

Penetration Pattern Recognition Based on Artificial Neural Network

Wang Shuo^{1, a}, Shi Quan²

¹ Ordnance Engineering College, Shijiazhuang 050003, China

² Ordnance Engineering College, Shijiazhuang 050003, China

^a1031799669@qq.com

Abstract: In order to study the specific pattern of fragmentation penetrating the target plate under the influence of multi factors, this paper uses the artificial neural network method to identify the input parameters and obtain the corresponding target damage model. Based on the orthogonal design principal, this paper uses the ANSYS/LS-DYNA simulation to simulate the input data of the 60 groups of fragments penetrating the target plate as the input data trained by the neural network. In addition, three sets of data are selected as the verification data to the training effect of the neural network verification. The results show that the artificial network can effectively identify the specific damage pattern of fragment on the target under the multi-factor effect.

Keywords: artificial neural network; penetration model; finite element simulation; pattern recognition

1. Introduction

Fragmentation for the target of the penetration model is divided into three types: penetration, jumping, penetrating. The specific penetration factors are influenced by many factors. The main influencing factors are the speed of the fragments, the angle of incidence, the aspect ratio, the mass, the thickness and the shape of the target plate. These six factors have different effects on the damage model of the target, and the relevant scholars have discussed this problem. Such as Mi Shuangshan and others studied the finite element penetration of hard aluminum alloy with equivalent target plate as a model, and established the corresponding phase diagram. But the phase diagram is established for the specific shape of the projectile, the specific target thickness and other related fixed factors, although it is intuitive to understand, but in the case of a large number of fixed factors, it reflects the limited damage, can not fully reflect the discrete situation of multiple factors on the impact of penetration injury. In order to realize the judgment and recognition of the damage pattern of the input discrete samples, this study uses the artificial neural network as the platform to train the obtained samples to achieve the purpose of judging the damage pattern under the influence of multiple factors.

2. Artificial neural networks

2.1 Section Headings

Artificial Neural Networks (ANN) [1], is a description of the first-order characteristics of the human brain system, a theoretical mathematical model of the human brain and its activities. It is composed of a large number of processing units through the appropriate way of interconnection, is a large-scale nonlinear adaptive system, is a method of artificial intelligence research. The biggest feature of neural network is that it has the function of learning, through a variety of learning algorithms to adjust the weight of the network, the method is to make similar input similar output, making the neural network with generalization capabilities. The purpose of network learning is to minimize the error between the desired output value and the actual output.

In 1986, Rumelhart and Hinton proposed a neural network that can be self-organized toward a given input-output relationship. When the actual output on the output layer does not coincide with the given desired output, the BP algorithm uses the error of the output layer to estimate the error of the direct preamble of the output layer, and then estimate the error of the previous layer with this error. This

results in a process of passing the error at the output to the input of the network step by step in the opposite direction to the input signal transmission. Since the direction of error propagation is opposite to the direction of signal propagation, it is called an error back propagation network. The manual programming and debugging of the BP algorithm takes a long time, so using the neural network toolbox in MATLAB in this study can improve efficiency and problem solving.

3. BP Neural Network in MATLAB

3.1 Common Types of BP Network Transfer Functions in MATLAB

In the BP neural network, the logarithmic S-shaped function, the tangent S-shaped function and the linear function are often used as the transfer function of the neuron. The function graph and symbolic representation and the function expression are as follows:

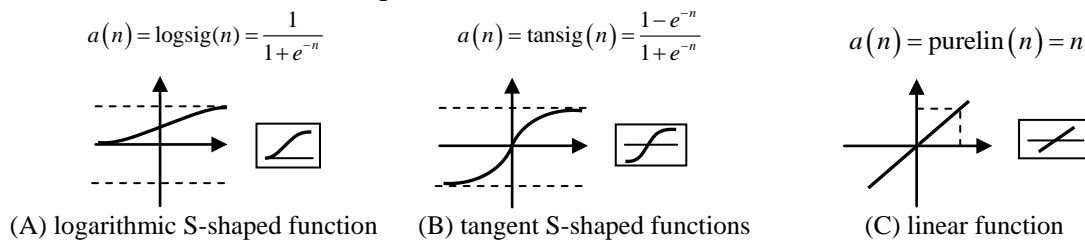


Figure 1 Commonly used network transfer function

3.2 BP network training algorithm

Neural networks need to constantly adjust the weights and thresholds, making the network input and output to meet the requirements of the relationship, this process is often not overnight, but the need to constantly adjust the weights and thresholds, this adjustment process is called "training". In this process, the neural network has the ability to map the input space to the output space. It is called the "learning" of the neural network, and the algorithm of adjusting the weights and thresholds is called learning rules or training algorithms. The main training methods are divided into the following three methods: standard BP algorithm (steepest gradient descent method), momentum BP algorithm, learning rate variable steepest descent BP algorithm.

3.3 Compile MATLAB simulation program

3.3.1 Determine the input sample

According to the analysis of influencing factors, the input sample vector contains six factors, namely, the speed of the fragment, the angle of incidence, the aspect ratio, the mass, the target thickness and the shape. In order to facilitate the BP program design, it is necessary to input the sample vector factor data processing, we must first consider the shape factor quantization.

- (1) Quantification of shape factor. According to the experimental analysis, the influence of the shape of the fragment on the penetration is: cone > cylindrical > diamond > spherical > square. So the hypothesis: square = 10; spherical = 20; diamond = 30; cylindrical = 40; conical = 50. In order to make the quantification closer to reality, combined with the analysis of the impact of each shape factor on the penetration index, the proportional difference method is used to accurately reflect the quantitative representation of each factor. The effect of each shape on the index is: square = 326.17; spherical = 332.66; diamond = 385.55; column = 416.28; conical = 621.41. The difference between the cone and the column is fixed by the equivalent ratio method, and the result is: square = 35.62; spherical = 35.93; diamond = 38.51; column = 40; taper = 50.
- (2) input processing of other factors. In order to make the difference between the input sample vectors not too large, to avoid the impact of pattern recognition, the speed input value is reduced by 100 times, the aspect ratio and the mass value are expanded by 10 times, and the angle and target thickness values remain unchanged.

3.3.2 Determine the output mode

In order to study the convenience of the problem, we set the three damage modes as follows: Jump: (1 0 0); Embed: (0 1 0); Penetration: (0 0 1). Based on the design principle of orthogonal test, 60 sets of data were used as training data by ANSYS / LS-DYNA simulation [4] [5]. Solid model and

pressure pit simulation results shown in Figure 1, part of the input sample and the corresponding output mode list 1 as follows.

Table 1 Network Input Sample and Expected Output Mode List

Sample number	velocity	incident angle	length to diameter ratio	mass	target thickness	shape	output mode
1	8	30	25	30	10	40	1 0 0
2	9	30	25	30	10	40	1 0 0
3	10	30	25	30	10	40	1 0 0
...
58	15	30	20	10	14	35.62	1 0 0
59	15	45	25	30	6	38.51	0 0 1
60	15	60	40	40	8	40	0 1 0

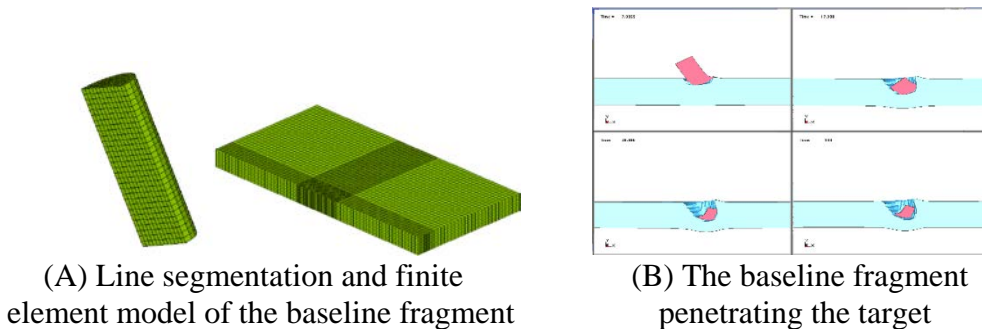


Figure 2 Solid model and pressure pit simulation results

3.3.3 Determine the parameters required to establish a BP network

It is proposed to establish two layers of BP network, the relevant parameters are determined as follows:

(1) The number of hidden neurons

For BP networks for pattern recognition / classification, according to previous experience, the following formula can be used to design.

$$n = \sqrt{n_i + n_o} + a \tag{1}$$

Where n is the number of hidden neurons; n_i is the number of input neurons, that is, the number of input sample elements; n_o is the number of output neurons, that is, the number of output samples; a is a number between 1 and 10. Type $n_i = 6$, $n_o = 3$ into the formula, take $a = 2$, so to determine the number of hidden neurons as 5.

(2) The selection of the activation function

According to the general procedure default, the first layer chooses the S-shaped function logsig, the second layer selects the linear function purelin. The selection of the transfer function depends mainly on what kind of output, network programming settings between 0 and 1, so the first layer and the second layer of the transfer function can be selected as a logarithmic S-shaped function. Therefore, this study will compare the training of different transmission functions to determine the best performance requirements to meet the transmission function selection.

(3) Choice of training methods and parameters

The training method was adopted to train the network with different training methods. The initial learning rate was 0.01, the number of training times was 20,000, and the expected error was 0.001. The initialized weights and thresholds of the network are randomly selected so that the results of each run will be different and run repeatedly until satisfied.

3.3.4 Compile MATLAB simulation program

In the established BP network, a total of two transfer functions are required, since the target output is the number between 0 and 1. As far as possible to avoid a negative value, the input layer and the output layer of the transfer function do not use the first tangent S-shaped function. The input layer selects the logarithmic S-shaped or linear transfer function for the output layer to be selected for the

weighted and as fast convergence as the logarithmic S-shaped function. The training of the network uses the commonly used traingd, traingdm, traingda and traingdx functions [6]. There are eight cases, as shown in Table 2.

For the above eight cases, respectively, the establishment of the network, the network training, the number of training for 20000 times, the error performance index of 0.001, other network parameters are used in MATLAB default value, due to the BP network each training initialized weights and thresholds are taken randomly. Therefore, the same procedure for each training error performance are different, so the number of training for each situation 30 times, select the minimum error performance of the situation summarized as shown in Figure (a) - (h) below:

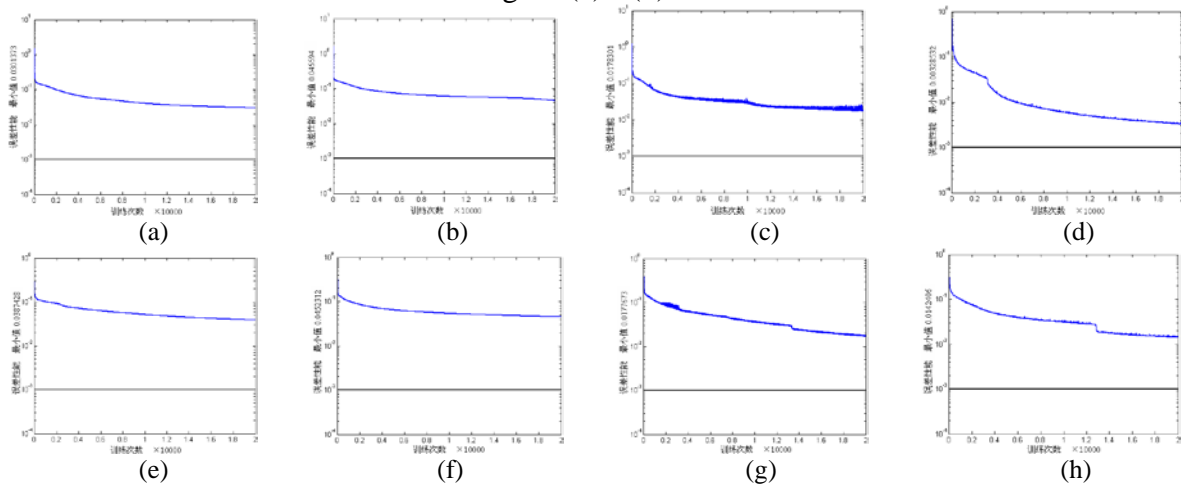


Figure 3 Error performance curves for each training

According to the table (Table 2), we can see that the purelin function of the output layer transfer function is closer to the error performance than the minimum error performance of the training function using the logsig function in the case of other cases. From the above table can be seen, in addition to tragedm function slightly different, the other are the above situation, so the final MATLAB program selected the first four cases established.

Table 2 Summary of training

f^2 Output layer transfer function	Training function	Number of training	Minimum error	Error performance index
purelin	traingd	30	0.03013	0.001
	traingdm	30	0.04559	0.001
	traingda	30	0.01783	0.001
	traingdx	30	0.00329	0.001
logsig	traingd	30	0.03874	0.001
	traingdm	30	0.04523	0.001
	traingda	30	0.01777	0.001
	traingdx	30	0.01424	0.001

3.4 Simulation results and verification

It can be seen from the table, after the network training, simulation of the various output mode results after rounding, and the actual expected output coincides with the input sample to meet the output mode requirements. Although the network error performance value does not meet the required value of 0.001, but for the pattern recognition has been within the allowable error range. Change the value of the input factor, take three other samples different from the training samples to verify, verify the test number 61 *, 62 *, 63 *, verify the sample input factor and the simulation output and the expected output as shown in Table 4.

Table 3 Neural network simulation results data

Test number	Expected output			Simulation results		
1	1	0	0	1.0265	-0.0307	0.0042
2	1	0	0	1.0249	-0.0290	0.0041
3	1	0	0	0.9209	0.0778	0.0013
...
58	1	0	0	1.0213	-0.0253	0.0040
59	0	0	1	0.0016	0.0229	0.9755
60	0	1	0	0.0223	0.9759	0.0019

Table 4 Verify sample input and output

Verify the sample	Velocity	Incident angle	Aspect ratio	Quality	Target thickness	shape	Expected output			Simulation results		
61*	9	5	15	20	7	35.62	1	1	0	-0.0217	1.0430	-0.0213
62*	12	20	30	40	9	40	0	0	1	-0.0072	0.1194	0.8878
63*	14	40	45	70	11	50	0	0	1	-0.0133	0.0945	0.9189

The simulation results show that it is feasible to identify the damage pattern with the neural network. The simulation results can be used to predict and identify the damage pattern of different input samples.

4 Conclusions

In this paper, the neural network toolbox in MATLAB is used to design the BP network. Based on the principle of orthogonal experiment design, 60 sets of small projectile penetrating target model are simulated by ANSYS / LS-DYNA as the experimental sample data Simulation program for training. By comparing the network error performance obtained by using different transfer function and training algorithm, the output layer is purelin function, and the two-layer BP network of traingdx algorithm is used to simulate the simulation. And the error accuracy of 0.00329, meets the test requirements. It is proved that the output mode of the network is consistent with the actual output, and it is feasible to prove the pattern recognition of the damage through the neural network.

5. References

- [1] Jiang Zongli. Introduction to Artificial Neural Networks [M]. Higher Education Press, 2003.
- [2] Gao Jun. Establishment of artificial neural network principle and simulation example [M]. Machinery Industry Press, 2003.
- [3] Wen Xin, Zhou Lu, Li Xiang et al. MATLAB neural network simulation and application [M]. Science Press, 2003.
- [4] CHEN Jian, MEI Shuang-shan, ZHANG Xi-en. Study on meshing based on LS-DYNA spherical fragmentation armor [M]. Journal of Projectiles, Rockets and Guidance, 2005, 25 (2): 51-55.
- [5] XIANG Yong-yong, LI Yu-chun, ZHANG Sheng-min. Experimental dynamic analysis based on ANSYS / LS-DYNA8.1 [M]. Tsinghua University Press, 2005.
- [6] Zhou Kaoli, Kang Yaogong. Neural network model and its MATLAB simulation program design [M]. Tsinghua University Press, 2005.