Using a Neural Network for Forecasting in an Organic Traffic Control Management System

Matthias Sommer, Sven Tomforde, Jörg Hähner
University of Augsburg, Organic Computing Group, Eichleitnerstr. 30, 86159 Augsburg, Germany
Email: {matthias.sommer|sven.tomforde|joerg.haehner}@informatik.uni-augsburg.de

Abstract

Increasing mobility and rising traffic demands cause serious problems in urban road networks. Approaches to reduce the negative impacts of traffic include an improved control of traffic lights and the introduction of dynamic traffic guidance systems that take current conditions into account. One solution for the former aspect is Organic Traffic Control (OTC) which provides a self-organized and self-adaptive system founding on the principles of Organic Computing. This paper introduces further steps in enhancing the current OTC system with a forecasting technique based on neural networks. The prediction of short-term traffic conditions is an important component of an advanced traffic management system. It enables the system to prevent congestions and is able to react faster to changes in the traffic flow.

1 Introduction

The signalization of traffic lights at intersections is managed by special intersection controllers in the sense of embedded systems. These are highly specialized microprocessors containing components for operation modes, GPS or loop sensors. Current traffic light controllers normally have fixed-time signalizations with predefined sequences. Therefore, these controllers are not able to react to the current traffic conditions which leaves room for improvement.

Organic Traffic Control (OTC) [13, 20] offers a new approach to deal with the rising complexity of urban road traffic. By applying the principles of Organic Computing [14] to the field of urban traffic networks, the system becomes self-organized. It is capable of adapting to the changing environment while handling unforeseen situations, i.e. accidents.

With sophisticated techniques which reliably forecast the complex interactions of urban traffic, the OTC system can not just react to higher traffic flows and traffic jams, but proactively take action to prevent congestions. There are numerous methods used for traffic forecasting, but no one outperforms all other methods [1]. It was also discussed that traffic tends to exhibit a chaotic behavior [18], but many of the existing methods for traffic flow prediction lack the ability to cope with this aspect. In this paper we are exploring the benefits of using an artificial neural network (ANN) for a short-term traffic flow forecast and present further steps in enhancing the existing OTC system by a traffic flow forecasting component. This approach has shown to be able to deal with complex nonlinear predictions [1], which lets ANN appear as appropriate for the traffic domain.

An ANN is a highly simplified mapping of a biological neural system. It offers intelligent processing functions for learning, memorizing and predicting, while simultaneously dealing with uncertainty and non-linearity. The traditional approach for predicting upcoming traffic flows of a road network with a neural network is to learn one task at a time [22]. Another approach is using an ANN with more than one output [19]. As [9] and [12] state, multitask learning (MTL) may offer improvements to the generalization performance of the ANN by integrating field-specific training information contained by the extra tasks. The most considered task is the so-called main task, while the others are called extra tasks. In this paper we use a multitask learning recurrent neural network for predicting upcoming traffic flows. Experiments show that this approach offers reliable and robust predictions.

This paper is structured as follows. Section 2 briefly presents the current state of the OTC system. Afterwards, the constructed neural network is described in detail and possible ways of its integration in the OTC system are presented in Section 3. Section 4 reports the experimental results and discusses the possible benefit of applying this forecast technique to an automated traffic management system. Section 5 concludes with a summary of this paper and gives an outlook to further research.

2 Organic Traffic Control

Urban road networks are characterized by their great number of distributed, signalized intersections. Considering the dynamic nature of road traffic and the autonomous behavior of drivers, the traffic domain offers several characteristics that make it an interesting application for Organic Computing techniques. Earlier work applied the Observer/Controller architecture [16] for traffic signal control resulting in the OTC system [13, 20]. This system is able to optimize an intersection's signalization according to the observed traffic flows. Furthermore, organic intersections (meaning OTC-controlled) are able to interact and establish progressive signal systems (also called "green waves") in the network. The resulting signal coordination has shown to be very effective in minimizing the network-wide number of stops and travel times and in consequence the fuel consumption and pollution emission.

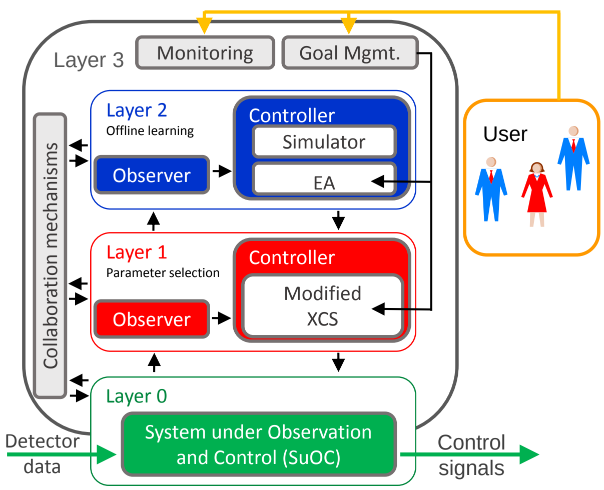

Figure 1: Architecture of the Organic Traffic Control system

As shown in Fig. 1, the OTC architecture extends the intersection controller within an existing road network, the System under Observation and Control (SuOC), by adding several layers on top. Traffic flows are recorded by detectors in the street surface and passed to an intersection observer on Layer 1. This Layer 1 contains a modified learning classifier system (based on Wilson's XCS [23]) where parameter sets for the signalization, based on the observed traffic flow data, are selected. In case of a new situation (no parameter set is known), the offline learning component on Layer 2, represented by an evolutionary algorithm, creates new classifiers and passes these back to Layer 1, where the new parameter set is applied. Simultaneously, Layer 1 reacts with the best possible action while Layer 2 searches for a new solution. Mechanisms for a decentralized collaboration between intersections enable these to communicate and exchange data. This allows the network to coordinate signalizations of traffic lights in response to changing traffic demands.

3 Short-term traffic forecasting

As presented in [1], numerous approaches exist for dealing with the problem of forecasting of traffic flows. These contain amongst others techniques like time series models [10], Bayesian networks [2] or Kalman filter theory [11]. Many of these approaches lack the ability to handle unforeseen situations like accidents or other abnormalities in the traffic flow (which are particularly targeted by Organic Computing techniques). In contrast, ANNs are able to deal with complex nonlinear predictions. As summed up in [6], neural networks were not only successfully applied to the task of traffic forecasting, but also to many other traffic-related topics like vehicle classification or traffic pattern analysis. They are a common method for forecasting short-term traffic flows [1]. By taking the current or the historical data of an intersection, we can predict its future traffic flows. [8] uses an ANN to forecast congestions for the next 30 minutes. The network is trained with historical data and continuously adapted to the last available data by a shifting learning method. The network is able to forecast the correct trend 85% of the time. [5] takes an object-oriented approach for predicting short-term traffic conditions of a section of the Pacific Highway. The ANN was capable of predicting speed data up to five minutes into the future. [3] compared different training methods for an ANN and came to the conclusion that a combination of a hybrid exponential smoothing and the Levenberg-Marquardt algorithm performed best for the prediction of traffic flow for some freeways. Besides that, the prediction of traffic patterns of highways tends to be simpler than the forecast of urban traffic flows as the traffic flow is usually more continuous.

3.1 The neural network model

A three-layered Elman neural network [4] with an input layer, a hidden layer and an output layer was used. This kind of neural network has no cycles between its nodes. Each neuron has directed connections to all neurons in the subsequent layer. Figure 2 illustrates the network. The input neurons I_1 to I_5 are fully connected to the nodes H_1 to H_{10} of the hidden layer. Same applies for the hidden layer and the output layer (Nodes O_1 to O_3). Additional nodes B_1 and B_2 add a bias to the hidden and the output nodes. Biases are values that are added to the output of each node (except input nodes). Thereby, an ANN is able to represent more functions. In addition to a normal feedforward neural network [7], Elman's network consists of an extra context layer (nodes C_1 to C_{10}). The neurons of the context layer receive information of the hidden neurons and pass their stored information back to them during the following iteration. By storing information, the context layer offers some kind of short-term memory. Therefore, it performs well on tasks as sequence-prediction, which let it seem appropriate for our application.

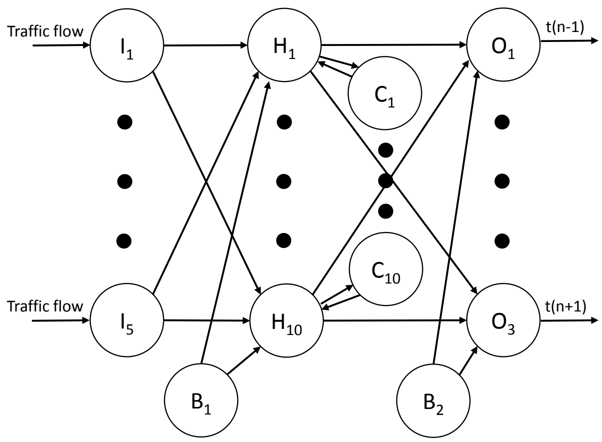

Figure 2: Structure of the Elman neural network

The traffic flows of the last five time instants (measured in vehicles per hour) were given as input to the neurons of the input layer. That input is preprocessed in the sense of correcting missing or false data. Filling in missing values or correcting false data in the datasets becomes necessary when information is lost because of defect street detectors or transmission errors. Afterwards, the processed input data is normalized to a range from 0 to 1. These preprocessed traffic flows are then passed to the nodes of the hidden layer. For the hidden layer, ten hidden neurons were chosen, as experiments proved this to be the most promising setup. An experiment consisted of the repeated run of 50 iterations while the number of hidden neurons was increased each iteration by one, starting from 5 up to 20 hidden neurons. By evaluating the average training error, the number of hidden neurons of the run with the best (lowest) training error was taken. The used activation function between input and hidden layer is the Sigmoid function, which is defined by formula 1. An activation function defines the way a node passes his output data to his output nodes.

$$f(x) = \frac{2}{1+e^{-2x}} - 1 \tag{1}$$

A linear function is selected as activation function between hidden layer and output layer. The output layer consists of three output neurons for the main and two extra tasks. The form of this linear function is described as formula 2.

$$f(x) = x \tag{2}$$

In general, literature in the ANN domain shows that finding a setup for these parameters is highly application-specific. We use a hybrid strategy for the training of the neural network which allows us to use a secondary training algorithm. First a resilient propagation algorithm [17] is executed until it is no longer improving on the training error. The second algorithm used for training was the classic Levenberg-Marquardt algorithm [24]. This approach leads to faster and better improvements during the training phase. By furthermore using early stopping to prevent overfitting, the training algorithm has the ability to complete the training when the error on the validation set would increase. Further stopping criteria are the maximal training error and a maximal number of training epochs. After the training, the network gets pruned. This is a common method for reducing the size of the hidden layer. By removing the least important neuron, the network size decreases and improves its generalization capabilities; besides that, a smaller network is faster and cheaper to build [15].
At last, the prediction is made using MTL for the forecast. By choosing the last k traffic flows, we can predict the vehicle flow at time n (denoted by $t(n)$) by selecting the traffic flows at the timesteps $t(n-1)$ and $t(n+1)$ as extra tasks correlated to the main task. The traffic flows one time step in the past $t(n-1)$ and one time step in the future $t(n+1)$ apparently are related to the forecast of $t(n)$. Therefore, they play an inductive bias role so as to increase the forecast accuracy of $t(n)$.

3.2 Further implementation in the OTC environment

The generic Observer/Controller architecture [16] (Fig. 3) contains a prediction component to forecast trends in the SuOC. At a signalized intersection, predictions show possible future traffic developments. These predictions can be based on current or historical traffic data. Current traffic data allows for short-term predictions, whereas historical data enables long-term forecasts by recognizing recurring patterns in traffic flows. At the moment, only a short-term prediction is made by the presented neural network.

As shown in Figure 3, the observer on Layer 1 consists of a component monitoring the detection data from an intersection, a preproccesor (depicted by the component labeled "Situation Analyzer") extrapolating the current traffic flows and a performance analyzer deriving performance measures for the intersection's current signal plan.

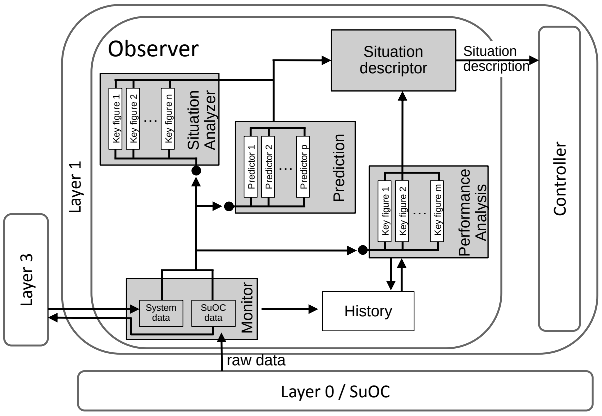

Figure 3: An observer/controller architecture for traffic signal control [20]

The prediction component might be added right after the preprocessor component. The extrapolated traffic flow of the preprocessor then serves as an input for the forecast. Based on these traffic flows, a short-term prediction is made and passed as extra input to the situation descriptor along with the data from the preprocessing component and the performance analyzer. The situation descriptor creates an description of the situation and passes it to the controller on Layer 1.

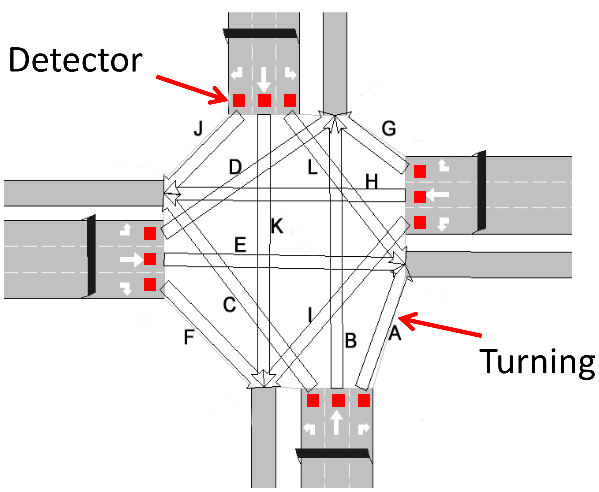

Figure 4: An exemplary intersection with turnings and detectors

Figure 4 shows an example of a four-armed intersection. The turning movements crossing the intersections are labeled from A to L. Detectors in the street surface measure flow-values for each turning movement. The output of the prediction component are the traffic flow forecasts of every turning movement of an intersection (12 forecast values for the example intersection).

4 Experimental results

Beyond others, ANNs are divided into feedforward, Jordan and Elman neural networks [4]. Jordan and Elman add an additional context layer to the network, which stores the value of the previous iterations, therefore offering some kind of short term memory. For measuring the performance of our approach, we implemented following variants, which will be compared in the following.
We constructed a MTL ANN with five input neurons, ten hidden neurons and three output neurons for each of the mentioned types. The data used for the inputs are the vehicle flow rates measured in $\frac{vehicles}{hour}$ of a discrete time series which were raised every five minutes. The used traffic flows refer to the intersection shown in Fig. 4 which depicts an intersection in Hamburg, Germany. Data sets for testing were previously classified in traffic flows from Monday to Thursday, Friday, Saturday and Sunday[1]. The traffic flows of Monday to Thursday have great similarities and therefore can be combined in one data set. The validation set and the training set are based on traffic flows of two typical weeks resulting in 2016 sample points each, also measured in $\frac{vehicles}{hour}$. The maximum flow was 1050 cars per hour. The prediction output is the future traffic flow of a single intersection five minutes into the future. The network was initially trained with the mentioned training set. 300 epochs are selected as the maximal training duration. The training also finishes in case the training error drops below 0.002 or if the error on the validation set increases during training.

As seen in Table 1, the Elman network with MTL performed very good for the prediction of Monday to Thursday resulting in an overall prediction error of 4% to 5%. A graphical representation of the traffic flow forecasting compared to the ideal results is given in Figure 5. In average the prediction of the future traffic flow in the next five minutes was off by 48.9 cars. The network performed almost equally good for Friday. Saturday and Sundays performed even better as the number of traffic participants is decreased and the changes in the traffic flow are smaller on weekends.

Overall, MTL performed better than singletask learning (STL) for all three networks (Tab. 2). By only looking at the results of STL, the Jordan neural network performed best. This approach had an average root mean square error (RMSE) of 51.7 cars compared to the real flow values. While Jordan only improved slightly by using MTL (1.5%), feedforward and Elman increased its prediction accuracy by 12.1%, respectively 10.9% (denoted by e). The feedforward network had the biggest

[1]The traffic data and the signal schedules are provided by the Schmeck Ingenieurgesellschaft mbH, Hamburg

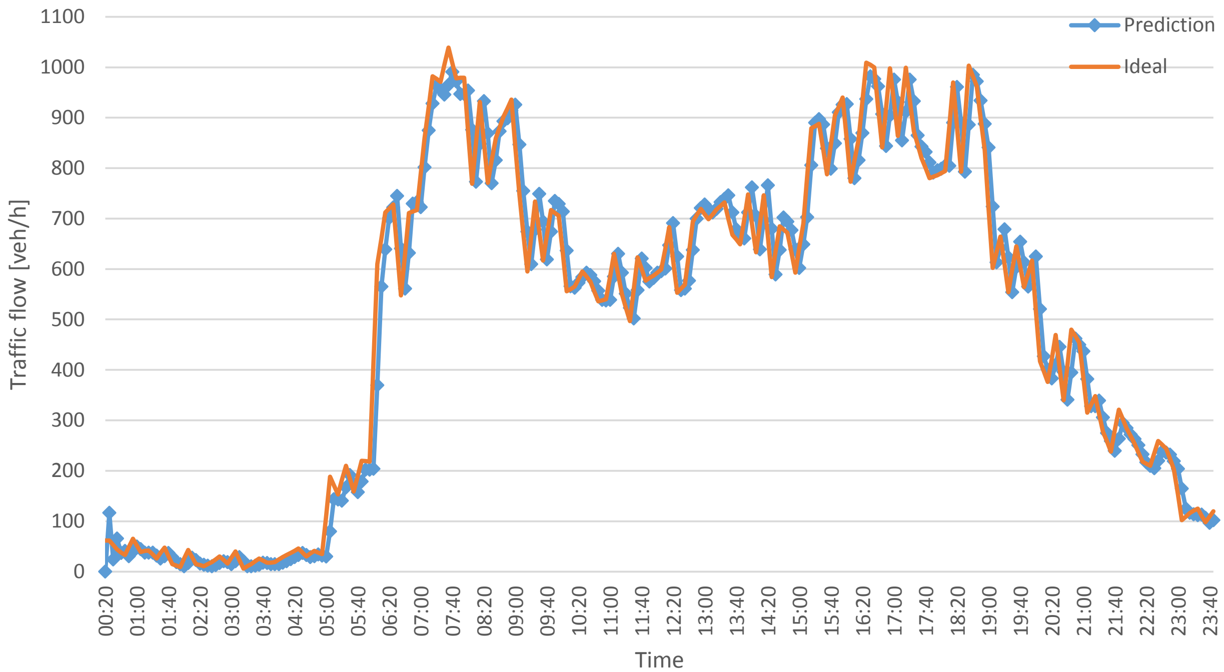

Figure 5: Traffic flow forecasting for monday with an MTL Elman network compared to the ideal values

	Mo.-Th.	Friday	Sat.	Sunday
RMSE	48.9	47.4	32.1	30.9
Deviation in %	4.6%	4.5%	3.0%	2.9%

Table 1: Performance of the MTL Elman network for daily test sets (root mean square error (RMSE), lower is better)

RMSE	Feedforward	Jordan	Elman
STL	57.6	51.7	54.9
MTL	50.6	50.9	48.9
e	12.1%	1.5%	10.9%

Table 2: Comparison of the prediction accuracy of different neural network types with STL and MTL (lower values are better)

additional benefit of MTL (57.6 cars), compared to STL (50.6 cars). The Elman network with MTL resulted in an average RMSE of 48.9 cars for the prediction of a Monday, while with STL it had an average error of 54.9 cars. Compared to the results of [21], this ANN results in a 38% better prediction accuracy. Therefore the combination of an Elman network with the technique of MTL showed to be the most promising approach and will therefore be used in the OTC system.

5 Conclusion

This paper presented a forecasting technique based on artificial neural networks. Based on the awareness, that other approaches have weaknesses, we chose an artificial neural network for the traffic flow forecast of an organic intersection. The evaluation of the neural network was made outside of the Organic Traffic Control system. Based on real data for an intersection in Hamburg, Germany, the experiments for the neural network showed promising results. A multitask learning Elman recurrent neural network with three outputs offered the best predictions resulting in a prediction error below 5 percent. Compared to a feedforward and a Jordan neural network, this network performed best. Using multitask learning the prediction resulted in a 11% better prediction accuracy compared to using singletask learning. By adding other types of data like the current day or time, or the traffic flows of neighboring intersections as input, the network might increase its prediction accuracy. By performing a second training phase with traffic data occurred at incidents, the network might improve its ability to deal with abnormal situations. Enhancing the prediction component by a long-term forecast may further enable the system to classify upcoming traffic flows better and consequently improve the forecasting accuracy. Next steps will also include the integration of the neural network as prediction component into the Observer component of the Organic Traffic Control environment.

References

[1] Bolshinsky, E., and Freidman, R. Traffic flow forecast survey. Tech. rep., Technion - Israel Institute of Technology, 2012.

[2] Castillo, E., Menendez, J., and Sanchezcambronero, S. Predicting traffic flow using Bayesian networks. *Transportation Research Part B: Methodological 42*, 5 (June 2008), 482–509.

[3] Chan, K. Y., Dillon, T. S., Singh, J., and Chang, E. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg-marquardt algorithm. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS 13* (2012).

[4] Cruse, H. *Neural Networks as Cybernetic Systems*. 2006.

[5] Dia, H. An object-oriented neural network approach to short-term traffic forecasting. *Euorpean Journal Operation Research 131* (2001), 253–261.

[6] Dougherty, M. A review of neural networks applied to transport. *Transportation Research Part C: Emerging Technologies 3*, 4 (1995), 247 – 260.

[7] Fine, T. L. *Feedforward neural network methodology*. Statistics for engineering and information science. Springer, New York, 1999.

[8] Gilmore, J., and Abe, N. A neural networks system for traffic congestion forecasting. In *Proceedings of 1993 International Joint Conference on Neural Networks* (1993).

[9] Jin, F., and Sun, S. Neural network multitask learning for traffic flow forecasting. In *IJCNN* (2008), pp. 1897–1901.

[10] Moorthy, C. K., and Ratcliffe, B. G. Short term traffic forecasting using time series methods. *Transportation Planning and Technology 12*, 1 (1988), 45–56.

[11] Okutani, I., and Stephanedes, Y. J. Dynamic prediction of traffic volume through kalman filtering theory. *Transportation Research Part B: Methodological 18*, 1 (1984), 1–11.

[12] Orr, G., and Müller, K.-R. *LNCS 1524*. Springer, 1998.

[13] Prothmann, H. *Organic Traffic Control*. KIT Scientific Publishing, 2011.

[14] Prothmann, H., Tomforde, S., Branke, J., Hähner, J., Müller-Schloer, C., and Schmeck, H. Organic Traffic Control. In *Organic Computing – A Paradigm Shift for Complex Systems*, C. Müller-Schloer, H. Schmeck, and T. Ungerer, Eds. Birkhäuser Verlag, 2011, ch. 5.1, pp. 431–446.

[15] Reed, R. Pruning algorithms-a survey. *Trans. Neur. Netw. 4*, 5 (1993), 740–747.

[16] Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., and Schmeck, H. Towards a Generic Observer/Controller Architecture for Organic Computing. In *Beiträge zur Jahrestagung der Gesellschaft für Informatik 2006* (2006), pp. 112–119.

[17] Riedmiller, M., and Braun, H. Rprop - a fast adaptive learning algorithm. Tech. rep., Proc. of ISCIS VII), Universitat, 1992.

[18] Smith, B. L., Williams, B. M., and Oswald, R. K. Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies 10*, 4 (2002), 303 – 321.

[19] Thrun, S., and Mitchell, T. M. Learning one more thing. Tech. rep., 1994.

[20] Tomforde, S. *Runtime adaptation of technical systems*. Südwestdeutscher Verlag für Hochschulschriften, 2012.

[21] Volhard, C. Vorhersage der Verkehrsentwicklung für autonome Lichtsignal-Anlagensteuerungen. Master's thesis, University of Hannover, 2009.

[22] William, B. M. *Modeling and forecasting vehicular traffic flow as a seasonal stochastic time series process*. PhD thesis, Virginia. Dept. of Transportation, University of Virginia, 1999.

[23] Wilson, S. W. Classifier fitness based on accuracy. *Evolutionary Computation 3*, 2 (1995), 149–175.

[24] Xi, S. *Nonlinear Optimization Methods*. Higher Education Press, 1992.