An Integrated Forecasting and Regularization Framework for Light Rail Transit Systems

Fabio Stella  
DISCo, Università degli Studi di Milano-Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, ITALY

Davide Bogni, Matteo Benzoni  
Project Automation, Viale Elezia 42, 20052 Monza, ITALY.

Vittorio Viganò  
Consorzio Milano Ricerche, Via Cicognara 7, 20135 Milano, ITALY.

In recent years, with half the world’s population living in towns and cities and most of them relying heavily on public transport to meet their mobility needs, efficient and effective public transport operations have become critical to sustainable economic and social development. Nowadays, Light Rail Transit Systems are considered to be the most promising technological approach to satisfy these needs, i.e. to ensure efficient and reliable urban mobility. However, Light Rail Transit Systems are subject to frequent minor disrupted transit operations, often caused by stochastic variations of passenger demand at stations and traffic conditions on the service routes, which increase passenger waiting times discouraging them from using the transit system. Although these minor disruptions usually last no longer than a few minutes, they can degrade the level of service significantly on a short headway service. In this paper the authors propose a real-time disruption control model for Light Rail Transit Systems based on an integrated quantitative forecasting and regularization approach. The forecasting component relies on Artificial Neural Networks, a non-parametric computational model that has proved to be particularly efficient for the forecasting task in several applicative domains. The regularization engine involves the formulation of a constrained mathematical programming problem which can be solved quickly and therefore is well suited for real-time disruption control. The conceptual model is applied to a case study concerning the transit line number 7 operating in the urban area of Milan. To validate the proposed forecasting and regularization framework an experimental plan has been designed and performed under different traffic and passengers demand fluctuation conditions. The results of the simulation study witness the efficacy of the overall approach to forecast and regularize the considered Light Rail Transit System.

KEYWORDS: intelligent light rail train; Feed-forward Neural Networks; headway variance minimization; real-time regularization; discrete event dynamic simulation.

1 This work is part of the “iLRT - intelligent Light Rail Train” Research Project (Bogni et al., 2000) coordinated by Project Automation S.p.A. and co-funded by the Italian Ministry of Research (MIUR) within the framework of the Law n° 46/82.
INTRODUCTION

Light Rail Transit Systems (LRTSs) are becoming more and more important for offering efficient and reliable mobility alternatives to private cars and bus transportation within urban areas. Indeed, despite economic reasons energy saving together with environmental issues is driving both public and private companies, responsible for the management of the urban mobility network, to deal with innovative and intelligent transportation systems. Such transportation systems must be capable of providing high transportation capacity as well as high transportation quality together with low environmental impact and adequate power management.

Nowadays, LRTSs are considered to be the most promising answer to fulfill the above mentioned requirements and a great deal of effort has been devoted to study, analyze, design and implement efficient and reliable examples of such a technological approach to urban transportation. It is well known that transit operations in urban areas are characterized by frequent disturbs due to many factors such as signal delays at intersections, unexpected congestion states caused by high traffic volume eventually degenerating to incidents and last but not less important passenger demand fluctuations at stations. All these disturbance sources strongly increase the complexity of managing transit operations in the sense that they make it very complex to maintain transit headway adherence (Chien et al., 2001; Ding and Chien, 2001) which is considered the most important measure of effectiveness when considering LRTSs. Indeed, riders, having longer waiting times than their expectation, will be discouraged to use the transit system and consequently will resort to other solutions such as personal cars.

Intelligent transit control strategies, capable of adjusting transit operations in real-time maintaining headway regularity and reducing average passenger waiting times, are today the main challenge connected to the implementation of LRTSs. Scientific literature suggests that the
central measures of effectiveness, for evaluating control strategies include stochastic variations of system performance (e.g., headway and schedule variances) and average passenger waiting times (Koffman, 1978; Chien et. al, 2001; Ding and Chien, 2001). Operation control strategies for rail transit systems discussed in the literature can be classified in two categories (Vuchic, 1981):

- manual control,
- automatic control.

The first control category assists drivers that usually operate vehicles under the guidance of signal systems to keep safe spacing between pairs of successive vehicles, while the second category involves vehicle operations instructed from a centralized computer system, while drivers are only functioned to start automatic driving processes or to deal with emergencies such as vehicle malfunctions and breakdowns.

In this paper the authors focus their attention on automatic control strategies to cope with heavy ridership and high frequency that usually characterize LRTSs. The proposed real-time control system aims to minimize the headway variance and average passenger waiting times and is based on a centralized computer that offers real-time reliable forecasts for arrival times at each station. The forecasting module is based on a powerful and flexible computational approach implemented by Feedforward Neural Networks (FNNs). Arrival time forecast represents the core component of the proposed control system to guide operating vehicles based on real-time information, such as vehicle locations and delays, obtained from transit monitoring and communication systems. In this study the real-time control system tries to maintain preplanned headways between pairs of consecutive vehicles. In particular, the headway discussed in this paper refers to the elapsed time duration between the departure times of consecutive vehicles.
leaving from a station. Therefore, the control system is assumed to adjust vehicle departures in real-time based on their optimal arrival time at the next station subject to constraints on the maximum attainable speed and the headways to its leading and following vehicles.

The contribution of the paper is threefold; first the FNN, used to forecast traveling time, is different from those proposed in the specialized literature. In fact, the input variables are different from those described by the papers already published on the subject (Najjar et al., 2000; Chien et al., 2002; Abdulhai et al., 2002). Second, the paper investigates the interplay between forecasting and regularization systems by empirically showing that these two systems, when jointly used, allow a significant improvement of the regularity. It is worthwhile to notice that this is theoretically ensured only in the limit of the forecasting error going to zero. Third, the numerical experiments have been organized and performed by exploiting the stochastic simulation framework. Indeed, a set of independent simulation runs has been performed and the obtained results, for the two main performance measures, i.e. headway variance and passenger waiting time, are summarized through theirs respective average values and confidence intervals. This is not the case for many papers previously published on the same subject which do perform a full set of simulation runs but only on a single run.

The proposed approach to forecast and regularization is validated through a set of simulated numerical experiments concerning the transit line number 7 that operates in the Milan urban area.
CONCEPTUAL MODEL

The conceptual model (Figure 1), of the proposed forecasting and regularization system, consists of the following components:

- **TRAIN LOCATION DATABASE**: contains real-time train location data, originated from transit to line checkpoints (stops and intersections), collected by means of inductive loops,
- **MODEL REPOSITORY DATABASE**: maintains the available forecasting models (FNNs) associated with each transit line,
- **TRAIN TIMETABLES DATABASE**: contains information concerning timetables associated with each transit line,
- **FORECASTING MODULE**: is responsible for forecasting the arrival time at each station for each train traveling on any given transit line,
- **REGULARIZATION MODULE**: uses the input from the forecasting module together with timetable data to real-time compute the optimal control action and to communicate it to train drivers.

![Figure 1](image_url). Conceptual model of the forecasting and regularization system.
The computational components of the conceptual model are the forecasting and the regularization modules consisting of several elements that collaborate to achieve two of the main research project goals (Bogni et al., 2000).

As soon as a train is approaching a checkpoint, station or intersection, the associated inductive loop originates the automatic insertion of the corresponding record into the Train Locations DB (service request). This new record originates the invocation of the Forecasting Module that is responsible for activating and querying the appropriate FNN model. The FNN model exploits actual position (checkpoint) of the train, the available historical data (data related to the delay or anticipation of the leading train with respect to its scheduled travel time) as well as covariate data, i.e. those data that cannot be influenced but only recorded (month, weekday, daytime, weather conditions, train type, …), to provide the forecast of the arrival times for all those stations located downstream of the actual checkpoint. The vector containing the arrival time forecasts is sent to the Regularization Module that combines it with the corresponding timetable data allowing the comparison of the arrival time forecast to the scheduled arrival time provided by the Train Timetables DB. The results of such comparison are used to formulate the mathematical problem for the headway optimization (Problem Formulation) that is subsequently solved by means of a dedicated optimization engine. The optimal regularization solution is then communicated to the train driver of the train that originated the service request. According to the optimal regularization solution the train driver is required to delay or speed his/her departure time from the current station thus trying to maintain preplanned headways between pairs of consecutive vehicles.

The conceptual model relies on a compact and coded representation of the transit line, train and FNN forecasting models. In particular, each transit line is described by means of an
ordered list of checkpoints that can be stations, road intersections or trains depots. Formally, a transit line is described with an ordered list of checkpoints:

\[(cp_1, cp_2, \ldots, cp_n)\]  

(1)

The transit line is defined in such a way that trains start from checkpoint \(cp_1\) run to \(cp_2\) and so on to arrive at the final checkpoint \(cp_n\). Each checkpoint \(cp_j\) is univocally identified by a code and maintains information about its typology, i.e. station, road intersection or depot. Each train, operating on a given transit line, is uniquely identified by means of a code maintained in the Train Locations DB. Finally, each FNN devoted to the arrival times forecast is uniquely identified by means of the FNNid depending on the transit line and checkpoint identifiers.

The forecasting and regularization system is persistent, therefore all FNN models, in which the quantitative information is stored in the Models Repository DB, are loaded into central memory and wait for an invocation from the Forecasting Module. The FNN invocation is performed as soon as a train approaches a checkpoint and signals its location to the Train Locations DB through the inductive loops system. The FNN invocation consists of the following:

\[(Tid, TLid, cp)\]  

(2)

where:

- \(Tid\) represents the identifier associated with the train requiring the forecasting service,

- \(TLid\) is the identifier of the transit line where the train \(Tid\) is actually traveling,

- \(cp\) is the code associated with the checkpoint, belonging to transit line \(TLid\), where the train \(Tid\) is approaching.
Once a given FNN model is invoked it computes the forecast of the arrival time for each checkpoint belonging to \((c_{p_1}, c_{p_2}, \ldots, c_{p_n})\) and downstream the actual checkpoint \(c_p\). The forecast is communicated to the Regularization Module that automatically activates the headway variance minimization problem formulation and solution through a dedicated optimization engine. The optimal control solution, i.e. the waiting time associated with the pair (Train, Checkpoint), is sent to the driver of the train that originated the service request.

**ANNs AND ARRIVAL TIME FORECAST**

The forecasting module is the core component of the proposed real-time control system. Its performance, ability to provide fast and reliable arrival time forecasts, is fundamental to the proper working of the proposed regularization approach. Indeed, the capability to correctly anticipate the immediate future evolution of train locations allows to control in real-time each transit line by exploiting the anticipative paradigm that can capture complex traffic dynamics and allows the evaluation of cyclic congestion behaviors.

Amongst several alternatives, including linear regression models, time series models and Kalman filters, the choice of the computational paradigm, for implementing the Forecasting Module, has been oriented towards the non-parametric approach offered by Artificial Neural Networks (ANNs) (Hertz et al., 1991; Kung, 1993). Indeed, ANN models have already been successfully applied to transportation forecasting problems as described in (Najjar et al., 2000; Abdulhai et al., 2002).

In particular, Kalaputapu and Demetsky (1995) reported that, on a basis of several quantitative investigations, ANNs appear to be superior to traditional statistical techniques, i.e. the Box-Jenkins model, when dealing with travel time forecasting. Furthermore, in Chien et al.
(2002) the authors have shown the effectiveness of ANNs to forecast transit arrival times by combining two artificial neural networks trained by means of link and stop based data.

The choice of the specific ANN model, to develop for the travel time forecasting problem, has been performed following what has been presented and discussed in Smith and Demetsky (1994). Indeed, we decided to investigate and exploit the potential of the ANN model called Feedforward Neural Network to solve the travel time forecasting problem. Feedforward Neural Networks (FNNs) are computational models which have proved to be useful and effective for solving several classes of problems (Fu, 1982; Ripley, 1996; Poggio and Girosi, 1990).

The main property of FNNs is that they are universal approximators in the sense described by White (White, 1989). This property establishes that FNNs, with as many as one hidden layer, are capable of approximating with any degree of accuracy any mapping between independent variables and response variables such as the size of the data set, describing the Data Generating Process (DGP), and the number of neurons of the hidden layer go to infinity.

A FNN (Figure 2) is a layered structure consisting of computing units named artificial neurons (circles) and connections between neurons called synapses (directed links). Artificial neurons, or simply neurons, of the input layer are associated with independent variables, namely input variables \(x_1, x_2, ..., x_n\); while the output neurons are associated with response variables, namely the output variables \(y_1, y_2, ..., y_m\). Synapses are oriented connections linking neurons from the input layer to neurons of the hidden layer and neurons from the hidden layer to neurons of the output layer.

The information flows from neurons of the lower layers to neurons of the higher layers cannot flow between neurons of the same layer or between higher layers to neurons of lower layers.
The choice of the FNN architecture, i.e. number of hidden layers and number of neurons for each hidden layer, strongly influences the performance of this computational device and several quantitative approaches have been proposed to cope with it (Hassibi and Stork, 1993; Thimm and Fiesler, 1995; Silani and Stella, 2001). Given the FNN’s architecture, the network learns the mapping, from the input variables to the output variables, through the process called network learning consisting of two separate and subsequent computational steps, namely network training and network testing. During network training sample data are presented to the FNN which adapts its parameters, synapses weights and neurons biases, to minimize a given error function. The network training can be accomplished by means of several iterative algorithms such as back-propagation (Hertz et al., 1991; Gaivoronski, 1994), conjugate gradient, Gauss-Newton or Levenberg-Marquardt (McKeown, 1975; McKeown et al., 1997). After the network training step has been accomplished it is required to estimate the generalization ability of the trained FNN, i.e. its capability to provide reliable forecasts for previously unseen data. To this extent a new dataset, is used through a quantitative step called network testing. In the case of
the precision estimate, for the trained FNN, when the FNN model is satisfactory, it is saved and retained for further usage while on the contrary a new FNN’s architecture is trained and the network learning process continues until a satisfactory solution is obtained.

The proposed FFNs based Forecasting Module (Figure 3) consists of the following software components:

- **TRANSIT LINE MAPPER:** using data from the Lines and Paths databases it provides, for each transit line, the corresponding ordered list of checkpoints \((c_{p1}, c_{p2},\ldots, c_{pn})\).

- **TRANSIT ENQUIRER:** is activated from the Transit Line Mapper and it is responsible for recovering the historical data concerning the selected transit line, i.e. the historical data associated with the given ordered list of checkpoints,

- **DATASET COMPILER:** elaborates the historical data provided by the Transit Enquirer and compiles the corresponding dataset, i.e. input and output variables, to be used during the FNN learning phase,

- **DATA PREPROCESSOR:** elaborates the dataset obtained from the Dataset Compiler to preprocess input and output variables. This component implements normalization and transformation of discrete variables that cannot directly be used by FNNs,

- **DATASETS BUILDER:** generates, for each checkpoint associated with a given transit line, the corresponding dataset to be used during the FNN learning phase,

- **DATASETS SPLITTER:** partitions the dataset, obtained by means of the Dataset Builder, associated with each checkpoint in two mutually disjoint subsets called training and testing or validation datasets,

- **LEARNER:** implements the FNN training step, associated with each checkpoint, using the Levenberg-Marquardt iterative algorithm together with the Empirical Model Building approach (Silani and Stella, 2001) for selecting the optimal architecture of the FNN,

- **VALIDATOR:** implements the testing or validation step associated with each FNN trained using the Inducer software component,
- **MODEL EXPORTER**; saves the trained and tested FNNs to the Models Repository DB,
- **MODELS LOADER**; when activated loads all FNN models, maintained in the Models Repository DB, into the central memory of the forecasting and regularization system,
- **INDUCER**; is invoked in real-time from the Transits DB each time a train reaches any given checkpoint belonging to its transit line. Furthermore, the Inducer is also responsible for the activation of the Regularization Module that will be described in the next section.

![Diagram of the FNNs based Forecasting Module](image)

**Figure 3.** FNNs based Forecasting Module.
The Forecasting Module assumes that each FNN model is associated with a given checkpoint belonging to a transit line. In particular, for each pair (transit line, checkpoint) the corresponding FNN model uses the following input variables:

- Month,
- Weekday,
- Daytime,
- Weather Conditions,
- Train Type,
- Delay/anticipation of the lead train with respect to its scheduled travel time, i.e. difference between the scheduled travel time (maintained in the Train Timetables DB) and the real travel time (maintained in the Train Locations DB) for the last train that reached the next checkpoint with respect to the checkpoint of the pair (transit line, checkpoint),


to forecast the travel time (expressed in seconds) of the train, actually located at the checkpoint associated with the given FNN model, to reach the downstream checkpoints. Therefore, each FNN model uses a fixed number of input variables to obtain forecasts related to a variable number of output variables depending on the transit line definition (ordered list of checkpoints).

The proposed approach to forecast is computationally intensive during the FNN model learning. Indeed, the training and testing steps are time intensive even if they are accomplished periodically and in an off-line setting. However, while the training and testing phases are computationally intensive the real-time forecast is obtained very quickly. Indeed, FNN models compute their output in polynomial time with respect to the input dimension. The proposed framework to forecast is particularly well suited to exploit the available historical data recorded during the operations of the transit line. It allows for the efficient capture of complex dynamics
associated with any given transit line. To this extent, each FNN model is periodically submitted to maintenance (i.e. re-training) depending on the result of a statistical test that compares the FNN “certified performance”, i.e. the performance of the FNN obtained from the “Validator” software component, to the FNN “historical performance”, i.e. the performance of the FNN model during its daily working.

Finally, for any given transit line, the forecasting framework works properly when the transit line is operating under “normal conditions”. In particular, in the case when an unexpected event occurs (e.g. accidents), at a given checkpoint, the FNN models associated with all the upstream checkpoints are not queried further until the unexpected event has been managed and solved. This policy has been implemented because, within the iLRT research project, all the unexpected events, i.e. accidents, emergencies, …, are managed by means of a dedicated set of software components related to the “Safety Module” which is not described here for reasons of brevity.

REAL-TIME HEADWAY REGULARIZATION

The real-time regularization module relies upon the quantitative formulation of the headway variance minimization problem described in Ding and Chien (2001).

In order to clarify the mathematical structure of the regularization model let us introduce the following notation. Let \( h^{(i)}_{k,i} \) be the headway between train “k” and its leading train “k-1” for checkpoint “i” at time “t”. Thus, \( h^{(i)}_{k,i} \) can be formally expressed as follows:

\[
h^{(i)}_{k,i} = p_{k,i} - p_{k-1,i}
\]

where \( p_{k,i} \) and \( p_{k-1,i} \) represent respectively the departure time, from checkpoint “i”, of train “k” and the departure time, from checkpoint “i”, of its leading train “k-1”.

\[ (3) \]
Notice that \( p_{k,i} \) is either known, in the case when train “k” has already left checkpoint “i”, or can be computed according to the following equation:

\[
p_{k,i} = a_{k,i} + d_{k,i}  
\]

(4)

where \( a_{k,i} \) represents the arrival time of train “k” at the checkpoint “i” while \( d_{k,i} \) is the dwell time, for train “k”, associated with checkpoint “i”. The arrival time \( a_{k,i} \) is either known, because train “k” has just reached checkpoint “i” or can be estimated using forecasting models with the same consideration applying to the dwell time \( d_{k,i} \). In particular, within the framework of the research project, it is expected that in the future the dwell time \( d_{k,i} \) will be forecasted by exploiting images provided by means of video cameras operating at each station “i”. Therefore, at this stage of the research project the dwell time \( d_{k,i} \) is assumed to be defined, as described in Ding and Chien (2001), by the following equation

\[
d_{k,i} = t_b \lambda_i^{(t)} (a_{k,i} - p_{k-1,i})  
\]

(5)

where \( t_b \) is the average boarding time for a single passenger, \( \lambda_i^{(t)} \) is the hourly passenger arrivals at station “i” and \( (a_{k,i} - p_{k-1,i}) \) represents the elapsed time between the arrival time of train “k” and the departure time of its leading train “k-1” at station “i”.

Along the transit line each checkpoint is characterized by a specific service demand, i.e. number of waiting passengers. Therefore, the proposed mathematical model should allow for appropriate assessment of the impact of passenger demand on the overall value of the measure of effectiveness to be optimized. Ding and Chien (2001) propose each checkpoint “i” to be associated with a relative importance weight \( w_i^{(t)} \), i.e. a real number that represents the relative importance of checkpoint “i” at any given time “t”. Notice that at each time “t” the relative
importance weights are normalized to sum to one for each transit line (ordered list of checkpoints). In particular, Ding and Chien (2001) suggest the exploitation of the estimated hourly passenger arrivals $\lambda^{(t)}_i$ at time “$t$”, at each station “$i$” and to define the corresponding relative importance weight for station “$i$” as follows:

$$w^{(t)}_i = \frac{\lambda^{(t)}_i}{\sum_{j=1}^{S} \lambda^{(t)}_j} \quad i = 1,2,\ldots,S$$

with $S$ being the number of stations associated with the transit line.

According to notations and definitions the implemented real-time regularization module minimizes, whenever train “$k$” is ready to depart from station “$i-1$”, the total headway variance:

$$\min \sum_{j=1}^{S} \sum_{k=2}^{N} \frac{w^{(t)}_j \left( h^{(t)}_{k,j} - H^{(t)}_{k,j} \right)^2}{N - 1}$$

s.t.

$$L^{(t)}_{k,i} \leq e^{(t)}_{k,i} \quad k = 1,2,\ldots,N \quad i = 1,2,\ldots,S$$

where $N$ represents the number of trains, numbered from 1 to $N$ (from downstream to upstream), traveling along the transit line, $H^{(t)}_{k,j}$ is the scheduled headway, at time “$t$”, between train “$k$” and train “$k-1$” for station “$j$”, $e^{(t)}_{k,i}$ is the decision variable (optimal arrival time, at time “$t$”, at station “$i$” for train “$k$”), while $L^{(t)}_{k,i}$ represents the earliest arrival time for train “$k$” at station “$i$” for the actual time “$t$”. The earliest arrival time is defined as the departure time for train “$k$” from station “$i$” plus the minimum travel time from station “$i-1$” to “$i$” that has been determined, according to transportation experts, based on several factors as: space between station “$i-1$” and station “$i$”, train tractive force and acceleration/deceleration rates as well as maximum operating speed allowed between station “$i-1$” and station “$i$”. Therefore, constraints (8), one for each pair of checkpoint and train, are technological in the sense that they ensure that at any given time “$t$”
any given train is not assumed to travel at a speed higher than what is really possible or allowed along the considered transit line.

The optimization problem (7, 8) consists of minimizing, at time “t”, the overall headway variance along the considered transit line. Indeed, the objective function is a sum, over all trains traveling at time “t” along the transit line, of the squared difference between the estimated headway $h_{k,j}^{(t)}$ and the corresponding scheduled headway $H_{k,j}^{(t)}$ obtained from the transit line timetables. In particular, the scheduled headway $H_{k,j}^{(t)}$ depends on several factors such as the month, the weekday and the specific time window during the day journey. However, in normal operating conditions the scheduled headway $H_{k,j}^{(t)}$ is the same ($H_{k,j}^{(t)} = H$) across all trains concerned in the objective function (7). Notice that each term of the sum over N in (7), is weighted by means of its own relative importance weight $w_j^{(t)}$ as previously described. Finally, the decision variable $e_{k,i}^{(t)}$, for the optimization problem (7) and (8), represents the optimal arrival time for train “k” at checkpoint “i” while equations developed to estimate train arrival and departure times at station “i” are reported, according to Ding and Chien (2001), through Table 1, where train “k” is assumed to depart from station “i-1”.

<table>
<thead>
<tr>
<th>Train Number</th>
<th>Arrival Time</th>
<th>Departure Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$a_{1,i}$</td>
<td>$p_{1,i}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>k-1</td>
<td>$a_{k-1,i}$</td>
<td>$p_{k-1,i} = a_{k-1,i} + t_b \lambda_i^{(t)}(a_{k-1,i} - p_{k-2,i})$</td>
</tr>
<tr>
<td>k</td>
<td>$e_{k,i}^{(t)}$</td>
<td>$p_{k,i} = e_{k,i}^{(t)} + t_b \lambda_i^{(t)}(e_{k,i}^{(t)} - p_{k-1,i})$</td>
</tr>
<tr>
<td>k+1</td>
<td>$a_{k+1,i}$</td>
<td>$p_{k+1,i} = a_{k+1,i} + t_b \lambda_i^{(t)}(a_{k+1,i} - p_{k,i})$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>N</td>
<td>$a_{N,i}$</td>
<td>$p_{N,i} = a_{N,i} + t_b \lambda_i^{(t)}(a_{N,i} - p_{N-1,i})$</td>
</tr>
</tbody>
</table>

Table 1. Equations for estimating train “k” arrival and departure times at station “i”.
It is worthwhile to notice that the arrival and departure times in Table 1 are either known, because the train (Train Number) has arrived at or departed from station “i” before the actual time “t”, or are forecasted by means of the corresponding FNN models according to the forecasting framework. If at the current time “t” train “k” is ready to depart from station “i-1” its optimal arrival time at station “i”, denoted by $e_{k,i}^{(t)}$, is the decision variable to be optimized for minimizing the objective function (7) subject to constraints (8). In particular, departure times $p_{k,i}$ ($k = 1, 2, ..., N$), reported in Table 1, allow to compute, by substitution into (3), the headway variance at station “i” and thus the total headway variance (7) can be computed. The regularization module formulates and solves the variance headway minimization problem (7, 8) following the real-time vehicle control model depicted in Figure 4.

![Diagram](image)

**Figure 4.** Configuration and application of the real-time control.
THE TRANSIT LINE NUMBER 7

The quantitative performance of the proposed forecasting and regularization framework has been studied and validated in the case when transit line number 7, operating in the north-east area of Milan (Figure 5), is considered. This transit line travels from the Testi depot to the Gilardi Mattei station and back from the Gilardi Mattei station to the Testi S.Monica station thus involving a total of 12 stations. The case study relies upon a simulation model (Sanchini, 2003), for the transit line, which has been implemented following the Discrete Event Dynamic Systems (DEDS) paradigm. The choice of the simulation model approach, for validating the proposed forecasting and regularization framework, is mandatory because transit line number 7 is still under development. In particular, the development and validation, of the forecasting and regularization framework, has to cope with the following practical limitations:

- Several hardware components, devoted to the detection of train arrivals and departures at and from stations, are not physically installed yet,
- The communication system responsible to transmit data, from the hardware components already installed along the transit line, to the operating center is currently under testing,
- The central database management system, devoted to maintain all transits data, is actually in the design phase.

Figure 5. Transit line number 7.
The simulation study is concerned with trains traveling from Monday to Friday in the well known problematic traffic conditions, i.e. from 07:00 a.m. to 09:00 a.m. Indeed, in such a time window it is known that different traffic conditions are present from low flow to highly congested traffic states. Notice that line number 7 is characterized by several road intersections, with and without traffic lights, as well as one downgrade/upgrade located between Stz. Greco and Arcimboldi Ateneo Nuovo.

The network learning process has been performed by exploiting data originated from 504 simulation runs of the DEDS simulation model (Sanchini, 2003). In particular, the performed simulation runs, devoted to accumulate the FNN learning data, concerned three weeks of journeys characterized by different months, weekdays, weather conditions, traveling trains types and passengers demands. In order to clarify the structure of FNNs used by the Forecasting Module the FNN model associated with the Rodi Testi station is reported in Figure 6.

![Figure 6](image-url)
The FNN model performance, associated with the forecasts of travel time from Rodi Testi to Stz. Greco, is reported in Figure 7. In particular, Figure 7 (a) shows the real travel time versus the forecasted travel time associated with observations belonging to the FNN training set while Figure 7 (b) shows the real travel time versus the forecasted travel time associated with data points belonging to the test set, i.e. those data points that have not been used during the FNN learning phase. The split of the available data set into the train and test datasets has been performed following the hold-out testing procedure and by using 2/3 of the available data, generated from the DEDS simulation model, for the FNN training while the remaining 1/3 has been exploited to estimate the FNN out of sample performance.

The out of sample performance of the FNNs trained to forecast the trains travel time, between subsequent stations for the considered transit line, is satisfactory. Indeed, the out of sample percent error never exceeds 3.5%, in absolute value.

Figure 7. Travel time, Real Vs FNN forecasted, from Rodi Testi to Stz. Greco.
The FNN model associated with station “i-1” allows to forecast for each train “k” the departure times from downstream stations $p_{k,i}$, by manipulating the input variable “Daytime” that allows to estimate how delaying the departure of train “k” at station “i-1” affects the departure times of train “k” from downstream stations.

The validation of the proposed forecasting and regularization framework has been performed through a numerical experiment concerned with the last generation of Light Rail Trains called “Eurotram Series 7000” whose main characteristics are reported in Table 2.

<table>
<thead>
<tr>
<th>Maximum Passengers</th>
<th>262</th>
<th>Maximum Speed (full load)</th>
<th>70 Km/h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>34105 mm</td>
<td>Acceleration at Start (full load)</td>
<td>1 m/sec$^2$</td>
</tr>
<tr>
<td>Height</td>
<td>3189 mm</td>
<td>Service Deceleration (full load)</td>
<td>1.2 m/sec$^2$</td>
</tr>
<tr>
<td>Net Weight</td>
<td>371887 N</td>
<td>Emergency Deceleration (full load)</td>
<td>2.2 m/sec$^2$</td>
</tr>
<tr>
<td>Full Load Weight</td>
<td>648095 N</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 2.** Eurotram Series 7000 – main characteristics.

Additional information and a more detailed description concerning the “Eurotram Series 7000” can be found at [http://www.cliccaitalia.it/teb/tranvie_mi/eurotram/eurotram,0,0303.htm](http://www.cliccaitalia.it/teb/tranvie_mi/eurotram/eurotram,0,0303.htm).

The numerical experiment is concerned with the Eurotram Series 7000 trains traveling from Monday to Friday from 7:00 a.m. to 9:00 a.m. from the Testi depot to the Testi S.Monica station with a planned headway of 4 minutes. The average passenger boarding time is assumed to be 1.7 seconds while the number of passengers arriving at each station is assumed to be a random variable distributed according to a Poisson distribution. According to (5), the Poisson parameter $\lambda^{(t)}_i$ for each station “i” of the transit line, except for the Testi depot where no passengers are present, is reported in Table 3. In particular, the Poisson parameter $\lambda^{(t)}_i$, i.e. the average number of passenger arriving in one minute at station “i” during the time stripe “t”, has been estimated by transportation experts, and has been specified for different time stripes of 20 and 10 minutes from 07:00 a.m. to 09:00 a.m.
The dynamics of the trains, i.e. the trains travels from a station to the following one as well as the trains stops at each station for the passengers boarding, from the Rodi Testi station (station number 1) to the Testi S.Monica station (station number 11) and for each of the 16 simulation runs are illustrated through Figure 8(a) and Figure 8(b) (no real-time control) and Figure 9(a) and Figure 9(b) (the proposed forecasting and regularization framework is active). In particular, the station numbers, in accordance with Table 3, are reported on the X axis while the simulation time, measured in seconds, is reported on the Y axis. The number of trains, starting from station 1 and reaching station 11, during the simulation time ranges, simulation run by simulation run, from 22 to 23. Furthermore, the number of trains contemporaneously traveling from station 1 to station 11, i.e. traveling together on the transit line, ranges between 7 and 8.

From Figure 8 ((a) and (b)) it is possible to observe how subsequent trains tend to form “platoons” especially in correspondence to the Arcimboldi Ateneo Nuovo station (station number 6) and to the Università Bicocca Scienza station (station number 8). However, a severe deviance, of the real headway, from the scheduled headway frequently occurs at the Stz. Greco station (station number 3).
Figure 8(a). No Real-Time Control: Trains’ dynamic (X=station number, Y=simulation time in seconds) from Rodi Testi to Testi S.Monica. (Simulation runs from 1 to 8).
Figure 8(b). No Real-Time Control: Trains’ dynamic (X=station number, Y=simulation time in seconds) from Rodi Testi to Testi S.Monica. (Simulation runs from 9 to 16).
Figure 9(a). **Real-Time Control**: Trains’ dynamic (X=station number, Y=simulation time in seconds) from Rodi Testi to Testi S.Monica. (Simulation runs from 1 to 8).
Figure 9(b). **Real-Time Control:** Trains’ dynamic (X=station number, Y=simulation time in seconds) from Rodi Testi to Testi S.Monica. (Simulation runs from 9 to 16).
From Figure 9 ((a) and (b)) it is possible to observe how, when the forecasting and regularization framework is active, the trains platooning effect is consistently reduced even if some moderate deviations of the real headway from the scheduled headway still occur with specific reference to the most downstream stations from the Testi depot (station number 0), i.e. the Rodi Testi station (station number 9), the Testi station (station number 10) and the Testi S.Monica station (station number 11).

The comparison of the trains dynamics in the case when no real-time control is employed (Figure 8 (a) and (b)) with the trains dynamics in the case when the proposed forecasting and regularization framework is active (Figure 9 (a) and (b)) witnesses in favor of the latter. Indeed, the proposed forecasting and regularization framework is capable of efficiently managing stochastic variations induced by travel times as well as frequent passenger demand fluctuations occurring, along the transit line number 7, from 7:00 a.m. to 9:00 a.m.

The temporal evolution of the Weighted Headway Variance (7), related to the 16 simulation runs, is depicted in Figure 10. In particular, the green line is related to the weighted headway variance (Y axis in secs²) evolution with respect to the simulation time (X axis in secs) when the forecasting and regularization framework is active while the red line describes the behavior of the weighted headway variance without the real-time control. Figure 10 shows that the weighted headway variance in the case when the proposed forecasting and regularization framework is active is always lower than the weighted headway variance in the case when the real-time control is not employed. In particular, as soon as the transit line becomes fully operative, i.e. the first train reaches station number 11 (after 1800-1900 seconds), the weighted headway variance without real-time control becomes significantly greater than the weighted headway variance obtained with the use of the proposed real-time control.
Figure 10. Weighted headway variance Vs simulation time (16 simulation runs). (The X axis is associated with the simulation time, in seconds, while the Y axis is associated with the weighted headway variance in secs²).

The temporal evolution of the average passenger waiting time (Y axis in seconds) with respect to the simulation time (X axis in seconds) is depicted in Figure 11. The green line is still associated with the application of the forecasting and regularization framework while the red line is associated with no real-time control. The comparison between the average passenger waiting time (Figure 11) when no real-time control is employed and when the forecasting and regularization framework is active, for each simulation run, witnesses the efficacy of the proposed forecasting and regularization framework. Indeed, for each simulation run, the average
passenger waiting time, when the real-time control system is active, is significantly lower than the average passenger waiting time obtained when no real-time control is employed.

![Graph showing average passenger waiting time vs simulation time](image)

**Figure 11.** Average passenger waiting time Vs simulation time (16 simulation runs). (The X axis is associated with the simulation time, in seconds, while the Y axis is associated with the average passenger waiting time in seconds).

The behavior of the mean value of the weighted headway variance with respect to the simulation time, computed across the 16 simulation runs, is depicted in Figure 12. In particular, the continuous green line (circles) is associated with the proposed forecasting and regularization framework while the continuous red line (squares) is associated with no real-time control.
Furthermore, Figure 12 also shows the 90% confidence intervals for the *mean value of the weighted headway variance* (black dashed lines around the continuous green and red lines). Figure 12 shows that the *mean value of the weighted headway variance* in the case when the real-time control is active is significantly lower that the *mean value of the weighted headway variance* in the case when no real-time is employed. Indeed, the 90% confidence intervals never overlap during the simulation time.

![Graph showing mean value of weighted headway variance vs simulation time](image)

**Figure 12.** Mean value of the weighted headway variance vs simulation time.

The behavior of the *mean value of the average passenger waiting time* with respect to the *simulation time*, for the 16 simulation runs, is depicted in Figure 13 by adopting the same
graphical notation as in Figure 12. This figure shows that, along the simulation time, the *mean value of the average passenger waiting time* in the case when the real-time control is active (circles) is always lower than the *mean value of the average passenger waiting time* in the case when no real-time control is employed (squares). Furthermore, along the simulation time the 90% confidence interval for the *mean value of the average passenger waiting time* in the case when the real-time control is active, never overlaps with its counterpart associated with no real-time control. Therefore, we are allowed to conclude that the proposed forecasting and regularization framework significantly reduces the *mean value of the average passenger waiting time*.

![Graphical Representation of Figure 13](image)

**Figure 13.** Mean value of the average passenger waiting time vs simulation time.
To understand how the real-time control affects the *average passenger waiting time* for each station, its mean value, for each of the 16 simulation runs, has been computed and graphically reported by exploiting the box plot graphical representation. The box plot graphical representation, for the mean of the average passenger waiting time, in the case when no real-time control is employed and in the case when the forecasting and regularization framework is active are reported respectively in Figure 14 and in Figure 15.

Figure 14 shows that the lowest value of the *mean of the average passenger waiting time* is achieved at station number 3 while the highest one is associated with station number 11. Furthermore, from this figure it is possible to observe how the *mean of the average passenger waiting time* drastically increases from station number 8 to station number 11, i.e. it increases from upstream to downstream stations.

![Box plot of the mean value of average passenger waiting time vs station (no control).](image-url)
Figure 15 when compared with Figure 14 shows that the proposed forecasting and regularization framework significantly reduces the mean of the average passenger waiting time for all the stations along the considered transit line. Furthermore, the uncertainty about the mean of the average passenger waiting time is also reduced with respect to the case when no real-time control in employed. However, the greatest value for the mean of the average passenger waiting time is still achieved at station number 11 that is the most downstream station while the lowest value is achieved at station number 3 and at station number 8 that are those stations where most passengers board during the simulation time as reported in Table 4.

![Box plot](image)

**Figure 15.** Box plot of the mean value of the average passenger waiting time vs station (control).

<table>
<thead>
<tr>
<th>STATION NUMBER</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
</table>

**Table 4.** Average passengers demand during the simulation time.
CONCLUSION AND FURTHER DEVELOPMENTS

In this paper the authors described an integrated forecasting and regularization framework Light Rail Transit Systems based on FNNs and headway variance minimization. The paper presents the conceptual model of the forecasting and control system and describes its main modules, namely the forecasting module, based on FNNs, and the real-time control module, based on headway variance minimization. The paper also presents a case study concerning the LRTS operating in the city of Milan related to transit line number 7.

The results of the performed numerical experiments support the relevance and effectiveness of the overall approach and furthermore show how the proposed conceptual model of the real-time control policy can significantly improve the two major measures of effectiveness when considering LRTS, namely the headway variance and the average passenger waiting time.

The research activity is actually in progress and in the next months the tuning and the validation phases, in which the integrated forecasting and regularization framework will be evaluated on the real transit line number 7, are expected to start.

ACKNOWLEDGEMENTS

The authors acknowledge the precious contribution of Dr. Andrea Sanchini (Sanchini, 2003), who designed and implemented the DEDS simulation model of transit line number 7 as well as the precious help of Matteo Cecchinello (Cecchinello, 2003) who performed the planned numerical experiment. The authors are grateful to the anonymous referees whose comments and criticisms significantly contributed to improve the quality of the paper. The authors are indebted with Prof. Hickman for his precious help and assistance during the referring process.
REFERENCES


McKeown, J.J. (1975) Specialized versus general-purpose algorithms for minimizing functions that are sums of squared terms. Mathematical Programming, 9: pp. 57-68.


