An Optimized Neural Network Acoustic Model for Porous Hemp Plastic Composite Sound-Absorbing Board

Haizhen Wang 1, Hong Zhao 2,* , Zuozheng Lian 1, Bin Tan 3, Yongjie Zheng 2 and Erdun E 4

1 College of Computer and Control Engineering, Qiqihar University, Qiqihar 161006, China; 01559@qqhru.edu.cn (H.W.); 00816@qqhru.edu.cn (Z.L.)
2 College of Light Industry and Textile, Qiqihar University, Qiqihar 161006, China; 01648@qqhru.edu.cn
3 School of Chemical Engineering and Materials Science, Michigan State University, East Lansing, MI 48913, USA; tanbin1@msu.edu
4 Elastic Block Store Performance Team, Amazon Web Services, Boston, MA 02210, USA; erdunwork@gmail.com
* Correspondence: 02130@qqhru.edu.cn

Abstract: Current acoustic modeling methods face problems such as complex processes or inaccurate sound absorption coefficients, etc. Therefore, this paper studies the topic. Firstly, the material samples were prepared, and standing wave tube method experiments were conducted. Material acoustic data were obtained, while a model using improved genetic algorithm and neural network was subsequently proposed. Secondly, the acoustic data obtained from the experiment were analyzed; a neural network structure was designed; and the training, verification, and test data were all divided. In order to facilitate data processing, a symmetrical method was used to inversely normalize all the data. Thirdly, by the design of real coding scheme, fitness function, crossover, and mutation operators, an improved genetic algorithm was proposed to obtain the optimal solution, as the initial weight and threshold, which were then input into the neural network along with the training and verification data. Finally, the test data were input into the trained neural network in order to test the model. The test results and statistical analysis showed that compared with other algorithms, the proposed model has the lower root mean squared error (RMSE) value, the maximum coefficient of determination ($R^2$) value, and shorter convergence time.

Keywords: porous hemp plastic composite; sound-absorbing board; acoustic modeling; neural network; genetic algorithm; Levenberg–Marquardt algorithm

1. Introduction

Noise pollution has always been an issue [1–4] in fields such as architecture, transportation, and construction machinery, and many people have now recognized it as a serious problem. Directly using a sound-absorbing medium is an effective method of controlling an increasing variety and number of noise sources. In this method, noise waves pass through a sound absorber containing a series of energy conversion technologies, and any sound energy is subsequently transferred into other forms. Usually, sound absorbers used in engineering are categorized as either resonant or porous sound-absorption structures. Porous sound-absorption structures absorb sound energy through the internal structure of the material itself. These include porous metal materials, glass wool, wood fiber board, and foamed plastics, all of which are effective sound-absorption porous materials [5–7]. These are the main source of sound-absorbing materials currently used in noise control.

As a porous fiber material [8,9], hemp has good properties, such as dissipating sound wave, moisture absorption and permeability, fine and soft, corrosion resistance, etc., but its strength is low and cannot be applied to harsh environments such as high temperature and strong airflow. In order to improve the strength of hemp, hemp-based composites have been widely studied [10–14], but most of them study the preparation methods of materials,
and there is little research on the acoustics modeling of hemp composites, especially the research on modeling and analysis using neural network. However, the research on acoustic modeling is of great significance to the application of sound absorption and noise reduction. Therefore, the acoustic modeling method of porous hemp plastic composites is studied in this paper. The main contributions of this paper are as follows:

(1) Study the principle of measuring sound-absorption coefficients by a standing wave tube method, prepare porous hemp plastic composite samples, and experimentally measure the acoustic data of the materials. (2) According to the characteristics of the acoustic data and experiments, an optimized model is proposed. Firstly, a three-layer neural network structure is designed, and the number of neurons in the input, hidden, and output layer is determined to identify the number of neural network weights and thresholds. Secondly, the improved genetic algorithm with population diversity, including the design of real coding scheme, fitness function, selection, crossover, and mutation operators, and an optimal solution is proposed. Finally, the optimal solution is used as the initial weight and threshold of the neural network, and the optimal model is generated by training the network. (3) Experimental test results and statistical analysis show that compared with a one-dimensional CA model, unoptimized neural network model, support vector regression, random forest regression and XGBoost regression, the proposed model has the lower root mean squared error (RMSE) value, the maximum coefficient of determination ($R^2$) value and shorter convergence time, reflecting better performance.

The rest of the paper is organized as follows: Section 2 gives a literature review. Section 3 introduces the design process and ideas of the proposed model. Section 4 gives the experimental results and compares analyses of the existing related modeling algorithms. The conclusion and future work are outlined in Section 5.

2. Literature Review

Recently, some studies have been undertaken on porous material acoustic modeling. Using Delany–Bazley–Miki (DBM) and Johnson–Champoux–Allard (JCA) models, Ramamoorthy et al. [9] analyzed the sound absorption performance of a double-layer, porous, shielded, sound-absorption system supported by a rigid cut-off wall. Subsequently, they produced a program for multilayer systems based on a steady-state genetic algorithm and calculated the relevant acoustic indexes with trimmed mean of M value (TMM) and fragments trimmed mean of M value (FTMM) solvers. Based on Biot theory and different interfacial layered media boundary conditions, Bai et al. [10] established a theoretical multi-layer composite media acoustic model supporting a rigid wall structure, the calculated normal incidence absorption coefficient of which was in good agreement with the measured results. Mareze et al. [11] proposed a cellular, porous, medium model based on microscopic images, consisting of a two-dimensional network of pipes comprised of simple one-dimensional acoustic cylinders or pores. Sun et al. [12] reconstructed the spatial topological structure of porous materials, and consequently proposed an equivalent mass-spring-damping vibration system model, which was verified through an impedance tube test. Gao et al. [13] improved the average sound absorption coefficient of a porous metamaterial structure, based on a mixture of composites within the frequency range of 0 to 10 kHz. This created a teaching-based optimization algorithm using the layer spacing and side plate length as the optimization variables. Pereira et al. [15] compared the 2D finite element method (FEM) and 3D boundary element method (BEM) [16] using a reverberation chamber for the analytical equations and experimental data and investigated how the size of the porous material plate influenced the sound absorption coefficient in a diffusion field. Gao et al. [17] proposed a composite porous metamaterial (CPM) and characterized the CPM sound-absorption coefficient using the acoustic parameters of the JCA model. Hassani et al. [18] used recycled denim bonded with phenolic resin as a composite sound-absorption material and prepared corresponding specimens to determine the sound-absorption coefficient. Furthermore, to verify the relevant test data, they modeled the relationship between the sound-absorption coefficient and frequency through logarith-
mic regression and the JCA model. Semeniuk et al. [19] proposed the design and analysis of a foam pool model and studied the methods used for typical melamine foam pool acoustic performance prediction. A highly satisfactory effect was obtained. The cellular automata (CA) method is suitable for describing complex dynamic systems and has been widely used in engineering acoustics modeling [20–24].

A review of the above relevant studies demonstrates that describing the soundwave propagation in materials is highly important in material acoustic modeling. Currently, several relevant methods, including the Biot equation, DBM model, JCA model, one-dimensional CA, exist for this purpose. The Biot equation considers the coupled propagation of elastic waves in elastic porous material and pressure waves in saturated pore fluid, while incorporating the damping effect of pore fluid. The Biot equation is elaborate in its description but complicated to solve, while the DBM and JCA models treat porous material and saturated pore fluid as homogeneous equivalent fluids to model pressure waves and describe soundwave loss. When the DBM or JCA model is used, many acoustic parameters, such as acoustic and flow resistivity, porosity, tortuosity factor, thermal characteristics length, etc., need to be measured using complex methods or special equipment. For example, the porosity needs to be measured by a scanning electron microscope and the effective density of porous materials needs to be measured via an impedance tube to obtain the flow resistivity. In addition, the optimization method is used for inverse calculation. The DBM model is applied to porous materials with low porosity, and the resulting sound-absorption coefficient error exists. One-dimensional CA usually deduces the local evolution rules between cells through the difference calculation of a one-dimensional acoustic wave equation in a standing wave tube model to obtain the sound pressure distribution of plane acoustic waves. The limitations on the cell size design minimum sound pressure are inaccurate, which affects the accuracy of the sound absorption coefficient value. One-dimensional CA modeling is, therefore, not sufficiently accurate. Neural network modeling can approximate any complex nonlinear relationship, but it is rarely used in porous material acoustic modeling, which inspires us to study the acoustic modeling of materials based on the widely used BP neural network. However, the BP algorithm suffers some major drawbacks, such as the likelihood to be trapped in a local minimum, slow convergence, and dependency on the initial starting point of their search. Therefore, in recent years, Prof. Mirjalili [25,26] and other scholars [27–33] have proposed many neural network optimization methods, using traditional metaheuristics. Recently, novel metaheuristics, such as red deer and social engineering optimizer [34–39], have been proposed. Compared with traditional metaheuristics, they show advantages in some field, but they are rarely combined with neural network. Additionally, GA has the characteristics of group search, and it searches based on probability rules with the value of objective function, which is more flexible and scalable. Inspired by them, and from the perspective of higher accuracy and lower computational complexity, we propose the improved GA optimized neural network acoustic model. We attempt to further analyze the performance of the model and compare it with other optimization algorithms in depth in the following sections.

3. Materials and Methods

3.1. Obtaining Data

In this paper, a standing wave method was used to evaluate a porous hemp plastic composite sound-absorbing board and obtain relevant material acoustic data. The procedure was as follows: a porous hemp plastic composite sound-absorbing board was prepared and cut into appropriately sized samples (low frequency diameter: 100 mm, medium frequency diameter: 50 mm, as shown in Figure 1). The experimental setup used in this paper was a JTZB sound-absorption coefficient test system. Figures 2 and 3 show the experimental setup and its block diagram.
The experimental setup measures the acoustic data of samples based on the principle of standing wave tube; the key steps of the experiment are as follows:

Step 1. Connect the hardware devices according to the block diagram. A sample was mounted at one end of standing wave tube, with the other end connected to a loudspeaker.

Step 2. Turn on the power for all hardware devices.

Step 3. Install JTZB special software on the computer and set the loudspeaker output frequency as the center frequency.

Step 4. Move the test car to the loudspeaker side, and then slowly slide to the right to find a maximum sound pressure value and record it. Then slowly move the test car, find the first minimum sound pressure value adjacent to the maximum sound pressure, and record it.

Step 5. Adjust the loudspeaker output frequency and repeat steps 3–5.

Figure 1. Two size samples. (a) A sample of 100 mm diameter; (b) a sample of 50 mm diameter.

Figure 2. The experimental setup.

Figure 3. The block diagram of the experimental setup.
Step 6. The absorption coefficient of the sample is calculated according to Equation (1).

$$\alpha = 1 - \frac{|P_{\text{max}} - P_{\text{min}}|}{P_{\text{max}} + P_{\text{min}}}^2$$  \hspace{1cm} (1)

where $\alpha$ is the sound absorption coefficient, $P_{\text{max}}$ is the maximum sound pressure value and $P_{\text{min}}$ is the minimum sound pressure value.

Finally, acoustic data of the sample were obtained at a central frequency between 200 Hz and 4000 Hz. A total of 50 samples were measured, and Table 1 presents one of them; therefore, there are 700 pieces of data in total. The experimental principle is as follows: due to the small diameter of the tube relative to the audio sound wave wavelength, the acoustic wave surface can be regarded as a plane incident wave propagating linearly in the tube, which is reflected to the sample after incidence. As the reflected and incident waves travel in opposite directions and phases, the sound pressure superposes and interferes to form standing waves, with the maximum and minimum sound pressure values at two positions within the tube, the distance between which is $1/4$ wavelength.

**Table 1.** The acoustic data of a sample.

<table>
<thead>
<tr>
<th>Center Frequency/Hz</th>
<th>Maximum Sound Pressure/dB</th>
<th>Minimum Sound Pressure/dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>96.9</td>
<td>71.8</td>
</tr>
<tr>
<td>250</td>
<td>100.7</td>
<td>79.8</td>
</tr>
<tr>
<td>315</td>
<td>109.6</td>
<td>88.3</td>
</tr>
<tr>
<td>400</td>
<td>106</td>
<td>85.4</td>
</tr>
<tr>
<td>500</td>
<td>105</td>
<td>85.4</td>
</tr>
<tr>
<td>630</td>
<td>90.8</td>
<td>73.5</td>
</tr>
<tr>
<td>800</td>
<td>90.8</td>
<td>74.2</td>
</tr>
<tr>
<td>1000</td>
<td>87</td>
<td>72.3</td>
</tr>
<tr>
<td>1250</td>
<td>73.8</td>
<td>61.6</td>
</tr>
<tr>
<td>1600</td>
<td>74.5</td>
<td>62.1</td>
</tr>
<tr>
<td>2000</td>
<td>63.5</td>
<td>52.6</td>
</tr>
<tr>
<td>2500</td>
<td>54</td>
<td>39.8</td>
</tr>
<tr>
<td>3150</td>
<td>49.6</td>
<td>37.2</td>
</tr>
<tr>
<td>4000</td>
<td>47.4</td>
<td>37.4</td>
</tr>
</tbody>
</table>

3.2. Proposed Model

The principle for the proposed model, shown in Figure 4, consists of designing the neural network structure, data processing, training the neural network, and result analysis. In the model design, in order to process the data, a symmetrical method is used to inverse normalize all the data. The following describes the design process of the model in detail.

![Figure 4. The principle of model.](image-url)
3.2.1. Design of the Neural Network Structure

Back propagation (BP) neural networks possess high nonlinear mapping capability. A BP neural network of three or more layers can approximate a nonlinear function with arbitrary precision, if there are sufficient hidden layer neurons. Consequently, a BP neural network was adopted for this study. The designed neural network structure is displayed in Figure 5. Relevant analysis was needed for this study. Therefore, for the given center frequency, maximum and minimum sound pressures, and the material sound absorption coefficient, the network structure design consisted of three layers: a three-neuron input layer, one-neuron output layer, and eight-neuron hidden layer. $I_1$, $I_2$, $I_3$ indicate the input; $a_{1,1}^1$, $a_{1,2}^1$, …, $a_{8,3}^1$, the hidden layer output; $a_{1,1}^2$, the network final output; $f^1$, the tansig function, as shown in Equation (2); $f^2$, the linear function, as shown in Equation (3); $iw^1$, the weight from the input layer to the hidden layer, as shown in Equation (4); and $lw^2$, the weight from the hidden layer to the output layer, as shown in Equation (5).

![Neural network structure](image)

Figure 5. Neural network structure.

Consequently, there were 32 weights and 9 threshold values (total 41 parameters) to be learned by the neural network. A genetic algorithm was used to optimize the initial parameter values.

\[ f^1(x) = \frac{1}{\arctan(x)} + 1 \]  
\[ f^2(x) = x \]

\[ iw^1 = \begin{bmatrix} iw_{1,1}^1 & \cdots & iw_{1,3}^1 \\ \vdots & \ddots & \vdots \\ iw_{8,1}^1 & \cdots & iw_{8,3}^1 \end{bmatrix} \]  
\[ lw^2 = \begin{bmatrix} lw_{1,1}^2 \\ lw_{2,1}^2 \\ \vdots \\ lw_{8,1}^2 \end{bmatrix} \]

3.2.2. Data Processing

Data were divided into training, verification, and test data, and were 70%, 15%, and 15% of the total sample data, respectively. The data needed to be normalized due to the difference in meaning and range. The processing method is illustrated with a row of data in Table 1 as an example. Data to be processed are represented by $X$, as shown in Equation (6), and data already processed by $Y$. The data processing was performed according to Equation (7).

\[ X = [200, 96.9, 71.8, 0.2] \]  
\[ Y = \text{process}(X) \]
\[ Y(i) = (y_{\text{max}} - y_{\text{min}}) \times \frac{X(i) - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} + y_{\text{min}} \] (7)

where \( y_{\text{max}}\), \( y_{\text{min}}\) are 1, −1, and \( x_{\text{max}}\), \( x_{\text{min}}\) are the maximum and minimum \( X \) values. Thus, the normalized data were as displayed in Equation (8).

\[ Y = [-1, -0.0320, -0.2833, -1] \] (8)

### 3.2.3. Improved Genetic Algorithm

Essentially, the BP neural network was utilized for optimization. The realization of an optimal solution is closely related to the initial value selection, that is, if the initial value is fairly close to the local optimal point, the resulting optimal solution will be incorrect. Subsequently, an improved genetic algorithm was designed, and the optimal parameters learned by it were input into the BP neural network as the initial value in order to obtain the optimal solution of the BP neural network. The flow chart of the improved genetic algorithm is shown in Figure 6.

![Figure 6. The pseudo-code of selection operator.](image)

As shown in Figure 6, a real coded strategy, the fitness function, crossover, and mutation operators were designed in the improved genetic algorithm. In order to improve individual diversity, crossover and mutation individuals and operation positions are randomly generated, and an individual coding effectiveness verification method is proposed to verify the individual coding after crossover and mutation operations. The main process of improved genetic algorithm to learn the optimal parameters is as follows: Firstly, the population size, evolutionary generation, crossover and mutation probability are defined, and all individuals of the population are real coded, which are used as the initial solution search space. Secondly, the fitness of all individuals is calculated and the best individual is selected. The roulette method is repeatedly used to select individuals until the number of individuals reaches the population size. Two individuals are selected for crossover operation until the individuals reach the population size. Two individuals are selected for mutation operation until the individuals reach the population size. Now, the individuals...
are used as the new solution search space. The fitness function value is calculated again, the best individual is updated, and this is repeated until the evolutionary generation is reached. The best individual is the optimal parameter learned.

(1) Initialization and Encoding

The search space is from $-3$ to $3$, namely, each chromosome was encoded to a real number in the range of $[-3, 3]$, and each individual stored in a $1 \times 41$ row vector. The other parameters were as shown in Table 2.

Table 2. Genetic algorithm parameter initialization.

<table>
<thead>
<tr>
<th>Evolitional Epochs</th>
<th>Population Size</th>
<th>Crossover Probability</th>
<th>Mutation Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>30</td>
<td>0.3</td>
<td>0.1</td>
</tr>
</tbody>
</table>

(2) Fitness function design

The genetic algorithm searches for the optimal solution according to the fitness function value of each individual, which then determines whether it is eliminated or inherited in the next generation. Consequently, the fitness function was designed with the aim that the individual’s actual output is closer to the expected output. With $i$ set as $i = 1, 2, 3, \ldots, 30$, the fitness function of individual $i$ is represented by $fit(i)$. With the model actual output matrix set as $TO$, and the expected output matrix set as $EQ$, the fitness function is as shown in Equation (9). More precisely, the training data are inputted into the neural network to calculate the difference between the actual and expected outputs for each specimen datum. The sum of the absolute values for these differences is then taken as the fitness function value.

$$fit(i) = \text{sum}(\text{abs}(TO - EQ))$$  \hspace{1cm} (9)

(3) Selection operator

The experiment shows that the roulette wheel method has a good optimization effect. As a result, this method was used for individual selection; the pseudo-code is shown in Figure 7.

```
Initialization and encoding;
Establish neural network training;
All individuals are used as weights, and the threshold is input into the trained network to obtain TO;
Calculate individual i fitness $fit(i) = \text{sum}(\text{abs}(TO - EO))$;
Calculate the sum of all fitness $\text{sumfit} = fit(1) + fit(2) + \cdots + fit(\text{Population size})$;
Calculate the selection probability $\text{sumf}(i) = fit(i) / \text{sumfit}$;
i=1;
j=1;
whilen i<=Population size
    while (! pi=rand);
        while j<=Population size
            pi=pi+sumf(j);
            if pi=0
                index=[index j];
                break;
        endif
    j=j+1;
endwhile
i=i+1;
endwhile
individuals index are selected;
```

Figure 7. The pseudo-code of selection operator.
(4) Crossover operator

When two individuals are randomly selected, the crossover probability chooses whether to perform the crossover, and a random number between 0 and 1 is generated. If the number is greater than the crossover probability, no crossover is performed for either individual. If it is lower, then a crossover operation is conducted. Specifically, a random crossover position was selected for the operation, and the individuals were checked for the effective range after the crossover. If ineffective, the crossover operation was then repeated until an effective individual was produced. The pseudo-code is shown in Figure 8.

```
j=1;
while j<=Population size
  while (!(pj=rand(1,2));
    endwhile
  index=pj*Population size;
  while (!(pi=rand));
    endwhile
  pi=rand;
  po=round(pi*encoding length);
  if pi<=Crossover probability
    while !(pi=rand);
      endwhile
    pi=rand;
  endif
  endif
  go to 1;
endif
j=j+1;
endwhile
```

Figure 8. The pseudo-code of crossover operator.

(5) Mutation operator

The pseudo-code is shown in Figure 9.

```
j=1;
while j<=Population size
  while !(pj=rand(1,2));
    endwhile
  index=pj*Population size;
  while !(pi=rand);
    endwhile
  pi=rand;
  po=round(pi*encoding length);
  f1=(pi*(1-current evolutional epochs)/total evolutional epochs)^2
  if f1>0.5
    the new coding of po position of individual index(1) is po position coding- (po position coding-3)*f1;
    endif
  the new coding of po position of individual index(2) is po position coding+ (3-po position coding)*f1;
  if the coding of index(i) is valid
    j=j+1;
  endif
  go to 1;
endif
j=j+1;
endwhile
```

Figure 9. The pseudo-code of mutation operator.
The mutation method is similar to the crossover method, and whether the mutation is performed depends on the mutation probability, that is, a random number between 0 and 1 is generated. If the number is higher than the mutation probability, no individual mutation is performed and if it is lower, then a mutation operation is performed. Specifically, an individual is randomly selected to generate a random mutation position for the operation, and the individual is checked for the effective range following the mutation. If ineffective, the crossover operation is repeated until an effective individual is produced.

3.2.4. Training of the Neural Network Model

The optimized weights and thresholds obtained after running the genetic algorithm, as shown in Equations (10)–(13), were then input into the neural network for model training. The Levenberg–Marquardt (LM) algorithm is suitable for nonlinear optimization since it has both the local convergence of the Gauss–Newton method and the global property of the gradient descent method. Consequently, the algorithm was used to train the network, and the specific procedure was as follows:

\[
\begin{bmatrix}
0.4712 & 0.7444 & 1.7771 \\
-1.5763 & 1.0748 & -2.4077 \\
-0.2469 & -0.6269 & -2.4077 \\
2.7785 & -0.7954 & -0.9879 \\
0.2808 & 2.9279 & 1.0784 \\
0.0666 & 2.4797 & -2.594
\end{bmatrix}
\]

(10)

\[
w_2 = [-1.8131 \quad -2.8168 \quad 1.4644 \quad 1.3461e-04 \quad -0.1205 \quad 2.4283 \quad 0.6592 \quad 0.7060]
\]

(11)

\[
1 = \begin{bmatrix}
0.9225 \\
-0.0350 \\
1.6743 \\
1.2902 \\
2.4223 \\
2.3653 \\
-0.9950 \\
1.1925
\end{bmatrix}
\]

(12)

\[
b = 2.1567
\]

(13)

(1) Set the allowable training error \( \varepsilon = 10^{-5} \), damping constant \( \mu_0 = 0.001 \), number of iterations \( k = 1 \), initial value \( \beta = (0, 1) \), and let \( \mu = \mu_0 \). Then input the weight and threshold obtained by the genetic algorithm into the network, together with the training data, as the initialized weight and threshold vector.

(2) Calculate the network output and error. The error function is shown in Equation (14), where the mean squared error (MSE) is adopted. \( Y_i \) represents the ideal network output in the formula, \( Y'_i \), the actual network output, \( p \), the number of specimens, \( w^k \), the vector composed of the weight and threshold of iteration \( k \), and \( e_i(w) \), the error generated by specimen \( i \).

\[
E(w^k) = \frac{1}{2} \sum_{i=1}^{sp} \| Y_i - Y'_i \|^2 = \frac{1}{2} \sum_{i=1}^{sp} e_i^2(w)
\]

(14)
(3) Calculate the Jacobian matrix \( J(w) \) according to Equation (15).

\[
J(w) = \begin{vmatrix}
\frac{\partial e_1(w)}{\partial w_1} & \frac{\partial e_2(w)}{\partial w_1} & \cdots & \frac{\partial e_N(w)}{\partial w_1} \\
\frac{\partial e_1(w)}{\partial w_2} & \frac{\partial e_2(w)}{\partial w_2} & \cdots & \frac{\partial e_N(w)}{\partial w_2} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial e_1(w)}{\partial w_n} & \frac{\partial e_2(w)}{\partial w_n} & \cdots & \frac{\partial e_N(w)}{\partial w_n}
\end{vmatrix}
\]  

(15)

(4) Calculate the increment \( \Delta w \) of weight and threshold using the method shown in Equation (16), where \( I \) represents the unit matrix, \( e(w) \) is the error matrix of all the specimens, and \( J(w) \) is the Jacobian matrix.

\[
\Delta w = \left[ J^T(w)J(w) + \mu I \right]^{-1}J^T(w)e(w)
\]  

(16)

(5) If \( E(w^k) \) is smaller than \( \varepsilon \), the set error value, end the training. If not, continue to step (6).

(6) Update the weight and threshold vector according to Equation (17), calculate the error \( E(w^{k+1}) \) according to Equation (14), and if \( E(w^{k+1}) < E(w^k) \), update \( k \) and \( \mu \) according to Equations (18) and (19), respectively, and continue to step (2). Otherwise, update \( k \) and \( \mu \) according to Equations (18) and (20), and then progress to step (4).

\[
w^{k+1} = w^k + \Delta w
\]  

(17)

\[
k = k + 1
\]  

(18)

\[
\mu = \mu \beta
\]  

(19)

\[
\mu = \mu / \beta
\]  

(20)

3.2.5. Model Testing

The function expression corresponding to the trained model is shown in Equation (21), where \( w_1, w_2, b_1, \) and \( b_2 \) are the final learned weights and thresholds, as shown in Equations (22)–(25). The test data are then input into the network, and the expected and actual outputs are compared, as discussed in Section 3.

\[
y = w_2 \ast \frac{1}{\arctan(w_1 \ast x + b_1)} + b_2
\]  

(21)

\[
w_1 = \begin{bmatrix}
0.0870 & -0.1795 & 1.1493 \\
0.0080 & 2.0550 & -1.886 \\
-1.0805 & -0.1563 & -1.6347 \\
2.4651 & -0.7324 & -1.3040 \\
0.5039 & 2.6006 & 1.3831 \\
-0.0210 & -2.7042 & -1.7194 \\
-1.3135 & 2.1890 & 1.7641 \\
-0.0128 & 2.7358 & -2.5980
\end{bmatrix}
\]  

(22)

\[
w_2 = \begin{bmatrix}
-0.2811 & -2.4392 & -0.2705 & 0.0237 & 0.0988 & -0.0266 & 0.0667 & 2.0293
\end{bmatrix}
\]  

(23)
4. Experiment Results and Discussion

This section discusses the experimental results of the proposed model, including four sub-sections. Section 4.1 gives the proposed model test output results, and Sections 4.2 and 4.3 compare and statistically analyze the test output of the proposed model and other models. The performance analysis of the proposed model is given in Section 4.4.

4.1. Test Output Results

Figure 10 shows the expected and actual output curves for two test data. To further reflect the relationship between noise frequency and the material sound absorption coefficient, the x-axis represents the frequency, and the y-axis the sound absorption coefficient.

![Figure 10. Relationship between frequency and sound absorption coefficient in the test data.](image)

As shown in Figure 10, when the sound frequency is below 2000 Hz, the sound absorption coefficient increases with the frequency, and when the sound frequency is above 2500 Hz, the sound absorption coefficient is generally between 0.9 and 1. This indicates that the model fits the relationship between the frequency and sound absorption coefficient well. The model actual output test data are roughly as expected, at 99.57%.

4.2. Comparison of Output Results

In order to evaluate the proposed model, the unoptimized neural network and cellular automata approaches and regression models of support vector machine, random forest and XGBoost are discussed below for comparison.

Figure 11 shows the actual and expected outputs of the three methods in a test dataset. The light blue upper triangle curve represents the expected output, and the green dot dotted curve, the pink lower triangle curve, and the blue star curve represent the actual output of one-dimensional cellular automata (1D CA), the proposed model, and the unoptimized neural network model approach, respectively. The model method in this paper is the best and is more accurate compared with the model based on CA in reference [40]. The reasons are determined as follows: to design a CA model based on an impedance tube acoustic experimental test, it is necessary to design the cell size, which is usually limited

\[
b_1 = \begin{bmatrix} 1.0344 \\ 0.0284 \\ 2.1240 \\ 0.7019 \\ 2.4726 \\ 2.3199 \\ -0.3505 \\ 1.2751 \end{bmatrix} \tag{24}
\]

\[
b_2 = -0.7410 \tag{25}
\]
for the convenience of modeling, leading to dense nodal distribution at the sound pressure antinode throughout the cellular space during cellular automata modeling. Consequently, for high frequency acoustic excitation, the exact position of the node is obscure, and the minimum sound pressure is not precise enough to accurately calculate the sound-absorption coefficient. Consequently, the relationship between the frequency and sound-absorption coefficient cannot be modeled precisely.

![Figure 11. Comparison between proposed model and other approaches.](image1)

Figure 11 shows the expected output and actual outputs results of our model, our model+ 10-fold cross-validation, and regression models of support vector machine, random forest and XGBoost. The light blue upper triangle curve represents the expected output, and the pink lower triangle curve represents the actual output of our model. As shown in Figure 12, our model is the best, and our model +10-fold cross-validation is better. In order to evaluate our model, statistical performance analysis is conducted below.

![Figure 12. Comparison between our model and other algorithms.](image2)

4.3. Validating Efficacy

Root mean squared error (RMSE) and the coefficient of determination ($R^2$) [41,42] were used to evaluate the efficacy of models including the neurogenetic model. The mathematical expression of RMSE and $R^2$ are shown in Equations (26) and (27), where $n$ is the number of
samples; \( \hat{y}_i \) is the \( i \)-th sample actual output during the experiment; \( y_i \) is the \( i \)-th sample expected output; and \( \bar{y} \) is the mean value of the expected output of \( n \) samples.

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} |\hat{y}_i - y_i|^2}
\]  \hspace{1cm} (26)

\[
R^2 = 1 - \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|^2}{\sum_{i=1}^{n} |y_i - \bar{y}|^2}
\]  \hspace{1cm} (27)

RMSE is a crucial indicator to check the effectiveness of any regression model; the lower the RMSE value, the better the model. \( R^2 \) explains the relationship between the actual and expected outputs, the larger the \( R^2 \) value, the better the model. Through experimental analysis, the RMSE values and \( R^2 \) values of our model, our model + 10-fold cross-validation (our model +10), 1D CA, the unoptimized neural network model (UNN), support vector machine (SVM), random forest (RF) and XGBoost regression models are calculated respectively. Because almost all these models use metaheuristics, which have a random search mechanism, in order to verify the accuracy and robustness of our model, through statistical analysis combined with experiments, we obtained the value range of RMSE, \( R^2 \) and convergence time, as shown in Table 3, and drew the means plot and intervals diagram; blue stars represent the mean, as shown in Figure 13. Our model has a maximum \( R^2 \) value, lower RMSE value, and shorter convergence time. Our model +10-fold cross-validation has longer convergence time. XGBoost has shorter convergence time but poor RSME and \( R^2 \) values.

<table>
<thead>
<tr>
<th>Evaluation Index</th>
<th>Our Model</th>
<th>Our Model +10-fold Cross-Validation</th>
<th>1D CA</th>
<th>Unoptimized Neural Network</th>
<th>Support Vector</th>
<th>Random Forest</th>
<th>XGBoost</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSE</td>
<td>(1.1728 ± 0.312) × 10^{-6}</td>
<td>(1.1366 ± 0.543) × 10^{-6}</td>
<td>2.1846 × 10^{-4}</td>
<td>(8.691 ± 1.136) × 10^{-3}</td>
<td>(8.783 ± 1.826) × 10^{-3}</td>
<td>(5.2765 ± 2.138) × 10^{-3}</td>
<td>(7.1863 ± 0.768) × 10^{-3}</td>
</tr>
<tr>
<td>( R^2 )</td>
<td>0.95 ± 0.046</td>
<td>0.94 ± 0.041</td>
<td>0.91</td>
<td>0.82 ± 0.06</td>
<td>0.65 ± 0.06</td>
<td>0.75 ± 0.07</td>
<td>0.72 ± 0.06</td>
</tr>
<tr>
<td>Convergence time/s</td>
<td>85 ± 6.25</td>
<td>121 ± 10.95</td>
<td>65.1</td>
<td>85 ± 10.9</td>
<td>78 ± 7.5</td>
<td>83 ± 10.8</td>
<td>55 ± 4.75</td>
</tr>
</tbody>
</table>

**Figure 13.** The means plot and intervals for our model and comparison algorithm. (a) \( R^2 \) means plot and intervals; (b) convergence time means plot and intervals.

4.4. Performance Analysis of the Proposed Model

In order to further understand the model mechanism, the following is a performance analysis of the proposed model.
4.4.1. Enhanced Solution Generation

An enhanced solution was obtained by a genetic algorithm as the initial model weight and threshold. Figure 14 shows the change of average fitness curve after running the genetic algorithm. The average fitness is for all individual fitness values, and the evolutional generation number is 10. For Generation 9, the average fitness value was at the minimum, and basically remained unchanged until the end of the evolution when the optimal weight and threshold were obtained, as shown in Equations (10)–(13). The optimal weight and threshold were input into the neural network model as the initial parameters needed for model training.

![Average fitness curve](image)

**Figure 14.** Change of average fitness curve.

4.4.2. Performance Analysis of Trained Model

The model performance was subsequently analyzed. As shown in Figure 15, the three curves respectively represent the MSE changes with the iteration number as training, validation, and test data were input for the model. There were 13 iterations in total. From iteration 6, the MSE curves decrease slowly, and the MSE of the validation data reaches the minimum value, i.e., $5.7468 \times 10^{-5}$ by iteration 7. Another 6 iterations were performed, and the MSE basically remained unchanged, that is, it no longer decreased, and thus, the model training was complete.

![Mean Squared Error (mse)](image)

**Figure 15.** Model performance.

The model effect was evaluated after training. Figure 16 shows the model conditions at the end of training, i.e., the changes of gradient, damping factor, and generalization
ability with the iteration number. Initially, the gradient was the highest, and tended to stabilize after iteration 7. The damping factor changed to $1 \times 10^{-6}$ from iteration 7, and then remained constant, indicating the model convergence. The generalization ability curve was at 0 for the first 7 iterations, indicating that the MSE decreased continuously during this stage. In the subsequent 6 iterations, the MSE was unchanged overall, and thus, the iteration was complete.

![Graph showing gradient, damping factor, and validation checks](image)

**Figure 16.** Model effect check. (a) Gradient change; (b) change of damping factor; (c) model generalization capability check.

To reflect the model effect more intuitively, the model regression ability was visualized, as shown in Figure 17, which visualizes the regression ability for all the model data, including the training, validation, and test data. The line formed by circles reflects the relationship between the input and expected output for different data types. The different color line represents the regression line, which reflects the relationship between the input data and the model actual output data for various data types. The figure illustrates that the regression line is basically consistent with the actual data line. In other words, the model regression effect is good.

![Graph showing model regression ability for different data types](image)

**Figure 17.** Cont.
5. Conclusions

In this work, a new acoustic model for porous hemp plastic composite is developed. Namely, samples were prepared, and the acoustic data of the materials was obtained by standing wave tube method experiments, while an acoustic model using improved genetic algorithm and neural network was subsequently proposed. The experimental results show that when the sound frequency is above 2500 Hz, the sound absorption coefficient is generally between 0.9 and 1, and the proposed model actual output test data are roughly as expected, at 99.57%. To prove the efficacy of the proposed model, experiments and statistical analyses were conducted. It was found that compared with one-dimensional cellular automata, unoptimized neural network modeling methods, support vector machine, random forest and XGBoost regression models, the modeling method proposed in this paper has the best RMSE and $R^2$ values, with more accurate, shorter convergence time, and less acoustic data required.

The proposed model was applied to predict sound absorption coefficient of the material, the results showed that it has better performance. In regard to experiments, it was observed that according to the designed three-layer neural network, the population size is 30 and the search space is from $-3$ to $3$. The improved genetic algorithm can probe the search space and find an optimal solution, which is used as the initial weights and thresholds of the neural network for training. A larger population size or search space is not recommended because the time to find the optimal solution will be increased. Moreover, our model still needs to be further studied for the material sound-absorption coefficient for frequencies exceeding 4000 Hz. Based on this, the subsequent study will focus on how to improve the material sound absorption coefficient for enhanced noise control.

Author Contributions: Conceptualization, H.Z.; methodology, H.W.; validation, H.Z.; formal analysis, H.W.; writing—original draft preparation, H.W.; writing—review and editing, Z.L.; supervision, Y.Z.; project administration, E.E.; funding acquisition, B.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China Youth Fun (Grant No. 51803029), the Heilongjiang Education Department Basic Business Fun Industrial Marijuana Project (Grant No.145109502), the Fundamental Research Funds for Heilongjiang Province Higher Education Institution (Grant Nos. 135409505, 135409421, 135509315, and 135209245), and Heilongjiang Province Higher Education Teaching Reform Project (Grant No. SJGY20200070).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.
Data Availability Statement: The data presented in this study are available in article.

Acknowledgments: The authors are grateful to their family for their support.

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

References
24. Nishawala, V.V.; Ostoja-starzewski, M.M.; Leamy, J.; Demmine, P.N. Simulation of elastic wave propagation using cellular automata and peridynamics, and comparison with experiments. *Wave Motion* 2016, 60, 73–83. [CrossRef]


