processes

MDPI

Article

Optimization of Combustion Chamber Structure in a Rotary Engine Based on a Fuzzy Neural Network and a Genetic Algorithm

Min Yue [1], Liangyu Li [2], Run Zou [1,*], Tiexiong Su [1], Nana Wang [3] and Huan Wen [1]

[1] School of Energy and Power Engineering, North University of China, Taiyuan 030051, China
[2] School of Mechanical and Electrical Engineering, North University of China, Taiyuan 030051, China
[3] Science and Technology Innovation Center, Beijing 100012, China
* Correspondence: zourun@nuc.edu.cn

Abstract: In order to solve the problems of the method involving the optimization of the traditional combustion chamber structure, which has a long computation cycle, high computation cost, and can easily fall into the local optimal solution, this paper refers to the concept of a fuzzy neural network in machine learning. This study proposes a method of combustion chamber structure optimization that uses a fuzzy neural network to prejudge the results of the fitness function before calculating it in order to reduce the periodicity of computation and improve computational accuracy. The validation results show that the combustion chamber structure optimization method proposed in this paper can effectively reduce the computational cost under the premise of guaranteeing optimization accuracy. Using the test function, compared with the traditional genetic algorithm, the average number of iterations at convergence is reduced by 28.59%, and the average number of calculations of the fitness function is reduced by 25.59%. When optimizing the combustion chamber structure, the peak pressure of the optimal combustion chamber structure is increased by 10.32%, the computational count is reduced by 23.33%, and the time consumed is reduced by 23.91% compared with the traditional genetic algorithm.

Keywords: rotary engine; combustion chamber structure; fuzzy neural network; fitness function

check for updates

Academic Editor: Cherng-Yuan Lin

Received: 3 November 2024
Revised: 2 December 2024
Accepted: 6 December 2024
Published: 5 January 2025

Citation: Yue, M.; Li, L.; Zou, R.; Su, T.; Wang, N.; Wen, H. Optimization of Combustion Chamber Structure in a Rotary Engine Based on a Fuzzy Neural Network and a Genetic Algorithm. *Processes* **2025**, *13*, 122. https://doi.org/10.3390/pr13010122

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

1. Introduction

Rotary engines have been widely used in many applications such as in airplanes, automobiles, and motorcycles for many years due to their simple structure, small size, light weight, and low vibration [1]. For the small size and high power-to-weight ratio powerplants required by small unmanned aerial vehicles and range-extended electric vehicles, which have rapidly developed in recent years, rotary engines offer a new alternative [2,3]. However, the long and narrow combustion chamber structure of rotary engines can cause problems such as low combustion efficiency and poor emissions, which constrain their further application [4–6]. Whether it is possible to iteratively optimize the combustion chamber structure of rotary engines in a fast and efficient manner is an important factor affecting their development [7–9].

Scholars have conducted extensive research on the optimization of combustion chambers in engines. Chen et al. [10] designed four different types of combustion chambers and utilized the orthogonal decomposition method to match the key parameters of the combustion system and the injection system, and analyzed the schemes and screened the optimal combinations according to different evaluation indexes. Zhao et al. [11] synthesized the uniform design method and regression analysis method to improve the dynamics and economy of rotary engines, and comprehensively considered a variety of parameters such as air

intake, the injection system, and combustion chamber shape. Additionally, the researchers carried out multi-objective and multi-parameter optimization, finally achieving the optimal combination of combustion system parameters. Hu et al. [12] took the indicated power as the optimization objective and carried out multi-parameter optimization matching through an orthogonal design method, finally obtaining a better-performing combustion system. Li et al. [13] used a comprehensive objective evaluation function to evaluate and screen the optimal combustion chamber profile and injector projection height. Bai et al. [14] examined the influence of key combustion parameters on the combustion and emission characteristics of a diesel engine based on the orthogonal design method. In order to address the problem of uneven gas flame length and outlet temperature in the cylinder of a micro-gas turbine, Zong et al. [15] created a structural optimization design for the combustion chamber head and doping holes.

However, these traditional optimization methods have greater objective function requirements, making it easy for the algorithm to fall into the local optimal solution when facing complex, non-convex, discontinuous, or non-analytic problems and failing to achieve the ideal effect. In order to solve the shortcomings of traditional optimization methods, a series of intelligent optimization algorithms have been proposed and applied in optimization design in the fields of architecture, transportation, machinery, etc. The mainstream intelligent optimization algorithms can be divided into three types [16], among which the algorithm based on evolutionary theory and represented by the genetic algorithm [17] is the most widely recognized. The second type comprises intelligent optimization algorithms based on the laws of physics, sociology, or other disciplines represented by simulated annealing algorithms [18]. The third type, the population-based intelligent optimization algorithm, is designed based on the behavior of animal herds in nature to forage for food or avoid enemies, as represented by gray wolf optimization [19].

However, the abovementioned intelligent optimization algorithms usually require a large number of iterations and the evolution of a population to achieve the expected results, which requires the consumption of a large amount of computational resources. These algorithms are expensive to develop and take a long time to reach the optimal solution. To solve these problems and improve the convergence speed, various improved intelligent optimization algorithms have been proposed [20], for example, for cloud environment task scheduling [21], flexible job-shop scheduling [22], and green fresh logistics [23]. However, the abovementioned algorithms are highly dependent on the superiority or inferiority of the initial population and are prone to irregular and inaccurate coding, as well as slow convergence and a poor optimization effect. To address these problems, the concept of an agent model that does not rely on simulation and only relies on the network structure to output computational results has been proposed, but has not been widely used because of its overly large training sample requirements and low accuracy.

The emergence of fuzzy neural networks provides a new approach in the development of intelligent optimization. As a product of the combination of fuzzy theory and neural networks, it brings together the advantages of neural networks and fuzzy theory, integrating learning, association, recognition, and information processing. Compared with traditional neural networks, it achieves great superiority in dealing with nonlinearity, fuzziness, and other problems, and it has considerable potential in intelligent information processing, encouraging an increasing number of experts and scholars to invest in this field and obtain fruitful research results.

In this paper, a method for combustion chamber structure optimization based on a fuzzy neural network is proposed by combining the research on fuzzy neural networks in machine learning. Different from the traditional optimization method that directly simulates structural combinations to obtain the fitness value, the optimization method

proposed in this paper uses a fuzzy neural network to prejudge the combinations of combustion chamber structures for classification before simulation. If the prejudged results are in line with expectations, then the structural combinations can be simulated and the result is used as a fitness value; otherwise, a fixed value is directly assigned as the fitness value, thus improving the optimization accuracy and reducing the duration of iteration. The optimization results of the fitness function optimization problem and the problem of combustion chamber structure optimization are compared with those of the traditional genetic algorithm to verify the correctness and accuracy of the optimization method proposed in this paper. The predictive classification module constructed using a fuzzy neural network in this study can be flexibly added to various optimization algorithms and has strong generality compared to other methods, speeding up the iteration process. The results of this study can provide new ideas for the optimization of the combustion chamber structure in rotary engines and has some reference value for practical engineering applications. In addition, the improved optimization algorithm proposed in this paper can optimize a wide range of combustion chamber structures, making it generalizable. The next step is to validate the improved optimization algorithm proposed in this paper on a real machine, so that its optimization effect can be experimentally verified and its correctness and accuracy can be further proven.

2. Establishment of Optimization Model Based on Fuzzy Neural Network–Genetic Algorithm

Similar to the optimization methods used for conventional engines, the current methods of combustion chamber structure optimization in rotary engines are generally used to determine a structural combination that satisfies the requirements through the regularization study of a single variable. The optimization process is shown in Figure 1. However, this method is only applicable to relatively simple optimization problems, and the combustion chamber structure in rotary engines is a complex optimization problem composed of multiple variables with similar influencing factors [24,25]. Solving this problem using the optimization method based on a regularization study may lead to a scenario in which the observed pattern of change during the second stage of the process is no longer applicable to the various parameter combinations in the third stage, thus leading to the failure of the optimization to meet expectations.

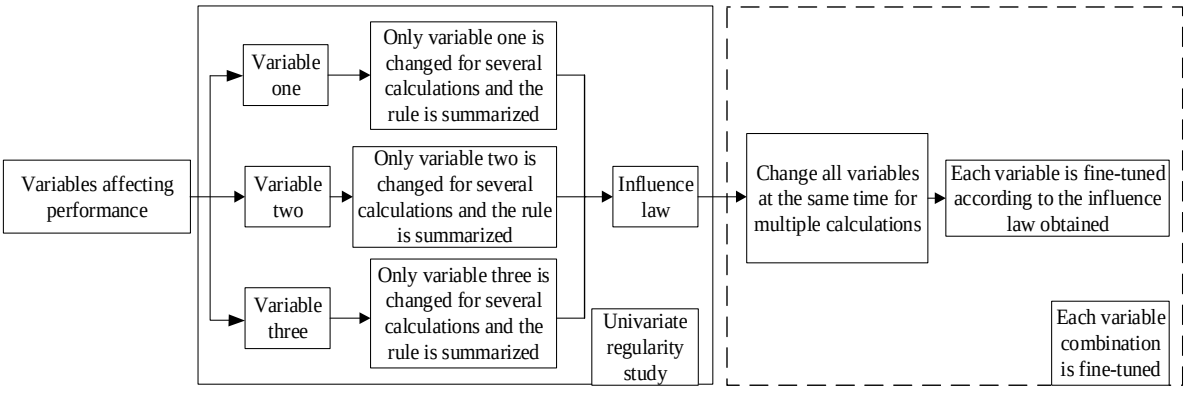


Figure 1. Optimization process based on regularization study.

In recent years, with the development of computer science, a series of intelligent optimization algorithms have emerged and made great progress in aerospace, intelligent manufacturing, route optimization, energy planning, and other fields. Compared with traditional optimization methods based on regularization studies, intelligent algorithms have characteristics such as a simple optimization process and high optimization accu-

racy [26]. Currently, there are three main intelligent optimization algorithms, with most widely recognized being the one based on evolutionary theory represented by the genetic algorithm (GA) [27]. This intelligent optimization algorithm imitates biological evolution or the physical laws of nature. Other types include the Artificial Immune System [28], which mimics the immune system's defenses against microbes and viruses, and Biogeographic-based Optimization (BBO) [29], which mimics the law of biological migration in the natural environment. Another widely used method is the intelligent optimization algorithm based on the laws of physics, sociology, or other disciplines, and includes the Spring Search Algorithm (SSA), which mimics the law of spring motion [30]; the Gravity Search Algorithm (GSA), which mimics the law of mutual attraction among substances [31]; and the Teaching–Learning-based Optimization (TLBO) algorithm, which mimics the teacher–student interaction process [32]. The population-based intelligent optimization algorithm is designed according to the behavior of animals during foraging or avoiding enemies in nature, which was first proposed by Gerardo Beni and Jing Wang in 1989. Typical examples include Ant Colony Optimization (ACO), which mimics the foraging behavior of ants [33]; Particle Swarm Optimization (PSO), which mimics the foraging behavior of birds [34]; and the Zebra Optimization Algorithm (ZOA) [35], which simulates the behavior of zebras avoiding predators. The flow of the optimization process for a typical intelligent optimization algorithm is shown in Figure 2.

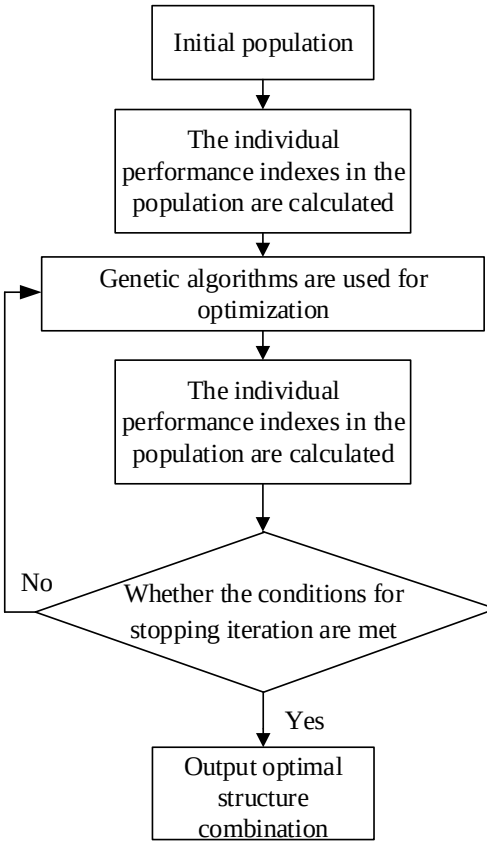


Figure 2. Structure optimization flow chart.

The abovementioned intelligent optimization algorithms have made great progress in terms of convergence speed, optimization accuracy, and global search capability. However, when the structural parameters of various components are optimized, all of them have the problem of a long optimization time. Taking the traditional genetic algorithm as an example, when the initial population number is 50, 150 simulation computations are required for

three iterations of optimization, which takes a total of 75 days, clearly deviating from the original intention of using a fast intelligent optimization algorithm.

To solve the abovementioned problems, in this section, we propose a fuzzy neural network–genetic algorithm-based optimization method for the combustion chamber structure, which is based on traditional genetic algorithms combined with the fuzzy classification of a fuzzy neural network (FNN) [36–38]. Figure 3 shows the optimization flow chart. Different from other methods that utilize optimization algorithms to optimize the combustion chamber structure, the method based on the fuzzy neural network–genetic algorithm proposed in this paper uses a trained fuzzy neural network at first to prejudge and classify the combinations of combustion chamber structure parameters when performing simulations. If the expected peak pressure is lower than a certain value, then no simulations are performed and a lower peak pressure will be assigned to it as a performance index of the structural parameter combinations. If the expected peak pressure is higher than a certain value, then the simulation is performed and the calculated peak pressure is used as a performance index of the structural parameter combinations. The fuzzy neural network takes the combination of the combustion chamber structure's parameters as the network input, and whether the peak pressure is greater than a certain value as the network output. The network is then trained with the relevant data established and calculated in Section 3. The poorer data are replaced by a value, and the convergence speed can be improved when computing with genetic algorithms by shortening the number of convergent generations and the computational time of the fitness function, thus saving computational time and improving the computational accuracy at the same time.

A fuzzy neural network is a kind of multi-layer forward neural network that combines fuzzy theory and neural networks. When processing information, the network can achieve a larger processing range and faster information processing speed, so the self-learning ability and mapping of the network are relatively high. Similar to the backpropagation neural network (BPNN), its training process is divided into two parts: forward propagation and backpropagation. Professor Zaden, the founder of fuzzy theory, once gave an example of parking when introducing fuzzy theory. Even a novice can easily park a car between two other cars after several practice exercises, but a computer that is good at finding accurate values must establish a model with great difficulty. In real-life applications, there are often a large number of fuzzy problems, some of which can certainly be solved using high-speed computers, but a large number of inaccurate controls can be solved simply by using fuzzy judgment, experience, and reasoning. Traditional solutions based on accurate mathematical models are naturally flawed. Fuzzy theory and artificial neural networks have been developed to solve these problems [39,40]. Fuzzy neural networks are combinations of fuzzy theory and artificial neural networks, giving full play to the advantages of each approach, compensating for the shortcomings of each other, and exhibiting an excellent ability to deal with large-scale fuzzy application problems.

Fuzzy neural networks combine fuzzy systems and neural networks, the essence of which is to change the input of the neural network into fuzzy input signals and fuzzy weights after the processing of the fuzzy system, and to defuzzify the output of the neural network, which is called intuitively effective value. Specifically, in fuzzy neural networks, the input and output of the neural network represent the input and output of the fuzzy system, and the membership function and fuzzy rules of the fuzzy system are added to the hidden nodes of the neural network to give full play to the parallel processing ability of the neural network and the reasoning ability of the fuzzy system [41]. Figure 4 shows the structure of fuzzy neural networks.

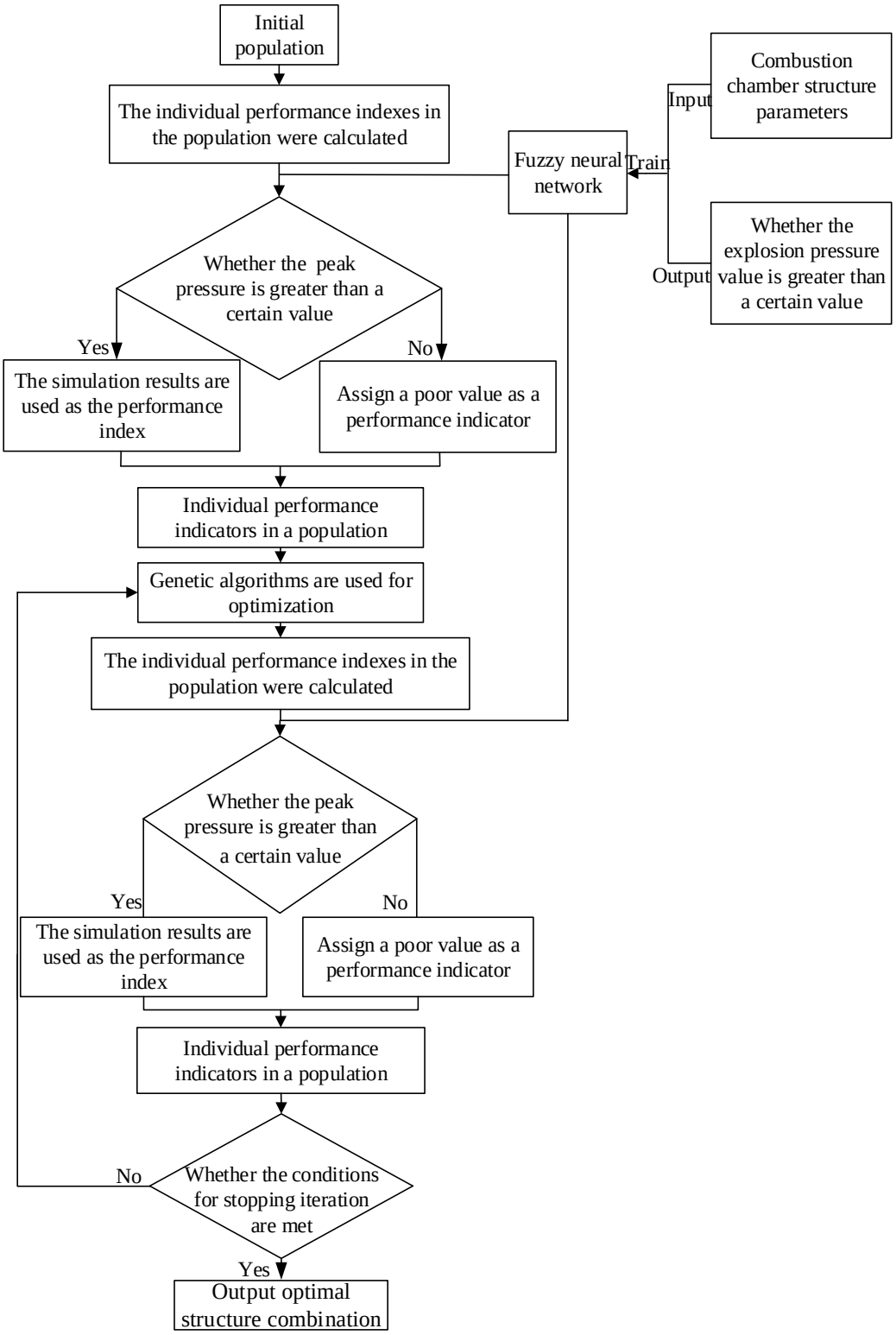


Figure 3. Flow chart of combustion chamber structure optimization based on fuzzy neural network–genetic algorithm.

In Figure 4, the first layer is the input layer, and each node is directly connected with each component of the input vector x_i, transmitting the input value $X = (x_1, x_2, x_3, \ldots, x_n)^T$ to the next layer. The number of nodes in this layer $N_1 = n$, where n is the input vector dimension, that is, the number of sample features.

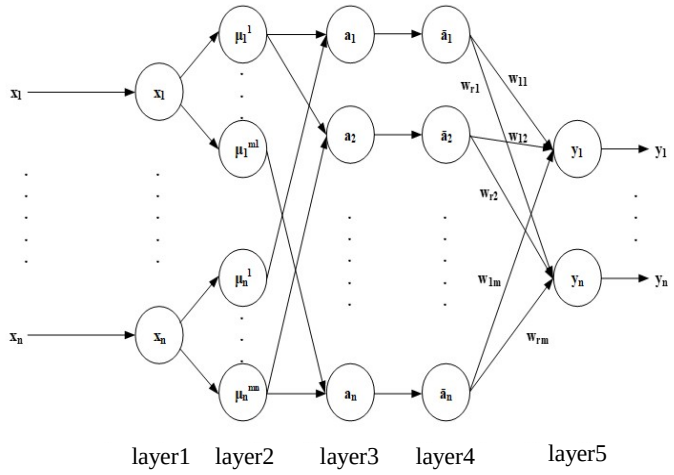


Figure 4. Schematic diagram of fuzzy neural network structure.

The second layer is the fuzzy layer, where each input component corresponds to a set of nodes, and the node number m_i is the number of fuzzy hierarchies relative to x_i, where each node represents a fuzzy speech variable value. The role of each node in the second layer is to calculate the membership function of each component of the input vector belonging to the fuzzy set of speech variables $\mu_{ij}(i = 1, 2, \ldots, n; j = 1, 2, \ldots, N_2)$. Here, n is the dimension of the input, and m_i is the number that needs to be divided in the fuzzy rule, so it can be considered that there are a total of n membership function groups and each group has m_i membership functions. The membership function can be expressed as:

$$\mu_{ij} = e^{\frac{-(x_i - c_{ij})^2}{\sigma_{ij}^2}} \tag{1}$$

In the formula, c_{ij} and σ_{ij}^2 represent the center value and width value of the membership function, respectively.

The total number of nodes in the second layer can be expressed as:

$$N_2 = \sum_{i=1}^{n} m_i \tag{2}$$

Each node in the third layer represents a fuzzy rule, which is mainly used to calculate the practicality of each rule. The membership functions of n groups in the second layer are combined together without repetition by taking one membership function from each group to form the nodes in the third layer, that is, the following two expressions:

$$a_j = \mu_1^{i_1} \mu_2^{i_2} \ldots \mu_n^{i_n}$$

Alternatively, it can be expressed in the following form:

$$\begin{gathered} a_j = \min\{\mu_1^{i_1}, \mu_2^{i_2}, \ldots, \mu_n^{i_n}\} \\ i_1 \in \{1, 2, \ldots, m_1\}, i_2 \in \{1, 2, \ldots, m_2\}, \ldots, i_n \in \{1, 2, \ldots, m_n\} \end{gathered} \tag{3}$$

The total number of nodes in the third layer can be expressed as:

$$N_3 = \prod_{i=1}^{n} m_i \tag{4}$$

For a given input, only those linguistic variable values that are near the input point have large membership degree values, and the linguistic variable values that are far away

from the input point have small membership degree values (Gaussian membership functions) or zero membership values (Triangular membership functions). When the membership function is small (for example, less than 0.05), it is generally close to 0. Therefore, only a few nodes in a_j have non-0 outputs, while most of the nodes have 0 outputs, which is similar to some neural networks using the principle of local approximation.

The number of nodes in the fourth layer is the same as that in the third layer, which is $N_4 = N_3$. It normalizes the applicability of each rule, i.e.:

$$\overline{a_j} = \frac{a_j}{\sum_{i=1}^{m} a_i} \tag{5}$$

Similar to the third layer, only a small number of nodes in this layer output large values and the output of most nodes is close to 0.

The fifth layer is the output layer used to achieve a clear calculation, namely:

$$\begin{gathered} y_i = \sum_{j=1}^{m} w_{ij}\overline{a_j} \\ i = 1, 2, \ldots, r \end{gathered} \tag{6}$$

Here, r is the expected number of categories, and w_{ij} is the central value of the j_{th} membership function of y_i. Its vector form is shown in the following equation:

$$\begin{gathered} Y = W\overline{a} \\ Y = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_r \end{bmatrix} \\ W = \begin{bmatrix} w_{11} & \cdots & w_{1m} \\ \vdots & \ddots & \vdots \\ w_{r1} & \cdots & w_{rm} \end{bmatrix} \\ \overline{a} = \begin{bmatrix} \overline{a_1} \\ \overline{a_2} \\ \cdot \\ \cdot \\ \cdot \\ \overline{a_m} \end{bmatrix} \end{gathered} \tag{7}$$

Similar to common neural networks with other structures, many parameters of fuzzy neural network model structures need to be predetermined, such as the fuzzy hierarchies number m_i for each input component x_i or the selection of the membership function form. The main parameters to be learned are the connection weight w_{ij} of the last layer, the central value c_{ij}, and the width σ_{ij}^2 of the membership function of the second layer. Figure 5 shows the relationships between each node.

The node in Figure 5 is the j_{th} node of the q layer in the fuzzy neural network. The input f^q and output x_i^q of this node can be expressed as:

$$\begin{gathered} f^q = \left(x_1^{q-1}, x_1^{q-1}, \ldots, x_1^{q-1}; x_1^{q-1}, x_1^{q-1}, \ldots, x_1^{q-1}\right) \\ x_i^q = g^q f^q \end{gathered} \tag{8}$$

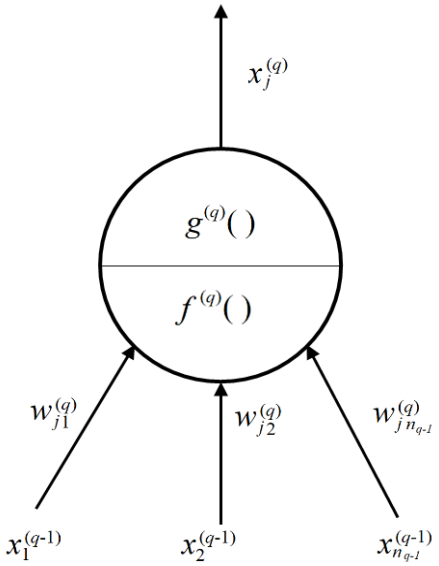


Figure 5. Node relationship diagram of fuzzy neural network.

For general neuronal nodes, there are usually:

$$\begin{aligned} f^q &= \sum_{i=1}^{n_{q-1}} w_{ij}^q x_i^{q-1} \\ x_i^q &= \frac{1}{1+e^{-\mu f^q}} \end{aligned} \tag{9}$$

In this paper, the Gaussian function is selected as the membership function, and then the node function of the first layer can be expressed as:

$$\begin{aligned} f_i^1 &= x_i^0 = x_i \\ x_i^1 &= g_i^1 = f_i^1 \\ i &= 1, 2, \ldots, n \end{aligned} \tag{10}$$

The second layer is:

$$\begin{aligned} f_{ij}^2 &= -\frac{\left(x_i^1 - c_{ij}\right)^2}{\sigma_{ij}^2} \\ x_{ij}^2 &= e^{-\frac{\left(x_i^1 - c_{ij}\right)^2}{\sigma_{ij}^2}} \\ i &= 1, 2, \ldots, n \quad j = 1, 2, \ldots, m_i \end{aligned} \tag{11}$$

The third layer is:

$$f_j^3 = \min\{x_{1i_1}^{i_1}, x_{2i_2}^{i_2}, \ldots, x_{ni_n}^{i_n}\}$$

Alternatively, it can be expressed in the following form:

$$\begin{aligned} f_i^3 &= x_{1i_1}^{i_1} x_{2i_2}^{i_2} \ldots x_{ni_n}^{i_n} \\ x_j^3 &= f_j^3 \\ i_1 \in \{1, 2, \ldots, m_1\}, i_2 \in \{1, 2, \ldots, m_2\}, \ldots, i_n &\in \{1, 2, \ldots, m_n\} \\ j &= (1, 2, \ldots, m) \\ m &= \prod_{i=1}^{n} m_i \end{aligned} \tag{12}$$

The fourth layer is:

$$\begin{gathered} f_j^4 = \frac{x_j^3}{\sum_{i=1}^m x_i^3} \\ x_j^4 = f_j^4 \\ j = (1, 2, \ldots, m) \end{gathered} \tag{13}$$

The fifth layer is:

$$\begin{gathered} f_i^5 = \sum_{j=1}^m w_{ij} x_j^4 \\ x_i^5 = f_i^5 \\ i = (1, 2, \ldots, r) \end{gathered} \tag{14}$$

Then, the error cost function E is:

$$E = \frac{\sum_{i=1}^r (y_{di} - y_i)^2}{2} \tag{15}$$

where y_{di} is the expected output and y_i is the actual output.

In this section, an error reversal algorithm similar to a BP neural network is used to design and adjust the parametric learning algorithm. For the fifth layer, there are:

$$\begin{gathered} \delta_i^5 = -\frac{\partial E}{\partial f_i^5} = y_{di} - y_i \\ \frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial f_i^5} * \frac{\partial f_i^5}{\partial w_{ij}} = -(y_{di} - y_i) x_j^4 \end{gathered} \tag{16}$$

For the fourth to second layers, there are:

$$\begin{gathered} \delta_j^4 = -\frac{\partial E}{\partial f_j^4} = \sum_{i=1}^r \delta_i^5 w_{ij} \\ \delta_j^3 = -\frac{\partial E}{\partial f_j^3} = \frac{\delta_i^4 \sum_{i=1, i \neq j}^m x_i^3}{\left(\sum_{i=1}^m x_i^3\right)^2} \\ \delta_j^2 = -\frac{\partial E}{\partial f_j^2} = \sum_{k=1}^n \delta_k^3 s_{ij} e^{-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}} \end{gathered} \tag{17}$$

When f^3 uses the minimum operation function, then, when $g_{ij}^2 = \mu_i^j$ is the minimum of the k_{th} node:

$$s_{ij} = 1 \tag{18}$$

When it is not:

$$s_{ij} = 0 \tag{19}$$

When f^3 uses the multiplicative function, then, when $g_{ij}^2 = \mu_i^j$ is the minimum value of the K_{th} node:

$$s_{ij} = \prod_{i=1, i \neq j}^n \mu_i^j \tag{20}$$

When it is not:

$$s_{ij} = 0 \tag{21}$$

For the second layer, the first step degree can be expressed as:

$$\begin{aligned} \frac{\partial E}{\partial c_{ij}} &= -\delta_{ij}^2 \frac{2\left(x_i - c_{ij}\right)}{\sigma_{ij}^2} \\ \frac{\partial E}{\partial \sigma_{ij}} &= -\delta_{ij}^2 \frac{2\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^3} \end{aligned} \tag{22}$$

The learning algorithm for parametric adjustment can be obtained as follows:

$$\begin{aligned} w_{ij}(k+1) &= w_{ij}(k) + \tau \frac{\partial E}{\partial w_{ij}} \\ c_{ij}(k+1) &= c_{ij}(k) + \varphi \frac{\partial E}{\partial c_{ij}} \\ \sigma_{ij}(k+1) &= \sigma_{ij}(k) + \omega \frac{\partial E}{\partial \sigma_{ij}} \\ i = (1, 2, \ldots, r) \quad & j = (1, 2, \ldots, m_i) \end{aligned} \tag{23}$$

Where τ, φ, and ω are the learning efficiency of connection weight, center, and width, respectively; k is the number of iterations.

The fuzzy neural network is established according to the above formula. When the actual output of the network is consistent with the ideal output, this indicates the end of training. Otherwise, the parameters of each layer are corrected by the backpropagation of the error until the error falls within the required range.

3. Optimization Model Analysis and Research Based on Fuzzy Neural Network–Genetic Algorithm

In order to verify the optimized accuracy and optimized speed of the optimization model based on the fuzzy neural network–genetic algorithm proposed in this paper, this section establishes the optimized model as described in the previous section and uses the fitness function as shown in Equation (24) as the test function.

$$\begin{gathered} y = f(x_1, x_2) = \sin(x_1) + \cos(x_2) + 0.1 \times x_1 + 0.1 \times x_2 \\ x_1 \in [-10, 10] \quad x_2 \in [-10, 10] \end{gathered} \tag{24}$$

Here, x_1 and x_2 are the two inputs of the fitness function, and y is the output of the fitness function. An image of this fitness function is shown in Figure 6. From the figure, it can be seen that this function fluctuates greatly in the domain of definition and has more extremely large values; hence, it can be used as a performance test for the optimization algorithm. The improved optimization algorithm proposed in this paper, which utilizes a fuzzy neural network first to classify the combinations of structural parameters, is a single-objective optimization algorithm with the fitness function being the peak pressure.

To train the fuzzy neural network, 2000 sets of input combinations are taken randomly and their outputs are calculated within the defined domain, where 1000 sets of inputs and outputs are used as the training set and 1000 sets of inputs and outputs are used as the test set. The training label R of the fuzzy neural network is shown in Equation (25).

$$\begin{gathered} if \;\; y > \gamma \\ R = 1 \\ else \\ R = -1 \\ \gamma \in (\min(y), \max(y)) \end{gathered} \tag{25}$$

where γ is the classification threshold. The maximum number of iterations is set to 100, the number of input nodes is set to 2, the number of hidden layer nodes is set to 16, and the number of output nodes is set to 1. The training is performed at the classification threshold $\gamma = 0$.

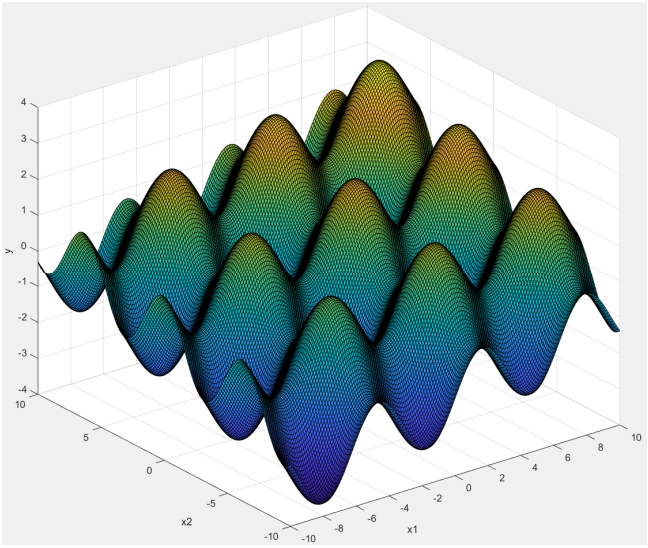


Figure 6. Fitness function diagram.

Figure 7 shows the comparison between the classified and expected outputs of the test set, where actual outputs less than zero are classified into the set labeled -1, and actual outputs greater than zero are classified into the set labeled 1. Table 1 shows the correct classification rate.

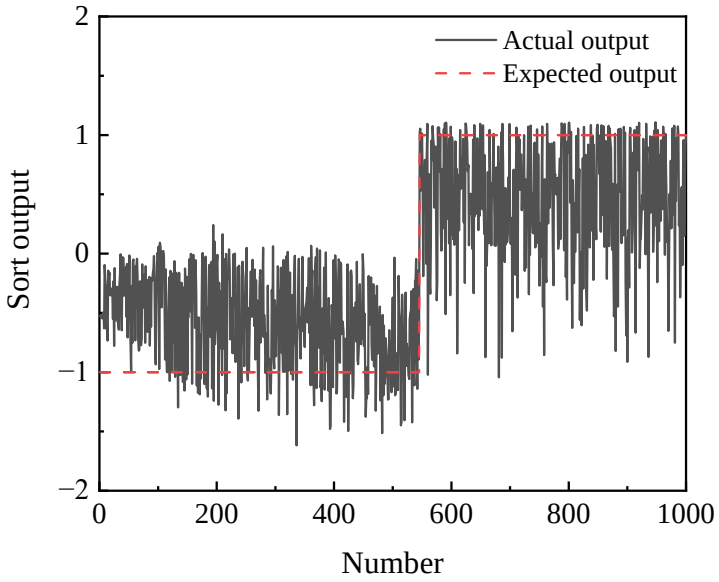


Figure 7. Comparison diagram for classification output of the test set.

Table 1. Classification accuracy rate of the test set.

Type	Number of Sets	Number of Incorrect Categories	Accuracy
Category labeled 1	455	65	85.71%
Category labeled -1	545	20	96.33%
Total	1000	85	91.50%

From the above graph, it can be seen that the use of fuzzy neural networks for prejudged classification has a high correct classification rate in general, which can be up to 91.5% on the test set. The correct classification rate is higher, up to 96.33%, for sets smaller than the classification threshold, but is lower for sets larger than the classification threshold at only 85.71%. Although the number of sets larger than the classification threshold is only 45.5% of the total number of sets, 65 out of the 85 wrong classifications, or 76.47%, occurred in the case of prejudging the classification of sets larger than the classification threshold.

For all classification thresholds γ in the range of $[-1.5, 1.5]$, training is performed at intervals of 0.01 and it is validated using the test set, which is performed 1000 times to obtain the correct classification rate at different classification thresholds, as shown in Figure 8.

As can be seen from Figure 8, the total correct classification rate floats at around 92% and hardly changes with the change in the classification threshold, but the rate for the sets larger and smaller than the classification threshold changes more with the classification threshold. In the range of $\gamma = [-1.5, 1.5]$, the correct classification rate for the sets smaller than the classification threshold increases slowly with the increase in the classification threshold, changing from 85% at $\gamma = -1.5$ to 95% at $\gamma = 1.5$. The correct classification rate for the sets larger than the classification threshold decreases slowly and then sharply as the classification threshold increases, decreasing sharply from 92% at $\gamma = -1.5$ to 40% at $\gamma = 1.5$.

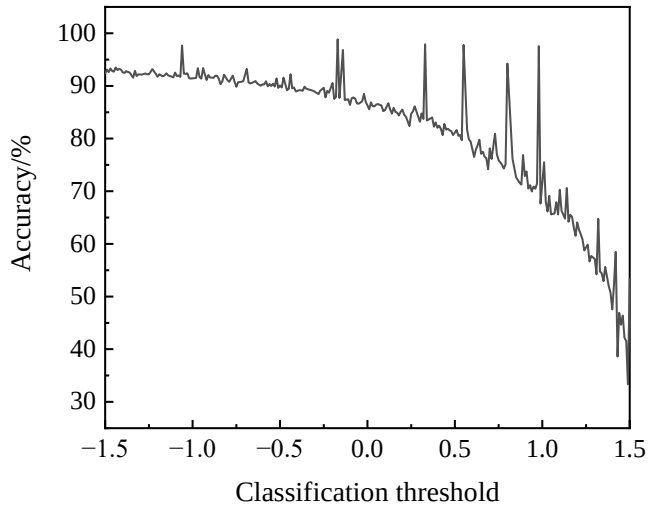
Ideally, when the correct classification rate for the fuzzy neural network is 100%, the prejudged classification module should prejudge and classify according to the method shown in Equation (26).

$$\begin{gathered} if\ f(x_1, x_2) > \gamma \\ y = f(x_1, x_2) \\ else \\ y = \gamma \end{gathered} \tag{26}$$

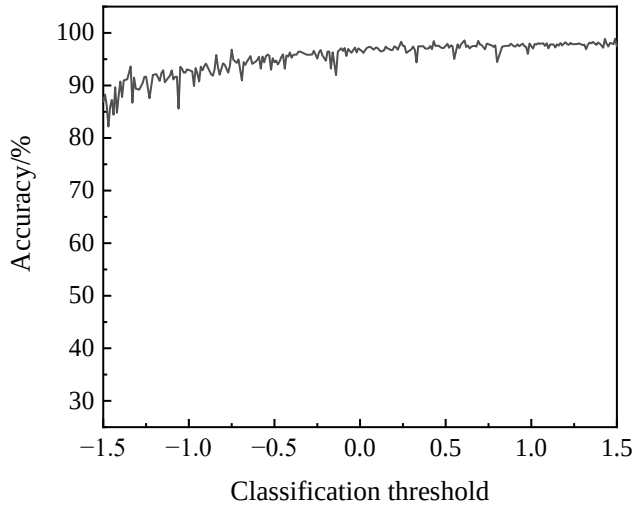
The optimization iterations for the fitness function in Equation (24) are run with the following parameters: taking the maximum value as the optimization objective, using one-point variation, two-point crossover, and the roulette strategy selection method, the initial number of populations is 100, the minimum number of single iterations is 40, the maximum number of single iterations is 10, the initial variation probability is 0.07, and the initial crossover probability is 0.7, using Equation (26) as a prejudged classification module. Shown in Figure 9 is the average number of converged generations when the traditional genetic algorithm converges to 3.4237 (hereinafter referred to as the convergence time) after performing 100 optimization calculations and the average number of converged generations when the genetic algorithm with the addition of the idealized prejudged classification module converges to 3.4237. Figure 10 shows the average number of fitness function calculations when the traditional genetic algorithm converges to 3.4237 after performing 100 optimization calculations and the average number of fitness function calculations when the genetic algorithm, with the addition of the idealized prejudged classification module, converges to 3.4237.

As can be seen from Figures 9 and 10, the average number of iterations and the average number of fitness function calculations when convergence is reached decreases as the classification threshold increases when the idealized prejudged classification module is added. The average number of iterations when the convergence is reached decreases from 20.10 at the classification threshold $\gamma = -1.5$ to 12.13 at $\gamma = 1.5$, which is a reduction of 7.15% and 43.39%, respectively, compared to 21.68 without the addition of the prejudged classification module. The average number of fitness function calculations when the convergence is reached decreases from 1994.62 at the classification threshold of $\gamma = -1.5$ to

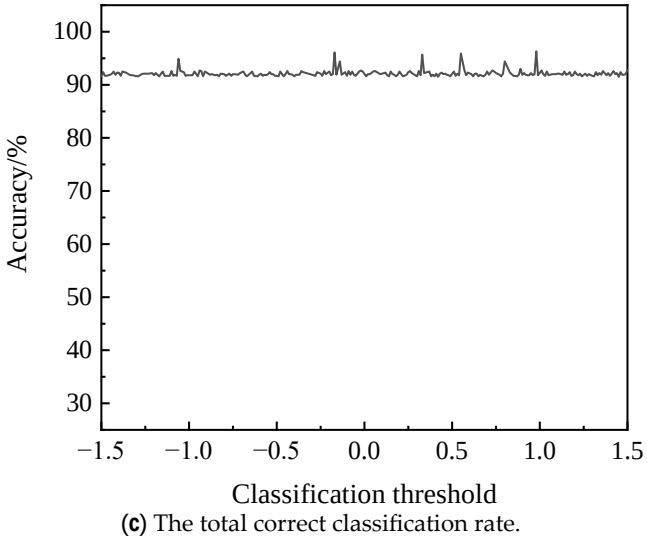
1126.26 at $\gamma = 1.5$, compared to 2168.50 without the addition of the prejudged classification module, which is a reduction of 8.01% and 48.06%, respectively.



(**a**) The correct classification rate for the set larger than classification threshold.



(**b**) The correct classification rate for the sets smaller than the classification threshold.



(**c**) The total correct classification rate.

Figure 8. The classification accuracy rate at different classification thresholds.

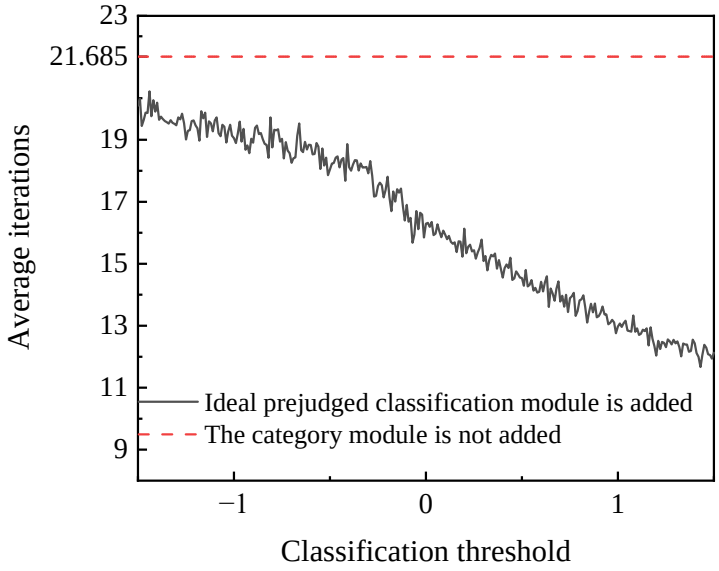


Figure 9. The average number of iterations when adding the ideal prejudged classification module.

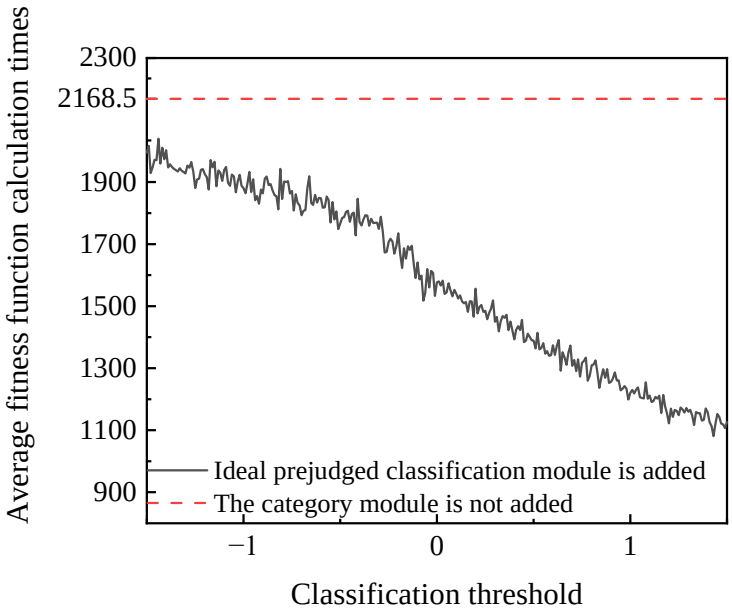


Figure 10. The average number of fitness function calculations when adding the ideal prejudged classification module.

However, the correct classification rate for the fuzzy neural network is not 100%. As can be seen from the figure, the total correct classification rate fluctuates around 92% in the range of $\gamma = [-1.5, 1.5]$, and its correct classification rate for sets larger and smaller than the classification thresholds will also change under different classification thresholds. The prejudged classification module is replaced by the trained fuzzy neural network mentioned above, and the fitness function in Equation (24) is optimized iteratively with the maximum value as the optimization objective. As shown in Figure 11, after 100 optimization calculations, the average number of converged generations with the traditional genetic algorithm converges to 3.4237, and the average number of converged generations with the genetic algorithm combined with the prejudged classification module converges to 3.4237. Figure 12 shows the average number of fitness function calculations when the traditional genetic algorithm converges to 3.4237 and those when the genetic algorithm combined with the prejudged classification module converges to 3.4237 after performing 100 optimization calculations.

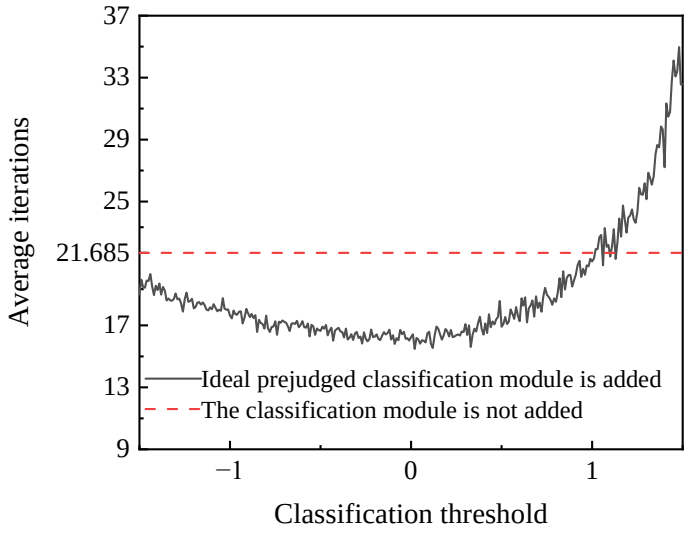


Figure 11. Average number of iterations during convergence.

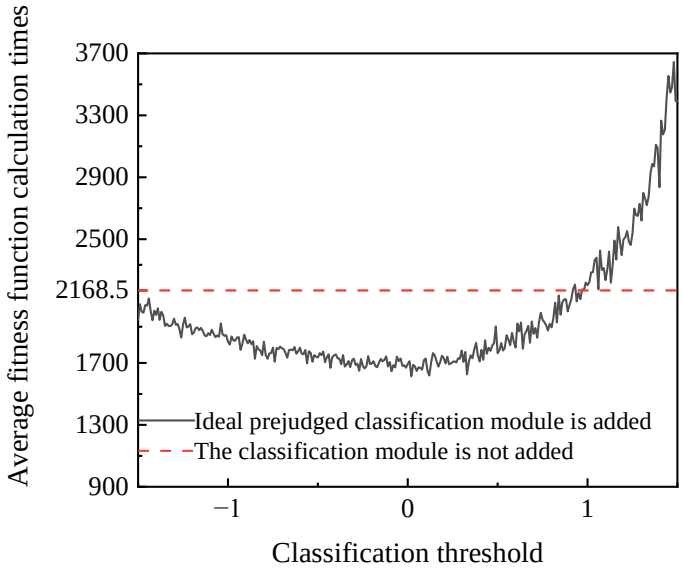


Figure 12. The average number of fitness function calculations.

From the figure, it can be seen that, unlike the ideal state, the average number of converged generations and the average number of fitness calculations at the time of convergence after the addition of the fuzzy neural network-trained prejudged classification module first decreases slowly and then increases sharply with the increase in the classification threshold. As the classification threshold increases, the average number of iterations first slowly decreases from 19.95 at $\gamma = -1.5$ to 15.48 at $\gamma = 0.02$, and then sharply increases to 32.57 at $\gamma = 1.5$; these values are the same as the number of iterations when no classification module is added at $\gamma = 1.11$. As the classification threshold increases, the average number of fitness function calculations first decreases slowly from 2079.92 at $\gamma = -1.5$ to 1613.58 at $\gamma = 0.02$, and then increases sharply to 3394.43 at $\gamma = 1.5$, and at $\gamma = 0.97$ is the same as the number of iterations when the classification module is not added.

From the above analysis, it can be seen that adding the prejudged classification module with an appropriate classification threshold can greatly improve the convergence speed of the optimization iteration of the genetic algorithm under the premise of guaranteeing the accuracy of the optimization, and can reduce the average number of fitness function calculations. When choosing a classification threshold of $\gamma = 0.02$, the average number of convergence iterations is 15.48, which is reduced by 28.59% compared with 21.68 when

the prejudged classification module is not used. The average number of fitness function calculations at convergence is 1613.58, which is reduced by 25.59% compared with 2168.5 when the prejudged classification module is not used. Therefore, the optimization model based on the fuzzy neural network–genetic algorithm proposed in this paper can improve the convergence speed by reducing the number of fitness function calculations under the premise of guaranteeing the accuracy of the optimization, so as to save the computational time. When selecting the classification threshold, this approach can ensure a high correct classification rate for sets larger and smaller than the classification threshold.

4. Analysis of Comprehensive Optimization Results Based on Fuzzy Neural Network–Genetic Algorithm

4.1. Establishment of Optimization Model of Combustion Chamber Structure in a Rotary Engine

The effectiveness of the fuzzy neural network–genetic algorithm proposed in this paper in optimizing the combustion chamber structure of a rotary engine required verification. In this section, the combustion chamber structure of a rotary engine is optimized iteratively based on the optimization model mentioned above, with the peak pressure taken as the optimization objective. Figure 13 shows the optimization flow chart.

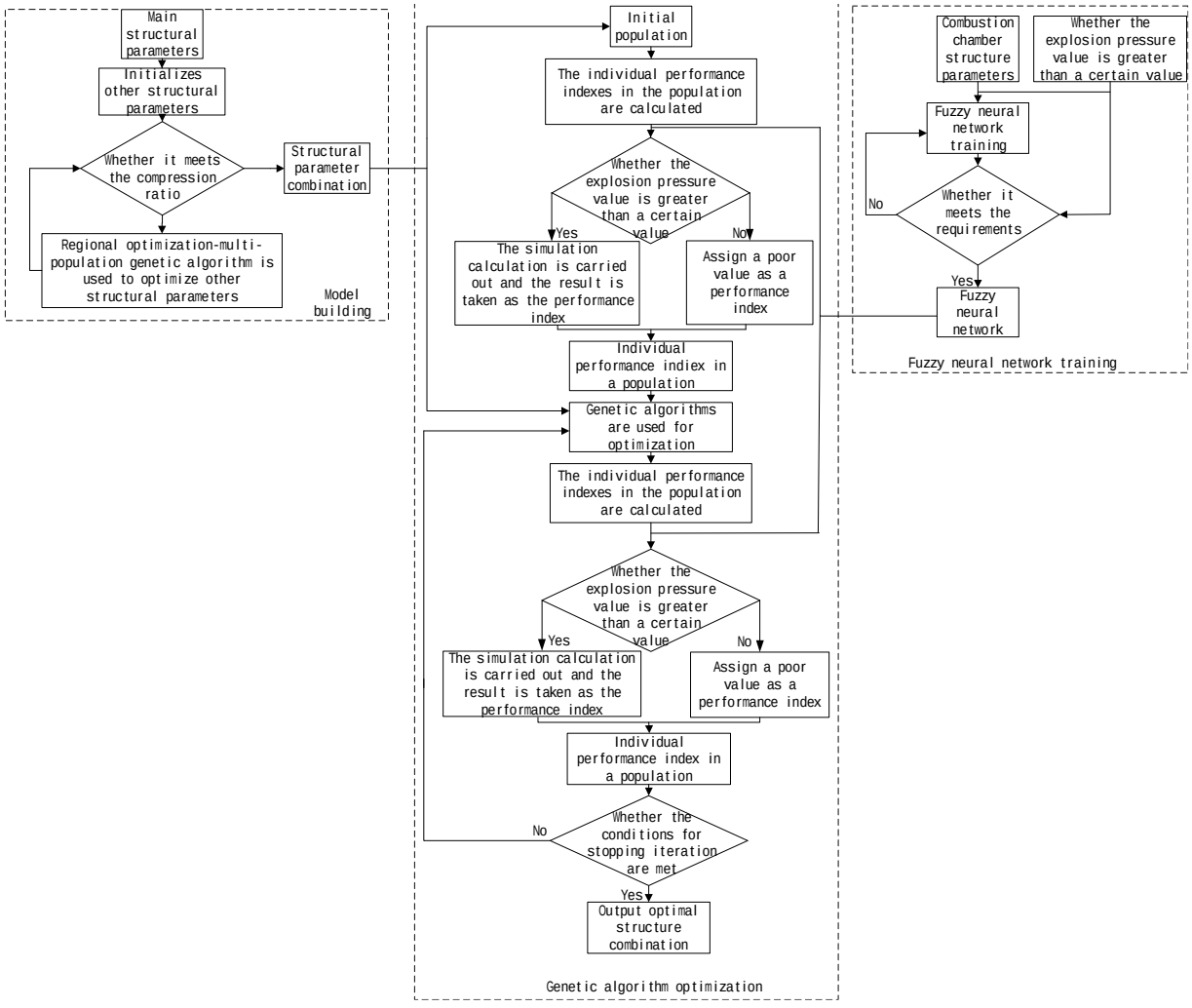


Figure 13. Optimization flow chart.

In Figure 13, Equation (26) is used as the classification label, the maximum number of iterations is set to 500, the number of input nodes is 8, the number of hidden layer nodes is 24, and the number of output nodes is 1. $\gamma = 3.5$ is taken as the classification threshold,

120 structural parameter combinations are used, eight important structural parameters are taken as inputs, and the peak pressure obtained by simulation is taken as the output. Then, 80 groups are randomly selected as the training set and the remaining 40 groups serve as test sets. Figure 14 shows a comparison between the test set classification output and the expected output. In this figure, an actual output less than 0 is classified as the set labeled −1, while an actual output greater than or equal to 0 is classified as the set labeled 1. Table 2 shows the correct classification rate.

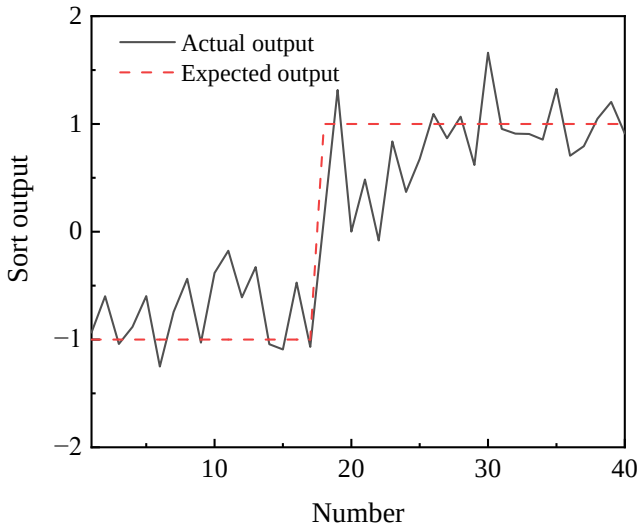


Figure 14. Classification output comparison diagram.

Table 2. Classification accuracy rate.

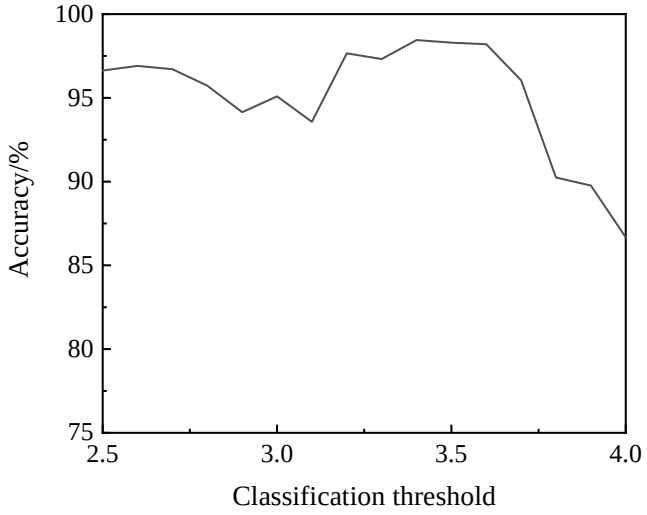
Type	Number of Sets	Number of Errors	Accuracy
Category labeled 1	23	1	95.65%
Category labeled −1	17	0	100%
Total	40	1	97.50%

From Figure 14 and Table 2, it can be seen that the use of fuzzy neural networks for prejudged classification has a high correct classification rate in general. Its correct classification rate on the test set reached 97.50%; the correct classification rate for the sets smaller than the classification threshold is higher, up to 100%, but the correct classification rate for the sets larger than the classification threshold is lower at 95.65%, and all of the classification errors occur in the case of prejudged classification for the sets larger than the classification threshold.

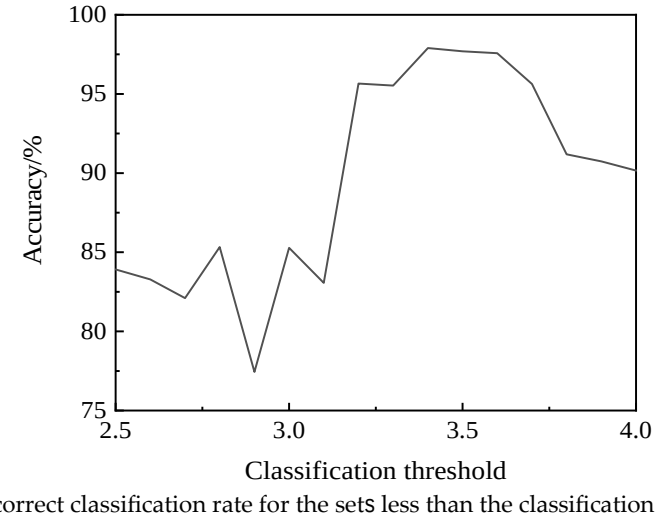
All of the classification thresholds γ in the range of $[2.5, 4]$ were conducted at an interval of 0.1 and verified using a test set. The test set was evaluated 1000 times to obtain the correct classification rate under the same classification threshold, as shown in Figure 15.

From Figure 15, it can be seen that the total correct classification rate increases and then decreases with the increase in the classification threshold, and the maximum correct classification rate of 98.21% occurs at $\gamma = 3.4$. In the range of $\gamma = [2.5, 4]$, the correct classification rate for the sets less than the classification thresholds increases and then decreases with the increase in the classification threshold. Additionally, the maximum correct classification rate of 97.9% occurs at $\gamma = 3.4$. The correct classification rate for the set greater than the classification threshold increases and then decreases with the increase in the classification threshold. Additionally, the maximum correct classification rate of 98.44% occurs at $\gamma = 3.4$. From the figure, it can be seen that when $\gamma = 3.6$, the fuzzy neural network can still maintain a high correct classification rate, with a total correct

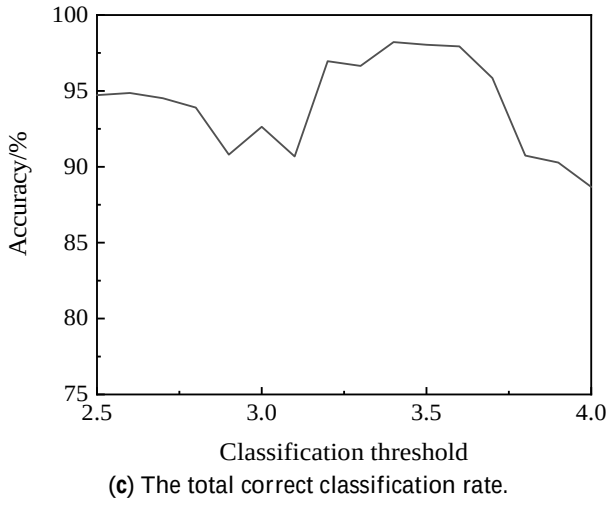
classification rate of 97.96% and 95.63% for the sets less than the classification threshold, and a correct classification rate of 96.04% for the sets greater than the classification threshold. According to the above analysis of the fuzzy neural network's correct classification rate and the performance of genetic algorithm, $\gamma = 3.6$ was chosen as the classification threshold.



(**a**) The correct classification rate for the sets larger than the classification threshold.



(**b**) The correct classification rate for the sets less than the classification threshold.



(**c**) The total correct classification rate.

Figure 15. Classification accuracy rate.

Taking the peak pressure as the optimization objective and using the single-point variation, two-point crossover, and roulette strategy selection method, the initial population number was 90, the maximum number of iterations was 3, the initial variation probability was 0.07, the initial crossover probability was 0.7, and the parameters of the combustion chamber structure of the rotary engine including the spoiler were optimized by using the optimization algorithms based on the fuzzy neural network–genetic algorithm proposed in this paper and the traditional genetic algorithm for optimization iterations.

4.2. Establishment and Verification of Three-Dimensional Model of Combustion Chamber in a Rotary Engine

4.2.1. Geometric Model and Calculation Domain of Rotary Engine

In this study, a single-spark-plug premixed rotary engine with a peripheral intake is taken as the object, and the method of optimizing the combustion chamber structure of a rotary engine based on a fuzzy neural network and genetic algorithm is investigated. In order to simplify the calculation, this section establishes a simplified 3D computational model of the rotary engine, as shown in Figure 16. Its rotational speed is 8000 r/min, and one working cycle is 1080° CA. Table 3 shows the structural parameters.

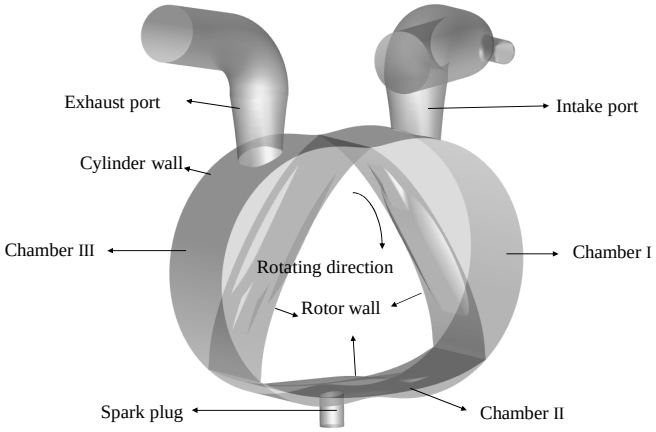


Figure 16. Computational domain model.

Table 3. Structural parameters of rotary engine.

Parameter Name	Numerical Value
Generating radius (mm)	60
Eccentricity (mm)	10
Width of rotor (mm)	42
Translation distance (mm)	1.45
Rated speed (r/min)	8000
Displacement (cc)	133
Compression ratio	10
Ignition advance angle	30° before top dead center
Ignition source	Single spark plug
Ignition energy (J)	0.04
Kernel diameter (mm)	0.8

4.2.2. Mathematical Model of Rotary Engine

The flow field simulated in this study is assumed to be transient compressible flow and the inlet flow is uniform. In this study, the CFD simulation software Converge studio 3.0 is used for transient solution and the finite volume method is used to solve the control conservation equations to obtain the required flow and combustion data to simulate the in-cylinder flow and combustion process of the rotor engine. The RNG K-εmodel

(Renormalization K-εGroup Model), which is able to take into account the rotary engine in-cylinder flow line of the vortex clusters as well as mechanical variations, improves the accuracy of the high-speed flow and makes the computational structure more accurate. SAGE version 9.1 (Software for the Analysis of Gas-phase chemistry and Aerosol formation in the Environment) is capable of describing, in detail, the phenomenon of in-cylinder flow and obtaining the results of the distribution and concentration change of each component in the combustion process of the rotary engine. It can improve the computational efficiency and shorten the computational time under the premise of computational accuracy, so as to accurately simulate the combustion process. The chemical reaction mechanism of fuel combustion is the PRF mechanism, which can be applied to different ratios of base fuels and fully considers the emissions. Its simplified mechanism contains 73 components and 296 reaction steps, which can well predict the ignition timing, flame speed, and concentration of main substances. Table 4 shows the mathematical models and Table 5 shows the boundary conditions of the rotary engine.

Table 4. Mathematical models and chemical mechanisms.

Purpose	Model Selection and Mechanism
Turbulence model	RNG K − ε model
Combustion model	SAGE model
Ignition model	Spark energy deposition model
Wall heat transfer model	Wall function model
Chemical reaction mechanism	PRF skeletal mechanism

Table 5. The boundary conditions of the rotary engine.

Boundary Region	Type	Temperature (K)	Pressure (Pa)
Intake port	Inflow	300	101,325
Inlet	Fixed wall	300	/
Exhaust port	Outflow	570	101,325
Exhaust	Fixed wall	550	/
Rotor	Moving wall	400	/
Spark plugs	Fixed wall	624	1,172,100
Spark electrode	Fixed wall	600	/

In this paper, the grid independence analysis of the simulation model is used to ensure the calculational accuracy, as shown in Figure 17 for the cylinder pressure variation curves of four different grid sizes. Under the consideration of the calculational accuracy and efficiency, a 2mm grid with AMR is selected, in which the number of encryption levels in the combustion chamber area is two and the number of encryption levels in the spark plug area is five. Figure 18 shows that, in the calculation, the total number of grids for the rotary engine is about 200,000 hexahedral cells.

In order to ensure the calculational accuracy of the rotary engine 3D CFD simulation model established in this paper, the combustion model under the original combustion chamber structure and the original ignition scheme is verified. Under the working conditions, the fuel is gasoline, the ignition time is -30° CA, the intake pressure is 101325 Pa, and the rotational speed is 8000 r/min; a comparison of the simulation results and test results for the cylinder pressure is shown in Figure 19. As can be seen from the figure, the overall agreement between the simulation results and the test results is good, and the maximum error is less than 5%. This shows that the 3D combustion model constructed in this study can accurately predict the combustion process of the rotary engine, and its simulation data are highly reliable, which can be used in the research for the method of optimizing the

combustion chamber structure of the rotary engine based on the fuzzy neural networks and genetic algorithms mentioned in this paper.

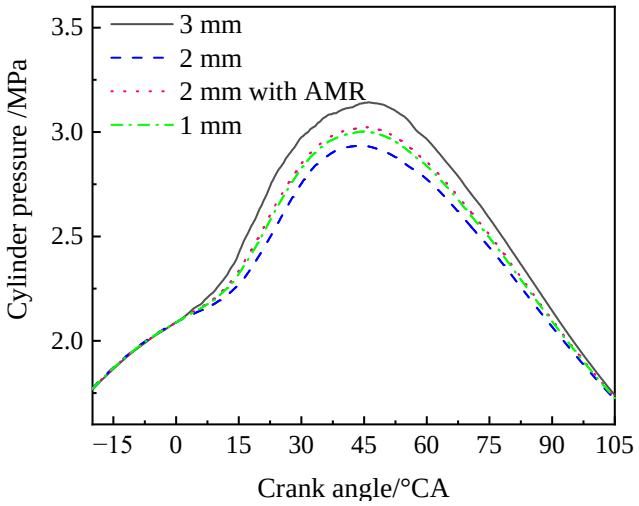


Figure 17. Variation curve of cylinder pressure under different grid quantities.

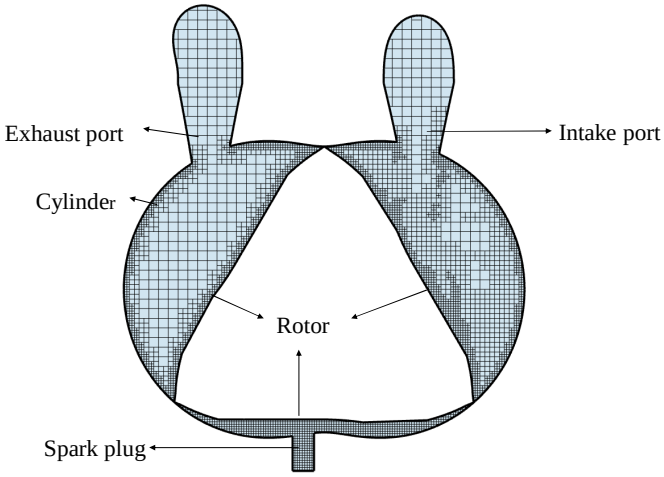


Figure 18. Grid model.

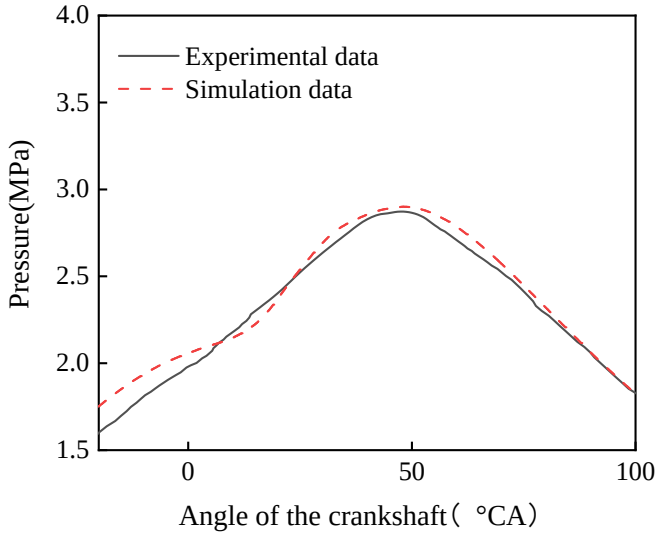
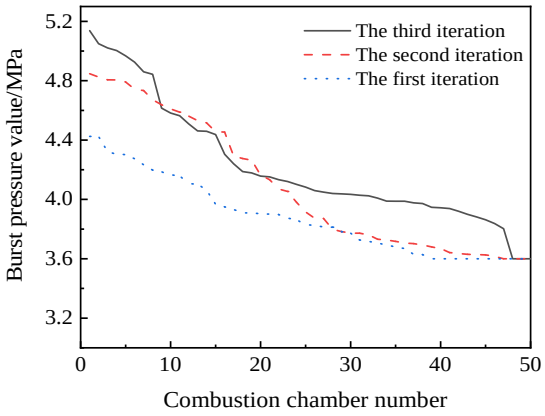


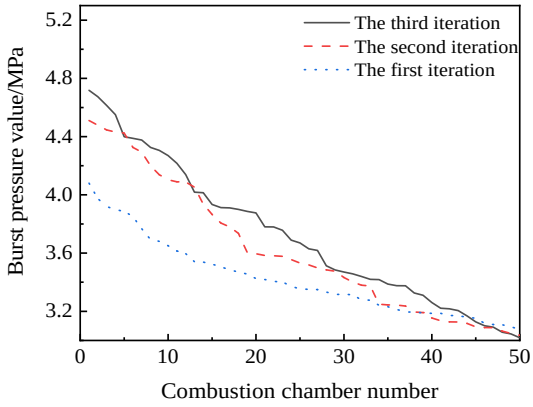
Figure 19. Comparison of simulation results and experimental results.

4.3. Structural Optimization Results and Analysis of Rotary Engine Combustion Chamber

The two optimization algorithms, iterated through 50 parameter combinations of the chamber structure per iteration, for a total of 150 combinations were evaluated. Subsequent simulation calculations yielded varying peak pressures. These pressures were then sorted in descending order, thereby generating the distribution diagram presented in Figure 20. Figures 21 and 22 show comparisons of the maximum peak pressure, the average peak pressure, and the peak pressure of the original combustion chamber for the three iterations of the two optimization algorithms.

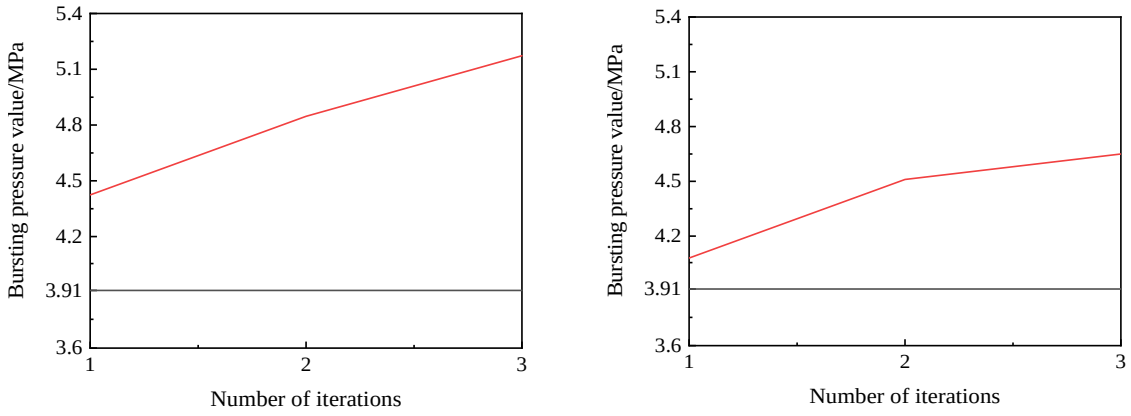


(**a**) Using improved optimization algorithms



(**b**) Using traditional optimization algorithms

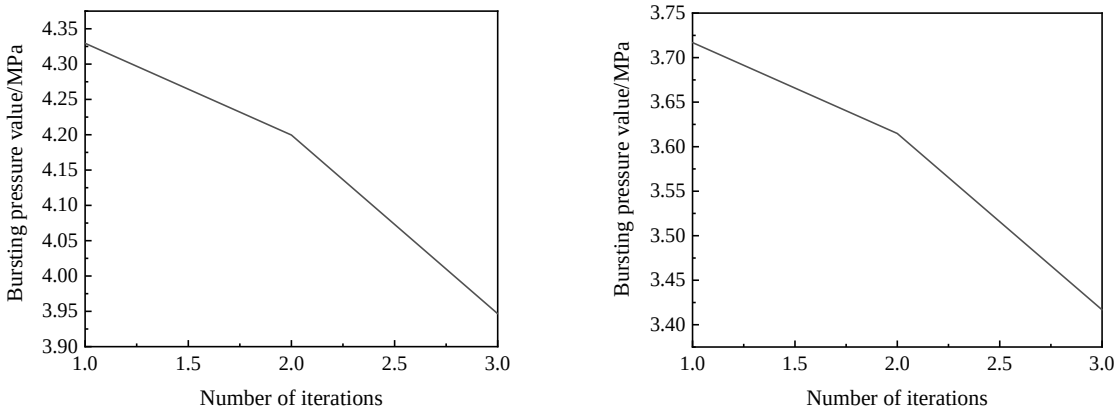
Figure 20. Distribution of burst pressure values for three iterations.



(**a**) Using improved optimization algorithms

(**b**) Using traditional optimization algorithms

Figure 21. Comparison of maximum burst pressure values for three iterations.



(**a**) Using improved optimization algorithms (**b**) Using traditional optimization algorithms

Figure 22. Comparison of average burst pressure values for three iterations.

From Figures 20–22, it is evident that the two optimization algorithms show a more significant improvement in the average and optimal performance of the population as the number of iterations increases. When using the optimization algorithm based on the fuzzy neural network–genetic algorithm proposed in this paper, the maximum peak pressure increased from 4.42 MPa in the first iteration to 5.13 MPa in the third iteration, which is an increase of 16.06%. Compared to the peak pressure of 3.91 MPa for the original combustion chamber, the maximum peak pressure increased by 13.04% for the first iteration, 23.78% for the second iteration, and 31.20% for the third iteration. When optimized by the traditional genetic algorithm, the maximum peak pressure increased by 9.09%, rising from 4.08 MPa in the first iteration to 4.51 MPa in the third iteration. Compared to the peak pressure of 3.91 MPa for the original combustion chamber, the maximum peak pressure increased by 4.34% for the first iteration, 15.34% for the second iteration, and 18.92% for the third iteration. It can be seen from the above analysis, compared to the peak pressure of the optimal combustion chamber obtained using the traditional genetic algorithm, that the peak pressure obtained using the optimization algorithm based on the fuzzy neural network–genetic algorithm proposed in this paper increased from 4.65 MPa to 5.13 MPa, which is an improvement of 10.32%.

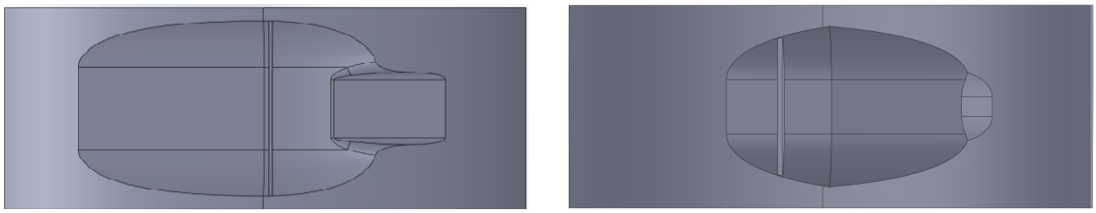
Table 6 shows a comparison of the number of simulation calculations and time consumption between the two optimization methods. From the table, it can be seen that, compared with the traditional genetic algorithm, optimization using the fuzzy neural network–genetic algorithm proposed in this paper has significant advantages in efficiency, and the number of simulation calculations required for its three iterations is reduced from 240 to 184, which is a reduction of 23.33%, and the time consumption is reduced from 481 to 366 h, which is a reduction of 23.91%. Therefore, adding the fuzzy neural network classification module can significantly improve the optimization efficiency. Moreover, the prejudged classification module constructed using the fuzzy neural network can be flexibly added to various optimization algorithms, possessing strong universality compared with other methods for accelerating the iteration.

A comparison chart between the combustion chamber structure corresponding to the combination of structural parameters with the largest peak pressure in the third iteration (hereinafter referred to as the optimal combustion chamber) and the original combustion chamber structure is shown in Figure 23, and the structural parameters of the optimal combustion chamber and the original combustion chamber are shown in Table 7. From the above diagrams, it can be seen that, compared with the original combustion chamber, the optimal combustion chamber has greater depth in the middle and rear parts, a greater

perpendicular distance between the spoiler and the X–Z cross-section, and a greater angle between the bottom edge of the middle and rear parts of the combustion chamber and the Y–Z plane.

Table 6. Comparison of computational efficiency.

Type	Number of Combinations of Structural Parameters	Number of Simulation Calculations	Time/h
Optimization based on fuzzy neural network–genetic algorithm	240	184	366
Genetic algorithm	240	240	481



(**a**) Original combustion chamber (**b**) Optimal combustion chamber

Figure 23. Comparison of combustion chamber structures.

Table 7. Structural parameters.

Structural Name (of a Building)	Original Combustion Chamber	Optimal Combustion Chamber
Radius of the front arc of the combustion chamber/mm	1.51	32.96
Length of bottom edge of combustion chamber front/mm	6.45	2.11
Distance between the bottom of the front section of the combustion chamber at the X–Z section of the rotor and the center of gravity of the rotor/mm	38.84	42.40
Angle between the bottom edge of the front of the combustion chamber and the Y–Z plane/°	2.00	7.26
Radius of arc in the middle and rear of the combustion chamber/mm	13.04	13.35
Length of bottom edge at center rear of combustion chamber/mm	8.43	5.52
Distance between the bottom of the middle and rear section of the combustion chamber at the X–Z section of the rotor and the center of gravity of the rotor/mm	37.31	36.03
Angle between the bottom edge of the center of the combustion chamber and the Y–Z plane/°	3.40	6.44
Angle between the bottom edge of the rear of the combustion chamber and the Y–Z plane/°	0.21	12.33
Vertical distance between spoiler and X–Z section of rotor/mm	0.00	10.17
Distance between top center of spoiler and bottom edge of combustion chamber/mm	4.09	3.19
Spoiler width/mm	0.80	1.03
Radius of the excess arc between the front and the middle of the combustion chamber/mm	3.00	2.47
Radius of the excess arc between the middle and the rear of the combustion chamber/mm	0.10	3.57

4.4. Analysis of Rotor Engine Combustion Chamber Structure Optimization Results

Shown in Figure 24 are the variation curves of the exothermic rate of the original and optimal combustion chamber, and shown in Figure 25 are the variation curves of the in-cylinder temperature of the original and optimal combustion chamber. Shown in Figure 26 are the variation curves of the in-cylinder pressure of the original and optimal combustion chamber, and the maximum values of the in-cylinder pressure and in-cylinder temperature

of the original and optimal combustion chamber and their corresponding crank angles are shown in Table 8. From the figures, it can be seen that the exothermic rate starts to rise at different moments under different schemes, and the crank angle corresponding to the peak exothermic rate is different. The exothermic rate rise moment of the optimal combustion chamber is significantly earlier than that of the original combustion chamber. The exothermic rate of the optimal combustion chamber structure rises faster, and the crank angle corresponding to the peak exothermic rate is close to the upper stop, but the peak is smaller than that of the original combustion chamber. The temperature rise varies with the combustion rate and the optimal combustion chamber has a lower peak temperature, although the exothermic rate rises faster in the early stage.

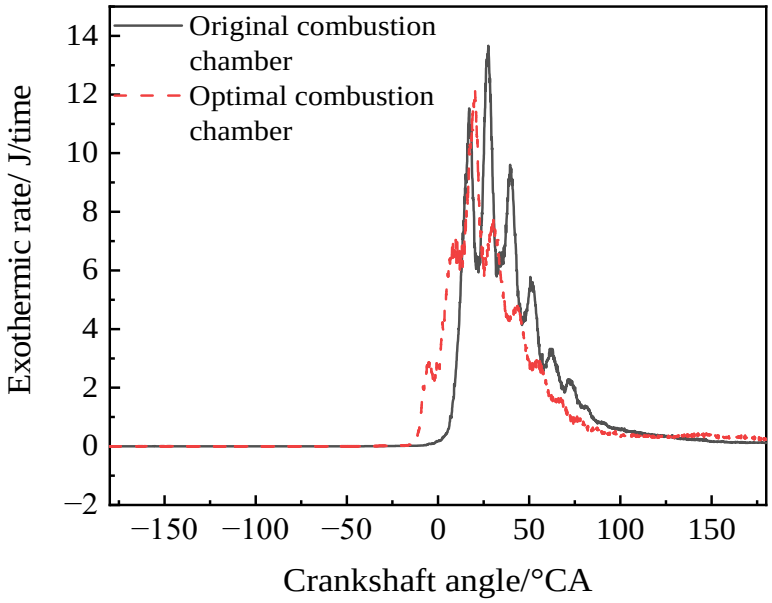


Figure 24. Curve of heat release rate.

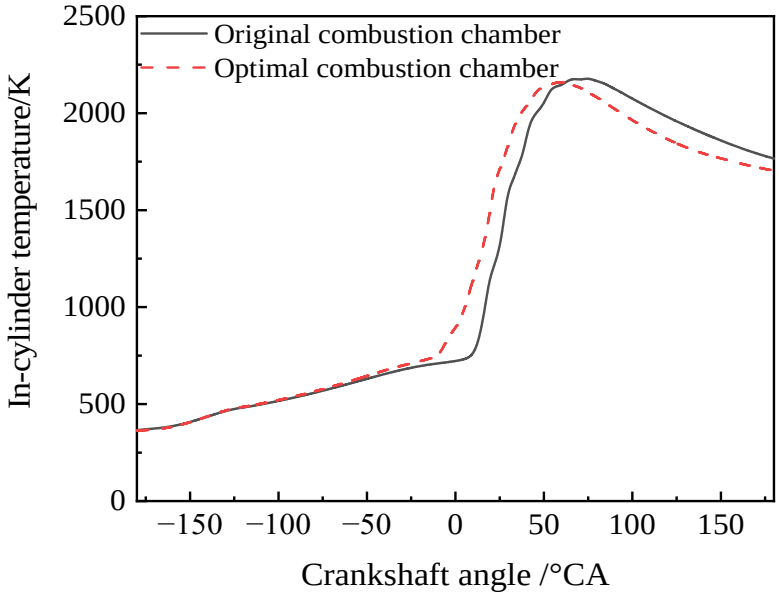


Figure 25. Temperature change curves.

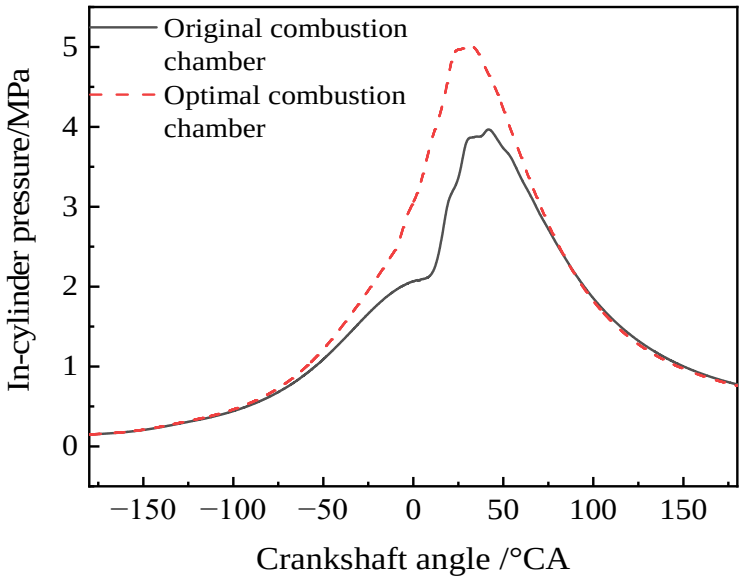


Figure 26. Cylinder pressure change curves.

Table 8. Crankshaft angle corresponding to peak temperature, and peak temperature for different scenarios.

Program	Peak Temperature/K	Crankshaft Angle of Rotation/° CA	Maximum Cylinder Pressure/MPa	Crankshaft Angle of Rotation/° CA
Original combustion chamber	2175.29	67.27	3.91	41.68
Optimal combustion chamber	2159.67	59.33	5.13	31.84

Figure 27 shows the P-V diagram of the original combustion chamber and the optimal combustion chamber, from which it can be concluded that the thermal efficiency of the original combustion chamber is 42.303% and that of the optimal combustion chamber is 43.897% mmt^3.

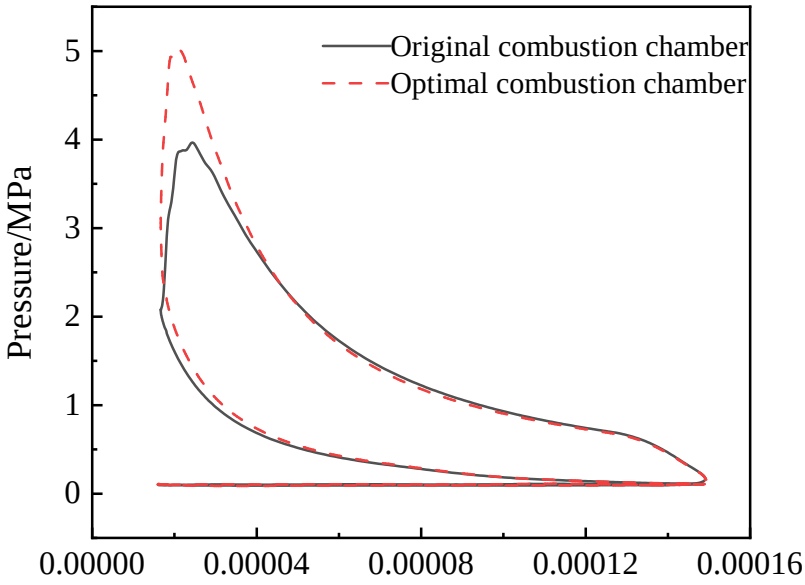


Figure 27. P-V Diagram of combustion chamber.

Figure 28 shows the in-cylinder flow velocity distributions of the original and optimal combustion chambers. From the figure, it can be seen that the flow velocity of the optimized combustion chamber is obviously larger than that of the original combustion chamber. At the moment of ignition, a large number of moving fluids with a flow velocity greater

than 100 m/s exist in the optimal combustion chamber, and with the combustion, the high-velocity fluids continue to increase. In the optimized combustion chamber, in the compression process, the fluid is more likely to hit the wall, increasing the fluid disturbance in the cylinder and thus increasing the fluid motion velocity and turbulent kinetic energy. Increasing the fluid motion velocity at the ignition moment can promote flame propagation during the combustion process.

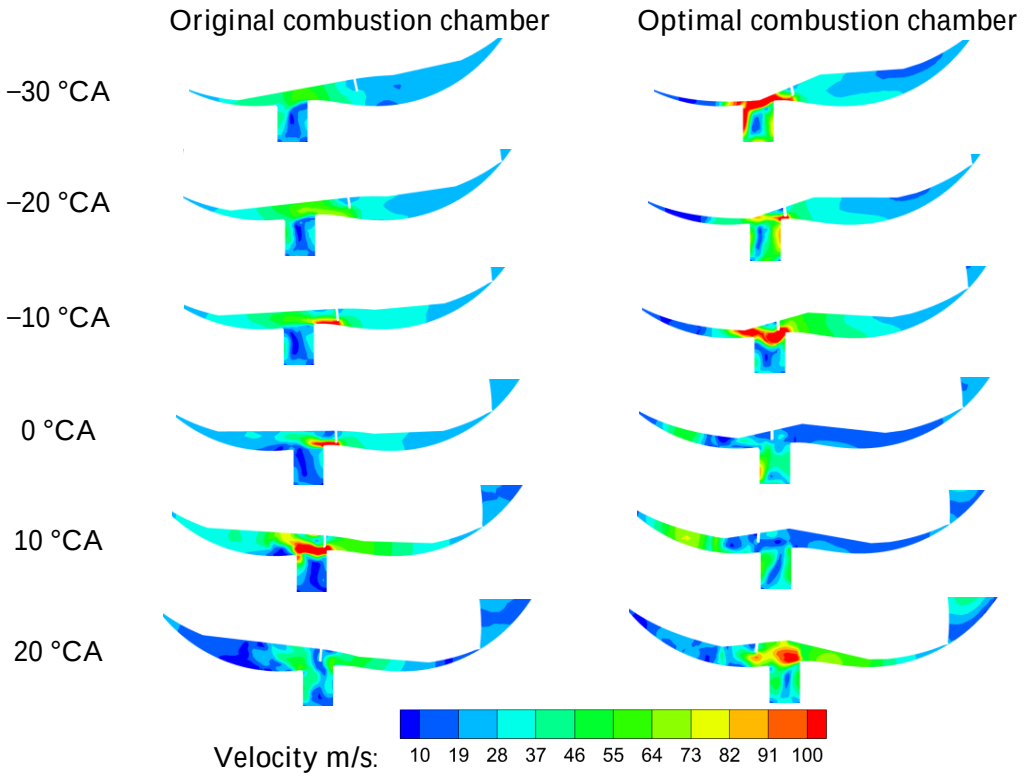


Figure 28. Flow velocity distribution in cylinder.

The flame fronts of the original combustion chamber and the optimal combustion chamber with a temperature of 1600 K are shown in Figure 29. The figure shows the distributions of the flame fronts at different moments for each ignition location scenario, where the flame front is characterized by an iso-surface with a temperature of 1600 K. From the figure, it can be seen that the flame propagation speed of the optimal combustion chamber is significantly larger than that of the original combustion chamber, and its 1600 K flame coverage area is significantly larger than that of the original combustion chamber from -10° CA. This is due to the fact that, during the combustion process, the flame is more likely to contact the wall and burn near the wall under the action of the high-velocity fluid. In summary, the optimization algorithm proposed in this paper can significantly improve the optimization efficiency and effect. However, the optimization algorithm was not verified in the experiment. The prejudged classification module constructed using a fuzzy neural network in this study can be flexibly added to various optimization algorithms, and has stronger universality than other methods in accelerating iteration.

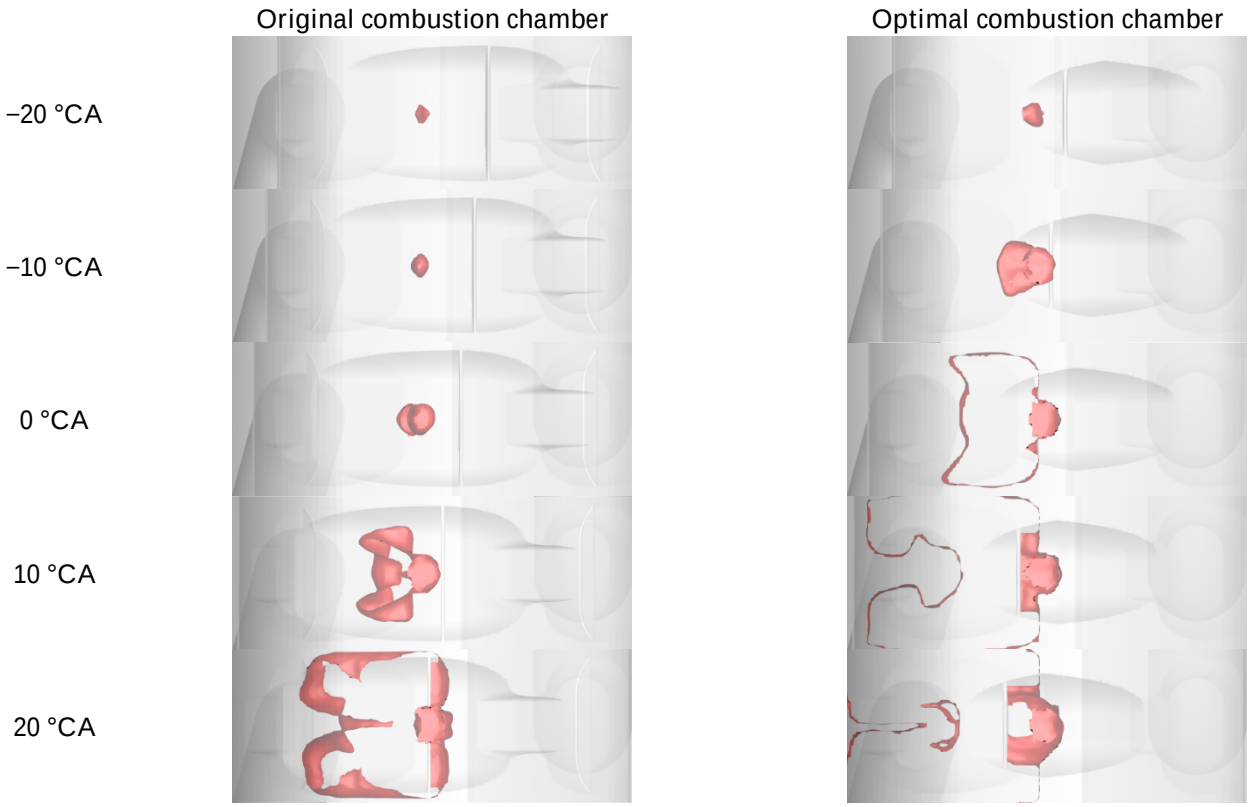


Figure 29. In-cylinder flame fronts.

5. Conclusions

In this paper, a fuzzy neural network-based method for the optimization of a combustion chamber structure is proposed with reference to the concept of fuzzy neural networks in machine learning. Using the optimization method proposed in this paper and combined with the traditional genetic algorithm, the problems of optimization iteration for the fitness function and the combustion chamber structure of a rotary engine are tested. Subsequently, the results of both are analyzed and compared. The conclusions are as follows:

1. Different from the traditional optimization method that directly simulates the structural combinations to obtain the fitness value, the optimization method proposed in this paper uses a fuzzy neural network to prejudge the combinations of combustion chamber structure parameters for classification before simulation, so as to improve the optimization accuracy and reduce the iteration time. Tests using the fitness function as the objective function found that, when using a fuzzy neural network for prejudged classification, the correct classification rate can reach 91.5% if an appropriate classification threshold is selected. On this basis, the results of iterative optimization for the fitness function show that the optimization method proposed in this paper can significantly reduce the optimization time. Compared to the traditional genetic algorithm's average number of convergence generations of 21.68, the average number of convergence generations for the fuzzy neural network proposed in this paper is 15.48, which is a reduction of 28.59%. The average calculation time of the fitness function for the traditional neural network when converging is 2168.5, while that based on the fuzzy neural network proposed in this paper is 1613.58, which is a reduction of 25.59%.
2. In order to further verify the effectiveness of the optimization method proposed in this paper, the combination of combustion chamber structure parameters in a rotary engine is optimized and iterated with the peak pressure taken as the optimization objective.

The results show that after three rounds of optimization iterations, the peak pressure of the optimal combustion chamber when using the optimization method proposed in this paper is 5.13 MPa, which is 31.20% higher than that of the prototype combustion chamber, at 3.91 MPa. The flow rate in the optimal combustion chamber is significantly higher than that in the original combustion chamber. The flame propagation speed of the optimal combustion chamber is significantly greater than that of the original combustion chamber, and its flame front is significantly larger than that of the original combustion chamber from -10° CA.

3. The peak pressure of the optimal combustion chamber obtained using the fuzzy neural network-based combustion chamber structure optimization method proposed in this paper is 5.13 MPa, which is 10.32% higher than the value of 4.65 MPa obtained by the traditional genetic algorithm. In terms of efficiency, the optimization algorithm proposed in this paper requires a total of 184 simulations after three iterations, which is 23.33% lower than the 240 simulations required for the traditional genetic algorithm. The total time consumed for three rounds of iterative optimization using the optimization algorithm proposed in this paper is 366 h, which is 23.91% lower than the 481 h required by the traditional genetic algorithm. This proves that, after adding the fuzzy neural network classification module, this algorithm can significantly improve the optimization efficiency while improving the optimization effect.

Author Contributions: Conceptualization, M.Y.; Methodology, M.Y.; Software, M.Y.; Writing—original draft, M.Y.; Validation, L.L.; Formal analysis, R.Z.; Writing—review and editing, R.Z., T.S. and H.W.; Supervision, R.Z.; Funding acquisition, R.Z. and N.W.; Investigation, N.W. All authors have read and agreed to the published version of the manuscript.

Funding: This article was supported by the National Natural Science Foundation of China (No. 52106167) and the Applied Basic Research Programs of Shanxi Province in China (202203021222045).

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

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

References

1. Zambalov, S.; Kasaev, D.; Yakovlev, I.; Ji, C.; Yang, J.; Maznoy, A. Effect of over-expansion in a cycloidal rotary engine. *Energy* **2024**, *302*, 131794. [CrossRef]
2. Meng, H.; Ji, C.; Yang, J.; Wang, H.; Wang, Z.; Zambalov, S.; Yakovlev, I. Performance analysis of the ammonia-enriched hydrogen-fueled Wankel rotary engine. *Int. J. Hydrogy Energy* **2024**, *49*, 462–472. [CrossRef]
3. Janusz, P. A Review of Shock Wave Compression Rotary Engine Projects, Investigations and Prospects. *Energies* **2022**, *15*, 9353.
4. Fan, B.; Zeng, Y.; Pan, J.; Fang, J.; Salami, H.A.; Wang, Y. Numerical study of injection strategy on the combustion process in a peripheral ported rotary engine fueled with natural gas/hydrogen blends under the action of apex seal leakage. *Energy* **2022**, *242*, 122532. [CrossRef]
5. Fan, B.; Song, A.; Liu, W.; Jiang, P.; Xu, L.; Pan, J.; Zhang, Y. Potential improvement in combustion performance of a natural gas rotary engine mixed with hydrogen by novel bluff-body. *Energy* **2024**, *295*, 131077. [CrossRef]
6. Kim, Y.J.; Park, T.J.; Yang, J.H.; Lee, C.E. Optimal Design for New Rotary Engine with Geometric Shape Functions on Combustion Chamber and Ports. *Energies* **2024**, *17*, 1754. [CrossRef]
7. Fan, B.; Pan, J.; Yang, W.; Liu, Y.; Bani, S.; Chen, W. Numerical investigation of the effect of injection strategy on mixture formation and combustion process in a port injection natural gas rotary engine. *Energy Convers. Manag.* **2017**, *133*, 511–523. [CrossRef]
8. Zou, R.; Liu, J.; Jiao, H.; Zhao, J.; Wang, N. Combined effect of intake angle and chamber structure on flow field and combustion process in a small-scaled rotary engine. *Appl. Therm. Eng.* **2022**, *203*, 117652. [CrossRef]
9. Amrouche, F.; Erickson, P.; Park, J.; Varnhagen, S. An experimental investigation of hydrogen-enriched gasoline in a Wankel rotary engine. *Int. J. Hydrogy Energy* **2014**, *39*, 8525–8534. [CrossRef]
10. Chen, X. Design and Optimization of 210 New Combustion Chamber for Marine Diesel Engine. Ph.D. Thesis, Dalian University of Technology, Dalian, China, 2017.

11. Peng, Z.; Guoxiu, L.; Yusong, Y. Multi-dimensional simulation of uniform optimization design of direct injection diesel engine combustion system. *J. Eng. Thermophys.* **2010**, *31*, 689–692.
12. Hu, D. Simulation study on Effect of High-Intensification Combustion Chamber on Combustion Process and Performance of Diesel Engine. Ph.D. Thesis, Beijing Jiaotong University, Beijing, China, 2012.
13. Li, H.; Lyu, J.; Li, Y.; Zhong, L. Combustion System Optimization Design of a Heavy Duty V8 Diesel Engine. *J. Automot. Engine* **2016**, *3*, 14–19.
14. Bai, G. Optimization Analysis of Oil and Gas Parameters in Combustion Chamber of Heavy-Duty Diesel Engine. Ph.D. Thesis, Huazhong University of Science and Technology, Wuhan, China, 2017.
15. Cheng, J.; Zong, C.; Ji, C.; Bao, J.; Zhu, T. Combustion chamber Structure optimization and Numerical Simulation of a 100 kW microgas turbine. *J. Combust. Sci. Technol.* **2017**, *23*, 68–74.
16. Yang, D.; Yu, Z.; Yuan, H.; Cui, Y. An improved genetic algorithm and its application in neural network adversarial attack. *PLoS ONE* **2022**, *17*, e0267970. [CrossRef] [PubMed]
17. Wu, M.; Cao, N.; Wang, H.; Xu, L.; Li, G. A hybrid optimisation algorithm based on GA algorithm and ACO algorithm improvements for routing selection in heterogeneous sensor networks. *Int. J. Embed. Syst.* **2020**, *12*, 177–185. [CrossRef]
18. Gao, J.; Lv, Y.; Liu, M.; Cai, S.; Ma, F. Improving simulated annealing for clique partitioning problems. *J. Artif. Intell. Res.* **2022**, *74*, 1485–1513. [CrossRef]
19. Chen, S.; Chen, Y.; Feng, M. Indoor Infrared Sensor Layout Optimization for Elderly Monitoring Based on Fused Genetic Gray Wolf Optimization (FGGWO) Algorithm. *Sensors* **2024**, *24*, 5393. [CrossRef]
20. Cao, B.; Zhao, J.; Liu, X.; Arabas, J.; Tanveer, M.; Singh, A.K.; Lv, Z. Multiobjective evolution of the explainable fuzzy rough neural network with gene expression programming. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 4190–4200. [CrossRef]
21. Zhou, Z.; Li, F.; Zhu, H.; Xie, H.; Abawajy, J.H.; Chowdhury, M.U. An improved genetic algorithm using greedy strategy toward task scheduling optimization in cloud environments. *Neural Comput. Appl.* **2020**, *32*, 1531–1541. [CrossRef]
22. Zhang, G.; Hu, Y.; Sun, J.; Zhang, W. An improved genetic algorithm for the flexible job shop scheduling problem with multiple time constraints. *Swarm Evol. Comput.* **2020**, *54*, 100664. [CrossRef]
23. Li, D.; Cao, Q.; Zuo, M.; Xu, F. Optimization of Green Fresh Food Logistics with Heterogeneous Fleet Vehicle Route Problem by Improved Genetic Algorithm. *Sustainability* **2020**, *12*, 1946. [CrossRef]
24. Wang, H.; Ji, C.; Shi, C.; Yang, J.; Wang, S.; Ge, Y.; Chang, K.; Meng, H.; Wang, X. Multi-objective optimization of a hydrogen-fueled Wankel rotary engine based on machine learning and genetic algorithm. *Energy* **2023**, *263*, 125961. [CrossRef]
25. Lingras, P. Fuzzy-rough and rough-fuzzy serial combinations in neurocomputing. *Neurocomputing* **2001**, *36*, 29–44. [CrossRef]
26. Tang, J.; Liu, G.; Pan, Q. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1627–1643. [CrossRef]
27. Li, W.; Yu, J.; Ren, N.; Huang, L.; Dang, Y.; Wu, Y.; Li, G. Exploration of the prediction and generation patterns of heterocyclic aromatic amines in roast beef based on Genetic Algorithm combined with Support Vector Regression. *Food Chem.* **2024**, *463 Pt 1*, 141059. [CrossRef]
28. Liu, X.; Yang, G.; Wang, L.; Fu, J.; Wang, Q. A novel immune detector training method for network anomaly detection. *Appl. Intell.* **2024**, *54*, 2009–2030. [CrossRef]
29. Huang, J.; Gu, X. Distributed assembly permutation flow-shop scheduling problem with sequence-dependent set-up times using a novel biogeography-based optimization algorithm. *Eng. Optim.* **2022**, *54*, 593–613. [CrossRef]
30. Jiang, Z.; Chen, G.; Qiao, R.; Feng, P.; Chen, Y.; Su, J.; Zhao, Z.; Jin, M.; Chen, X.; Li, Z.; et al. Point convolutional neural network algorithm for Ising model ground state research based on spring vibration. *Sci. Rep.* **2024**, *14*, 2643. [CrossRef]
31. Zhang, P.; Cui, Z.; Wang, Y.; Ding, S. Application of BPNN optimized by chaotic adaptive gravity search and particle swarm optimization algorithms for fault diagnosis of electrical machine drive system. *Electr. Eng.* **2022**, *104*, 819–831. [CrossRef]
32. Zhou, G.; Zhou, Y.; Deng, W.; Yin, S.; Zhang, Y. Advances in teaching-learning-based optimization algorithm: A comprehensive survey. *Neurocomputing* **2023**, *561*, 126898. [CrossRef]
33. García, M.; López, N.; Rodríguez, I. A full process algebraic representation of Ant Colony Optimization. *Inf. Sci.* **2024**, *658*, 120025. [CrossRef]
34. van Zyl, J.P.; Engelbrecht, A.P. Set-based particle swarm optimisation: A review. *Mathematics* **2023**, *11*, 2980. [CrossRef]
35. Trojovská, E.; Dehghani, M.; Trojovský, P. Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm. *IEEE Access* **2022**, *10*, 49445–49473. [CrossRef]
36. Chan, H.; Nai-He, Y. A pretreatment method of wastewater based on artificial intelligence and fuzzy neural network system. *J. Intell. Fuzzy Syst.* **2020**, *39*, 1711–1720. [CrossRef]
37. Lawrence, K.R.; Huang, Z.; Nguyen, X.P.; Balasubramanian, D.; Gangula, V.R.; Doddipalli, R.R.; Le, V.V.; Bharathy, S.; Hoang, A.T. Exploration over combined impacts of modified piston bowl geometry and tert-butyl hydroquinone additive-included biodiesel/diesel blend on diesel engine behaviors. *Fuel* **2022**, *322*, 124206. [CrossRef]

38. Shi, C.; Ji, C.; Wang, S.; Yang, J.; Ma, Z.; Meng, H. Potential improvement in combustion behavior of a downsized rotary engine by intake oxygen enrichment. *Energy Convers. Manag.* **2020**, *205*, 112433. [CrossRef]
39. Jeyaseelan, T.; Ekambaram, P.; Subramanian, J.; Shamim, T. A comprehensive review on the current trends, challenges and future prospects for sustainable mobility. *Renew. Sustain. Energy Rev.* **2022**, *157*, 112073. [CrossRef]
40. Zhang, H.; Wang, L.; Shi, W. Seismic control of adaptive variable stiffness intelligent structures using fuzzy control strategy combined with LSTM. *J. Build. Eng.* **2023**, *78*, 107549. [CrossRef]
41. Mowbray, M.; Savage, T.; Wu, C.; Song, Z.; Cho, B.A.; Del Rio-Chanona, E.A.; Zhang, D. Machine learning for biochemical engineering: A review. *Biochem. Eng. J.* **2021**, *172*, 108054. [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.