Application of neural networks to long term prediction of queue length at an urban traffic junction

Corinne Ledoux
Institut National de Recherche sur les Transports et leur Sécurité
2, Avenue du Général Malleret Joinville
94114 Arcueil Cedex - France
email : ledoux@inrets.fr

Mots clefs : Réseaux connexionnistes, prédiction de séries temporelles, application aux transports

Ce rapport présente un exemple d'application des réseaux connexionnistes dans le domaine du transport. Les réseaux connexionnistes font ici partie intégrante d'un systéme de régulation de trafic en milieu urbain. Ils ont pour tâche de prédire la longueur de la file de véhicules sur un tronçon dont les départs sont régulés par des feux tricolores. Peu introduits dans le domaine du transport, les réseaux connexionnistes ont démontré leurs capacités à s'adapter à la dynamique complexe du trafic.

Keywords : Neural networks, time series prediction, transportation

Our current research focuses on the integration of neural network prediction techniques into an adaptive real time urban traffic control system. The conventional techniques applied within the traffic community are model based techniques, originating from complex hydrodynamic and particles theories. Neural networks can be considered here as a competitive way of achieving traffic flow models. They are introduced here to predict the queue length of vehicles, on a signalized urban link.

1 Introduction

Our current research lies within the general framework of Urban Traffic Control. More specifically, we are involved in the project called "Intelligent Junction" which aims at managing the traffic within a local urban junction. It is built on two major functions, but the one of particular interest here is the local control of the traffic lights. A "good" local control policy is mainly based on an accurate traffic flow model. The purpose of this paper is to present the results of our investigations on the potentials of neural networks to model and predict traffic flow.
The paper is organized as follows. The first part is dedicated to an overall presentation of the project "Intelligent Junction". Then, we focus on the local control function we are currently developing and within which neural networks are integrated. Two reviews are then proposed to place our research in the context of time series prediction and neural network field; the first one presents the conventional techniques applied to perform prediction, the second one presents the main neural architectures widely used for forecasting. Next part describes the neural predictor applied. Finally, based on numerical simulations, we conclude the paper on their effectiveness to queue length prediction.

2 The project "Intelligent Junction"

Today, most large cities in industrialized countries are confronted with chronic traffic congestion. To cope with this problem, traffic management in urban areas seems to be inevitable. The project "Intelligent Junction" currently developped at INRETS aims at managing the traffic within a local urban junction. Managing the traffic involves two major functions : the Automatic Incident Detection and the local Control (see Figure 2).

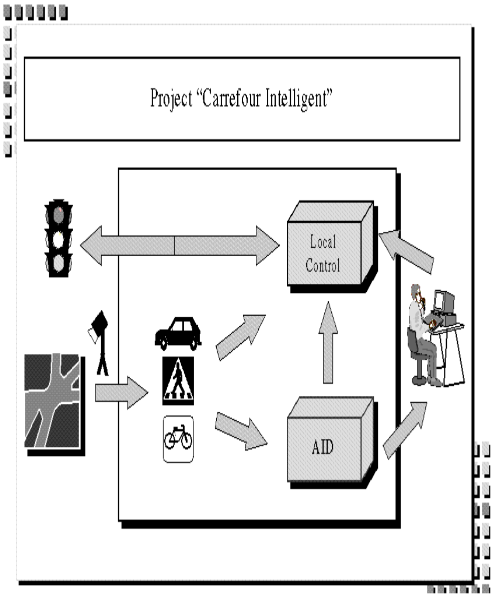

Figure 2: The project "Intelligent Junction"

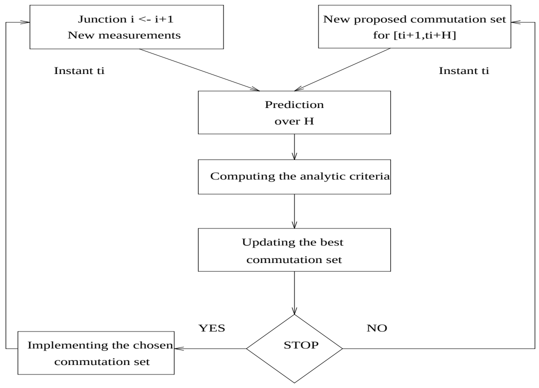

Figure 1: The fully adaptive UTC system

Based on information - provided by sensors such as video sensors, magnetic loops or pneumatic detectors - about the behaviour of vehicles, cycles, pedestrian within the junction, the former function is to detect any abnormal functioning. An abnormal functioning can be a vehicle stuck in the middle of the junction without any particular reason, an accident, a misfunctioning of the traffic lights.
In order to avoid the blocking of the junction, resulting from an incident, decisions have to be taken. An human operator can be warned. He is then to provide information to the local control function, which is in charge of modifying the traffic lights. This modification can also be done automatically, without the intervention of any human operator.
The local control function can be activated even though no incident has been detected. It is mostly required to adapt in real time to time varying traffic conditions over the whole day. Once again, time varying traffic conditions are detected by sensors.
Next section presents in further details the local control function we are currently developping, and within which neural networks are integrated.

3 The adaptive local control proposed

The policy of the adaptive Urban Traffic Control systems designed nowadays is the following. The decision to switch or not to switch the traffic lights is taken according to the optimization of a criterion whose value is evaluated over the next H seconds, representing the chosen time horizon.
The following section deals with the overall architecture of the adaptive local control proposed. The second one goes deeper into the architecture and focuses on the task that has to be performed by neural networks .i.e. the traffic flow modelling. At first, section 3.2 presents it from the conventional viewpoint of traffic engineering. Then, in a second time, section 3.4 justifies the application of neural networks to traffic flow modelling.

3.1 The architecture

The local control function we propose runs as depicted in Figure 1. Predictions of traffic characteristics for the next control period, say $[t_i+1, t_i+H]$, are computed, providing measurements from the sensors obtained at current time t_i, and the set of switches of the traffic lights over the horizon H. It is important to notice that the traffic conditions can not be predicted without the set of potential switches over H being made available. Based on these predictions, the analytic criterion to be minimized is computed. Finally, the system determines whether a new control policy is warranted or not. In case a new policy is required, the most beneficial one is selected. The most beneficial control policy is then applied at time t_i+1. The same sequence is to be repeated for time t_i+2.
The criterion minimized is usually the total delay of the vehicles in the network but additionally it can be the sum of the stops at the traffic lights ... It is expressed either in term of the queue length or the

number of vehicles on the links.
So far our research focuses on the traffic flow and prediction model.

3.2 Traffic flow modelling

The purpose of traffic flow modelling is to reconstruct from data provided by sensors a realistic vision of the true traffic conditions. It is obvious from what precedes that, for that particular Urban Traffic Control system, only the use of a traffic flow model makes the projection in the future possible.
To establish such models, a theory is needed, but there is no unified theory of traffic flow, rather there are several theoretical approaches to describe this phenomenon. The theories, proposed through the years, consider traffic either as a fluid or as particles interacting in a complex network.
Generally speaking the conventional traffic flow models are built on complex analytical equations. They present the disadvantage of requiring a large know how and long term experience in understanding and describing traffic flow phenomenon. Therefore they are usually far too complex to be used in real time applications. Most fully adaptive traffic control systems designed nowadays integrate a simplified macroscopic description of traffic flow. No doubt the resulting modelling is even rougher, but as far as the main traffic evolution tendency is concerned, it has been proven to be sufficient. [1] presents in further details the different traffic model types.

3.3 Originality

As an alternative to the previously introduced traffic flow models, we propose neural networks. At the present time, no operational UTC system integrating such techniques exists. A mathematical neural model is built to describe the traffic flow over a single signalized link. Our attention focuses on the modelling of the queue length, number of vehicules and departures evolving over time. Our objective is to perform long term prediction; based on the true value of the queue length, provided by sensors at time t_0, its value is to be computed at time $t_0 + H$, with $H \geq 1$.

3.4 Justifying the neural approach to traffic flow modelling

Neural networks are flexible non-linear data driven models. They exhibit interesting properties that make them good candidates for traffic modelling :

1. They provide a mathematical model based on empirical data.
2. They are universal approximators and can hence theoretically model highly complex and strongly nonlinear systems.
3. They can be trained off line and adapted on line to adjust to time varying dynamics.
4. They can handle multi-variate systems.
5. They are suited for real time applications.
6. They can easily be integrated into hardware.

The first feature means it is possible to build a neural network that models the dynamics of any traffic variable, directly from data. No analytical assumption is to be made. The second characteristic is strongly required for urban traffic modelling. Indeed, switching the signal state induces strong non-linearities in the traffic flow. The third property is hopefully to lead to a better adequation between the modelling and real time varying traffic conditions.
Furthermore, neural networks require :

- less know how and long term experience in understanding and describing traffic flow phenomenon than conventional modelling techniques, because they are built directly from empirical data.
- no preliminary knowledge about real traffic flow conditions.

4 Traditional approaches to Time series analysis

The statistical approach for forecasting involves the building of stochastic models to predict the value of an observation. It is based on the strong hypothesis of stationarity of the signal to predict. Depending on the length of the prediction horizon - short, medium or long term - different techniques are applied [2]:

- Short term. The techniques used are mainly extrapolation methods ; the prediction is a weighted sum of past values and/or past errors. They consist in identifying the parameters of average models (MA) or auto-regressive models (AR) or a combination of both models (ARMA).
- Medium term. Accurate prediction can not be computed only from past values and/or past errors. Exogenous explicative variables have to be introduced into the model. Models such as auto-regressive exogenous model (ARX), auto-regressive exogenous average model (ARMAX) or the Box and Jenkins model are identified.
- Long term. The pattern to predict exhibits a trend, seasonal and periodic characteristics. Long term prediction techniques aim at estimating the trend. It is usually adjusted to the logistic function.

One often does not make a so clear distinction between the short term and medium term prediction techniques ; one refers to system identification techniques. For further details on the topic, see [3]. All the previously mentioned models are linear. Although they have been broadly studied and understood, linear techniques tend to fail when the system exhibits strong non-linearities. That is the reason why we concentrate on non-linear modelling.

The general results from the identification of dynamical linear systems, presented in [3], can naturally be applied to non-linear systems. In [4], Ljung addresses "some aspects on non-linear black box modeling in system identification". It is clear that neural networks are not the only potential non-linear model that can identify a dynamical system. See [5] for an unified overview of non-linear black box models in system identification.

5 Neural networks and time series prediction

Within the neural network framework, several neural architectures have been designed to address the specific problem of time series prediction. In [6], the authors present a classification of neural networks with respect to time integration. It is depicted in Figure 3.

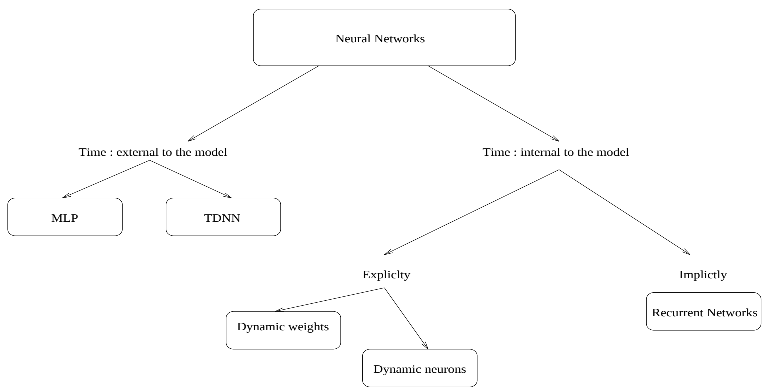

Figure 3: A classification of neural models with respect to time integration

The lowest time integration level consists in not introducing time into the artificial neural network; it is external to the model. The temporal data are preprocessed in a time to space transformation so that classical neural network architectures such as MLP or TDNN can perform the prediction task. It is the interpretation of the space transformation that is related to time. It is the most popular neural network based forecasting technique used so far. It has proved its efficiency in predicting electricity load, water load, in financial modelling.

Short term memory of the preceding states of the neurons is kept and further reintroduced into the network, thanks to local recurrent connexions. Time is internal to the model and taken into account in an implicit way. Popular recurrent architectures are those of Jordan, Elman. Nevertheless, a lack of efficient training algorithms makes these models not that easy to handle.
A further step into the time integration is its explicit representation in the neural architecture. Time can either be introduced at the synaptic weight level, or at the neuron level.
The traffic prediction results presented in the remaining of the paper are provided by a MLP. Even though, it is clear that it is not the most suited architecture to perform long term prediction, it is rather straightforward to build and it leads to fairly good results.

6 Experimentations

6.1 Inputs to the neural network

Let us remind that our objective is to perform long term prediction of the queue length. Our approach is the following. At first, the neural net is trained to perform a one step ahead prediction. Then the H step ahead prediction of the queue length is reached by iterating H times the one step ahead predictor. Information relevant to the one step ahead prediction of the queue length is then to be determined. The value of the queue length at time $t+1$, refered as to the output from the neural network and denoted $lq(t+1)$, depends on the following inputs (see Figures 4 and 5):

- its value at time t, $lq(t)$,
- the values of arrivals into the queue over the time interval $[t, t+1[$. They are obtained from the incoming arrivals at the beginning of the link, denoted $A(t)$, with a time delay τ which is a non-linear function of the length of the link, the length of the queue at time t, and the speed of the vehicles.
- the departures from the link over $[t, t+1[$, denoted $D(t)$. They are strongly connected to the state light. Indeed, during red light, departures are null. During green light, they are constrained by the saturation flow rate.

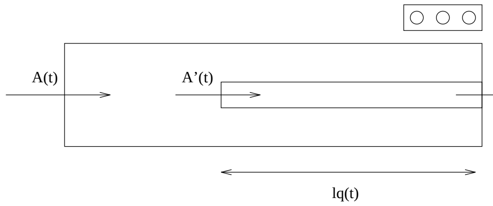

Figure 4: Information relevant to the prediction of the queue length

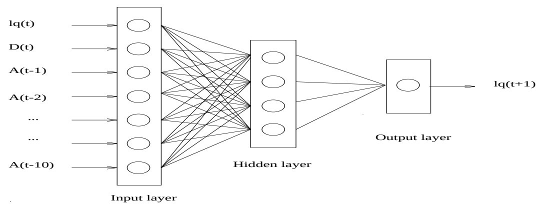

Figure 5: Neural network architecture used

The neural architecture used is depicted in Figure 5. The activation function is the classical sigmoid function. It is obvious that the input output relationship modeled by the neural network results in the approximation of the unknown function $\mathcal{F}$:

$$lq(t+1) = \mathcal{F}(lq(t), A(t-1), \cdots, A(t-\tau), D(t))$$

The neural network designed results in the generalization of a linear ARX model into a non-linear ARX model, so the latter comparisons between both models.

6.2 Training the network

No specific training algorithm, taking explicitly into account the temporal aspect of the problem, has been used so far. It is a rather classical one step ahead minimization gradient descent based process.
Let us denote by $lq(t)$ the queue length measured by the sensor, and $\hat{l}(q \mid \theta)$ the queue length predicted by the neural network parameterized by the synaptic weight vector θ. The root mean square error then writes

$$rmse = \sqrt{\frac{1}{N}\sum_{t=1}^{t=N}(lq(t) - \hat{l}q(t))^2}$$

The "best" neural predictor is the one whose synaptic weight vector θ leads to the minimization of the root mean square error. The estimation of the synaptic weight vector is then reduced to a complex non-linear multivariate minimization procedure. The minimization is usually a local iterative process in which an approximation of the function in the neighborhood of the current point in the weight space is minimized. The approximation is often given by a first or second order Taylor expansion. Each iteration i consists of estimating the step size μ^i, the gradient of the mean square error g^i and the matrix that modifies the search direction R^i.
Let us denote $\hat{\theta}^{i+1}$ the estimated value of the synaptic vector θ reached at the $(i+1)^{th}$ iteration. It writes

$$\hat{\theta}^{i+1} = \hat{\theta}^i - \mu^i * (R^i)^{-1} * g^i$$

In the optimization technique used, R^i is the Gauss Newton direction search.

6.3 Data used

The data set needed is to consist of the time series made of :

- the value of the queue length at time t,
- the values of the incoming arrivals far back in the past.
- the values of the departures from the link over $[t, t+1[$.

Unfortunatly, at the present time, no such real data set is available. Preditions are then to be performed on simulated data. Data are provided by SSMT which stands for Simulateur Semi Macroscopique de Trafic. It is a traffic flow simulator, designed at INRETS in 1983. Based on its outputs, the queue length and departures variables are "rebuilt". Plugged into the architecture depicted in Figure 1, the neural net is fed with queue length measurements provided by sensors. As far as SSMT provides a realistic vision of the true traffic conditions, learning measured or simulated queue length is about the same.
The conditions of our experimentations are as follows. Simulations involve a 200 meter long link, the speed is fixed to 20 m/s. In fluid traffic conditions, 10 seconds are needed to travel the link. It means a vehicle entering the link at time $t_0 - 10$ is added to the queue length at time t_0, at the latest. So the time delay parameter τ is fixed to 10 seconds. Only one particular set of switches of the traffic lights has been evaluated so far. Its consists of periodic sequences of 26 seconds of green light and 44 seconds of the red light.

6.4 Some results

Two sets of simulations have been run. The first one involves linear models. The System Identification ToolBox, available in MATLAB has been used [7]. The second set of simulations involves non-linear models. The Neural Network ToolBox, developped in MATLAB by Jonas Sjöberg, is used [5]. All well known linear models, available in the SITB are generalized into non-linear Multi Layer Perceptron based models, in this Neural Network ToolBox.
The linear model is an Auto Regressive eXogenous model whereas the nonlinear model is a NARX. The NARX model includes 4 hidden units ; this architecture which provides the best prediction is reached after successive trials.
They both have the same inputs, that are depicted in Figure 5. The data are centered and reduced before estimating the parameters.

The parameters of both models are estimated based on the minimization of the one step ahead prediction error. As far as the NARX model is concerned, about 40 iterations are needed to converge into a minimum of the mean square error, using the Gauss Newton algorithm. A 30 minute estimation set, leading to 1800 patterns, is used. The same amount of patterns is used for validation purpose. A so huge data base is required to cover the length queue space variation.
So far, the prediction process assumes that the future values of the incoming arrivals A and departures D are provided by external models.
The k step ahead prediction ability of both models is compared using the following statistical test. It has previously been used in [8] for short term traffic forecasting. Let us assume that the time series is generated by a model M_0. Let us denoted H_0 this hypothesis. Let H_1 be the hypothesis that the time series is generated by the model M_1 which is different from M_0. The purpose of the statistical test is to verify the validity of the hypothesis H_1 against H_0. The statistical test writes

$$U = \sqrt{\frac{\sum_{t=1}^{t=N} (\frac{M_0(t)-lq(t)}{lq(t)})^2}{\sum_{t=1}^{t=N} (\frac{M_1(t)-lq(t)}{lq(t)})^2}}$$

wherein

$M_0(t)$ is the predicted value from model M_0 at time t,
$M_1(t)$ is the predicted value from model M_1 at time t,
lq(t) is the observed value of the queue length at time t

The interpretation which can be given to this statistic is as follows :

$U \approx 1$: Both hypotheses are equal,
$U < 1$: The hypothesis H_0 is true , the time series is better modelled by M_0
$U > 1$: The hypothesis H_1 is true , the time series is better modelled by M_1

As far as our application is concerned, the model M_0 is the direct k step ahead linear prediction model, whereas M_1 is the k time iterated non-linear prediction model.

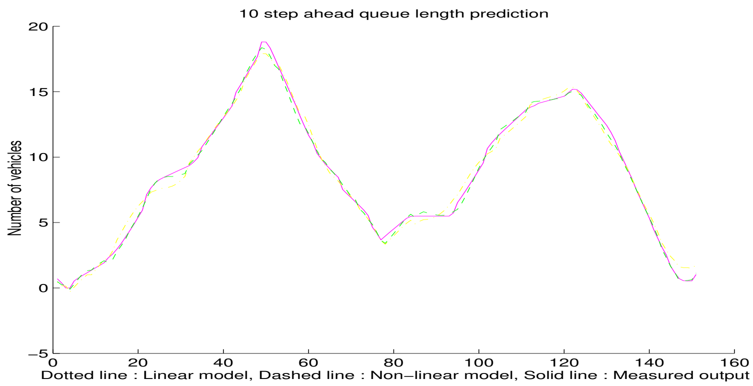

Figure 6: Predicted queue length dynamics

	R.M.S.E. for k = 10	
Model	Est. set	Valid. set
ARX	0.618	0.728
NARX	0.129	0.138

Figure 7: Numerical results

In Figure 6 is depicted the evolution of the queue length over time. The x axis represents time expressed in second. The increase of the queue length corresponds to a red light, and on the contrary its decrease results from a green light. According to the value of the incoming arrivals, the queue length may not decrease down to zero during the green light. The queue length may not increase during the red light, if past arrivals are null. In this figure, are plotted the 10 step ahead queue length prediction performed by the ARX model (in dotted line) and by the NARX model (on dashed line). The solid line represents the measured queue length. Those data are extracted from the validation set - data not used during the estimation of the parameters. They are the continuation of the estimation set. Note the fairly good fit of both curves. Nevertheless, let us emphasize that the non-linear predictor performs better than the linear model : almost no distinction between the dashed and solid line can be seen. The main non-linearities in the queue length dynamics, resulting from changing in the incoming arrivals, are better modelled. Taking into consideration the changing incoming arrivals is the key of accurate prediction.

Table 7 provides the values of the root mean square error on the estimation and validation set after training the network. Let us mention that the mean value of the queue length, on the whole data set, is 12.17 vehicles, with a standard deviation of 6.88. Let us at first consider the linear model. On the estimation set, the root mean square error is 0.618 vehicle and on validation data it rises up to 0.728 vehicle. These figures can be compared with those of the non-linear model. The rmse on estimation data is 0.129, whereas on validation data it is 0.138. These results emphasize the very first conclusion; non-linear model is better at forecasting. As far as the statistical test is concerned, its value on the estimation set is 1.50 and on validation set is 2.26 ; it underlines that once again the non-linear performs better than the linear model.

7 Conclusions

Our current research is directed towards the integration of neural modelling techniques into an adaptive real time urban traffic control system. Neural networks can be considered here as a competitive way of achieving traffic flow models.
A first set of simulations have underlined the benefits of non-linear models, such as NARX feed forward neural network based models, over linear models such as ARX models. Our current research aims at a deeper comparison of both models, at evaluating the potentials and limits of a MLP net to perform long term prediction of the queue length. Even though, the non-linear predictor exhibits a fairly good behaviour, it is clear that it is not the most suited architecture to perform long term prediction. Our future research will focus on prediction using recurrent networks. In order to reach a fully integrable neuro-traffic flow modelling, a prediction of both the incoming arrivals and departures is required.

References

[1] C. Ledoux. An urban traffic control system integrating neural networks. In *Eigth International Conference on Road Traffic Monitoring and Control*, London, UK, 23 - 25 April 1996.

[2] B. Burtschy. Séries chronologiques - les outils du statisticien, les besions du prévisionniste. In *Statistiques et Méthodes Neuronales - Ecole Modulad*, 6,7,8 Décembre 1995.

[3] L. Ljung. *System Identification : Theory for the User*. P.T.R Prentice Hall, 1987.

[4] L. Ljung. Some aspects of nonlinear black box modeling in system identification. Technical report, LiTH-ISY-R-1795, 1995.

[5] J. Sjöberg, Q. Zhang, A. Benveniste, B. Deylon, P.Y Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification : a unified overview. *Automatica*, 31, 1995.

[6] J. C. Chappelier and A.Grumbach. Time in neural networks. *SIGART Bulletin*, 5, 1994.

[7] L. Ljung. *System Identification TOOLBOX*. The Math Works, Inc., 1995.

[8] M. S. Dougherty, H.R. Kirby, and R. D. Boyle. Using neural networks to recognise, predict and model traffic. *Artificial Intelligence Applications to Traffic Engineering*, pages 233–250, 1994.