russian.)
10. Piestrak, S.J., *Design of Self-Testing Checkers for Unidirectional Error Detecting Codes*, Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 1995.
11. Tarnik, M., Design of Embedded Constant Weight Code Checkers Based on Averaging Operations, *Proceedings of 2010 IEEE 16th International On-Line Testing Symposium*, Corfu, Greece, 2010. DOI: 10.1109/IOLTS.2010.5560193
12. Matrosova, A.Yu., Butorina, N.B., and Yakimova, N.O., Checker Design Based on Monotonous Function Implementation, *Izvestiya Vuzov. Fizika*, 2013, vol. 56, no. 9-2, pp. 171–173. (In Russian.)
13. Efanov, D.V., Some Features of Error Detection by Uniform Indivisible Codes, *Journal of Instrument Engineering*, 2019, vol. 62, no. 7, pp. 621–631. DOI: 10.17586/0021-3454-2019-62-7-621-631. (In Russian.)
14. Korshunov, A.D., Some Unsolved Problems in Discrete Mathematics and Mathematical Cybernetics, *Russian Math. Surveys*, 2009, vol. 64, no. 5, pp. 787–803. https://doi.org/10.1070/RM2009v064n05ABEH004640.
15. Mozharov, G.P., Fault-Tolerant Computer Networks Constructed on the Basis of Combinatory Block Designs, *Herald of the Bauman Moscow State Tech. Univ. Ser. Instrum. Eng.*, 2016, no. 6, pp. 41–53. DOI: 10.18698/0236-3933-2016-6-41-53. (In Russian.)
16. Chioktour, V. and Kakarountas, A., Adaptive BIST for Concurrent On-Line Testing on Combinational Circuits, *Electronics*, 2022, vol. 19, art. no. 3193. DOI: 10.3390/electronics11193193
17. Sahana, A.R., Chiraag, V., Suresh, G., et al., Application of Error Detection and Correction Techniques to Self-Checking VLSI Systems: An Overview, *Proceedings of 2023 IEEE Guwahati Subsection Conference (GCON)*, Guwahati, 2023. DOI: 10.1109/GCON58516.2023.10183449
18. Slabakov, E.V. and Sogomonyan, E.S., Design of Completely Self-Checking Combinational Circuits with the Use of Constant-weight codes, *Autom. Remote Control*, 1980, vol. 41, no. 9, pp. 1326–1333.
19. Saposhnikov, V.V., Morosov, A., Saposhnikov, Vl.V., and Göessel, M., A New Design Method for Self-Checking Unidirectional Combinational Circuits, *Journal of Electronic Testing: Theory and Applications*, 1998, vol. 12, no. 1-2, pp. 41–53. DOI: 10.1023/A:1008257118423

20.Efanov, D.V., Sapozhnikov, V.V., and Sapozhnikov, Vl.V., Conditions for Detecting a Logical Element Fault in a Combination Device under Concurrent Checking Based on Berger's Code, *Autom. Remote Control*, 2017, vol. 78, no. 5, pp. 891–901.
21.Efanov, D.V., Sapozhnikov, V.V., Sapozhnikov, V.V., Synthesis of Self-Checking Combination Devices Based on Allocating Special Groups of Outputs, *Autom. Remote Control*, 2018, vol. 79, no. 9, pp. 1609–1620. https://doi.org/10.1134/S0005117918090060.
22.Efanov, D.V., Sapozhnikov, V.V., and Sapozhnikov, Vl.V., Organization of a Fully Self-Checking Structure of a Combinational Device Based on Searching for Groups of Symmetrically Independent Outputs, *Automatic Control and Computer Sciences*, 2020, vol. 54, no. 4, pp. 279–290. DOI: 10.3103/S0146411620040045.
23.Efanov, D.V. and Pogodina, T.S., Properties Investigation of Self-Dual Combinational Devices with Calculation Control Based on Hamming Codes, *Informatics and Automation*, 2023, vol. 22, no. 2, pp. 349–392. DOI: 10.15622/ia.22.2.5. (In Russian.)
24.Efanov, D., Sapozhnikov, V., Sapozhnikov, Vl., et al., Self-Dual Complement Method up to Constant-Weight Codes for Arrangement of Combinational Logical Circuits Concurrent Error-Detection Systems, *Proceedings of 17th IEEE East-West Design & Test Symposium (EWDTS'2019)*, Batumi, Georgia, 2019, pp. 136–143. DOI: 10.1109/EWDTS.2019.8884398.
25.Lala, P.K., *Self-Checking and Fault-Tolerant Digital Design*, San Francisco: Morgan Kaufmann Publishers, 2001.
26.Sapozhnikov, V.V., Sapozhnikov, Vl.V., and Efanov, D.V., Design of Self-Checking Concurrent Error Detection Systems Based on "2-out-of-4" Constant-Weight Code, *Control Sciences*, 2017, no. 1, pp. 57–64. (In Russian.)
27.Efanov, D.V. and Pivovarov, D.V., Checkers of Self-Dual and "`Close in Meaning'" Signals, *Journal of Instrument Engineering*, 2024, vol. 67, no. 1, pp. 5–19. https://doi.org/10.17586/0021-3454-2024-67-1-5-19 (In Russian.)
28.Nikolos, D., Self-Testing Embedded Two-Rail Checkers, *Journal of Electronic Testing: Theory and Applications*, 1998, vol. 12, no. 1-2, pp. 69–79. DOI: 10.1023/A:1008281822966
29.Dmitriev, A., Saposhnikov, V., Saposhnikov, V., and Goessel, M., New Self-Dual Circuits for Error Detection and Testing, *VLSI Design*, 2000, vol. 11, no. 1, pp. 1–21. DOI: 10.1155/2000/84720
30.Gessel', M., Moshanin, V.I., Sapozhnikov, V.V., and Sapozhnikov, Vl.V., Fault Detection in Self-Test Combination Circuits Using the Properties of Self-Dual Functions, *Avtomat. Telemekh.*, 1997, no. 12, pp. 193–200. (In Russian.)
31.Korshunov, A.D., Computational Complexity of Boolean Functions, *Russian Math. Surveys*, 2012, vol. 67, no. 1, pp. 93–165. https://doi.org/10.1070/RM2012v067n01ABEH004777.
32.Efanov, D.V., Self-Checking Combinational Devices Synthesis Based on the Boolean Signal Correction Method Using Bose–Lin Codes, *Information Technologies*, 2023, vol. 29, no. 10, pp. 503–511. DOI: 10.17587/it.29.503-511.
33.Drozd, A., Kharchenko, V., Antoshchuk, S., et al. Checkability of the Digital Components in Safety-Critical Systems: Problems and Solutions, *Proceedings of 9th IEEE East-West Design & Test Symposium (EWDTS'2011)*, Sevastopol, Ukraine, 2011, pp. 411–416. DOI: 10.1109/EWDTS.2011.6116606.
34.Sapozhnikov, Vl.V., *Sintez sistem upravleniya dvizheniem poezdov na zheleznodorozhnykh stantsiyakh s isklyucheniem opasnykh otkazov* (Design of Train Traffic Control Systems at Railway Stations with the Elimination of Dangerous Failures), Moscow: Nauka, 2021. (In Russian.)
35.Hahanov, V., Litvinova, E., Davitadze, Z., et al., Truth Table Based Intelligent Computing, *Proceedings of 31st International Conference "Mixed Design of Integrated Circuits and Systems,"* Gdansk, Poland, 2024. DOI: 10.23919/MIXDES62605.2024.10614035
36.Meliqyan, V., Davtyan, V., Harutyuyan, A., et al., Novel Adaptive Crossover Mechanism for Genetic Algorithm on Integrated Circuit Cell Placement, *Proceedings of IEEE East-West Design & Test Symposium (EWDTS)*, Yerevan, Armenia, 2024. DOI: 10.1109/EWDTS63723.2024.10873659

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

Received March 10, 2025, and revised May 12, 2025. Accepted June 18, 2025.

Author information

Efanov, Dmitry Viktorovich. Dr. Sci. (Eng.), Peter the Great Saint Petersburg Polytechnic University, St. Petersburg, Russia; Russian University of Transport, Moscow, Russia
✉ TrES-4b@yandex.ru
ORCID iD: https://orcid.org/0000-0002-4563-6411

Cite this paper

Efanov, D.V., Compositions of Two Constant-Weight Codes with Orthogonal Combinations over All Bits for Self-Checking Discrete Device Design. *Control Sciences* 3, 41–51 (2025).

Original Russian Text © Efanov, D.V., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 49–62.

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

A MATHEMATICAL MODEL OF ADAPTIVE TRAFFIC CONTROL IN MOBILE NETWORKS WITH VARIABLE SIGNAL QUALITY

A. A. Esin

AO MirWiFi, Moscow, Russia

✉ anton.esin@imm.am

Abstract. This paper considers the mathematical modeling problem of traffic transmission in mobile networks under high user mobility and spatially heterogeneous coverage, including signal degradation ("dead") zones. Traffic aggregation at the channel level is applied to increase the reliability and stability of data transmission. A Markov model of a communication channel is proposed to study the effectiveness of aggregation algorithms and adapt them to network operating parameters and user velocity. The model is based on a periodic affine motion of a mobile device between base stations (BSs) uniformly distributed along a straight line. Within this model, the concepts of stable coverage zones and transition zones are introduced in terms of the distances to the nearest and next BSs. The channel state is described by a Markov chain with states corresponding to signal quality sampling: stable connection, degraded connection, and disconnection. Transitions between states are governed by a continuous-time Markov process with constant rates, and the parameters of this process are determined from empirical network data. An extension of the model to incorporate the time dependence of channel states is also considered, leading to a semi-Markov framework. For both cases (Markov and semi-Markov chains), explicit expressions are derived for the stationary probabilities of states, and system stability conditions are formulated to ensure bounded traffic queues. In addition, an adaptive control model of the channel throughput is proposed; this model optimizes transmission parameters depending on the current channel state, request queue length, and user velocity. The effectiveness of the approach is demonstrated by numerical simulations: the network has stable performance across a wide range of mobility levels and coverage parameters. The model can be applied to the reliability analysis and optimization of network protocols in highly mobile environments, including high-speed railway transport, vehicle networks, and mobile platforms.

Keywords: mobile networks, Markov model, semi-Markov process, system stability, Laplace transform.

INTRODUCTION

Modern communication systems providing stable data transmission from mobile objects (high-speed trains, unmanned vehicles, drones, and cars) actively use the infrastructure of public cellular networks. However, the quality of communication is significantly reduced outside of densely built-up areas due to the sparsity of base stations (BSs), incomplete coverage, and radio channel instability. This effect is manifested by signal attenuation, disconnections, and significant delays in data transmission.

At the same time, *the quality of service* (QoS) requirements of users in a highly mobile environment are not lower than the demands of users in stable coverage conditions, namely, the stable transmission of heavy multimedia traffic, video streaming support, and real-time data transmission from mobile platforms. Such scenarios are particularly relevant, e.g., in video surveillance, telemetry, or automated control of mobile objects.

Traffic aggregation is an effective way to solve these problems [1, 2]. The existing traffic aggregation techniques, e.g., Link Aggregation Control Protocol (LACP) and Multilink Point-to-Point Protocol (Multilink PPP), are used in LTE and LTE Advanced (LTE-A) architectures. This paper is oriented toward common scenarios in LTE and LTE-A, where aggregation is implemented at the Packet Data Convergence Protocol (PDCP) level via Carrier Aggregation (CA) mechanisms. The model can also be adapted for 5G scenarios, but the features of the New Radio (NR) ar-

chitecture (New Radio) are not considered in detail. Note that these methods demonstrate high efficiency in networks with stable or moderate coverage but become significantly worse under abrupt changes in signal quality and high user velocity, which requires frequent switching between BSs [2].

The analysis of typical network operation scenarios under high mobility of users allows identifying several fundamental problems:

• Incomplete network coverage. There are significant dead zones outside urban areas, where the signal level is insufficient for a stable connection; as a result, the overall network throughput is reduced, and packet losses are increased.

• Signal dynamics. The quality of the communication channel essentially depends on the path and velocity of the user. The Doppler shift, multipath propagation, and heterogeneous coverage lead to channel instability and frequent disconnections.

• Nonoptimal behavior when losing communication. In case of a disconnection or signal attenuation, multiple packet retransmission attempts are initiated, making both the channel and network infrastructure overloaded, reducing overall transmission efficiency, and increasing power consumption.

The existing methods for mobile traffic aggregation and control are not effective enough to cope with the above problems. They suffer from the following main drawbacks:

• No time correlation. Most models assume independent transitions between network states, thereby ignoring equipment inertia and the physical regularities of signal propagation.

• No adaptation to network topology and user velocity. Several approaches suppose a stationary or quasi-stationary environment, therefore being inapplicable in the case of high velocities and dynamically changing (variable) channel quality.

• High switching latency. Switching algorithms between BSs can lead to long pauses in traffic transmission, especially in unstable signal conditions.

An obvious solution is to improve the existing aggregation methods. However, appropriate mathematical models should be developed to optimize network performance and find the best solution. As a rule, the existing models are either simplified or neglect key parameters of real systems, such as the changing spatial position of servers and users [3, 4], time correlations in the change of network state, dynamic load redistribution, and adaptive data buffering; or they consider the spatial geometry of network elements but not the complex dynamics of queues [5–7]; or they analyze only the connectivity problems of network components using graph theory [8] or percolation theory [9–12], adding time-varying node states [13] and adapting classical methods of statistical physics to wireless network analysis problems [14, 15].

The objective of this paper is to develop a new traffic control model for mobile networks with stochastic variability in communication channel throughputs, the time correlation of network states, and adaptive load balancing to improve connection stability and reduce delays. We present a stochastic aggregate traffic transmission model in which the channel characteristics are described by a finite Markov chain obtained via signal level sampling [16]. The Markov model is justified due to the exponential nature of the statistics of transitions between states in real measurements (e.g., in the analysis of switching scenarios between BSs in LTE) [17].

In contrast to the existing models assuming either a stationary channel state or the independent transitions between states, we propose a stochastic model incorporating time correlation and spatial coverage structure with the user's motion between BSs along an affine path. The main contribution of this paper consists in:

– the construction of Markov and semi-Markov models of a communication channel based on signal level sampling,

– the analytical derivation of stationary state distributions and queue stability conditions,

– the development of an adaptive traffic control model that optimizes channel throughput considering the current channel state, user velocity, and queue length.

This paper is organized as follows. In Section 1, we build a channel model, present a mathematical model of adaptive traffic control with the main equations, describe a Markov process of transitions between states, and analyze stability conditions of the system. The formulations and proofs of the main results are given in Section 2, including explicit analytical formulas for the stationary distribution and a stability condition of the model. Section 3 provides numerical simulation results for the model in different network operation scenarios; also, the model is compared with the existing traffic aggregation methods. In the Conclusions, we summarize the key findings of the study, assess the effectiveness of the model, and outline directions for further research.

This study is focused on the LTE and LTE-A architectures, where traffic aggregation is performed at the PDCP level via Carrier Aggregation (CA) mechanisms. The model considered is applicable to high-speed travel scenarios (e.g., railways) that are typical of these technologies. Although more sophisticated aggregation schemes (including Dual Connectivity and Service Data Adaptation Protocol (SDAP)) are used in 5G NR, they go beyond the scope of this paper and

may be the subject of a later generalization. The main attention is paid to the formalized description of the stochastic structure of a radio channel and its adaptive throughput control in the LTE/LTE-A framework.

1. BUILDING THE MATHEMATICAL MODEL

We will consider a mobile network as a set of BSs and mobile users traveling along routes with different signal coverage levels. In this case, a mobile user will be understood as a single aggregator or server that collects and forwards traffic from multiple devices aboard a high-speed train or car. In practice, such a scheme is widespread in modern 4G/5G networks (an example is multi-relay systems, where a train or bus has a single access point to a cellular network) [1]. This allows aggregating the traffic of multiple devices into a single channel and simplifying the process of switching between BSs while individual devices within the coach connect to this "server" [1, 18, 19]. In the context of LTE- and LTE-A architectures, the aggregator is realized as a user terminal or a multi-relay platform that unites onboard traffic of a vehicle and interacts with multiple component carriers (CA).

Assume that the network coverage is not dense (the distances between BSs are considerable enough for coverage breaks) and user velocity is sufficiently high (several BSs should be involved sequentially to transmit the required traffic).

1.1. Model Assumptions

The network topology is naturally modeled as a directed graph $G=(V, E)$ [3, 5, 8], where V stands for a set of network nodes (e.g., BSs) and $E \subseteq V \times V$ is a set of its oriented edges (possible data transmission routes between nodes).

We make several assumptions regarding the model, which simplify the analysis significantly without limiting the generality of the approach and the main qualitative conclusions.

• **The graph of BSs is linear:** they are located along a straight line at points

$$x_j = j\,\Delta x,\ \ j \in \mathbb{Z}_+,$$

where Δx is a fixed distance between neighbor BSs. In other words, the BSs are uniformly distributed at the nodes of the one-dimensional grid $\mathbb{Z}_+$. This assumption reflects the typical coverage on main lines (highways or railways).

Remark. Note that the linear uniform arrangement of BSs along a straight line is an idealized assumption, characteristic primarily of main lines (highways, railways, etc.). In real systems, there may be variations in the mutual location of BSs, curved route segments, inhomogeneous relief zones (tunnels, bridges, etc.). Nevertheless, the linear model provides a convenient basic framework for analyzing the key regularities, especially when moving along relatively straight route segments. This simplification does not affect the overall logic of the methods and conclusions but can be refined if necessary.

• **The aggregator's motion is uniform:** the aggregator moves along the above straight line with a constant velocity:

$$v(t) = v_0,\ v_0 > 0.$$

This assumption is justified to model standard scenarios (e.g., high-speed trains or cars on highways) where velocity changes are relatively small and the aggregator is moving steadily.

Remark. We suppose a constant aggregator velocity $v(t) = v_0$ (or the normalized unit velocity) due to stationary high-speed transportation conditions on long main lines. At the same time, real scenarios may include segments of acceleration (braking) as well as velocity fluctuations. If such effects are significant, the model can be extended. This study is focused on the basic scenario with a constant (or nearly constant) velocity, which ensures periodic traffic and simplifies further analytical considerations without loss of generality.

• **The impact of BSs is local:** the quality of communication at each point of the aggregator's path is determined only by the two nearest BSs, namely, the current one (behind the aggregator) and the next one (in front of the aggregator). This simplification is justified due to the rate of signal attenuation with increasing distance and allows neglecting the impact of far BSs without significant loss of accuracy.

Thus, under the uniform arrangement of BSs and uniform motion of the aggregator, the latter's movements can be modeled as a periodic motion of a point along the half-interval [0, 1); when the aggregator reaches the position 1, the current and next BSs change, and the aggregator starts moving again from the point 0. This interpretation significantly simplifies the mathematical analysis and numerical simulation of network dynamics.

Under these assumptions, the aggregator's state at any time instant can be fully parameterized by its position on the affine half-interval [0, 1); the model can be investigated only on the period of traversing this distance (further called the transit time).

Thus, given the uniform arrangement of BSs and uniform motion of the aggregator, the latter's motion can be formally described as follows.

Let $T = \dfrac{\Delta x}{v_0}$ denote the aggregator's transit time between two neighbor BSs.

Then the aggregator's position at any time instant t is modeled by a periodic affine parameter $s(t) \in [0,1)$ given by the formula

$$s(t) = \frac{t \bmod T}{T},$$

where $s(t) = 0$ corresponds to the aggregator's position exactly at the current BS point and $s(t) = 1$ to reaching the next BS.

When the aggregator reaches the position $s(t) = 1$, it instantly starts moving on a new segment where the next BS becomes the current one, and the new neighbor BS becomes the next one.

Thus, the aggregator's motion through the network is a periodic process represented by this model as a repetitive motion of a point along the half-interval [0, 1):

$$x(t) = s(t).$$

An important special case is the unit velocity $v_0 = 1$ of the aggregator and the unit distance $\Delta x = 1$ between BSs, which simplifies the model and subsequent analysis as much as possible.

In this case, the transit time between neighbor BSs becomes $T = 1$, and the aggregator's position $x(t)$ on the half-interval [0, 1) is given by the simple periodic dependence

$$x(t) = t \bmod 1.$$

Thus, the aggregator moves along the half-interval [0, 1) with constant unit velocity and, having reached the point 1, instantly moves to the next half-interval, starting again from the position 0. Graphically, this function represents a periodic sawtooth dependence with unit period.

Thus, the aggregator's position on the half-interval [0, 1] completely determines its distance to two neighbor BSs.

In view of the model periodicity, the properties of this system can be analyzed by considering its behavior only on one fixed period $t \in [0, 1]$. The results for the other time intervals will be similar due to periodicity.

1.2. Dynamics of Distances to BSs and Signal Quality Given the Network Topology

At a position $x(t) \in [0, 1]$, the distances between the aggregator and the current (behind it) and next (in front of it) BSs are given by

$$d_{\text{current}}(t) = x(t),\ d_{\text{next}}(t) = 1 - x(t),\ \ 0 \le x(t) \le 0.5;$$

$$d_{\text{current}}(t) = 1 - x(t),\ d_{\text{next}}(t) = x(t),\ 0.5 < x(t) \le 1.$$

These formulas imply that:

- At the beginning of the period ($x(t) = 0$), the aggregator is exactly at the current BS (the distance to it equals 0), and the distance to the next station is maximal (equals 1).
- In the middle of the period ($x(t) = 0.5$), the aggregator is equidistant (0.5) to the two neighbor BSs.
- At the end of the period ($x(t) = 1$), the aggregator reaches the next BS and performs an instantaneous transition to a new segment, where the next BS becomes the current one and the aggregator's position again becomes 0.

Thus, in one period, the aggregator first moves away from the current BS (when passing from 0 to 0.5) and then approaches the next station (when passing from 0.5 to 1). As a result, the distance to the nearest BS in one period first monotonically increases from 0 to 0.5 and then monotonically decreases back to 0.

In addition, assume that each BS has a finite range R of a stable signal received by the aggregator (the state S_1). Beyond the range R, there exists a small transition zone of a width δ, where the signal is degraded (the state S_2).

Thus, there are three coverage zones for each BS:

– a stable signal zone (the state S_1), described by $0 \le d \le R$,

– a degraded signal (transition) zone (the state S_2), described by $R < d \le R + \delta$, and

– a dead zone (the state S_3), described by $d > R + \delta$, where d is the current distance to the BS:

$$d = d(t) = \min\{d_{\text{current}}(t), d_{\text{next}}(t)\}.$$

By assumption, the following natural condition holds:

$$2(R + \delta) < 1.$$

In other words, the total diameter of the coverage zones of one BS (the stable signal zone plus the transition zone on both sides) is significantly smaller than the distance between neighbor BSs. This ensures the presence of dead zones between the coverage zones of neighbor BSs, where the signal completely vanishes.

When the aggregator moves along the half-interval [0, 1), the aggregator sequentially passes:

– the stable signal zone of the current station,

– the transition zone of the current station,

– the dead zone,

– the transition zone of the next station, and
– the stable signal zone of the next station.

In general, the impact of the aggregator's velocity and distance to the nearest BS on the quality of communication is modeled by a special loss function combining linear and nonlinear effects [20]:

$$L\big(d(t),\ v(t)\big) = L_0 + a\,d(t)^{\gamma} + k_1 v(t) + k_2 \log\big(1+v(t)\big) + k_3 v(t)^{\alpha},$$

where L_0 is the basic loss in the ideal case (zero distance and zero velocity); $a\,d(t)^{\gamma}$ is the signal attenuation component depending on the distance $d(t)$ to the BS (the power signal propagation model, usually with $\gamma \in [2,4]$; $k_1 v(t)$ is the linear component due to the Doppler frequency shift; $k_2 \log\big(1+v(t)\big)$ is the logarithmic attenuation component caused by multipath fading; $k_3 v(t)^{\alpha}$ is the nonlinear effects arising due to switching between BSs and high velocities ($\alpha > 1$).

Thus, the above model simultaneously considers both key factors, i.e., the aggregator's distance to the BS and velocity, therefore being physically adequate and practically useful for further analysis.

The linear model $L(v) = L_0 + ad(t)^{\gamma} + k_1 v(t)$ is a special case ($k_2 = k_3 = 0$) to simplify the basic analysis [20]; this model will be investigated below.

1.3. The Process of Transitions between Discrete Signal Levels in the General Case

Within the model proposed, the aggregator sequentially passes three zones with different signal quality (stable signal, weak signal, and dead zone). In real communication networks, transitions between channel states $(S_1,\ S_2,\ S_3)$ occur not instantaneously but with some delay. The main causes of such effects are [20]:

- constant external conditions: the parameters of the communication channel remain relatively stable for some time, so transitions between states occur gradually rather than abruptly;
- communication equipment inertia: switching between BSs requires additional time for coordination and signal processing;
- physical path: along a certain path, transitions between states are related to the network topology and the regularity of BS locations.

These effects create a time correlation between channel states: the probabilities of transitions depend not only on the current state but also on its dwell time. A strict mathematical description of such situations leads to a more general class of processes called *semi-Markov processes* (SMPs) [21, 22].

Consider a random process $X(t)$ modeling the change of signal level in a mobile network. This process is a semi-Markov model with three states: S_1 (stable connection), S_2 (degraded connection), and S_3 (disconnection). The instants of transitions form an increasing sequence $\{\tau_n\}$ of random variables, with $X(t)$ keeping the current state between transitions [23].

The semi-Markov model of signal levels is described by the following system of integral and differential equations for the probabilities of states $p_i(t) = P\{X(t) = S_i\},\ i = 1,\ 2,\ 3$:

$$\frac{dp_i(t)}{dt} = \sum_{j\neq i} \int_0^t p_j(\tau)\,\alpha_{ji}(t-\tau)\,d\tau - \sum_{k\neq i} \int_0^t p_i(\tau)\,\alpha_{ik}(t-\tau)\,d\tau, \tag{1}$$

with the transition rate

$$\alpha_{ij}(t) = \lambda_{ij}(t) \exp\left(-\int_0^t \lambda_i(u)\,du \right),$$

$$\lambda_i(t) = \sum_{k\neq i} \lambda_{ik}(t),\ i = 1,\ 2,\ 3,$$

Let the initial conditions be fixed: $p_1(0) = 1, p_2(0) = p_3(0) = 0$.

The Laplace transform [24] can be conveniently applied to solve system (1). In this case, we obtain the following system of algebraic equations in the Laplace space:

$$s\tilde{p}_i(s) - p_i(0) = \sum_{j\neq i} \tilde{p}_j(s)\tilde{\alpha}_{ji}(s) - \sum_{k\neq i} \tilde{p}_i(s)\tilde{\alpha}_{ik}(s).$$

Generally, the dwell times can have arbitrary distributions $F_i(t)$: $T_i \sim F_i(t)$. Then the system of equations (1) becomes rather complicated for analytical treatment. In such a situation, one can apply numerical schemes on a time grid $\{t_n\}$, e.g.,

$$p_i(t_{n+1}) = p_i(t_n) + \Delta t \left[\sum_{j\neq i} \sum_{m=0}^{n} p_j(t_m)\,\alpha_{ji}(t_{n+1} - t_m) - \sum_{k\neq i} \sum_{m=0}^{n} p_i(t_m)\,\alpha_{ik}(t_{n+1} - t_m) \right].$$

This direct numerical approach is effective for more complex distributions (gamma, Weibull, etc.).

In practice, the gamma (Erlang), Weibull, and uniform distributions of dwell times are often used. Below, we present the expressions for the transition rates $\alpha_{ij}(t)=\lambda_{ij}(t)\exp\left(-\int_0^t\lambda_i(u)du\right)$ and their Laplace transforms $\tilde{\alpha}_{ij}(s)=\mathcal{L}\{\alpha_{ij}(t)\}$ under these distributions of dwell times:

- the gamma distribution (the Erlang distribution):

$$\lambda_i(t)=\frac{\lambda_i^k t^{k-1}e^{-\lambda_i t}}{(k-1)!-\sum_{m=0}^{k-1}\frac{(\lambda_i t)^m}{m!}},$$

$$\alpha_{ij}(t)=p_{ij}\lambda_i(t),\ \tilde{\alpha}_{ij}(s)=p_{ij}\frac{\lambda_i^k}{(s+\lambda_i)^k};$$

- the Weibull distribution:

$$\lambda_i(t)=\frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1},\ \alpha_{ij}(t)=p_{ij}\frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1}\exp\left[-\left(\frac{t}{\alpha}\right)^{\beta}\right],$$

$$\tilde{\alpha}_{ij}(s)=p_{ij}\frac{\beta}{\alpha^{\beta}}\int_0^{\infty}t^{\beta-1}\exp\left[-\left(\frac{t}{\alpha}\right)^{\beta}-st\right]dt;$$

- the uniform distribution on the segment $[a,b]$:

$$\lambda_i(t)=\begin{cases}0, & t<a,\\ \dfrac{1}{b-t}, & a\le t<b,\\ \infty, & t=b,\end{cases}$$

$$\alpha_{ij}(t)=\begin{cases}0, & t<a,\\ \dfrac{p_{ij}}{b-a}, & a\le t\le b,\\ 0, & t>b,\end{cases}$$

$$\tilde{\alpha}_{ij}(s)=p_{ij}\frac{e^{-as}-e^{-bs}}{(b-a)s}.$$

By substituting the above parameters into the system of equations (1), we can derive explicit expressions for the probabilistic characteristics of the model with different distributions of dwell times and the most important quality metrics of the system.

However, the analytical study of such models generally leads to cumbersome expressions complicating their interpretation and application. Therefore, we will consider the Markov case with the exponential distribution of dwell times:

$$\mathbb{P}(T_i>t)=e^{-\lambda_i t}.$$

This case corresponds to the absence of memory and considerably simplifies the analysis. Here, time correlations and channel inertia are assumed to be averaged and approximated by constant transition rates of states.

This simplification allows obtaining strictly analytical and easily interpretable results (see Section 2) as the main analytical implications of the model.

2. THE MAIN RESULTS

2.1. An Optimal Control Mechanism

To adaptively adjust the traffic transmission parameters, we formally introduce a function $u(t)$ depending on the current state of the network:

$$u(t)=\begin{cases}\mu_{\text{norm}} & \text{if } X(t)=S_1\\ \mu_{\text{slow}}\,g(\tau_2,\theta) & \text{if } X(t)=S_2\\ 0 & \text{if } X(t)=S_3,\end{cases}$$

where $g(\tau_2,\theta)$ is an adaptive function depending on the dwell time in the state S_2 and the parameters θ to be optimized.

To optimize the system's operation, we define an objective function of the form

$$J(u)=\int_0^T\left[\omega_1 L_{\text{pack}}(u(t))+\omega_2 D(u(t))\right]dt, \qquad (2)$$

where $L_{\text{pack}}(u(t))$ is a function reflecting packet losses depending on the chosen control strategy $u(t)$; $D(u(t))$ is a function reflecting data transmission delays; ω_1 and ω_2 are significance coefficients of the corresponding criteria; T is a time interval under consideration (the control horizon).

An optimal strategy $u^*(t)$ is achieved by minimizing the objective function (2). For optimization, one can employ, e.g., gradient descent or dynamic programming methods [25, 26]. With gradient descent applied to the parameters θ, the correction is performed using the well-known algorithm [25, 27]

$$\theta_{n+1}=\theta_n-\eta\nabla_\theta J(u),$$

where η is the algorithm parameter and $\nabla_\theta J(u)$ denotes the gradient of the objective function with respect to the parameters θ. Such approaches are widely

used in the calibration of communication systems and flow control in networks.

In addition to controlling the transmission rate, the function $u(t)$ can also affect the transition rates between network states. For example, one can modify these rates to reduce the probability of disconnection or accelerate recovery:

$$\lambda_{ij}\left(v,\, u(t)\right) = \lambda_{ij}^{0}\, h\left(u(t)\right) e^{-k_v v(t)}, \qquad (3)$$

where λ_{ij}^{0} are the baseline transition rates; the function $h\left(u(t)\right)$ describes the corrective control impact, and $e^{-k_v v(t)}$ reflects the impact of the aggregator's velocity $v(t)$ (if necessary). Such corrections model physical and protocol mechanisms where, e.g., more aggressive resource utilization (higher transmission power) increases the value of $h\left(u(t)\right)$, thereby reducing the probability of passing to a dead zone.

In this study, we use the exponential control function (3). It reflects well the intuitively implied exponential deterioration of the channel parameters when staying long in the degraded connection state [28]. The choice of an exponential function is justified by its analytical simplicity and application in known models of channels with fading, as well as by the possibility of calibrating the parameter α within an optimization procedure [29].

However, other types of functions (logistic, linear, etc.) are possible, depending on physical or protocol constraints. Such a choice of the function $g(\tau_2, \theta)$ can be justified by available data or when approximating mechanisms implemented, e.g., in the adaptive modulation of modern communication protocols (LTE and 5G [30]).

In this paper, the objective function (2) is considered in a general form. For numerical simulation, it can be specified as follows:

$$L_{\text{pack}} = \frac{1}{T}\int_{0}^{T} \mathbb{I}_{\text{loss}}(t)\, dt,$$

where $L_{\text{pack}}\left(u(t)\right)$ is the share of lost packets; $\mathbb{I}_{\text{loss}}(t) = 1$ if a packet at a time instant t was discarded and $\mathbb{I}_{\text{loss}}(t) = 0$ otherwise.

The average packet transmission delay $D\left(u(t)\right)$ can be estimated, e.g., as

$$D\left(u(t)\right) = \frac{1}{N}\sum_{i=1}^{N} W_i,$$

where W_i is the total dwell time of the ith packet in the system.

Thus, introducing the control $u(t)$ and objective functions yields a comprehensive approach to optimization in mobile networks [31]: one can adapt the transmission rate (or other channel parameters) and control the probability of disconnections and recoveries by changing the transition rates λ_{ij}. In practice, a suitable control function $u(t)$ and an appropriate optimization method are chosen depending on quality requirements (reduction of delays and losses) and available hardware (protocol) solutions [32].

Such control can be realized via an additional module (board) with program control, e.g., using neural networks to predict the channel state and adapt the parameters dynamically. In this way, it is possible to respond, in due time, to connection degradations and adapt, in real time, the transmission rate and the probability of disconnections in advance.

Depending on particular service requirements (e.g., delay constraints, loss probability thresholds, or transmission power limits), the objective function (2) can be refined by adding other criteria or penalties for exceeding quality standards. Thus, in particular engineering problems, the form of $L_{\text{pack}}(\cdot)$ and $D(\cdot)$ can be chosen by considering the practical significance of partial network quality metrics (packet losses, downtime, etc.). In this paper, the function $J(u)$ remains in a general form; in the section devoted to partial assumptions and simplifications, we will demonstrate the choice of the objective function in special cases.

2.2. Basic Properties of the System

Consider now the basic properties of the system. For the simplicity and transparency of further analysis, let the transition rates between channel states be constant within each zone and depend only on the current state of the aggregator. This means that the transitions between signal quality levels can be described using fixed parameters reflecting the physical characteristics of the medium and the motion dynamics.

Thus, the model has a finite number of constant transition rates, each characterizing a typical transition regime between the coverage zones of a BS. Under the assumed exponential distribution of dwell times in each state, the model turns into a continuous Markov process with a finite number of states given by the system of Kolmogorov equations. Therefore, from system

(1) we proceed to the consideration of the infinitesimal transition matrix

$$Q=\begin{pmatrix} -\lambda_{12} & \lambda_{12} & 0 \\ \lambda_{21} & -(\lambda_{21}+\lambda_{23}) & \lambda_{23} \\ 0 & \lambda_{32} & -\lambda_{32} \end{pmatrix},$$

and the vector of state probabilities $p(t)=(p_1(t), p_2(t), p_3(t))$ satisfies the system of Kolmogorov equations

$$\frac{dp(t)}{dt}=p(t)Q.$$

Here, the coefficients λ_{ij} have the following physical interpretation:

- λ_{12} is the transition rate from the state S_1 (stable connection) to the state S_2 (degraded connection), which corresponds to passing from the stable signal zone $d \le R$ to the transition zone $R < d \le R+\delta$.
- λ_{21} is the signal recovery rate from the state S_2 to S_1 during the reverse transition to the stable signal zone.
- λ_{23} is the signal loss rate (transition from the state S_2 to the state S_3 (disconnection)), which corresponds to reaching the dead zone.
- λ_{32} is the signal recovery rate when returning from the state S_3 to S_2 (i.e., when passing from the dead zone to the transition zone).

Remark. The network structure is not explicitly included in this model. However, it can be indirectly considered by parameterizing the matrix Q based on the adjacency matrix of the coverage graph or other structural characteristics of the network. Such an approach allows generalizing the model to the case of spatially heterogeneous networks or real telecommunication infrastructures.

Theorem 1 (on the existence of a piecewise smooth control function). *For the communication channel in one of the three states: S_1 (stable connection), S_2 (degraded connection), and S_3 (disconnection), there exists a piecewise smooth control function of the general form $u(t)$ depending only on the current channel state $X(t)$ such that*

$$u(t)=\begin{cases} u_1, & X(t)=S_1, \\ u_2, & X(t)=S_2, \\ u_3, & X(t)=S_3, \end{cases}$$

where u_i is the control parameters in the state S_i $(i=1, 2, 3)$. In addition, the service rates $\mu_i(u(t))$ in each channel state and the transition rates $\lambda_{ij}(u(t))$ between channel states are well-defined and controllable on each state constancy interval.

This control function determines the service rate $\mu_i(u(t))$ in each channel state and can also impact the transition rates $\lambda_{ij}(u(t))$ between the states (e.g., by a controllable acceleration/deceleration of channel recovery/degradation). In other words, when the channel is in the state S_i, the system applies the control action u_i, which sets the current service rate $\mu_i(u_i)$ and can change the transition rates $\lambda_{ij}(u_i)$ to other states.

P r o o f. Rationale behind choosing this form of $u(t)$. Control depending only on the current state $X(t)$ is natural in the context of adaptive communication systems. Such a piecewise function $u(t)$ allows instantaneously responding to changes in channel quality: in the event of passing from stable to degraded connection or disconnection, the control jumps to a new value corresponding to the degraded communication conditions. This provides adaptive service control: e.g., in case of channel degradation (S_2), it is possible to reduce the service rate or activate a more reliable transmission regime; in case of disconnection (S_3), initiate communication restoration procedures (which is equivalent to increasing the transition rate back to the state S_1 or S_2). In the stable connection state S_1, control can return to the maximum throughput regime. With this control function $u(t)$, the system is dynamically adapted depending on the current channel state, improving service stability during channel degradation and reducing downtimes during disconnections. ♦

Theorem 2 (a stability criterion for the queueing system). *Consider a Markov process $X(t)$ with three states $\{S_1, S_2, S_3\}$, where S_1 and S_2 have service rates μ_1 and μ_2, respectively, and no transmission is possible in S_3 ($\mu_3=0$). Let λ_{in} be the rate of an incoming (Poisson) flow of requests. We denote by π_1, π_2, and π_3 the stationary probabilities of channel's dwelling in the states S_1, S_2, and S_3, respectively (if they exist).*

The queueing system with unlimited buffer and variable signal quality is stable if and only if

$$\lambda_{\text{in}} < \pi_1\mu_1 + \pi_2\mu_2. \qquad (4)$$

According to condition (4), the average input load must be less than the average channel throughput $\pi_1\mu_1 + \pi_2\mu_2$. If condition (4) is violated, the queue (with unlimited buffer) grows infinitely; see [3, 24].

P r o o f. In the *M/M/1* model, the stability condition $\lambda_{\text{in}} < \mu$ is necessary and sufficient [2]. For the case with the channel state-dependent variable μ, the average service rate is $\sum_{i=1}^{3}\pi_i\mu_i$. Since $\mu_3 = 0$, the state S_3 has zero contribution to the throughput. Hence, the criterion turns into inequality (4). If condition (4) holds, there is a unique stationary distribution of the queue length; otherwise, no stationary distribution exists.

The stationary probabilities π_1, π_2, and π_3 are determined from the Kolmogorov equations (1). The stationary vector $\pi = (\pi_1, \pi_2, \pi_3)$ satisfies the system

$$\begin{cases} -\lambda_{12}\pi_1 + \lambda_{21}\pi_2 = 0 \\ \lambda_{12}\pi_1 - (\lambda_{21} + \lambda_{23})\pi_2 + \lambda_{32}\pi_3 = 0 \\ \pi_1 + \pi_2 + \pi_3 = 1. \end{cases}$$

Resolving this system gives

$$\pi_2 = \frac{\lambda_{12}}{\lambda_{21}}\pi_1,\ \pi_3 = \frac{\lambda_{23}}{\lambda_{32}}\pi_2 = \frac{\lambda_{12}\lambda_{23}}{\lambda_{21}\lambda_{32}}\pi_1.$$

Substituting these expressions into the normalization condition, we obtain

$$\pi_1 + \pi_2 + \pi_3 = \pi_1\left(1 + \frac{\lambda_{12}}{\lambda_{21}} + \frac{\lambda_{12}\lambda_{23}}{\lambda_{21}\lambda_{32}}\right) = 1$$

$$\Rightarrow \pi_1 = \left(1 + \frac{\lambda_{12}}{\lambda_{21}} + \frac{\lambda_{12}\lambda_{23}}{\lambda_{21}\lambda_{32}}\right)^{-1}.$$

Finally, with π_1, π_2, π_3 substituted into inequality (4), the explicit numerical condition takes the form

$$\lambda_{\text{in}} \le \left(\frac{\lambda_{21}\lambda_{32}}{\Delta}\right)\mu_1 + \left(\frac{\lambda_{12}\lambda_{32}}{\Delta}\right)\mu_2,$$

where

$$\Delta = \lambda_{21}\lambda_{32} + \lambda_{12}\lambda_{32} + \lambda_{12}\lambda_{23}. \ \blacklozenge$$

Corollary (a stability condition for the system with control). *In the stationary regime, the stability condition for the system with optimal control is given by*

$$\lambda_{\text{in}} < \pi_1(u^*)\mu_{\text{norm}} + \pi_2(u^*)\mu_{\text{slow}},$$

where the stationary probabilities $\pi_i(u^*)$ *depend on the optimal strategy* $u^*(t)$.

2.3. Performance Characteristics of the System in the Stationary Mode

The previous sections have been focused on the description of the channel (the states S_1, S_2, S_3) and its transition rates. In a real queueing system, however, the full state at a time instant t is given by the "channel state–queue length" pair.

With the channel evolution estimated in terms of the states $\{S_1, S_2, S_3\}$ and the control function $u(t)$, we can assess the quality of the system considering traffic in real systems. For this purpose, let the Markov model of the channel be combined with the queueing model of the *M/M/1* type (or *M/M/1/N* with a finite buffer), where the service rate depends on the current channel state and, if necessary, on the control function $u(t)$.

This approach is based on the classical results of queueing theory [3, 4], applied to the situation where service parameters (and even channel transitions between states) may vary depending on the control strategy. As shown below, in the stationary regime, the key stochastic performance characteristics—the probability of packet loss, the average number of requests in the system, and throughput—are expressed through the standard Erlang and Little formulas considering the average (effective) speeds given by the channel states.

Consider a queueing system served by a *controllable* Markov channel with the three states $\{S_1, S_2, S_3\}$, where the control action $u(t)$ can affect both the service rate $\mu_i(u)$ in the state S_i and the transition rates $\lambda_{ij}(u)$. Assume that:

- The queue has a finite buffer: at most N packets waiting (hence, a total of $N+1$ packets along with the one being served).
- In the stationary regime, the channel spends a fraction of time $\pi_i(u^*)$ in the state S_i (under optimal or fixed control u^*).
- If $\mu_3(u) = 0$ (the dead zone), then $\bar{\mu} = \pi_1(u^*)\mu_1(u^*) + \pi_2(u^*)\mu_2(u^*)$ is the *effective* (average) service rate.
- The rate of the incoming flow of requests is λ_{in}.

Under the above assumptions, we can find the performance characteristics of the system.

• *The load factor.* The relative load is given by

$$\rho = \frac{\lambda_{\text{in}}}{\bar{\mu}} = \frac{\lambda_{\text{in}}}{\pi_1(u^*)\mu_1(u^*) + \pi_2(u^*)\mu_2(u^*)}.$$

If $\rho < 1$ for a given value u^*, the system can operate without collapse in an unlimited buffer; if $\rho \geq 1$, the queue grows or large losses occur.

• *The loss probability.* For a finite number N, similar to Erlang's formula, the probability of packet loss in an overloaded queue is given by

$$P_{\text{loss}} = P\{\text{queue} = N\} = \frac{\rho^{N+1}}{\sum_{k=0}^{N+1}\rho^k},$$

where ρ is defined through $\pi_i(u^*)$ and $\mu_i(u^*)$. As $N \to \infty$, we have $P_{\text{loss}} \to 0$ if $\rho < 1$ and $P_{\text{loss}} \to 1$ if $\rho > 1$.

• *The average number of packets and delay.* We denote by $L = E[n]$ the expectation of the number of packets in the queue and service. For $\rho \neq 1$, the following formula is valid:

$$L = \frac{\rho}{1-\rho} - \frac{(N+2)\rho^{N+2}(1-\rho)}{1-\rho^{N+2}}.$$

The average number of packets in the queue, L_q, is obtained by subtracting the average busy service position; similarly, for the average dwell time (W) and waiting time (W_q), we apply Little's law:

$$L = \lambda_{\text{eff}} W, \quad L_q = \lambda_{\text{eff}} W_q,$$

where $\lambda_{\text{eff}} = \lambda_{\text{in}}(1 - P_{\text{loss}})$ is the rate of the actual flow of incoming and served packets.

• *The average throughput.* In the stationary regime, the system serves the flow with the rate

$$\lambda_{\text{out}} = \lambda_{\text{eff}} = \lambda_{\text{in}}(1 - P_{\text{loss}}).$$

If $\rho < 1$, $\lambda_{\text{out}} = \lambda_{\text{in}}$ as $N \to \infty$ (no losses). If $\rho > 1$, the queue is full and $\lambda_{\text{out}} \approx \bar{\mu}$.

2.4. Implications and Remarks

• **Boundary cases (reducibility to classical models).** If the communication channel is almost always in the stable connection (good) state S_1 (i.e., $\pi_1 \approx 1$, $\pi_2 \approx \pi_3 \approx 0$, and $\mu_1 = \mu$), then the entire system reduces to the classical *M/M/1*. In this case, the stability criterion takes the form $\lambda_{\text{in}} < \mu$, and the above formulas for the probability of packet loss, the average number of packets, and the waiting time are simplified to well-known results (Erlang's formula, Little's formulas, etc.). In contrast, if the aggregator stays in the dead zone (the disconnection state S_3, in which π_3 is large and $\mu_3 \approx 0$) for a large share of time, the effective rate $\bar{\mu}$ will drop dramatically and the system will behave like a slow server with a high load ρ.

• **An example of the dead zone's impact on packet losses.** Let $\mu_1 = 1$, $\mu_2 = 0.5$, and $\mu_3 = 0$, and let the fractions of time in the corresponding states be $\pi_1 = 0.8$, $\pi_2 = 0.15$, and $\pi_3 = 0.05$. Then $\bar{\mu} = 0.8 \cdot 1 + 0.15 \cdot 0.5 = 0.875$. For an incoming flow with $\lambda_{\text{in}} = 0.7$, we have $\rho \approx 0.8 < 1$, and the system operates without significant packet losses. However, for $\lambda_{\text{in}} = 1.0$ ($\rho \approx 1.14 > 1$), the queue starts growing infinitely (an arbitrarily large buffer) or reaches the maximum N (a bounded buffer); an appreciable probability of packet loss arises ($P_{\text{loss}} \approx 0.2$), and the resulting throughput saturates at $\lambda_{\text{out}} \approx \bar{\mu} = 0.875$.

• **The physical meaning of controlling transitions and service rate.** If one can control the rates $\lambda_{ij}(u(t))$ (accelerating recovery from the state S_3 or decelerating withdrawal from the state S_1) and/or modify $\mu_i(u(t))$ (selecting more aggressive or safer regimes), then it is actually possible to control the share of time π_3 (staying in the disconnection state) and the effective rate $\bar{\mu}$. Thereby, the load ρ, the probability of packet loss, and the average delay can be significantly reduced. However, in practice, increasing μ_i or reducing the dwell time in the state S_3 may require additional resources (power, redundant channels, etc.), so a trade-off between cost and gain in the quality of service should be found.

• **The scope of application and prospects.** The above assumptions (exponential distributions, Markov or semi-Markov transitions) simplify the analysis and provide elegant formulas but can only approximate real channels with correlated traffic and complex signal propagation dynamics in a heterogeneous medium. Nevertheless, the main conclusion (the need to satisfy the stability condition $\lambda_{\text{in}} < \bar{\mu}$) and the optimization

principles of the control function $u(t)$ remain valid. The model allows estimating the marginal network performance and adapting the service and transition strategy in modern mobile networks, where important criteria include network throughput, power consumption, connection recovery time, etc..

• **Relation to *M/G/*1 systems with interruptions.** As a generalization of the system with $n>3$ states [33–35], the three-state channel model with variable service rate and periods of complete unavailability (disconnections) considered in this paper is conceptually close to *M/G/*1 systems with interruptions; for example, see [36, 37]. However, unlike typical M/G/1 models with interruptions, we use an additional control function $u(t)$ to impact the transition rates and the service rate. As a result, the parameters are adapted to the current conditions of the communication channel, which extends the classical approaches and opens up the possibility of adaptive optimization.

• Thus, there is a similarity with *M/G/*1 with interruptions, and the results of this paper (particularly the criterion $\lambda_{\text{in}} < \sum_i \pi_i \mu_i$) repeat the logic of classical models. However, owing to the dynamic control, the physical interpretation of the states (radio channel), and the objective function $J(u)$, the model presented has a more flexible structure, opening new possibilities of adaptation in engineering applications.

3. THE STABILITY AND PERFORMANCE OF THE ADAPTIVE MODEL: NUMERICAL ANALYSIS

In this section, we provide simulation results confirming the above theoretical conclusions. Consider a stochastic channel with the three states $\{S_1, S_2, S_3\}$ and an incoming flow of requests with a rate λ_{in}. The throughput of the system depends on the current channel state and the control function $u(t)$.

The simulation was carried out using the discrete event method on a horizon $T_{\max}$. The key characteristics investigated include system stability, queue length, the probability of packet loss, and the impact of the network parameters and control strategy on the network quality. The experiment parameters, metrics, and visualization of the system behavior in different regimes are given below.

3.1. Model Parameters

The flow of requests is modeled by a Poisson process with the rate λ_{in}. All requests are processed according to the FIFO (*first in, first out*) principle with a finite buffer.

The channel can be in three states:

– S_1 (stable connection), the corresponding throughput is μ_{norm},

– S_2 (degraded connection), the corresponding throughput is μ_{slow} (or $\mu_{\text{slow}} e^{-\alpha\tau_2}$), and

– S_3 (disconnection), the service is unavailable $(\mu = 0)$.

All transitions between the states are described either by a fixed matrix of transition rates (the Markov case) or by functions $\alpha_{ij}(t)$ in the semi-Markov model. A request is served if the communication channel has the state S_1 or S_2 and the queue is non-empty. In the state S_3, requests are still received, but their service is suspended.

The main quality metrics of the network are queue length dynamics and the impact of the rate λ_{in} and velocity on stability.

3.2. Simulation Results

The graphs below demonstrate the behavior of the model in two regimes:

– stable (subcritical, $\lambda_{\text{in}} < \pi_1\mu_{\text{norm}} + \pi_2\mu_{\text{slow}}$) and

– unstable (supercritical, $\lambda_{\text{in}} > \pi_1\mu_{\text{norm}} + \pi_2\mu_{\text{slow}}$).

In the subcritical regime (Fig. 1), the queue (the aggregator's buffer level) is stabilized and does not grow infinitely. However, there is a positive drift during overload (Fig. 2).

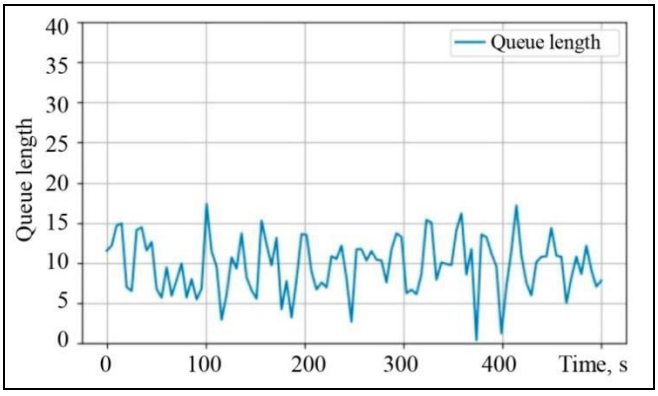

Fig. 1. Queue length dynamics in the stable regime.

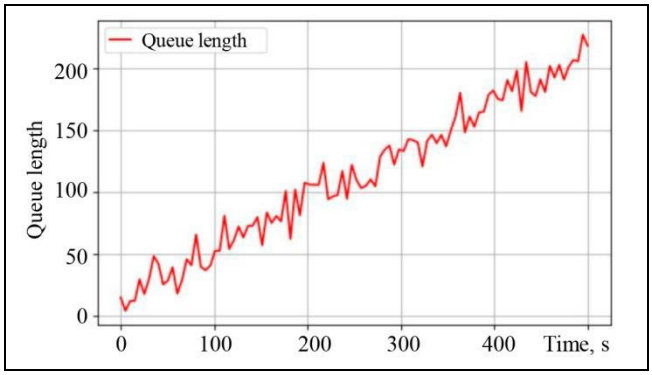

Fig. 2. Queue length dynamics in the unstable regime.

According to Fig. 3, in the subcritical regime, the communication is restored quite quickly even in case of disconnections. Under overload, even short service interruptions cause an avalanche-like growth of the aggregator's buffer.

Figure 4 shows the impact of exponential fading in the state S_2: under a high load, the dwell time in the state S_3 increases, which drastically deteriorates stability.

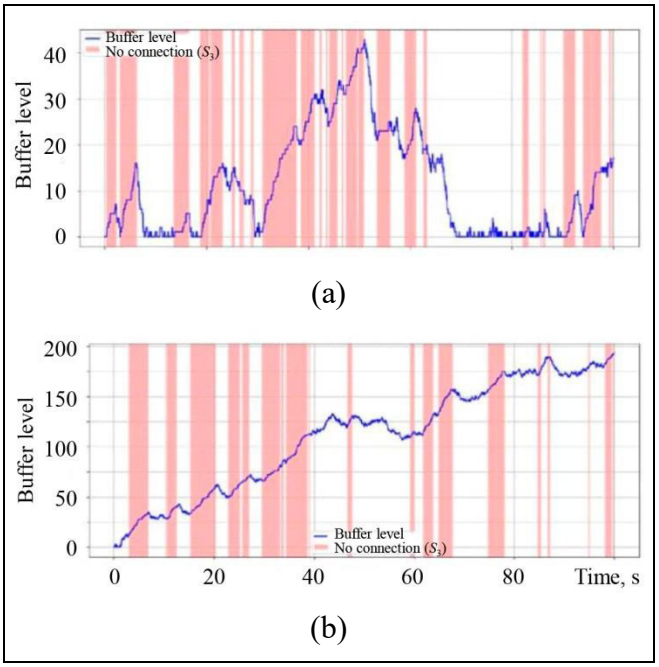

Fig. 3. Queue: (a) in the stable regime ($\lambda = 3.0$ reqs/s, $p = 30$ m/s) and (b) in the unstable regime ($\lambda = 7.0$ reqs/s, $p = 30$ m/s). The red zones indicate disconnection periods.

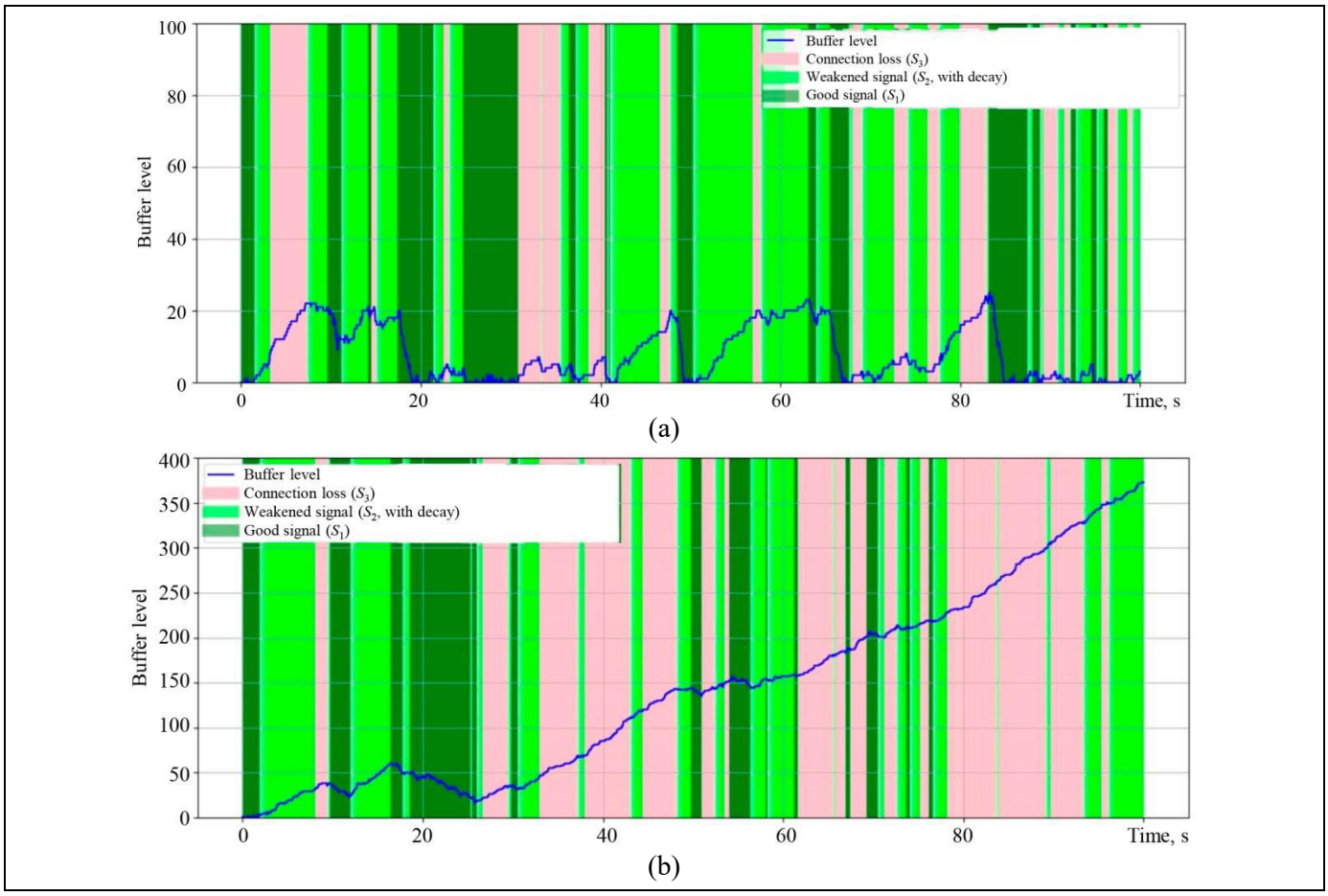

Fig. 4. Decaying throughput in the state S_2 : (a) stable regime ($\lambda = 3.0$) and (b) unstable regime ($\lambda = 7.0$) .

The advantage of adaptive control is demonstrated in Fig. 5: in the neighborhood of the critical load, this control effectively stabilizes the queue length.

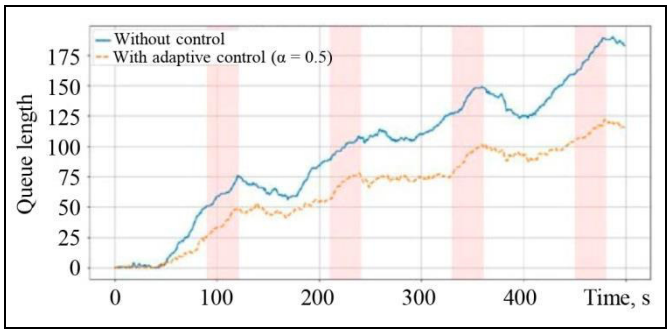

Fig. 5. Queue length dynamics: with adaptive control vs. without control.

Figure 6 shows the stability margin. The adaptive model demonstrates a shift in the overload threshold, allowing the system to operate closer to the critical values.

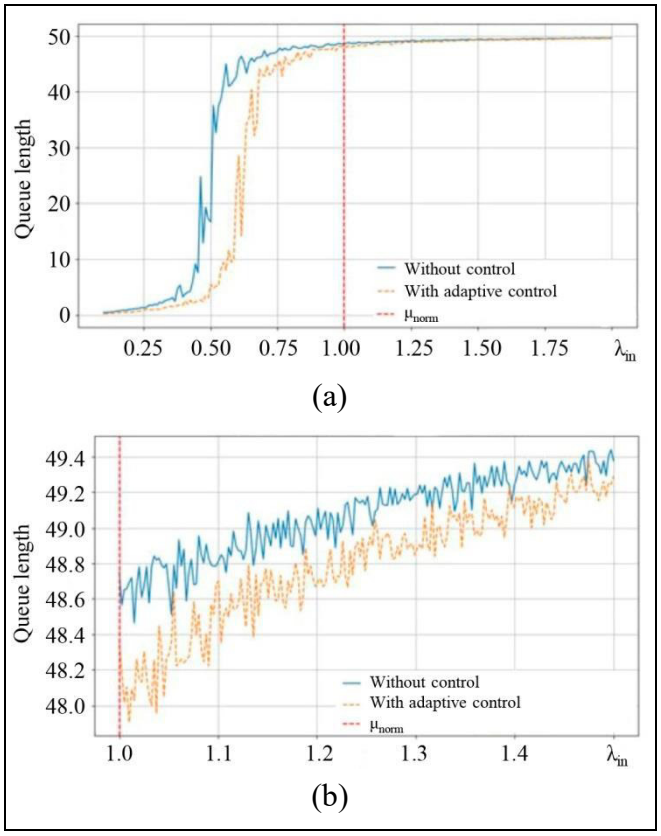

Fig. 6. Phase transition: average queue length as a function of λ_{in}.

The impact of semi-Markov effects is illustrated in Fig. 7: under the gamma distribution and distributions, the system becomes more inertial, which increases its sensitivity to overload.

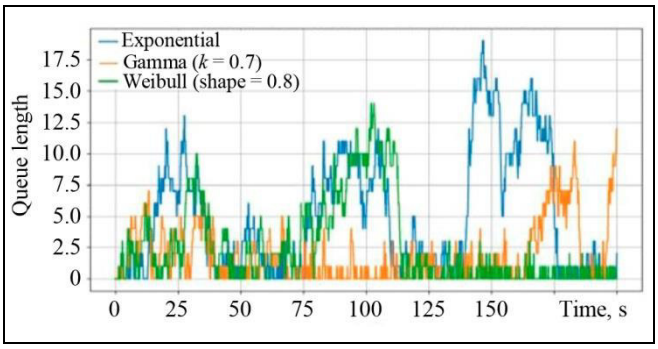

Fig. 7. Queue length dynamics under different distributions of service times.

CONCLUSIONS AND PROSPECTS

The model proposed in this paper describes an adaptive control system under stochastically varying characteristics of the communication channel, which is in one of three states: stable connection, degraded connection, and disconnection. As has been shown, introducing a control function $u(t)$ (an impact on both the service rate and the transition rates between states) allows one to adapt the system behavior dynamically to the current conditions. Unlike classical models with fixed parameters, this approach involves parameters as functions of the strategy to consider the impact of control actions on network performance.

According to the numerical analysis results, the system is stable for $\lambda_{in} < \bar{\mu}$: the queue remains bounded. In the case of overload ($\lambda_{in} > \bar{\mu}$), an avalanche-like growth of queue length occurs. However, adaptive control decelerates this growth and actually shifts the stability margin. The effects of temporal correlation have also been demonstrated: when passing to semi-Markov models, the system becomes inertial, but the general stability criteria are preserved. This feature emphasizes the universality of the model and its applicability in both classical and more complex telecommunication scenarios.

In practice, this model can be implemented in the form of a programmable unit embedded in the aggregation device. Such a module can utilize neural network schemes to estimate the current channel state and control transmission parameters. In addition, the model can be generalized towards Markov control (when solutions depend not only on the current state but also on the dwell time in it). This is naturally realized through systems with memory or logic with the history of states. It is also possible to realize intermediate communication states using multi-valued logic to de-

scribe not only abrupt but also smooth degradations of the channel.

Thus, the model combines stochastic dynamics, control, and queues into a single system and can be applied in theoretical stability analysis of such systems and in the development of new stacks of data transmission protocols, especially in the context of new-generation mobile networks. Prospects for further research include multiuser scenarios, the investigation of control strategies on data, and the construction of energy-efficient algorithms for devices with limited computational resources based on intelligent control. As an independent theoretical challenge, we note the study of phase transition; an open question is to eliminate the exponential nature of queue growth and decelerate adaptive control.

Acknowledgments. *The author is grateful to the reviewers for careful reading of the manuscript and helpful remarks.*

REFERENCES

1. Marchenkov, A.A. and Esin, A.A., Patent RU 2694025 C1, *Byull. Izobret*., 2019, no. 19.
2. Chang, C.S., *Performance Guarantees in Communication Networks*, London: Springer-Verlag, 2000.
3. Kleinrock, L., *Queueing Systems. Vol. 1*, New York: Wiley-Interscience, 1975.
4. Gross, D., Shortle, J., Thompson, J., and Harris, C., *Fundamentals of Queueing Theory*, New York: Wiley, 2011.
5. Baccelli, F. and Blaszczyszyn, B., Stochastic Geometry and Wireless Networks: Volume I Theory, *Foundations and Trends® in Networking*, 2010, vol. 3, no. 3–4, pp. 249–449. DOI: https://doi.org/10.1561/1300000006
6. Haenggi, M., Andrews, J.G., Baccelli, F., et al., Stochastic Geometry and Random Graphs for the Analysis and Design of Wireless Networks, *IEEE Journal on Selected Areas in Communications*, 2009, vol. 27, no. 7, pp. 1029–1046. DOI: https://doi.org/10.1109/JSAC.2009.090902
7. Baccelli, F. and Blaszczyszyn, B., On a Coverage Process Ranging from the Boolean Model to the Poisson–Voronoi Tessellation with Applications to Wireless Communications, *Advances in Applied Probability*, 2001, vol. 33, no. 2, pp. 293–323. DOI: https://doi.org/10.1017/S0001867800010806
8. Raigorodskii, A.M., Random Graph Models and Their Applications, *Proceedings of MIPT*, 2010, vol. 2, pp. 130–140.
9. Stauffer, D. and Aharony, A., *Introduction to Percolation Theory*, Milton Park: Taylor & Francis, 1994.
10. Saberi, A.A., Recent Advances in Percolation Theory and Its Applications, *Physics Reports*, 2015, vol. 578, pp. 1–32. DOI: https://doi.org/10.1016/j.physrep.2015.03.003
11. Olle, H., Yuval, P., and Jeffrey, E.S., Dynamical Percolation, *Annales de l'Institut Henri Poincaré (B) Probability and Statistics*, 1997, vol. 33, pp. 497–528.
12. Li, M., Liu, R.R., Lü, L., et al., Percolation on Complex Networks: Theory and Application, *Physics Reports*, 2021, vol. 907, pp. 1–68. DOI: https://doi.org/10.1016/j.physrep.2020.12.003
13. Badie-Modiri, A., Rizi, A.K., Karsai, M., and Kivelä, M., Directed Percolation in Temporal Networks, *Phys. Rev. Res*., 2022, vol. 4, art. no. L022047. DOI: https://doi.org/10.1103/PhysRevResearch.4.L022047
14. ElSawy, H., Zhaikhan, A., Kishk, M.A., and Alouini, M.S., A Tutorial-Cum-Survey on Percolation Theory with Applications in Large-Scale Wireless Networks, *IEEE Communications Surveys & Tutorials*, 2024, vol. 26, pp. 428–460. DOI: https://doi.org/10.1109/COMST.2023.3336194
15. Lin, H., Kishk, M.A., and Alouini, M.S., Performance Analysis of Infrastructure Sharing Techniques in Cellular Networks: A Percolation Theory Approach, arXiv:2502.08023, 2025. DOI: https://doi.org/10.48550/arXiv.2502.08023
16. Zhang, Q. and Kassam, S.A., Finite-State Markov Model for Rayleigh Fading Channels, *IEEE Transactions on Communications*, 1999, vol. 47, no. 11, pp. 1688–1692. DOI: 10.1109/26.803503
17. Paranthaman, V.V., Mapp, G., Shah, P., et al., Exploring Markov Models for the Allocation of Resources for Proactive Handover in a Mobile Environment, *Proceedings of 2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, Clearwater Beach, 2015, pp. 855–861. DOI: 10.1109/LCNW.2015.7365938
18. Kalimulina, E.Y., Application of Multi-Valued Logic Models in Traffic Aggregation Problems in Mobile Networks, *Proceedings of the 15th IEEE International Conference on Application of Information and Communication Technologies (AICT)*, Baku, Azerbaijan, 2021, pp. 1–6. DOI: 10.1109/AICT52784.2021.9620244
19. Kalimulina, E.Y., Lattice Structure of Some Closed Classes for Non-binary Logic and Its Applications, in *Mathematical Methods for Engineering Applications (ICMASE 2021)*, Yilmaz, F., Queiruga-Dios, A., Santos Sánchez, M.J., Rasteiro, D., Gayoso Martínez, V., and Martín Vaquero, J., Eds., *Springer Proceedings in Mathematics & Statistics*, Cham: Springer, 2022, vol. 384, pp. 25–34. DOI: 10.1007/978-3-030-96401-6_2
20. Goldsmith, A., *Wireless Communications*, Cambridge: Cambridge University Press, 2005.
21. *Semi-Markov Models and Applications*, Janssen, J. and Limnios, N., Eds., Cham: Springer Science & Business Media, 2013.
22. Limnios, N. and Oprisan, G., *Semi-Markov Processes and Reliability*, Cham: Springer Science & Business Media, 2012.
23. Cinlar, E., *Introduction to Stochastic Processes*, Hoboken: Prentice-Hall, 1975.
24. Ross, S., *Stochastic Processes*, New York: Wiley, 1996.
25. Kirk, D.E., *Optimal Control Theory: An Introduction*, Hoboken: Prentice-Hall, 1970.
26. Bertsekas, D.P., *Dynamic Programming and Optimal Control*, Raleigh: Athena Scientific, 2005.
27. Boyd, S. and Vandenberghe, L., *Convex Optimization*, Cambridge: Cambridge University Press, 2004.
28. Wang, T., Proakis, J.G., Masry, E., and Zeidler, J.R., Performance Degradation of OFDM Systems due to Doppler Spreading, *IEEE Transactions on Wireless Communications*, 2006, vol. 5, no. 6, pp. 1422–1432. DOI: 10.1109/TWC.2006.1638663
29. Yin, X. and Cheng, X., *Propagation Channel Characterization, Parameter Estimation, and Modeling for Wireless Communications*, Hoboken: John Wiley & Sons, 2016.
30. Modeas, I., Kaloxylos, A., Merakos, L., and Tsolkas, D., An Adaptive and Distributed Network Selection Mechanism for 5G Networks, *Computer Networks*, 2021, vol. 189, art. no. 107943. DOI: 10.1016/j.comnet.2021.107943

31.Kalimulina, E.Y., Mathematical Model for Reliability Optimization of Distributed Telecommunications Networks, *Proceedings of 2011 International Conference on Computer Science and Network Technology*, Harbin, China, 2011, pp. 2847–2853. DOI: https://doi.org/10.1109/ICCSNT.2011.61825570
32.Kalimulina, E.Y., A New Approach for Dependability Planning of Network Systems, *International Journal of System Assurance Engineering and Management*, 2013, vol. 4, pp. 215–222.
33.Esin, A.A., Analysis and Design Principles of Modern Control Systems Based on Multi-valued Logic Models, *Large-Scale Systems Control*, 2020, no. 88, pp. 69–98. DOI: https://doi.org/10.25728/ubs.2020.88.4 (In Russian.)
34.Esin, A.A., Characteristics of Structurally Finite Classes of Order-Preserving Three-Valued Logic Maps, *Logic Journal of the IGPL*, 2024, art. no. jzae128. URL: https://academic.oup.com/jigpal/article-lookup/doi/10.1093/jigpal/jzae128
35.Esin, A.A., Structural Analysis of Precomplete Classes and Closure Diagrams in Multi-Valued Logic, *Iranian Journal of Fuzzy Systems*, 2025, vol. 21, no. 6, pp. 127–145.
36.Gao, S. and Liu, Z., An M/G/1 Queue with Single Working Vacation and Vacation Interruption under Bernoulli Schedule, *Applied Mathematical Modelling*, 2013, vol. 37, no. 3, pp. 1564–1579. DOI: 10.1016/j.apm.2012.04.045
37.Zhang, M. and Hou, Z., Performance Analysis of M/G/1 Queue with Working Vacations and Vacation Interruption, *Journal of Computational and Applied Mathematics*, 2010, vol. 234, no. 10, pp. 2977–2985. DOI: 10.1016/j.cam.2010.04.010

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

Received May 17, 2025,
and revised May 19, 2025.
Accepted May 20, 2025.

Author information

Esin, Anton Anatol'evich. Development Director, MirWiFi JSC, Moscow, Russia
✉ anton.esin@imm.am
ORCID iD: https://orcid.org/0009-0007-0680-8288

Cite this paper

Esin, A.A., A Mathematical Model of Adaptive Traffic Control in Mobile Networks with Variable Signal Quality. *Control Sciences* 3, 52–66 (2025).

Original Russian Text © Esin, A.A., 2025, published in *Problemy Upravleniya*, 2025, no. 3, pp. 63–79.

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