Scientific Research Publishing

Open Journal of Social Sciences, 2016, 4, 59-68
http://www.scirp.org/journal/jss
ISSN Online: 2327-5960
ISSN Print: 2327-5952

Application of Artificial Neuron Network in Analysis of Railway Delays

Jia Hu, Bernd Noche

Department of Transport and Logistics, University Duisburg-Essen, Duisburg, Germany
Email: jia.hu@uni-due.de

How to cite this paper: Hu, J. and Noche, B. (2016) Application of Artificial Neuron Network in Analysis of Railway Delays. *Open Journal of Social Sciences*, **4**, 59-68.
http://dx.doi.org/10.4236/jss.2016.411005

Received: September 27, 2016
Accepted: November 22, 2016
Published: November 29, 2016

Abstract

Punctuality is a key performance indicator of train freight transport. However, train delay arises often in the practice. To improve the efficiency of cargo train, prediction of train-delay is always an important research area. In this paper, a prediction model is established on the base of artificial neural network (ANN). Due the endogen drawback of ANN, Genetic Algorithm is adopted to improve the performance of ANN. Consequently, an experiment is design to train and test the ANN-based model by a set of data from the practice. The results of the experiment demonstrate the significant propagation ability of the model.

Keywords

Train-Delay Prediction, Artificial Neuron Network, Improvement of Artificial Neuron Network, Delay Reasons of Railway Freight

1. Introduction

Current economic development is highly dependent on natural resources, such as coal and oil. Punctual delivery of products is a key indicator of service performance in the production of natural resources. The production sites of natural resources are usually located far from densely populated places. In terms of logistics, transported products play an important role in the production of natural resources. However, railway transport declined constantly over the past decades [1]. This decline is attributed to diverse reasons, such as unstable weather conditions, complex infrastructure, and bureaucratic inefficiency [2]. Given the endogenous characteristics of railway transport, its scheduling suffers from the vulnerability of the transport process; thus, the delivery time of the cargo train is always out of control [3]. To improve the efficiency of rail traffic, delay prediction, which is a necessary function of railway operation, should be explored.

DOI: 10.4236/jss.2016.411005 November 29, 2016

Many researchers have investigated the train-delay problem and have presented numerous models and algorithms [4]. The majority of previous works on delay estimation for railway networks concentrates on network and operational parameters (e.g., available resources, capacity, and timing), as well as their complex interactions [5]. In this paper, artificial neural network (ANN) is applied to support stakeholders of railway operation in solving less-structured problems in daily business. This algorithm simplifies the process of decision-making by imitating human decision-making behavior.

This study is organized as follows. The literature review of delay models are introduced in Section 2. The theoretical background of the ANN model is described in Section 3. Result analysis is conducted in Section 4 to evaluate the performance of the ANN model. Section 5 closes the study with a conclusion and suggestions further development.

2. Literature Review

At the present time, the analysis of train delays are based on either analytical and simulation models. The major concern related to railway transport is complicated because of the endogenous property of multiple agents and tiers [6]. The algorithms used to solve a system of equations may not be ideal for simple and accurate delay propagation. The effects of the interactions cannot be effectively captured in an analytical delay model [7].

By contrast, simulation models can develop simple and accurate mathematical relationships that effectively reflect the stochastic feature of the interactions of numerous traffic parameters, as well as their influence on delays in railway networks [8] [9] [10]. Various simulation methods are also applied to anticipate train delays, such as linear model [11], Markov-chain model [12], Monte-Carlo simulation [13], and grey system theory [14].

In the current problem, the set of the feasible solutions is complex because of the number of constraints (i.e., a solution that respects the constraints is unlikely to be determined through a hit and miss approach). The accuracy of the simulation results is highly sensitive to this possibility. However, an appropriate value for train delay possibility is difficult to set. This task requires not only a mature theory for support but also adequate experience in practical application [8]. From the literature review, many challenging and long-standing problems are closely related to the delays posed. Some types of delays are difficult to anticipate, e.g. natural disaster, and thus applying a pattern to analyze, estimate, or predict is difficult.

Furthermore, railway network scheduling is featured as a discrete-event dynamic system; therefore, linear regression is generally unsuitable for defining a dependent variable in relation to delays in train transportation. Studies conducted with disaggregate data report that the relationship between delay events and delay time is often nonlinear, time-variable, and S-shaped [16]. Artificial Neural Network (ANN) has distinguished advantages over traditional tactical methodologies (e.g., regression analysis, logistic regression, etc.) because it provides solutions to highly complex functions for nonlinear variables [17].

In the next section, a delay propagation model is presented based on ANN. It focuses on risks that are most frequently met in a railway network. Furthermore, the next section develops specific testable hypotheses.

3. Methodology

This paper does not discuss the complete box of prediction tools but rather focus on one component—multilayer perception (MLP). MLP is a type of ANN that is capable of forecasting the influence of important delay factors. The background and fundamentals of MLP are introduced in the first subsection. Given the endogen disadvantages of MLP, genetic algorithm is proposed as a performance-improvement method for MLP in the second subsection separately.

3.1. Theoretical Fundamentals of MLP

MLP has been inspired by biological neural connections in the human brain. The architecture of an MLP is significantly more simplified than that of a biological brain system. An MLP normally consists of an input layer, (at least) an intermediate layer (*i.e.*, the hidden layer), and an output layer, which shows the performance output of the system.

Let x_r represent the input values in a three-layer perceptron. The output of the hidden layer y_s and that of the output layer y_d are respectively expressed as follows:

$$y_c = G_c\left\{\left[\sum_{r=1}^{R} w_r^c x_r\right] + b_c\right\}, ?r \in R \ c \in C \tag{1}$$

$$y_d = G_d\left\{\left[\sum_{c=1}^{C} w_c^d y_c\right] + b_d\right\}, ?d \in D \tag{2}$$

where w_r^c is the connection weights between the input and hidden layer, b_c is the bias of the hidden layer, G_c is the activation function of the hidden layer; w_c^d is the connection weights between the hidden and output layer, b_d is the bias of the output layer, and G_d is the activation function of the output layer.In the system, the hidden layer enhances the adaptive learning in MLP, which is the ability to learn how to do tasks on the basis of the data given for training or initial experience.

The training scheme is activated by back–forward spreading error signals in this paper, *i.e.* back-propagation algorithm (BP). The difference between the actual and tar-get outputs is also considered an error (*i.e.*, training error). In the training process, the connection weights are modified through consistent comparisons of the desired and calculated output. The structure of the MLP is also adjusted correspondingly in iterations until the error lies in the acceptable area.

Let (x_d, y_d) be a training pair in training data D. Then, for arbitrary hidden layer neuron, error E in the weight space can be interpreted as follows:

$$E = \frac{1}{2}(t_d - y_d)^2 = \frac{1}{2}\sum_{d \in D}(t_d - y_d)^2 \tag{3}$$

where t_d is the targeted (desired) output of the MLP, y_d is the actual (calculated) output of the MLP, d is the d^{th} input for the MLP, and D is the training data of the MLP. In this paper, the definition of error is adopted the most popular in practice. The errors of an MLP are half squared the differences between the desired and actual outputs. The pre-factor 1/2 is not necessary but leads to a compact result [16]. The gradual adjustment in the training finally obtains a good weight distribution [18].

3.2. Improvement of the Performance of MLP

MLP has typical limitations, e.g. local minimum and convergence unsteadiness in its training procedure [19]. Enormous effort was proposed to improve the performance of BP, e.g. variable learning-rate back-propagation and Levenberg-Marquardt BP [11] [18] [20]. GA is allowed to perform the search in a general, representation-independent manner by emulating biological process. A hybrid GA-MLP model was proposed to decrease the possibility of trapping into local minima by combining the advantages of global search into the solution space of BP algorithm.

In selecting the operation in the genetic algorithm, the large probability must be chosen as the parent individual. This parent individual intersects with other individuals or possibly mutates according the probability of crossover and the probability of mutation that is initiated at the beginning of the genetic algorithm to breed a brand new group of individuals for next generation. When the maximal iteration number for the generation is reached, the training process for the network must be terminated, and the established MLP is considered as a well-trained network [21]. Given an initial set of weights, the connection weights of the network are updated through global search instead of through the traditional gradient descent. Thus, the quality of output of MLP is improved [22].

To obtain a problem with a reasonable combinatorial aspect, the delay variables selected for optimizing are the connection weights between the input and hidden layers in MLP. In the next section, an experiment is designed to apply MLP in delay propagation.

4. Experiment Design and Results Analysis

In this section, an experiment is presented that demonstrates how MLP makes an efficient prediction of delays in railway transport.

4.1. Research Methodology

4.1.1. Prerequisites of the Experiment

To establish a delay propagation model, train delays should be quantified as accurately as possible and communicated to related users to minimize the influence of disturbances. In this paper, the different kinds of delay are disregarded because MLP focuses on the outputs (the influence of reasons) and their classification rather than on the reasons themselves. In the following chapter, an experiment is presented that demonstrates how ANN supports decision-making.

To enhance the effectiveness of solution mechanisms, several specified operational constraints are considered. In this paper, the railroad network is a hub-and-spoke network where

- each rail station is a potential hub with consolidation capability that aims to minimize total cost, and the locations of facilities are fixed;
- train speed is dependent on the traffic situation, and maximum speed is limited;
- the headways of two trains are adequate for the schedule; and
- multiple periods are considered.

The delay in pre- and post-haulage (trucks) is neglected in this study because customers presumably deliver and pick up their commodities through terminal-to-terminal transportation.

4.1.2. Configuration of the BPNN Model in MATLAB®

In the experiment, the delay-prediction model is realized by program MATLAB®. In this section, MLP is therefore labelled as a BPNN to maintain its accordance with MATLAB®. Parameters are set to fix the topology of BPNN in MATLAB®, including activation function.

The activation function affects the signal transformation from the current layer to the next layer. Various activation functions can be applied between layers to map variables in MATLAB®. For instance, the log-sigmoid function is used to activate the connection between the input and the hidden layers; the linear function is employed to connect the hidden and the output layers. The relation of BPNN is expressed as follows [14]:

$$y_d = \sum_{s \in S} w_c^d \cdot G_c [\sum_{r \in R} w_r^c \cdot x_r + b_c] + b_d \tag{4}$$

where:

- *r*: node in the input layer.
- *c*:node in the hidden layer.
- *d*: node in the output layer.
- w_r^c : connection weights of the input and hidden layer.
- w_c^d : connection weights of hidden and output layer.
- b_c : bias of the input layer.
- b_d : bias of the output layer.

GA initially calculates the fitness value. This algorithm is then reiterated ten times in the experiment to optimize the initial value of the weight and the bias for the network [15]. The fitness function calculates the sum of the errors between the outputs and the targets.

$$fitness = \sum_{d \in D} \left| y_d - t_d \right| \tag{5}$$

A new population is generated through genetic operation crossover and mutation. This population is fed into the BPNN model for qualified prediction. Superior individuals are maintained in the succeeding generations, and inferior ones are eliminated.

Figure 1 depicts the flowchart of this model.

The BPNN model framework is now established. In the next step, the inputs are integrated into the model for training and testing. The details of the process are discussed in the following subsection.

4.2. Experiment Design on the Delay-Prediction Model

The methodology consists of three steps: first, the inputs (relevant variables) are identified and quantified. These inputs determine the configuration of the freight network, infrastructure, locations of facilities and depots, and order timing. Second, these variables are simulated in the model. Meanwhile, the model is configured. Finally, the results of the simulation are analyzed. Delay samples are collected from data sets from Romania. The source is used to train and test the prediction model.

In the experiment, the samples deployed in the model principally consist of two subsets, namely, training and test samples. Sample size is defined as the number of samples in one subset. The training subset data are used to recognize and analyze the potential structure of the connection weights by gradient descent in training phases. Meanwhile, the samples in the test subset are not used during training but are employed to verify these weights.

Once the network begins to over-fit, the training set and the error on the validation set occasionally begin to increase. Once the validation error increases for a specified number of iterations, the training ceases and the weights at minimal validation error are returned.

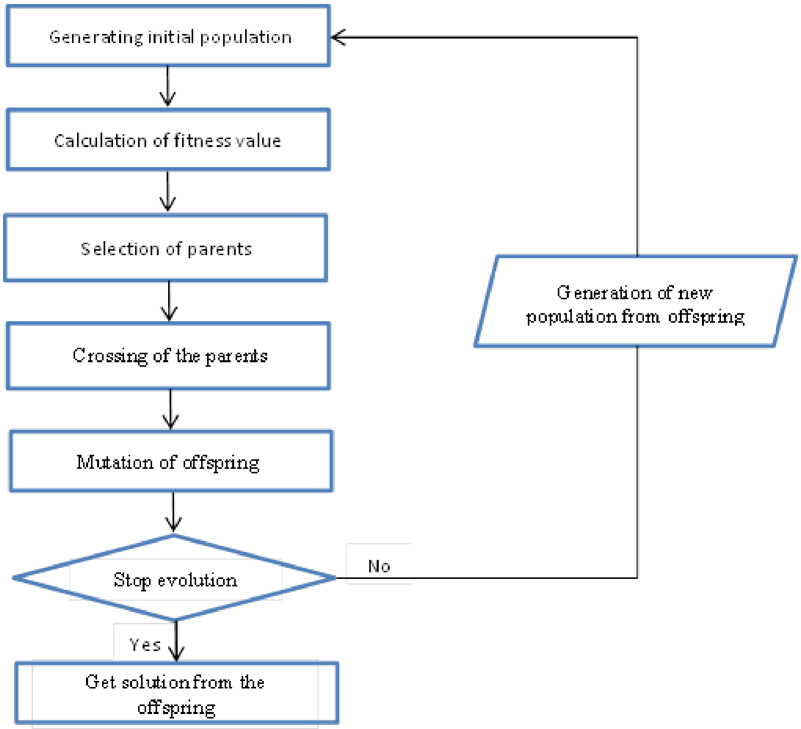
Generating initial population
Calculation of fitness value
Selection of parents
Crossing of the parents
Mutation of offspring
Stop evolution
No
Yes
Get solution from the offspring
Generation of new population from offspring

Figure 1. Flowchart of GA.

Principally, several experiments are conducted without GA to test the performance of different parameters as provided by MATLAB® Neural Network Toolbox™. The GA-BPNN model is principally designed in the following steps:

- The Train ID. No., Traction, Weekday, Delay code, Station, and Region of the training set are the input elements, and the Delay time is the target.
- The network is trained with different modification methods.
- The network is validated with the test data.

Correspondingly, the pseudo code of the GA-BPNN model is written in **Table 1**.

Once the input and output data are loaded and the training parameters are set up, the process of the model training begins. The results are explained in the following section.

4.3. Results Analysis

During evaluation, a neural network is established in MATLAB®, and 20 epochs are considered for each run. The number of created neural networks is similar to the size of the population. Each chromosome in a generation uses the fitness function to evaluate and then choose the selection option. Furthermore, the initial part is incorporated into the GA. The number of neural networks created is equal to the size of the population multiple (number generated + 1).

The case comparisons focus on the four key performance indexes (KPI): Mean error on test set; standard deviation (SD) of the errors on test sets; mean error on training sets and SD of the error on training sets. **Table 2** highlights the best overall results from all of the simulation experiments. All of these tests applied the GA-BPNN model. Thus, GA optimization can be sensibly incorporated into simulations although the run time is long.

Implementing GA in BPNN improves prediction performance. Specifically, disadvantages are overcome, including too-rapid convergence. The use of a GA-BPNN model is preferred over the direct application of a classical ANN procedure in the modelling of train delays for response time reasons. Therefore, the GA-based BPNN is an effective simulation method with which to estimate delays.

Table 1. Pseudo code of the GA-BPNN model.

```
Initialize network weights
  do
     For each training sample ex
     Desired Output = neural-net-output (network, ex)
    actual = calculated output(ex)
    compute error (difference between desired and actual output) at the output units
        compute  $\Delta w_c^d$  for all weights from the hidden to the output layer
        compute  $\Delta w_r^c$  for all weights from the input to the hidden layer
   update network weights
   until all samples are classified correctly or another stopping criterion is satisfied
  return the network
```

Table 2. Best values on each KPI.

Key Performance Index	Best Value
Mean error on test sets	3.47
SD of the errors on test sets	7.15
Mean error on training sets	4.07
SD of the errors on training sets	9.98

4.4. Interim Conclusion

A well-established BPNN is tested in MATLAB® to predict train delay at the operational level. The GA-based BPNN is demonstrated as a well-performance prediction model in comparison with other parameters of performance improvement which are mentioned in the dissertation. Furthermore, the results of the GA-BPNN case imply two distinguish drawbacks of the GA approach namely, excessive computational time and high computational labor. Without GA optimization, the run time is only a few minutes long. The run time of the GA-BPNN is usually 3 hours to 10 hours. The reason for this long run time is because the GA performs a global search of the solution space.

5. Conclusions

Train-delay prediction is a complex problem in the real world, and future trends are based on a huge amount of historical data of observed organization. To provide a convenient tool for the simulation of the biological decision system, a BPNN is designed to describe the main aspect of the biological neural network while ignoring the aspects that are insignificant to the simulation. This approach has been one of the cornerstones for train delays. Given the endogen disadvantages of BPNN, GA has been implemented as improvement because GA is well suited to the quick global exploration of a large search space to optimize an objective function and to determine the possible solutions of "good quality."

Although the model can provide efficient solutions, interactions of several risk factors are complex and require further studies in their specific context. The results of such tool provide are usually "good enough" given the complex nature of the conflict-resolution problem. Therefore, further investigation should be conducted to examine possible measures that may have been overlooked and to find ways of improving and optimizing the results of BPNN. Any specific circumstances may also be left unnoticed in the BPNN model in the data extracted for training and testing. Thus, these issues deserve further investigation.

Acknowledgements

We thank Mr. Horatiu Ionescu, head of Operation Department of CFR. He provided the data for the experiment presented in the paper.

References

[1] European Commission Eurostat (2015) Energy, Transport and Environment Indicators—

2015 Edition. European Commission Eurostat, Luxemburg.

[2] Finger, M. (2014) Governance of Competition and Performance in European Railways: An Analysis of Five Cases. *Utilities Policy*, **31**, 278-288. http://dx.doi.org/10.1016/j.jup.2014.03.003

[3] Dollevoet, T., Corman, F., D'Ariano, A. and Huisman, D. (2014) An Iterative Optimization Framework for Delay Management and Train Scheduling. *Flexible Services and Manufacturing Journal*, **26**, 490-515. http://dx.doi.org/10.1007/s10696-013-9187-2

[4] Caris, A., Macharis, C. and Janssensa, G.K. (2013) Decision Support in Intermodal Transport: A New Research Agenda. *Computers in Industry*, **64**, 105-112.

[5] Michiels, W. and Niculescu, S.-I. (2014) Stability, Control, and Computation for Time-Delay Systems: An Eigenvalue-Based Approach. 2nd Edition, SIAM-Society for Industrial and Applied Mathematics, Philadelphia.

[6] Srinivasan, M., Mukherjee, D. and Gaur, A.S. (2011) Buyer-Supplier Partnership Quality and Supply Chain Performance: Moderating Role of Risks, and Environmental Uncertainty. *European of Management Journal*, **29**, 260-271. http://dx.doi.org/10.1016/j.emj.2011.02.004

[7] Murali, P., Dessouky, M., Ordó, F. and Palmer, K. (2010) A Delay Estimation Technique for Single and Double-Track Railroads. *Transportation Research Part E*, **46**, 483-495. A Delay Estimation Technique for Single and Double-Track Railroads. http://dx.doi.org/10.1016/j.tre.2009.04.016

[8] Rodrigues, V.S., Stantchev, D., Potter, A., Naim, M. and Whiteing, A. (2008) Establishing a Transport Operation Focused uncertainty Model for the Supply Chain. *International Journal of Physical Distribution & Logistics Management*, **38**, 388-411. http://dx.doi.org/10.1108/09600030810882807

[9] Reis, V., Fabian Meier, J., Pace, G. and Palacin, R. (2013) Rail and Multi-Modal Transport. *Research in Transportation Economics*, **41**, 17-30. http://dx.doi.org/10.1016/j.retrec.2012.10.005

[10] Ivanov, D. and Sokolov, B. (2010) Adaptive Supply Chain Management. Springer Verlag, London. http://dx.doi.org/10.1007/978-1-84882-952-7

[11] Goverde, R.M. (2010) A Delay Propagation Algorithm for Large-Scale Railway Traffic Networks. *Transportation Research Part C*, **18**, 269-287. http://dx.doi.org/10.1016/j.trc.2010.01.002

[12] Häme, L. and Hakula, H. (2013) Dynamic Journeying under Uncertainty. *European Journal of Operational Research*, **225**, 455-471. http://dx.doi.org/10.1016/j.ejor.2012.10.027

[13] Barta, J., Rizzoli, A.E., Salani, M. and Gambardella, L.M. (2012) Statistical Modelling of Delays in a Rail Freight Transportation Network. *Proceedings of the* 2012 *Winter Simulation Conference*, Switzerland. http://dx.doi.org/10.1109/wsc.2012.6465188

[14] Kayacan, E., Ulutas, B. and Kaynak, O. (2010) Grey System Theory-Based Models in Time Series Prediction. *Expert Systems with Applications*, **37**, 1784-1789. http://dx.doi.org/10.1016/j.eswa.2009.07.064

[15] Hasan, M.K. (2010) A Framework for Intelligent Decision Support System for Traffic Congestion Management System. *Scientific Research*, **2**, 270-289. http://dx.doi.org/10.4236/eng.2010.24037

[16] Ludvigsen, J. and Klaebe, R. (2014) Extreme Weather Impacts in Freight Railways in Europe. *Natural Hazards*, **70**, 767-787.

[17] Graupe, D. (2013) Principles of Artificial Neural Networks, 3rd Ed, *Advanced Series in Circuits and Systems*, World Scientific Publishing Co. Pte. Ltd., Singapore. http://dx.doi.org/10.1142/8868

[18] Senties, O.B., Azzaro-Pantel, C., Pibouleau, L. and Domenech, S. (2009) A Neural Network and a Genetic Algorithm for Multiobjective Scheduling of Semiconductor Manufacturing Plant. *Industrial & Engineering Chemistry Research*, **48**, 9546-9555. http://dx.doi.org/10.1021/ie8018577

[19] Haykin, S. (2009) Neural Networks and Learning Machines, International Edition ed., Pearson Education, Inc., New Jersey.

[20] Hagan, M.T. and Demuth, H.B. (2014) Neural Network Design. 2. Ed., PWS Publishing Co., Boston.

[21] Roy Chatterjee, S., Mandal, R. and Chakraborty, M. (2013) A Comparative Analysis of Several Back Propagation Algorithms in Wireless Channel for ANN-Based Mobile Radio Signal Detector. *International Journal of Science and Modern Engineering*, **1**.

[22] Ghoddousi, P., Eshtehardian, E., Jooybanpour, S. and Javanmardi, A. (2013) When the Maximal Iteration Number for the Generation Is Reached, the Training Process for the Network Must Be Terminated, and the Established MLP Is Considered as a Well-Trained Network. *Automation in Construction*, **30**, 216-227. http://dx.doi.org/10.1016/j.autcon.2012.11.014