Analysis of neural network responses in calibration of microsimulation traffic model

Ištoka Otković, Irena; Varevac, Damir; Šraml, Matjaž

GRAĐEVINSKI I ARHITEKTONSKI FAKULTET OSIJEK
Faculty of Civil Engineering and Architecture Osijek

Repository / Repozitorij:

Repository GrAFOS - Repository of Faculty of Civil Engineering and Architecture Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

ANALYSIS OF NEURAL NETWORK RESPONSES IN CALIBRATION OF MICROSIMULATION TRAFFIC MODEL

Irena Ištoka Otković
Josip Juraj Strossmayer University of Osijek, Faculty of Civil Engineering Osijek, Assistant Professor
Corresponding author: iirena@gfos.hr
Damir Varevac
Josip Juraj Strossmayer University of Osijek, Faculty of Civil Engineering Osijek, Associate Professor
Matjaž Šraml
University of Maribor, Faculty of Civil Engineering Maribor, Associate Professor

Abstract: Microsimulation models are frequently used in traffic analysis. Various optimization methods are used in calibration, and the one method that has shown success is neural networks. This paper shows the responses of neural networks during calibration of a microsimulation traffic model. We analyzed two calibration methods by applying neural networks and comparing their neural network learning (according to their achieved correlation and the mean error of prediction) and their generalization ability (comparison of generalization results was analyzed in two steps). The best correlation between the microsimulation results and neural network prediction was 88.3%, achieved for the traveling time prediction, on which the first calibration method is based.

Keywords: microsimulation traffic models, calibration, response of neural networks, traveling time, queue parameters

ANALIZA ODZIVA NEURONSKIH MREŽA U POSTUPKU KALIBRACIJE MIKROSIMULACIJSKOG PROMETNOG MODELA

Sažetak: Učestala je primjena mikrosimulacijskih modela u prometnim analizama, a realnost dobivenih rezultata simulacije u funkciji je uspješnosti postupka kalibracije. Različite metode optimiranja primjenjuju se u postupku kalibracije, a jedna od metoda koja se pokazala uspješnom u postupku kalibracije je metoda koja primjenjuje neuralne mreže. U ovom radu prikazana je analiza odziva neuralnih mreža u kalibraciji mikrosimulacijskog prometnog modela. U radu su razmatrane dvije metode kalibracije primjenom neuralnih mreža koje su uspoređene prema rezultatima učenja neuralnih mreža (prema kriterijima postignute korelacije i srednje pogreške predikcije) i prema sposobnosti generalizacije (usporedba rezultata generalizacije analizirana je u dva koraka). Najbolja korelacija između rezultata mikrosimulacije i predikcije neuralne mreže (88,3%) dobivena je za parametar vrijeme putovanja, a na navedenom parametru bazirana je prva metoda kalibracije.

Ključne riječi: mikrosimulacijski prometni model, kalibracija, odziv neuronske mreže, vrijeme putovanja, parametri kolone vozila

1   INTRODUCTION

Microsimulation models are very useful for analyzing existing critical traffic network segments and for predicting traffic conditions on both existing and planned traffic network elements and intersections. The applicability of a traffic model in local conditions depends on the efficiency of calibration. Calibration of a traffic model is actually an adjustment of the model to a local traffic network and its users, which minimizes the difference between the modelling results and measured data.

A commonly used algorithm to calibrate input parameters is the genetic algorithm (GA). Many positive experiences have been reported in using GA in the FRESIM calibration model [1], in the PARAMICS calibration model [2], in the calibration of the VISSIM and CORSIM models [2], and in the VISSIM calibration model [3]. The analysis of acceptable time gaps and defining critical time interval by Greenshields' model [4] described the VISSIM microsimulation model calibration for some New York roundabouts. Although some calibration methods are more (GA) or less (trial and error) successful for a variety of microsimulation models, the possibility of applying new calibration methods has been explored. We have previously applied neural networks in calibrating a microsimulation traffic model [5]. In the present paper, we analyzed the response of neural networks in calibrating a microsimulation model.

2   ANALYSIS OF NEURAL NETWORK RESPONSE

It would be very time-consuming to test all combinations of model input parameters by applying realistic VISSIM simulations, using a separate run for every combination. A computer can examine many combinations of model input parameters in real time, in case that it can obtain software simulation output values of the microsimulation traffic model (VISSIM). The neural network predicts the values of the VISSIM simulations during software examination of the combinations of input parameters. Figure 1 shows a simplified schematic of program calibration.
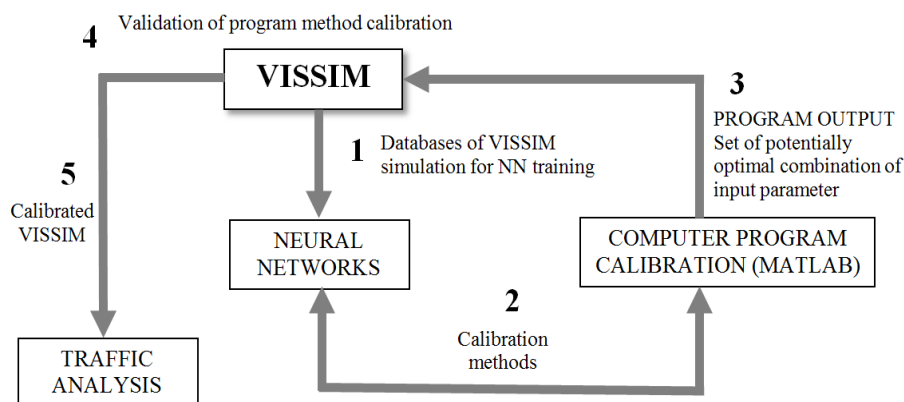

Figure 1 Schematic of computer program calibration [5]

2.1.   Databases for neural network training

We analyzed the response of neural networks for two calibration methods. Method I uses the travel time for calibration and queue parameters for model validation, while Method II uses both observed output parameters for calibration. The calibration methods are described in detail in [6]. VISSIM supports an increasing number of traffic parameters for analysis, from operational features and economic indicators to environmental impact parameters (e.g., noise and air pollution). When selecting the operational characteristics to use as output simulation values of the observed model, our main criterion was measurability in the field.

To predict the simulation values of observed traffic parameters using a neural network, it is necessary to create a database of VISSIM simulations from which the neural network will learn (train). Method I used Database I, which contains input data of various input parameters [5], and whose output indicator is traveling time between

measurement points. Method II used Database II, which has three output indicators: the traveling time between measurement points, the maximum queue, and the number of vehicle stopping at the intersection entrance.

Each database had 1379 variations of seven input parameters; the selection of the input parameters is detailed in [7]. Using a large enough database can partially prevent network overtraining[1] and makes it more likely to achieve satisfactory generalization [8]. To validate the network generalization, we used a database of 1379 examples of output VISSIM simulations to create a test dataset of 276 data (20% of the database size). The neural networks were compared using two basic criteria: training and the ability to generalize. Training is the success of prediction on a training data set (from which the network learned), and generalization is the success of prediction on an unknown data set.

We used the NeuroShell 2 software package to train the neural network and analyze the predictions. The following basic neural networks were tested: Ward nets, standard nets, jump connection nets, Jordan–Elman nets, and general regression net. These architectures include a group of backpropagation neural networks, including the standard type of backpropagation network in which every layer is connected or linked only to the previous layer (standard nets) or to every previous layer (jump connection nets). The Ward nets backpropagation network can detect various features in the data by using multiple slabs of neurons in the hidden layer, each with a different activation function. The general regression neural network (GRNN) is a one-pass learning algorithm with a highly parallel structure. Even with sparse data in a multidimensional measurement space, the algorithm can smoothly transition from one observed value to another. The GRNN is especially useful for approximating continuous functions. Its iterative algorithm adjusts during the training phase of the neural network within the selected set of training patterns. When the error between the network output and the desired output is minimized, the network can be used in a testing phase with test patterns. The GRNN of adaptive type uses a genetic algorithm to optimize the network structure.

The names of the researched networks were coded to make them easy to survey. The numbers indicate the number of neurons in hidden layers as well as the number of hidden layers; for example, 13-24-13 is a neural network with three hidden layers that has 13 neurons in the first and the third hidden layers and 24 neurons in the second hidden layer. The name of a model includes the activation functions of the hidden layers in situations where they differ from the default activation functions.

The best prediction on the test data set is the one that is the most often memorized, but there is also the possibility to memorize the best prediction on the training data set, which is given the "TR" mark. The general regression neural network (GRNN), adaptive (AD), and iterative (IT) network types were analyzed[2]. NeuroShell 2 saves the network at the point that gives the most accurate answers for patterns outside the training set. For all neural network architectures, we stopped the training manually once it passed many epochs from the minimum error on the test data set. We used the coefficient of network learning speed, or step iteration (η), in the process of correcting the weight coefficient. Empirically, when the selected learning speed coefficient is great, it can destabilize the network learning process, and when a network is small, it converges slowly. The impact of the coefficient of network learning speed, step iteration, and random weight initialization on neural network predictions should be examined in further research.

2.2 Indicators of neural networks learning

We chose the following indicators to compare our trained networks: the mean absolute error and the correlation coefficient between the results of neural network prediction and the results of VISSIM simulations.

2.2.1 Database I

Figures 2 and 3 compare the neural networks from Database I that gave the best predictions, according to the chosen criteria.

---

[1] One cause of overtraining may be the ratio of the number of input parameters to the dataset size used for network training. If there are many input parameters, and few examples in the learning set, the network will memorize rather than learn, causing it to generalize on the training dataset. However, this ratio is not the only cause of poor network generalization.

[2] For example: "SN 38" - standard nets, 1 hidden layer, number of neurons in hidden layer - 38, other network characteristics were unchanged (default); "GR3500-tanh-IT" - general regression type networks, one hidden layer with 3500 neurons, modified default activation function in hyperbolic tangent function, an iterative type of network.
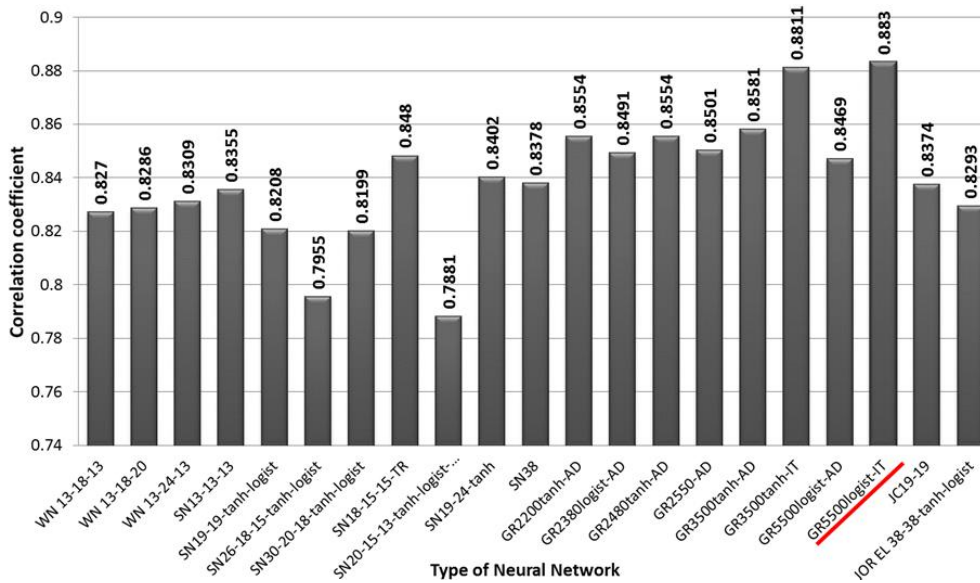
Analysis of neural network responses in calibration of microsimulation traffic model

Figure 2 Correlation coefficient of trained neural networks – the best results from Database I

The best correlation came from the regression neural network of iterative type. Considering that traffic flow modeling is a stochastic problem and that a basic VISSIM model uses a random number generator, the achieved correlation of 88.3% is very satisfactory. The minimum average error (0.26 s) came from the neural network of standard type, as shown in Figure 3.
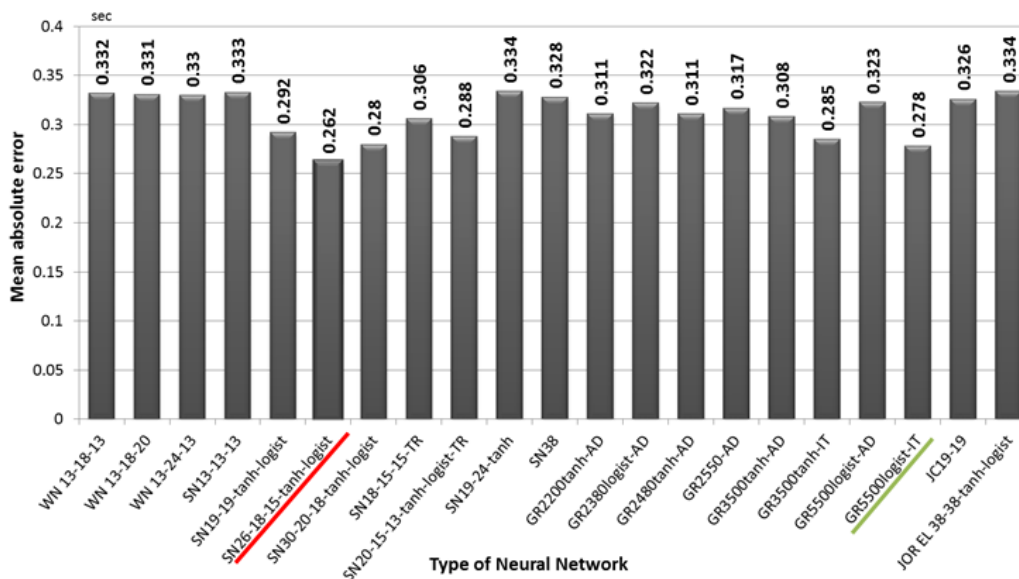
Figure 3 Mean absolute error of trained neural networks – the best results from Database I

2.2.2 Database II

We tested another calibration method, which is based on the ability of a neural network to predict multiple output results. This method simultaneously calibrates the traveling time and queue parameters: the maximum queue and the number of stopping at the entrance. Figure 4 compares the correlations, and Figure 5 compares the mean absolute error achieved for the best-trained networks of Database II.

The results of neural network learning for Database II (with three output indicators) were not as good as we expected. It gave a satisfactory correlation between the predicted traveling time and VISSIM results of more than 83%, but the queue parameters—the maximum queue and the number of stopping at the entrance—achieved best correlations of 67% and 57%, respectively.
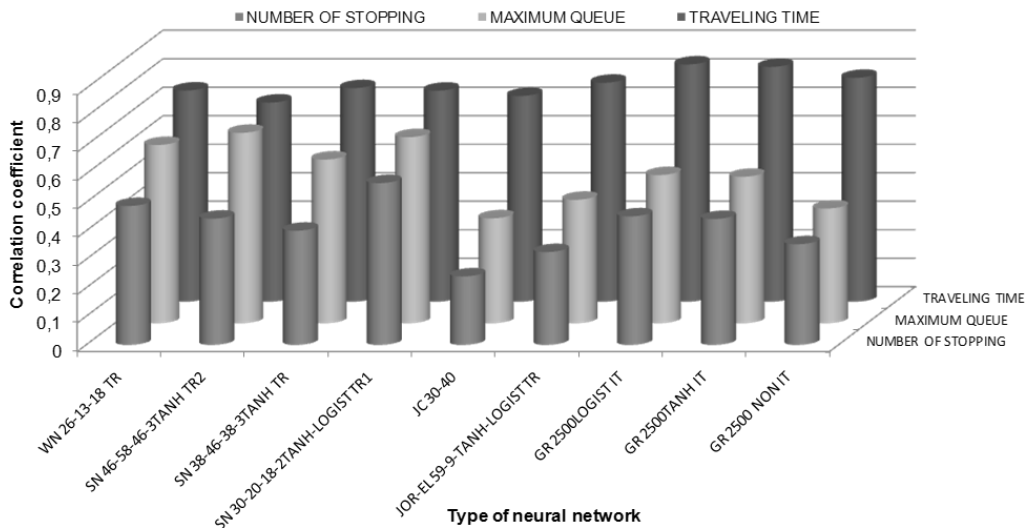

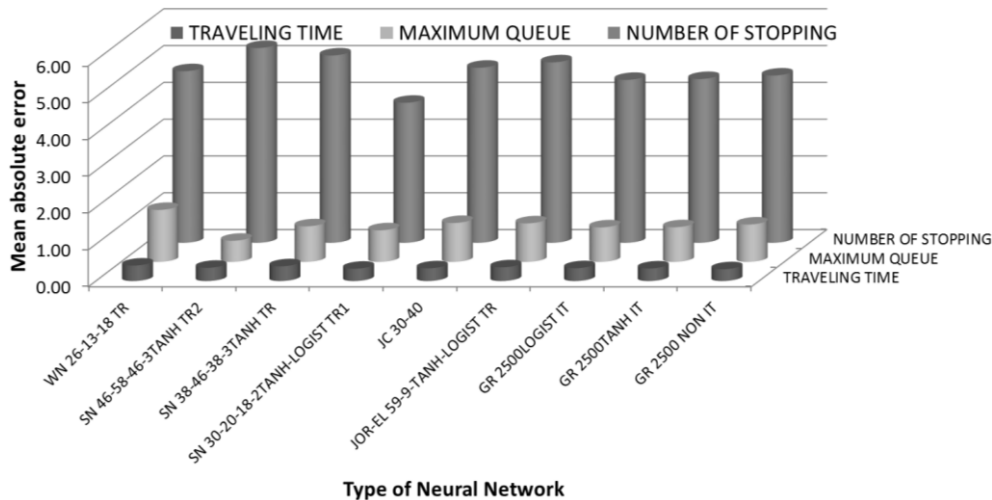**Figure 4 Correlation coefficient for neural networks that gave the best predictions**

Figure 5 Mean absolute error for neural networks that gave the best predictions

2.3 Generalization

The objective of neural network learning is to achieve generalization ability: in our case, the ability to predict the output traffic parameters from combinations of input parameters of a given accuracy. To compare the predictions of trained networks, we made test predictions for a particular calibration problem. The test prediction was performed for 25 combinations of input parameter values: 10 combinations of known input parameters (assessment of network's training level and its architecture) and 15 combinations of new input parameters (assessment of generalization capability). In this way, we could use the same diagram to evaluate every observed network by both its training level and generalization ability.

Test prediction was done in Excel by using function prediction, which connected the output function of every learned neural network (the output data file of NeuroShell2) and the input data specified in the Excel table.

Analysis of neural network responses in calibration of microsimulation traffic model

2.3.1 First validation

Figure 6 shows the test predictions for the adaptive and iterative types of general regression neural networks for Database I. The general regression neural network of iterative type gave the best correlation with the VISSIM simulated results, as shown in Figure 2. The test predictions showed that the network has good generalization ability (Figure 6); that is, its predictions correspond relatively well with the expected output parameters for combinations of unknown input data (the first 15). In contrast, the predictions from other types of neural networks differed greatly for combinations of unknown input data [6].

The best predictions from the validation data set were obtained by General regression neural networks of iterative type (Figures 7–9). For Database II, the neural networks that gave the best results were GR2500-logist IT for travel time between measurement points and for maximal queue and GR5500 IT for the number of stopping at the entrance. The GR2500-logist IT neural network gave the best overall results in test validation by two traffic indicators and also provided the best correlation of travel time from the training data set. These good results from the training data set demonstrate that the GR2500-logist IT network has an optimal level of network training, and its good results from the validation data set demonstrate its generalization capability.
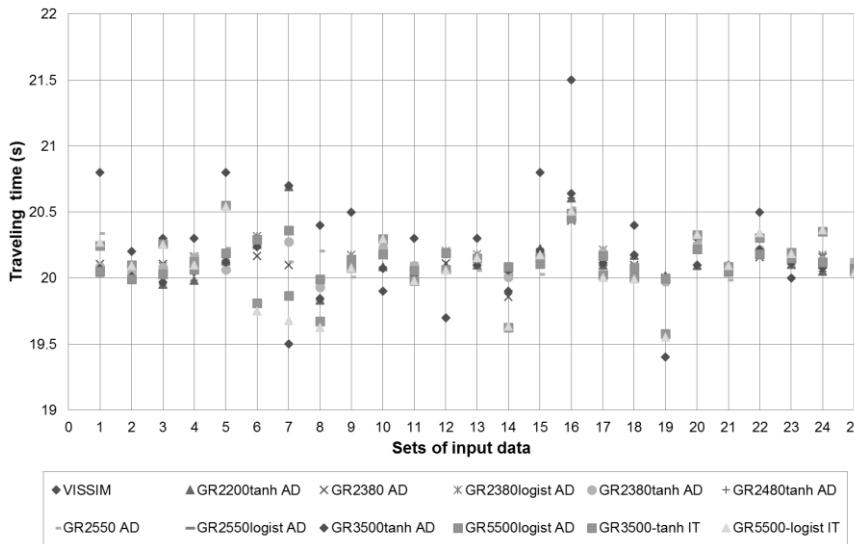

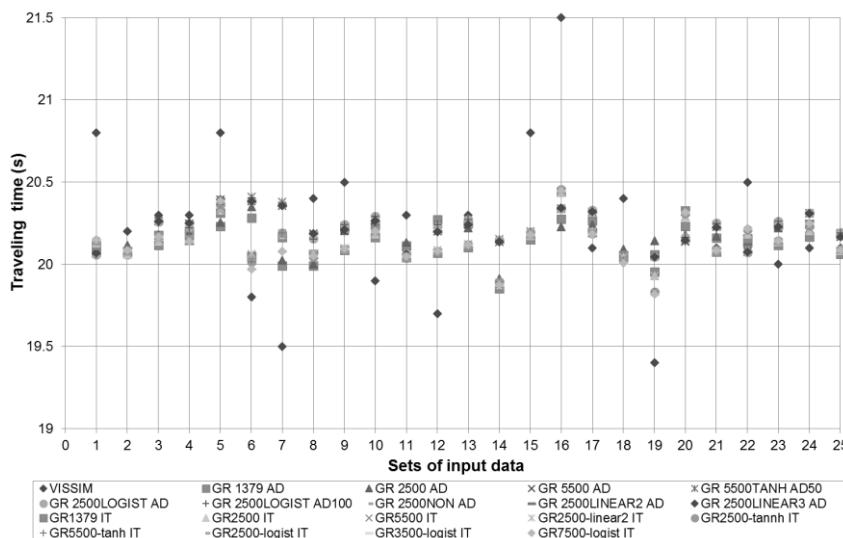**Figure 6 Test predictions for adaptive and iterative types of general regression networks – Database I**

Figure 7 Test predictions for general regression neural networks – traveling time – Database II

Analysis of neural network responses in calibration of microsimulation traffic model
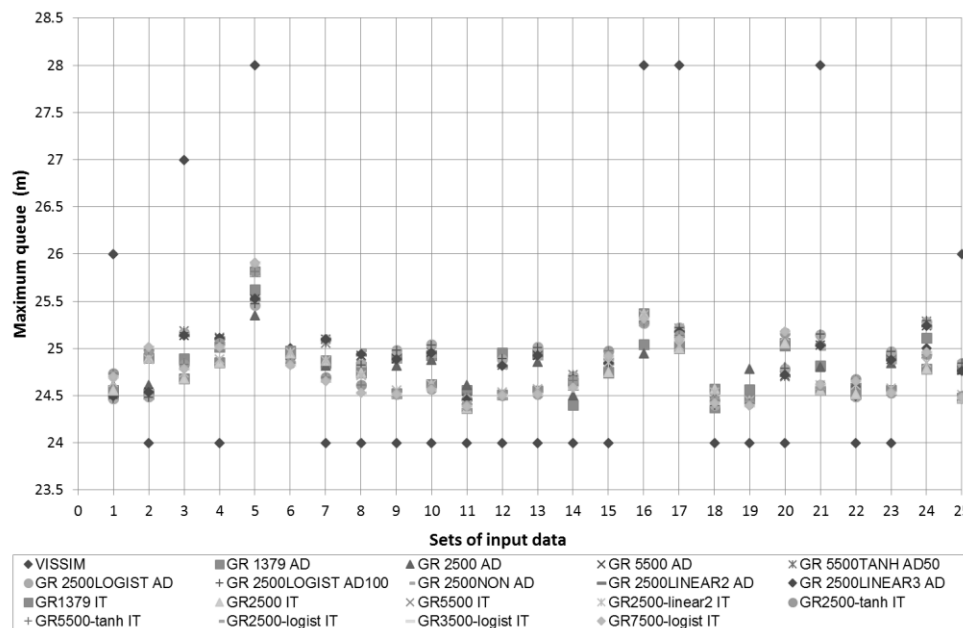

Figure 8 Test predictions for general regression neural networks – maximum queue – Database II
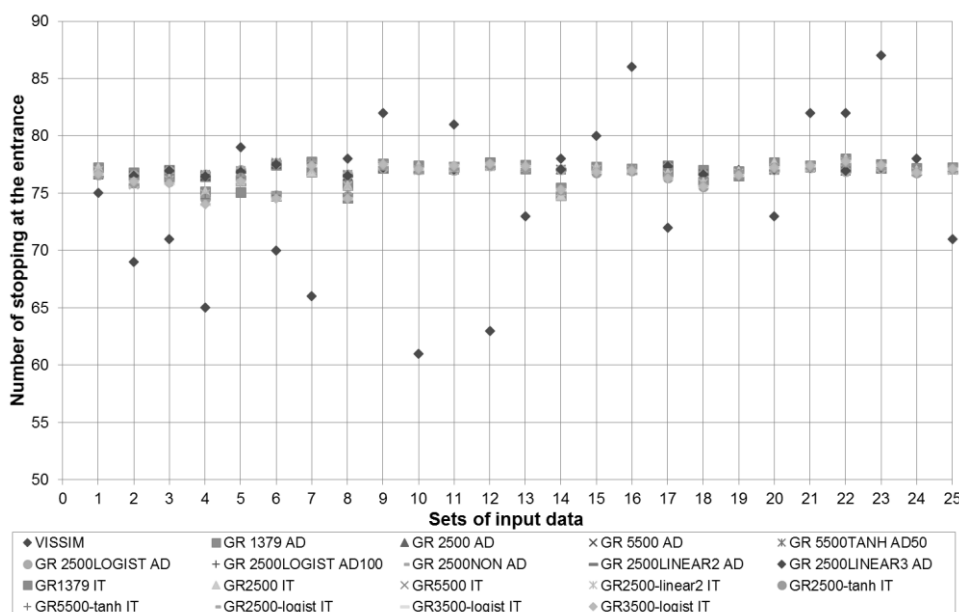

Figure 9 Test predictions for general regression neural networks – number of stopping - Database II

2.3.2 Second validation

The three neural networks with the best predictions from each database entered the final validation process, a prediction analysis using 64 new combinations of input parameters. The data set for final evaluation was used to evaluate generalization capability of the trained networks. We calculated the prediction error as the absolute difference between predictions obtained by the neural network and expected values obtained by the VISSIM simulation. The selection criterion of a neural network to be used in calibration is the minimum mean absolute error for the validation data set. Figure 10 shows the predictions for the validation data set for the three networks that gave the best results in the test procedure, along with the expected values from the VISSIM simulations. From

these results we selected the GR5500-logist IT network, trained to predict travel time, for the calibration. This network gave the best correlation (88.3%) on the training data set and showed the best generalization capability. It also had the lowest mean absolute prediction error for both the test and final validation data sets.

Figures 11–13 show the validation results of the best neural networks from Database II. The final selected network, judged by the criterion of minimal mean absolute error, was the GR2500-logist IT neural network. That network had the lowest mean absolute error for the two of the three traffic parameters: travel time and maximum queue.
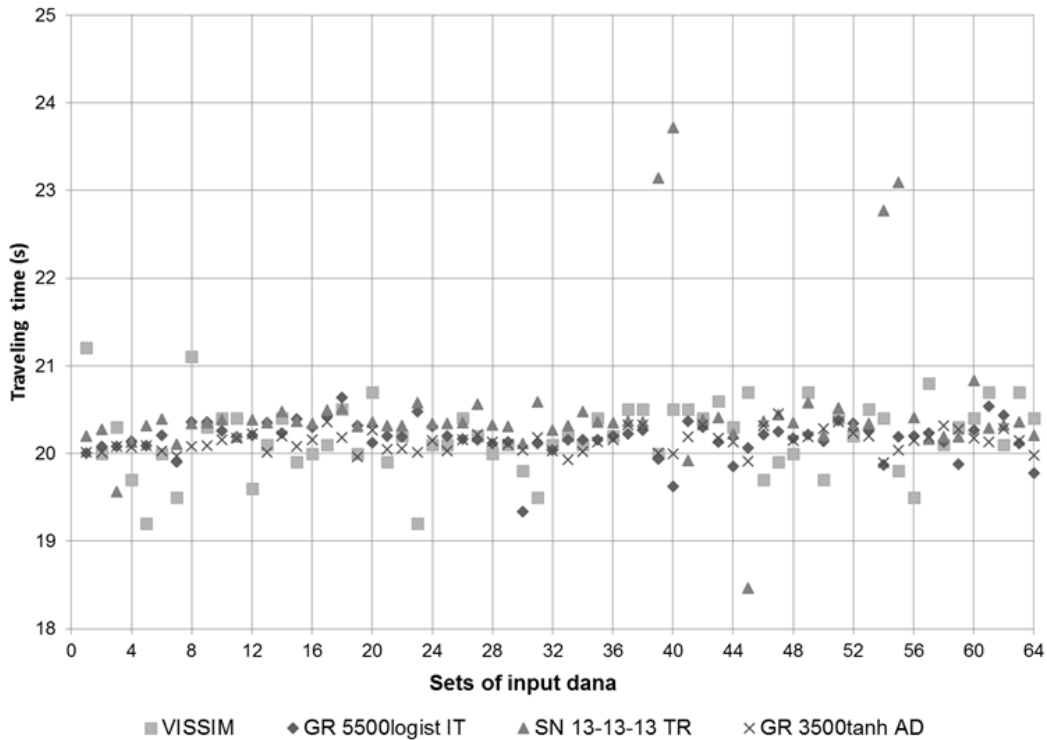

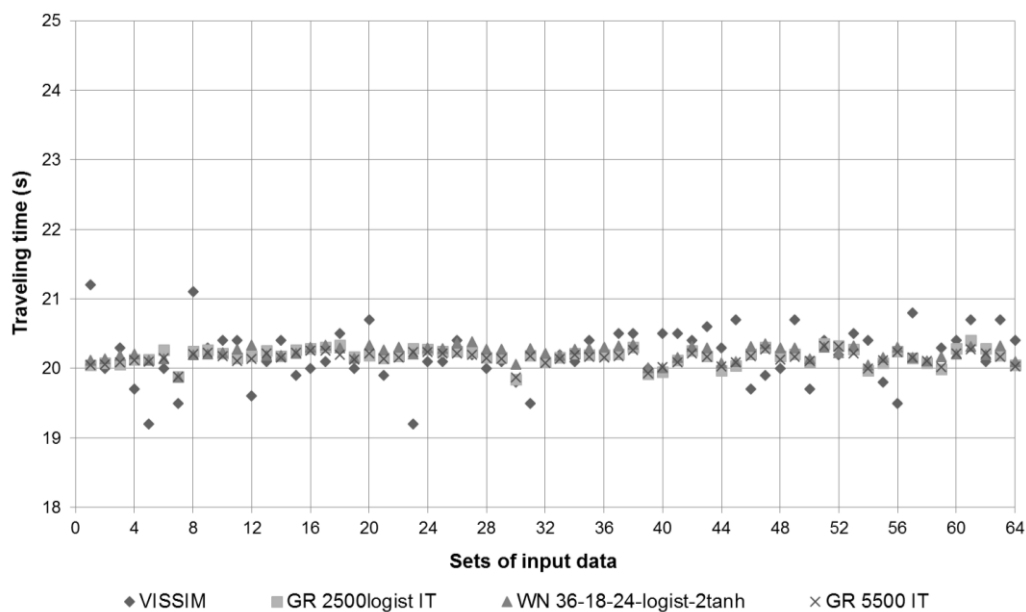**Figure 10 Predictions of the best neural networks on the validation set – Database I**

Figure 11 Predictions of traveling time of best networks on validation set – Database II
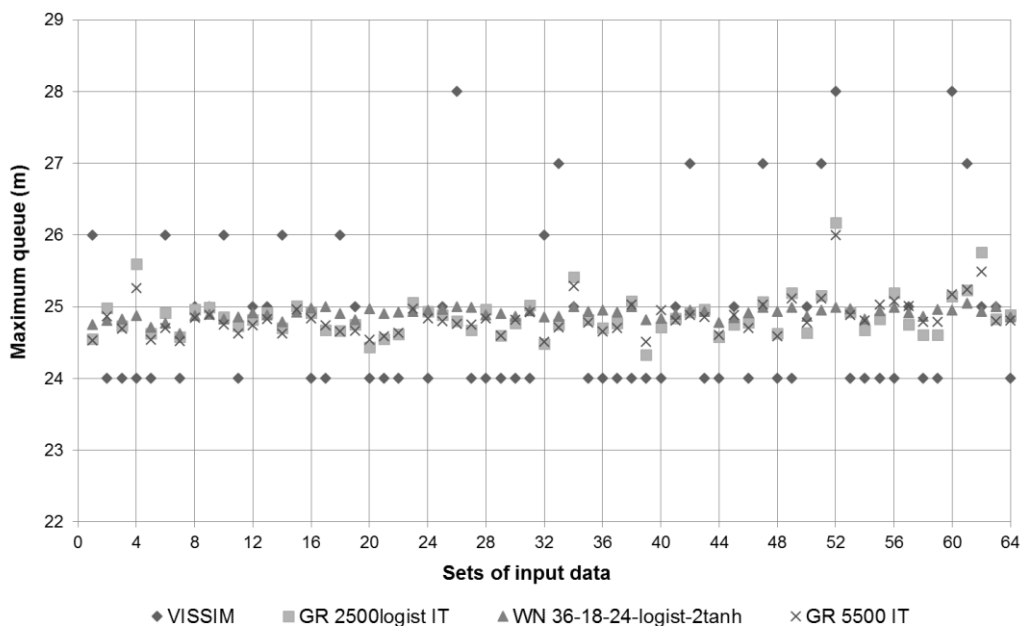
Figure 12 Predictions of maximum queue of best networks on validation set – Database II
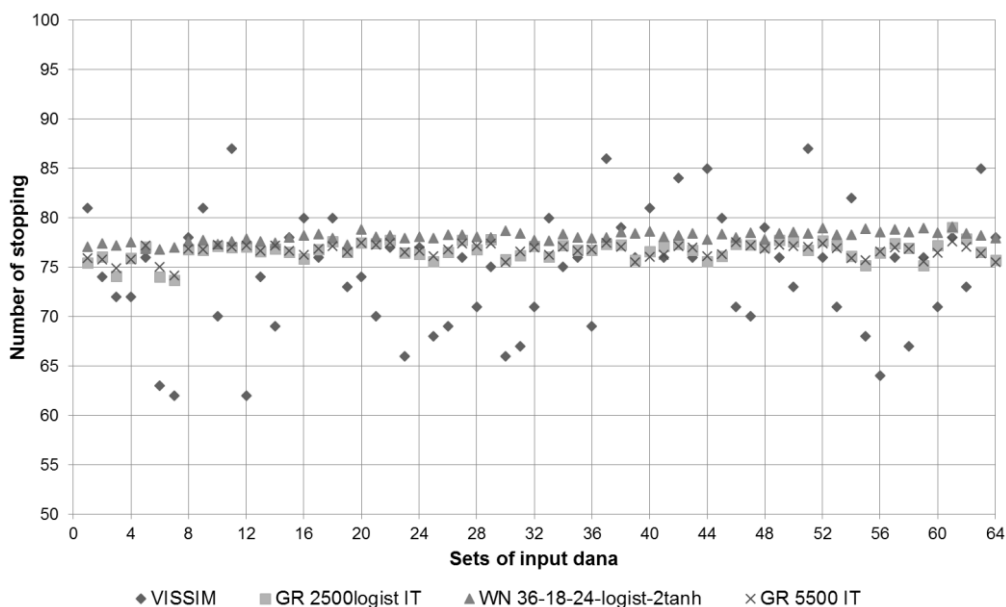

Figure 13 Predictions of the number of stopping of best networks on validation set – Database II

3   DISCUSION

To train the neural networks, we used a database of 1379 combinations of the model input parameters and the simulated VISSIM results of traveling time and queue parameters. We used such a large database so we could investigate the real neural network response and to prevent network overtraining or lack of variability in the network database as possible causes of bad results. However, such a large database is likely not necessary for achieving optimal results of neural network learning. In testing and evaluating the optimal neural networks for prediction needs by the two calibration methods, we analyzed 142 neural networks all together. The seventy neural networks trained on the Database I differed in architecture, number of hidden layers, number of neurons in hidden layers, and activation functions. Seventy-two neural networks were trained on Database II. The trained neural networks,

working on the training data, were compared by the criteria of correlation and mean absolute error; comparison by additional criteria is presented in [6].

During test validation, we examined the networks according to their state of training and generalization capability (ability to predict from unknown input data). The state of training was examined on 10 combinations of input parameters from the training set, and the prediction ability (network generalization) was tested on 15 new combinations of input parameters.

The best neural networks for each database were selected based on the final validation results. The validation data set contained 64 new combinations of input parameters and output traffic parameters. The networks that gave the best predictions compared to the expected results (VISSIM results), according to their minimum mean absolute prediction error, were GR5500-logist IT for Database I and GR2500-logist IT for Database II.

The calibration procedure for the computer program can be found in [6], as can the details of the MATLAB program used for the calibration procedure.

4   CONCLUSION

In this paper, we studied the response of neural networks for predicting the results of a microsimulation traffic model within the computer calibration program. We analyzed the traveling time and queue parameters during calibration and validation of the model; we chose these parameters because they are easily measurable in real traffic conditions. To study the applicability of the neural networks in calibration, we used the VISSIM microsimulation model. Our results show that a neural network can be used in calibration of this microsimulation model.

The neural networks showed good correlation for predicting travel times (Method I), but not for predicting queue parameters (Method II). Method II has potential in further research of the appropriate configuration of neural networks that would give better correlations for the queue parameters. However, we recommend Method I (Database I) for calibration using a neural network. Our results clearly show that a neural network is not an alternative to a microsimulation model. The neural network accurately predicted the traveling time, but no neural network accurately predicted the queue parameters. This paper does not implement neural networks to predict other traffic indicators obtained by a microsimulation model.

References

[1]   Cheu, R. L., Jin, X., Ng, K. C., Ng, Y. L., Srinivasan, D., 1998: Calibration of FRESIM for a Singapore expressway using genetic algorithm, Journal of Transportation Engineering, Vol. 124, No. 6, pp. 526-535., http://dx.doi.org/10.1061/(ASCE)0733-947X(1998)124:6(526)

[2]   Park, B., Won, J., Yun, I., 2006: Application of microscopic simulation model calibration and validation procedure: a case study of coordinated actuated signal system, Transportation Research Record, TRB, National Research Council, Washington, D.C., http://dx.doi.org/10.3141/1978-16

[3]   Kim, S.J., 2006: Simultaneous calibration of a microscopic traffic simulation model and OD matrix, PhD dissertation, USA, Texas A&M University.

[4]   Cicu, F., Illotta, P. F., Bared, J., Isebrands, H. 2011: VISSIM Calibration of Roundabouts Traffic Performance, In: Transportation Research Board Annual Meeting 2011, Washington D.C.

[5]   Ištoka Otković, I., Tollazzi, T., Šraml, M., 2013: Calibration of microsimulation traffic model using neural network approach, Expert systems with applications. Vol. 40, No. 15, pp. 5965–5974., http://dx.org.doi:10.1016/j.eswa.2013.05.003

[6]   Ištoka Otković, I., 2011: Using neural networks in the process of calibrating the microsimulation models in the analysis and design of roundabouts in urban areas, PhD thesis, University of Maribor, Faculty of Civil Engineering;
      http://dkum.uni-mb.si/Iskanje.php?type=napredno&stl0=Avtor&niz0=Irena+I%C5%A1toka+Otkovi%C4%87.

[7]   Ištoka Otković, I., Tollazzi, T., Šraml, M., 2013: Analysis of the influence of car-following input parameters on the modelled travelling time, Tehnički vjesnik - Technical Gazette. Vol. 20, No. 5, pp. 919–925.

[8]   Lawrence, S., Giles, C. L., Tsoi, A. C. 1997: Lessons in Neural Network Training: Overfitting May be Harder than Expected, Proceedings of the Fourteenth National Conference on Artificial Intelligence, AAAI-97, AAAI Press, Menlo Park, California, pp. 540–545.