processes

MDPI

Article

Robust Combustion Prediction for Alternative-Fuel Engines Using an Equilibrium Optimizer-Based Echo State Network with ϵ-Insensitive SVR Readout

Shengyuan Pan [1,*], Xiaoqing Tian [2], Xingquan Wang [3], Tao Xu [3] and Xiaofei Du [4]

1 School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China
2 Binzhou Polytechnic, Binzhou 256602, China; 13326297155@163.com
3 Shandong Key Laboratory of Advanced Engine Piston Assembly, Binzhou Bohai Piston Co., Ltd., Binzhou 256602, China; xingquan.wang@bhpiston.com (X.W.); tao.xu@bhpiston.com (T.X.)
4 School of Mechanical Engineering, Southeast University, Nanjing 211189, China; 230169634@seu.edu.cn
* Correspondence: 320220943211@lzu.edu.cn

Abstract

Alternative-fuel engines, such as diesel–compressed natural gas (CNG) dual fuel systems, exhibit increased cycle-to-cycle combustion variability, placing demanding requirements on the accuracy and robustness of prediction models for key combustion parameters. Echo state networks (ESNs), owing to their reservoir computing architecture, can capture nonlinear temporal dynamics, yet their performance is sensitive to reservoir hyperparameters, and the conventional linear readout trained by minimizing mean squared error is susceptible to outliers in noisy observations. This paper proposes a robust ESN framework based on the equilibrium optimizer (EO), termed EO-Robust-ESN, that automatically searches for key model hyperparameters and replaces the conventional squared loss with the ϵ-insensitive loss by adopting linear support vector regression (SVR) to train the readout weights, thereby enhancing prediction robustness under noisy conditions. Results on the Mackey–Glass chaotic time series benchmark and a peak in-cylinder pressure series from a diesel–CNG dual fuel engine demonstrate that EO significantly outperforms the genetic algorithm and manual tuning on the Mackey–Glass benchmark, reducing the mean RMSE by approximately 5.6% relative to GA, and achieves comparable accuracy with higher search stability on the $P_{\max}$ series. The ϵ-insensitive SVR readout further reduces prediction errors, with the MSE on the noisy $P_{\max}$ series reduced by approximately 42% compared with the ridge regression readout, suggesting that the proposed framework provides an effective data-driven tool for robust combustion prediction in alternative-fuel engines.

Keywords: alternative-fuel engine; echo state network; equilibrium optimizer; ϵ-insensitive support vector regression; robust combustion prediction

Check for updates

Academic Editor: Mattia Bartoli

Received: 23 May 2026
Revised: 24 June 2026
Accepted: 26 June 2026
Published: 1 July 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) license.

1. Introduction

Under the global push toward carbon neutrality, alternative fuels such as compressed natural gas (CNG), hydrogen, and methanol have attracted growing research interest as promising pathways to decarbonize internal combustion engines [1]. Among various implementation strategies, dual fuel combustion, where a gaseous alternative fuel is combined with a conventional pilot fuel, has emerged as a practical approach that leverages existing engine infrastructure while reducing carbon emissions [2]. However, the distinct physicochemical properties of alternative fuels, including differences in ignition delay,

flame propagation speed, and calorific value, introduce stronger nonlinear combustion dynamics and increased cycle-to-cycle variability compared with conventional single-fuel operation [3]. The peak in-cylinder pressure (P_{max}), as a key combustion diagnostic parameter that directly reflects the intensity and stability of the combustion process, is of particular importance for combustion optimization and engine durability assessment [4]. Traditional physics-based approaches such as computational fluid dynamics and detailed chemical kinetic modeling can achieve high fidelity but incur substantial computational costs that limit their applicability to cycle-resolved prediction tasks [5]. Data-driven methods, by contrast, can learn complex input–output mappings directly from experimental observations and offer fast inference after training, making them attractive candidates for combustion parameter prediction in alternative-fuel engines [6,7].

Among data-driven approaches, recurrent neural networks (RNNs) are capable of sequence modeling through their recurrent connections, but gradient-based training suffers from optimization difficulties such as vanishing gradients when capturing long-term dependencies [8,9]. Although related studies have explored approaches such as adaptive gradient activation [10] and fuzzy-neural parameter regulation under uncertainty in complex dynamical systems [11], fundamental challenges remain in RNN training, and the theoretical analysis of RNN dynamics often requires stringent conditions on the underlying mappings and network parameters [12,13]. Furthermore, real-world combustion observations inevitably contain noise with complex, time-varying, and nonlinear characteristics [14], posing additional demands on model robustness. Reservoir computing (RC) offers an alternative paradigm that circumvents such gradient-related issues by using a high-dimensional nonlinear dynamical system to map the input history into a rich state space, where only the readout layer needs to be trained [15]. Within the RC framework, the echo state network (ESN) has been extensively studied and applied to time-series prediction, owing to its conceptual simplicity and ease of implementation [16,17]. Structurally, an ESN consists of an input layer, a reservoir of randomly initialized and sparsely connected neurons, and a trainable readout layer [18,19]. Since only the readout weights need to be trained, the ESN avoids BPTT, resulting in a simpler and more efficient training procedure. This computational efficiency, combined with its ability to capture nonlinear temporal dynamics, makes the ESN a suitable candidate for combustion parameter prediction tasks where the available data are limited in volume, as is common in engine experiments with a finite number of recorded cycles.

However, the performance of ESNs in practical prediction tasks is still constrained by several factors. This paper focuses on two key bottlenecks. First, the configuration of reservoir hyperparameters has a significant impact on prediction performance. Parameters such as the reservoir size, spectral radius, and connectivity density typically require systematic optimization to achieve satisfactory results [20,21]. Second, the conventional readout layer trained by minimizing mean squared error (MSE) is sensitive to outliers [22] and lacks robustness when the data are contaminated by anomalous observations. These two issues are particularly relevant in alternative-fuel engine combustion prediction, where cycle-to-cycle variability and sensor signals acquired under harsh in-cylinder conditions inevitably introduce noise and occasional anomalous readings into the observed data, and where the complex combustion dynamics place stringent demands on the reservoir hyperparameter configuration. The predictive performance of ESNs is sensitive to hyperparameter settings, and improper settings can significantly degrade network performance. In practice, manual tuning and predefined grid search strategies are prone to missing high-performing configurations [23]. To reduce the reliance on manual expertise, researchers have explored automated search methods and systematic validation strategies for ESN hyperparameter tuning. For example, Bayesian optimization is used in [24] to automatically tune key ESN

hyperparameters under memory and time constraints. Multiple validation strategies for the robust selection of hyperparameters for learning chaotic dynamics are systematically investigated in [25], where Bayesian optimization is compared with grid search. Particle swarm optimization has also been applied to tune ESN hyperparameters, with repeated trials revealing non-negligible uncertainty arising from the randomness of reservoir weight matrices [26]. In addition, evolutionary methods such as the genetic algorithm (GA) have been used for ESN hyperparameter tuning [27,28]. However, GA may suffer from loss of population diversity and premature convergence during the search process [29], which limits its effectiveness in ESN hyperparameter optimization. More broadly, recent studies in optimization and neural dynamics have explored methods with improved convergence and robustness for complex optimization problems, such as gradient-based differential neural solutions for time-dependent optimization [30] and coevolutionary neural dynamics for noisy nonconvex optimization [31]. These studies highlight the value of adaptive and stability-enhancing mechanisms in complex optimization settings. The equilibrium optimizer (EO) [32] is a physics-inspired metaheuristic optimization algorithm. It maintains an equilibrium pool composed of four best-so-far candidate solutions and their arithmetic mean, and balances global exploration and local exploitation adaptively through the regulation of a generation rate term. Systematic evaluations on various mathematical benchmark functions and engineering optimization problems show that EO achieves competitive or superior overall optimization performance compared with GA and other classical metaheuristics. The adoption of EO for ESN hyperparameter tuning in this work is motivated by the following considerations. The ESN search space comprises both integer and continuous variables, with a fitness function that lacks an analytic gradient, necessitating a derivative-free search strategy. Among derivative-free methods, surrogate-model-based approaches such as Bayesian optimization require the selection of a surrogate model and an acquisition function, introducing additional configuration decisions. Standard CMA-ES is designed for continuous optimization, and applying it to mixed-integer search spaces requires specialized extensions to avoid premature convergence on discrete variables. Evolutionary methods such as GA and differential evolution involve multiple operator parameters that themselves need tuning. In contrast, EO requires only three algorithm-specific control parameters, all with recommended default values, and has been validated on engineering problems with mixed continuous and discrete variables, making it well suited to the present task. Based on these considerations, this paper adopts EO to tune the key hyperparameters of the ESN.

Beyond hyperparameter configuration, anomalous observations and outliers in real-world time series pose another important robustness challenge for predictive modeling [33]. In the context of alternative-fuel engine combustion, such anomalous observations may arise not only from sensor noise and measurement errors under harsh in-cylinder conditions, but also from abnormal combustion events such as partial burn and misfire cycles, and may adversely affect prediction performance if left untreated. The readout layer of a conventional ESN typically uses a linear readout and estimates the output weights via least squares [34]. However, least squares estimation is sensitive to outliers: even a small number of anomalous observations can substantially bias the parameter estimates [35], thereby degrading prediction performance. To enhance the robustness of the ESN to noise and outliers, several improved strategies have been proposed. For example, support vector regression (SVR) is introduced into the echo state mechanism in [36] by performing linear SVR in the reservoir state space, resulting in the support vector echo-state machine (SVESM). This work demonstrates that incorporating a regularization term and a robust loss function can improve the model's robustness to outliers. The ESN is modeled within a variational Bayesian framework in [37], using a Gaussian mixture distribution as the

likelihood function of the model output and performing parameter estimation through variational inference, thereby enhancing robustness to outliers. In addition, a robust ESN is constructed in [38] using the Cauchy loss function combined with a hybrid regularization term, achieving favorable prediction performance in several noisy time series prediction experiments. The correntropy-induced loss function from information-theoretic learning is introduced into ESN readout layer training in [39]. The bounded gradient of this loss function for large errors helps suppress the influence of outliers on readout weight estimation. These studies indicate that replacing the squared loss with a robust loss function helps enhance the robustness of the ESN to noise. Inspired by this line of work and drawing on the basic idea of SVESM, this paper employs linear SVR [40] to train the readout weights, substituting the ϵ-insensitive loss for the conventional squared loss to reduce the influence of outliers on weight estimation. Unlike SVESM and the other robust ESN variants reviewed above, which address only the readout layer and rely on manual or separate hyperparameter configuration, the present work integrates the reservoir structural hyperparameters and the SVR readout hyperparameters into a unified automated optimization framework, enabling the reservoir feature quality and the readout robustness to be co-optimized rather than configured independently.

Based on the above analysis, this paper proposes an EO-based robust ESN framework (EO-Robust-ESN), which integrates EO and ϵ-insensitive SVR into a unified approach, with improvements targeting two key aspects: hyperparameter configuration and readout layer training. To the best of the authors' knowledge, this is one of the first frameworks to jointly and automatically optimize both the reservoir structure and the robust readout hyperparameters for ESN-based time series prediction, and the first application of such a framework to cycle-resolved combustion parameter prediction in alternative-fuel engines. The remainder of this paper is organized as follows. Section 2 describes the model architecture and algorithmic details. Section 3 presents the experimental setup and results analysis. Section 4 concludes the paper. The main contributions of this work are summarized as follows:

1. We propose the EO-Robust-ESN framework, which employs EO to automatically search for ESN hyperparameters with high search stability, and replaces the conventional squared loss with the ϵ-insensitive loss by adopting linear SVR for training the readout weights, enhancing prediction robustness under noisy conditions.
2. We conduct an experimental evaluation on the Mackey–Glass (MG) chaotic benchmark and a peak in-cylinder pressure series from a diesel–CNG dual fuel engine along two dimensions, namely hyperparameter search quality and readout layer robustness to noise, supplemented by a robust baseline comparison with a Cauchy-loss ESN, computational cost analysis, and sensitivity studies, validating the effectiveness of the proposed framework for combustion parameter prediction in alternative-fuel engines.

2. Methodology

As discussed in Section 1, predicting combustion parameters in alternative-fuel engines poses two key challenges for data-driven models: the complex nonlinear dynamics arising from dual fuel combustion demand careful model configuration, while the cycle-to-cycle variability and harsh in-cylinder measurement conditions inevitably introduce noise and anomalous observations that compromise prediction reliability. These two challenges motivate the design of the proposed EO-Robust-ESN framework, whose components and integration are described in this section. Specifically, Section 2.1 reviews the basic architecture and training procedure of the ESN, providing the foundation for subsequent improvements. Section 2.2 introduces a linear SVR readout layer based on the ϵ-insensitive loss to enhance robustness against anomalous observations. Section 2.3 presents the basic

principle and update mechanism of EO. Finally, Section 2.4 integrates the above components into a unified framework and defines the hyperparameter search space, the fitness function, and the overall workflow.

2.1. Echo State Network

As an efficient variant of RNNs, the ESN exhibits strong capability in capturing nonlinear dynamics in time series, owing to its RC architecture. A conventional ESN consists of three components: an input layer, a reservoir, and a readout layer.

Let N_u, N_x, and N_y denote the number of neurons in the input layer, the reservoir, and the readout layer, respectively. The external input signal is mapped into the reservoir through the input weight matrix $\boldsymbol{W}_{\text{in}} \in \mathbb{R}^{N_x \times N_u}$. The reservoir consists of a large number of sparsely connected neurons, whose internal connections are described by the weight matrix $\boldsymbol{W}_{\text{res}} \in \mathbb{R}^{N_x \times N_x}$. The reservoir states are then mapped to the network output through the output weight matrix $\boldsymbol{W}_{\text{out}} \in \mathbb{R}^{N_y \times (N_x + N_u)}$.

The key innovation of the ESN lies in its training mechanism. Unlike conventional RNNs trained by BPTT, the input weight matrix $\boldsymbol{W}_{\text{in}}$ and the reservoir weight matrix $\boldsymbol{W}_{\text{res}}$ are randomly generated at initialization and remain fixed throughout training. Only the output weight matrix $\boldsymbol{W}_{\text{out}}$ needs to be trained. In addition, to ensure that the reservoir satisfies the ESP, $\boldsymbol{W}_{\text{res}}$ is typically scaled so that its spectral radius is less than one. This training efficiency, combined with the ability of the reservoir to capture nonlinear temporal dynamics, makes the ESN particularly suitable for combustion parameter prediction in engine experiments, where the available data are typically limited to a finite number of recorded engine cycles.

At time step t, given the external input vector $\boldsymbol{u}(t) \in \mathbb{R}^{N_u}$, the reservoir state $\boldsymbol{x}(t) \in \mathbb{R}^{N_x}$ and the network output $\boldsymbol{y}(t) \in \mathbb{R}^{N_y}$ are updated as follows:

$$\boldsymbol{x}(t) = f(\boldsymbol{W}_{\text{res}}\boldsymbol{x}(t-1) + \boldsymbol{W}_{\text{in}}\boldsymbol{u}(t)), \tag{1}$$

$$\boldsymbol{y}(t) = g(\boldsymbol{W}_{\text{out}}\boldsymbol{z}(t)), \tag{2}$$

where $\boldsymbol{z}(t) = [\boldsymbol{x}(t); \boldsymbol{u}(t)] \in \mathbb{R}^{N_x + N_u}$ denotes the vertical concatenation of the current reservoir state and the input vector; $f(\cdot)$ is the nonlinear activation function of the reservoir neurons, typically chosen as the hyperbolic tangent function (tanh); and $g(\cdot)$ is the activation function of the readout layer, typically set to a linear function.

During the training phase of the ESN, the initial reservoir state is typically set randomly, so the states at the first few time steps are affected by the initial transient. Therefore, a washout period is introduced, during which the first T_{trans} time steps affected by the initial conditions are discarded. After the washout period, the collected T concatenated vectors $\boldsymbol{z}(t)$ are arranged column-wise into the state matrix $\boldsymbol{Z} \in \mathbb{R}^{(N_x + N_u) \times T}$, and the corresponding target outputs are arranged column-wise into the matrix $\boldsymbol{Y} \in \mathbb{R}^{N_y \times T}$.

Once the state matrix $\boldsymbol{Z}$ and the target output matrix $\boldsymbol{Y}$ are collected, solving for the output weight matrix $\boldsymbol{W}_{\text{out}}$ can be formulated as a standard linear regression problem. The objective is to minimize the squared error between the network output and the target output:

$$\min_{\boldsymbol{W}_{\text{out}}} \|\boldsymbol{Y} - \boldsymbol{W}_{\text{out}}\boldsymbol{Z}\|_2^2. \tag{3}$$

The closed-form solution to this optimization problem can be obtained by computing the pseudoinverse of the state matrix:

$$\boldsymbol{W}_{\text{out}} = \boldsymbol{Y}\boldsymbol{Z}^{\dagger}, \tag{4}$$

where the superscript † denotes the pseudoinverse.

However, due to the sparsity of the reservoir connections and the presence of outliers in real-world data, directly computing the pseudoinverse may lead to an ill-posed problem, resulting in numerically unstable solutions. To address this, ridge regression is introduced to add a regularization term:

$$\min_{\boldsymbol{W}_{\text{out}}} \|\boldsymbol{Y} - \boldsymbol{W}_{\text{out}}\boldsymbol{Z}\|_2^2 + \lambda\|\boldsymbol{W}_{\text{out}}\|_2^2, \tag{5}$$

where $\lambda > 0$ is the regularization parameter. The closed-form solution to this problem is given by

$$\boldsymbol{W}_{\text{out}} = \boldsymbol{Y}\boldsymbol{Z}^\top(\boldsymbol{Z}\boldsymbol{Z}^\top + \lambda\boldsymbol{I})^{-1}, \tag{6}$$

where $\boldsymbol{I}$ is the identity matrix of appropriate dimension, and the superscripts $^\top$ and $^{-1}$ denote the matrix transpose and inverse, respectively.

2.2. Robust Readout via ϵ-Insensitive SVR

Although ridge regression alleviates numerical instability through regularization, the objective function is still based on the squared loss, which is sensitive to outliers [41]. In alternative-fuel engine combustion prediction, such outliers may originate from abnormal combustion events such as partial burn and misfire cycles, as well as from sensor measurement errors under harsh in-cylinder conditions, making readout layer robustness a practical necessity. Inspired by the SVESM [36], we replace the readout layer with one based on SVR. Since the reservoir already maps the input history into a high-dimensional state space, there is no need for a kernel function. The linear SVR is performed directly on the concatenated state vector $\boldsymbol{z}(t) \in \mathbb{R}^{N_x+N_u}$.

To align with the standard SVR formulation, we assume a univariate output (i.e., $N_y = 1$) in the following. The output weight matrix $\boldsymbol{W}_{\text{out}}$ then reduces to a weight vector $\boldsymbol{w} \in \mathbb{R}^{N_x+N_u}$. With an additional bias term $b \in \mathbb{R}$, the network output at time step t can be written as

$$y(t) = \boldsymbol{w}^\top\boldsymbol{z}(t) + b. \tag{7}$$

We replace the squared loss with the ϵ-insensitive loss function, defined as

$$L_\epsilon(y(t), y_{\text{d}}(t)) = \begin{cases} 0, & |y(t) - y_{\text{d}}(t)| < \epsilon, \\ |y(t) - y_{\text{d}}(t)| - \epsilon, & \text{otherwise}, \end{cases} \tag{8}$$

where $y_{\text{d}}(t)$ is the target output at time step t, and $\epsilon > 0$ is the half-width of the ϵ-insensitive tube. This loss function defines an ϵ-insensitive tube of width 2ϵ: prediction errors within the tube incur no loss, while those outside the tube are penalized linearly rather than quadratically, thereby reducing the influence of outliers on the optimal solution.

Based on the principle of structural risk minimization, slack variables $\xi_t \geq 0$ and $\hat{\xi}_t \geq 0$ are introduced, and the optimal readout weights can be obtained by solving the following primal optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w}, b, \xi, \hat{\xi}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{t=1}^{T}(\xi_t + \hat{\xi}_t) \\ \text{s.t.} \quad & (\boldsymbol{w}^\top\boldsymbol{z}(t) + b) - y_{\text{d}}(t) \leq \epsilon + \xi_t, \\ & y_{\text{d}}(t) - (\boldsymbol{w}^\top\boldsymbol{z}(t) + b) \leq \epsilon + \hat{\xi}_t, \\ & \xi_t, \hat{\xi}_t \geq 0, \quad t = 1, \ldots, T, \end{aligned} \tag{9}$$

where the regularization term $(1/2)\|\boldsymbol{w}\|_2^2$ controls the model complexity, and the penalty parameter $C > 0$ trades off model complexity against training error.

By introducing Lagrange multipliers $\alpha_t^* \geq 0$ and $\alpha_t \geq 0$ for the upper and lower bound constraints, respectively, and applying standard Lagrangian duality theory and the Karush–Kuhn–Tucker conditions, the optimal weight vector can be expressed as

$$\boldsymbol{w} = \sum_{t=1}^{T} (\alpha_t - \alpha_t^*)\, \boldsymbol{z}(t). \tag{10}$$

That is, $\boldsymbol{w}$ is a linear combination of the concatenated state vectors of the training samples, with the coefficients determined by the dual variables $\alpha_t - \alpha_t^*$. After eliminating the primal variables, the primal problem can be transformed into the following dual problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \frac{1}{2} \sum_{i=1}^{T} \sum_{j=1}^{T} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\, \boldsymbol{z}(i)^\top \boldsymbol{z}(j) \\ - \sum_{i=1}^{T} (\alpha_i - \alpha_i^*)\, y_{\mathrm{d}}(i) + \sum_{i=1}^{T} (\alpha_i + \alpha_i^*)\, \epsilon \end{aligned} \tag{11}$$

$$\text{s.t.} \quad 0 \leq \alpha_i,\ \alpha_i^* \leq C, \quad \sum_{i=1}^{T} (\alpha_i - \alpha_i^*) = 0, \quad i = 1, \ldots, T. \tag{12}$$

The dual problem above is a convex quadratic programming problem with linear constraints, which admits a global optimum and can be solved by standard linear SVR solvers.

However, the performance of the SVR readout layer depends not only on the settings of C and ϵ, but also on the quality of the state features $\boldsymbol{z}(t)$ produced by the reservoir. If the reservoir hyperparameters are poorly configured, prediction performance may remain limited even with a robust readout layer. Therefore, the next section introduces EO to jointly optimize the structural hyperparameters of the ESN and the regression hyperparameters of the SVR.

2.3. Parameter Optimization: Equilibrium Optimizer

As noted at the end of the previous section, the prediction performance of the proposed model depends on both the structural hyperparameters of the reservoir and the hyperparameters of the SVR readout layer. In this paper, EO [32] is adopted for joint optimization over the hyperparameter space. Inspired by the mass balance equation for a control volume, EO formulates the global optimization process as a system evolving toward an equilibrium state. In EO, each candidate solution is represented as a vector $\boldsymbol{p} \in \mathbb{R}^D$, where D is the total number of hyperparameters to be optimized. The goal is to find the optimal hyperparameter combination $\boldsymbol{p}^*$ that minimizes the fitness function.

EO initializes the positions of N particles uniformly at random within the lower and upper bounds ($\boldsymbol{p}_{\mathrm{lb}}$ and $\boldsymbol{p}_{\mathrm{ub}}$) of the hyperparameter search space:

$$\boldsymbol{p}_i = \boldsymbol{p}_{\mathrm{lb}} + \mathbf{rand}_i(\boldsymbol{p}_{\mathrm{ub}} - \boldsymbol{p}_{\mathrm{lb}}), \quad i = 1, 2, \ldots, N, \tag{13}$$

where $\mathbf{rand}_i$ is a random vector with elements drawn uniformly from $[0, 1]$. The fitness of all particles is then evaluated and ranked.

At each iteration, the algorithm retains the four particles with the best fitness values found throughout the optimization process, denoted as $\boldsymbol{p}_{\mathrm{eq}(1)}$ through $\boldsymbol{p}_{\mathrm{eq}(4)}$, and computes their arithmetic mean as a fifth candidate $\boldsymbol{p}_{\mathrm{eq(ave)}}$. These five particles together form the equilibrium pool:

$$\boldsymbol{p}_{\mathrm{eq,pool}} = \left\{ \boldsymbol{p}_{\mathrm{eq}(1)},\ \boldsymbol{p}_{\mathrm{eq}(2)},\ \boldsymbol{p}_{\mathrm{eq}(3)},\ \boldsymbol{p}_{\mathrm{eq}(4)},\ \boldsymbol{p}_{\mathrm{eq(ave)}} \right\}. \tag{14}$$

The four best-so-far particles primarily enhance the global exploration capability of the algorithm, while the mean particle $\boldsymbol{p}_{\mathrm{eq(ave)}}$ facilitates local exploitation. During the subsequent position update, each particle randomly selects a target state $\boldsymbol{p}_{\mathrm{eq}}$ from the equilibrium pool with equal probability as its reference point.

To achieve a proper balance between global exploration and local exploitation, EO introduces an exponential term $\boldsymbol{F}$:

$$\boldsymbol{F} = a_1 \operatorname{sign}(\boldsymbol{r} - 0.5)\left[\mathrm{e}^{-\boldsymbol{\tau}\ell} - 1\right], \tag{15}$$

where $\boldsymbol{\tau}$ and $\boldsymbol{r}$ are random vectors with elements drawn uniformly from $[0, 1]$, a_1 is a constant that controls the exploration capability, and the $\operatorname{sign}(\cdot)$ function determines the direction of exploration and exploitation. The term ℓ is a decreasing function of the iteration count:

$$\ell = \left(1 - \frac{\mathrm{Iter}}{\mathrm{Max_iter}}\right)^{\left(a_2 \frac{\mathrm{Iter}}{\mathrm{Max_iter}}\right)}, \tag{16}$$

where Iter and Max_iter denote the current and maximum iteration counts, respectively, and a_2 is a constant that controls the exploitation capability. Following [32], we set $a_1 = 2$ and $a_2 = 1$.

To enhance local exploitation and further refine the solution quality, EO introduces a generation rate term $\boldsymbol{G}$:

$$\boldsymbol{G} = \boldsymbol{G}_0 \boldsymbol{F}, \tag{17}$$

where the initial generation rate is given by

$$\boldsymbol{G}_0 = \mathbf{GCP}\left(\boldsymbol{p}_{\mathrm{eq}} - \boldsymbol{\tau}\boldsymbol{p}\right). \tag{18}$$

The generation rate control parameter **GCP** determines whether the generation rate term participates in the position update of a particle. This parameter is first computed as a scalar for each particle and then replicated across all dimensions to form a vector:

$$\mathrm{GCP} = \begin{cases} 0.5 r_1, & r_2 \geq \mathrm{GP}, \\ 0, & r_2 < \mathrm{GP}, \end{cases} \tag{19}$$

where r_1 and r_2 are random numbers drawn uniformly from $[0, 1]$, and GP is the generation probability. Following [32], we set $\mathrm{GP} = 0.5$.

Combining the above mechanisms, the position update rule for each particle in EO is defined as follows:

$$\boldsymbol{p}_{\mathrm{new}} = \boldsymbol{p}_{\mathrm{eq}} + \left(\boldsymbol{p} - \boldsymbol{p}_{\mathrm{eq}}\right)\boldsymbol{F} + \frac{\boldsymbol{G}}{\boldsymbol{\tau}}(1 - \boldsymbol{F}). \tag{20}$$

On the right-hand side of this equation, the first term $\boldsymbol{p}_{\mathrm{eq}}$ represents the equilibrium state. The second term is governed by the exponential term $\boldsymbol{F}$ and performs global exploration by leveraging the difference between the particle and the equilibrium state. The third term is driven by the generation rate $\boldsymbol{G}$ and primarily contributes to local exploitation. However, when the turnover rate $\boldsymbol{\tau}$ takes small values in certain dimensions, the denominator effect can produce large displacements, thereby complementing global exploration [32].

In addition, EO incorporates a memory saving mechanism: each particle updates its stored position only when its fitness improves, and otherwise retains its previous position. This mechanism helps strengthen the exploitation capability of the algorithm, although it may increase the risk of being trapped in local optima [32]. Through the synergy of iterative updating and memory saving, EO is able to automatically search for well-performing hyperparameter configurations.

2.4. EO-Robust-ESN Framework

The preceding three sections introduce the ESN (Section 2.1), the robust readout layer based on ϵ-insensitive SVR (Section 2.2), and EO (Section 2.3), respectively. This section integrates these components into a unified framework for automated hyperparameter optimization and robust time-series prediction. The core idea is to replace the conventional squared loss with the ϵ-insensitive loss by adopting linear SVR for training the readout weights to enhance robustness against noise and outliers, while employing EO as the outer-loop optimizer to automatically search for model hyperparameters. These two components are linked through a fitness function based on expanding window cross-validation. In addition, the framework is compatible with the standard ridge regression readout as a simplified configuration, allowing the individual contribution of each component to be isolated in experiments.

In this framework, each particle $\boldsymbol{p}$ in EO encodes a set of model hyperparameters to be optimized. Under both readout layer configurations, three core reservoir hyperparameters are included in the search space, forming the base dimensions:

$$\boldsymbol{p} = [N_x,\ \rho,\ d], \tag{21}$$

where N_x is the number of reservoir neurons, ρ is the spectral radius of the reservoir weight matrix $\boldsymbol{W}_{\text{res}}$, and d is the connectivity density of the reservoir. When the ϵ-insensitive SVR readout is adopted, the search space is extended to five dimensions to enable joint optimization of the reservoir and SVR readout hyperparameters:

$$\boldsymbol{p} = [N_x,\ \rho,\ d,\ \log_{10} C,\ \log_{10} \epsilon], \tag{22}$$

where C is the penalty parameter in Equation (9) and ϵ is the half-width of the ϵ-insensitive tube in Equation (8). Since the effective ranges of C and ϵ typically span several orders of magnitude, both are searched in log space to allow the optimizer to explore across different scales. In addition, since the reservoir size N_x must be a positive integer, it is rounded to the nearest integer at each fitness evaluation. The search ranges of the above hyperparameters are listed in Table 1.

Table 1. Hyperparameter search space definition.

Hyperparameter	Symbol	Lower Bound	Upper Bound	Configuration
Reservoir size	N_x	50	500	Both
Spectral radius	ρ	0.01	1.0	Both
Connectivity density	d	0.01	0.1	Both
Penalty parameter (log)	$\log_{10} C$	-2	6	SVR only
Tube half-width (log)	$\log_{10} \epsilon$	-4	-1	SVR only

The fitness function serves as the key bridge connecting the outer-loop EO optimizer and the inner-loop ESN model, and is used to evaluate the prediction performance corresponding to each candidate hyperparameter set $\boldsymbol{p}$. Since time-series data exhibit strict temporal dependencies, standard randomized k-fold cross-validation may introduce future information leakage. To address this, we adopt an expanding window cross-validation strategy: all non-test data form the candidate data pool $\mathcal{D}_{\text{cv}}$, from which K folds are constructed in chronological order, with the training window expanding and the validation window sliding forward at each fold.

Each fold involves a complete ESN construction and evaluation procedure: the reservoir is configured according to the candidate hyperparameters $\boldsymbol{p}$, driven by the state update equations in Section 2.1 to collect the state matrix after discarding the washout period, and

the readout weights are then trained according to the readout layer type and the validation error S_k is computed on the validation subset. The final fitness value is defined as the sum of the K-fold mean validation error and a mild complexity penalty term:

$$\mathcal{F}(\boldsymbol{p}) = \frac{1}{K}\sum_{k=1}^{K} S_k(\boldsymbol{p}) + \mu\frac{N_x}{N_{x,\max}}, \tag{23}$$

where $\mu \ll 1$ is the penalty coefficient and $N_{x,\max}$ is the upper bound of the reservoir size search range. This penalty term is intended to provide a mild preference for smaller reservoirs when the prediction performance is comparable, acting as a weak complexity tie-breaking term. In all experiments, the normalized root mean squared error is used as the validation error S_k.

Based on the search space and fitness function defined above, the complete workflow of the proposed framework can be divided into two stages.

1. Hyperparameter optimization: All non-test data form the candidate data pool $\mathcal{D}_{\mathrm{cv}}$. EO randomly initializes the particle population within the search ranges defined in Table 1 according to Equation (13), and then enters the iterative optimization loop. At each iteration, each particle is evaluated using the fitness function defined in Equation (23). EO then updates the equilibrium pool (Equation (14)) and guides all particles toward better regions according to the position update rule (Equation (20)). After the prescribed maximum number of iterations, the particle with the best fitness value in the equilibrium pool, $\boldsymbol{p}^* = \boldsymbol{p}_{\mathrm{eq}(1)}$, is taken as the optimal hyperparameter combination.
2. Final model training and prediction: The ESN is configured with the optimal hyperparameters $\boldsymbol{p}^*$ and trained on the entire data pool $\mathcal{D}_{\mathrm{cv}}$. After training, predictions are made on the held-out test set, and evaluation metrics are computed to assess the generalization performance of the model.

In summary, the proposed framework forms a two-level time series prediction framework. At the outer level, the global optimization capability of EO enables an efficient automated hyperparameter optimization, alleviating the inefficiency of manual tuning and the instability of GA. At the inner level, the reservoir of the ESN captures the nonlinear dynamics of the time series, while the ϵ-insensitive SVR readout layer enhances prediction robustness under noisy conditions through its robustness to outliers. The synergy between outer-loop optimization and inner-loop robust modeling endows the overall framework with both automated hyperparameter optimization and robust prediction capability. In the context of alternative-fuel engine combustion prediction, the outer-loop automated optimization addresses the stringent model configuration demands imposed by the complex dynamics of dual fuel combustion, while the inner-loop robust modeling mitigates the impact of inherent cycle-to-cycle variability and measurement noise present in engine experimental data.

3. Experiments

This section evaluates the proposed framework on two datasets. Section 3.1 first introduces the diesel–CNG dual fuel engine experimental setup and the peak in-cylinder pressure dataset used in this study. The MG time series, a widely used benchmark in time series prediction, is then employed in Section 3.2 to validate the basic effectiveness of the proposed method. The framework is subsequently applied to the $P_{\max}$ series in Section 3.3 to examine its performance on real combustion data with inherent cycle-to-cycle variability. The experiments are organized along two main lines. The first compares the solution quality and stability of manual tuning, GA, and EO for ESN hyperparameter search. The second compares the robustness to noise of the conventional ridge regression readout

and the ϵ-insensitive SVR readout under the same EO search framework. In addition, a Cauchy-loss robust ESN readout layer [38] is included as an additional baseline under the same EO framework to further assess the relative advantage of the proposed readout, and computational cost analysis and sensitivity studies are provided to further characterize the proposed method. All experiments are conducted on the one-step-ahead prediction task, i.e., predicting the next value of the time series. All simulations and experiments are performed 10 times independently, and the results are reported as mean $\pm$ std. The two-sided Wilcoxon signed-rank test is applied to all main pairwise comparisons based on the 10 paired independent runs to assess statistical significance at a significance level of $\alpha = 0.05$.

The data are split chronologically into a training set and a held-out test set. Both the input and the prediction target are standardized using Z-score standardization, where the mean and standard deviation are computed from the training set only. All methods share the same ESN configuration: leaking rate $\alpha_{\text{leak}} = 0.8$ and input scaling $s_{\text{in}} = 0.5$, confirmed by the sensitivity analysis in Section 3.5. The washout length is set to $T_{\text{trans}} = 100$ for the MG series and $T_{\text{trans}} = 20$ for the $P_{\max}$ series; the shorter washout for the $P_{\max}$ series is adopted to retain sufficient training samples given the limited 500-cycle dataset. Two readout configurations are considered. For the ridge regression readout, the regularization parameter is fixed at $\lambda = 10^{-6}$. For the ϵ-insensitive SVR readout, the penalty parameter C and the insensitive tube half-width ϵ are jointly optimized by EO. The hyperparameter search space is listed in Table 1, and the fitness value is computed via three-fold expanding window cross-validation (Equation (23)), which guards against overfitting by selecting hyperparameters based on held-out validation folds rather than the training data, while its chronological ordering prevents information leakage. For a fair comparison, EO and GA use the same population size and maximum number of iterations on the same dataset, with specific values given in the respective subsections. The EO parameters are set to $a_1 = 2$, $a_2 = 1$, and GP $= 0.5$, following the recommended default values in [32]. For GA, all parameters other than the population size and maximum number of iterations are set to their default values. The sensitivity of prediction performance to the fixed hyperparameters (α_{leak}, s_{in}, and the complexity penalty coefficient μ) and to the noise injection parameters is examined in Section 3.5. Model performance is evaluated on the held-out test set using the following metrics:

$$\text{MSE} = \frac{1}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} (y(t) - \hat{y}(t))^2, \tag{24}$$

$$\text{MAE} = \frac{1}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} |y(t) - \hat{y}(t)|, \tag{25}$$

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{t=1}^{N_{\text{test}}} (y(t) - \hat{y}(t))^2}, \tag{26}$$

where $y(t)$ and $\hat{y}(t)$ denote the target and predicted values, respectively, and N_{test} is the number of test samples.

3.1. Diesel–CNG Dual Fuel Engine and Combustion Dataset

This subsection describes the diesel–CNG dual fuel engine test bench and the combustion dataset used to evaluate the proposed framework on a real-world prediction task. The $P_{\max}$ is a key combustion diagnostic parameter in internal combustion engines, directly reflecting the intensity and stability of the combustion process. In dual fuel engines that blend diesel with CNG, the introduction of gaseous fuel increases the cycle-to-cycle variability of

combustion, making P_{max} prediction both practically important and technically challenging. This study uses experimental data acquired from a Toyota 2KD-FTV 2.5-litre direct-injection turbocharged four-cylinder diesel engine that was converted to a diesel–CNG dual fuel system. The main specifications of the engine are listed in Table 2, and the schematic of the experimental setup is shown in Figure 1.

Table 2. Main specifications of the test engine.

Parameter	Value
Engine code	2KD-FTV
Number of cylinders	4 in-line
Valve mechanism	16-valve DOHC
Displacement	2494 mL
Bore × Stroke	92 mm × 93.8 mm
Compression ratio	17.4:1
Rated power	80 kW @ 3600 rpm
Maximum torque	325 N·m @ 2000 rpm

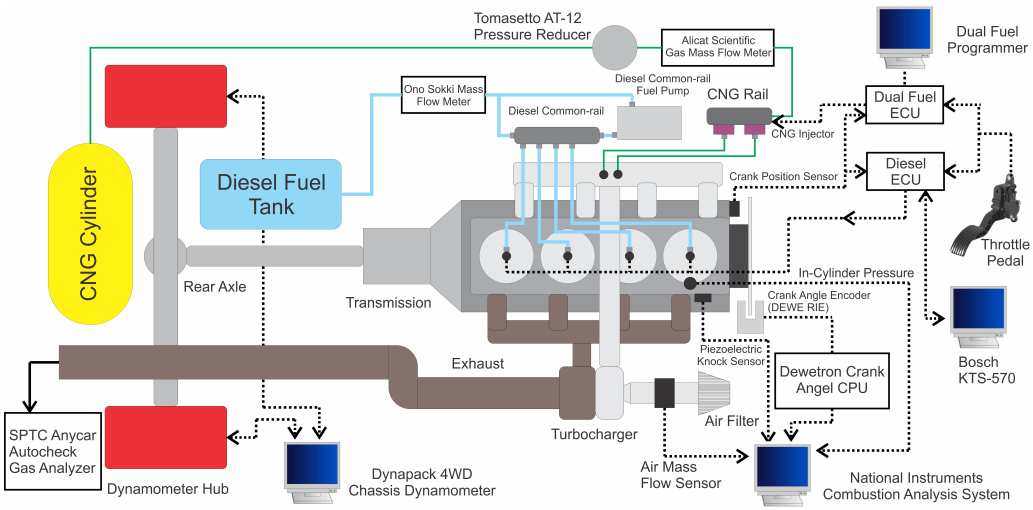

Figure 1. Schematic of the diesel–CNG dual fuel engine experimental setup. Solid green lines indicate the CNG fuel supply path, solid light blue lines indicate the diesel fuel supply path, dashed black lines with arrows indicate electrical signal and control connections, and the solid black arrow indicates the exhaust gas sampling flow to the gas analyzer.

The dataset covers engine speeds from 1400 rpm to 3000 rpm at 200 rpm intervals, with diesel-to-CNG fuel mass ratios of 100:0, 90:10, 80:20, and 70:30 at a constant equivalence ratio of 0.7. For each operating condition, the dataset provides instantaneous in-cylinder pressure, diesel injection timing, air mass flow, and engine block vibration signals over 500 consecutive engine cycles, along with engine performance and exhaust emission data.

The operating condition selected for the present study corresponds to an engine speed of 2400 rpm and a diesel-to-CNG fuel mass ratio of 70:30. This condition was chosen because, among all available test points in the dataset, the highest CNG substitution ratio introduces the most pronounced cycle-to-cycle combustion variability, thereby imposing the most stringent demand on the robustness of the prediction model. In-cylinder pressure traces were recorded over 500 consecutive engine cycles at a resolution of 720 crank angle samples per cycle. For each cycle, the maximum value of the pressure trace was extracted to obtain the P_{max}, yielding a univariate time series of 500 data points arranged in cycle order. Figure 2 shows the resulting P_{max} sequence, which fluctuates around a mean of approximately 119.5 bar with a coefficient of variation of 1.78%, reflecting the inherent

cycle-to-cycle combustion variability under dual fuel operation. Unlike the controllable MG simulation, the noise sources and statistical properties of this engine dataset are unknown. Moreover, the series naturally contains anomalous combustion cycles, providing a realistic setting for evaluating both the search stability of the optimizer and the robustness of the readout layer.

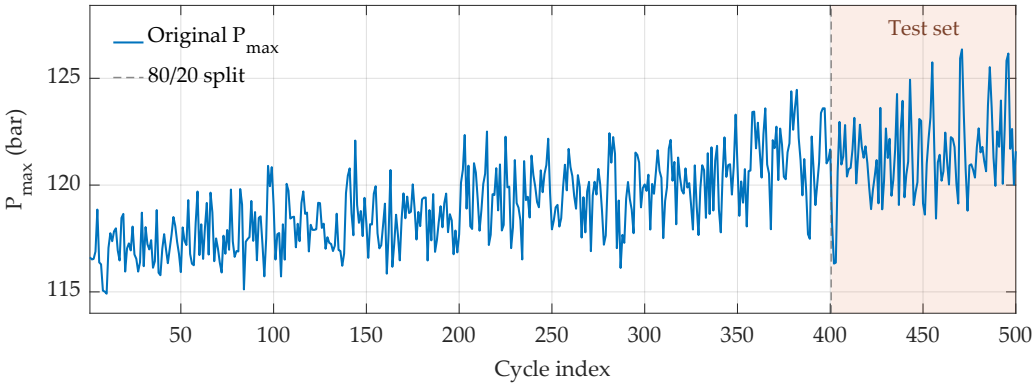

Figure 2. The $P_{\max}$ sequence extracted from 500 consecutive engine cycles of the diesel–CNG dual fuel engine at 2 400 rpm with a 70:30 diesel-to-CNG fuel mass ratio. The dashed line indicates the 80%/20% train–test split.

3.2. Mackey–Glass Time Series

The MG time series is a widely used benchmark for time series prediction. Its dynamics is governed by the following delay differential equation:

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = -\gamma v(t) + \frac{\beta v(t-\tau)}{1 + v(t-\tau)^n}. \tag{27}$$

In this study, the parameters are set to $\beta = 0.2$, $\gamma = 0.1$, $n = 10$, and $\tau = 17$. The equation is numerically integrated using the fourth-order Runge–Kutta method with a step size of $\Delta t = 0.1$. To eliminate the effect of initial conditions, the first 1000 time steps are discarded as an initial transient, and the subsequent 10,000 samples are retained for the experiments.

We first compare the solution quality of three hyperparameter configuration strategies under noise-free conditions. The three methods, namely manual tuning (Manual-ESN), GA search (GA-ESN), and EO search (EO-ESN), share the same ESN architecture and ridge regression readout, differing only in the hyperparameter configuration strategy. The data are split into a training set and a test set in an 80%/20% ratio. Both EO and GA use a population size of 20 and a maximum of 30 iterations. Table 3 reports the test set statistics over 10 independent runs.

As shown in Table 3, all three methods achieve high prediction accuracy on this dataset. However, the automated search methods (GA and EO) further reduce the test error compared with manual tuning, and EO-ESN outperforms GA-ESN in the mean values of all three metrics, with an MSE reduction of 11.15%. Moreover, EO-ESN yields substantially lower standard deviations across all three metrics, indicating not only better average solution quality but also higher search stability. All pairwise differences are statistically significant. Figure 3 plots the mean best-so-far validation fitness of EO and GA against the number of iterations over 10 independent runs, with the shaded areas indicating the standard deviation. The fitness of EO not only drops faster in the early iterations but also converges to a lower level, and its standard deviation band is noticeably narrower than that of GA.

Table 3. Performance comparison of different hyperparameter configuration strategies on MG under noise-free conditions.

Metric	Manual-ESN	GA-ESN	EO-ESN (Ours)
MSE	$7.48\times10^{-12}\pm5.74\times10^{-13}$	$6.45\times10^{-12}\pm7.49\times10^{-13}$	$\mathbf{5.73\times10^{-12}\pm2.54\times10^{-13}}$
RMSE	$2.73\times10^{-6}\pm1.04\times10^{-7}$	$2.54\times10^{-6}\pm1.45\times10^{-7}$	$\mathbf{2.39\times10^{-6}\pm5.28\times10^{-8}}$
MAE	$2.13\times10^{-6}\pm9.45\times10^{-8}$	$1.96\times10^{-6}\pm1.13\times10^{-7}$	$\mathbf{1.85\times10^{-6}\pm4.92\times10^{-8}}$

The best result for each metric is shown in bold. Wilcoxon signed-rank test p-values: Manual vs. GA: $p = 0.014$ for all three metrics; Manual vs. EO: $p = 0.002$ for all three metrics; GA vs. EO: MSE/RMSE $p = 0.006$, MAE $p = 0.010$.

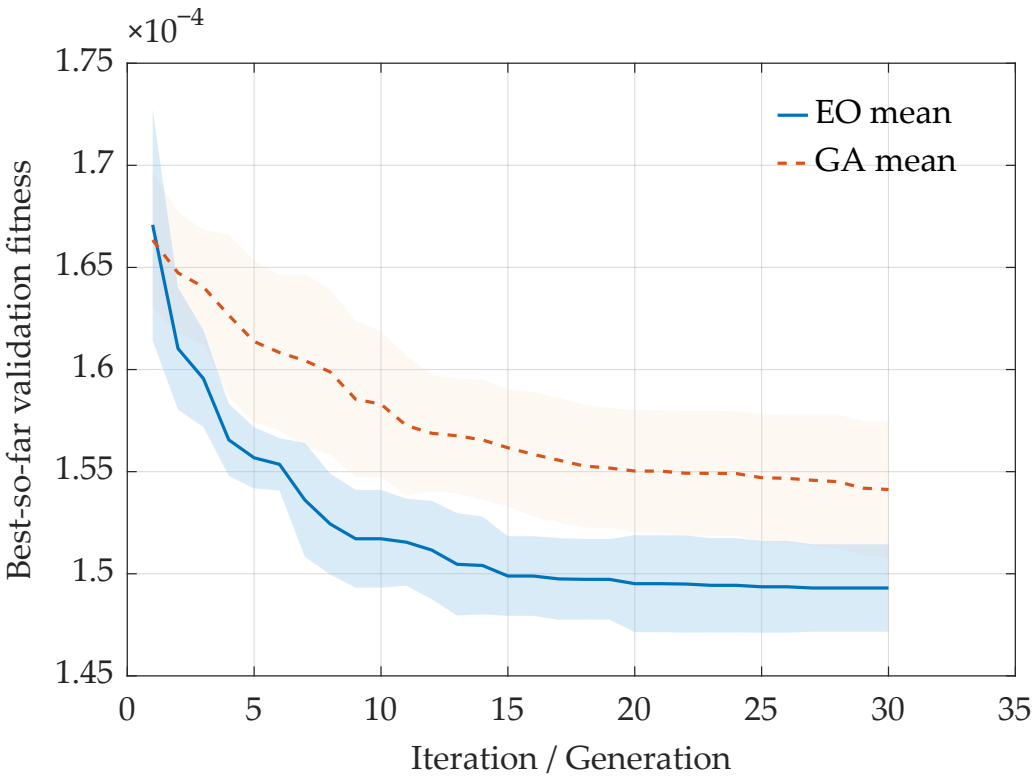

Figure 3. Convergence behavior of EO and GA in the noise-free MG experiment. The solid blue line and the dashed orange line represent the mean best-so-far validation fitness of EO and GA over 10 independent runs, respectively, and the shaded areas in the corresponding colors indicate the standard deviation.

To evaluate the robustness of the proposed ϵ-insensitive SVR readout layer under noisy conditions, additive white Gaussian noise at SNR $= 20$ dB is added to the MG time series. In addition, zero-mean impulsive perturbations with a standard deviation of $4\sigma_{\text{noisy}}$ are injected at randomly selected 5% of the time steps, where σ_{noisy} denotes the standard deviation of the noisy series before impulse injection. This setup simulates an observation environment containing both background noise and sparse outliers; the sensitivity of prediction performance to these noise parameters is examined on the $P_{\max}$ series in Section 3.5. The data are split into a training set and a test set in an 80%/20% ratio. During the hyperparameter optimization stage, both the input and the target are the noisy series. During testing, the model input is also the noisy observation, but the test error is computed against the clean ground truth, so as to assess the ability of the model to recover the underlying dynamics from noisy observations. The two models, EO-ESN and EO-Robust-ESN, share the same EO search framework and the same noisy series, differing only in the readout layer.

As shown in Table 4, EO-Robust-ESN achieves lower mean values than EO-ESN across all three metrics, with the MSE reduced by 14.16%, with smaller standard deviations in MSE and RMSE as well. This indicates that the ϵ-insensitive loss helps suppress the interference of impulsive perturbations on readout layer training while improving the stability of the results. Figure 4 visualizes the prediction results of EO-ESN and EO-Robust-ESN on the noisy MG test set from four perspectives: prediction trajectories, a zoomed-in segment, absolute errors over time, and the empirical cumulative distribution function (ECDF) of absolute errors. As shown in Figures 4a,b, the robust model tracks the clean ground truth more closely. Figure 4c further shows that the error spikes are reduced in both magnitude and number. In addition, the ECDF in Figure 4d shifts toward smaller values overall, confirming that the ϵ-insensitive SVR readout yields lower prediction errors across the entire error distribution.

Table 4. Performance comparison of EO-ESN and EO-Robust-ESN on MG under noisy conditions.

Metric	EO-ESN	EO-Robust-ESN (ϵ-SVR)
MSE	$2.43 \times 10^{-3} \pm 1.87 \times 10^{-4}$	$\mathbf{2.09 \times 10^{-3} \pm 1.59 \times 10^{-4}}$
RMSE	$4.93 \times 10^{-2} \pm 1.87 \times 10^{-3}$	$\mathbf{4.57 \times 10^{-2} \pm 1.70 \times 10^{-3}}$
MAE	$3.81 \times 10^{-2} \pm 1.30 \times 10^{-3}$	$\mathbf{3.56 \times 10^{-2} \pm 1.31 \times 10^{-3}}$

The best result for each metric is shown in bold. Wilcoxon signed-rank test: $p = 0.006$ for all three metrics.

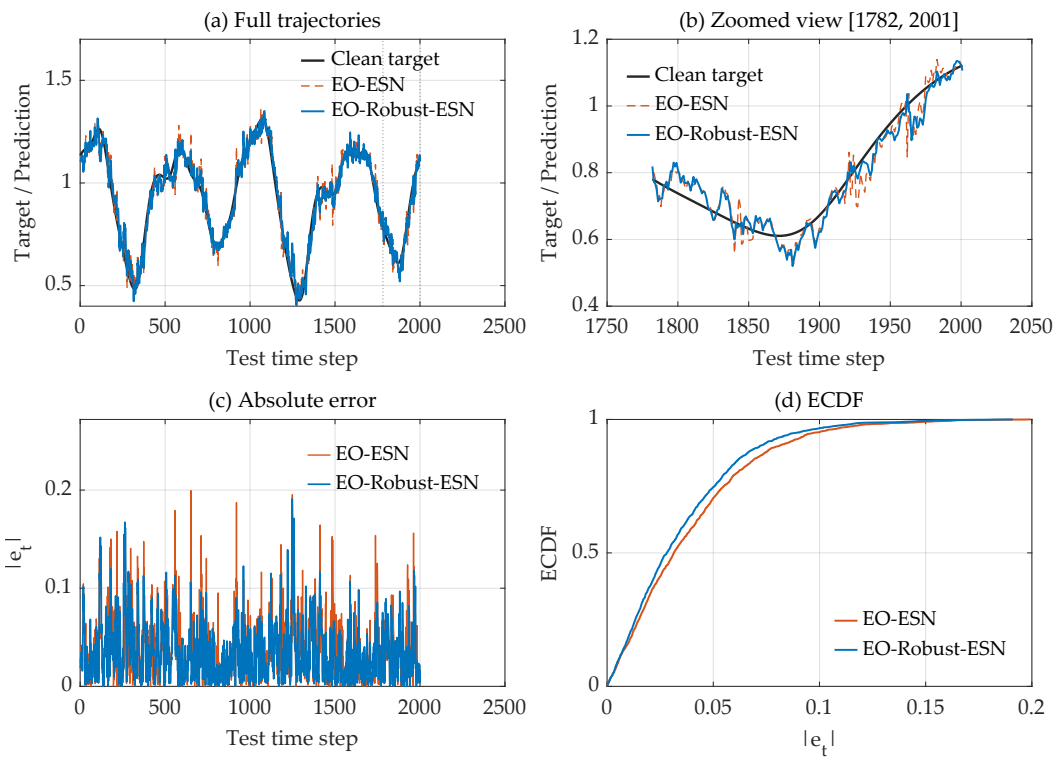

Figure 4. Robustness visualization on MG under noisy conditions. (**a**) Overall prediction trajectory; (**b**) Zoomed-in view; (**c**) Absolute error over time; (**d**) ECDF of absolute errors.

To further summarize the stability of different methods over repeated trials, Figure 5 presents the box plots of RMSE over 10 independent runs on the MG series. As shown in Figure 5a, EO-ESN exhibits a lower median RMSE and a more compact distribution than Manual-ESN and GA-ESN, confirming the superior search stability of EO under noise-free conditions. Figure 5b further shows that EO-Robust-ESN yields a lower and more concentrated RMSE distribution than EO-ESN under noisy conditions, demonstrating the improved stability brought by the ϵ-insensitive SVR readout.

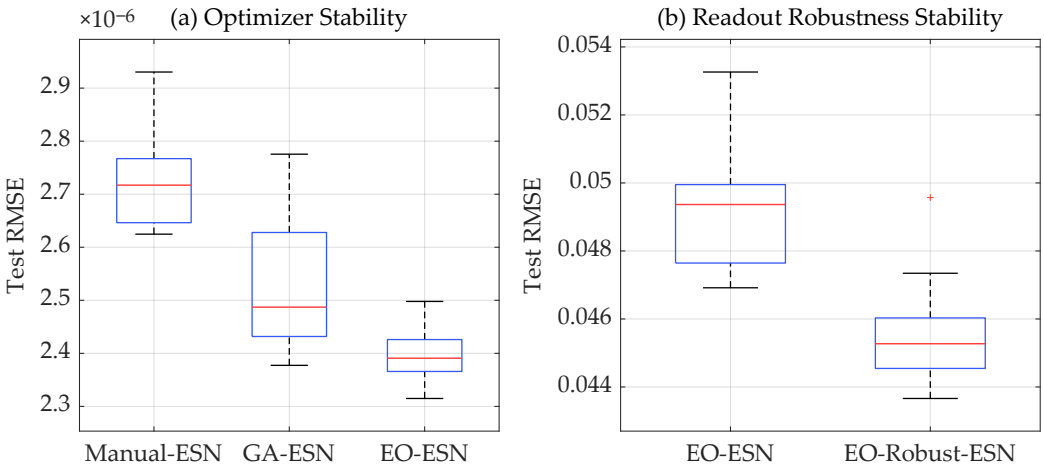

Figure 5. Box plots of RMSE over 10 independent runs on the MG series. (**a**) Optimizer comparison; (**b**) Readout layer comparison under noisy conditions.

3.3. Peak In-Cylinder Pressure Prediction

Having validated the basic effectiveness of the proposed framework on the MG benchmark, we now apply it to the $P_{\max}$ series of the diesel–CNG dual fuel engine described in Section 3.1 to further examine its practical applicability. The data are split into a training set and a test set in an 80%/20% ratio. Both EO and GA use a population size of 30 and a maximum of 20 iterations.

We first compare the performance of three hyperparameter configuration strategies on the one-step-ahead prediction task for the $P_{\max}$ series. Table 5 reports the statistics over 10 independent runs. As shown in Table 5, the automated search methods clearly outperform manual tuning, with all pairwise differences between manual tuning and the automated methods being statistically significant ($p < 0.03$). EO-ESN and GA-ESN achieve comparable mean values across all three metrics, with no statistically significant difference ($p = 0.846$). However, EO-ESN exhibits smaller standard deviations across all three metrics, indicating higher search stability.

Table 5. Performance comparison of different hyperparameter configuration strategies on the $P_{\max}$ series.

Metric	Manual-ESN	GA-ESN	EO-ESN (Ours)
MSE	5.397 ± 0.531	4.728 ± 0.326	$\mathbf{4.704 \pm 0.207}$
RMSE	2.320 ± 0.116	2.173 ± 0.075	$\mathbf{2.168 \pm 0.048}$
MAE	1.745 ± 0.046	1.676 ± 0.038	$\mathbf{1.674 \pm 0.024}$

The best result for each metric is shown in bold. Wilcoxon signed-rank test p-values: Manual vs. GA: MSE/RMSE $p = 0.020$, MAE $p = 0.014$; Manual vs. EO: MSE/RMSE $p = 0.020$, MAE $p = 0.027$; GA vs. EO: $p = 0.846$ (n.s.) for all three metrics.

Figure 6 plots the mean best-so-far validation fitness of EO and GA against the number of iterations over 10 independent runs, with the shaded areas indicating the corresponding standard deviation. The fitness of EO not only descends faster in the early iterations but also converges to a lower level, and its standard deviation band is substantially narrower than that of GA, nearly vanishing after the initial iterations. This indicates that EO can more reliably find high-quality hyperparameter configurations when facing the cycle-to-cycle variability inherent in real engine combustion data. Among the three methods, EO-ESN achieves a mean RMSE of 2.168 bar with the smallest standard deviation of 0.048 bar, demonstrating stable $P_{\max}$ prediction with the lowest run-to-run variability under dual fuel combustion conditions.

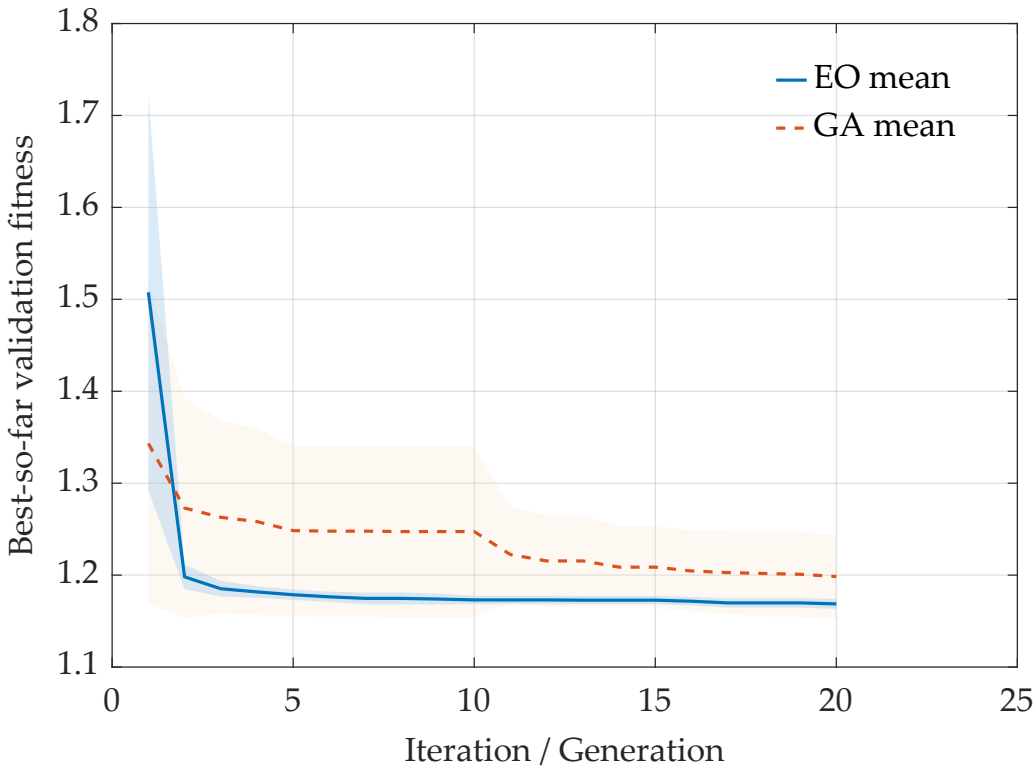

Figure 6. Convergence behavior of EO and GA in the $P_{\max}$ experiment. The solid blue line and the dashed orange line represent the mean best-so-far validation fitness of EO and GA over 10 independent runs, respectively, and the shaded areas in the corresponding colors indicate the standard deviation.

To evaluate the robustness of the proposed ϵ-insensitive SVR readout layer under noisy conditions, additive white Gaussian noise at SNR $= 20\,\mathrm{dB}$ is added to the $P_{\max}$ series. In addition, zero-mean impulsive perturbations with a standard deviation of $4\sigma_{\mathrm{noisy}}$ are injected at randomly selected 5% of the time steps, where σ_{noisy} denotes the standard deviation of the noisy series before impulse injection; the sensitivity of prediction performance to these noise parameters is examined in Section 3.5. As in the MG experiment, the test error is computed against the clean ground truth to assess the ability of the model to recover the underlying dynamics from noisy observations. The two models, EO-ESN and EO-Robust-ESN, share the same EO search framework and the same noisy series, differing only in the readout layer. Table 6 reports the statistics over 10 independent runs.

Table 6. Performance comparison of EO-ESN and EO-Robust-ESN on the $P_{\max}$ series under noisy conditions.

Metric	EO-ESN	EO-Robust-ESN (ϵ-SVR)
MSE	8.097 ± 3.179	$\mathbf{4.696 \pm 0.865}$
RMSE	2.805 ± 0.502	$\mathbf{2.159 \pm 0.195}$
MAE	2.051 ± 0.181	$\mathbf{1.733 \pm 0.179}$

The best result for each metric is shown in bold. Wilcoxon signed-rank test: $p = 0.002$ for all three metrics.

As shown in Table 6, EO-Robust-ESN achieves substantially lower mean values than EO-ESN across all three metrics, with markedly smaller standard deviations for MSE and RMSE as well. In particular, the MSE is reduced by approximately 42%, and this improvement is statistically significant ($p = 0.002$). The standard deviation of MSE is reduced by more than 72%, indicating that the ϵ-insensitive loss effectively suppresses the interference of impulsive perturbations on readout layer training while substantially improving the stability of the results. This robustness is of practical value for $P_{\max}$ prediction in dual

fuel engines, where sensor signals are inevitably contaminated by measurement noise and occasional anomalous readings. This result is consistent with the findings from the noisy MG experiment, confirming that the robustness advantage of the ϵ-insensitive SVR readout layer generalizes to real engine combustion data.

Figure 7 visualizes the prediction results of EO-ESN and EO-Robust-ESN on the noisy $P_{\max}$ test set from four perspectives: prediction trajectories, a zoomed-in segment, absolute errors over time, and the ECDF of absolute errors. As shown in Figure 7a,b, the EO-ESN predictions exhibit large deviations from the clean target due to the influence of impulsive perturbations on the ridge regression readout, whereas EO-Robust-ESN maintains substantially more stable predictions with fewer extreme deviations. Figure 7c further shows that the error spikes of EO-ESN reach nearly 10 bar, while those of EO-Robust-ESN are reduced in both magnitude and number. In addition, the ECDF in Figure 7d shows that, beyond the small-error region, EO-Robust-ESN rises notably faster and remains above EO-ESN, confirming that the ϵ-insensitive SVR readout effectively suppresses moderate-to-large prediction errors.

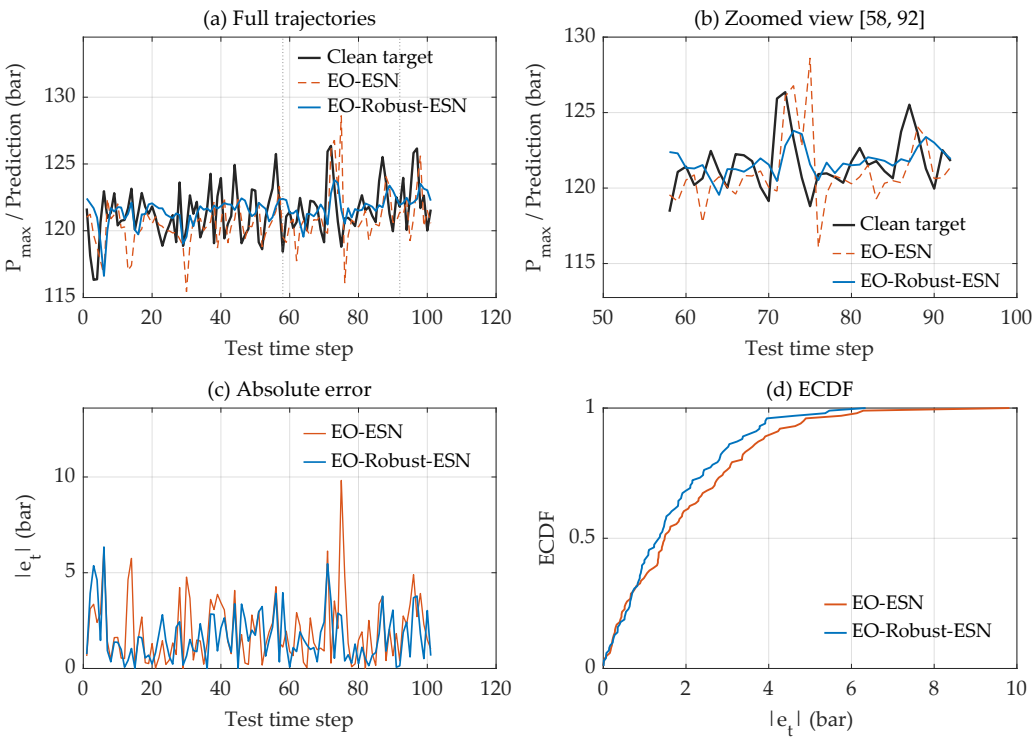

Figure 7. Robustness visualization on the $P_{\max}$ series under noisy conditions. (**a**) Overall prediction trajectory; (**b**) Zoomed-in view; (**c**) Absolute error over time; (**d**) ECDF of absolute errors.

Figure 8 presents box plots of RMSE across 10 independent runs on the $P_{\max}$ series. In the optimizer comparison experiment, as shown in Figure 8a, EO-ESN exhibits both the lowest median error and the most compact box among the three methods, confirming comparable solution quality and higher search stability of EO. In the readout layer comparison experiment under noisy conditions, Figure 8b shows that EO-ESN suffers from a widely dispersed distribution, reflecting its sensitivity to impulsive perturbations. Conversely, EO-Robust-ESN exhibits a substantially more compact distribution concentrated in a lower range, without such extreme deviations. This is consistent with the quantitative results in Table 6 and further demonstrates the advantage of the ϵ-insensitive SVR readout layer in suppressing the influence of outliers on prediction performance.

To further assess the relative advantage of the proposed ϵ-insensitive SVR readout, we compare it against the Cauchy-loss ESN with hybrid regularization proposed in [38],

implemented under the same EO framework with the same noisy $P_{\max}$ series and paired seeds. Table 7 reports the results over 10 independent runs.

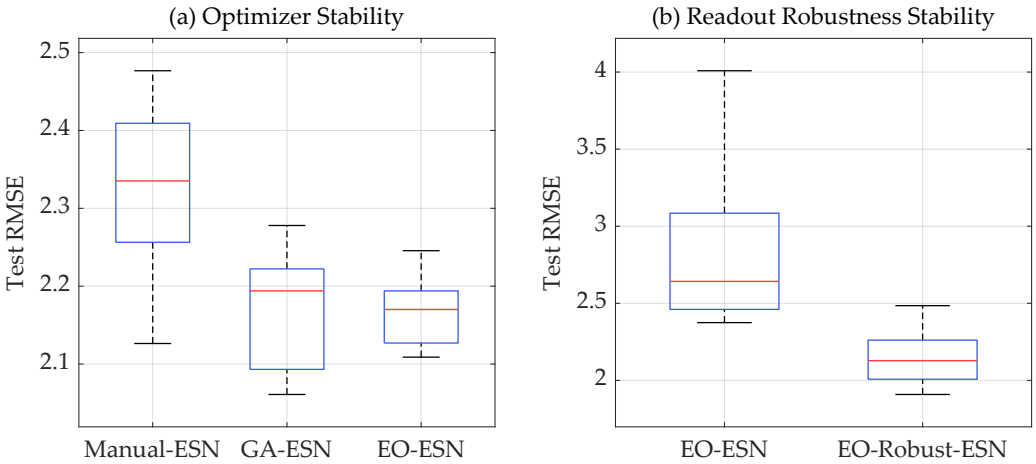

Figure 8. Box plots of RMSE over 10 independent runs on the $P_{\max}$ series. (**a**) Optimizer comparison; (**b**) Readout layer comparison under noisy conditions.

Table 7. Performance comparison of three readout layers on the noisy $P_{\max}$ series under the same EO framework.

Metric	EO-ESN (Ridge)	EO-Robust-ESN (ϵ-SVR)	EO-Cauchy-ESN
MSE	8.097 ± 3.179	4.696 ± 0.865	4.737 ± 0.264
RMSE	2.805 ± 0.502	2.159 ± 0.195	2.176 ± 0.061
MAE	2.051 ± 0.181	1.733 ± 0.179	1.775 ± 0.045

Wilcoxon: Cauchy vs. Ridge $p = 0.002$ for all three metrics; Cauchy vs. ϵ-SVR: MSE/RMSE $p = 0.695$, MAE $p = 0.557$ (n.s.).

As shown in Table 7, both robust readout layers significantly outperform the conventional ridge regression readout ($p = 0.002$), confirming that replacing the squared loss with a robust loss function effectively enhances the noise resilience of the ESN. The proposed ϵ-insensitive SVR readout and the Cauchy-loss readout achieve comparable prediction accuracy with no statistically significant difference ($p = 0.695$ for MSE). The ϵ-SVR readout achieves a slightly lower mean MSE (4.696 vs. 4.737), while the Cauchy-loss readout exhibits a smaller standard deviation (0.264 vs. 0.865). From a practical standpoint, the regularized linear ϵ-SVR readout problem is convex and admits a unique global optimum that can be found using standard solvers, whereas the Cauchy-loss readout requires iterative reweighted least squares on a non-convex objective and exhibits a substantially higher offline computation cost under the same EO framework (see Section 3.4).

Overall, the results on the $P_{\max}$ series are consistent with those on the MG benchmark. Both the EO-based hyperparameter search and the ϵ-insensitive SVR readout contribute to improving prediction performance: EO provides search stability comparable to or better than GA across datasets, while the SVR readout significantly enhances robustness on both the noisy $P_{\max}$ series ($p = 0.002$) and the noisy MG benchmark ($p = 0.006$). The comparison with the Cauchy-loss readout further confirms that the proposed ϵ-SVR readout achieves competitive accuracy with a substantially lower offline computation cost.

3.4. Computational Cost

To assess the practical feasibility of the proposed framework, Table 8 summarizes the offline computation time and the online one-step-ahead inference time for each method. All experiments are performed on an AMD Ryzen 7 6800H processor without parallel fitness

evaluation. EO and GA use the same nominal evaluation budget on each dataset: population × iterations = 600 fitness evaluations, each involving three-fold cross-validation, yielding approximately 1 800 ESN training runs per search.

Table 8. Computational cost comparison across methods and datasets.

Dataset	Method	Offline Computation (s)	Online Inference (μs/Step)
MG (noise-free)	GA-ESN	647.26 ± 54.16	—
MG (noise-free)	EO-ESN	657.36 ± 93.48	—
MG (noisy)	EO-ESN	82.24 ± 18.29	4.124 ± 0.366
MG (noisy)	EO-Robust-ESN	1008.68 ± 196.87	9.345 ± 8.601
$P_{\max}$ (noise-free)	GA-ESN	54.61 ± 29.19	—
$P_{\max}$ (noise-free)	EO-ESN	76.88 ± 27.12	—
$P_{\max}$ (noisy)	EO-ESN	22.04 ± 6.20	6.623 ± 0.638
$P_{\max}$ (noisy)	EO-Robust-ESN	70.08 ± 13.66	21.480 ± 9.607
$P_{\max}$ (noisy)	EO-Cauchy-ESN	240.49 ± 72.97	7.159 ± 1.219

The dashes (—) indicate that online inference time was not measured for that configuration. Offline computation time includes hyperparameter search, final training, and test prediction. Online inference measured with 50 warm-up test-sequence passes and 1000 timed passes.

As shown in Table 8, EO and GA require comparable offline computation times under the same evaluation budget: on the MG series, the two optimizers take approximately 650 s each; on the $P_{\max}$ series, EO takes 76.9 s versus 54.6 s for GA. This difference likely arises from the different reservoir sizes explored along the search path rather than from a fundamental complexity gap. The offline computation time of EO-Robust-ESN is higher than that of EO-ESN (e.g., 70 s versus 22 s on the noisy $P_{\max}$ series), which is expected because the search space is extended from three to five dimensions and the SVR solver is more expensive per evaluation than ridge regression. EO-Cauchy-ESN requires approximately 240 s, about 3.4 times the offline computation time of EO-Robust-ESN, consistent with its six-dimensional search space and iterative reweighted readout fitting. Nevertheless, these remain one-time offline costs.

The online one-step-ahead inference time, which determines real-time deployment feasibility, ranges from approximately 4 to 21 μs per step across all measured configurations. At an engine speed of 2400 rpm, a four-stroke combustion cycle lasts approximately 50 ms; for the three noisy $P_{\max}$ models, this leaves a margin of approximately 2300× to 7500× relative to the inference time. Among them, EO-Robust-ESN has the highest inference time at 21.5 μs per step, compared with 6.6 μs for EO-ESN and 7.2 μs for EO-Cauchy-ESN. This difference is attributable to the larger reservoir sizes found under the ϵ-SVR configuration, not to the readout layer computation itself: at deployment time, all three readouts compute a single linear operation $\mathbf{w}^\top \mathbf{z}$, so the choice of training loss function incurs no additional online cost.

3.5. Sensitivity Analysis

This subsection examines the sensitivity of prediction performance to the noise injection parameters and to the fixed model hyperparameters, all on the noisy $P_{\max}$ series.

3.5.1. Noise Parameter Sensitivity

To decouple the noise sensitivity from the hyperparameter optimization process, the model hyperparameters are fixed to the optimal values obtained under the reference noise condition (SNR = 20 dB, 5% impulse ratio, 4σ impulse magnitude) for each paired seed, and EO is not re-run. Ten paired noise seeds are used, with impulse positions strictly nested across ratios (1% ⊂ 3% ⊂ 5% ⊂ 10%) and impulse directions held constant across

magnitudes (2σ, 4σ, 6σ); both methods share the same noise sequences. Table 9 reports the MSE over 10 runs for each condition.

Table 9. Noise parameter sensitivity on the noisy $P_{\max}$ series (MSE, mean $\pm$ std).

Impulse Ratio	Impulse Magnitude	EO-ESN	EO-Robust-ESN
1%	4σ	5.416 ± 1.472	5.129 ± 1.821
3%	4σ	8.914 ± 11.162	5.256 ± 2.424
5%	4σ	13.161 ± 24.000	5.656 ± 2.296
10%	4σ	9.463 ± 11.630	5.881 ± 2.948
5%	2σ	7.947 ± 5.657	4.994 ± 2.550
5%	6σ	20.177 ± 36.433	5.251 ± 1.518

As shown in Table 9, EO-Robust-ESN maintains a stable MSE within a narrow range of 4.99–5.88 across all six conditions. As the impulse ratio increases from 1% to 10%, its MSE rises by only 14.7%; as the impulse magnitude increases from 2σ to 6σ, its MSE remains within a narrow range of 4.99–5.66. By contrast, the MSE of EO-ESN increases sharply with impulse magnitude (from 7.95 at 2σ to 20.18 at 6σ) and exhibits large standard deviations under most conditions, reflecting a high risk of extreme prediction failures on individual noise realizations. The MSE improvement from using the robust readout ranges from 5.3% at the mildest condition (1% ratio) to 74.0% at the most severe (6σ magnitude), indicating that the advantage of the ϵ-insensitive SVR readout becomes particularly pronounced as the impulse magnitude increases.

The EO-ESN results do not follow a strictly monotonic trend across impulse ratios; for instance, the 10% mean is lower than the 5% mean. This is because EO-ESN exhibits high variance under impulsive noise, where a few poorly performing runs can shift the mean substantially. By contrast, EO-Robust-ESN maintains consistent performance with substantially smaller variance under most conditions. Note that the absolute values differ from those in Table 6 because the hyperparameters are not re-optimized for each noise setting.

3.5.2. Penalty Coefficient μ

The fitness function (Equation (23)) includes a complexity penalty term $\mu \cdot N_x / N_{x,\max}$. Table 10 examines the effect of μ on the prediction performance of EO-Robust-ESN under the reference noise condition.

Table 10. Sensitivity of EO-Robust-ESN to the penalty coefficient μ on the noisy $P_{\max}$ series.

μ	MSE (Mean $\pm$ Std)	Mean N_x
0	4.696 ± 0.865	284.3
10^{-6}	4.696 ± 0.865	284.3
10^{-5}	4.696 ± 0.865	284.3
10^{-4}	4.696 ± 0.865	284.3
10^{-3}	4.837 ± 0.819	354.8

As shown in Table 10, both the prediction performance and the selected hyperparameters are identical for $\mu \leq 10^{-4}$. The maximum contribution of the penalty term in this range is 10^{-4}, which is several orders of magnitude smaller than the mean validation error (NRMSE ≈ 1.0); consequently, it does not alter the selected solutions or test performance across the tested seeds. At $\mu = 10^{-3}$, the MSE increases slightly to 4.837, indicating that a larger μ begins to alter the fitness landscape. The value adopted in this paper, $\mu = 10^{-6}$, therefore serves as a weak complexity tie-breaking term with no measurable effect on prediction performance.

3.5.3. Leaking Rate and Input Scaling

Table 11 examines the sensitivity of EO-Robust-ESN to the two fixed ESN hyperparameters under the reference noise condition, varying one parameter at a time while keeping the other at its adopted value. For each setting, EO is re-run to optimize the remaining hyperparameters.

Table 11. Sensitivity of EO-Robust-ESN to α_{leak} and s_{in} on the noisy $P_{\max}$ series.

α_{leak}	s_{in}	MSE (Mean ± Std)
0.5	0.5	5.799 ± 2.602
0.8	0.5	**4.696 ± 0.865**
1.0	0.5	6.042 ± 2.253
0.8	0.1	4.981 ± 1.503
0.8	1.0	5.731 ± 2.995

The best-performing result is shown in bold.

As shown in Table 11, the adopted setting $\alpha_{\text{leak}} = 0.8$, $s_{\text{in}} = 0.5$ yields the lowest MSE and the smallest standard deviation. Deviations from this setting increase the MSE by approximately 6% to 29%, with the standard deviation also increasing in all cases. These results empirically support the parameter choices adopted in this paper and are consistent with the practical guidance that the leaking rate and input scaling should be selected according to the task dynamics [19].

4. Conclusions

This paper has proposed the EO-Robust-ESN framework for robust combustion parameter prediction in alternative-fuel engines, addressing two key limitations of conventional ESNs: hyperparameter sensitivity and the lack of robustness to noisy observations. These limitations become particularly critical in dual fuel combustion, where the complex nonlinear dynamics and inherent cycle-to-cycle variability place stringent demands on both model configuration and prediction reliability. The framework employs EO to automatically search for model hyperparameters and replaces the conventional squared loss with the ϵ-insensitive loss by adopting linear SVR to train the readout weights. Experiments on the MG chaotic time-series benchmark and a peak in-cylinder pressure series from a diesel–CNG dual fuel engine demonstrate that EO significantly outperforms GA and manual tuning in solution quality on the MG benchmark, and achieves comparable accuracy to GA on the $P_{\max}$ series while exhibiting consistently smaller run-to-run variance across both datasets, indicating higher search stability. The ϵ-insensitive SVR readout further reduces prediction errors under noisy conditions, with the MSE on the noisy $P_{\max}$ series reduced by approximately 42% compared with the ridge regression readout, confirming its effectiveness in suppressing the interference of measurement noise and anomalous combustion cycles. Additional analyses confirm that the proposed ϵ-SVR readout achieves accuracy comparable to the Cauchy-loss robust readout at a substantially lower offline computation cost, that the proposed framework permits real-time inference well within the engine cycle time budget, and that its prediction performance remains stable across a range of noise conditions. Sensitivity analyses further support the adopted fixed hyperparameter values. These results suggest that the proposed framework provides an effective data-driven tool for robust combustion prediction in alternative-fuel engines.

Several limitations of the current work should be noted. The inclusion of SVR hyperparameters extends the search space from three to five dimensions, increasing the offline optimization cost. As a metaheuristic, EO does not guarantee convergence to the global optimum. The engine-data validation is limited to a single operating condition,

500 consecutive engine cycles, and one-step-ahead prediction. Future work will extend the framework to multiple operating conditions and multi-step-ahead prediction, and explore online adaptive learning strategies that update the model as new engine cycles are acquired. Application to other alternative fuels such as hydrogen and methanol, as well as embedded deployment on engine control units, are also of interest.

Author Contributions: Conceptualization, S.P.; methodology, S.P.; software, S.P.; validation, S.P., X.T., X.W. and X.D.; formal analysis, S.P.; investigation, S.P. and T.X.; resources, X.W. and T.X.; data curation, S.P. and X.W.; writing—original draft preparation, S.P.; writing—review and editing, S.P., X.T. and X.D.; visualization, S.P.; supervision, X.T. and X.D.; project administration, X.T. and T.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The Mackey–Glass time-series data were generated numerically using the fourth-order Runge–Kutta method with parameters $\beta = 0.2$, $\gamma = 0.1$, $n = 10$, and $\tau = 17$, and are fully reproducible from the description provided in Section 3.2. The diesel–CNG dual fuel engine in-cylinder pressure data are publicly available at https://data.mendeley.com/datasets/2yvx7f9yfm/1 (accessed on 24 June 2026).

Conflicts of Interest: Xingquan Wang and Tao Xu were employed by the company [Shandong Key Laboratory of Advanced Engine Piston Assembly, Binzhou Bohai Piston Co., Ltd., Binzhou, China]. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Bae, C.; Kim, J. Alternative fuels for internal combustion engines. *Proc. Combust. Inst.* **2017**, *36*, 3389–3413. [CrossRef]
2. Pham, Q.; Park, S.; Agarwal, A.K.; Park, S. Review of dual-fuel combustion in the compression-ignition engine: Spray, combustion, and emission. *Energy* **2022**, *250*, 123778. [CrossRef]
3. Cheng, Q.; Ahmad, Z.; Kaario, O.; Larmi, M. Cycle-to-cycle variations of dual-fuel combustion in an optically accessible engine. *Appl. Energy* **2019**, *254*, 113611. [CrossRef]
4. Karagöz, Y.; Sandalcı, T.; Koylu, U.O.; Dalkılıç, A.S.; Wongwises, S. Effect of the use of natural gas–diesel fuel mixture on performance, emissions, and combustion characteristics of a compression ignition engine. *Adv. Mech. Eng.* **2016**, *8*, 1687814016643228. [CrossRef]
5. Stipic, M.; Basara, B.; Schmidt, S.J.; Adams, N.A. Tabulated chemistry combustion model for cost-effective numerical simulation of dual-fuel combustion process. *Energies* **2023**, *16*, 8040. [CrossRef]
6. Aliramezani, M.; Koch, C.R.; Shahbakhti, M. Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions. *Prog. Energy Combust. Sci.* **2022**, *88*, 100967. [CrossRef]
7. Zhou, L.; Song, Y.; Ji, W.; Wei, H. Machine learning for combustion. *Energy AI* **2022**, *7*, 100128. [CrossRef]
8. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef] [PubMed]
9. Liu, M.; Chen, L.; Du, X.; Jin, L.; Shang, M. Activated gradients for deep neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 2156–2168. [CrossRef] [PubMed]
10. Qin, C.; Liu, M.; Jin, L. Adaptive gradient activation for federated learning on consumer electronic devices. *IEEE Trans. Consum. Electron.* **2025**, *71*, 5489–5500. [CrossRef]
11. Zhang, J.; Jin, L.; Wang, Y. Collaborative control for multimanipulator systems with fuzzy neural networks. *IEEE Trans. Fuzzy Syst.* **2023**, *31*, 1305–1314. [CrossRef]
12. Zeng, Z.; Wang, J.; Liao, X. Global exponential stability of a general class of recurrent neural networks with time-varying delays. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **2003**, *50*, 1353–1358. [CrossRef]
13. Hu, X.; Wang, J. Solving pseudomonotone variational inequalities and pseudoconvex optimization problems using the projection neural network. *IEEE Trans. Neural Netw.* **2006**, *17*, 1487–1499. [CrossRef] [PubMed]
14. Li, J.; Cao, Y.; Xie, Z.; Jin, L. A k-winners-take-all (kWTA) network with noise characteristics captured. *IEEE/CAA J. Autom. Sin.* **2025**, *12*, 734–744. [CrossRef]
15. Cucchi, M.; Abreu, S.; Ciccone, G.; Brunner, D.; Kleemann, H. Hands-on reservoir computing: A tutorial for practical implementation. *Neuromorph. Comput. Eng.* **2022**, *2*, 32002. [CrossRef]
16. Sun, C.; Song, M.; Cai, D.; Zhang, B.; Hong, S.; Li, H. A systematic review of echo state networks from design to application. *IEEE Trans. Artif. Intell.* **2024**, *5*, 23–37. [CrossRef]

17. Lei, Z.; Meng, H.; Yang, J.; Liang, B.; Cheng, J. Time series prediction of aerodynamic noise based on variational mode decomposition and echo state network. *Appl. Sci.* **2025**, *15*, 7896. [CrossRef]
18. Tanaka, G.; Yamane, T.; Héroux, J.B.; Nakane, R.; Kanazawa, N.; Takeda, S.; Numata, H.; Nakano, D.; Hirose, A. Recent advances in physical reservoir computing: A review. *Neural Netw.* **2019**, *115*, 100–123. [CrossRef] [PubMed]
19. Lukoševičius, M. A practical guide to applying echo state networks. In *Neural Networks: Tricks of the Trade*, 2nd ed.; Montavon, G., Orr, G.B., Müller, K.R., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 659–686. [CrossRef]
20. Soltani, R.; Benmohamed, E.; Ltifi, H. Echo state network optimization: A systematic literature review. *Neural Process. Lett.* **2023**, *55*, 10251–10285. [CrossRef]
21. Chen, X.; Luo, X.; Jin, L.; Li, S.; Liu, M. Growing echo state network with an inverse-free weight update strategy. *IEEE Trans. Cybern.* **2023**, *53*, 753–764. [CrossRef] [PubMed]
22. Huber, P.J. Robust estimation of a location parameter. *Ann. Math. Stat.* **1964**, *35*, 73–101. [CrossRef]
23. Matzner, F. Hyperparameter tuning in echo state networks. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '22), Boston, MA, USA, 9–13 July 2022; pp. 404–412. [CrossRef]
24. Cerina, L.; Santambrogio, M.D.; Franco, G.; Gallicchio, C.; Micheli, A. EchoBay: Design and optimization of echo state networks under memory and time constraints. *ACM Trans. Archit. Code Optim.* **2020**, *17*, 1–24. [CrossRef]
25. Racca, A.; Magri, L. Robust optimization and validation of echo state networks for learning chaotic dynamics. *Neural Netw.* **2021**, *142*, 252–268. [CrossRef] [PubMed]
26. Hua, Z.; Zheng, Z.; Péra, M.C.; Gao, F. Statistical analysis on random matrices of echo state network in PEMFC system's lifetime prediction. *Appl. Sci.* **2022**, *12*, 3421. [CrossRef]
27. Zhong, S.; Xie, X.; Lin, L.; Wang, F. Genetic algorithm optimized double-reservoir echo state network for multi-regime time series prediction. *Neurocomputing* **2017**, *238*, 191–204. [CrossRef]
28. Chen, X.; Jin, L.; Li, S. An online learning strategy for echo state network. *IEEE Trans. Syst. Man Cybern. Syst.* **2024**, *54*, 644–655. [CrossRef]
29. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [CrossRef] [PubMed]
30. Jin, L.; Wei, L.; Li, S. Gradient-based differential neural-solution to time-dependent nonlinear optimization. *IEEE Trans. Autom. Control* **2023**, *68*, 620–627. [CrossRef]
31. Fan, J.; Jin, L.; Li, P.; Liu, J.; Wu, Z.G.; Chen, W. Coevolutionary neural dynamics considering multiple strategies for nonconvex optimization. *Tsinghua Sci. Technol.* **2025**. [CrossRef]
32. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2020**, *191*, 105190. [CrossRef]
33. Blázquez-García, A.; Conde, A.; Mori, U.; Lozano, J.A. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv.* **2021**, *54*, 1–33. [CrossRef]
34. Lukoševičius, M.; Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Comput. Sci. Rev.* **2009**, *3*, 127–149. [CrossRef]
35. Filzmoser, P.; Nordhausen, K. Robust linear regression for high-dimensional data: An overview. *Wiley Interdiscip. Rev. Comput. Stat.* **2021**, *13*, e1524. [CrossRef]
36. Shi, Z.; Han, M. Support vector echo-state machine for chaotic time-series prediction. *IEEE Trans. Neural Netw.* **2007**, *18*, 359–372. [CrossRef] [PubMed]
37. Shen, L.; Chen, J.; Zeng, Z.; Yang, J.; Jin, J. A novel echo state network for multivariate and nonlinear time series prediction. *Appl. Soft Comput.* **2018**, *62*, 524–535. [CrossRef]
38. Li, F.; Li, Y. Robust echo state network with Cauchy loss function and hybrid regularization for noisy time series prediction. *Appl. Soft Comput.* **2023**, *146*, 110640. [CrossRef]
39. Guo, Y.; Wang, F.; Chen, B.; Xin, J. Robust echo state networks based on correntropy induced loss function. *Neurocomputing* **2017**, *267*, 295–303. [CrossRef]
40. Smola, A.J.; Schölkopf, B. A tutorial on support vector regression. *Stat. Comput.* **2004**, *14*, 199–222. [CrossRef]
41. Chen, X.; Liu, M.; Li, S. Echo state network with probabilistic regularization for time series prediction. *IEEE/CAA J. Autom. Sin.* **2023**, *10*, 1743–1753. [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.