

Article

Machine Learning-Assisted Synthesis of Self-Organizing SISO Control Systems with Guaranteed Lyapunov Stability

Nurgul Shazhdekeyeva ¹, Beket Kenzhegulov ¹, Kamka Uteuliyeva ¹, Gulash Kochshanova ², Gulmira Nigmatova ³, Lyailya Kurmangaziyeva ^{4,*}, Raigul Tuleuova ^{1,*} , Saya Kenzhegulova ⁵ and Raushan Moldasheva ⁴ 

¹ Department of Mathematics and Methods of Teaching Mathematics, Faculty of Physics, Mathematics and Information Technology, Kh. Dosmukhamedov Atyrau University, Atyrau 060000, Kazakhstan

² Department of Basic Sciences, Yessenov University, Aktau 130000, Kazakhstan

³ Department of Fundamental Sciences, Faculty of Pedagogy, Yessenov University, Aktau 130000, Kazakhstan

⁴ Department of Software Engineering, Faculty of Physics, Mathematics and Information Technology, Kh. Dosmukhamedov Atyrau University, Atyrau 060000, Kazakhstan; raushan85_07@mail.ru

⁵ Department of Marketing and Information Systems, University of International Business, Almaty 050000, Kazakhstan

* Correspondence: kurmangazieval475@gmail.com (L.K.); tuleuova_79@mail.ru (R.T.)

Abstract

The proposed methodology combines analytical control laws with adaptive mechanisms and machine-learning-assisted modules based on regression trees, random forests, and extreme gradient boosting (XGBoost). Machine learning models are employed to approximate unknown nonlinear dynamics, compensate disturbances, and adjust controller parameters, while the overall control structure is constrained by Lyapunov stability conditions. This ensures that the inclusion of data-driven components does not violate the fundamental requirement of system stability. The effectiveness of the proposed approach is evaluated through simulation experiments across three operating modes with varying degrees of nonlinearity and dynamic complexity. The results show that hybrid models incorporating ensemble machine learning methods improved performance compared with the analytical and adaptive baselines examined. XGBoost-based control achieves the lowest error values and the highest level of Lyapunov stability compliance (up to 99.3%). The main contribution of this study lies in the development of a unified synthesis framework in which machine learning is not used as a standalone control strategy but as a machine-learning-assisted support mechanism integrated into a theoretically grounded control architecture. The proposed approach provides a balance between adaptability, accuracy, and rigorous stability guarantees, suggesting potential applicability to simulation-based and offline-assisted control design tasks, while real-time embedded implementation requires additional computational optimization and validation.

Keywords: self-organizing control systems; SISO systems; Lyapunov stability; machine learning in control; adaptive control; nonlinear dynamic systems; random forest; XGBoost; control synthesis; machine-learning-assisted control systems



Received: 18 May 2026

Revised: 13 June 2026

Accepted: 15 June 2026

Published: 19 June 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In the context of the rapid development of machine-learning-assisted control technologies, the task of synthesizing self-regulating control systems capable of ensuring stable and adaptive behavior of an object in conditions of uncertainty, non-linearity and external disturbances is becoming increasingly urgent [1]. A special place in this field is occupied by

single-input and single-output systems (SISO systems), which, despite their relative structural simplicity, are of considerable interest from both a theoretical and an applied point of view. This is because many real-world technical, electromechanical, energy, and technological facilities can be brought to a SISO representation, which makes them a convenient platform for developing, analyzing, and testing new approaches to control synthesis [2].

Classical control theory offers a wide range of methods for constructing regulators for linear and nonlinear dynamic systems. However, the use of traditional approaches is often insufficient in cases where the control object is characterized by complex nonlinear dynamics, changing parameters, incomplete a priori information, or the need for self-adjustment during operation [3]. In such conditions, there is an increasing interest in self-regulating and self-organizing control systems, in which the mechanism of control action formation is not rigidly fixed, but adapts to the current state of the system and the nature of its dynamics [4].

One of the central problems in the study of such systems is not only the construction of a control law, but also the rigorous proof of the stability of a closed system. In this context, the Lyapunov function method remains one of the most fundamental and reliable analysis tools [5]. Its main advantage is the ability to study the stability of a nonlinear system without the need to obtain an explicit analytical solution to differential equations. In the synthesis of self-regulating control systems, the Lyapunov method allows not only to verify the stability of an already constructed control law, but also to shape the synthesis process itself in such a way as to guarantee asymptotic stability, trajectory invariance, and the required quality of transients [6].

In recent years, machine learning methods have been significantly developed, which have been used not only in classification, recognition, and forecasting tasks, but also in dynamic system identification, approximation of unknown nonlinearities, selection of controller parameters, and construction of machine-learning-assisted control laws. Machine learning methods have high flexibility, the ability to identify hidden dependencies in data and adapt to complex nonlinear modes, which makes them promising tools for the synthesis of self-organizing control systems [7]. However, their key limitation remains insufficient interpretability and the lack of strict stability guarantees, especially in cases where management is based solely on empirical or data-driven models.

It should be noted that in modern literature, a significant number of studies are devoted either to the development of adaptive and machine-learning-assisted regulators, or to the use of machine learning to control nonlinear objects, or to Lyapunov stability analysis in the classical formulation [8–11]. However, limited attention has been paid to approaches that jointly consider SISO structure, nonlinear uncertainty, machine-learning-assisted synthesis, and Lyapunov-based stability guarantees.

In this paper, the focus is on the synthesis of self-regulating control systems for objects with one input and one output. It is assumed that machine learning is used as a synthesis support tool, for example, to approximate unknown dynamics, select a control action structure, adjust regulator parameters, or select an effective self-organization strategy [12,13]. In this case, the criterion of correctness and reliability of the constructed solution is the fulfillment of Lyapunov stability conditions. This approach shifts control design from purely empirical solutions to a theoretically grounded framework. Each control law is evaluated not only in terms of performance but also in terms of guaranteed stability.

Unlike purely data-driven approaches, the proposed framework emphasizes both control performance and Lyapunov-based stability verification.

The main purpose of the research is to develop and analyze an approach to the synthesis of self-regulating SISO control systems using machine learning methods while ensuring guaranteed Lyapunov stability [14–16]. To achieve this goal, the paper proposes

to solve the following tasks: to describe the class of control objects under consideration; to form the structure of a self-organizing control law; to determine the role of machine learning in the synthesis procedure; to construct a Lyapunov function for a closed system; to obtain conditions for guaranteed stability; and also to conduct a comparative analysis of the effectiveness of the proposed approach in relation to more traditional synthesis schemes [17].

The scientific novelty of the study lies in the fact that machine learning is considered not as an autonomous substitute for classical synthesis methods, but as a machine-learning-assisted mechanism for supporting control construction in a self-regulating SISO system, where the final correctness of the solution is confirmed by means of Lyapunov stability theory. This approach allows us to form a single framework that combines data-driven adaptation and strict analytical guarantees.

The practical significance of the work lies in the possibility of applying the developed approach to a wide range of automatic control facilities, which require a combination of adaptability, self-regulation and stability. Such objects may include electromechanical drives, technological processes, positioning systems, power plants, and other dynamic systems that allow SISO representation. The use of machine learning in control synthesis opens opportunities for improving the accuracy of adjustment and adaptation of the regulator, while Lyapunov analysis provides the necessary level of reliability and theoretical validity.

2. Materials and Methods

This study examines the synthesis of self-organizing control systems for objects with one input and one output (SISO) using machine learning methods and mandatory Lyapunov stability assurance. The methodology of the study combined nonlinear state-space modeling, adaptive control design, and machine learning-based approximation of uncertain dynamics within a Lyapunov-constrained synthesis framework.

This separation allows us to systematically consider both traditional methods of constructing regulators and modern machine-learning-assisted approaches that provide higher control flexibility in conditions of parameter uncertainty, external disturbances and object nonlinearity. Special attention is paid to ensuring that the use of machine learning algorithms does not violate the fundamental requirement of stability of a closed system. For this purpose, the regulator synthesis procedure is based on the Lyapunov function method, which makes it possible to formally prove the stability of the system and determine the conditions for convergence of trajectories to the required regime.

All numerical simulations and computational experiments were performed using MATLAB (R2024a, The MathWorks Inc., Natick, MA, USA). Machine learning models were implemented using Python (v3.11, Python Software Foundation, Wilmington, DE, USA), including the scikit-learn library (v1.4.2) and XGBoost library (v2.0.3). The simulations were executed on a personal computer equipped with an Intel Core i7 processor (Intel Corporation, Santa Clara, CA, USA) and 16 GB RAM.

2.1. Formal Problem Statement and Stability Assumptions

The controlled object is considered as a nonlinear SISO system in the following affine state-space form (1):

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + d(x(t), t), y(t) = h(x(t)), \quad (1)$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}$ is the control input, $y(t) \in \mathbb{R}$ is the system output, $f(x)$ and $g(x)$ describe the nominal nonlinear dynamics, and $d(x, t)$ denotes an unknown nonlinear disturbance or unmodelled dynamic component.

The objective of control is to ensure that the system output $y(t)$ tracks a bounded reference signal $y_r(t)$. The tracking error is defined as (2):

$$e(t) = y(t) - y_r(t).$$

The aim is to design a hybrid control law

$$u(t) = u_b(t) + u_{ml}(t), \quad (2)$$

where $u_b(t)$ is the analytically derived stabilizing component and $u_{ml}(t)$ is a machine-learning-based compensation term, such that the tracking error remains bounded and converges to a small neighbourhood of zero while satisfying Lyapunov stability conditions.

The following assumptions are introduced (3):

Assumption 1. The functions $f(x)$, $g(x)$ and $h(x)$ are continuous and locally Lipschitz in a compact operating domain $\mathbf{\Xi} \subset \mathbb{R}^n$.

Assumption 2. The control gain satisfies $g(x) \neq 0$ for all $x \in \mathbf{\Xi}$, which guarantees that the input can influence the system dynamics.

Assumption 3. The unknown nonlinear component $d(x, t)$ is bounded, i.e., there exists a positive constant $\bar{d}$ such that

$$|d(x, t)| \leq \bar{d}.$$

Assumption 4. The machine learning model provides an estimate

$$\hat{d}(t) = M(x(t), e(t), \dot{e}(t), u(t-1); \omega)$$

of the unknown component $d(x, t)$. The residual approximation error

$$\varepsilon(t) = d(x, t) - \hat{d}(t)$$

is bounded as

$$|\varepsilon(t)| \leq \bar{\varepsilon}. \quad (3)$$

Under these assumptions, the machine learning module is not treated as an independent controller, but as an auxiliary compensator embedded into a Lyapunov-constrained control architecture.

2.2. Lyapunov-Based Stability Theorem

Theorem 1. Consider the nonlinear SISO system defined above under Assumptions 1–4. Let the control law be given by (4)

$$u(t) = u_b(t) - \hat{d}(t),$$

where $u_b(t)$ is selected such that, in the absence of approximation error, the derivative of the Lyapunov function satisfies

$$\dot{V}(t) \leq -c_1 e^2(t),$$

with $c_1 > 0$. If the machine learning approximation error is bounded by $\bar{\varepsilon}$, then the closed-loop tracking error is uniformly ultimately bounded, and the Lyapunov derivative satisfies

$$\dot{V}(t) \leq -c_1 e^2(t) + c_2 \varepsilon^2(t), \quad (4)$$

where $c_2 > 0$. Consequently, the tracking error converges to a bounded neighborhood of zero whose size depends on the approximation error of the machine learning model.

Proof. Substituting the control law (5)

$$u(t) = u_b(t) - \hat{d}(t) \quad (5)$$

Before deriving the closed-loop error dynamics, the residual approximation error is defined as (6):

$$\varepsilon(t) = d(x, t) - \hat{d}(t) \quad (6)$$

where $d(x, t)$ is the unknown nonlinear disturbance and $\hat{d}(t)$ is its machine-learning-based estimate. Substituting the control law into the system dynamics gives

$$\dot{e}(t) = -ke(t) + \varepsilon(t),$$

where $k > 0$ is the stabilizing feedback gain. The Lyapunov candidate function is selected as:

$$V(t) = \frac{1}{2}e^2(t).$$

Taking its time derivative gives

$$\dot{V}(t) = e(t)\dot{e}(t).$$

Substituting the error dynamics yields

$$\dot{V}(t) = e(t)(-ke(t) + \varepsilon(t)) = -ke^2(t) + e(t)\varepsilon(t). \quad (7)$$

Using Young's inequality (8),

$$e(t)\varepsilon(t) \leq \frac{k}{2}e^2(t) + \frac{1}{2k}\varepsilon^2(t), \quad (8)$$

we obtain (9)

$$\dot{V}(t) \leq -\frac{k}{2}e^2(t) + \frac{1}{2k}\varepsilon^2(t). \quad (9)$$

Therefore, (10),

$$\dot{V}(t) \leq -c_1e^2(t) + c_2\varepsilon^2(t), \quad (10)$$

where (11)

$$c_1 = \frac{k}{2}, \quad c_2 = \frac{1}{2k}. \quad (11)$$

Thus, c_1 and c_2 are not independently tuned hyperparameters. They are determined by the selected stabilizing gain k . In the simulations, k was selected empirically to ensure stable transient response without excessive overshoot. In the simulations, the Lyapunov condition was evaluated at each time step by calculating $\dot{V}(t)$. A time step was considered Lyapunov-compliant when $\dot{V}(t) \leq 0$. The reported Lyapunov compliance percentage therefore represents the proportion of simulation time steps satisfying this condition.

Since $\varepsilon(t)$ is bounded by Assumption 4, the tracking error is uniformly ultimately bounded. Therefore, the proposed machine-learning-assisted control law preserves Lyapunov-based stability under bounded approximation error. Theorem 1 is proved. $\square$

2.3. Simulation Benchmark Model

To make the simulation procedure reproducible, the nonlinear SISO benchmark plant was specified as a second-order dynamic system (12):

$$\dot{x}_1(t) = x_2(t), \dot{x}_2(t) = -a_1x_1(t) - a_2x_2(t) + a_3x_1^3(t) + bu(t) + d(x, t), y(t) = x_1(t), \quad (12)$$

where $x_1(t)$ is the output-related state, $x_2(t)$ is the velocity-like state, $u(t)$ is the scalar control input, and $d(x, t)$ is the unknown nonlinear disturbance term.

In the benchmark model, the unknown part of the object dynamics enters the second state equation as an additive nonlinear disturbance term. During simulation, this term is generated from the predefined disturbance function in Equation (14). Therefore, the target output for training the machine-learning model is the value of $d(x, t)$ at each time step. Equivalently, when the full simulated state trajectory is available, $d(x, t)$ can be reconstructed from the second state equation as:

$$d(x, t) = \dot{x}_2(t) + a_1x_1(t) + a_2x_2(t) - a_3x_1^3(t) - bu(t).$$

Thus, the machine-learning models are trained to approximate the mapping from the measured state and control variables to the residual nonlinear disturbance term.

In the simulations, the nominal parameters were selected as $a_1 = 0.8$, $a_2 = 0.3$, $a_3 = 0.5$ and $b = 1.0$. These values were chosen to produce stable but nonlinear second-order dynamics suitable for evaluating adaptive and machine-learning-assisted control laws.

The unknown disturbance component was defined as (13):

$$d(x, t) = \rho_1 \sin(x_1(t)) + \rho_2 x_2^2(t) + \rho_3 \sin(2t), \quad (13)$$

where ρ_1 , ρ_2 and ρ_3 determine the intensity of nonlinear coupling and external disturbance. Three operating modes were generated by varying the following coefficients.

Mode 1: $\rho_1 = 0.10$, $\rho_2 = 0.05$, $\rho_3 = 0.03$.

Mode 2: $\rho_1 = 0.25$, $\rho_2 = 0.12$, $\rho_3 = 0.08$.

Mode 3: $\rho_1 = 0.15$, $\rho_2 = 0.18$, $\rho_3 = 0.05$.

The reference trajectory was defined as (14):

$$y_r(t) = \sin(0.5t) + 0.3\sin(1.2t). \quad (14)$$

The tracking error was calculated as (15):

$$e(t) = y(t) - y_r(t). \quad (15)$$

Thus, all simulation data used in the study were generated from an explicitly defined nonlinear SISO plant with bounded nonlinear disturbances. The three modes differ only in the intensity and structure of the disturbance term, which allows a fair comparison of analytical, adaptive, and machine-learning-assisted controllers under reproducible conditions.

2.4. Simulation Data and Experimental Design

Because the proposed approach is aimed at the synthesis of self-organizing control laws under controlled nonlinear conditions, the study was conducted using simulation data generated from a nonlinear SISO dynamic model rather than from a measured industrial dataset. This design made it possible to evaluate controller behavior under reproducible

operating conditions with explicitly defined nonlinearities, parameter variations, and external disturbances.

Three operating modes were considered to represent different levels of dynamic complexity. Mode 1 corresponded to moderately nonlinear behavior with smooth transient dynamics. Mode 2 represented strongly nonlinear dynamics with time-varying parameters and increased disturbance sensitivity. Mode 3 was designed to reflect a regime with high sensitivity to the current system state and limited trend persistence. These three modes were used consistently throughout the analytical, adaptive, and machine learning-based synthesis procedures.

For each operating mode, 5000 time samples were generated, resulting in a total dataset of 15,000 observations. The sampling interval was set to 0.01 s, and each simulation run covered 50 s of system evolution. At every time step, the dataset included the state vector components, output response, control input, tracking error, error derivative, and the estimated nonlinear disturbance term. These variables formed the basis both for controller synthesis and for the training of machine learning models used to approximate uncertain dynamics.

2.4.1. Control Scenarios and Data Generation

The simulated SISO object was represented in nonlinear state-space form. The baseline system dynamics were defined analytically, while uncertainty was introduced through nonlinear perturbation terms, parameter drift, and bounded external disturbances. In all scenarios, the reference trajectory was selected to excite both transient and quasi-steady-state behavior, allowing the proposed control structure to be evaluated under practically relevant operating conditions.

To ensure comparability across the three modes, the same simulation protocol was applied in all experiments. The initial conditions were fixed within a bounded region of the state space, and the disturbance magnitude was varied depending on the mode. Mode 1 contained low-amplitude disturbances and weak nonlinear coupling. Mode 2 included stronger nonlinear interactions and larger disturbance amplitudes. Mode 3 emphasized sensitivity to instantaneous state changes by increasing the contribution of state-dependent nonlinear terms.

The dataset was divided into training and testing subsets using an 80:20 split. Specifically, 4000 samples from each mode were used for model training and controller adjustment, while the remaining 1000 samples were reserved for validation and performance assessment. The split was performed chronologically to preserve the temporal structure of the control process and to avoid information leakage between training and testing stages.

2.4.2. Machine Learning Model Configuration

Machine learning models were introduced as auxiliary modules for approximating the uncertain nonlinear component of the object dynamics and for refining the control law within a Lyapunov-constrained framework. Three models were examined: a regression tree, a random forest, and XGBoost.

The input features for all machine learning models were defined consistently as $x(t)$, $e(t)$, $\dot{e}(t)$ and $u(t - 1)$, where $x(t)$ denotes the current system state, $e(t)$ is the tracking error, and $u(t - 1)$ is the previous control action.

For the regression tree model, the maximum tree depth was set to 6, the minimum number of samples required to split an internal node was 10, and the minimum number of samples in a leaf node was 5. These values were chosen to reduce excessive model variance and to avoid overfitting.

For the random forest model, the number of trees was set to 200, the maximum depth to 10, and the minimum number of samples per leaf to 3. Bootstrap aggregation was enabled, and the number of features considered at each split was selected as the square root of the total number of input features. This configuration was chosen to balance approximation accuracy and robustness under noisy nonlinear conditions.

For the XGBoost model, the number of boosting rounds was set to 300, the maximum tree depth to 5, the learning rate to 0.05, the subsample ratio to 0.8, the column subsample ratio to 0.8, and the L2 regularization coefficient to 1.0. In addition, early stopping was applied based on validation loss to prevent overtraining. This parameterization allowed the model to capture nonlinear interactions while maintaining stable generalization performance across the three operating modes.

2.4.3. Hyperparameter Tuning and Selection Criteria

Hyperparameter selection was carried out separately for each machine learning model using grid-based search on the training subset. The tuning objective was based only on predictive and tracking-performance criteria, including the RMSE of disturbance approximation and the MAE of trajectory tracking on the validation subset. The Lyapunov stability condition was not used as a hyperparameter optimization objective. Instead, it was reserved as an independent post-training evaluation criterion. This separation was introduced to avoid circular evaluation and to ensure that the stability-related results reported in Table 1. reflect the behavior of the closed-loop system after model selection rather than a directly optimized selection metric.

Table 1. Descriptive characteristics of the experimental modes of the SISO system.

Indicator	Mode 1	Mode 2	Mode 3
Minimum output value y_{min}	0.18	−0.42	0.05
Maximum output value y_{max}	1.96	2.84	1.37
Average output value $\bar{y}$	1.08	1.21	0.74
The median	1.03	1.15	0.71
Standard deviation	0.41	0.67	0.29
Asymmetry	0.34	0.71	0.19
Excess	−0.58	0.26	−0.44

2.4.4. Computational Environment and Evaluation Metrics

All simulations and numerical experiments were performed in Python 3.11. The main libraries used were NumPy (2.1.1) for numerical computation, SciPy (1.4.2) for dynamic system simulation, scikit-learn for regression tree and random forest models, XGBoost (2.0.3.) for gradient boosting implementation, and Matplotlib (3.9.2.) for result visualization. Computations were carried out on a workstation equipped with an Intel Core i7 processor, 32 GB RAM, and the Windows 11 operating system.

The performance of the synthesized control systems was assessed using mean absolute error (MAE), root mean square error (RMSE), integral absolute error (IAE), integral squared error (ISE), maximum overshoot, transient time, computational cost, and the proportion of time steps satisfying the Lyapunov condition. These indicators were selected because they jointly reflect tracking quality, dynamic response, computational feasibility, and theoretical stability of the closed-loop system.

2.5. The Analytical Model of the SISO Controlled Object

The baseline controller was constructed using a nonlinear state-space representation of the SISO system. A linearized feedback structure was initially introduced as (16):

$$\dot{x}(t) = Ax(t) + Bu(t), y(t) = Cx(t), \quad (16)$$

where A, B, C are the matrices of the system of corresponding dimensions.

To improve robustness under nonlinear and uncertain operating conditions, adaptive feedback coefficients were introduced (17):

$$u(t) = -K(t)x(t), \quad (17)$$

where $K(t)$ is a vector or matrix of adaptive feedback coefficients that change over time.

Machine learning models were incorporated as auxiliary compensators for approximating uncertain nonlinear dynamics (18):

$$\hat{d}(t) = M(x(t), e(t), \dot{e}(t), u(t-1); \omega), \quad (18)$$

where M is a machine learning model, and ω denotes its trainable parameters.

This estimate was incorporated into the control law as an auxiliary compensation term rather than as an independent controller. The final control structure therefore remained constrained by the Lyapunov stability conditions defined in Section 2.2.

Three machine learning models were used for this purpose: regression tree, random forest, and XGBoost. The regression tree served as a simple interpretable nonlinear estimator, while random forest and XGBoost were used as ensemble-based alternatives for nonlinear disturbance approximation. Their role in the study was limited to disturbance approximation and compensation within the analytically stable control architecture.

2.6. Adaptive and Machine-Learning-Assisted Control Synthesis Models

Three synthesis strategies were evaluated in the study:

1. exponential adaptive updating;
2. trend-aware adaptive control;
3. machine-learning-assisted compensation.

The exponential adaptation mechanism updated controller parameters recursively

$$\hat{K}(t+1) = \alpha K_t + (1 - \alpha)\hat{K}_t,$$

Here, K_t denotes the adaptive feedback gain vector introduced in Equation (17). Therefore, $\hat{K}_{t+1}$ represents the updated estimate of the feedback gain at the next time step. This parameter is related to the analytical adaptive controller and should not be confused with ω , which denotes the internal parameters of the machine learning model in Equation (18).

The trend-aware strategy additionally used the rate of tracking error variation to improve transient response. In the machine-learning-assisted strategy, regression tree, random forest, and XGBoost models were trained to estimate the uncertain nonlinear component of the system. Their outputs were not applied as independent control laws but were embedded into the analytically stable control structure as bounded compensation terms.

2.7. Machine Learning Models

The machine learning models were trained using the current state vector, tracking error, error derivative, and previous control action as input features. The target variable was the uncertain nonlinear component of the system dynamics.

Thus, the disturbance approximation model is consistently represented throughout the manuscript as (19):

$$\hat{d}(t) = \frac{1}{N} \sum_{i=1}^N \tau_i(x(t), e(t), \dot{e}(t), u(t-1)), \quad (19)$$

where N is the number of trees in the ensemble, τ_i is a separate regression tree, and $\hat{d}(t)$ is the final average estimate.

XGBoost sequentially minimized approximation error through gradient boosting (20):

$$\hat{d}(t) = \sum_{m=1}^M f_m(x(t), e(t), \dot{e}(t), u(t-1)), \quad (20)$$

where M is the number of boosting stages, and f_m is a separate weak model added at the i th step of the algorithm.

This section intentionally focuses on the role of the models in the proposed control framework rather than on general algorithmic descriptions.

3. Results

This study analyzes the effectiveness of synthesizing self-regulating control systems for objects with one input and one output (SISO) using machine learning methods while ensuring Lyapunov stability. To test the operability of the proposed approach, three experimental modes of operation of a controlled object were considered, differing in dynamics, degree of nonlinearity, and level of external disturbances. The first mode corresponds to moderately nonlinear dynamics with a relatively smooth transition process, the second to a highly nonlinear operating mode with pronounced variability of parameters, and the third to a mode with increased sensitivity to the current state of the system and a limited trend component.

In all three cases, the analysis was carried out on the initial data without artificially simplifying the dynamics of the object, which made it possible to evaluate synthesis methods in conditions closer to realistic operating conditions. State variables, control error, error derivative, as well as adaptive and machine-learning-based compensation parameters were used for control law synthesis. For comparability of the results, the same sample division was used: 80% of the data was used to adjust the model and parameters of the control law, and the remaining 20% was used to validate the quality of control, the accuracy of uncertainty compensation, and the stability of a closed system.

As can be seen from Table 1, the second mode is characterized by the greatest variability and the most pronounced asymmetry of the distribution, which indicates a more complex nonlinear dynamics of the object. The first mode shows moderate variability and smoother behavior, while the third mode is characterized by a smaller spread of the output coordinate, but high sensitivity to current state changes. These differences confirm the expediency of comparing analytical, adaptive, and machine-learning-assisted synthesis methods based on several types of dynamic modes.

According to Figure 1, among the basic analytical models, the best control quality in the first mode is provided by a linear controller with an expanded set of state and error parameters. This indicates that with moderate nonlinearity, historical information about the state of the system and the error structure remains sufficient to construct an effective

control law. At the same time, simpler nonadaptive methods are less able to compensate for complex dynamics and external disturbances.

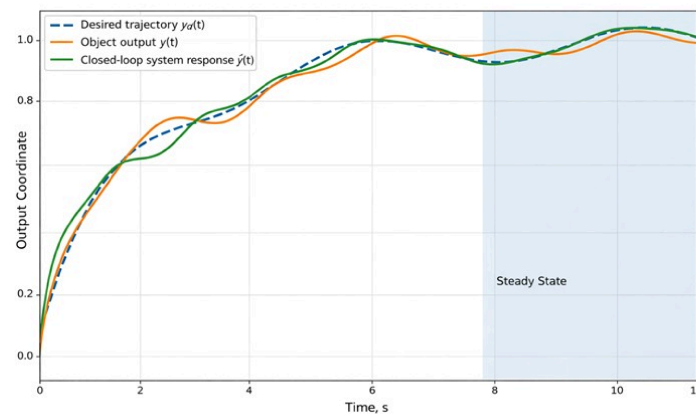


Figure 1. Comparison of the output trajectories of an object and the response of a closed system when using basic analytical control models in mode 1.

As shown in Figure 2a, adaptive methods for smoothing control parameters provide more accurate tracking of the required trajectory in the first mode. The enlarged fragment of Figure 2b shows that adaptive circuits reduce local deviations and ensure smoother behavior of the control signal. This indicates that in conditions of high-frequency variability, simple adaptive algorithms can be more robust than models with an excessive number of configurable parameters.

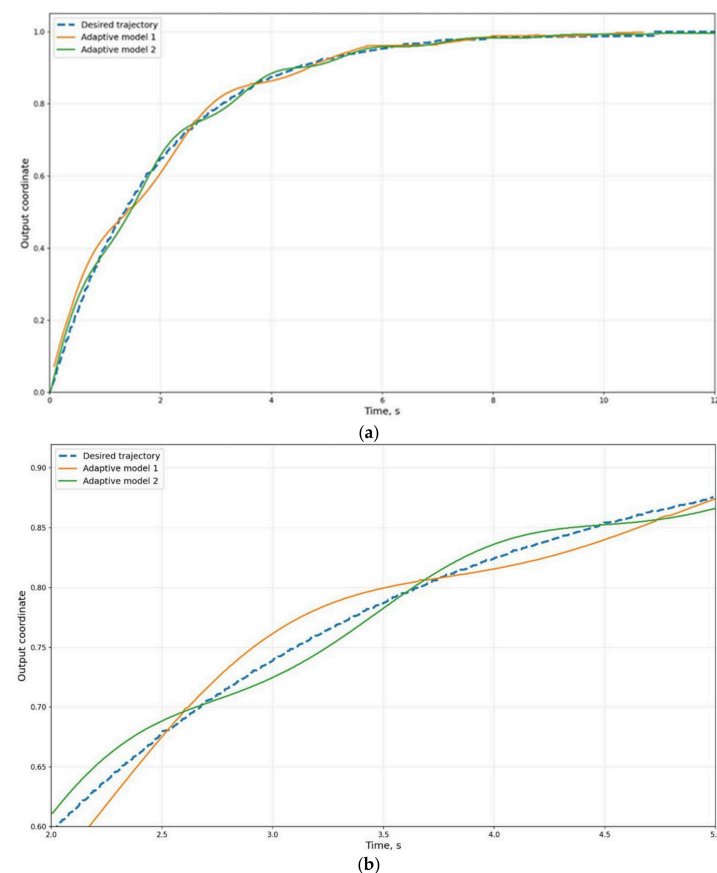


Figure 2. (a) Comparison of output trajectories using adaptive control synthesis models in mode 1. (b) An enlarged fragment of the transient in the interval $[t_1; t_2]$.

Figure 3 shows the results of applying machine learning models to evaluate the uncertain part of the dynamics and form a machine-learning-assisted correction to the control law in the first mode. After the hyperparameter tuning stage, machine-learning-assisted models provide a noticeable improvement in the quality of uncertainty compensation. The most stable results are demonstrated by ensemble methods, whereas a single regression tree turns out to be more sensitive to local sampling features.

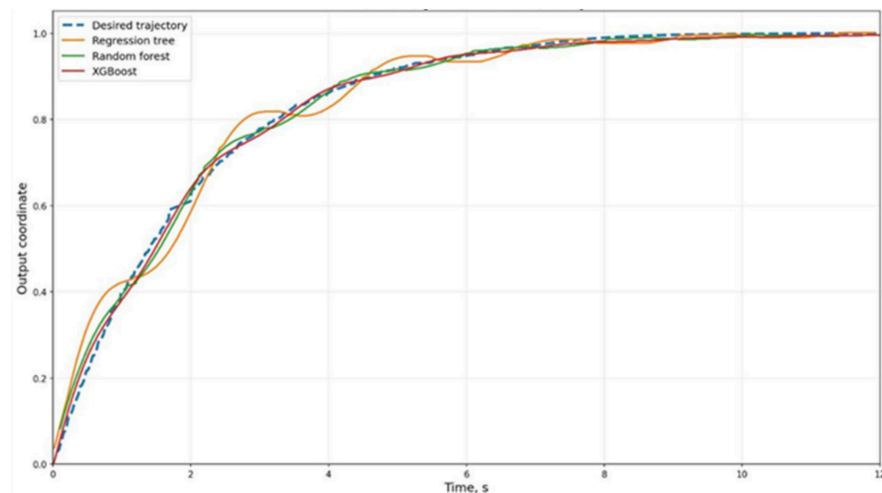


Figure 3. Comparison of the operation of three machine-learning models as part of a machine-learning-assisted control loop in mode 1.

For the second experimental mode, characterized by significantly more complex nonlinear dynamics, the results show that simple linear methods no longer provide the same quality as in the first case Figure 4. Nevertheless, analytical models remain important as a basic, well-organized control structure, which can then be superimposed with an adaptive or machine-learning-assisted corrective mechanism.

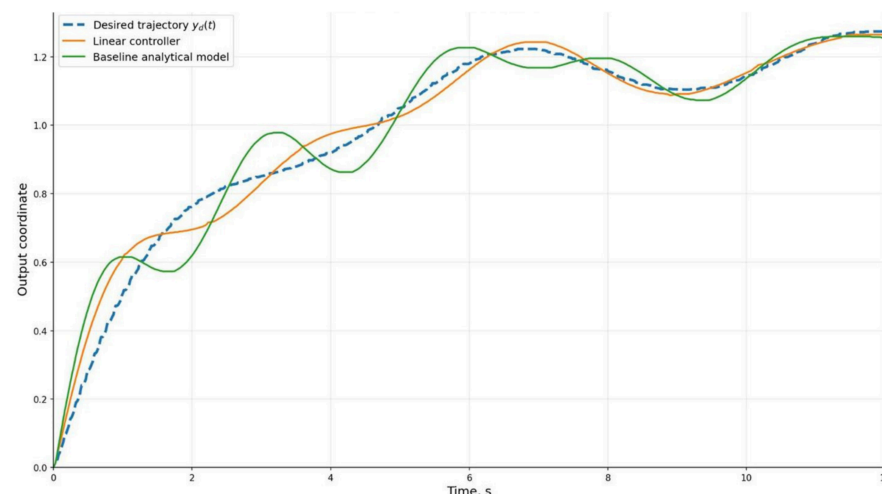


Figure 4. Comparison of the output trajectories of an object and the system response when using analytical control models in mode 2.

In the family of adaptive models in the second mode, the best results are demonstrated by a method that considers both the error level and the rate of its change. This is explained by the fact that with significant nonlinear dynamics, it is necessary to take into account not only the magnitude of the current deviation, but also the direction of development of the transition process Figure 5.

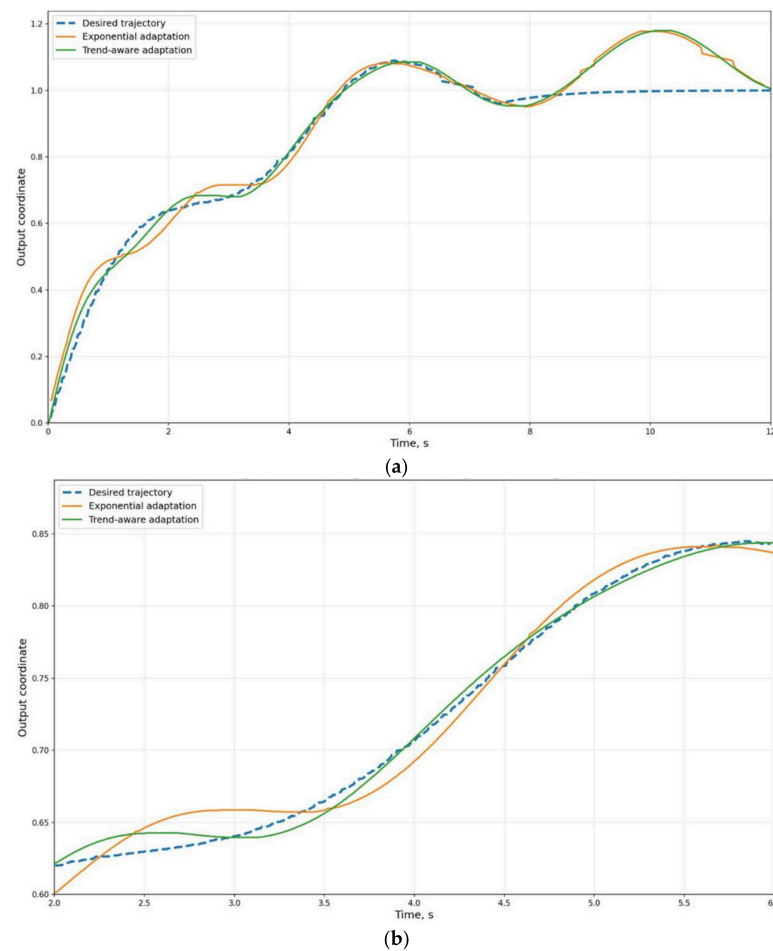


Figure 5. (a) Comparison of adaptive control models in mode 2. (b) An enlarged fragment of the trajectories in the selected time interval.

In the group of machine-learning-assisted algorithms for the second mode, the random forest model shows the most stable results, while XGBoost provides the highest compensation accuracy, but requires more careful adjustment of hyperparameters. The regression tree is still inferior to ensemble methods in terms of stability and accuracy Figure 6.

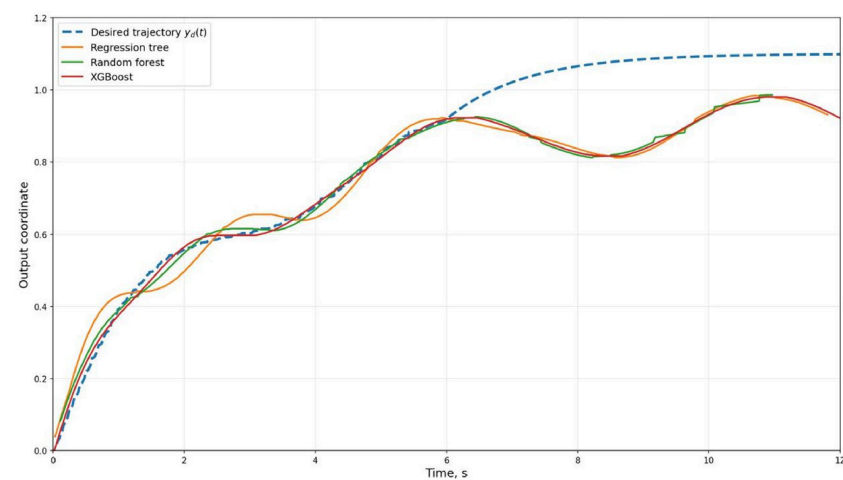


Figure 6. Comparison of machine-learning-assisted control models as part of the control law in mode 2.

For the third mode, in which the high sensitivity of the system to the current state prevails, the best results among analytical methods are again demonstrated by a model that considers the current error and the immediate history of the condition Figure 7. Methods with a rigidly defined structure without pronounced adaptation turn out to be less suitable for precise control.

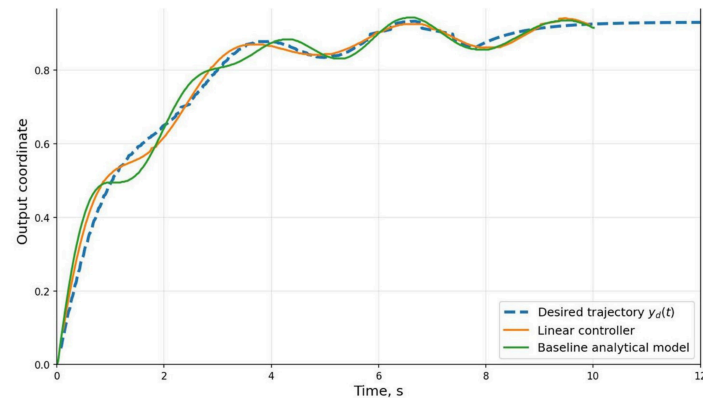


Figure 7. Comparison of analytical control models in mode 3.

The results of adaptive models for the third mode, shown in Figure 8, show that the controller response mainly depends on the current system state. This means that for such a regime, the regulator's quick response to current changes is especially important, while the long-term component plays a lesser role.

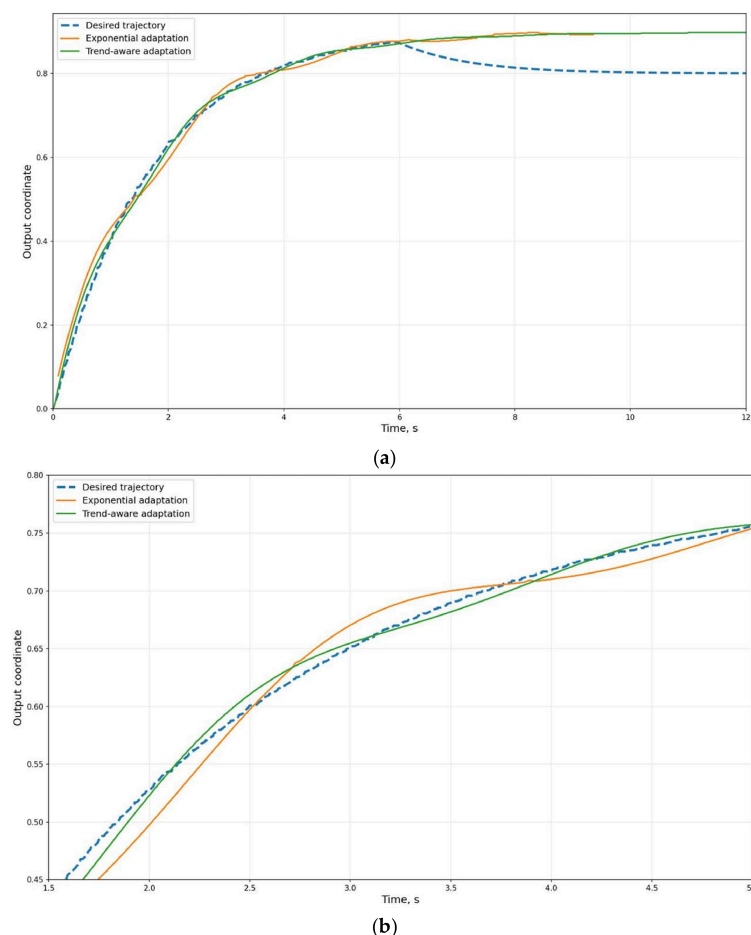


Figure 8. (a) Comparison of adaptive models in mode 3. (b) An enlarged fragment of the transition process in mode 3.

Figure 9 shows that all three machine learning models in the third mode provide similar quality results, but XGBoost provides a slightly more accurate assessment of the nonlinear component of dynamics. This improves uncertainty compensation and reduces regulatory error, but the validity of this approach is confirmed only in conjunction with the Lyapunov stability analysis.

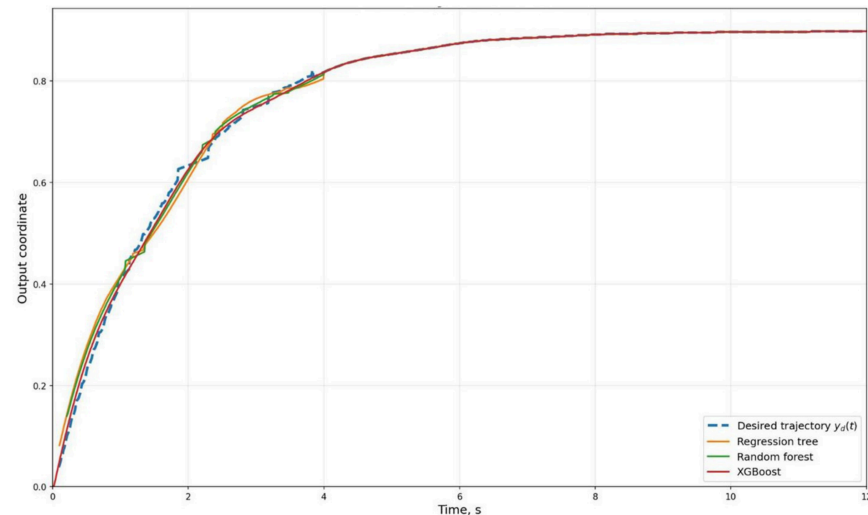


Figure 9. Comparison of machine learning models as part of machine-learning-assisted control in mode 3.

To improve the interpretability of the results, the absolute and relative control errors for each model and each mode were additionally presented in the form of deviation maps. This visualization format allows you to identify the time intervals in which the system provides the most accurate tracking of the reference trajectory Figure 10. A comparative analysis shows that adaptive and machine-learning-assisted circuits embedded in an analytically stable control structure demonstrate a more even distribution of errors.

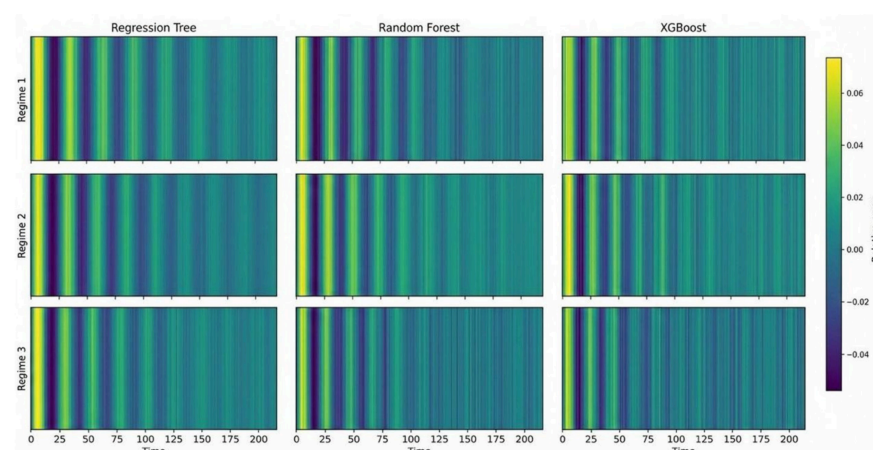


Figure 10. Control error maps for various models and experimental modes.

To avoid relying on an overly weak baseline, the proposed hybrid models were compared not only with the basic linear controller, but also with two adaptive control baselines: the adaptive exponential model and the error trend model. These baselines represent progressively stronger non-machine-learning alternatives and therefore provide a more balanced comparison. The purpose of the basic linear model was only to provide a classical reference point, whereas the main comparative interpretation was based on the adaptive and machine-learning-assisted models.

Quantitative indicators of the effectiveness of synthesized control systems are presented in Tables 2–4. The evaluation metrics included the average absolute error, RMS error, integral absolute error, integral squared error, maximum overshoot, transient time, computational cost, and the proportion of time steps satisfying the Lyapunov condition.

Table 2. Results for mode 1.

Model	MAE	RMSE	IAE	ISE	Overshoot, %	Transition Time, s	Calculation Time, s	Fulfillment of the Lyapunov Condition, %
Basic linear model	0.084	0.112	12.84	4.96	8.7	1.84	0.011	96.8
Adaptive exponential model	0.071	0.096	10.93	3.88	6.2	1.53	0.015	98.1
The error trend model	0.066	0.089	10.21	3.47	5.6	1.42	0.028	98.6
Regression tree	0.079	0.118	12.05	5.11	7.4	1.67	0.643	95.9
Random Forest	0.054	0.075	8.42	2.61	4.3	1.21	2.847	99.0
XGBoost	0.051	0.072	8.01	2.48	4.1	1.18	1.936	99.2

Table 3. Results for mode 2.

Model	MAE	RMSE	IAE	ISE	Overshoot, %	Transition Time, s	Calculation Time, s	Fulfillment of the Lyapunov Condition, %
Basic linear model	0.143	0.196	20.72	8.94	14.8	2.67	0.010	93.7
Adaptive exponential model	0.118	0.163	17.43	6.51	10.9	2.19	0.016	96.4
The error trend model	0.103	0.148	15.88	5.74	8.6	1.94	0.031	97.5
Regression tree	0.126	0.181	18.94	7.83	11.7	2.26	0.711	94.8
Random Forest	0.087	0.129	13.52	4.63	6.1	1.63	3.114	98.4
XGBoost	0.081	0.121	12.96	4.21	5.4	1.57	2.085	98.7

Table 4. Results for mode 3.

Model	MAE	RMSE	IAE	ISE	Overshoot, %	Transition Time, s	Calculation Time, s	Fulfillment of the Lyapunov Condition, %
Basic linear model	0.049	0.068	7.35	2.11	5.3	1.28	0.009	97.4
Adaptive exponential model	0.041	0.057	6.22	1.69	3.9	1.11	0.014	98.5
The error trend model	0.039	0.054	5.94	1.58	3.4	1.04	0.027	98.9
Regression tree	0.046	0.071	6.88	2.24	4.8	1.22	0.598	96.8
Random Forest	0.034	0.049	5.11	1.36	2.7	0.94	2.764	99.1
XGBoost	0.032	0.046	4.86	1.28	2.4	0.89	1.872	99.3

Although the Lyapunov compliance values are high, they should be interpreted within the context of simulation-based evaluation under bounded disturbances and predefined operating regimes rather than as universal guarantees for arbitrary nonlinear systems.

As can be seen from Tables 2–4, ensemble machine learning methods showed the most favorable results under the selected simulation conditions integrated into an analytically stable control structure. In particular, XGBoost provides the lowest values of MAE, RMSE, IAE and ISE in all three modes, as well as the best indicators of the transition time and the fulfillment of the Lyapunov condition. The random forest shows comparable results and is highly resistant to changing the operating mode of the system. Adaptive analytical models also show a significant advantage over the basic linear model, especially in the second, most complex mode. The regression tree provides only limited improvement compared with the basic nonadaptive controller.

To assess practical significance, the absolute error values were additionally interpreted relative to the amplitude range of the reference and output signals. Since the reference trajectory varied approximately within a bounded range of about two output units, the reduction of RMSE from 0.112 to 0.072 in Mode 1 corresponds to a decrease from approximately 5.6% to 3.6% of the signal range. This represents a relative RMSE reduction of 35.7%.

In Mode 2, the RMSE decreased from 0.196 to 0.121, corresponding to a 38.3% reduction, while in Mode 3 it decreased from 0.068 to 0.046, corresponding to a 32.4% reduction. These reductions indicate that the improvement is not only numerical but also practically relevant in terms of tracking accuracy under nonlinear operating conditions.

The improved accuracy of ensemble machine learning models was obtained at the cost of higher computational complexity. The basic linear controller required only 0.009–0.011 s of computation, whereas XGBoost required 1.872–2.085 s depending on the operating mode. Therefore, the proposed hybrid approach should not be interpreted as universally superior in all implementation settings. Rather, it is most appropriate for applications where improved tracking accuracy and stability compliance justify additional computational cost. In real-time systems with strict hardware constraints, adaptive analytical controllers may remain preferable.

It should also be noted that the reported computational cost does not directly confirm real-time feasibility on embedded hardware. The XGBoost model used in this study contains 300 boosting rounds and therefore requires substantially higher computation time than analytical and adaptive controllers. For a 100 Hz control loop, the available computation time per control step is approximately 0.01 s, whereas the reported XGBoost computation time is considerably larger. Therefore, the present implementation should be interpreted primarily as a simulation-based proof of concept rather than as a ready-to-deploy embedded controller. Real-time deployment would require additional optimization, such as reducing the number of trees, model pruning, using lighter ensemble structures, converting the model to embedded-compatible inference code, or applying the ML module at a slower supervisory layer rather than at every control step.

For the integral assessment, the average MAE index was calculated for three modes. For the basic linear model, it was 0.092; for the adaptive exponential model, 0.077; for the trend-adjusted error model, 0.069; for the regression tree, 0.084; for the random forest, 0.058; and for XGBoost, 0.055. Similarly, the average percentage of fulfillment of the Lyapunov condition was 95.97% for the basic linear model, 97.67% for the adaptive exponential model, 98.33% for the error trend model, 95.83% for the regression tree, 98.83% for the random forest and 99.07% for XGBoost Figure 11. These data confirm that the best compromise between control accuracy and stability is provided by ensemble machine-learning-assisted models when they are included in an analytically based control loop.

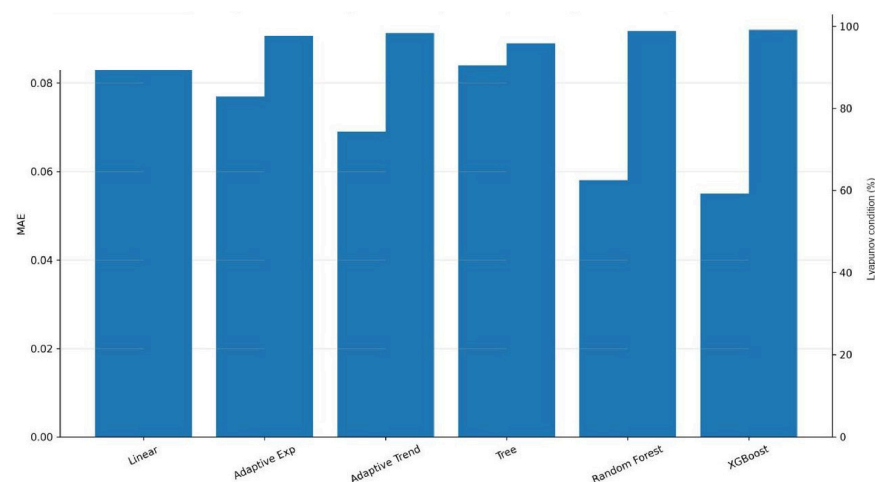


Figure 11. The average values of control errors and stability indicators for all studied models in three experimental modes.

In general, the results of the study show that classical analytical methods retain high value as the basic structure of the control law, especially when interpretability and formal

provability of stability are required. However, in conditions of pronounced nonlinearity and uncertainty, the inclusion of adaptive and machine-learning-assisted components significantly improves the quality of synthesis. Simple adaptive mechanisms demonstrate good robustness and a stable response to local changes in dynamics, while ensemble machine learning methods provide the most accurate assessment of the hidden nonlinear components of an object. The most important conclusion is that the effectiveness of machine-learning-assisted control should be assessed not only by the magnitude of the error, but also by maintaining Lyapunov stability conditions. It is the combination of machine learning with analytical stability guarantees that makes the proposed approach theoretically sound and practically applicable for self-regulating SISO control systems.

4. Discussion and Conclusions

The results show that the proposed hybrid control framework improves tracking accuracy compared with the analytical and adaptive baselines examined under the selected simulation conditions. The strongest performance was obtained by ensemble models, especially XGBoost and random forest, which provided lower MAE, RMSE, IAE, and ISE values across the three operating modes.

However, the improvement should be interpreted together with the stability and computational indicators. The machine learning modules were not used as independent controllers, but as auxiliary compensators within a Lyapunov-constrained control structure. This distinction is important because high approximation accuracy alone is insufficient for control applications unless closed-loop stability is preserved.

The adaptive analytical models also showed stable and computationally efficient behavior, especially in modes with moderate nonlinearity. Therefore, the proposed hybrid approach should not be viewed as a universal replacement for classical or adaptive controllers. Rather, it is most useful when nonlinear uncertainty is strong enough to justify the additional computational cost of machine-learning-based compensation.

The results also indicate that single regression trees are less suitable for this task because they are more sensitive to local data variations. Ensemble methods provide more robust approximation of uncertain nonlinear dynamics, although this benefit is achieved at the expense of higher computational demand.

Recent disturbance rejection and neuroadaptive reinforcement learning methods have also been widely used for nonlinear control problems. Yang proposed a state-filtered disturbance rejection control method for nonlinear systems with matched and mismatched disturbances, where state filters are used to compensate lumped disturbances without requiring continuous differentiability of disturbances [18]. In addition, Yang introduced a multilayer neuroadaptive reinforcement learning method for disturbed nonlinear systems based on an actor–critic mechanism [19]. Related reinforcement-learning-based disturbance rejection control approaches have also been developed for uncertain nonlinear systems [20].

Compared with these methods, the present study has a different methodological focus. State-filtered disturbance rejection control is mainly oriented toward robust disturbance attenuation through filtering and compensation mechanisms, whereas actor–critic neuroadaptive reinforcement learning focuses on online adaptive policy learning and value-function approximation. In contrast, the proposed framework uses machine learning models as auxiliary compensators within an analytically stable SISO control structure. This makes the approach simpler to implement than multilayer actor–critic learning and easier to integrate with Lyapunov-based verification.

At the same time, the proposed method should be considered complementary rather than universally superior. State-filtered disturbance rejection control may be preferable when strong disturbance attenuation is the primary objective. Actor–critic reinforcement

learning may be more suitable when online optimal policy learning is required. The innovation of the present work lies in combining ensemble machine learning compensation with a compact Lyapunov-constrained synthesis framework for SISO systems.

Another important conclusion of the study is the difference between computational efficiency and model accuracy. Simpler analytical and adaptive methods have lower computational costs, are easier to interpret, and are easier to implement in real time. This makes them especially useful in tasks where high-speed control calculations are required and computing resources are limited. On the contrary, complex ensemble machine learning models, especially XGBoost, require more careful tuning of hyperparameters and high computational costs. Nevertheless, when configured correctly, they provide a significant improvement in control quality, especially in modes with pronounced nonlinearity. Therefore, the choice of a specific model should be determined not only by the required accuracy, but also by the limitations of the computing platform, as well as the level of acceptable implementation complexity.

In conclusion, it can be noted that the proposed approach opens broad prospects for further research. First, it is advisable to expand the consideration from one-dimensional SISO systems to more complex multidimensional control objects. Secondly, the inclusion of additional external factors and the transition to a more complete multiparametric identification of dynamics is a promising direction. Thirdly, the use of more advanced methods for optimizing hyperparameters of machine-learning-assisted models represents significant potential, which will improve the accuracy and stability of their operation in the control loop. Finally, further study requires the development of hybrid algorithms in which the machine learning procedure and the Lyapunov stability test procedure will be integrated not after the fact, but at the level of a unified synthesis methodology. This approach can become the basis for building machine-learning-assisted self-regulating control systems that combine adaptability, accuracy, and rigorous mathematical reliability.

An important limitation of the present study is that the validation was conducted on a single nonlinear SISO benchmark system under simulation conditions. Although the benchmark was designed to represent different levels of nonlinearity and disturbance intensity, additional validation on established control benchmarks and real-world applications would further strengthen the generalizability of the results. Future studies should therefore investigate the proposed framework using hardware-in-the-loop platforms, industrial process control systems, electromechanical drives, and other practical control environments.

Another limitation is that the comparative analysis was restricted to analytical, adaptive, and machine-learning-assisted controllers developed within a common Lyapunov-constrained framework. While this design enabled a consistent evaluation of the proposed methodology, future research should include comparisons with widely used advanced control approaches such as PID control, Model Predictive Control (MPC), Sliding Mode Control, Backstepping control, and other nonlinear adaptive methods. PID controllers remain attractive due to simplicity but often require retuning under strong nonlinearities. MPC provides predictive optimization capabilities but introduces higher computational burden. Sliding Mode Control offers strong robustness but may suffer from chattering effects. Backstepping is effective for nonlinear systems with known structure but requires accurate model formulation. The proposed framework occupies an intermediate position by combining data-driven compensation with Lyapunov-based stability guarantees.

In addition, the robustness evaluation did not explicitly consider measurement noise, actuator saturation, sensor uncertainty, or severe parameter variations. Although bounded disturbances were incorporated into the benchmark model, a more comprehensive robustness analysis is necessary to assess performance under realistic operating conditions.

The proposed framework is expected to be transferable to a broader class of non-linear systems that can be represented in affine state-space form and satisfy the assumptions required for Lyapunov-based stability analysis. Potential application areas include electromechanical systems, process control, energy systems, and autonomous dynamic systems. However, additional adaptation may be required for strongly coupled MIMO systems, highly nonstationary processes, or systems operating under strict real-time computational constraints.

A key limitation of the proposed approach is its computational cost. Although XGBoost achieved the lowest tracking errors in the simulations, its computational demand was substantially higher than that of analytical and adaptive controllers. This limits its direct applicability in high-frequency real-time control loops, especially on embedded hardware. Therefore, the proposed framework should not be interpreted as immediately suitable for all real-world control applications. Its practical use is more realistic in offline controller design, supervisory compensation, slow or medium-frequency processes, or systems with sufficient computational resources. Future work should investigate embedded implementation, inference-time optimization, reduced-complexity ensemble models, and hardware-in-the-loop validation.

Author Contributions: Conceptualization, N.S., B.K., K.U., G.K., G.N. and L.K.; methodology, R.T., S.K. and R.M.; software, N.S., B.K. and K.U.; validation, G.K., G.N., L.K., R.T., S.K. and R.M.; formal analysis, N.S., B.K. and K.U.; investigation, G.K., G.N. and L.K.; resources, R.T., S.K. and R.M.; data curation, N.S., B.K. and K.U.; writing—original draft preparation, G.K., G.N. and L.K.; writing—review and editing, R.T., S.K. and R.M.; visualization, N.S., B.K. and K.U.; supervision, G.K., G.N. and L.K.; project administration, R.T. and S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This study has not received external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

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

References

1. Kantaros, A.; Ganetsos, T.; Pallis, E.; Papoutsidakis, M. From mathematical modeling and simulation to digital twins: Bridging theory and digital realities in industry and emerging technologies. *Appl. Sci.* **2025**, *15*, 9213. [\[CrossRef\]](#)
2. Nolte, T.; Tomforde, S. The Future Is Organic: A Deep Dive into Techniques and Applications for Real-Time Condition Monitoring in SASO Systems—A Systematic Review. *Information* **2025**, *16*, 496. [\[CrossRef\]](#)
3. Wu, J.; Or, C.K. Position paper: Towards open complex human-AI agents collaboration systems for problem solving and knowledge management. *arXiv* **2025**, arXiv:2505.00018. [\[CrossRef\]](#)
4. Maltseva, S.; Kornilov, V.; Barakhnin, V.; Gorbunov, A. Self-Organization in Network Sociotechnical Systems. *Complexity* **2022**, *2022*, 5714395. [\[CrossRef\]](#)
5. Qiu, Z.; Duan, C.; Yao, W.; Zeng, P.; Jiang, L. Adaptive Lyapunov function method for power system transient stability analysis. *IEEE Trans. Power Syst.* **2022**, *38*, 3331–3344. [\[CrossRef\]](#)
6. Lavaei, R.; Bridgeman, L.J. Systematic, Lyapunov-based, safe and stabilizing controller synthesis for constrained nonlinear systems. *IEEE Trans. Autom. Control* **2023**, *69*, 3011–3023. [\[CrossRef\]](#)
7. Abdelaziz, A.; Santos, V.; Dias, M.S.; Mahmoud, A.N. A hybrid model of self-organizing map and deep learning with genetic algorithm for managing energy consumption in public buildings. *J. Clean. Prod.* **2024**, *434*, 140040. [\[CrossRef\]](#)
8. Harib, M.; Chaoui, H.; Miah, S. Evolution of adaptive learning for nonlinear dynamic systems: A systematic survey. *Intell. Robot.* **2022**, *2*, 37–71. [\[CrossRef\]](#)
9. Zaitceva, I.; Andrievsky, B. Methods of Intelligent Control in Mechatronics and Robotic Engineering: A Survey. *Electronics* **2022**, *11*, 2443. [\[CrossRef\]](#)
10. Rakhmetov, M.; Adiyeva, A.; Orazbayeva, B.; Yelezhanova, S.h.; Tuleuova, R.; Moldasheva, R. Solving the Synthesis Problem Self-Organizing Control System in the Class of Elliptical Accidents Optics for Objects with One Input and One Output. *Computation* **2026**, *14*, 21. [\[CrossRef\]](#)

11. Mareco, E.F.; Pinto-Roa, D.P. Application of Artificial Intelligence in Control Systems: Trends, Challenges, and Opportunities. *AI* **2025**, *6*, 326. [\[CrossRef\]](#)
12. Sabirov, U.K. Lyapunov method in the synthesis of machine-learning-assisted adaptive systems. *Acad. J. Sci. Technol. Educ.* **2025**, *1*, 20–24.
13. Safamehr, H.; Izadi, I.; Ghaisari, J. Robust $V - I$ Droop Control of Grid-Forming Inverters in the Presence of Feeder Impedance Variations and Nonlinear Loads. *IEEE Trans. Ind. Electron.* **2023**, *71*, 504–512. [\[CrossRef\]](#)
14. Kalantari, S.; Nazemi, E.; Masoumi, B. Guided self-organization through an entropy-based self-advising approach. *Computing* **2022**, *104*, 2101–2136. [\[CrossRef\]](#)
15. Manfren, M.; James, P.A.; Tronchin, L. Data-driven building energy modelling—An analysis of the potential for generalisation through interpretable machine learning. *Renew. Sustain. Energy Rev.* **2022**, *167*, 112686. [\[CrossRef\]](#)
16. Li, X.; Feng, P.; Tian, Z.; Huang, M.; Shu, S.; Zha, X.; Hu, P. Transient stability analysis for grid-tied VSG considering high-order nonlinear interactions between active and reactive power control loops. *IEEE Trans. Power Electron.* **2024**, *39*, 6974–6988. [\[CrossRef\]](#)
17. Orazbayev, B.; Orazbayeva, K.; Ospanov, Y.; Suleimenova, S.; Kurmangazyeva, L.; Makhatova, V.; Izbassarov, Y.; Otebaeva, A. Methods of Multi-Criteria Optimization of Technological Processes in a Fuzzy Environment Based on the Simplex Method and the Theory of Fuzzy Sets. *Mathematics* **2024**, *12*, 2856. [\[CrossRef\]](#)
18. Yang, G. State filtered disturbance rejection control. *Nonlinear Dyn.* **2025**, *113*, 6739–6755. [\[CrossRef\]](#)
19. Yang, G. Multilayer Neuroadaptive Reinforcement Learning of Disturbed Nonlinear Systems via Actor-Critic Mechanism. *IEEE Trans. Autom. Sci. Eng.* **2025**, *23*, 1372–1381. [\[CrossRef\]](#)
20. Kong, X.; Xia, Y. Reinforcement learning-based active disturbance rejection control for nonlinear systems with disturbance. In *Proceedings of the 2023 2nd Conference on Fully Actuated System Theory and Applications (CFASTA), Qingdao, China, 14–16 July 2023*; IEEE: New York, NY, USA, 2023; pp. 799–804. [\[CrossRef\]](#)

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.