sensors

MDPI

Article

Research on Weigh-in-Motion Algorithm of Vehicles Based on BSO-BP

Suan Xu *, Xing Chen, Yaqiong Fu, Hongwei Xu and Kaixing Hong

School of Mechanical and Electrical Engineering, China Jiliang University, Hangzhou 310018, China; p1901085204@cjlu.edu.cn (X.C.); fuyaqiong@cjlu.edu.cn (Y.F.); xhw@cjlu.edu.cn (H.X.); hongkaixing@cjlu.edu.cn (K.H.)

* Correspondence: xusuan@cjlu.edu.cn; Tel.: +86-0571-86914559

Abstract: Weigh-in-motion (WIM) systems are used to measure the weight of moving vehicles. Aiming at the problem of low accuracy of the WIM system, this paper proposes a WIM model based on the beetle swarm optimization (BSO) algorithm and the error back propagation (BP) neural network. Firstly, the structure and principle of the WIM system used in this paper are analyzed. Secondly, the WIM signal is denoised and reconstructed by wavelet transform. Then, a BP neural network model optimized by BSO algorithm is established to process the WIM signal. Finally, the predictive ability of BP neural network models optimized by different algorithms are compared and conclusions are drawn. The experimental results show that the BSO-BP WIM model has fast convergence speed, high accuracy, the relative error of the maximum gross weight is 1.41%, and the relative error of the maximum axle weight is 6.69%.

Keywords: WIM; BSO algorithm; wavelet transform; BP neural network

check for updates

Citation: Xu, S.; Chen, X.; Fu, Y.; Xu, H.; Hong, K. Research on Weigh-in-Motion Algorithm of Vehicles Based on BSO-BP. *Sensors* **2022**, *22*, 2109. https://doi.org/10.3390/s22062109

Academic Editor: Biswajeet Pradhan

Received: 23 January 2022
Accepted: 4 March 2022
Published: 9 March 2022

1. Introduction

The problem of vehicle overload seriously endangers traffic safety, not only easily causing traffic accidents, but also causing fatal damage to the structure of roads and bridges [1]. Weight measurement is the most direct way to identify overloaded vehicles. Static weighing is a very mature method for weighing stationary vehicles with an accuracy of 0.1%. However, static weighing of moving vehicles on the road requires the vehicle to stop, which is likely to cause traffic jams and affect the efficiency of vehicle traffic. Compared with static weighing, weigh-in-motion (WIM) can measure the weight of a moving vehicle without affecting vehicle traffic, so it has a higher detection efficiency. However, the application of WIM is limited due to its low precision. A proper estimation of weighing accuracy is crucial for WIM systems to be efficient in direct enforcement of overloading [2]. The study of Lhoussaine Oubrich et al. shows that the accuracy of a WIM system is mainly affected by the quality of the road surface, axle load, speed, and suspension system structure [3]. In addition, the distance between the sensor and the wheel, the installation depth position of the sensor, and the pavement temperature also affect the weighing accuracy of the WIM system [4,5]. There are many factors that affect the accuracy of WIM, and the relationship between different influencing factors is unknown. Therefore, it is difficult to determine the relationship between the accuracy of WIM and different factors. Artificial neural networks (ANN) possess the characters of strong nonlinear mapping, adaptive learning, fault-tolerance performance, and robustness. They are widely used in pattern recognition, data processing, fault diagnosis, and so on. Some researchers have used neural networks to improve the accuracy of WIM systems. Sungkon Kim et al. used ANN to analyze the WIM signal of the main girders and the weighing information of the vehicle to calculate the vehicle weight and axle weight of the vehicle. Their study shows that the accuracy of ANN method is higher than that of the influence line method in axle load calculation [6]. Yun Zhou et al. used a deep convolutional neural network to distinguish vehicle weight and bridge structure response to estimate vehicle attributes [7]. Zhixin Jia et al. used the BP model to identify the sensor closest to the tire–pavement contact area, and then used the signal from the sensor to predict the vehicle weight. In their experiments, 96.5% of the test samples had a relative error of less than 5% [8]. However, the BP neural network is sensitive to the initial weights and can easily fall into the local optimum, which makes its performance unable to reach the optimum. Optimization algorithms such as the swarm intelligence algorithm and genetic algorithm (GA) can adjust the initial weight of the BP neural network and reduce the possibility of falling into the local optimum. Tiantian Wang et al. compared the performance of BSO, particle swarm optimization (PSO), GA, and the grasshopper optimization algorithm (GOA) on 23 benchmark functions. Experiments show that the performance of BSO algorithm is better than other algorithms [9]. The BSO algorithm combines the information sharing mechanism of the PSO algorithm and the search mechanism of the beetle antennae search (BAS) algorithm is applied in different fields [10]. Jianming Zhou et al. used the BP neural network optimized by the BSO algorithm to predict the parameters of each part of the twisted pair and improved the prediction accuracy of the crosstalk of the twisted pair [11]. Lei Wang adjusted the control parameters of the robot hand trajectory planning through the BSO algorithm and compared it with the PSO algorithm and the genetic algorithm to verify the superiority of the BSO algorithm [12]. F.N. Al-Wesabi et al. used BSO to optimize the weights and bias parameters of the least squares support vector machine to achieve a better classification effect [13]. Parminder Singh et al. proposed an adaptive neuro-fuzzy inference system for heart disease and multi-disease diagnosis and used the BSO algorithm to optimize the parameters of the inference system, which improved the accuracy and precision of diagnosis [14]. Aiming at the defects of the BP neural network and the superiority of the BSO algorithm, this paper proposes a WIM model based on the BSO algorithm and BP neural network. Firstly, the structure and principle of the WIM system used in this paper are analyzed. Secondly, the WIM signal is denoised and reconstructed by wavelet transform. Then, a BP neural network model optimized by the BSO algorithm is established to process the WIM signal. Finally, the predictive ability of BP neural network models optimized by different algorithms are compared and conclusions are drawn. The purpose of this paper is to study the effect of improving the weighing accuracy of dynamic weighing system based on a BP neural network optimized by different algorithms.

The paper is organized as follows:

1. Structure and principle of the WIM system.
2. Wavelet transform algorithm.
3. BSO-BP algorithm.
4. Experimental results and analysis.
5. Conclusions.

2. Structure and Principle of the WIM System

The components of the WIM system include an integrated weighing platform, embedded strain sensor, signal processing circuit, ground sensing coil, industrial computer, and cloud platform. A structural diagram of a WIM system is shown in Figure 1. When the vehicle passes the weighing platform, the embedded strain sensor deforms under pressure and converts the weight signal into the voltage signal. The signal processing circuit converts the voltage signal into the digital signal. The industrial computer calculates parameters such as vehicle weight, vehicle speed, and number of axles, and uploads these parameters to the cloud platform via the network.

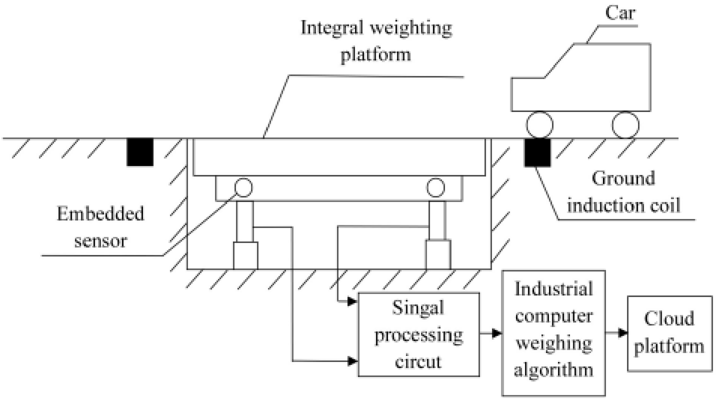

Figure 1. Structural diagram of a WIM system.

The test platform is built on the road and the weighing platform is shown in Figure 2.

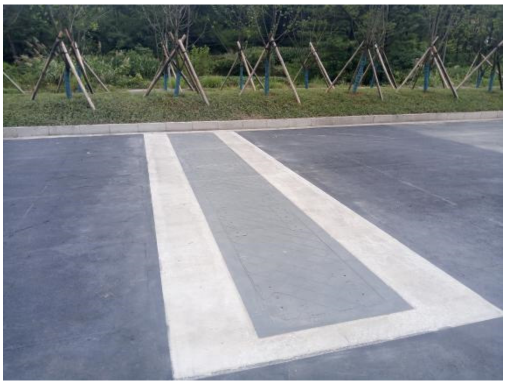

Figure 2. Weighing platform.

Two weighing platforms are installed in one lane. The two platforms are side by side. A single weighing platform has a horizontal width of 1.75 m and a longitudinal width of 0.8 m. The bottom of the weighing platform is supported by two symmetrically placed I-shaped beams. Two pressure sensors are installed symmetrically on an I-shaped beam. The structure diagram of the weighing platform is shown in Figure 3.

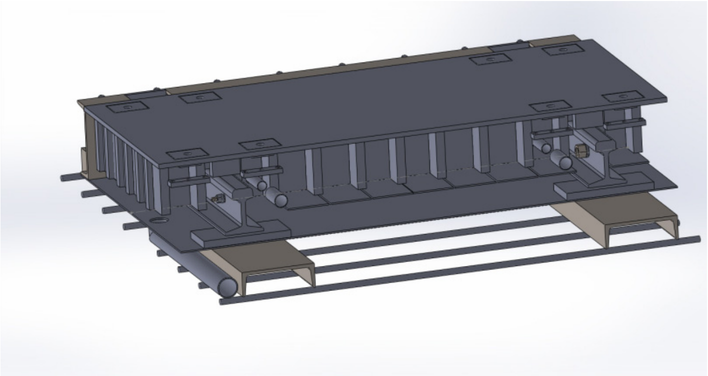

Figure 3. Schematic diagram of the scale structure.

3. Pre-Processing of the WIM Signal

Wavelet transform is a transform analysis method which is widely used in the fields of fault detection, image processing, and signal analysis [15–19]. Since the WIM signal is a discrete signal, this paper uses discrete wavelet transform to process the WIM signal.

Daubechies (*dbN*) wavelet function is sensitive to irregular signals and it is widely used in signal analysis. *N* is the order of the *dbN* wavelet function. When *N* is equal to 4, the *dbN* wavelet has the characteristics of good orthogonality and large vanishing moment. Multi-layer decomposition and reconstruction of a weighing signal by *db4* wavelet can reduce the interference of noise signal and obtain a real WIM signal.

The waveforms before and after the wavelet transform are shown in Figures 4 and 5, respectively.

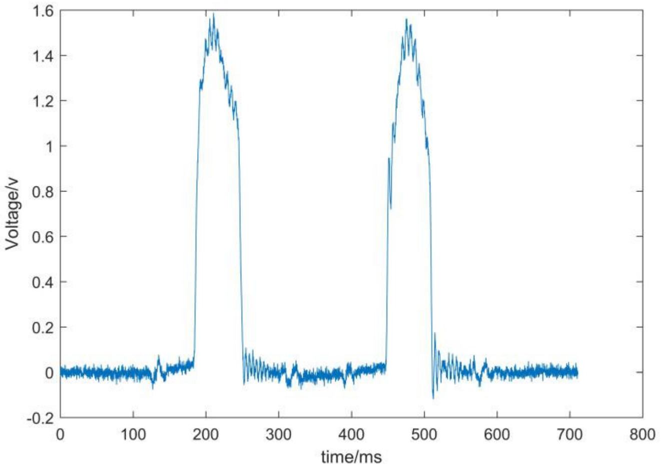

Figure 4. Weighing signal without wavelet transform.

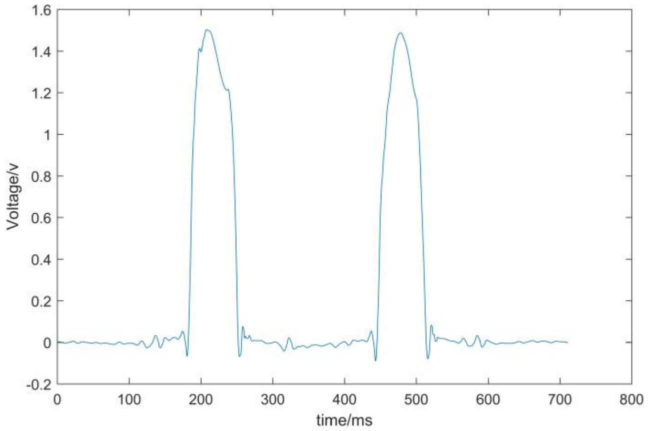

Figure 5. The weighing signal after wavelet transform.

According to Figures 4 and 5, it can be seen that the high-frequency noise in the waveform is significantly reduced. The results show that the wavelet transform reduces the interference of noise and makes the waveform closer to the ideal waveform.

4. BSO-BP Algorithm

4.1. BP Neural Network

The BP neural network has strong nonlinear mapping ability, which is widely used in the fields of pattern recognition, parameter estimation, motion control, and classification [20–24]. The structure of the BP neural network is shown in Figure 6.

In Figure 6, X_t $(1 < t < n)$ is the input nodes and Y_u $(1 < u < m)$ is the output nodes. In addition, w_{ij} represents the weight of the input layer to the hidden layer and w_{jk} represents the weight of the hidden layer to the output layer.

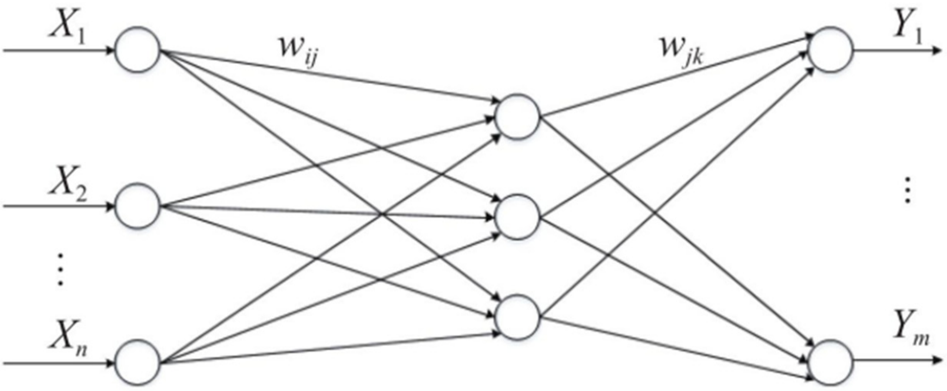

Figure 6. Structure diagram of the BP neural network.

4.2. PSO Algorithm

The PSO algorithm can search the best region in complex space through group cooperation [25]. Each particle constantly adjusts its search behavior by learning from its own experience and that of other particles. Particles update their velocity and position by the following formulas during each iteration:

$$V_i(k+1) = \omega V_i(k) + c_1 r_1[p_{ibest} - X_i(k)] + c_2 r_2[p_{gbest} - X_i(k)] \tag{1}$$

$$X_i(k+1) = X_i(k) + V_i(k) \tag{2}$$

where V_i is the velocity of the particle i, X_i is the position of the particle i, p_{ibest} is the historical optimal position of the particle i, p_{gbest} is the global optimal position, k is an evolution algebra, ω is an inertia weight, c_1 and c_2 represent learning factors, and r_1 and r_2 represent random numbers between (0,1). In addition, the position and speed of the particles need to be limited by parameters.

4.3. Principle of BAS and BSO Algorithms

The BAS algorithm imitates the beetle's search mechanism and random behavior [26]. The beetle uses its left and right antennae to detect the strength of food smells and adjusts the direction of its search. Based on this simple principle, it can easily find food. Yinyan Zhang et al. studied the convergence of the BAS algorithm, and the results show that BAS algorithm has good performance [27]. The steps of the BAS algorithm are as follows:

1. Create a normalized random vector b. The calculation formula is as follows:

$$b = \frac{\mathrm{rands}(m,1)}{\|\mathrm{rands}(m,1)\|} \tag{3}$$

where rands() is a random function, m is a spatial dimension, and b represents the vector from the left antennae to the right antennae;

2. Create a spatial search model for the left and right antennae:

$$x_{rk} = x^k + d^k \times b/2 \tag{4}$$

$$x_{lk} = x^k - d^k \times b/2 \tag{5}$$

where x_{rk} represents the position coordinates of the right antennae at the kth iteration, x_{lk} represents the position coordinates of the left antennae at the kth iteration, x^k represents the centroid coordinates of the beetle at the kth iteration, and d^k represents the search distance between the left and right antennae at the kth iteration;

3. $f(x_{lk})$ and $f(x_{rk})$ are calculated using the fitness function f(x);
4. Iteratively update the position of the beetle through search behavior:

$$x^{k+1} = x^k - \delta^k \times b \times \mathrm{sign}(f(x_{rk}) - f(x_{lk})) \tag{6}$$

where δ^k represents the step size of the tth iteration, and sign() represents the sign function;

5. Update search distance and step size:

$$d^k = \eta_d \times d^{k-1} + d_0 \tag{7}$$

$$\delta^k = \eta_\delta \times \delta^{k-1} + \delta_0 \tag{8}$$

where η_d is is the attenuation coefficient of the search distance, η_δ is the attenuation coefficient of the step size, and d_0 and δ_0 represent the minimum distance threshold and the minimum step threshold. Set d_0 and δ_0 to avoid the value of d^k and δ^k being 0.

The objective function is calculated only twice in each iteration, so the BAS algorithm has strong global search ability [28]. However, the single search characteristic of the BAS algorithm makes it easy to fall into the local optimum when processing high-dimensional data, and it can only solve the single-objective optimization problem [29]. The BSO algorithm combines the search method of the BAS algorithm and the group information sharing mechanism of the PSO algorithm, which not only accelerates iterative convergence, but also reduces the possibility of falling into local optimum [30]. In the BSO algorithm, the random direction b in the expression of the antennae is replaced by the speed of the PSO algorithm. The expression is as follows:

$$x^k{}_{ird} = x^k{}_{id} + v^k{}_{id} \times d^k/2 \tag{9}$$

$$x^k{}_{ild} = x^k{}_{id} - v^k{}_{id} \times d^k/2 \tag{10}$$

where $x^k{}_{ild}$ and $x^k{}_{ird}$ are the position vectors of the left and right antennae of particle i in the dth dimension at the kth iteration.

Increment of searching behavior of beetle antennae:

$$\xi^{k+1}{}_{id} = \delta^k \times v^k{}_{id} \times \mathrm{sign}(f(x^k{}_{ird}) - f(x^k{}_{ild})) \tag{11}$$

where $\xi^{k+1}{}_{id}$ represents the increment of the search behavior at the (k + 1)th iteration.

The updated formula for the position and velocity of the beetle swarm is as follows:

$$v^{k+1}{}_{id} = \omega v^k{}_{id} + c_1 r_1(\mathrm{pbest}^k{}_{id} - x^k{}_{id}) + c_2 r_2(\mathrm{gbest}^k{}_{id} - x^k{}_{id}) \tag{12}$$

$$x^{k+1}{}_{id} = x^k{}_{id} + v^{k+1}{}_{id} + \xi^{k+1}{}_{id} \tag{13}$$

where $x^k{}_{id}$ represents the position of the beetle i at the kth iteration, $x^{k+1}{}_{id}$ represents the speed of the beetle i at the (k + 1)th iteration, c_1 and c_2 represent learning factors, r_1 and r_2 represent random numbers between (0,1), $\mathrm{pbest}^k{}_{id}$ is the historically optimal position of the beetle i, $\mathrm{gbest}^k{}_{id}$ represents the historical global optimal position of the beetle swarm.

In the PSO algorithm, inertia weight ω is a fixed value. The research shows that the inertia weight has a great influence on the searching range of particles. This paper introduces the inverted S-shaped function to adjust the inertia weight to improve the searching ability of the algorithm. The updated formula of the inertia weight is as follows:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{1 + e^{a-bt}} \tag{14}$$

where ω represent random numbers between (0.4,0.9), b is equal to 0.2, and a is equal to 5.

4.4. Establishment of the BSO-BP WIM Model

The random initial weight of the BP neural network makes the network unstable and easily falls into local optimum. The BSO algorithm is used to optimize the weights and thresholds of the BP neural network, which can reduce the possibility of the network falling into a local optimum. The optimization steps are as follows:

1. The data processed by wavelet transform is used as the input sample of BSO-BP neural network;
2. Set the size of the beetle swarm, the maximum number of iterations, the inertia weight, and the search space dimension, etc. The search space dimension is calculated as follows:

$$D = m \times n + r \times n + n + r \tag{15}$$

where m represents the number of neurons in the input layer, n represents the number of neurons in the hidden layer, and r represents the number of neurons in the output layer;

3. Randomly generate the position and speed of the beetle. According to formula 25, the fitness function value is calculated. Save the individual optimal value and the group optimal value:

$$f(x_i) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i' - x_i\right)^2 \tag{16}$$

where N represents the number of training samples, x'_i represents the predicted value of the training sample, and x_i represents the true value of the training samples;

4. Update the step size of the beetle:

$$\delta^{k+1} = eta \times \delta^{k} \tag{17}$$

where k is the number of iterations and eta is equal to 0.95.

5. Iterative optimization. Iteratively update the position, velocity and inertia weight of the beetle. The individual optimal value of the beetle and the group optimal value of the beetle swarm are updated according to the fitness value of the beetles;
6. The final iteration result is taken as the initial weight and threshold value of BP neural network;
7. Train the BSO-BP WIM model. Continuously update the weight and threshold of the network according to the error until the set accuracy is reached.

5. Experimental Results and Analysis

A total of three vehicles were tested. Among them, there is one two-axle car, one four-axle car, and one six-axle car. In addition, each vehicle is divided into two states, empty and full, to simulate different cargo loading conditions. Before the test, three vehicles were statically weighed to obtain the static gross weight and static axle weight. After the test, the dynamic gross weight, dynamic axle weight, and vehicle speed of the vehicle are recorded. The test vehicles pass as close to a constant speed as possible, and the vehicle speed is less than 70 km/h. A total of 261 groups of experimental data were collected. According to the experimental data, the gross weight dataset and the axle weight dataset were respectively composed. The gross weight dataset includes vehicle speed, number of axles, dynamic gross weight, and static gross weight. The axle weight dataset includes vehicle speed, number of axles, axle number, dynamic axle weight, and static axle weight. The parameters of all tested vehicles are shown in Table 1.

Table 1. Data from three test vehicles.

Number of Axles	Load	First Axle (kg)	Second Axle (kg)	Third Axle (kg)	Fourth Axle (kg)	Fifth Axle (kg)	Sixth Axle (kg)	Static Weight (kg)
2	Empty	1870	1440	/	/	/	/	3310
	Full	2358	2932	/	/	/	/	5290
4	Empty	5512	4746	6326	5976	/	/	22,560
	Full	7339	8001	24,756	25,434	/	/	65,530
6	Empty	4847	2905	2981	1197	1823	2957	16,710
	Full	5591	7123	7318	10,315	11,523	12,080	53,950

5.1. Data Pre-Processing

This paper used MATLAB to construct a BSO-BP WIM model. The input data and output data were normalized using the mapminmax function to improve the training speed of the samples.

After the training of the BSO-BP WIM model, the prediction results of the model were processed by inverse normalization.

5.2. Establishment of WIM Model and Parameter Selection

The input layer has three nodes, namely dynamic vehicle weight, vehicle speed, and axle number. The purelin is the transfer function from the hidden layer to the output layer. In the BSO algorithm, the population size N is 10, the learning factors c_1 and c_2 are both equal to 2, and the convergence accuracy E is 0.001. The training time of the BP network is 20,000, the learning rate is 0.01, and the target error is 0.000004. The gross weight test set and the gross weight training set are randomly selected from the gross weight data set. Among them, 10% of the data is used as the gross weight test set, and the remaining 90% of the data is used as the gross weight training set for model training. Axle weight test set and axle weight training set are established in the same way.

5.3. Prediction of Gross Vehicle Weight and Analysis of Results

The BSO-BP WIM model, the BP WIM model, and the PSO-BP WIM model are established through the gross weight training set. The prediction errors of the three models on the gross weight test set are compared. The comparison result is shown in Figure 7.

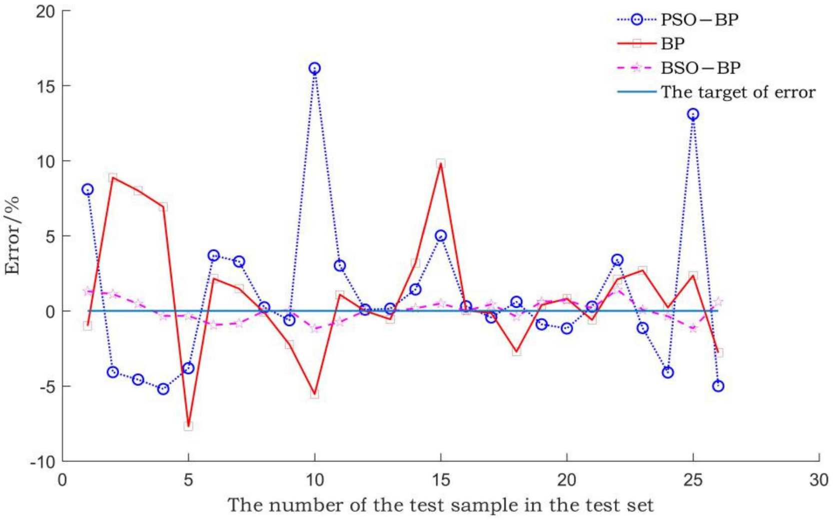

Figure 7. Comparison of gross weight prediction errors of different models.

In Figure 7, the gross weight prediction error of the BSO-BP WIM model is significantly lower than that of the BP WIM model and the PSO-BP WIM model.

The average gross weight relative error and the maximum gross weight relative error of the three models are shown in Table 2.

Table 2. Mean gross weight relative error and maximum gross weight relative error for the three models.

Model	Mean Gross Weight Relative Error/%	Maximum Gross Weight Relative Error/%
BP	2.83	9.82
PSO-BP	2.62	18.78
BSO-BP	0.53	1.41

As can be seen from Table 2, the average gross weight relative error and maximum gross weight relative error of the BSO-BP WIM model are lower than those of the PSO-BP WIM model and the BP WIM model. The mean relative error of the gross weight test set is reduced from 5.58% to 0.53%, and the maximum relative error is reduced from 15.07% to 1.41%. The results show that the BSO algorithm improves the accuracy and generalization ability of the BP neural network, and its optimization effect is higher than that of the PSO algorithm.

The fitness function values of the PSO-BP neural network and BSO-BP neural network are compared. The fitness function value is the absolute value of the difference between the predicted gross weight and the static gross weight. The comparison result is shown in Figure 8.

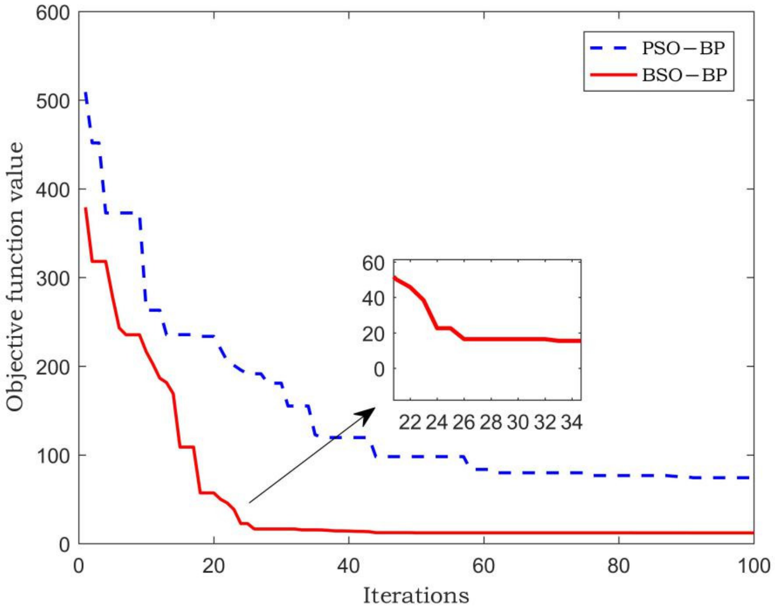

Figure 8. Iterative comparison of fitness function values for gross weight.

In Figure 8, the PSO-BP algorithm needs more than 50 iterations to reach the optimum and its fitness value is about 70. The BSO-BP algorithm only needs 25 iterations, and its fitness value is about 20. The results show that the BSO algorithm can improve the convergence speed, enhance the optimization ability, and prevent the algorithm from falling into local optimum.

5.4. Prediction of Axle Weight and Analysis of Results

The BSO-BP WIM model, the BP WIM model, and the PSO-BP WIM model are established through the axle weight training set. The prediction errors of the three models on the axle weight test set were compared. The comparison results are shown in Figure 9.

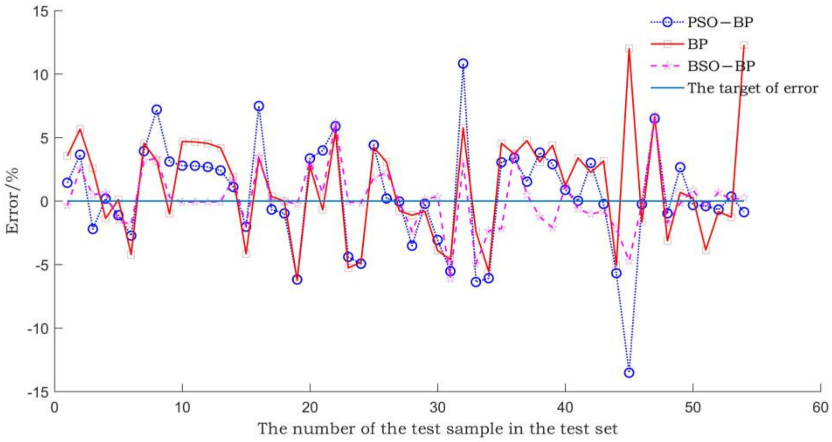

Figure 9. Comparison of axle weight prediction errors of different models.

The average axle weight relative error and the maximum axle weight relative error of the three models are shown in Table 3.

Table 3. Mean axle weight relative error and maximum axle weight relative error of three models.

Model	Mean Axle Weight Relative Error/%	Maximum Axle Weight Relative Error/%
BP	3.51	12.28
PSO-BP	3.12	13.52
BSO-BP	1.61	6.69

As can be seen from Table 3, the axle weight prediction error of the BSO-BP WIM model is smaller than that of other models. The mean relative error of the axle weight test set is reduced from 3.38% to 1.61%, and the maximum relative error is reduced from 6.86% to 6.69%. From the comparison results of Tables 2 and 3, it can be seen that the BSO algorithm has a higher prediction accuracy for gross weight or axle weight.

Three axle weight samples with different axle numbers were selected from the axle weight dataset. The original waveforms of the three axle weight samples were corrected according to the predicted axle load. The corrected waveforms, original waveforms, and static waveforms of the three axle weight samples were compared, respectively. The comparison results are shown in Figures 10–12.

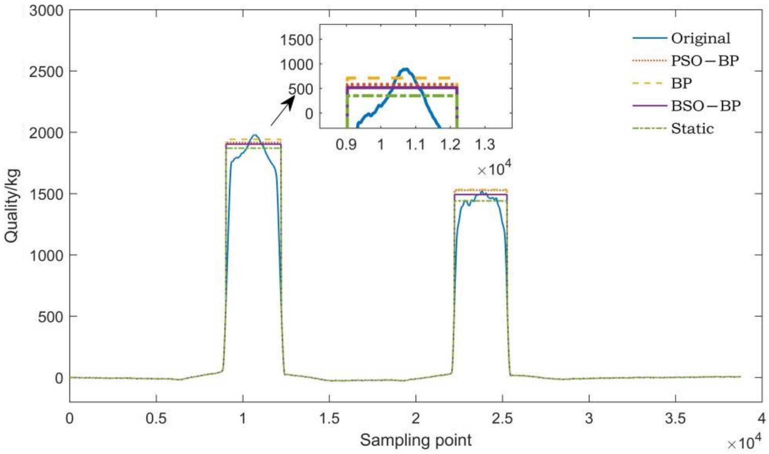

Figure 10. Corrected axle weight waveform comparison of a two-axle vehicle.

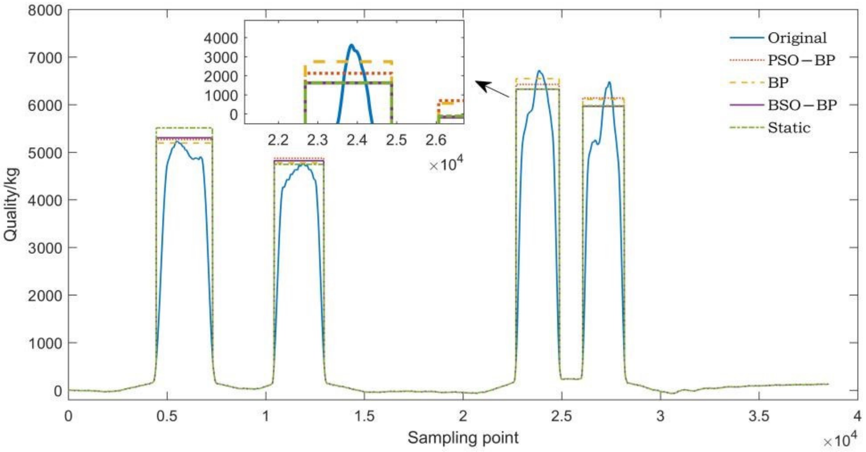

Figure 11. Corrected axle weight waveform comparison of a four-axle vehicle.

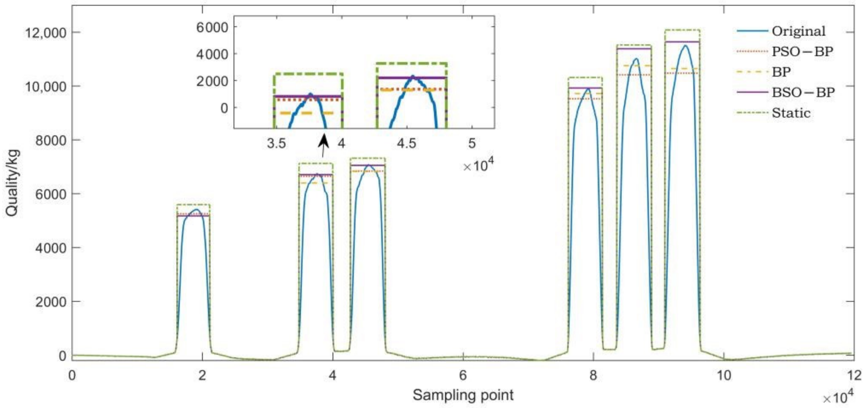

Figure 12. Corrected axle weight waveform comparison of a six-axle vehicle.

As can be seen from Figures 10–12, the predicted axle weight of the BSO-BP WIM model is the closest to the static axle weight. The comparison of axle weight waveforms is more intuitive, which further verifies that the BSO algorithm has better performance.

6. Conclusions

This paper proposed a BP WIM model optimized based on the BSO algorithm to improve the accuracy of the WIM system. The BSO algorithm combines the information sharing mechanism of the PSO algorithm and the single search mechanism of the BAS algorithm and has a stronger ability to find capabilities. After the initial weights of the BP neural network are optimized by the BSO algorithm, the generalization and convergence capabilities of the network are improved. After the optimization of the BSO-BP model, the average relative error of axle weight of test set was reduced from 3.38% to 1.41%, the maximum relative error of axle weight was reduced from 6.86% to 6.69%, the average relative error of gross weight was reduced from 5.89% to 0.53%, and the maximum relative error of gross weight was reduced from 15.07% to 1.41%. Experimental results show that BSO-BP model can improve the accuracy of the WIM system, and the prediction accuracy of gross weight was higher than that of axle weight. Meanwhile, the prediction accuracy of gross weight and axle weight using the BSO-BP model is higher than that using the PSO-BP and BP models. The results verify the superiority of the BSO algorithm, indicating that the optimization ability of the algorithm is stronger than that of the PSO algorithm.

Although the BSO algorithm can improve the accuracy of the dynamic weighing system, its performance is easily affected by parameters, and it takes more time to adjust the parameters of the model to achieve better results. In addition, limited by the experimental conditions, the number of test vehicles is small, and the actual vehicle traffic situation cannot be completely simulated.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/s22062109/s1.

Author Contributions: Conceptualization, Y.F. and H.X.; software, X.C.; investigation, K.H.; resources, S.X.; data curation, X.C.; writing—original draft preparation, X.C.; writing—review and editing, S.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Public Welfare Technology Application Research Project of Zhejiang Province Science and Technology Department (LGG20E050013), National Natural Science Foundation of China (grant nos.51105348).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available in supplementary material.

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

References

1. Park, S.; On, B.W.; Lee, R.; Park, M.W.; Lee, S.H. A Bi-LSTM and k-NN Based Method for Detecting Major Time Zones of Overloaded Vehicles. *Symmetry* **2019**, *11*, 1160. [CrossRef]
2. Burnos, P.; Gajda, J.; Sroka, R. Accuracy criteria for evaluation of weigh-in-motion systems. *Metrol. Meas. Syst.* **2018**, *25*, 743–751.
3. Oubrich, L.; Ouassaid, M.; Maaroufi, M. Dynamic loads, source of errors of high speed weigh in motion systems. In Proceedings of the 2017 14th International Multi-Conference on Systems, Signals & Devices (SSD), Marrakech, Morocco, 28–31 March 2017; pp. 354–359.
4. Qin, T.; Lin, M.; Cao, M.; Fu, K.; Ding, R. Effects of Sensor Location on Dynamic Load Estimation in Weigh-in-Motion System. *Sensors* **2018**, *18*, 3044. [CrossRef] [PubMed]
5. Burnos, P.; Gajda, J. Thermal Property Analysis of Axle Load Sensors for Weighing Vehicles in Weigh-in-Motion System. *Sensors* **2016**, *16*, 2143. [CrossRef]
6. Kim, S.; Lee, J.; Park, M.S.; Jo, B.W. Vehicle Signal Analysis Using Artificial Neural Networks for a Bridge Weigh-in-Motion System. *Sensors* **2009**, *9*, 7943–7956. [CrossRef]
7. Zhou, Y.; Pei, Y.; Zhou, S.; Zhao, Y.; Hu, J.; Yi, W. Novel methodology for identifying the weight of moving vehicles on bridges using structural response pattern extraction and deep learning algorithms. *Measurement* **2021**, *168*, 108384. [CrossRef]
8. Jia, Z.; Fu, K.; Lin, M. Tire-Pavement Contact-Aware Weight Estimation for Multi-Sensor WIM Systems. *Sensors* **2019**, *19*, 2027. [CrossRef]
9. Wang, T.; Yang, L.; Liu, Q. Beetle Swarm Optimization Algorithm: Theory and Application. *Filomat* **2020**, *34*, 5121–5137. [CrossRef]
10. Zhou, L.; Chen, K.; Dong, H.; Chi, S.; Chen, Z. An Improved Beetle Swarm Optimization Algorithm for the Intelligent Navigation Control of Autonomous Sailing Robots. *IEEE Access* **2021**, *9*, 5296–5311. [CrossRef]
11. Zhou, J.; Li, S.; Zhang, W.; Yan, W.; Zhao, Y.; Ji, Y.; Liu, X. Prediction and Suppression of Twisted-wire Pair Crosstalk Based on Beetle Swarm Optimization Algorithm. *Appl. Comput. Electrom.* **2021**, *36*, 435–441. [CrossRef]
12. Wang, L.; Wu, Q.; Lin, F.; Li, S.; Chen, D. A New Trajectory-Planning Beetle Swarm Optimization Algorithm for Trajectory Planning of Robot Manipulators. *IEEE Access* **2019**, *7*, 154331–154345. [CrossRef]
13. Al-Wesabi, F.N.; Alsolai, H.; Hilal, A.M.; Hamza, M.A.; Al Duhayyim, M.; Negm, N. Machine Learning Based Depression, Anxiety, and Stress Predictive Model during COVID-19 Crisis. *Cmc Comput. Mater. Contin.* **2022**, *70*, 5803–5820. [CrossRef]
14. Singh, P.; Kaur, A.; Batth, R.S.; Kaur, S.; Gianini, G. Multi-disease big data analysis using beetle swarm optimization and an adaptive neuro-fuzzy inference system. *Neural Comput. Appl.* **2021**, *33*, 10403–10414. [CrossRef]
15. Hasan, N.; Islam, M.S.; Chen, W.; Kabir, M.A.; Al-Ahmadi, S. Encryption Based Image Watermarking Algorithm in 2DWT-DCT Domains. *Sensors* **2021**, *21*, 5540. [CrossRef] [PubMed]
16. Rohan, A.; Raouf, I.; Kim, H.S. Rotate Vector (RV) Reducer Fault Detection and Diagnosis System: Towards Component Level Prognostics and Health Management (PHM). *Sensors* **2020**, *20*, 6845. [CrossRef] [PubMed]
17. Aileni, R.M.; Pasca, S.; Florescu, A. EEG-Brain Activity Monitoring and Predictive Analysis of Signals Using Artificial Neural Networks. *Sensors* **2020**, *20*, 3346. [CrossRef] [PubMed]
18. Chu, W.; Lin, C.; Kao, K. Fault Diagnosis of a Rotor and Ball-Bearing System Using DWT Integrated with SVM, GRNN, and Visual Dot Patterns. *Sensors* **2019**, *19*, 4806. [CrossRef]
19. Fadhil, A.F.; Kanneganti, R.; Gupta, L.; Eberle, H.; Vaidyanathan, R. Fusion of Enhanced and Synthetic Vision System Images for Runway and Horizon Detection. *Sensors* **2019**, *19*, 3802. [CrossRef]

20. Guo, W.; Xian, Y.; Zhang, D.; Li, B.; Ren, L. Hybrid IRBM-BPNN Approach for Error Parameter Estimation of SINS on Aircraft. *Sensors* **2019**, *19*, 3682. [CrossRef]
21. Song, Y.; Zhao, X.; Li, B.; Hu, Y.; Cui, X. Predicting Spatial Variations in Soil Nutrients with Hyperspectral Remote Sensing at Regional Scale. *Sensors* **2018**, *18*, 3086. [CrossRef]
22. Wang, Z.; Wang, D.; Chen, B.; Yu, L.; Qian, J.; Zi, B. A Clamping Force Estimation Method Based on a Joint Torque Disturbance Observer Using PSO-BPNN for Cable-Driven Surgical Robot End-Effectors. *Sensors* **2019**, *19*, 5291. [CrossRef] [PubMed]
23. Wang, J.; Li, M.; Jiang, W.; Huang, Y.; Lin, R. A Design of FPGA-Based Neural Network PID Controller for Motion Control System. *Sensors* **2022**, *22*, 889. [CrossRef] [PubMed]
24. Yao, Z.; Lei, Y.; He, D. Early Visual Detection of Wheat Stripe Rust Using Visible/Near-Infrared Hyperspectral Imaging. *Sensors* **2019**, *19*, 952. [CrossRef] [PubMed]
25. Clerc, M.; Kennedy, J. The particle swarm—Explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evolut. Comput.* **2002**, *6*, 58–73. [CrossRef]
26. Khan, A.T.; Cao, X.W.; Li, Z.; Li, S. Enhanced Beetle Antennae Search with Zeroing Neural Network for online solution of constrained optimization. *Neurocomputing* **2021**, *447*, 294–306. [CrossRef]
27. Zhang, Y.; Li, S.; Xu, B. Convergence analysis of beetle antennae search algorithm and its applications. *Soft Comput.* **2021**, *25*, 10595–10608. [CrossRef]
28. Zheng, Q.; Xiang, D.; Fang, J.; Wang, Y.; Zhang, H.; Hu, Z. Research on performance seeking control based on Beetle Antennae Search algorithm. *Meas. Control.* **2020**, *53*, 1440–1445. [CrossRef]
29. Jiang, X.; Li, S. Beetle Antennae Search without Parameter Tuning (BAS-WPT) for Multi-objective Optimization. *Filomat* **2020**, *34*, 5113–5119. [CrossRef]
30. Mu, Y.; Li, B.; An, D.; Wei, Y. Three-Dimensional Route Planning Based on the Beetle Swarm Optimization Algorithm. *IEEE Access* **2019**, *7*, 117804–117813. [CrossRef]