applied sciences

MDPI

Article

Control of Linear Multichannel Objects with Numerical Optimization

Vadim Zhmud [1,2,*] and Lyubomir Dimitrov [3]

[1] Institute of Laser Physics SB RAS, Novosibirsk 630090, Russia
[2] Altai-Sayan Branch of Geophysical Survey of the Russian Academy of Sciences, Department of Monitoring of Crustal Deformations, Novosibirsk 630090, Russia
[3] Technical University of Sofia, 1000 Sofia, Bulgaria; lubomir_dimitrov@tu-sofia.bg
* Correspondence: oao_nips@bk.ru; Tel.: +7-9134732997

Abstract

The article is devoted to the control of multichannel objects by the method of numerical optimization. In this area, there are several not entirely accurate, but deeply rooted, ideas and approaches that we would like to analyze. This analysis is presented through typical examples and with the use of usually inexpensive software, which can also be obtained for free in a demo version. Using this software is easy with the help of intuitive buttons and options. Thus, solving problems in the design of a regulator for multichannel systems becomes accessible to anyone who has a mathematical model of the object. The principles of control of two-channel objects are considered, i.e., objects of size 2×2, which by induction can be extended to objects of size 3×3 and, possibly, even more. At the beginning, a solution to the problem of controlling a two-channel object of size 2×2 is proposed using a PID regulator of the same size or its simplest modifications. Then the problem with 3×3 objects is solved. There are two main factors that influence the result: the presence or absence of dominance of the absolute value of the diagonal elements of the object's transfer function matrix over the remaining elements; the selected optimization objective function. In the case of dominance, the control task is significantly simplified. In its absence, in some cases it can be ensured by changing the numbering of the channels. Recommendations for forming the objective function are given. For the first time, the use of a fractional degree in the objective function is proposed, and the effectiveness of this approach is justified and shown. For the first time, an additional modification of the test signals during optimization is proposed, and the effectiveness of this modification is shown. It is also shown how the control quality of some channels can be improved at the expense of some deterioration in the control quality of other channels.

Keywords: controller; automation; PID; MIMO; symmetry; modeling; optimization

check for updates

Academic Editor: Augusto Ferrante

Received: 5 July 2025
Revised: 31 July 2025
Accepted: 7 August 2025
Published: 13 August 2025

Citation: Zhmud, V.; Dimitrov, L. Control of Linear Multichannel Objects with Numerical Optimization. *Appl. Sci.* **2025**, *15*, 8927. https://doi.org/10.3390/app15168927

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Controlling objects with two inputs and two outputs, where each input affects each output, requires careful controller design [1–10]. An example of such an object is the simultaneous stabilization of the power and frequency of a semiconductor laser by controlling its current and temperature, the temperature being controlled by a Peltier-based micro cooler. Another example is the automatic mixing of several substances, when it is necessary to simultaneously control both the increase in the volume of the mixture and the concentration of each substance. A third example is the well-known problem of filling a swimming pool,

with the need to separately control the flow rate and temperature of the water. Similar problems arise in controlling the movement of a vehicle in air or water. There are many examples of such problems in robotics, chemistry, metallurgy, and scientific research.

Let us consider the problem of controlling linear objects. The proposed approach can also be applied to nonlinear objects, but the result obtained should be more carefully checked for the case of test signals of different amplitudes. In general, the problem of controlling nonlinear objects is much more complex. It is practically impossible to consider it in sufficient depth within the framework of one article, except for such types of nonlinearities as a limit or a relatively small deviation of the characteristic from the linear one (no more than 10%). The discussed method is unsuitable for controlling objects with nonlinearity of the "dead zone" type, as well as "dry friction", "hysteresis", or other ambiguous nonlinearities. The proposed method is insufficient for controlling nonstationary objects. Adaptive control systems are needed for such objects.

The mathematical model of a linear two-channel object is defined in the form of a matrix transfer function [11,12].

$$W(s) = \begin{bmatrix} w_{11}(s) & w_{12}(s) \\ w_{21}(s) & w_{22}(s) \end{bmatrix}. \quad (1)$$

There are many papers that consider objects of size n × n, but no articles have been found with reliable information about successful control in such a formulation of the problem, i.e., with multichannel negative feedback and a multichannel sequential controller, even for objects of a size of only 3 × 3. Therefore, the term "multichannel objects" should be recommended to be used more carefully than is currently accepted in the literature. Most often, "two-channel" objects are described even in such articles where "three-channel" objects are mentioned. In this article, we prove the validity of our thesis (established by induction) on the example of an object of size 3 × 3. If necessary, we are ready to go further, to objects of size 4 × 4, but we do not know such problems from practice, so we are in no hurry with these studies.

This approach should not be confused with multiparameter optimization problems or multicriteria optimization problems.

Control in a negative feedback loop is carried out according to the structure shown in Figure 1.

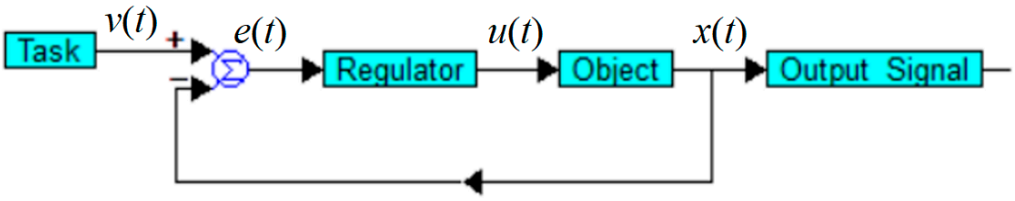

Figure 1. Traditional structure of a negative feedback system.

This structure best satisfies the requirement that the output signal $x(t)$ corresponds to its set value $v(t)$ ("task" or "target" value). It is necessary to ensure the equality $x(t) \approx v(t)$ as accurately as possible. This is ensured by a feedback loop that acts in such a way as to ensure the relationship $e(t) = v(t) - x(t) \approx 0$. Here, the element marked with a circle with a sign "Σ" inside is an adder, but if there is a minus sign before the input, this input is subtractive. This and the following illustrations are made in the VisSim 5.0e software, which has some features for displaying various mathematical symbols, numerical values, and other elements. In particular, the minus sign looks like a dash, blue rectangles indicate composite blocks that have a complex structure inside, and their name is set by the user. The arrows indicate the direction of the signals. When using the structure according to

Figure 1, the task of controlling an object is reduced to calculating a mathematical model of the regulator (controller) based on a known mathematical model of the object.

The paper [13] considers the problem of adaptive control of multichannel objects using neural networks in the class of fractional nonlinear feedback. The paper considers three examples with sizes 2×2, 3×3, and 2×2, respectively.

In the paper [14] the problem of quantized control of a heavy launch vehicle with failures of the propulsion mechanism and gyroscope is analyzed. For monitoring the unmeasured derivative, a predetermined-time observer is proposed. The parameter setting in this case is simpler than for fixed-time observers.

In [15] two low-gain controllers for two types of systems are proposed. The same authors in [16] propose an observer based on a partial differential equation. In these papers, as in many similar ones, the matrix representation of the object equations is not a consequence of the multichannel nature but a way to simplify the description of a high-order object, while the controlled object remains single-channel.

In [17], an optimal preliminary iterative control problem based on the Padé approximation with equivalent input disturbance is proposed for a class of continuous-time indefinite systems.

In [18], an adaptive nonlinear current controller with active noise suppression is proposed for a distributed system to solve the large ripple problem. In addition, an adaptive internal model controller is built into the current controller to suppress periodic noise.

In [19], a time-synchronized controller is proposed. Such a design leads to an increase in control efficiency compared to traditional sliding modes. This efficiency is probably associated with the elimination of the self-oscillating mode when switching between different modes at a certain threshold level in noise conditions. Here, it would obviously be more efficient to introduce hysteresis so that the hysteresis value exceeds the noise from determining this threshold. If the sliding mode switches to a strong oscillation mode due to an unaccounted delay, then synchronization of the controller will not be able to eliminate this problem. For example, such a method was used in the article [20], where microwave heating is needed to combat ice on the contact wires of railway tracks. For the microwave generator to not operate continuously, but only when necessary, a tracking system is used. This system uses a PID controller in which a constraint is introduced.

The paper [21] has many external signs of connection with the topic of this article: it contains the term MIMO, uses a functional matrix to describe the connections between elements, and talks about the control of unmanned aerial vehicles. However, the topic of this article is different. It talks about ensuring optimal stable communication between individual aerial vehicles.

In [22], an adaptive event-driven PI control method is proposed for a class of nonlinear multiple-input multiple-output (MIMO) systems with indefinite input delay. For this purpose, a new variable is developed based on the Padé approximation and the Laplace transform. Then, a nonlinear controller with a PI structure is developed and an event-driven control strategy for dynamic communication resource allocation based on the nonlinear PI controller is established. With the proposed method, nonlinear MIMO systems are able to dynamically handle input delays of varying lengths while maintaining sufficient tracking performance.

In [23], the issue of adaptive finite-delay memory-driven lane control for autonomous heavy-duty trucks with rollover avoidance under nonlinear corner stiffness is addressed. A lane delay with a defined characteristic is used to achieve the expected convergence properties. In addition, problem-specific technical solutions are applied.

In [24], an adaptive pseudo-inverse control scheme based on a fuzzy logic system and a Lyapunov barrier function is proposed for a class of hysteretic nonlinear systems where all states are always strictly constrained in each set of constraints. In this paper, the hysteretic nonlinearity in actuators is considered and mitigated by the proposed pseudo-inverse control algorithms, which means that a direct inverse hysteresis model is not needed, but instead a mechanism is used to find the actual control signal from the temporal control signal. Also, the constraint management problem for all states of the hysteresis model is solved when the control signal is coupled into double integral functions using the mentioned methods and the proposed pseudo-hysteresis inverse algorithms.

In [25], an integrated framework including a distributed controller is proposed to systematically solve combined problems related to the control of the movement of a group of automated vehicles. A two-stage curvilinear coordinate scheme is proposed. In addition, constraints on both longitudinal and transverse acceleration are taken into account. The constrained longitudinal acceleration is included in the design of a nonlinear saturated controller, the convergence of which is proven. Adaptive fusion is implemented using a constrained transverse controller. A condition for limiting the initial states is formulated to ensure safe maintenance of the distance between vehicles.

In [26], a performance-specific adaptive robust control (PPARC) scheme for uncertain robotic arms is introduced. First, a state transformation is introduced to embed predefined output constraints into the constraints of the trajectory tracking servo. Second, PPARC is designed to perform reference trajectory tracking and ensure that the tracking errors are within the predefined output constraints regardless of the uncertainty and deviation of the initial conditions. Then, Lyapunov stability analysis is carried out to prove the uniform boundedness and uniform finite boundedness of the tracking error. Moreover, the control parameter optimization is transformed into an optimal design problem, and a fuzzy cost function is proposed for the optimal design problem. The existence and uniqueness of the solution of the optimal design problem are theoretically proved. Finally, the effectiveness and superiority of the proposed control scheme are demonstrated based on simulation modeling.

The article [27] studies the problem of formation-circle flight switching control for several heterogeneous omnidirectional intelligent navigation systems in an intermittent communication network. For each system, a distributed hybrid finite-time observer is proposed to evaluate target state information. The article does not propose fundamentally new control methods from the standpoint of automatic control theory.

In the article [28] a method of control with provision of stability in finite time is used and developed. The developed modified theory is applied for the example of control of a pendulum, that is, a scalar stable object, since the pendulum is not inverted.

The paper [29] presents a new predictive control scheme based on a learning economic model for uncertain nonlinear systems subject to input and output constraints and unknown dynamics. The paper develops a Lipschitz regression method that combines clustering and kernel regression to study the unknown dynamics. Sufficient conditions are derived to ensure recursive feasibility and stability of the closed-loop system from input to state.

In the paper [30], a modified control algorithm with active disturbance suppression based on feedforward compensation is proposed, aimed at the problem of train speed control under external disturbances, which reduces the dependence on the train model. In the mathematical model of the system studied in this paper, there are several output value sensors and several tasks, namely: two output values are estimated, as well as the derivative of one of them. However, this system cannot be considered multichannel, since all control signals are summed into a single scalar control signal received at the control input of the object.

In the paper [31], a quantized controller is presented for solving the problem of fast finite-time synchronization of multilayer networks, where each layer represents a separate type of interaction in complex systems. Based on the stability theory, a new criterion for fast finite-time stability is derived, which allows setting a smaller upper limit on the settling time compared to the general finite-time stability. By converting continuous error signals into piecewise continuous forms, a quantized control scheme is used to implement fast finite-time synchronization in multilayer networks, which allows saving control resources and reducing communication overload. In this paper, the matrix description also appears due to the high order of the object and not due to the large number of channels.

The paper [32] considers cooperative localization of multiautonomous underwater vehicles. This paper proposes a filtering algorithm for a slave underwater vehicle with a failed compass, based on the "two masters and one slave" cooperative localization model. The proposed algorithm allows for accurate estimation of the change in the unknown heading angle without expanding the state dimension under non-Gaussian measurement noise.

The articles reviewed solve similar or close problems, but the formulation of the problem in these articles is still significantly different and, in most cases, it is tied to the particular features of the problems being solved. The technical and theoretical solutions developed and proposed in these articles are not applicable to the class of problems considered in our article.

2. Statement of the Problem

In the early stages of the development of the theory of automatic control, the control task was posed and solved from the point of view of finding a control signal that should be fed to the input of the object so that its output values (called output signals) coincide with the "task" prescription. This approach is impractical. The task can change; therefore, the calculation of the control signal loses its relevance. In addition, the object is always affected by factors that cannot be measured and considered in such a statement of the task. Using the structure according to Figure 1 is a different statement of the task. Instead of calculating the signal controlling the object, a mathematical model of the device is calculated, which only calculates the required control signal depending on the task and the actual output signal of the object. A characteristic property of systems with negative feedback is that, with the correct calculation of the regulator, the control task is successfully solved even in the presence of unknown and uncontrolled influences on the object, which change its output signal in an unpredictable way. In the frequency band, where the transfer function of the controller and the object connected in series with it is much greater than unity, the output signal matches the task quite accurately.

In this case, the difference between the output signal $x(t)$ and the task $v(t)$, called the error $e(t)$, is inversely proportional to the product of the controller transfer function $W_C(s)$ and the object transfer function $W_O(s)$. If $W_O(s) \to \infty$, then $e(t) \approx Q(t)/W_O \to 0$. Here $Q(t)$ is all types of disturbing effects on the object, W_O is the approximate value of the transfer function of the object in a given frequency range, $s = -c + i\omega = -c + i2\pi f$ is the argument of the Laplace transform, similar to the frequency, but having a small negative real value in addition to the complex component $i\omega$.

The problem with using this method is that each object has a transfer function $W_O(s)$, limited by some frequency band. With increasing frequency, the transfer function decreases $W_O(s) \to 0$, although in some local frequency ranges it can increase. However, starting from some frequencies, it decreases so sharply that no amplification of the input control signal $u(t)$ can achieve a significant response at the object output. For this reason, the transfer function of the entire loop in a conditionally open form inevitably reaches a value

close to unity and then decreases even more. In the frequency range where this value is much less than unity, the control loop no longer has a noticeable effect on the object, so its output signal in this frequency range is not controllable. We must put up with this. But in the intermediate frequency band between those where the transfer function of the circuit is much greater than unity, which allows control of the object, and those where the transfer function is much less than unity, which excludes the possibility of controlling the object, there is a frequency region where the transfer function of the circuit is in the range from 0.1 to 10, i.e., from -20 dB/dec to $+20$ dB/dec. In this frequency range, the behavior of the transfer function of the loop affects the stability of the system. If at these frequencies the signal delay exceeds 180° (or π *rad* which is the same value in other measure units) then the system will be unstable. Therefore, the art of the designer of locked dynamic systems consists in calculating such a controller that would ensure the stability of the locked system, as well as its sufficient accuracy in the frequency region where control is provided.

For multichannel systems, this is not enough. An additional property called "decoupling" or diagonal decoupling is needed. It consists in the fact that any change in the task signal at any input should not only provide a corresponding change in the output signal at the corresponding output but also should not affect the output signals of all other inputs. This requirement can be met only approximately. There was a scientific direction called "invariant control" which studies such control methods so that the control error is always strictly zero, but this area of automatic control theory has lost its relevance. The object is always affected by uncontrolled disturbing factors, which are usually described as an unknown interference, which is applied to the output of the object in the form of an additive unknown signal. Instead of the unrealistic task of completely eliminating the influence of disturbance, the modern section of control theory dealing with high-precision systems considers the problem of limiting the error to a certain value measured as a percentage of the task increment. An error of less than 5% of the task was considered acceptable at the early stage of automatic control theory. In modern high-precision systems, requirements are imposed to suppress the error to values of no more than 0.0001% and, in some cases, to an even smaller value. In problems of controlling multichannel objects, as a rule, the error is reduced to values of the order of 1–5%. In this case, we are talking about a dynamic error, since a static error, i.e., an error that is established after a sufficiently long time, provided that the task signals are step functions, can be reduced strictly to zero. This error does not include errors in measuring the output signal, which can never be reduced strictly to zero.

Thus, the problem of controlling an object, given the above, is reduced to the problem of designing a controller for an object with a known mathematical model, using the structure shown in Figure 1. Other structures and other formulations of the control problem are also known, which we do not consider in this paper.

The controller for a linear object (1) can be represented as a matrix transfer function [33–36]. For example, for an object with two inputs and two outputs, such a transfer function has the following form:

$$R(s) = \begin{bmatrix} r_{11}(s) & r_{12}(s) \\ r_{21}(s) & r_{22}(s) \end{bmatrix}. \quad (2)$$

In the simplest case, this matrix is diagonal, that is, only the diagonal elements are nonzero, and the remaining elements are equal to zero:

$$R(s) = \begin{bmatrix} r_{11}(s) & 0 \\ 0 & r_{22}(s) \end{bmatrix}. \quad (3)$$

This means that there are only feedback connections from each output to the corresponding input and no other connections.

In more complex structures, additional loops may appear, for example, a loop that covers an object in addition to the device that subtracts the output signal from the task. There may also be a loop that covers the controller with feedback. The feedback may contain a transfer function that differs from unity. Ultimately, such modifications can be reduced by equivalent transformations to the structure shown in Figure 1, with the difference that the controller's transfer function will be different, more complex.

The choice of a controller containing parallel proportional, integrator, and derivative channels (which is therefore called a PID controller) is determined by the fact that for many practical tasks this choice is sufficient and also by the fact that simpler structures in which one or two of the specified paths are missing are a special case of such a controller. These simpler structures can be obtained in the same way as in the design of a PID controller, which does not change the approach to the problem of controlling an object of type (1).

The development of computer technology and software has simplified the task of designing controllers for complex systems to the maximum [37,38]. For this reason, it is not even necessary to think about whether the problem can be solved using only a diagonal matrix controller or whether a full-fledged multichannel controller with all nonzero elements of the matrix transfer function is needed [39–48]. In many cases, it is enough to simply calculate a full PID controller, which in practice for objects of size 2×2 and 3×3 turns out to be slightly more difficult than calculating simplified diagonal controllers. This will be demonstrated below.

The most common examples in the literature of multichannel objects are objects with dimensions 2×2. In this case, the object model is a matrix transfer function of the same dimensions.

For control, a matrix PID controller of the same size is required [33–46]. A more complex controller can also be considered, for example, containing additional channels for double differentiation, double integration, or some more complex structural solution. For a PID controller, its individual elements are described by the sum of the proportional, integrating, and derivative channels with the corresponding coefficients:

$$R_{ij} = k_{Pij} + \frac{k_{Iij}}{s} + k_{Dij}s = \frac{k_{Pij}s + k_{Iij} + k_{Dij}s^2}{s}. \quad (4)$$

The structure of the control system is shown in Figure 1.

3. Theory of Solving Problems of Multichannel Object Control

The theory of locked dynamic systems is well-developed: negative feedback ensures accuracy, provided that the entire system is stable. Structural design consists of choosing the structure of the controller. Parametric design consists of determining the parameters of the controller based on the chosen structure.

Structural design is not the subject of this article, since PID controllers are the most common, and the reason for this is their simplicity and efficiency. Simpler controllers, such as PI and PD, are not considered, since this simplification does not greatly simplify the method but limits its capabilities. Complication by introducing additional channels of double integration or (and) double derivation is also not considered, since the approach under consideration can easily be extended to these structures. Structures with a large number of additional loops, such as the Smith predictor, local and pseudo-local connections, etc., are not considered due to their excessive complexity and low efficiency, which has also already been confirmed by modeling and practice.

Parametric design in our case is the main content of the system design task. The methods for calculating the controller parameters can be grouped into the following broad categories: analytical, engineering, tabular, empirical, and numerical.

Analytical methods are based on the direct solution of matrix equations, which leads to the Riccati equation. If the matrix transfer function is not degenerate, it can be inverted. The inversion of this matrix is one of the important elements for the analytical solution of the problem under consideration. Even if the control problem is formulated in a different mathematical formulation, analytical methods cannot cope without inverting the matrix transfer function of the object.

The inverse matrix is a matrix that, when multiplied on the right or left by the original matrix, yields the identity matrix. If the matrix transfer function has at least one delay link, the inverse matrix transfer function yields a transfer function that cannot be implemented as a real device.

Let us look at an example.

$$W(s) = \begin{bmatrix} e^{-\tau s} & 0 \\ 0 & e^{-2\tau s} \end{bmatrix}.$$

Inverting this matrix gives the following result:

$$W^{-1}(s) = D(s) = \begin{bmatrix} e^{\tau s} & 0 \\ 0 & e^{2\tau s} \end{bmatrix}.$$

It is easy to verify that multiplication from the left or right by this matrix yields a unit matrix. But the resulting matrix $D(s)$ consists of functions that cannot be implemented in any physical device. The fact is that the function $e^{-\tau s}$ describes in operator form (in the form of Laplace transforms) such a transformation of the input signal, which consists in delaying the input signal by time τ. However, at the same time, the transfer function $e^{\tau s}$ describes such a transformation, which forms an input signal at the output of the element with a negative delay, that is, τ seconds earlier than this signal arrived at the input of the element. Such a "signal prediction link" cannot physically exist, it is something like a hypothetical "time machine". Thus, if there is a delay in the model of the object, then analytical methods cannot be applied to the calculation of the regulator for such an object, or it is necessary to invent methods for reducing the delay to some other mathematical description, for example, to the Padé approximation. But such an approximation makes all calculations excessively complex and yet unreliable, and the result of the calculation is unreliable. The main objection to this approach is based on the fact that any mathematical model of any real object that does not take into account the delay of this object is insufficiently detailed for its effective use. This thesis is substantiated in Appendix A.

Tabular methods such as the Cohen–Kuhn, Ziegler–Nichols, Cohen–Coon, and Chien–Hrones–Reswick methods and similar ones [29] are always unreliable, so they are hopelessly outdated. These methods assume that the mathematical model of the object is described accurately enough by a series connection of a first- or second-order filter and a delay element:

$$W(s) = \frac{k}{1 + 2\xi Ts + T^2 s^2} e^{-\tau s}. \quad (5)$$

Tabular methods suggest obtaining the graph of the transition process as a response to a single jump, after which this graph is used to determine several basic parameters of this model. Using the table, the parameters of the PID or PI controller for such an object are determined. A modification of this method is another method in which the object is covered with proportional negative feedback, after which the coefficient of this feedback

is selected so that undamped oscillations with constant amplitude are established in the system. Then, using the frequency and amplitude of these oscillations, the coefficients of the PID controller are also calculated from the table. All these methods rarely give an acceptable result and never give the best possible result. The problem with them is that they often lead to an unstable system.

Some authors report that, after using one of the tabular methods, they slightly adjusted the coefficients, which yielded satisfactory results. However, in this case, it must be acknowledged that the tabular method could not be used at all. It would be sufficient to set arbitrary coefficients for the controller and then adjust them. This empirical method of adjusting controllers is widely used. It is used much more often than is indicated in the articles, because the fact that the empirical method is used does not justify writing a scientific article. The controller coefficients are simply increased sequentially until the stability of the system is disrupted and then reduced to a value at which stability is restored.

Engineering methods consist in constructing an asymptotic logarithmic amplitude–frequency characteristic, as well as a phase characteristic with the subsequent use of the logarithmic Nyquist–Mikhailov stability criterion. These methods work well for single-channel systems but have not been developed for multichannel systems and it is too late to develop them, since they are morally obsolete. Before the active use of computing technology, the construction of logarithmic graphs was much more efficient and simpler than the construction of graphs on a linear scale. An effective technique has been developed for constructing such graphs and their use for analyzing the stability of the system, as well as for ensuring stability by correcting the logarithmic amplitude–frequency characteristic. A method is also known for calculating a mathematical model of a correcting device based on the obtained logarithmic amplitude–frequency characteristic of this device. Currently, the construction of logarithmic graphs that approximately describe the control object is already a more complex technique than numerical optimization. Thus, for linear objects with delay, the best method is the numerical optimization method if the object has at least the simplest nonlinear component. Then the numerical optimization method is the only reliable method that allows obtaining adequate results with the least labor costs.

The most reasonable approach is to make a preliminary attempt to design a PID controller [49–73], since simplifying it does not significantly simplify the design procedure and complicating it may not be effective enough.

The claim that more complex structures are not efficient is based on the fact that many types of regulatory complexity lead to coarse systems. Systems in which a slight deviation in the values of at least one of all the coefficients of the mathematical model of an object or regulator causes significant and even fatal changes in the nature of the transient process, including a violation of the stability of the entire system, are called coarse. Such methods leading to the indicated undesirable results include the Smith prediction method, the high-order derivative method, the compensator method—an element connected in series with the controller so that in this element the numerator of the transfer function contains in whole or in part the problematic coefficients of the denominator of the transfer function of the controlled object.

The advantage of numerical optimization is the ability to automatically obtain results. All that is needed is a regular computer, standard software, and a mathematical model of the object, as well as knowledge that can be obtained from the relevant literature, taking into account the recommendations of this article. The disadvantage of the numerical optimization method is the inability to solve the problem. Each solution is relevant only for a mathematical model with specific numerical values of all parameters of this model. However, some deviations of the actual values of the model parameters from the values used for the calculation are still permissible. If the solution remains valid with a deviation

of these parameters of 1–2%, the solution can be called rough. If the solution remains valid even with a deviation of 5–10% in the parameters, such a solution would be clearly sustainable. However, this is not always possible. If the model parameters can deviate from the calculated values, then an adaptive control system is needed, which is not considered in this article.

Before describing the solution to the problem, we will establish the requirements for the transfer function of the object (1) necessary for there to be a solution to the problem. First, the matrix transfer function of the object (1) should not be singular. Its rank should coincide with its dimensions; in this case, the rank should be equal to 2. In addition, it is highly desirable, although not necessary, that with $s = 0$ the matrix $W(0)$ should also be nonsingular. If this matrix is singular, then control in the static mode will have some features [73]. They consist in the fact that, even in a static mode, the control signals cannot be static. That is, to ensure that the output signals of the object take the set value in a stationary mode, it is not enough that the control signals entering the input of the object have some necessary static value. These signals will have to change over time, even to ensure a stationary output state. This can be proven theoretically and experimentally [57].

4. Method for Solving the Problem

The solution method consists of using the following input tools: (a) modeling and optimization software; (b) test signal; (c) modeling conditions (sampling step and modeling time); (d) objective function; (e) optimization method; (f) method for analyzing the results; and (g) method for improving the result if it is not good enough.

Naturally, the question arises: "If the result is optimal, then how can we raise the issue of its further improvement?"

However, the point is that the objective function is not the only option that follows from the management objectives. For a set of management objectives, a wide variety of objective functions can be justified, formed, and used. In addition, the result also depends on the modeling conditions and weighting coefficients in these objective functions, on which the method for further improving the result is based.

The optimization method deserves special comments. If the optimization is completed successfully, then it does not matter which optimization method was used. Any optimization method that gives a result should give the same result within the permissible error. If different optimization methods give different results, this indicates an error in solving the problem. Most likely, this can happen when the objective function has several local extrema. In this case, the designer can distinguish a local extremum from a global one by the value of the objective function at these points. If the minimum of the function is sought, then at the point of the global minimum the objective function takes the smallest value, and at the points of the local minimum its value is greater.

If one of the numerical optimization methods does not lead to the desired result, and another method does, then the system designer can simply use the results obtained by a more effective method, and there is no need to think about the reasons why the other method was unsuccessful. Therefore, if someone managed to purchase the necessary product in one store, he is not interested in the reasons for the absence of this product in other stores. In general, different optimization methods can be theoretically analyzed, and the different effectiveness of different methods can also be theoretically justified, although experimental justification is quite sufficient to move forward.

If none of the numerical optimization methods built into the software leads to the desired result, it is necessary to establish the reasons for this. The instability of the optimization procedure is most often a consequence of an insufficiently effective objective function. The objective functions proposed in this article and their modifications do not

create problems, except in cases where the problem cannot be solved by numerical optimization methods due to the lack of a global minimum for the mathematical model of the object used. This issue is also discussed below in Appendix A.

In VisSim 5.0e, as in other versions of this software, there are two types of optimizations. One of these types is the search for parameters that give the objective function a zero value. We do not use this type of optimization for the problem under discussion. The other type of optimization that we use is the search for parameters that give the minimum of the positive-definite objective function, which in this case is called the "cost function" or simply "cost". We will use this name further.

The methods we developed were applied in many robotic systems; the most illustrative example of a device in which they were applied is given in Appendix B.

VisSim has three built-in numerical optimization algorithms. The Powell method, the Polak–Ribière method, and the Fletcher–Reeves method. These methods are well-known enough to be described in this article. The possibility of using them is entirely due to the developers of this software. We have repeatedly established empirically that Powell's method is the best for this class of problems. In many cases, when one of the other two methods fails, or even both, Powell's method still succeeds, leading to a set of optimal parameters, i.e., parameters that give a minimum value of the cost function. If any of the other methods works, then it gives exactly the same result as Powell's method. For this reason, there is no point in using the other two numerical optimization methods. And there is no point in describing the algorithm of Powell's method. Some references to these methods are provided in Appendix C.

Powell's method is based on the calculation of tangents and the assumption that any line passing through an extremum intersects tangents to points with the same value of the objective function at the same angle. The other two methods are based on the calculation of the gradient of the objective function, which is likely to be unstable near the extremum. At these points, the gradient is small, and the influence of noise or random factors can lead to wandering around the extremum. However, if problems do not arise (discussed in Appendix A), then all three built-in optimization methods can give the same result. We do not propose a new method for numerical optimization. We use a ready-made method that can also be used by any reader using any version of the VisSim 5.0e software.

The proposed method for calculating the controller parameters has been developed and described in detail in our publications [11,12,37,38]. To solve the problems of controlling multichannel objects, some modification of the cost function is required.

The cost function for this case is the sum of the cost sub-functions for each individual channel [73]. For each channel, such a cost sub-function is the sum of the integral of the error modulus $e(t)$ multiplied by the time from the start of the transient process t during the process T and penalty additives $F_i(T)$. One of the most effective penalty additives is the positive part of the product of the error and its derivative [11,12,37,38].

The first member of the cost function has the following form:

$$F_1(T) = \int_0^T |e(t)| f_{ST}(t) dt \tag{6}$$

For the multichannel case, when several test jumps are fed to the system inputs, instead of the multiplier t in Equation (6), a sawtooth function $f_{ST}(t) = SawTooth(t)$ should be used, which increases linearly over the intervals between jumps but is reset to zero with each new jump.

The second term is determined by the following relationship:

$$F_2(T) = k_w \int_0^T \left[\max\{0, e(t)\dot{e}(t)\}\right] dt. \tag{7}$$

Here

$$\max\{0, M\} = \begin{cases} 0, \ if \ M < 0 \\ M, \ if \ M > 0 \end{cases}. \tag{8}$$

$$\dot{e}(t) = \frac{e(t)}{dt}. \tag{9}$$

Thus, the cost function has the following form:

$$F_{\Sigma}(T) = \sum_{m=1}^{n} F_{1m}(T) + F_{2m}(T). \tag{10}$$

This cost function is justified by the fact that it implements the necessary principles imposed on it. Terms of the form (6) provide growth (10) in the event that the control error does not decay quickly enough. Thus, the optimization procedure will find such values of the controller coefficients at which the system will not only be stable but the control error will decay quickly enough, so that the integral of the absolute values of all errors will be small. Weighting coefficients $f_{ST}(t)$ will ensure that, the more time that has passed from the onset of the transient process, the greater the contribution of the residual error to Function (10). Terms of the form (7) provide an increase in Function (10) in the event that the error in the system grows. Indeed, in this case the error itself and its derivative will have the same sign. This term will be small if the error grows by a small amount, or if it grows very slowly. Therefore, such a term will form the growth of the cost function as penalties for the growth of the error in magnitude quickly and by a large value. Using this term allows eliminating overshoot, oscillations, and nonmonotonic movements in the resulting system. If the error and its derivative have different signs, their product will be negative and the limiter cutting off the negative part will form zero at its output and, in this case, this term will not affect the cost function (10). The result of optimization will be such a set of regulator coefficients, at which (10) takes a minimum value, that is, the contribution of the first term and the penalties from the second term will be minimal.

The following modifications of the cost function components can also be used:

$$F_{1*}(T) = \int_0^T \left| e(t) \right| t^2 dt. \tag{11}$$

$$F_{2*}(T) = k_w \int_0^T \left[\sqrt{\max\{0, e(t)\dot{e}(t)\}} \right] dt. \tag{12}$$

$$F_{2**}(T) = k_w \int_0^T \left[\sqrt[4]{\max\{0, e(t)\dot{e}(t)\}} \right] dt. \tag{13}$$

The rationale for using the square of time in (11) is that the error decreases too slowly. This is because the contribution of the small error to the cost function is too small. The square of the function marking the time since the start of the transient process requires a faster reduction of the error. The more time that has passed since the start of the transient process, the more urgent the need to reduce the error modulus.

The rationale for using the square root in Relation (12) is very natural. It has never been noted or applied before. In fact, if the square root is not used, then the function $F_{2*}(T)$ acquires a quadratic dependence on the error and, therefore, on the test signal, while the main terms of the cost function defined by Relation (6) depend on the amplitude of the input signal not quadratically but only linearly. Therefore, the use of Relation (12) instead of Relation (7) is strongly recommended in all cases. The cubic root Variant (13) is derived by induction: since Variant (12) turned out to be better than Variant (7), it is advisable to try the fourth degree as well. This gives a dependence on the amplitude of the error in the form of a square root. In this case, the contribution of small errors increases compared to

the contribution of large errors. There is no theoretical basis for such a solution, but practice sometimes shows the usefulness of such a solution. Note that the nonlinear dependence of the objective function on the amplitude is not fatal, since in the end we still get a linear controller. Then all the results obtained for a linear object remain linear, i.e., when the input signals increase or decrease, the system behaves in the same way, but the output signals increase or decrease accordingly (the superposition principle is valid), as in any linear system.

The examples below show for what purpose it is possible to use a fractional power of the values included in the objective function, in particular the square root of the signal or the fourth root. The rationale for using the square root is that the product of the error and its derivative is a quadratic form that is proportional to the square of the error amplitude. This is not desirable because we are optimizing a linear system. The result should be invariant with respect to the value of the test signal. Using the square root is a natural solution that has not yet been published. Modeling has shown that this improves the performance of the cost function.

The purpose of using the fourth root is based on two arguments. The first argument is that, if introducing a square root improves the result, it is useful to try to apply a higher-order root by induction. The second argument for this approach is that such a nonlinear function increases the contribution of small signals. This allows for the reduction of small deviations of the signal from the desired value, when large deviations are already sufficiently suppressed.

During modeling, it is necessary to feed a test signal to the system model, which represents a task for the system, that is, a vector of input signals describing the desired values of the vector of output variables of the control object.

Since independent control of each output is required, the individual components of this vector must be substantially different functions and there must be no linear relationship between them. Traditionally, when optimizing a scalar linear system, a signal in the form of a single step jump is used [11,12,37,38]:

$$v(t) = \sigma(2\Delta t) = \begin{cases} 0, t < 2\Delta t, \\ 1, t \geq 2\Delta t \end{cases}. \tag{14}$$

Here Δt is the time discretization step. For this reason, the task vector is proposed as $v(t)$ in the form of stepwise jumps, shifted relative to each other by half the time of the transient process simulation [74].

$$V(t) = \begin{bmatrix} \sigma(2\Delta t) \\ \sigma(T/2) \end{bmatrix}. \tag{15}$$

In this case, instead of a multiplier proportional to the time from the start of the transient process, a sawtooth signal should be used, linearly increasing from the moment $t = 0$ to the moment $t = T/2$, after which it resets to zero and again linearly increases from zero at the same rate.

A delay in the occurrence of a step jump by at least the value equal to the doubled time sampling step Δt is a mandatory condition that we have not encountered anywhere in the literature but on which we have grounds to insist. If a jump at the input occurs without such a delay, then the simulation results may not coincide with the practical results, since in this case the system processes not the jump but the initial state of the object, and the results of the signal differentiation in the system in this case will differ. A reliable check of the quality of the system is based on its operation when feeding step jumps and not on the type of processing of the initial state error. In the practice of modeling and optimization,

we have had cases when the modeling results in these two noncoincident modes were noticeably different.

VisSim 5.0e software (in any version) contains a mixture of functions in the time domain and functions presented as transfer functions, i.e., in the Laplace transform domain. There is no contradiction in this, since no error will arise. The mathematical notation of equations in such a mixed form would be erroneous, but the mnemonic designation of the blocks used in programming the project is quite understandable and for this reason acceptable. The VisSim 5.0e software, designed for modeling and optimizing automatic control systems with feedback, uses various mathematical models. It contains models of linear elements that are described by transfer functions in the Laplace transform domain, as well as models of nonlinear connections that are described in the time function domain and other elements. The software does not require uniformity of terminology, since it is not necessary to create a full mathematical model of the system in the form of the one-type equations in the standard form for all of the forms as correct mathematical descriptions. The model is created from individual elements, between which, when this model is graphically specified, the appropriate connections are established from the inputs to the outputs; different ways of describing individual elements are not a problem.

5. Results

5.1. Two-Channel Object with a Dominant Main Diagonal

Example 1. *Let us consider an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{2}{1+2s+s^2} \cdot e^{-0.5s} & \frac{2}{1+3s+6s^2} \cdot e^{-0.8s} \\ \frac{0.5}{1+4s+2s^2} \cdot e^{-0.7s} & \frac{4}{1+4s+2.5s^2} \cdot e^{-0.3s} \end{bmatrix}. \quad (16)$$

We consider such objects for the following reason. A set of simple hypothetical models most accurately reflects all aspects of the possible problems considered in this article. Since not the only favorable type of model was used, but the reverse variant was also studied, then the point of view on the selected part of the problem can be treated as studied. More complex objects of the class considered in this article are characterized by positive poles of some polynomials in the denominators of most specific elements of the transfer function. In this article we do not consider such problems. This is in our plans.

If the polynomial of the transfer function denominators has no positive roots, then any example of the same order (an object of dimension 2 $\times$ 2, polynomials in the denominator of dimension 2, polynomials of order zero in the numerator) will not be more difficult than the examples considered in the article. If the polynomial of the transfer functions has positive roots, then slightly different methods are necessary, including, for example, a nonlinear controller or some others.

When modeling in the VisSim software, the object is formed from four models, sa in Figure 2. It should not be surprising that the coefficients of the free terms in the denominators of all transfer functions are equal to unit. This is the canonical form of these functions, because if this were not the case, then the numerator and denominator of these transfer functions would be sufficient to be divided by these coefficients to bring them into canonical form. In this form, the transfer function at $s = 0$ becomes equal to the coefficient in the numerator.

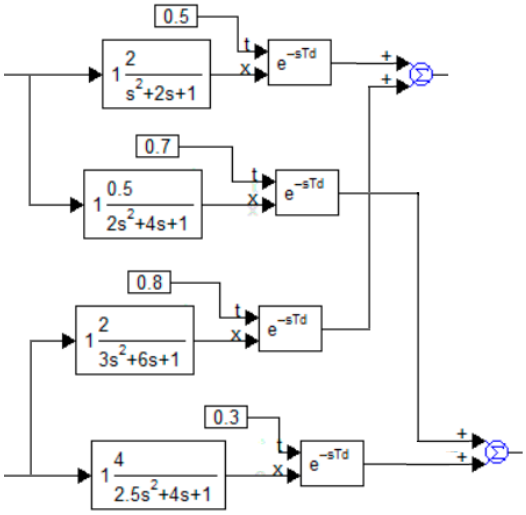

Figure 2. Example 1 of object.

Figure 3 shows the design for optimizing the controller for this case. In this figure, the individual PID controllers and the transfer function elements of the plant are combined into composite blocks called “PID” and “Compound”, respectively.

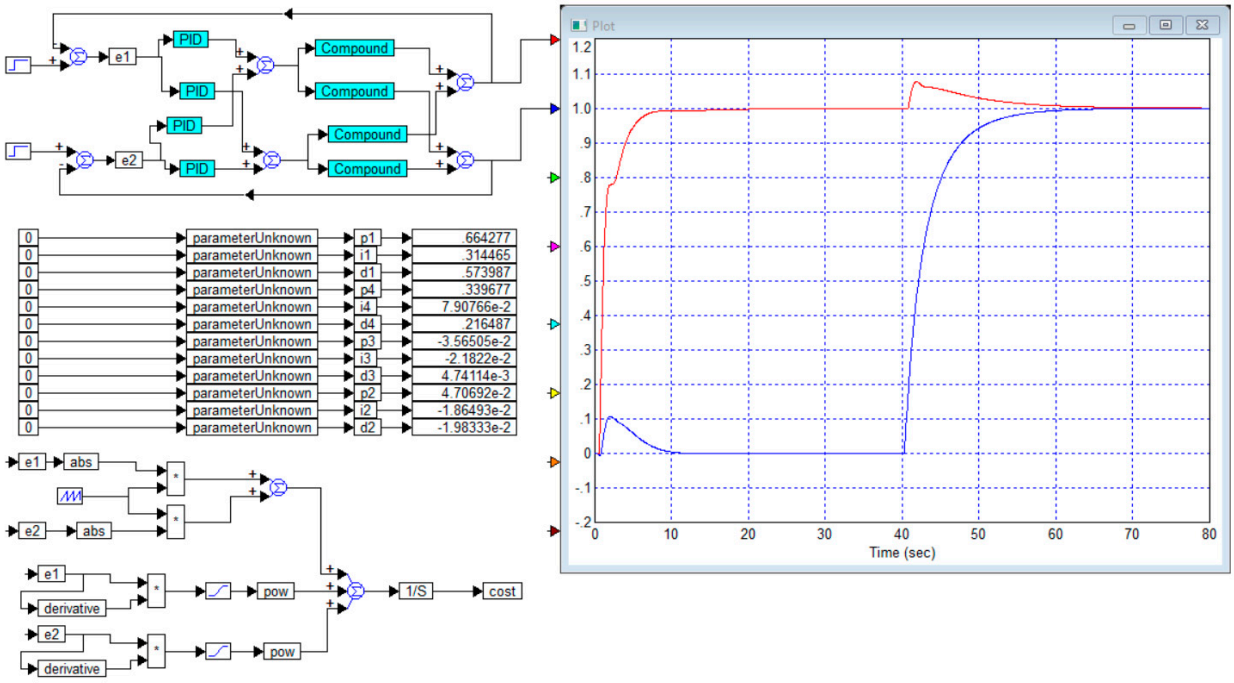

Figure 3. Project and result of optimization of the system with the object according to Example 1: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

Figure 3, as well as the following figures, requires some preliminary comments. All these figures are obtained by screenshots from the VisSim 5.0e software, so they are not edited. The rectangles with an asterisk inside denote signal multiplication devices. The rectangles with the inscription “pow” denote devices for raising the power and, unless otherwise indicated in the text, this is the second power. The rectangles with the inscription “derivate” are calculators of derivatives of the input signal. The rectangles with the symbol “1/S” are integrators, a rectangle with the symbol "*" means a signal multiplier. A rectangle with a step or sawtooth waveform drawn inside is a signal generator of this type. A

rectangle with the inscription “abs” is a calculator of the absolute value of the input signal, a rectangle with the inscription “cost” is the input of the optimization device, where the calculated value function is fed, which allows programming the formula for its calculation.

The remaining rectangles are bus labels. They give the name to the buses or the quantities used. For each identical mark, it is considered that they are all connected to each other, and only one such block can be supplied with a signal. The program does not allow supplying another signal to another such mark, which is a natural limitation. In addition, the signal value of each mark can be used as a gain factor in another place. The blocks that set the gain factor have the form of a pentagon. In order to set the value of the variable gain factor in this way, it is enough to enter the name of the corresponding bus in the field for recording its value. Also, instead of the bus name, an ordinary number can be written in this field. Decimal numbers in this program are shown in a special format: if the number is less than zero, then the zero before the decimal point is not written, and in negative numbers in this case there is a hyphen before the point, denoting a minus. The exponent is written after the number and separated by the letter “e”. Different colors of the lines correspond to the coloring of the input arrows. In particular, the output signal of the first channel is fed to the oscilloscope through the red arrow, so it is displayed as a red line. The output signal of the second channel is fed through the blue arrow and is displayed by a blue line. In all graphs here and below, the abscissa axis shows time in seconds, and the ordinate axis shows the output signal in conventional units.

The output values of each controlled object are physical quantities. These can be voltage, displacement, heating, rotation angle, etc. Each such value is measured by the corresponding sensor and then compared with the task. In modeling, it is not customary to record the physical units of these values, since control theory does not consider the absolute values of the output values but only their increments relative to the initial equilibrium state, and they are most often normalized. That is, they are divided by some standard value, so that only increments in relative units are displayed on the graphs. For linear systems, this is normal, since scaling the signal does not change the character of the graphs. In this article, we do not consider nonlinear systems.

The monitors in the center of Figure 3 show the obtained values of the corresponding coefficients of the matrix PID controller.

All the above-mentioned approaches to optimization were applied here.

The software for modeling and optimization is VisSim in any version (VisSim 5.0e was used). This software is very good for such tasks. Unlike MATLAB Simulink (any version), this software simply does not allow modeling of objects that cannot be implemented in practice. This software takes up little disk space, and the files it creates also take up little space. The main advantage is the precise reproduction of the algorithms by which real digital controllers work.

The test signal, as indicated above, is single step jumps, shifted relative to each other by half the simulation time (15).

The simulation conditions are as follows: time discretization step $\Delta t = 0.01$ *sec* and simulation time $T = 80$ *sec*. The integration method is the simple Euler method.

The objective function (10) with its constituent functions (6)–(9) is used.

The optimization method is Powell’s method. The optimization method can be selected in the corresponding program dialog box.

The method of analyzing the result is a visual assessment of transient processes, an assessment of the roughness of the solution by rounding the obtained coefficients to 1%, and, if necessary, also changing the parameters of the object model by values $\pm 3\%$ or more.

The method to improve the result is to change the weighting coefficients in the cost function. To eliminate overshoot, nonmonotonicity and oscillations, it is necessary to

increase the weighting function for the term that depends on the positive part of the product of the error and its derivative. To reduce the static error, it is necessary to reduce this coefficient or increase the modeling time. If this does not help to achieve the desired result, then the modeling time is also increased by modifying the cost function.

In this result, we do not see an overshoot in the transient process of each channel, but we see the response of each channel to a step of the task in the other channel, and this response is about 10% of the corresponding step.

Modeling and optimization showed that further changes in the cost function, including changes in the weighting coefficients of the terms dependent on the product of the error and its derivative, did not significantly improve the transient process. Any specific improvement in one of the parameters of the transient process is achieved only at the expense of worsening the other parameter. Namely, it is possible to completely suppress this reaction in the second channel by increasing such a reaction in the first channel and reducing the reaction speed of both channels. Or it is possible to increase the reaction speed of the second channel and almost completely suppress this reaction, but at the expense of increasing such a reaction in the first channel to 18%. It is also possible to significantly improve transient processes in the first channel, but at the expense of significantly worsening the reaction of which channel: in it, the jump reaction in the task of the first channel increases to 18%.

It should be noted that the relatively successful solution of the problem is achieved due to the fact that the transfer function has advantageous combinations of elements. This property lies in the following three facts. First, the diagonal elements contain transfer functions in which the transmission coefficients are significantly larger than similar coefficients in other elements of the object's matrix transfer function. Second, these same elements have a smaller delay value. Third, the denominators of these same elements have smaller coefficients in the second- and third-order terms.

Let us show that the indicated dependencies simplify the solution of the problem. To do this, we will reset the elements in the controller matrix that are not on the main diagonal. To do this, it is enough to reset the corresponding coefficients (as shown in Figure 3) and remove the additional "parameterUnknown" blocks from the design. As can be seen from the figure, the control remains successful and even the deterioration of the control quality is not too great.

However, this is so because in this case the simulation time had to be doubled and accordingly the transients doubled. If in Figure 3 the processes when applying a jump to the first input last 10 s and when applying a jump to the second input they last 20 s, then in Figure 4 the processes when applying a jump to the first input last 20 s and when applying a jump to the second channel they last 30 s.

5.2. Two-Channel Object with a Dominant Small Diagonal

Example 2. *Let us consider an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{0.5}{1+4s+2s^2} \cdot e^{-0.7s} & \frac{4}{1+4s+2.5s^2} \cdot e^{-0.3s} \\ \frac{2}{1+2s+s^2} \cdot e^{-0.5s} & \frac{2}{1+3s+6s^2} \cdot e^{-0.8s} \end{bmatrix}. \quad (17)$$

This example is derived from the object of Example 1, in which the first column is replaced by the second, and the second column is replaced by the first. As a result, the transfer function of the object has become maximally unfavorable according to the criterion discussed above. The elements of the main diagonal are characterized by a large delay, smaller coefficients of the transfer function, and large values of the coefficients in the denominator polynomials.

This example is one of the most unfavorable options for objects from this class.

Figure 5 shows the result of an attempt to optimize the diagonal regulator controller for the object of Example 2. The attempt was unsuccessful, the system turned out to be unstable.

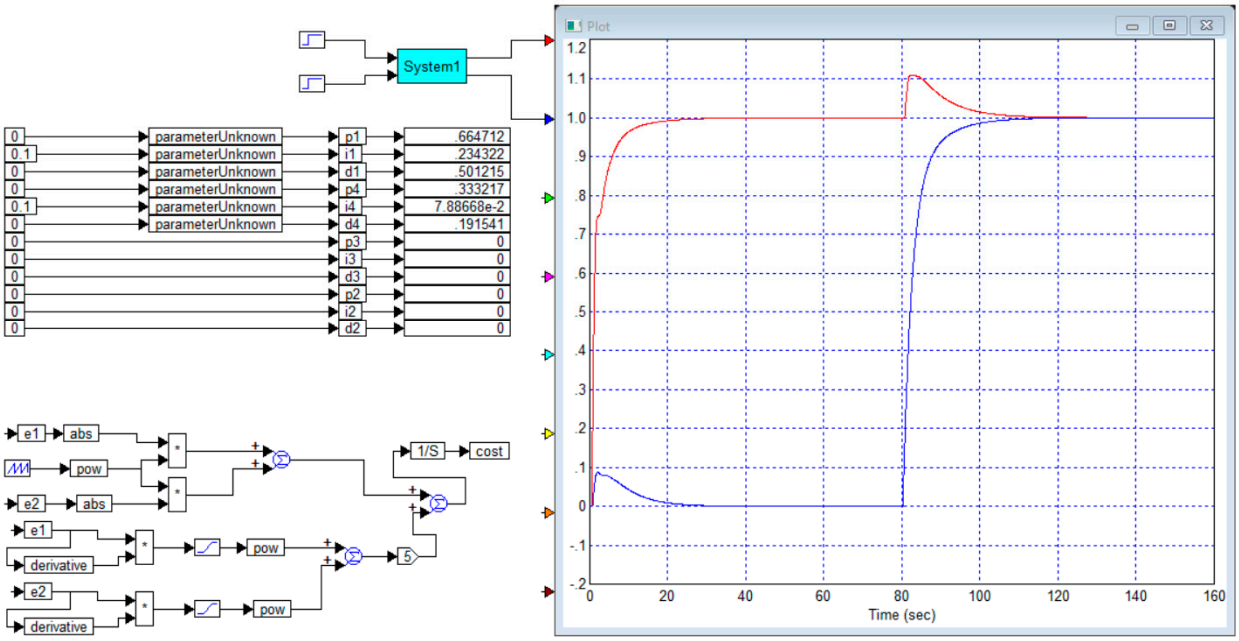

Figure 4. Project for solving Example 1 using a diagonal PID controller, i.e., actually two scalar controllers, as well as the obtained result: controller coefficients and transient process graphs: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

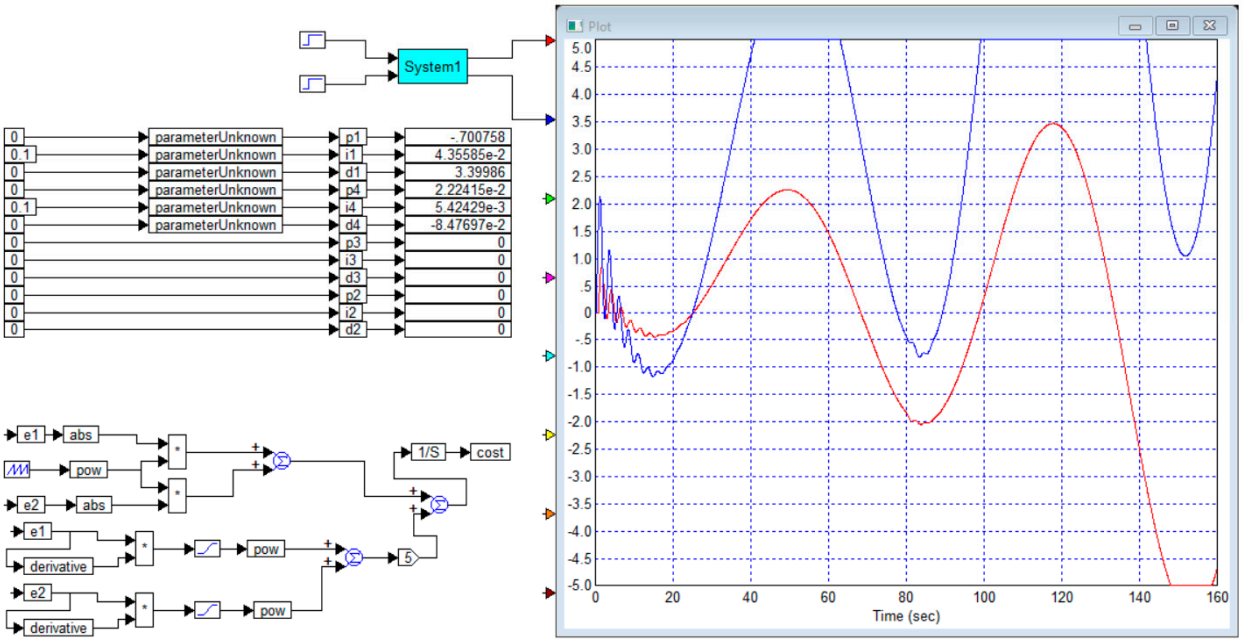

Figure 5. Result of attempt to optimize the diagonal regulator controller for the Example 2 object: the system turned out to be unstable: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

Figure 6 shows the project and the result of an attempt to calculate a full matrix PID controller for the same object from Example 2. A stable system is obtained. However, the

quality of the obtained system is low. The overshoot in the first control channel reaches 98%. The overshoot in the second channel is 12%. The overshoot in the first channel with a jump in the second channel is 50%, the overshoot in the second channel with a jump in the first channel is 98%, and the reverse overshoot is about 50%.

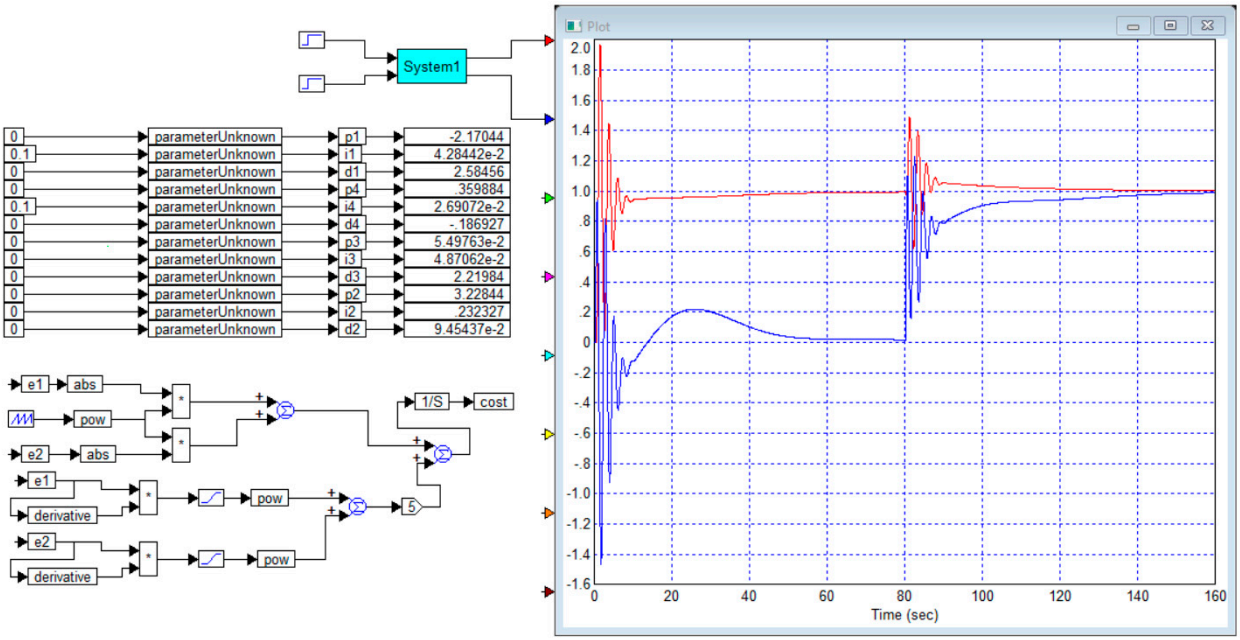

Figure 6. Project and result of an attempt to calculate a full matrix PID controller for the same object from Example 2: a stable system was obtained: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

In this case, the number of oscillations of all overshoots together with the first overshoot is at least four distinct oscillations. In addition, the duration of the process is very long, it is more than 60 s to the level of 5% and long residual relaxation processes of at least 40 s are also observed in the area of a small error. In addition, in the second channel, a second wave of response to a jump in the task of the first channel is observed, the peak of which is long and is between intervals of 20 s and 40 s.

From this example it should be concluded that, with an undesirable combination of all the parameters of the object model, multichannel control can be a rather complicated task. However, this case is clearly bad. In practice, such cases should not arise, since the designer should pay attention to the fact that in the secondary diagonal all elements of the matrix transfer function have a significantly larger value in the entire frequency range for the reasons indicated above. In this case, the designer should change the numbering of the outputs or the numbering of the inputs. To do this, the output of the first channel should feed the second input, and the output of the second channel should feed the first input. This will lead to the matrix transfer function of such an object, which will take the form given in Example 1, and this problem, as we have seen, is solved quite simply and successfully. This example shows that, along with formal methods for designing controllers, a preliminary analysis of the properties of the matrix transfer function of the object should be used to identify the strongest channels of influence. In this case, the first input has the greatest influence on the second output, and the second input has the greatest influence on the first output, which dictates the choice of the numbering rule for the outputs and inputs.

This approach is intuitive and should be used early in the design process, i.e., in advance. For example, we want to move a load from a tower crane in two directions and we have two settings, one for each arm. Let the left setting move the load strongly

vertically but weakly horizontally and the right setting move strongly horizontally but have little effect on the vertical movement. When creating an automated system, we should choose the left setting to control the vertical movement and the right setting to control the horizontal movement. Complete separation will ensure that the left setting will only affect the vertical movement but will not affect the horizontal movement and the right setting will only control the horizontal movement but will not affect the vertical movement. If the designer takes the opposite approach and tries to make the left setting control the horizontal movement and the right setting control the vertical movement, then the problem will still be solvable, but it will be similar to the one considered in Example 2. That is, the least influential inputs are selected to control these quantities, which complicates the controller design and can negatively affect the control quality.

5.3. Two-Channel Object Without an Obvious Dominant Diagonal

Example 3. *Let us consider an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{2}{1+3s+6s^2} \cdot e^{-0.8s} & \frac{4}{1+4s+2.5s^2} \cdot e^{-0.3s} \\ \frac{0.5}{1+4s+2s^2} \cdot e^{-0.7s} & \frac{2}{1+2s+s^2} \cdot e^{-0.5s} \end{bmatrix}. \quad (18)$$

This transfer function is also obtained from the transfer function of Example 1 with appropriate permutations. In this case, in the first row, the most significant element is the element that is not on the main diagonal, and in the second row, the most significant element is on the main diagonal.

Figure 7 shows the structure project and the result of optimization of the full PID controller for such an object from Example 3. This system is stable, but the response of the second channel to the step jump of the first channel is too large. It reaches 75%, and its duration is 30 s. The response of the first channel to the jump in the second channel's task is 14%, its duration to the level of 50% of the maximum value is less than 5 s, but the duration of its noticeable part is about 40 s. In this case, control for each channel is carried out without overshoot; the duration of the processes is from 40 s (for the second channel) to 60 s (for the first channel).

It is also advisable to try to control this object using a simplified diagonal controller. Figure 8 shows the design and result of controlling the object of Example 3 using a diagonal controller.

In this case, the result is noticeably worse, but the resulting system is stable. The response of the second channel to the step jump of the first reaches 98%, and its duration is 50 s. The response of the first channel to the jump in the second channel task is 13%, its duration is about 40 s. In this case, control for each channel is carried out without overshoot; the duration of the processes is 50 s for each channel.

Example 4. *Let us consider an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{2}{1+3s+6s^2} \cdot e^{-0.8s} & \frac{-0.5}{1+4s+2s^2} \cdot e^{-0.7s} \\ \frac{-4}{1+4s+2.5s^2} \cdot e^{-0.3s} & \frac{2}{1+2s+s^2} \cdot e^{-0.5s} \end{bmatrix} \quad (19)$$

This transfer function is derived from the transfer function of Example 3, with the difference that the elements in the minor diagonal are negative. Figure 9 shows the design and optimization result of the diagonal controller for this case.

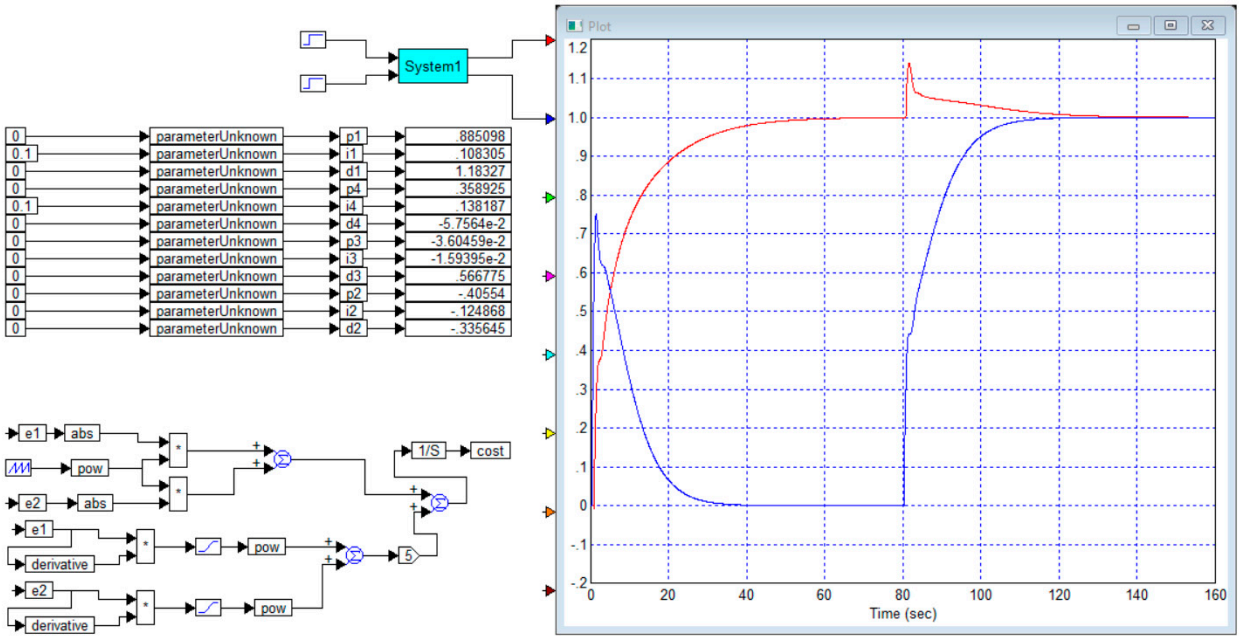

Figure 7. Result of designing and optimizing the full PID controller for the object from Example 3: the system is stable, but the response of the second channel to a jump in the setpoint the first channel is too big: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

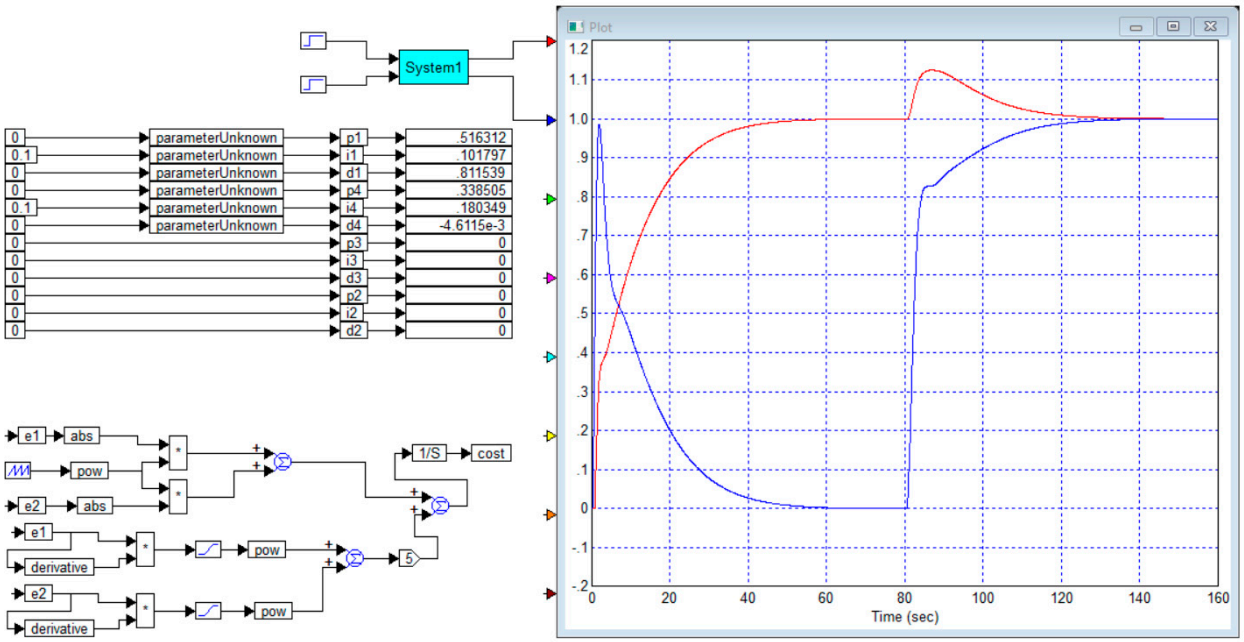

Figure 8. Project and result of controlling the object of Example 3 using a diagonal regulator: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

In terms of process quality indicators, there is no significant difference from the result presented in Figure 8. Cross-reactions and emissions are comparable in size and duration.

Figure 10 shows the result of the design and optimization of the complete controller for the object of Example 4. Compared to the result shown in Figure 7, it has a slight advantage, namely: in Figure 6, the response of the output of the second channel to a jump in the setpoint at the input of the first channel is 75%, and in Figure 10 the corresponding

response is 61%. In terms of duration, these responses are comparable, both being about 50 s. The remaining indicators are also comparable.

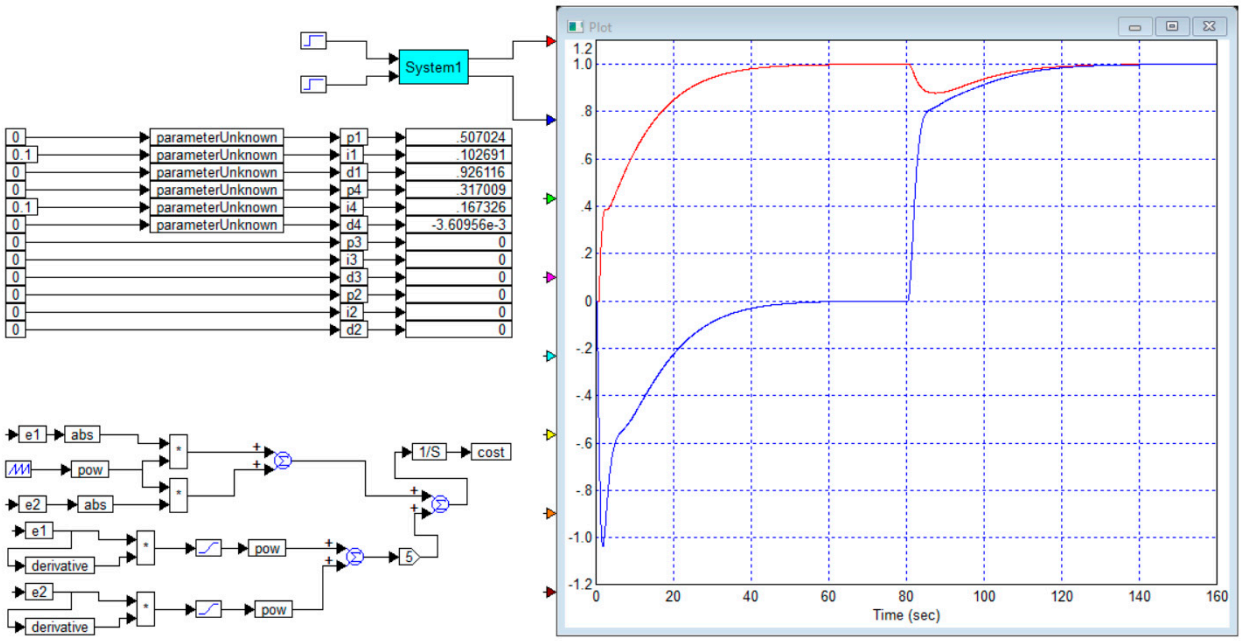

Figure 9. Result of design and optimization of the diagonal controller for Example 4: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

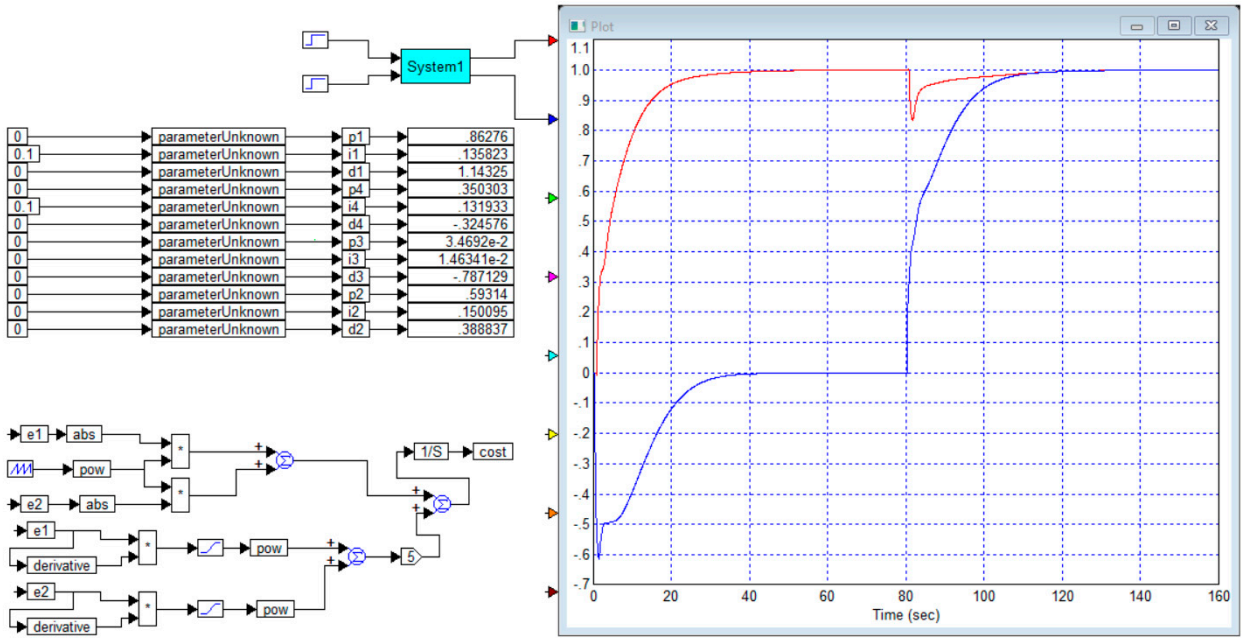

Figure 10. Result of designing and optimizing a complete controller for the object from Example 4: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

5.4. Three-Channel Object with Slightly Dominant Main Diagonal

Note that in the literature we have not found reliable reports on solving control problems for three-channel objects that would provide mathematical models of the object with numerical values of the parameters and the result of the control. In this case, we could

take these initial models and calculate regulators for them and compare the result with the result that would be presented in a similar article. A similar problem is solved in this article for the first time.

Example 5. *Let us consider an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{2}{1+2s+s^2} \cdot e^{-0.5s} & \frac{1.5}{1+4s+2s^2} \cdot e^{-0.6s} & \frac{1.2}{1+5s+2s^2} \cdot e^{-0.7s} \\ \frac{1.5}{1+3s+6s^2} \cdot e^{-0.6s} & \frac{1.8}{1+4s+2s^2} \cdot e^{-0.4s} & \frac{1.4}{1+2s+3s^2} \cdot e^{-0.6s} \\ \frac{1.2}{1+5s+2s^2} \cdot e^{-0.7s} & \frac{1.4}{1+3s+2s^2} \cdot e^{-0.6s} & \frac{1.7}{1+3s+s^2} \cdot e^{-0.5s} \end{bmatrix}. \quad (20)$$

As a test task, we will also use a vector consisting of time-shifted single jumps, but for a visual result, we will feed a jump with a negative value to the second channel:

$$V(t) = \begin{bmatrix} \sigma(2\Delta t) \\ -\sigma(T/3) \\ \sigma(2T/3) \end{bmatrix}. \quad (21)$$

Figure 11 shows the design and numerical optimization result of the controller for this example. The cost function calculator for this case is shown in Figure 12. The parasitic response of the second channel reaches 45% with jumps in the first channel. The parasitic response in the first channel is 20% with jumps in the second channel. The parasitic response in the third channel with a jump in the second channel is about 36%, with jumps in the first channel of 40%. The overshoot in each channel is negligible.

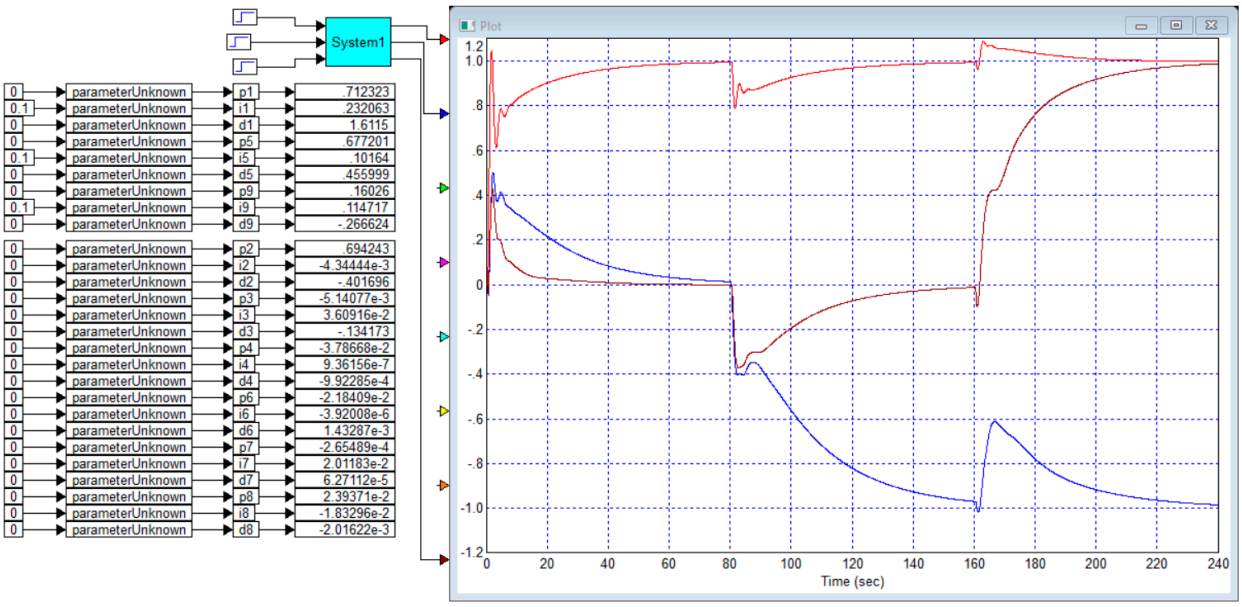

Figure 11. Design and result of numerical optimization of the controller for Example 5: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

For comparison, Figure 13 shows the design and optimization result of the diagonal regulator. It can be noted that the response in the first channel with a jump in the second channel increased by one and a half times, but the response in the same channel with a jump in the third channel decreased significantly. Also, the parasitic response in the third channel with a jump in the first channel decreased by two times. Thus, there is no reason to claim that in this case the diagonal regulator demonstrates a worse result than the full regulator.

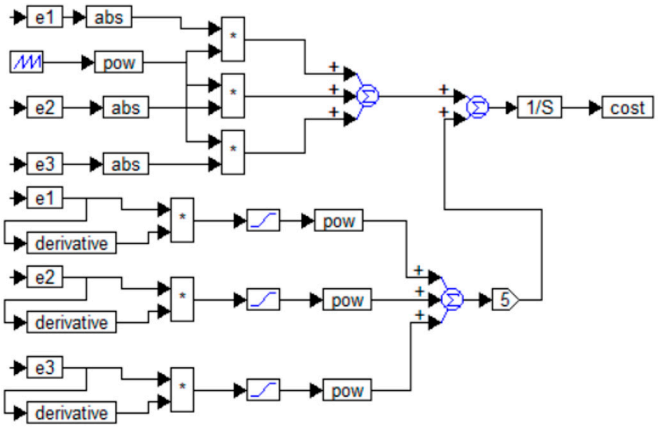

Figure 12. Cost function for the case of Example 5 according to the project in Figure 11.

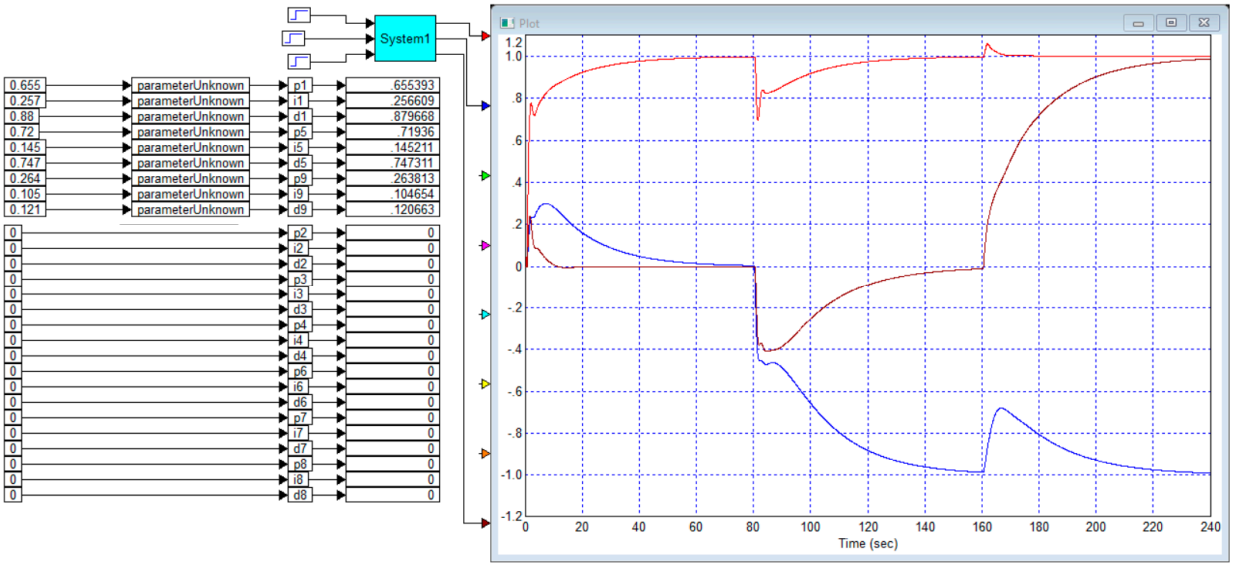

Figure 13. Result when decreasing the weighting coefficient of the second member, fourth root: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

Figure 14 shows the project and the optimization result for the case where the weight coefficient of the second term in the cost function is reduced to an insignificant value $k_w = 0.1$. In this case, the performance of each channel has increased more than twofold, parasitic responses have also increased by 1.5–2 times, and emissions of up to 10% have appeared in the first and third channels.

It is also possible to use the square root in the second term and the weighting factor, in this case $k_w = 0.1$. The result for this case with a full controller is shown in Figure 15, and Figure 16 shows the result for this case with a diagonal controller. In terms of transient response quality, these two results differ only slightly from each other, and at the same time, they both differ quite significantly from the results obtained with the other cost function parameters shown in the previous figures.

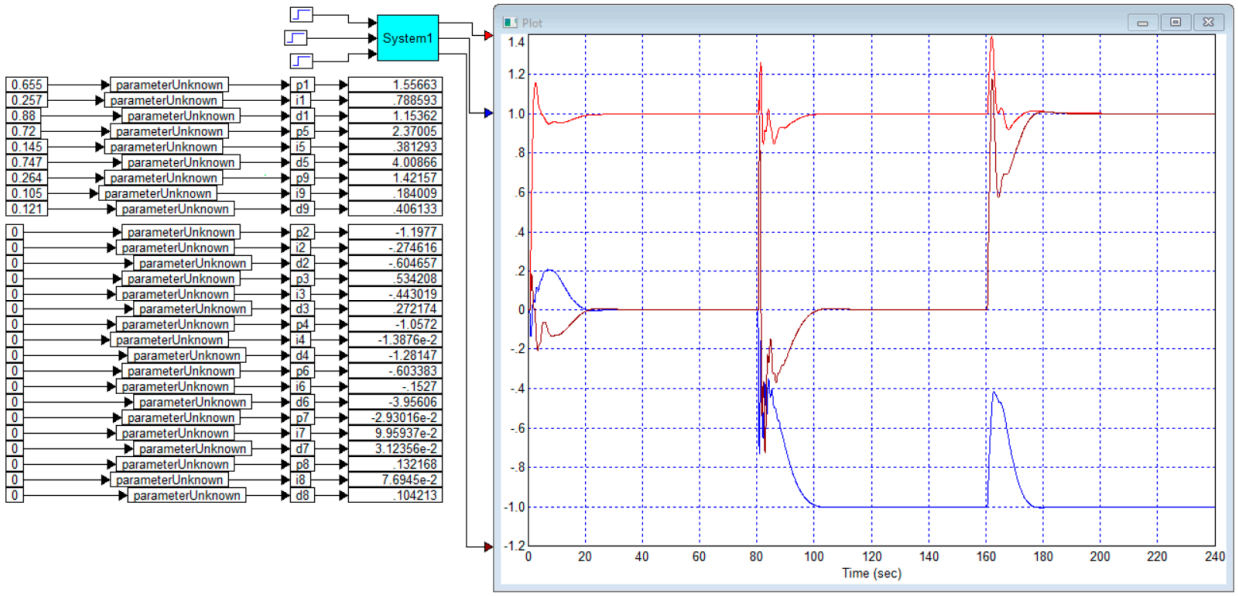

Figure 14. Result when using a small weighting factor in the second term: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

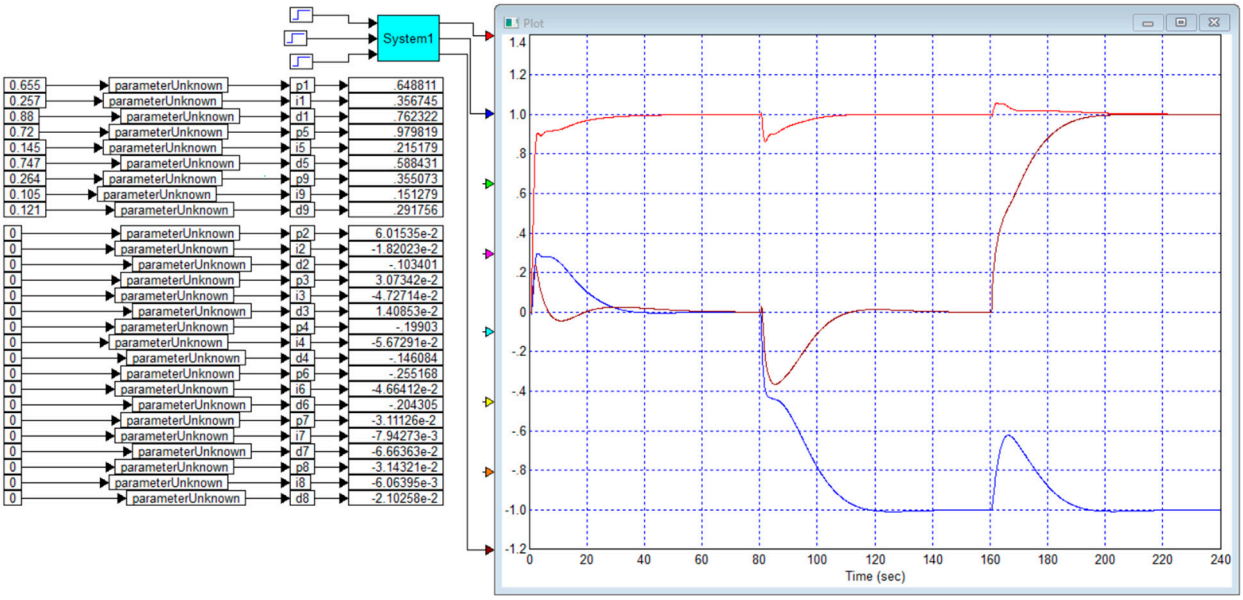

Figure 15. Result when using the square root in the second term of objective function: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

This example shows that, in the case where the matrix transfer function of the control object is characterized by a slight predominance of diagonal elements in absolute value over the remaining elements, the result (expressed as transients for each channel separately) depends to a much greater extent on the parameters of the objective function during optimization than on whether a diagonal regulator or a full regulator is used. The use of nonzero elements in the matrix PID regulator does not allow a radical improvement in the quality of the transient process, which is determined mainly by a trade-off between speed and overshoot for all channels.

To obtain more complete information and substantiate the conclusions, we will calculate the regulator for a slightly modified object.

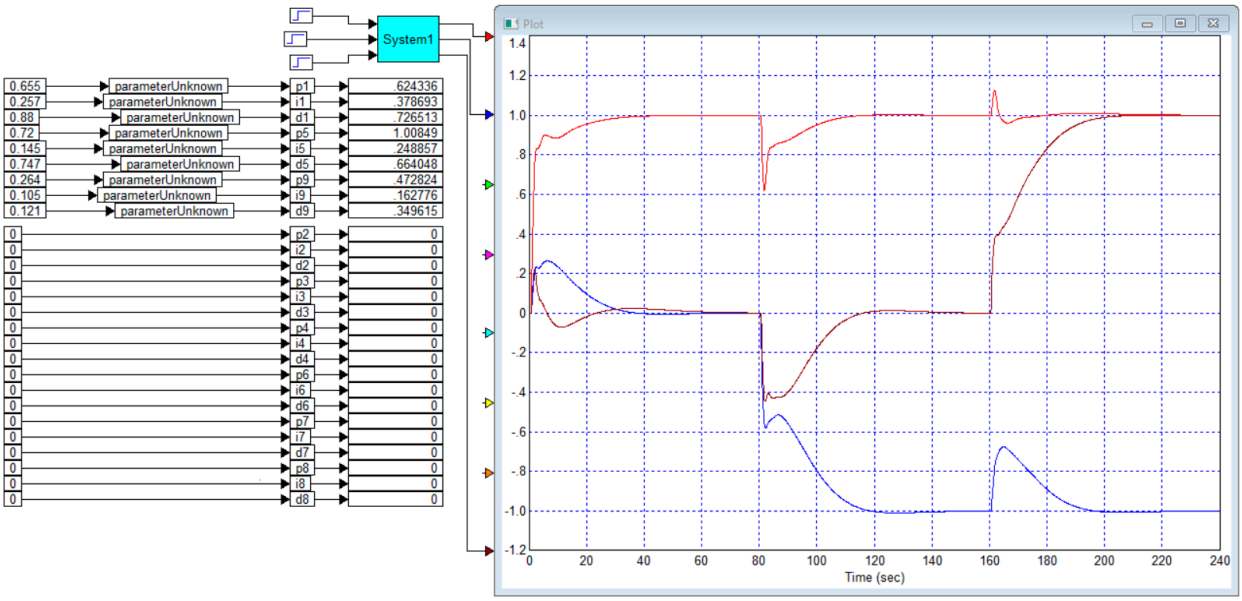

Figure 16. Slightly worse result (more peaks in the first channel) when using the diagonal control: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

5.5. Three-Channel Object with a Clearly Dominant Main Diagonal

Example 6. *We simply increase the coefficients at the main diagonal elements of the matrix transfer function of the object by approximately one and a half times. We obtain an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{3}{1+2s+s^2}\cdot e^{-0.5s} & \frac{1.5}{1+4s+2s^2}\cdot e^{-0.6s} & \frac{1.2}{1+5s+2s^2}\cdot e^{-0.7s} \\ \frac{1.5}{1+3s+6s^2}\cdot e^{-0.6s} & \frac{2.7}{1+4s+2s^2}\cdot e^{-0.4s} & \frac{1.4}{1+2s+3s^2}\cdot e^{-0.6s} \\ \frac{1.2}{1+5s+2s^2}\cdot e^{-0.7s} & \frac{1.4}{1+3s+2s^2}\cdot e^{-0.6s} & \frac{2.55}{1+3s+s^2}\cdot e^{-0.5s} \end{bmatrix}. \quad (22)$$

The cost function and test signal in this case are the same as in Example 5.

Figure 17 shows the design and optimization result of the diagonal controller for Example 6. The quality of the resulting transient process in all three channels has improved significantly in this case. The parasitic response in the first channel varies from 10% to 13%, in the second channel it remains within 20%, in the third channel it varies from 18% to 30%. There is no overshoot in any channel.

Figure 18 shows the design and optimization result of the full controller for Example 6. The result is slightly better than in Figure 17. All parasitic responses have become less than 10%, and the responses of the first and second channels to the task jump in the third channel are even less than 5%, and there is no overshoot in each channel either.

Based on this example, it can be argued that in the matrix transfer function of the control object there is a significant (two or more times) predominance of the values of the diagonal elements over the other elements. With a diagonal controller, acceptable control quality is achieved, and with a full controller this quality can be significantly (twice) improved.

This dominance can be called a majorizing property. It is expressed in the fact that each diagonal element is significantly larger in absolute value than the other elements in the same row of a given matrix.

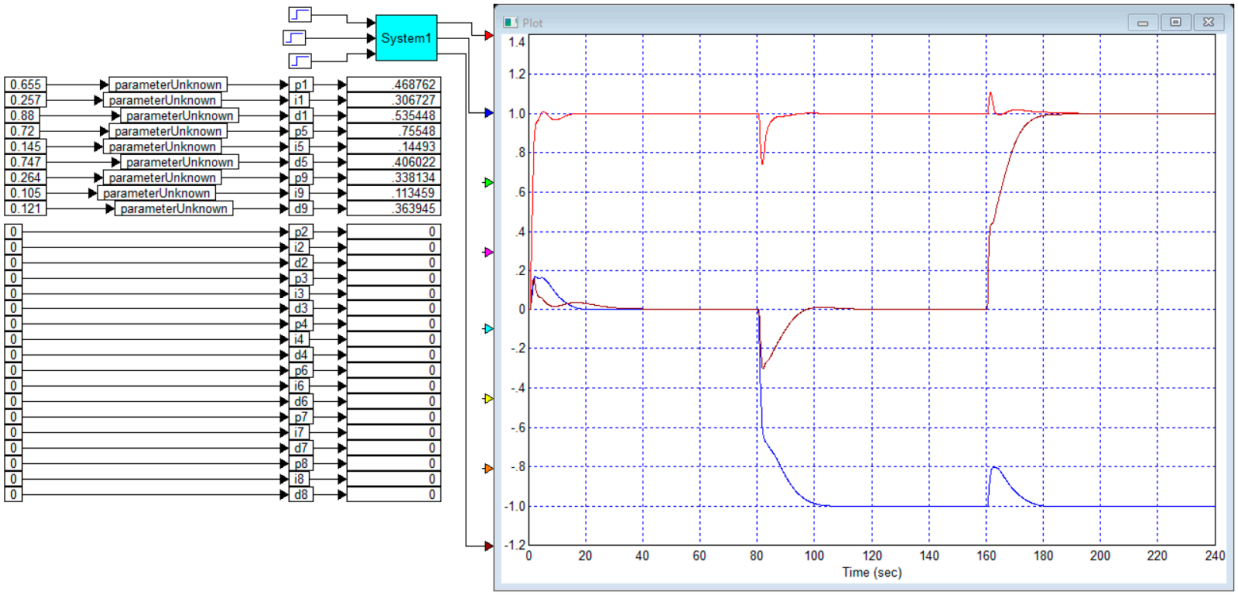

Figure 17. The result of designing and optimizing the diagonal controller for Example 6, the cost function is as in Figure 12: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

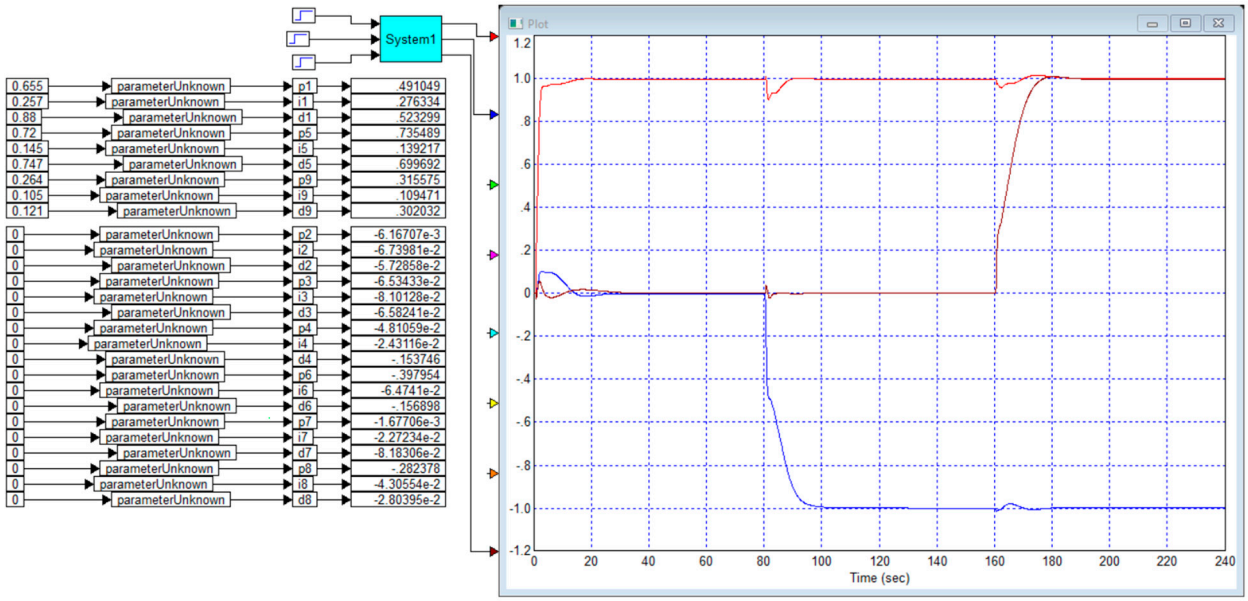

Figure 18. Design and optimization result of the full controller for Example 6, cost function as in Figure 12. The result is slightly better than in Figure 17: smaller responses in the first and in each channel in response to the task on the other channels: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

5.6. Three-Channel Object Without Dominant Main Diagonal

To make sure that dominance is important only in each row, consider another example based on the transfer function from Example 6, namely, we decrease the value of all coefficients in the first row by two times and increase the value of all coefficients in the second row by two times.

Example 7. *We obtain an object whose transfer function has the following form:*

$$W(s) = \begin{bmatrix} \frac{1.5}{1+2s+s^2} \cdot e^{-0.5s} & \frac{0.75}{1+4s+2s^2} \cdot e^{-0.6s} & \frac{0.6}{1+5s+2s^2} \cdot e^{-0.7s} \\ \frac{3}{1+3s+6s^2} \cdot e^{-0.6s} & \frac{5.4}{1+4s+2s^2} \cdot e^{-0.4s} & \frac{2.8}{1+2s+3s^2} \cdot e^{-0.6s} \\ \frac{1.2}{1+5s+2s^2} \cdot e^{-0.7s} & \frac{1.4}{1+3s+2s^2} \cdot e^{-0.6s} & \frac{2.55}{1+3s+s^2} \cdot e^{-0.5s} \end{bmatrix}. \quad (23)$$

Figure 19 shows the design for optimization and the result for this case. The result is slightly worse in quality than the result shown in Figure 17. This is probably due to the change in the relative influence of each channel on the other channels, as well as the effect of the result on the cost function.

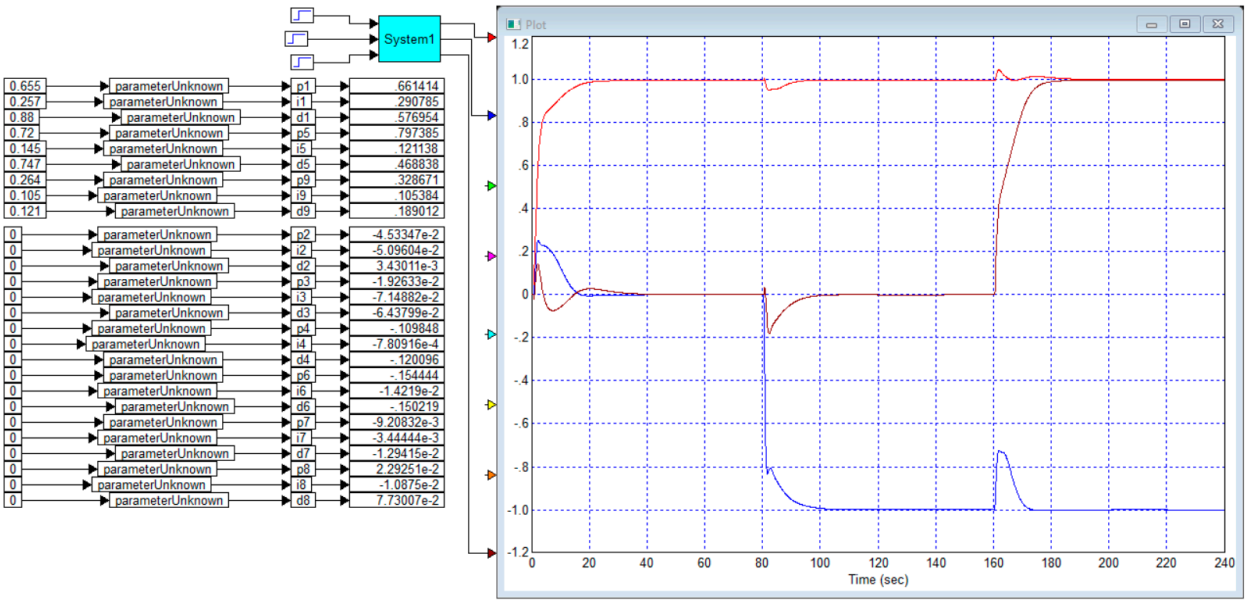

Figure 19. Controller optimization project for the Example 7 object and the result for this case: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

If the designer is not satisfied with the quality of the transient process in a given channel, then an additional weighting factor can be introduced into the corresponding term of the cost function, depending on the error in that channel. For example, in this case, the quality of the response of the second channel is noticeably worse than the quality of the response of the other channels. The best response is in the first channel. The designer can introduce a factor of 3 in the second channel, a factor of 2 in the third channel, and leave a factor of 1 in the first channel.

Figure 20 illustrates the result. As we can see, we were able to reduce the parasitic responses in the third channel by increasing the parasitic responses in the first channel. This further demonstrates the dominant influence of the cost function during the optimization and its numerical coefficients on the result.

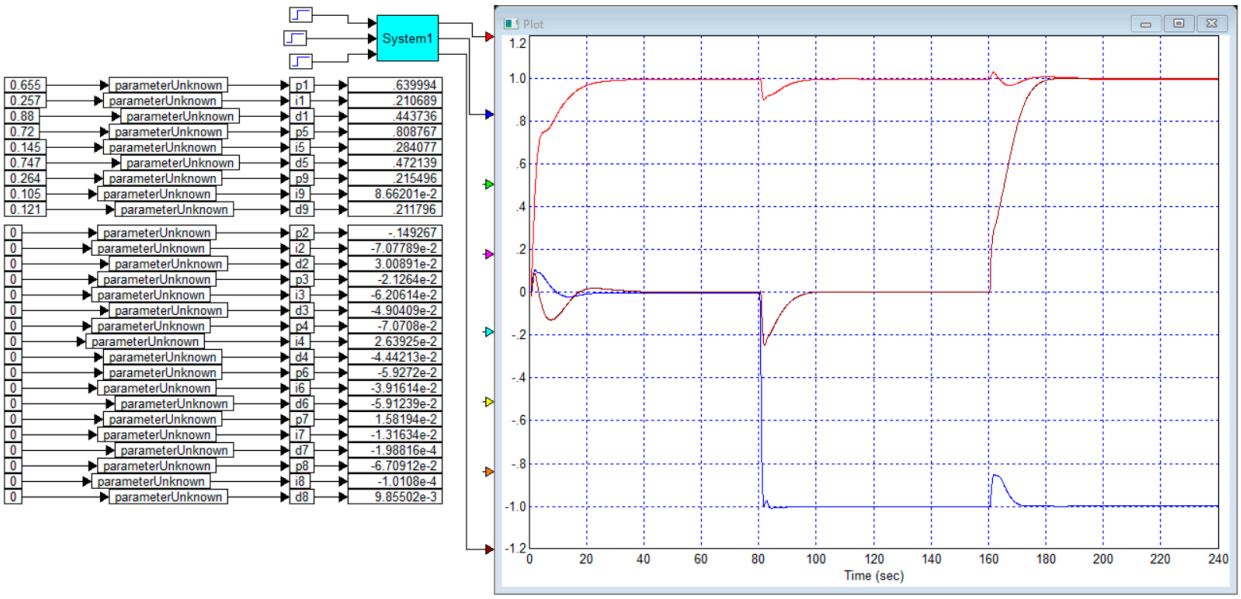

Figure 20. Controller optimization project for the Example 7 object using individual weighting factors and the result for this case: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

6. Method Modification: Ensuring the Requirements of Completeness of the Influence of Test Tasks

If we analyze the set of test signals used in the optimization, we can see that they are somewhat incomplete. Note that, with a step jump on one of the inputs, there is no jump on the other input. The optimization procedure provides better transients in these situations. However, the possibility of a simultaneous jump of the reference on both inputs, which can act together in such a way that an unacceptably large value is exceeded, is not considered. Such a situation can occur both when the signs of the reference jumps coincide and when they have opposite signs. To ensure completeness of all possible signal combinations, it is proposed to include in the set of test signals both the simultaneous application of in-phase jumps to all inputs and the simultaneous application of antiphase jumps to all inputs. For a 2×2 object, this gives two more jumps, and for a 3×3 object, this implies many more jumps in various combinations: in-phase and antiphase jumps on every two inputs in pairs, for a total of six jumps, for a total of nine jumps. The designer could also try adding in-phase jumps on all three inputs, and in-phase jumps on two inputs with an out-of-phase jump on the third input, for a total of four jumps. This gives a total of 13 different options, which seems overly complicated, so we recommend limiting it to nine jumps. We will demonstrate this method using the 2×2 object from Examples 1 and 2.

Example 8. *Let us consider the object from Example 1 and apply the following set of test tasks to it:*

$$V(t) = \begin{bmatrix} \sigma(2\Delta t) - \sigma\left(\frac{T}{2}\right) + \sigma(3T/4) \\ \sigma(T/4) - \sigma\left(\frac{T}{2}\right) - \sigma(3T/4) \end{bmatrix}. \tag{24}$$

Figure 21 shows the design and optimization result for this case using the simplest cost function consisting of the integral of the sum of the squares of the product of the error modulus and the time since the start of the transient process.

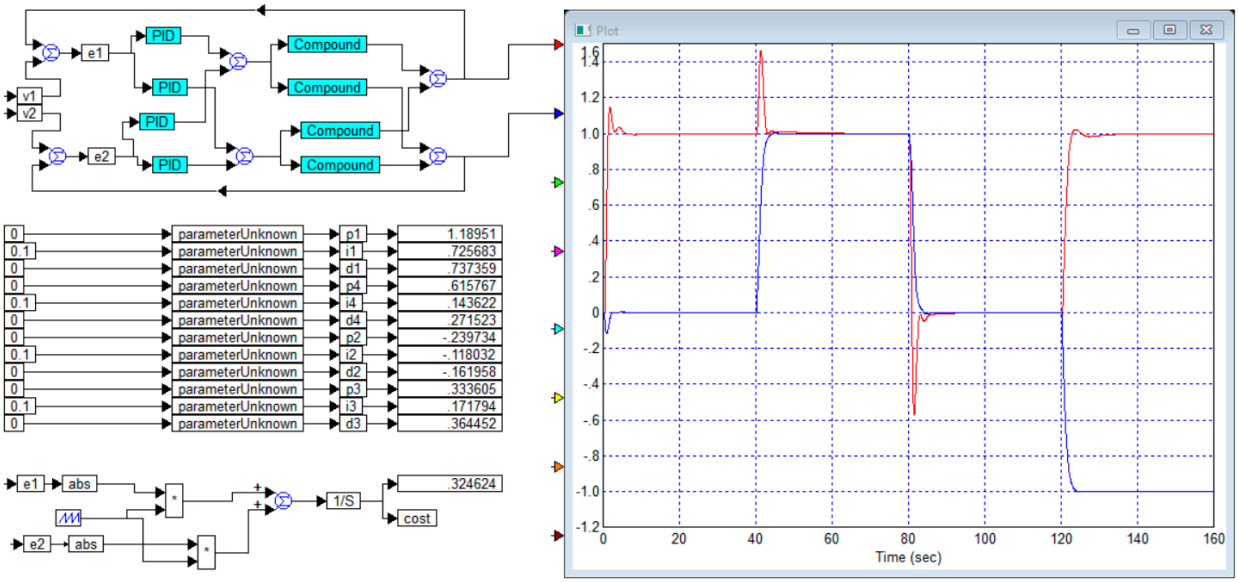

Figure 21. Design and optimization result using three task transitions in each channel and the simplest cost function according to Example 8: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

At the same time, there are two large parasitic jumps in the second channel, namely, when there is a jump only in the first channel and when there is a jump unidirectionally in both channels simultaneously. However, other components of the complex transient process have improved. Importantly, in this case there is no overshoot in the second channel, and the parasitic jump during jumps in the first channel is also small, only 10%. For comparison, we present the result using only one jump on each channel. This result is shown in Figure 22. In this case, a similar jump in the second channel is 25%. From this we can conclude that considering the set of all four types of jumps is more representative and, in this case, formally, the same cost function works differently.

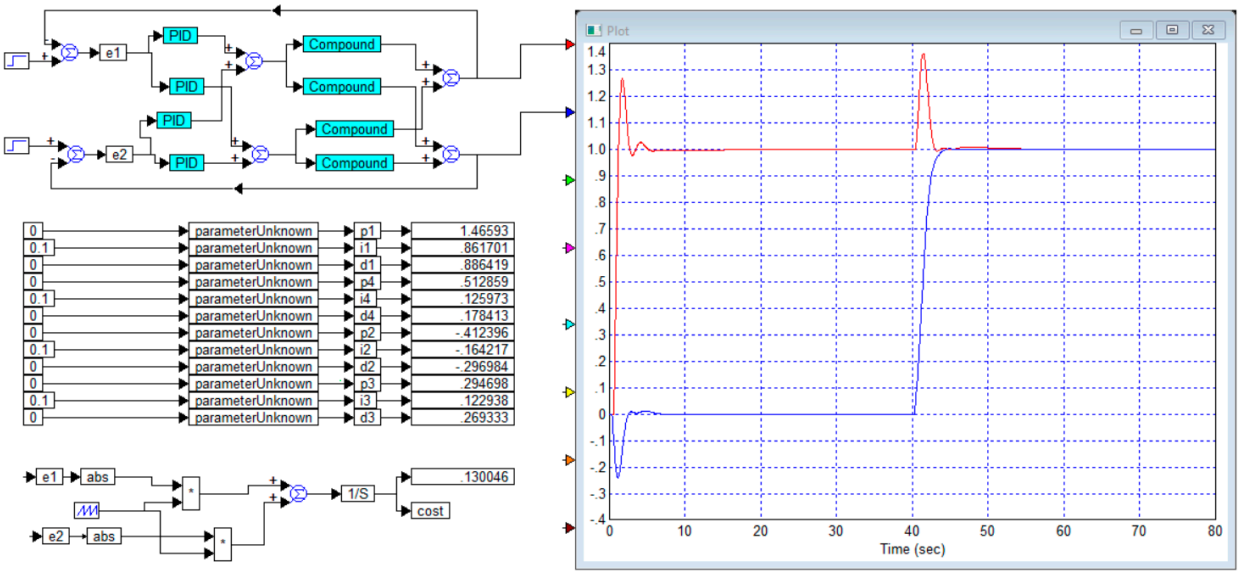

Figure 22. Design and optimization result using one transition to the task in each channel and the simplest cost function according to Example 8: the red curve is the output of the first channel; the blue curve is the output of the second channel, colored triangles are unused oscilloscope inputs.

It is noteworthy that for $t = 80$ *sec* in Figure 21 the overshoot in the first channel is 60%, from which it follows that the simultaneous synchronous jump in the task on both channels is a strong exciting factor for the first channel. For $t = 120$ *sec* the response at the outputs of both channels is insignificant, which indicates that the antiphase jumps of the steps are not an exciting factor for both channels.

Finally, it is important to use both methods to improve the transient process simultaneously, namely, the test signal is as in Example 8, and the cost function is complete, as shown in Figure 9.

Example 9. *Figure 23 shows the design and optimization result using four jumps in each channel and a complex cost function that helps reduce overshoot. The result can be called ideal: there is no overshoot in all cases for all channels, the parasitic response in both channels is negligible. Thus, the full accounting of all variations of the system input signals significantly increases the efficiency of the optimization design of the controller. It is enough to compare this result with the result shown for the same case in Figures 3 and 4—there the parasitic jumps in each channel were 10% with a duration of about 10 s.*

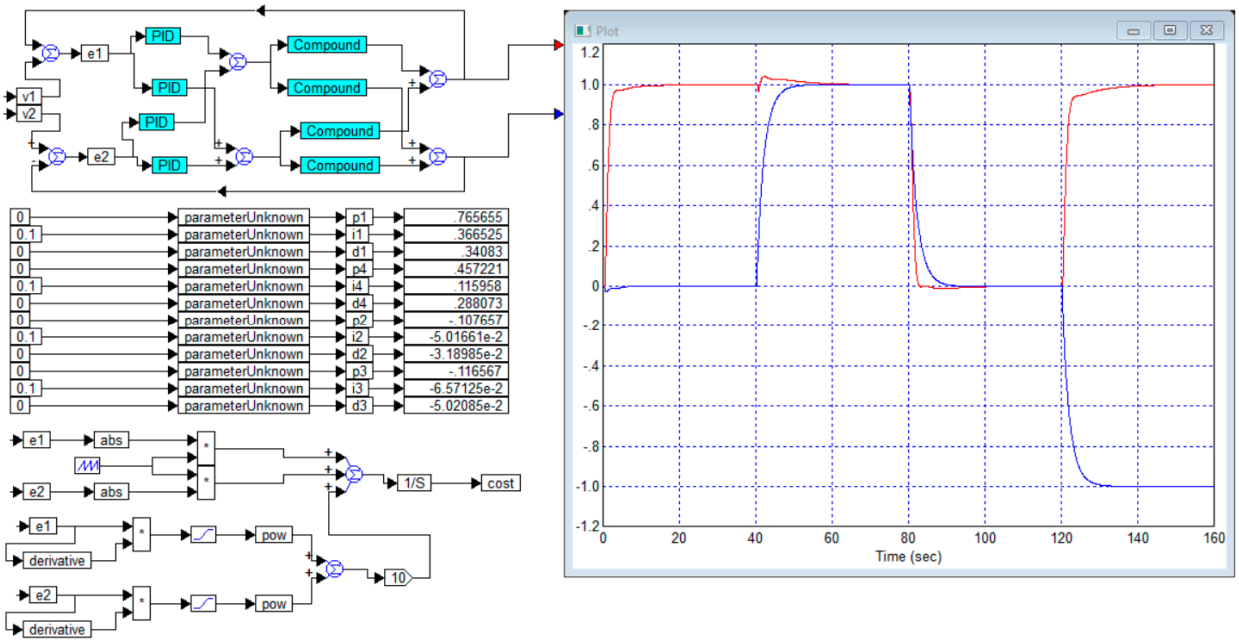

Figure 23. Design and optimization result using four hops in each channel and a complex cost function that helps reduce overshoot: the red curve is the output of the first channel; the blue curve is the output of the second channel.

For this reason, it is worth considering how the optimization of the controller for the object obtained from Example 1 would be implemented. In this case, this is carried out by replacing the first output with the second and the second with the first. The case considered in Example 2 (the result is shown in Figure 5) is extremely unsatisfactory: there is a large overshoot and a lot of flickers in each channel.

Example 10. *Consider the object of Example 2 with test signals as in Example 8.*

Figure 24 shows the design for this case and the result of the numerical optimization. The result exceeded all expectations. Despite the fact that in this case the elements of the main diagonal in the entire frequency range are smaller in absolute value than the elements of the nonmain diagonal, the result of the optimization shows excellent quality of the

transients: there are no peaks, and the parasitic responses in each channel with jumps in the other channel are also negligibly small in value, no more than 1–2%.

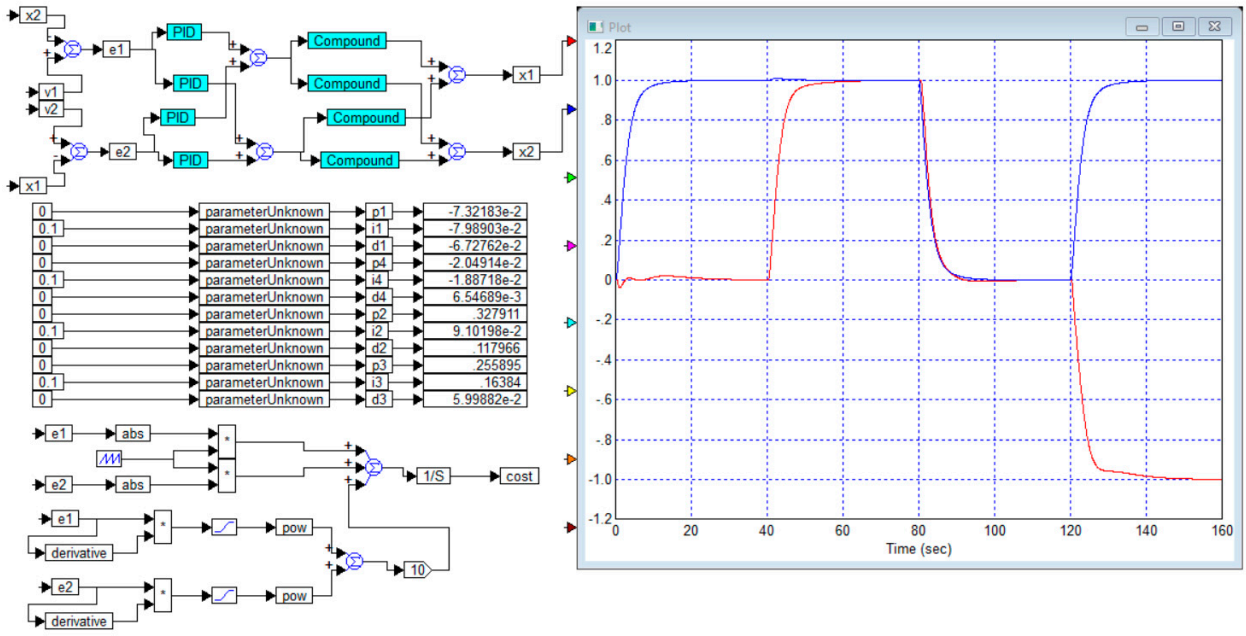

Figure 24. System optimization project for the object from Example 9 (and from Example 2) using the test task from Example 8: the red curve is the output of the first channel; the blue curve is the output of the second channel.

This example particularly clearly shows that, when presenting requirements for completeness in the quality of transient processes (including the requirement that not only transient processes for each channel separately in the absence of jumps in the task for other channels have a sufficiently high quality but also of processes with simultaneous in-phase and simultaneous antiphase jumps at all inputs of the system), transient processes would also be characterized by high quality: little overshoot or its complete absence.

To evaluate the efficiency of the proposed method, it is sufficient to compare the graphs shown in Figure 24 with the graphs shown in Figure 3. In the first case, we solved the problem using the method that we previously published in articles [37,38]. In this case, there was no overshoot in the system for both channels, but the duration of the process in the first channel was 10 s, and the duration of the process in the second channel was 20 s. In addition, parasitic responses to a jump in the other channel arose in each channel. Such a response in the first channel was 8% with a duration of 20 s, and in the second channel it was more than 10% with a duration of 10 s. In the system that is optimized using the proposed method, the transient processes shown in Figure 24 are distinguished by better quality. The duration of the transient process of the first channel is 10 s, the duration of the process of the second channel is 20 s, and there is no overshoot in both channels, but the response of each channel to a jump in the other channel is also practically absent. In the second channel, it is less than 1% and in the first channel less than 0.5%.

7. Discussion

As a result of the modeling and numerical optimization of two-channel complete and incomplete PID controllers, the following conclusions were made:

Particular attention should be paid to the ratio of the values of the elements of the main diagonal of the matrix of the transfer function of the object to the remaining elements of this transfer function. Signs of a larger value are: (a) a larger value of the gain coefficient;

(b) a smaller value of the delay; (c) a smaller value of the time constant of the polynomial in the denominator. If these requirements are not met at least partially, it is advisable to check whether they will be met to a greater extent by replacing the numbering of the outputs with another correspondence of the outputs and inputs.

When optimizing, it is recommended to use jumps with a time shift of the corresponding fraction of the simulation time as a test task. Since during optimization it is necessary to change the tasks at all n inputs in an arbitrary sequence so that none of the inputs remains without such a change, the minimum number of such jumps is equal to the number of channels in the object. It is recommended to supply these jumps at equal time intervals. In this case, a sawtooth function should be used as a multiplier to the error modulus, which linearly increases in the intervals between jumps but is reset to zero at each jump. For the first time, it was proposed to use not only as many jumps as there are channels but also jumps in each channel in combination with simultaneous jumps in other channels, both in-phase and out-of-phase. This gives a doubling of the number of jumps for a two-channel system and a tripling of the number of jumps for a three-channel system.

It is shown that the use of such a test signal, while observing all other optimization conditions, allows for a significant improvement in the quality of transients, which in the two examples we have considered can be called ideal. Such a test signal with its experimental justification is proposed for the first time.

It is also necessary to form a sawtooth function that begins with each new jump and then increases linearly at the same rate. This function $f_{ST}(t) = SawTooth(t)$ symbolizes the time from the beginning of each new jump.

It is shown that the very first jump should be shifted in time by an amount no less than twice the duration of the time quantization step.

The approach to solving the optimization problem proposed in the article recommends using the complex cost function (10) as the objective function. This cost function is an integral of the sum of several terms. The main n terms are the product of the error module of each signal and the function $f_{ST}(t)$, i.e., the time since the start of the change in the task at any of the inputs. In addition to these main n terms, n terms are introduced into the cost function, calculated as the positive part of the product of the error in each channel and its derivative. It is recommended to apply a weighting coefficient $k_w = 10 \div 500$ to these additional terms. The selection of the weighting coefficient is carried out based on the required suppression of oscillations and overshoot. Increasing this coefficient reduces and even completely suppresses overshoot but can lead to an increase in the duration of the transient process. For the first time, it is proposed to use the square root for each of these additional terms, which makes the cost function more natural and eliminates the quadratic dependence of these terms on the amplitude of the test signals.

Also, for the first time, it is proposed to use the second or higher power for the $f_{ST}(t)$ factor, which contributes to a faster decay of the control error. It is recommended to use the second or higher power if the decay rate of this error does not satisfy the designer.

Also, if optimization does not ensure the completion of transient processes, it is recommended to try to increase the simulation time.

The simulation confirmed that, if the matrix transfer function of the object is characterized by the prevailing absolute value of the elements of the main diagonal, then even the diagonal controller allows for quite acceptable channel autonomy, i.e., separate control of the output values of each channel of the controlled object. In this case, a complete controller can give a better result. However, this improvement is qualitative, not fundamental. The properties of the matrix transfer function can be unfavorable, which is expressed in the fact that the elements of its main diagonal are commensurate in magnitude with the remaining elements in the entire frequency range. Then the quality of the result of solving the opti-

mization problem will be significantly worse, since control using a complete controller can stabilize. But this can lead to large overshoots, large output characteristics of the channels with a step change of the task in other channels, as well as many oscillations in both types of these transients. In some cases, an incomplete controller can provide stable control.

It is obvious that, with a further increase in the number of channels, the design of a complete controller becomes not only more complicated but also less and less expedient. But in the case of ensuring a given property of the matrix transfer function of the object, a diagonal controller will be sufficient. In this case, the complexity of the problem of designing a diagonal controller will not increase so sharply with an increase in the number of channels. Then the problem is in the design of scalar controllers, the number of which corresponds to the number of channels in the object. Perhaps for this reason there are practically no publications describing the results of the design of controllers for multichannel objects with more than three channels, and the number of articles devoted to the design of three-channel controllers is negligibly small.

8. Conclusions

A methodology for designing sequential PID controllers is proposed, which, if necessary, can be applied to more complex controllers, including those with additional channels for double integration and/or additional differentiation. The methodology consists of choosing: a software optimization tool; an optimization method (Powell's method is recommended); an integration method (the simple Euler method is recommended); an objective function (recommended according to Relation (10) of two terms); a set of test signals in the form of single jumps, shifted by equal time intervals, and their superposition. This superposition implies: (a) individual jumps at each input in the absence of changes at other inputs; (b) simultaneous common-phase overvoltage; (c) simultaneous antiphase jumps. The objective function is based on the integral of the error modulus, multiplied by the sawtooth shape of the signal and with the addition of the integral of the square root of the positive part of the product of the error and its derivative. A method is proposed to increase the influence of one of the two terms in Function (10) by selecting the best weighting function based on an analysis of the optimization result. In case of failure, it is recommended to increase the simulation time.

Of course, each method has not only advantages but also its limitations. The limitations of each method are revealed when solving specific problems. The numerical optimization method has fewer limitations than other methods for solving problems of this class. Unlike tabular and analytical methods, the numerical optimization method not only easily considers delays and even nonlinear elements in the object model but is also not much more complicated in these specific cases. The only difference in the presence of nonlinearity is that in this case the use of single step jumps as a task is not justified. In the presence of nonlinearity, the test signals must coincide with the expected signals in the real system. The advantage of the numerical optimization method over engineering methods is its lower labor intensity due to the transfer of many calculations to the computer. The method will not lead to the expected results if they are fundamentally unattainable for an object with a given mathematical model and set requirements for quality indicators.

Unlike the Monte Carlo method (which is used in the MATLAB Simulink application), the numerical optimization method does not find a solution that fits within the given bounds. It offers the best solution in terms of the given cost function. If the boundaries in MATLAB Simulink are set too wide, the solution will be significantly worse than the best. If they are too narrow, the solution may be absent. Therefore, when using the Monte Carlo method in MATLAB Simulink, a method for determining the capabilities of the system for a given object is required. In the proposed method for numerical optimization for a

given cost function, this is not required. The result is optimal within the given modeling conditions and values of weight coefficients.

Author Contributions: Conceptualization, V.Z.; methodology, V.Z. and L.D.; validation, V.Z. and L.D.; formal analysis, L.D.; investigation, V.Z. and L.D.; resources, V.Z.; data curation, V.Z. and L.D.; writing—original draft preparation, V.Z.; writing—review and editing, L.D.; funding acquisition, V.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The work was carried out with the support of the Ministry of Education and Science of the Russian Federation (within the framework of state assignment No. 121033100068-7, 2025 year) "Development of scientific methods for precision optical measurements of seismic, acoustic and optical fields and methods for constructing atmospheric ultraviolet optical communication with luminescent antennas for monitoring objects with natural and anthropogenic hazards" (research) and the work was carried out with the support of the Ministry of Education and Science of Russia (within the framework of state assignment No. 075-00604-25) and using data obtained at the unique scientific installation "Seismo-infrasound complex for monitoring the Arctic permafrost zone and a complex for continuous seismic monitoring of the Russian Federation and adjacent territories of the world" (validation).

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Let us show that some optimization problems have no solution, although the calculation of the controller for such objects can be performed quite simply. We will also show that for these reasons the mathematical model of the object, which does not contain a pure delay unit, is always insufficiently complete.

Let us show this theoretically first. Let us take a simple first-order link.

It is described by the following transfer function:

$$W(s) = \frac{K}{1 + Ts} \tag{A1}$$

For this transfer function, the amplitude–frequency response and the phase–frequency response can be plotted as Figure A1 shows.

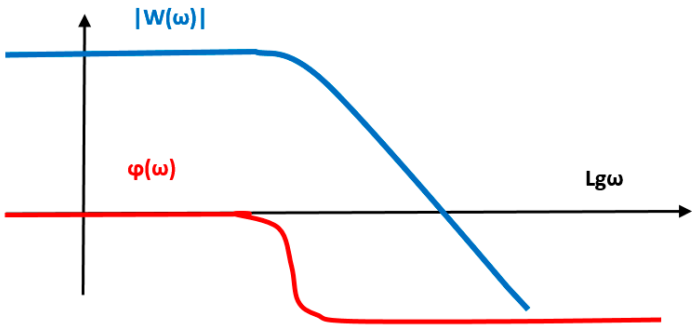

Figure A1. Amplitude–frequency $|W(\omega)|$ and phase–frequency $\phi(\omega)$ characteristics of a first-order object depending on the frequency on a logarithmic scale $Lg\omega$, according to the relation (A1).

The Nyquist–Mikhailov stability criterion for a closed linear dynamic system requires that, in the frequency range (where the logarithmic amplitude–frequency function of the open loop is greater than zero), the phase–frequency characteristic remains within the limits of not less than -180°, i.e., in absolute value not exceed π rad. In the graph shown

in Figure A1, the phase–frequency characteristic asymptotically tends to the value $-90°$. Thus, in the entire frequency range, including infinite frequencies, the phase–frequency characteristic does not exceed in absolute value even half of the critical value. Therefore, at any arrangement of the amplitude–frequency characteristic, a system with such a phase–frequency characteristic will remain stable. With increasing gain of such a system, the amplitude–frequency characteristic will rise along the ordinate axis. Since its right side is often an infinite beam with a constant slope of -20 dB/dec, as the gain of the circuit increases, the operating bandwidth of the system will increase proportionally, and the gain at each frequency will also increase proportionally. Such a system will get better and better with increasing gain, and it will never get worse, because the gain in it increases compared to the previous value.

Thus, it turns out that the theoretical model of an object in the form of a transfer function (A1) has the following property: a system with such an object and a sequential proportional regulator is better and the greater the gain. There is no limit to this value, after which its increase would not give a positive effect consisting in increasing the accuracy and expanding the range of the resulting system.

No real object with a proportional controller has such a property. In any real object, any controller, including a proportional or even a PID controller, always does not allow an infinite increase in any coefficient or the total transmission coefficient. Therefore, in any real object there is an optimal value such that not only a decrease in these coefficients but also their increase will lead to a deterioration in the performance of the system. This, as a rule, manifests itself in a violation of the stability of the system.

If we consider the delay link in the object model, then the object model will be described by the following transfer function:

$$W(s) = \frac{K}{1+Ts}\exp(-\tau s) \tag{A2}$$

In this case, the amplitude–frequency response graph will not change. But the phase–frequency response graph will change dramatically, as shown in the figure below, which will be fatal to the stability of the system. Both graphs together will look like Figure A2.

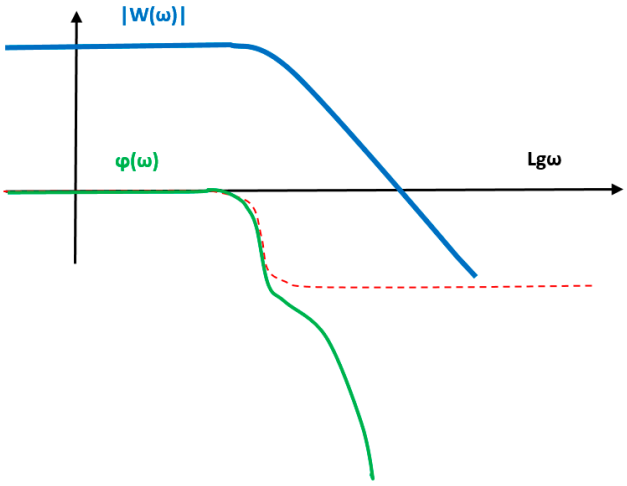

Figure A2. Amplitude–frequency $|W(\omega)|$ (blue curve) and phase–frequency $\phi(\omega)$ (red dash curve) characteristics of a first-order object depending on the frequency on a logarithmic scale Lgω in the presence of a delay link (red solid line), according to the relation (A2).

For the case of Figure A2, the reasoning for the case of Figure A1 is no longer applicable. The phase–frequency characteristic increases linearly with increasing frequency, and since

the frequency axis is plotted on a logarithmic scale, the phase graph increases exponentially in magnitude with increasing frequency logarithm, while remaining negative. The phase delay value in absolute value reaches and exceeds the value 180° quite quickly with increasing frequency. The frequency at which the phase delay is equal to −180° is the limiting frequency to which the frequency band of the closed system can be expanded. It is possible to locally reduce the phase shift value by using differentiation, but this does not solve the problem radically; this limiting point will only shift slightly to the left along the frequency axis.

On this basis, it can be argued that mathematical results obtained for objects without considering the delay are abstract mathematical results. For this reason, no model of an object in which there is a delay link, even if the delay value is very small, will ever have the property discussed above. A system with a proportional controller will never be stable in the entire range of gain coefficients of this controller. The same can be said about an object with a delay relative to any type of controller.

If we consider the model of a second-order object without delay, then the above-mentioned property of maintaining stability with an arbitrarily large gain coefficient of the locked loop will take place for the case of a PID controller, since ideal derivation reduces the phase shift of the loop in absolute value by 90°, as a result of which the phase–frequency characteristic of a series-connected derivative link and a second-order object in the high-frequency region will be the same as shown in Figure A1.

Thus, we see that there are problems that do not have an optimal solution, since for any solution it is always possible to specify a better solution with a higher value of the gain. We also see that this situation occurs only in theory when a simplified model is used that does not have a delay. In practice, every object has a delay due to the propagation of the signal in it, even if we take the fastest of all known control objects today, for example, a microwave transistor. It also has a delay, since the electrical signal propagates in conductors (and in semiconductors too) at a speed that does not exceed the speed of light in a vacuum. Even if the dimensions of the waveguide in the transistor are only 1 cm, then in this case it will give a delay of 33 ps. Thus, we have no right to use mathematical models of controlled objects without delay. And since analytical methods, as a rule, neglect the delay, they have no applied value. It is obvious that the delay can be neglected only if the minimum phase–frequency characteristic of the object model is a model of the third order or higher, and the value of the delay is so small that the delay begins to affect the actual value of the phase characteristic only from those frequencies at which this characteristic in absolute value already significantly exceeds the value of 180°. The publications we have considered, where the delay is neglected, as a rule, do not fall under this condition.

The experimental proof of this is as follows. One can take a model of a first-order object (Figure A1), then create a project for modeling and optimizing a proportional controller for it. The optimization will end with some solution, but one can easily verify that the reason for stopping the optimization procedure was that the modeled system reached the required value in one- or two-time sampling steps. If one then decreases the time sampling step, for example, by 20 times and continue the optimization procedure from the stated value, a new value of the “optimal” coefficient of the proportional controller will be found, and with this new value the cost function will decrease significantly (approximately by 20 times). This can be carried out ad infinitum. Thus, the stop factor for the optimization problem is a delay in the modeled system by a value equal to several time sampling steps. We will not provide the results of modeling this problem, since they can easily be obtained by any user who doubts this, which will be more convincing evidence of the described situation.

The conclusion to be drawn from this section is as follows. As a rule, any mathematical model of any controlled object is known only approximately and only in a limited

frequency band. Starting from some fairly high frequencies (in comparison with the frequencies characteristic of this object, where its transfer function changes most rapidly), a mathematical model of the object cannot be constructed based on experimental data, since in this frequency range there are simply no experimental data, or they are negligibly small in comparison with the measurement noise. Therefore, engineers create a mathematical model of the object based on the requirement that it should match its characteristics as accurately as possible with those obtained experimentally. Therefore, the task of matching the high-frequency properties of this model with its actual properties is simply not set.

However, the following paradox arises. In the absence of information about how the model behaves in the region of the highest frequencies, engineers refuse to make a clarifying addition. They choose the simplest model of all models that correspond to the experimental graphs of the object's reactions to test effects. But the simplest model is the most unrealistic. It corresponds to the truth to the least extent. Adding a low-pass filter with a very small time constant, as well as adding a delay unit with a very small delay value, does not affect the response in the region of low and medium frequencies. Therefore, such an addition should not be made. But the problem is that, without such an addition, the object becomes unrealistically ideal, and the theory of controlling such an object provides opportunities for expanding the frequency band and increasing accuracy much more than is the case in practice. It turns out that, if we cannot measure the properties of the object in the region of these high frequencies, we, due to the simplification of the model, assume the best possible and even impossible properties. Whereas, in fact, in the domain that cannot be investigated, we would have to assume the worst properties of all possible ones. In this case, our model, when used to design a regulator, would give the most reliable results in theory and numerical modeling, which are fully confirmed when these results are used in practice.

The simplest and, in our opinion, mandatory way to assume the worst-case properties of an object in the frequency range is to introduce a delay unit into the object model with a value small enough so that this delay does not manifest itself in any way when identifying the object model. This addition immediately turns the object into a realistic one, possessing all the necessary properties of a real object.

There is one additional reason for this approach. If the designer has mistakenly assumed during modeling and optimization that the object delay is greater than it actually is, the calculated controller will not perform worse than it did during modeling. However, if during optimization the designer has assumed that the delay is less than it actually is, then the system with the calculated controller will actually perform worse than it was during modeling and may even become unstable. At the very least, the overshoot in such a system will increase compared to the modeling results. An exception to this rule are systems using the Smith predictor: such systems, as modeling has shown, will lose their stability both if the actual object delay is greater than the assumed one and if it is less than the assumed one. For this reason, systems using the Smith predictor are not robust and simply should not be used if a reliable automatic control system is required.

Appendix B. On the Use of the Method

We have used fictitious examples to examine and illustrate control methods. This is in keeping with the tradition of publications in this area and has the advantage that we can examine the most common problems.

Numerical optimization has been used to calculate many controllers for practical and educational tasks. In particular, postgraduate student A. Yu. Ivoylov and senior lecturer V. G. Trubin designed and manufactured a compact balancing robot, shown in Figure A3. The video at the link [74] demonstrates the operation of two versions of this robot [75,76].

The robot on the left maintains balance but makes some deviations around the equilibrium state. The robot on the right maintains balance without deviations, which is ensured by optimizing the PID controller under the supervision of one of the authors of this article.

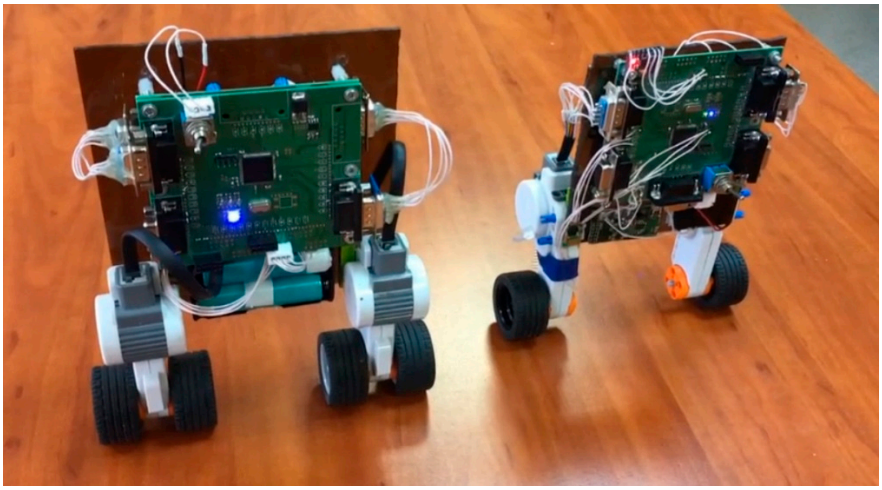

Figure A3. External view of two robots developed using the numerical optimization method of single-channel controllers [74–76].

Appendix C. Powell Method for the Optimization of Multiargument Function

Powell's method was published by the author himself in [77] as an effective method for finding the minimum of a function of several variables without calculating derivatives.

This method belongs to direct methods, that is, zero-order methods, which means that these methods do not require the calculation of derivatives.

The simplest of these methods is the Gaussian algorithm. It involves sequential optimization over individual variables. This method is very simple but not very efficient. Problems can arise when the level lines are very elongated and the ellipsoids of equal value of the objective function are oriented along inclined lines, for example, directed toward the center of coordinates, as shown in Figure A4. For example, in the two-dimensional case, this is a ravine that is located at an angle to all axes. In this case, the search quickly gets stuck at the bottom of such a ravine and then does not move to the minimum of this axis. Instead, the process stops at the point where it approaches this axis.

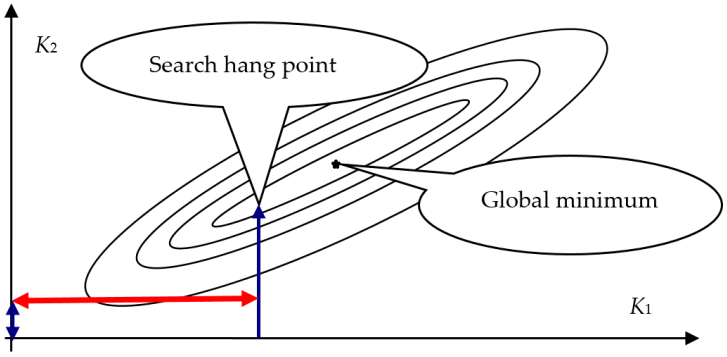

Figure A4. Hanging on a trough using the coordinate descent method.

Rosenbrock's method consists of rotating the search axes based on the results of the first steps of the Gauss method. New axes are obtained by rotating the old axes. The position of the new axes can be obtained by linearly transforming the old axes. Let there be

n arguments of the objective function. After optimization by the Gauss method for each of the n arguments, we obtain the point X_n. At this point, none of the arguments coincides with the starting value of this quantity. After the next optimization procedure, we obtain the point X_{2n}. At this point, none of the coordinates coincides with the coordinate from the point X_n. The vector $X_{2n} - X_n$ specifies the direction of the new search, and other vectors orthogonal to this vector are the directions of additional search scans. This method is similar to rotating the coordinates in such a direction that the first coordinate in the direction coincides with the ravine line.

The Hooke–Jeeves method is a successful modification of the Gauss method and is similar to the Rosenbrock method. According to this method, a series of n steps is first performed. Then an additional step is taken in the direction of the vector $X_{2n} - X_n$. If this step leads to an increase in the objective function, then a step is taken in the opposite direction. From the new point, steps are taken again for each coordinate separately, as in the Gauss method.

The Nelder–Mead or simplex method consists of searching by the corners of a deformable polyhedron. This method requires that n + 1 initial points be specified that form the polyhedron. It is then deformed using various methods: stretching, compressing, reducing, or reflecting, depending on where the vertex with the largest objective function value is located. This method should be effective, but it is not listed among the available optimization methods in VisSim, as are the methods discussed above.

Powell's method also refers to zero-order methods. This method most effectively performs minimization of functions close to quadratic. At each iteration of the algorithm, the search is performed along a system of conjugate directions.

When this algorithm operates, the information obtained in previous iterations is used to construct search direction vectors and to eliminate the looping of the coordinate search sequence. The method is oriented toward solving problems with quadratic objective functions and is based on fundamental theoretical results. The hypothesis that the actual form of the objective function near the extremum is approximately a quadratic function is apparently confirmed for the class of problems under consideration. In any case, a function with a smooth extremum near the extremum point is indeed close to quadratic, i.e., in its expansion into a Taylor series, terms of order 3 and higher can be neglected.

The basic idea of the algorithm is that, if a quadratic function of n variables is reduced to the form of a sum of perfect squares, then its optimum can be found because of implementing n one-dimensional searches along transformed coordinate directions. In fact, a coordinate system of n parameters, in which the target function has ravines whose lines are inclined relative to the coordinate axes, is transformed by certain mathematical methods into a coordinate system in which the axes of these ellipses or saddle functions are oriented along the new coordinate axes. In these new coordinates, a search is carried out using the Gauss method. The mathematical apparatus of matrix calculations is used to describe this method.

Thus, Powell's method exploits the property of a quadratic function that any line that passes through a minimum point of the function $F(x, y)$ intersects at equal angles the tangents to the level surfaces of the function at the intersection points, as shown in Figure A5.

The essence of the method is as follows. A certain initial point S_0 is selected and a one-dimensional search is performed along an arbitrary direction, leading to a point S_1. Then a point S_2 is selected that does not lie on the line $S_0 - S_1$, and a one-dimensional search is performed along a line parallel to this line $S_0 - S_1$. The resulting point S_3 together with the point S_2 determines the direction $S_1 - S_3$ of the one-dimensional search, which yields the minimum point. In the case of a quadratic function of n variables, the optimal

value is found in n iterations. The search for the minimum is ultimately performed in mutually conjugate directions.

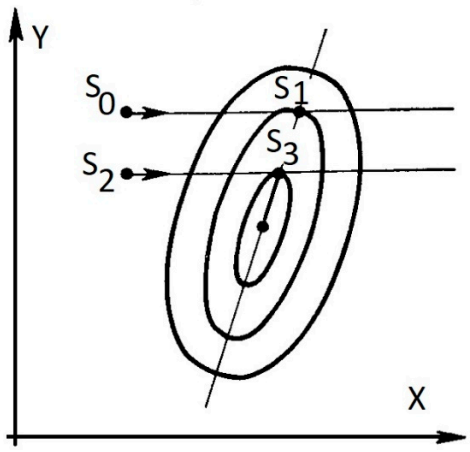

Figure A5. Geometric interpretation of Powell's method.

The algorithm of the parallel tangent's method is as follows.

1. They are defined by the initial point S_0. The initial directions of the search are taken to be the directions of the coordinate axes.

2. Perform n one-dimensional searches along the selected orthogonal directions. Each subsequent search is performed from the minimum point obtained in the previous step. A fully completed one-dimensional search cycle is performed for each direction, for example, using the golden section method. The starting direction of the i-th search is determined by the first increment equal to $\Delta S = [0, 0, \ldots 1, 0, \ldots 0]$, where the unit is at the i-th position.

The obtained result defines a new point S_k.

It is further proposed to search in the direction of the obtained increment:

$$S_{k+1} = S_k + a_k\,(S_k - S_0).$$

3. Select a new search direction and rename the axes.

4. Perform a one-dimensional search along the new directions. Replace S_0 with S_{k+1} and take this new point as the starting point for the next iteration. Go to step 1.

Thus, because of the execution of the procedure considered, the search directions adopted at the beginning are replaced one by one. As a result, after n steps, they will be mutually conjugate.

References

1. Konkov, A.E.; Mitrishkin, Y.V. Synthesis Methodology for Discrete MIMO PID Controller with Loop Shaping on LTV Plant Model via Iterated LMI Restrictions. *Mathematics* **2024**, *12*, 810. [CrossRef]
2. Krokavec, D.; Filasová, A. On PID Design Constraints in Relation to Control of Strictly Metzler Linear MIMO Systems. *Symmetry* **2021**, *13*, 1589. [CrossRef]
3. Guo, X.; Shirkhani, M.; Ahmed, E.M. Machine-Learning-Based Improved Smith Predictive Control for MIMO Processes. *Mathematics* **2022**, *10*, 3696. [CrossRef]
4. de Almeida, A.M.; Daga, A.L.; Lanzarini, R.P.S.P.; Lenzi, E.K.; Lenzi, M.K. High-Performance Identification and Control of MIMO (Multiple Input—Multiple Output) Experimental Module with Fractional-Order Approach Application. *Fractal Fract.* **2025**, *9*, 226. [CrossRef]
5. Garrido, J.; Garrido-Jurado, S.; Vázquez, F.; Arrieta, O. Design of Multivariable PID Control Using Iterative Linear Programming and Decoupling. *Electronics* **2024**, *13*, 698. [CrossRef]
6. Peretz, Y. A Randomized Algorithm for Optimal PID Controllers. *Algorithms* **2018**, *11*, 81. [CrossRef]
7. Afifa, R.; Ali, S.; Pervaiz, M.; Iqbal, J. Adaptive Backstepping Integral Sliding Mode Control of a MIMO Separately Excited DC Motor. *Robotics* **2023**, *12*, 105. [CrossRef]

8. Piñon, A.; Favela-Contreras, A.; Beltran-Carbajal, F.; Lozoya, C.; Dieck-Assad, G. Novel Strategy of Adaptive Predictive Control Based on a MIMO-ARX Model. *Actuators* **2022**, *11*, 21. [CrossRef]
9. La, H.; Oh, K. Development of a Universal Adaptive Control Algorithm for an Unknown MIMO System Using Recursive Least Squares and Parameter Self-Tuning. *Actuators* **2024**, *13*, 167. [CrossRef]
10. Kato, T.; Nishizawa, K.; Deng, M. MSVR & Operator-Based System Design of Intelligent MIMO Sensorless Control for Microreactor Devices. *Computation* **2024**, *12*, 2. [CrossRef]
11. Zhmud, V.A.; Frantsuzova, G.A.; Dimitrov, L.; Nosek, J. Formalization of Requirements for Locked-Loop Control Systems for Their Numerical Optimization, Recent Research in Control Engineering and Decision Making. In *Systems, Decision and Control*; Springer International Public: Cham, Switzerland, 2019; Volume 199, pp. 353–365, ISBN 978-3-030-12072-6. [CrossRef]
12. Zhmud, V.; Dimitrov, L. Using the Fractional Differential Equation for the Control of Objects with Delay. *Symmetry* **2022**, *14*, 635. [CrossRef]
13. Lv, J.; Ju, X.; Wang, C. Neural network prescribed-time observer-based output-feedback control for uncertain pure-feedback nonlinear systems. *Expert Syst. Appl.* **2025**, *264*, 125813. [CrossRef]
14. Ju, X.; Jiang, Y.; Jing, L.; Liu, P. Quantized predefined-time control for heavy-lift launch vehicles under actuator faults and rate gyro malfunctions. *ISA Trans.* **2023**, *138*, 133–150. [CrossRef]
15. Xu, X.; Li, B. Semi-global stabilization of parabolic PDE–ODE systems with input saturation. *Automatica* **2025**, *171*, 111931. [CrossRef]
16. Xu, X.; Li, B. PDE-based observation and predictor-based control for linear systems with distributed infinite input and output delays. *Automatica* **2024**, *170*, 111845. [CrossRef]
17. Lan, Y.; Zhao, J. Improving Track Performance by Combining Padé-Approximation-Based Preview Repetitive Control and Equivalent-Input-Disturbance. *J. Electr. Eng. Technol.* **2024**, *19*, 3781–3794. [CrossRef]
18. Cheng, Z.; Li, L.; Zhong, C.; Wang, J.; Bai, X.; Liu, J. Adaptive Nonlinear Active Disturbance Rejection Current Controller for Distributed Generation System Considering Uncertain Ripples. *IEEE Trans. Power Electron.* **2025**, *40*, 4984–4996. [CrossRef]
19. Li, D.; Tong, S.; Yang, H.; Hu, Q. Time-Synchronized Control for Spacecraft Reorientation with Time-Varying Constraints. *IEEE/ASME Trans. Mechatron.* **2025**, *30*, 2073–2083. [CrossRef]
20. Du, G.; Zhang, H.; Yu, H.; Hou, P.; He, J.; Cao, S.; Ma, L. Study on Automatic Tracking System of Microwave Deicing Device for Railway Contact Wire. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 3527611. [CrossRef]
21. Chen, J.; Wang, J.; Wang, J.; Bai, L. Joint Fairness and Efficiency Optimization for CSMA/CA-Based Multi-User MIMO UAV Ad Hoc Networks. *IEEE J. Sel. Top. Signal Process.* **2024**, *18*, 1311–1323. [CrossRef]
22. Wang, J.; Wu, Y.; Chen, C.L.P.; Liu, Z.; Wu, W. Adaptive PI event-triggered control for MIMO nonlinear systems with input delay. *Inf. Sci.* **2024**, *677*, 120817. [CrossRef]
23. Ding, F.; Zhu, K.; Liu, J.; Peng, C.; Wang, Y.; Lu, J. Adaptive Memory Event-Triggered Output Feedback Finite-Time Lane-Keeping Control for Autonomous Heavy Truck with Roll Prevention. *IEEE Trans. Fuzzy Syst.* **2024**, *32*, 6607–6621. [CrossRef]
24. Zhang, X.; Liu, Y.; Chen, X.; Li, Z.; Su, C. Adaptive Pseudoinverse Control for Constrained Hysteretic Nonlinear Systems and its Application on Dielectric Elastomer Actuator. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 2142–2154. [CrossRef]
25. Wu, J.; Wang, Y.; Yin, C. Curvilinear Multilane Merging and Platooning With Bounded Control in Curved Road Coordinates. *IEEE Trans. Veh. Technol.* **2022**, *71*, 1237–1252. [CrossRef]
26. Wang, F.; Chen, K.; Zhen, S.; Chen, X.; Zheng, H.; Wang, Z. Prescribed Performance Adaptive Robust Control for Robotic Manipulators with Fuzzy Uncertainty. *IEEE Trans. Fuzzy Syst.* **2024**, *32*, 1318–1330. [CrossRef]
27. Hu, J.; Chen, B.; Ghosh, B.K. Formation–Circumnavigation Switching Control of Multiple ODIN Systems via Finite-Time Intermittent Control Strategies. *IEEE Trans. Control. Netw. Syst.* **2024**, *11*, 1986–1997. [CrossRef]
28. Meng, Q.; Ma, Q.; Shi, Y. Adaptive Fixed-Time Stabilization for a Class of Uncertain Nonlinear Systems. *IEEE Trans. Autom. Control.* **2023**, *68*, 6929–6936. [CrossRef]
29. Xiong, W.; He, D.; Du, H. Learning economic model predictive control via clustering and kernel-based Lipschitz regression. *J. Frankl. Inst.* **2025**, *362*, 107787. [CrossRef]
30. Li, Y.; Jiang, Y.; Lu, J.; Tan, C. Improved active disturbance rejection control for electro-hydrostatic actuators via actor-critic reinforcement learning. *Eng. Appl. Artif. Intell.* **2025**, *158*, 111485. [CrossRef]
31. Tang, Q.; Qu, S.; Zheng, W.; Tu, Z. Fast finite-time quantized control of multi-layer networks and its applications in secure communication. *Neural Netw.* **2025**, *185*, 107225. [CrossRef]
32. Xu, B.; Wang, X.; Zhang, J.; Guo, Y.; Razzaqi, A.A. A novel adaptive filtering for cooperative localization under compass failure and non-gaussian noise. *IEEE Trans. Veh. Technol.* **2022**, *71*, 3737–3749. [CrossRef]
33. Baar, T.; Bauer, P.; Luspay, T. Parameter Varying Mode Decoupling for LPV systems. *IFAC Pap.* **2020**, *53*, 1219–1224. [CrossRef]
34. Feng, Z.Y.; Guo, H.; She, J.; Xu, L. Weighted sensitivity design of multivariable PID controllers via a new iterative LMI approach. *J. Process Control* **2022**, *110*, 24–34. [CrossRef]

35. Guo, H.; Feng, Z.Y.; She, J. Discrete-time multivariable PID controller design with application to an overhead crane. *Int. J. Syst. Sci.* **2020**, *51*, 2733–2745. [CrossRef]
36. Solgi, Y.; Fatehi, A.; Nikoofard, A.; Shariati, A. Design of optimal PID controller for multivariable time-varying delay discrete-time systems using non-monotonic Lyapunov-Krasovskii approach. *J. Frankl. Inst.* **2021**, *358*, 6634–6665. [CrossRef]
37. Zhmud, V.A.; Dimitrov, L.V. The method of correct calculation of regulators based on numerical optimization. *J. Phys. Conf. Ser.* **2019**, *1210*, 012168. [CrossRef]
38. Zhmud, V.A.; Dimitrov, L.V. The influence of the type of the test signal on the result of numerical optimization of regulators. *J. Phys. Conf. Ser.* **2017**, *803*, 012186. [CrossRef]
39. Konkov, A.E.; Mitrishkin, Y.V.; Korenev, P.S.; Patrov, M.I. Robust Cascade LMI Design of MIMO Control System for Plasma Position, Current, and Shape Model with Time-Varying Parameters in a Tokamak. *IFAC Pap.* **2020**, *53*, 7344–7349. [CrossRef]
40. Mitrishkin, Y.V.; Korenev, P.S.; Konkov, A.E.; Kartsev, N.M. Suppression of Vertical Plasma Displacements by Control System of Plasma Unstable Vertical Position in D-Shaped Tokamak. *Autom. Remote Control.* **2022**, *83*, 579–599. [CrossRef]
41. Konkov, A.E.; Mitrishkin, Y.V. Comparison Study of Power Supplies in Real-Time Robust Control Systems of Vertical Plasma Position in Tokamak. *IFAC Pap.* **2022**, *55*, 327–332. [CrossRef]
42. Konkov, A.E.; Korenev, P.S.; Mitrishkin, Y.V.; Balachenkov, I.M.; Kiselev, E.O. Real-Time Plasma Magnetic Control System with Equilibrium Reconstruction Algorithm in the Feedback for the Globus-M2 Tokamak. *Plasma Phys. Rep.* **2023**, *49*, 1552–1559. [CrossRef]
43. Krokavec, D.; Filasová, A. PID control design for SISO strictly Metzlerian linear systems. *Symmetry* **2020**, *12*, 1979. [CrossRef]
44. Tsavnin, A.; Efimov, S.; Zamyatin, S. Overshoot elimination for control systems with parametric uncertainty via a PID controller. *Symmetry* **2020**, *12*, 1092. [CrossRef]
45. Zhang, W.; Cui, Y.; Ding, X. An improved analytical tuning rule of a robust PID controller for integrating systems with time delay based on the multiple dominant pole-placement method. *Symmetry* **2020**, *12*, 1449. [CrossRef]
46. An, Q.; Guo, H.; Zheng, Y. On Robust Stability and Stabilization of Networked Evolutionary Games with Time Delays. *Mathematics* **2022**, *10*, 2695. [CrossRef]
47. Liu, X.; Li, W.; Yao, C.; Li, Y. Finite-Time Guaranteed Cost Control for Markovian Jump Systems with Time-Varying Delays. *Mathematics* **2022**, *10*, 2028. [CrossRef]
48. Tavoosi, J.; Shirkhani, M.; Abdali, A.; Mohammadzadeh, A.; Nazari, M.; Mobayen, S.; Asad, J.H.; Bartoszewicz, A. A New General Type-2 Fuzzy Predictive Scheme for PID Tuning. *Appl. Sci.* **2021**, *11*, 10392. [CrossRef]
49. Euzébio, T.A.; Yamashita, A.S.; Pinto, T.V.; Barros, P.R. SISO approaches for linear programming based methods for tuning decentralized PID controllers. *J. Process Control* **2020**, *94*, 75–96. [CrossRef]
50. Garrido, J.; Ruz, M.L.; Morilla, F.; Vázquez, F. Iterative Method for Tuning Multiloop PID Controllers Based on Single Loop Robustness Specifications in the Frequency Domain. *Processes* **2021**, *9*, 140. [CrossRef]
51. Carlucho, I.; De Paula, M.; Acosta, G.G. An adaptive deep reinforcement learning approach for MIMO PID control of mobile robots. *ISA Trans.* **2020**, *102*, 280–294. [CrossRef] [PubMed]
52. Veerasamy, V.; Wahab, N.I.A.; Ramachandran, R.; Othman, M.L.; Hizam, H.; Kumar, J.S.; Irudayaraj, A.X.R. Design of single-and multi-loop self-adaptive PID controller using heuristic based recurrent neural network for ALFC of hybrid power system. *Expert Syst. Appl.* **2022**, *192*, 116402. [CrossRef]
53. de Almeida, A.M.; Lenzi, M.K.; Lenzi, E.K. Decentralized fractional-order PI λ control applied to a TITO distillation column: A comparative study with PID and MPC. *Cuad. Educ. Desarro.* **2024**, *16*, e6021. [CrossRef]
54. Edet, E.B.; Chacón-Vásquez, M.; Onyeocha, E.C. Application of Fractional-order PID controllers in a Greenhouse Climate Control System. *IFAC Pap.* **2024**, *58*, 179–184. [CrossRef]
55. Martínez-Luzuriaga, P.N.; Reynoso-Meza, G. Influencia de Los Hiper-Parámetros En Algoritmos Basados En Evolución Diferencial Para El Ajuste de Controladores del tipo PID en Procesos SISO. *Rev. Iberoam. Automática Informática Ind.* **2023**, *20*, 44–55. [CrossRef]
56. Rojas, J.D.; Arrieta, O.; Vilanova, R. *Industrial PID Controller Tuning with a Multiobjective Framework Using MATLAB®*, 1st ed.; Springer: Cham, Switzerland, 2022; ISBN 978-3-030-72313-2.
57. Jeng, J.C.; Lee, M.W. Multi-Loop PID Controllers Design with Reduced Loop Interactions Based on a Frequency-Domain Direct Synthesis Method. *J. Frankl. Inst.* **2023**, *360*, 2476–2506. [CrossRef]
58. Mahapatro, S.R.; Subudhi, B. A New Hinf Weighted Sensitive Function Based Robust Multi-Loop PID Controller for a Multi-Variable System. *IEEE Trans. Circuits Syst. II Express Briefs* **2024**, *71*, 1256–1260. [CrossRef]
59. Mahapatro, S.R.; Subudhi, B. A Robust Decentralized PID Controller Based on Complementary Sensitivity Function for a Multivariable System. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *67*, 2024–2028. [CrossRef]
60. Euzebio, T.A.M.; Da Silva, M.T.; Yamashita, A.S. Decentralized PID Controller Tuning Based on Nonlinear Optimization to Minimize the Disturbance Effects in Coupled Loops. *IEEE Access* **2021**, *9*, 156857–156867. [CrossRef]
61. Garrido, J.; Ruz, M.L.; Morilla, F.; Vázquez, F. Iterative Design of Centralized PID Controllers Based on Equivalent Loop Transfer Functions and Linear Programming. *IEEE Access* **2022**, *10*, 1440–1450. [CrossRef]

62. Da Silva, M.T.; Barros, P.R. An Iterative Procedure for Tuning Decentralized PID Controllers Based on Effective Open-Loop Process. In Proceedings of the 7th International Conference on Control, Decision and Information Technologies, CoDIT 2020, Prague, Czech Republic, 29 June–2 July 2020; pp. 813–818.
63. Pongfai, J.; Angeli, C.; Shi, P.; Su, X.; Assawinchaichote, W. Optimal PID controller autotuning design for MIMO nonlinear systems based on the adaptive SLP algorithm. *Int. J. Control Autom. Syst.* **2021**, *19*, 392–403. [CrossRef]
64. Tajudin, A.I.; Izani, M.A.D.; Samat, A.A.A.; Omar, S.; Idin, M.A.M. Design a speed control for DC motor using an optimal PID controller implementation of ABC algorithm. In Proceedings of the IEEE 12th International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, 21–22 October 2022; pp. 97–102.
65. Li, F.; Yang, L.; Ye, A.; Zhao, Z.; Shen, B. Grouping Neural Network-Based Smith PID Temperature Controller for Multi-Channel Interaction System. *Electronics* **2024**, *13*, 697. [CrossRef]
66. Liu, Z.; Liu, Z.; Liu, J.; Wang, N. Thermal management with fast temperature convergence based on optimized fuzzy PID algorithm for electric vehicle battery. *Appl. Energy* **2023**, *352*, 121936. [CrossRef]
67. Han, S.Y.; Dong, J.F.; Zhou, J.; Chen, Y.H. Adaptive Fuzzy PID Control Strategy for Vehicle Active Suspension Based on Road Evaluation. *Electronics* **2022**, *11*, 921. [CrossRef]
68. Abdelwanis, M.I.; El-Sousy, F.F.M.; Ali, M.M. A Fuzzy-Based Proportional Integral Derivative with Space-Vector Control and Direct Thrust Control for a Linear Induction Motor. *Electronics* **2023**, *12*, 4955. [CrossRef]
69. Ding, Y.; Ren, X.; Zhang, X.; Liu, X.; Wang, X. Multi-Phase Focused PID Adaptive Tuning with Reinforcement Learning. *Electronics* **2023**, *12*, 3925. [CrossRef]
70. Gopmandal, F.; Ghosh, A. LQR-based MIMO PID control of a 2-DOF helicopter system with uncertain cross-coupled gain. *IFAC Pap.* **2022**, *55*, 183–188. [CrossRef]
71. Sagar, J.D. DC Motor Control using PID Controller. *Int. Res. J. Eng. Technol.* **2020**, *7*, 1765–1769.
72. Wati, T.; Sutarno, S. Simulation model of speed control DC motor using fractional order PID controller. *J. Phys. Conf. Ser.* **2020**, *1444*, 012022. [CrossRef]
73. Zhmud, V. Controlling of Multichannel Objects with Non-Square Transfer Function. *Automation* **2025**, *6*, 27. [CrossRef]
74. Design Bureau "Automatika", Content Video. Available online: https://kb-au.ru/wp-content/uploads/NSTU-PLATF2_3-2018.mp4 (accessed on 30 February 2022).
75. Zhmud, V.A.; Dimitrov, L.V.; Stukach, O.V.; Ivoilov, A.Y.; Roth, H.; Sablina, G.V. Suppression of oscillations of a two-wheeled balancing robot. In Proceedings of the Dynamics of Systems, Mechanisms and Machines (DYNAMICS), Omsk, Russia, 5–7 November 2019; p. 5, ISBN 978-1-7281-0970-1. Available online: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8944728 (accessed on 21 November 2019). [CrossRef]
76. Ivoilov, A.Y.; Zhmud, V.A.; Trubin, V.G.; Roth, H. Parametric Synthesis of the Control System of the Balancing Robot by the Numerical Optimization Method. *Mekhatronika Avtom. Upr.* **2019**, *20*, 352–361. [CrossRef]
77. Powell, M.J.D. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Comput. J.* **1964**, *7*, 155–162. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.