



12th International Conference on Design and Decision Support Systems in Architecture and Urban Planning, DDSS 2014

## Real time traffic models, decision support for traffic management

Luc Wismans<sup>a,b,\*</sup>, Erik de Romph<sup>a,c</sup>, Klaas Friso<sup>a</sup>, Kobus Zantema<sup>a</sup>

<sup>a</sup>*DAT.Mobility, Snipperlingsdijk 4, 7417 BJ Deventer, The Netherlands*

<sup>b</sup>*University of Twente, Faculty of Engineering, Centre for Transport Studies, P.O. Box 217, 7500 AE Enschede, The Netherlands*

<sup>c</sup>*Delft University of Technology, Faculty of Civil Engineering and Geosciences, Department of Transport & Planning, P.O. Box 5048, 2600 GA Delft, The Netherlands*

---

### Abstract

Reliable and accurate short-term traffic state prediction can improve the performance of real-time traffic management systems significantly. Using this short-time prediction based on current measurements delivered by advanced surveillance systems will support decision-making processes on various control strategies and enhance the performance of the overall network. By taking proactive action deploying traffic management measures, congestion may be prevented or its effects limited. An approach of short-term traffic state prediction is presented and implemented in a real life case for the city of Assen in the Netherlands. This prediction is based on connecting online traffic measurements with a real time traffic model using the macroscopic dynamic traffic assignment model StreamLine in a rolling horizon implementation. Different monitoring data sources consisting of both fixed-point and floating car data are used. The advantage of the rolling horizon approach is that no warming-up period is needed for the dynamic traffic assignment taking less computation time while keeping results consistent. Further, the current traffic state estimation is done by combining model estimates of previous predictions and current measurements. The results of predictions made in the real life case are presented as well as several tested methods for improving the current state estimations showing promising results.

© 2014 The Authors. Published by Elsevier B.V This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

Peer-review under responsibility of the Eindhoven University of Technology, Faculty of the Built Environment, Urban Planning Group

*Keywords:* Real time traffic models; traffic management; Sensor City Assen, dynamic traffic assignment; rolling horizon; state estimation

---

---

\* Corresponding author. Tel.: +31(0)570 666 840; fax: +31(0)570 666 888.  
E-mail address: [lwismans@dat.nl](mailto:lwismans@dat.nl)

## 1. Introduction

In modern society, mobility is a basic human need and an important prerequisite for economic growth. Due to growing demand and difficulties to match supply, recurrent and non-recurrent congestion are part of daily traffic. As a result, traffic problems arise for society related to accessibility and livability. In the past, these problems were often solved by increasing the capacity of the road network. Due to limited resources, both financial as available space, and because of increasing technical possibilities, many European and other countries in the world invest nowadays in dynamic traffic management measures to better utilize our existing road network.

Dynamic Traffic Management (DTM) measures are roadside or in-car measures, which vary over time. These measures are used to inform road users (e.g. providing route information using variable message signs) or controlling traffic streams (e.g. metering traffic using traffic signals). These DTM measures are part of the broader class of intelligent transport systems (ITS) measures. The invention of the first traffic signal already took place in the 19th century and controlling traffic was then relevant to guarantee safety on intersections. Although safety issues are still reasons to implement traffic signals, these first DTM measures evolved to instruments that improve accessibility on a local level. At the end of the 20th century new measures were introduced, mainly on highways, as a result of the information technology revolution (e.g. variable message signs (VMS), rush hour lanes and ramp metering). New possibilities will only increase further in the coming years due to developments in vehicle-to-vehicle and vehicle-to-infrastructure communication. Three levels of deployment of DTM measures can be distinguished. On an operational level, decisions are made by traffic operators or fully automatic in real time applications on the settings of the DTM measures, based on the current or short term predicted traffic conditions. On a tactical level, decisions are made by traffic engineers on the realization and usage of DTM measures for specific traffic conditions (i.e. scenarios) by providing a tactical framework. On a strategic level, decisions are made by policy makers on the deployment of DTM measures to achieve certain policy objectives<sup>1</sup>. In practice, network approaches are (optimal) control strategies for corridors in which (similar) measures are coordinated, based on measuring the current traffic situation (e.g. coordinated ramp metering on corridors or approaches coordinating traffic signals). The decisions on operational level are therefore only made on available traffic measurements (mainly loop detector data). This means that the decisions are mainly reactive or indirectly proactive dependent on the available prepared scenarios. In addition, the decisions are based on an incomplete state estimation (i.e. only information is available for measured locations).

Traffic management and therefore utilization of the existing road network can improve significantly if there is a complete state estimation of the total road network and even further if a reliable and accurate short-term prediction is available. Using this short-time prediction based on current measurements delivered by advanced surveillance systems will support the decision-making processes on various control strategies and enhance the performance of the overall network. For example, if congestion is predicted at certain locations within the next hour, traffic measures can be taken in advance (i.e. proactively). These may include roadside measures like ramp metering, lane opening or closure, traffic light regulation, but also (personalized) in-car information like in-car navigation advice (smart routing). By taking proactive action, congestion may be prevented or its effects limited. Real time traffic models can be of great help in this kind of decision-making.

In the Dutch municipality Assen, a large national R&D project on sensor networks has been conducted. The main goal was to improve traffic flow and environment in the city centre and surrounding region, by developing real-time intelligent traffic management systems making use of different sensing sources of real-time traffic data including loop detector and floating car data. For this purpose, a real-time traffic estimation and prediction model has been developed and deployed. This model combines sensor data, the Streamline dynamic traffic assignment (DTA) model<sup>2</sup>, state estimation in a rolling horizon approach. The model uses an estimate of an origin-destination (OD) matrix combining historic OD-patterns and current traffic counts. This OD matrix is assigned dynamically to the model network using the rolling horizon approach. The advantage of the rolling horizon approach is that no warming-up period is needed in the dynamic traffic assignment taking less computation time, while keeping results consistent. Further, the current traffic state estimation is done by combining model estimates of previous predictions and current measurements. In this paper we will present the architecture and implementation for the Assen region of the real time traffic model. We show results of the validation process comparing predictions and observations and the results of several tested methods for these estimations.

### 1.1. Structure

In Section 2 we will give background information on the Sensor City Assen project and real time traffic models. Section 3 will describe the architecture of the real-time model. Results of the real time model are presented in Section 4 and the methods tested to improve the state estimation procedure can be found in Section 5. Finally, the conclusions and recommendations are described in Section 6.

## 2. Background

### 2.1. Sensor City Assen

Sensor City Mobility is a striking innovative mobility project with one main goal: to facilitate travelers with a personal travel advice so that they can easily choose the most comfortable and smart way of traveling. The project aims at a revolution in traffic and travel information services through smarter use of data from sensor technology. The Dutch City of Assen has become a 'living lab' in the past few years where a large-scale practical experiment has been conducted. In this experiment hundreds of travelers tried new in-car services and smart phone apps. This project was implemented by a consortium of 14 parties consisting of companies, research institutions and government. The project ended in April 2014, but the living lab in Assen is available for further research goals on mobility and also on other domains like sound, smart lighting, etc. Also the real time model system we developed is still continuously running and will be further developed.

In Klunder et al.<sup>3</sup> a sensor network design was described to determine the best possible sensor locations. This design was based on the static transport model of the city of Assen in such a way that (1) it captures a maximum number of OD pairs and trips; (2) it is located at major decision points or road sections; and (3) it avoids too much overlapping with adjacent sensors. Based on local knowledge of the network situation and their limitations at some locations this technically derived network design was somewhat adjusted and in 2011 a start was made with implementing this sensor network.

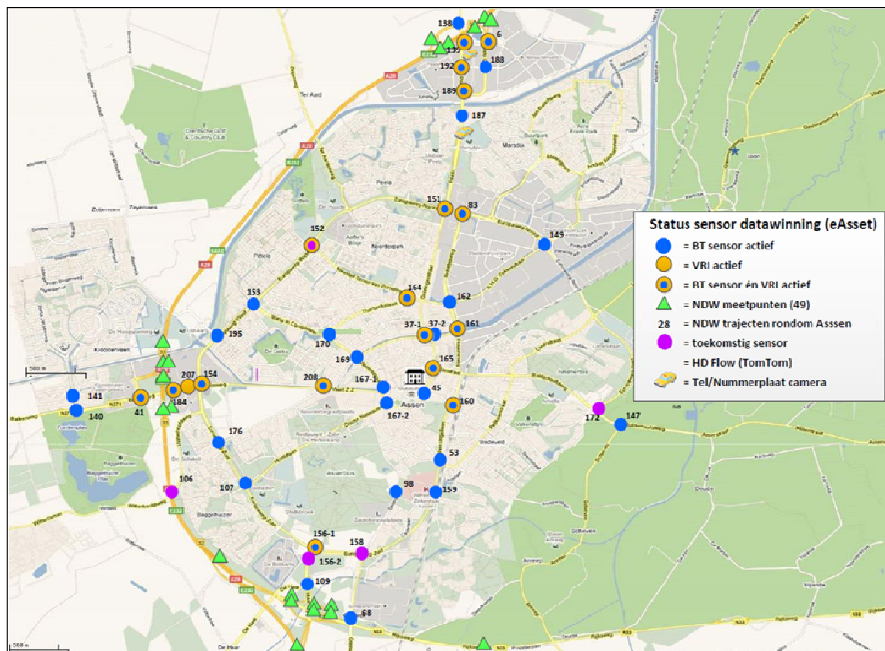


Fig. 1. Sensor network Assen.

In Figure 1 an overview of the sensor network in Assen is shown. The sensor network consists of more than 150 sensor nodes like traffic lights, cameras (for counting and for automatic number plate recognition as well), highway loop detectors and Bluetooth sensors. Furthermore, Floating Car Data will become available from navigation devices and OBU (On Board Units). From these devices different kind types of data become available like traffic flows, speeds, travel times and OD pairs. The real-time traffic model uses the traffic flows as input and the other types of data as validation tools. Important aspect in the data handling is the filtering of unreliable measurements. For example, in this urban area where with their mobile phones are also on the road or parallel cycling paths and will also be measured by Bluetooth sensors. These measurements are filtered out of the dataset.

## 2.2. Real-time traffic models

Before presenting the implemented real-time traffic model in a real life case, we will discuss comparisons and differences with other real-time systems available. A basic comparison is made with Aimsun Online and PTV Optima based on available literature.

The architecture of Aimsun Online<sup>4</sup> is comparable with our framework. The most important difference in the short term forecasting is that in Aimsun Online a pattern recognition module is used to select the best suitable OD-matrix from an historical OD-matrix database, where we use a matrix calibration module. In our opinion it is quite a challenge to possess a historical database, which includes all possible day-to-day variability and therefore it is more appropriate to use a matrix calibration procedure.

The idea of the OPTIMA architecture<sup>5</sup> is comparable with our approach. Also a matrix calibration procedure is used to estimate the current OD-matrix using traffic counts. The dynamic network loading is similarly based on kinematic wave theory (macroscopic flows) apart from the fact that OPTIMA used turn splitting rates where our approach uses route flows. The disadvantage of using splitting rates in a dynamic model is that it isn't ensured that all trips in the OD-matrix will reach their real destination.

## 3. Architecture real time traffic model

The real-time traffic model continuously monitors the current traffic status in Assen and makes a forecast of the expected traffic situation for the next half hour and the next hour. The model is updated every 5 minutes for the prediction of half an hour period and separately on another machine every 10 minutes to predict the next hour ahead. The reason for this is that it is currently not possible to perform the prediction up to one hour ahead every 5 minutes due to computational limitations. The model forecasts in terms of expected traffic flows, travel times. Spare road capacities are used as input for the in-car services and smart phone apps. The on-line traffic model is built in OmniTRANS, the transport planning application designed for integrated modeling of multi-modal transport systems. The dynamic traffic assignment model used in OmniTRANS is called StreamLine, an approach based on the cell transmission model (CTM), which is a derivative of the LWR model of Lighthill and Witham<sup>6</sup> and Richards<sup>7</sup>. This framework is capable of integrating aspects like route choice effects, traffic management, blocking back effects and departure time choices.

### 3.1. Framework

The architecture of the on-line traffic model consists of three basic components, which are run subsequently to make the state estimation. The overview of the architecture is displayed in Figure 2. At first the current traffic counts and travel times are taken from the Real-time traffic database within the Sensor City Assen project. These traffic counts contain loop data on the motorway, as well as loop data at traffic lights and at camera locations. The travel times are calculated based on Bluetooth measurements and floating car data. At the moment only the traffic counts are used in the online model calibration and forecasting.

After the traffic counts have been collected the a-priori OD-matrices for the current day and time are picked up from a large collection of historically estimated OD-patterns saved in matrices. A matrix has been estimated for every 5 minutes of the week, resulting in a total of 2.016 base matrices. These base matrices have been estimated by combining a demand pattern and static demand matrices derived from the existing regional traffic model. The

demand patterns are determined from yearly household travel surveys by OViN (Onderzoek Verplaatsingen in Nederland)<sup>8</sup>. For the peak periods a different static matrix has been used than for the off-peak period, considering the different flow pattern between periods.

After selection of the a-priori matrix, this matrix is calibrated with the current traffic counts. The calibration procedure is an information minimization algorithm<sup>9</sup>. Based on the results, the calibration effect is calculated and partially applied on the matrices for the next 30 minutes to create the current traffic forecast matrices. The calibration effect is in fact the difference between the actual measured situation and the average situation. This relative effect is also progressed to the short term forecast. At the moment a large part of the calibration effect is taken along in the forecast matrices. Starting with 95% for the first 5 minutes and decreasing continuously to 70% for 30 minutes ahead. The a-priori matrices (i.e. historical OD patterns) are continuously updated every 5 minutes by adding the calibration effect partly (by 3%) to the a-priori matrix of the current time period, to improve the a-priori matrices. The updated a-priori will be used the next week as the best available start for the process at that time.

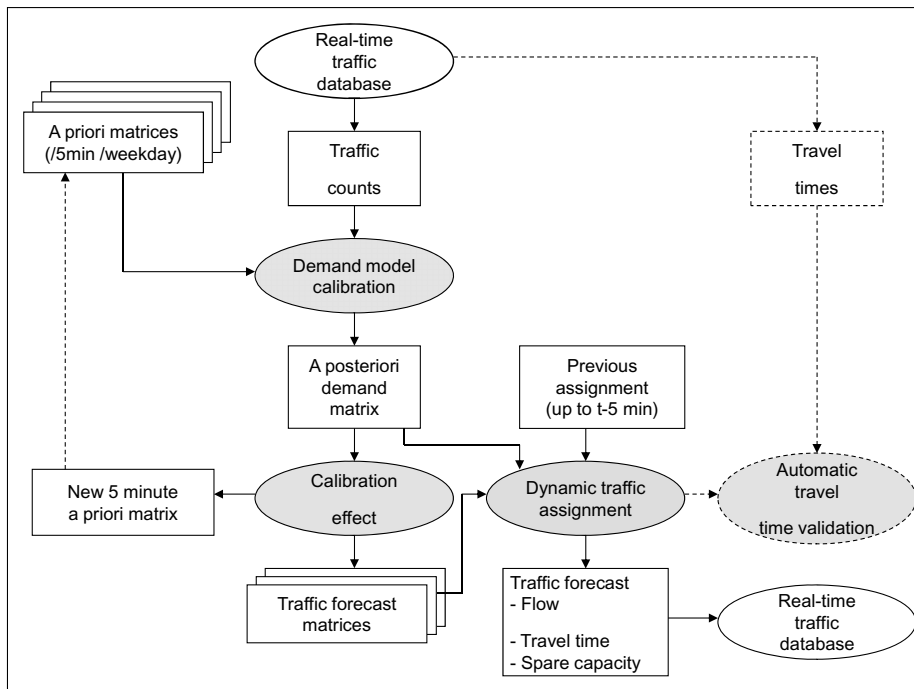


Fig. 2. Architecture on-line traffic model.

With the current and traffic forecast matrices ready, the dynamic traffic assignment is executed, resulting in traffic flows, speed, travel time and spare capacity on all links of the network. In order to keep the start-up phase of the dynamic assignment short, a rolling horizon method is used, which is described in more detail below. After the dynamic simulation the results are fed back to the real-time traffic database of Sensor City Assen, allowing other applications to use this additional information, for example for Smart Routing purposes.

### 3.2. Rolling horizon

Predicting the traffic's condition requires a realistically, dynamically loaded network. Normally a DTA simulation starts with an empty road network and traffic starts flowing from all zones into the network, based on the OD-matrices, routes, route choices etc. In order to obtain a realistic traffic situation for a prediction, the network has

to be loaded entirely, otherwise an unrealistic prediction would be made. In order to obtain a loaded network a normal DTA-simulation uses a so-called warm up period prior to the actual simulation period.

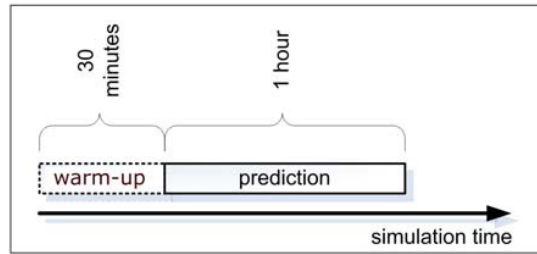


Fig. 3. Warm up simulation.

The duration of the warm-up period depends on the network’s size. In our case a 30 minute warm up period is required to obtain a fully loaded network prior to making the actual prediction of one hour. With an update interval of 5 minutes, this means that the same approach will be repeated every 5 minutes based on new measurements and OD matrices. The whole process needs to run within 5 minutes in order to be ready for the next interval.

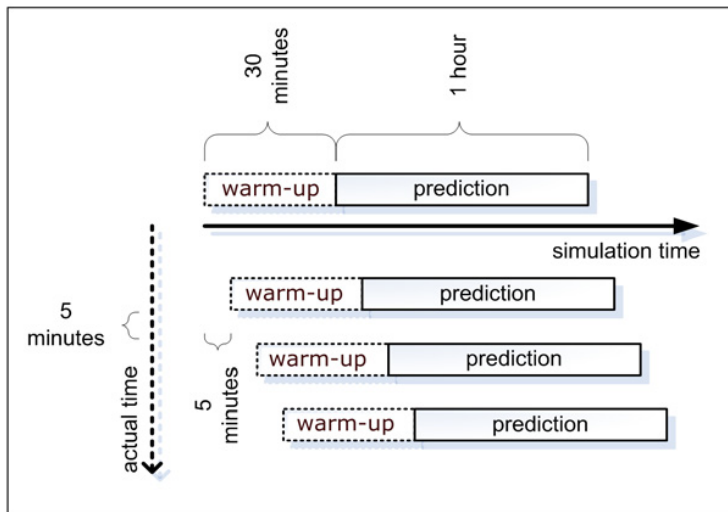


Fig. 4. Simulation process.

It needs no explanation that the approach above using a warm up period consumes considerably amounts of computing time just for obtaining a realistic prediction every 5 minutes. Observing the approach, one can argue that restarting the entire simulation including the warm up phase is not essential as long as it is possible to change the OD-matrices, routes and route-choice between each 5 minute-interval. In addition, an accurate current state estimation could improve the prediction. StreamLine has been adapted in such a way that it can store the simulation’s condition at any time at any segment during the simulation in a file. This situation represent everything the simulation knows about the condition on all links, link’s cells, speeds, spill back, junctions, zones etc for that particular time step.



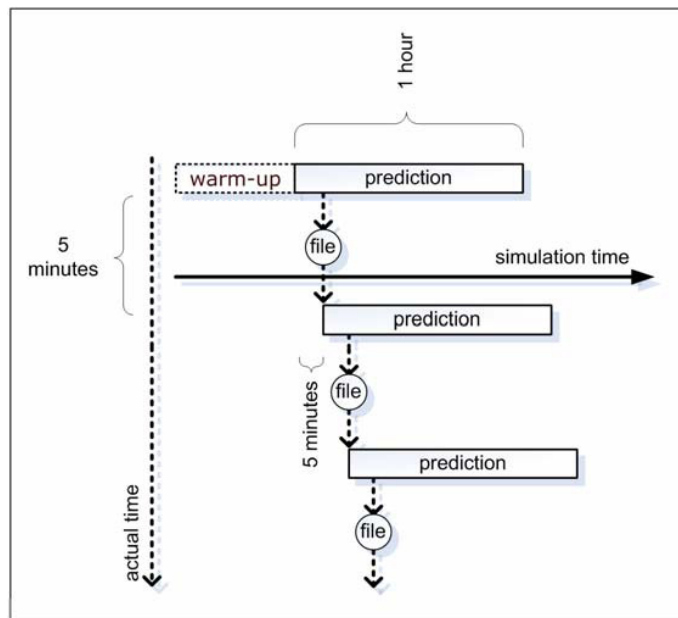


Fig. 5. Serialization to save computation time.

Using the stored internal situation in a file on disk, StreamLine can reproduce that particular situation simply by reading the file and configuring all objects internally as if the simulation never stopped. From that situation new OD-matrices, route choices etc. can be applied to the new simulation run. This process eliminates the warm up phase and starts off with a realistically, dynamically loaded network, i.e. a so-called hot start. This saves computing time and enables StreamLine to run within the 5 minutes update interval to satisfy the real time setting this application requires.

For the current state estimation (i.e. the starting conditions on every segment) several methods can be used. In the current application the previous prediction of the current state is used. However, this method can be further improved by using the current measurements as well. These methods will be described in Section 5.

#### 4. Real-time traffic model Assen

The framework presented in Section 3 has been used for the Sensor City Assen project. First the real life case was tested and calibrated off-line using real traffic data, of which results can be found in Friso et al<sup>10</sup>. One issue, which was presented in this earlier paper, concerned the smoothness of count patterns at locations of traffic lights. If these count values are directly used as input for the real time traffic model, the model has problems to follow these kinds of patterns and it results in very unstable model outcomes. The main reason for the unsmoothed pattern of these counts, lies in the fact of the cycle times of traffic lights. For example, if the cycle time is 2 minutes then the one period a branch can have had 3 green times with cars flowing, while the next period of 5 minutes it only has 2 green times. Therefore, we choose to perform a smoothening of the counts by not only taking the count value of the past 5 minutes into account but also from the 5 minutes before. By experimenting we obtained the most stable smoothening by using 60% of the smoothened count value of the period between 5 and 10 minutes ago and 40% of the actual counted value of the past 5 minutes.

Furthermore, a quality indicator of each count is calculated. If the quality of a count is not high enough it will not be used in the model calibration procedure and results in far more stable results. The quality indicator takes into account the previous quality, the counted value and the previous smoothened count value and is shown in formula 1.

$$\begin{aligned}
 Q_t &= \min(100, (Q_{t-1} + \alpha(1 - \left| \frac{(C_t - \overline{C}_t)}{\overline{C}_t} \right|))) \text{ if } (C_t > 0) \\
 Q_t &= \max(0, (Q_{t-1} - (100 - Q_{t-1}) - \beta \overline{C}_t)) \text{ if } (C_t = 0)
 \end{aligned}
 \tag{1}$$

where  $Q_t$  is the quality for the current time period,  $Q_{t-1}$  is the quality during the previous time period,  $C_t$  is the current actual count value and  $\overline{C}_t$  is the current smoothed count value. The  $\alpha$  and  $\beta$  are scale parameters and are set at 30 and 0.1 respectively for this experiment.

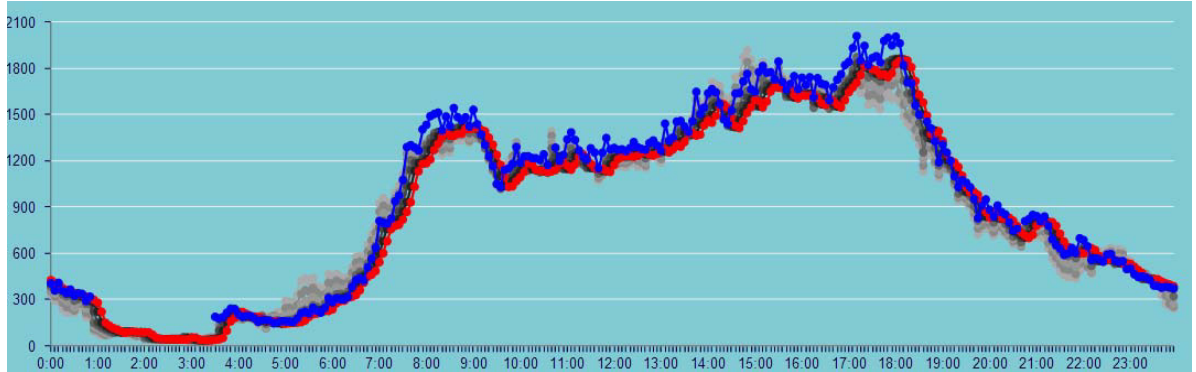


Fig. 6. Prediction versus measurements.

In 2013 the real time model went live, showing good results. Figure 6 shows an example of the model results on a specific link in the model network of Assen on the machine with the 5 minutes cycle. Because the model makes a prediction up to half an hour this means that 7 times an estimate is made: 6 times a prediction (30/25/..5 minutes ago and finally the last estimate of the first 5 minutes. Note that a prediction or estimate is related to the average state during a 5 minute period and therefore the last estimate is not completely the same as a current state estimation. Within all new predictions the OD-matrix is calibrated on the count values (but without correction in the state estimation (see Section 5)). Therefore, at each time stamp (every 5 minutes) 7 dots are presented. First of all, varying in 6 colors from light grey towards dark grey the predictions are presented. The red colored dot is the estimate after model calibration. The blue dots show the traffic counts that were available for this link. The figure shows that the model very well estimates the traffic situation during the day and that also the predictions are very close to the counts.

For a week (working days) in April 2014 the quality of the forecasts is analyzed. At 44 measured links the Root Mean Square Error (RMSE) and the Mean Absolute Percentage Error (MAPE) is determined for the predicted flow versus the measured flow for all time periods. In Table 1 the overall result is presented for the last estimate (which has been calibrated on the measurements) and the 6 previous predictions beforehand. For example, the RMSE for a link  $l$  at prediction time 30 minutes ago is defined as follows:

$$RMSE_{l,t=-30} = \sqrt{\sum_{i=1}^N (P_{t=-30,i,l} - M_{i,l})^2 / N}
 \tag{2}$$

where  $P_{t=-30,i,l}$  is the prediction of 30 minutes ago for time  $i$  at link  $l$  and  $M_{i,l}$  is the measurement at time  $i$  at link  $l$ . The overall RMSE is the average of the RMSE's of all considered links. All measurements and model flows are translated to hourly values by multiplying the 5 minutes flows by a factor 12. The average hourly flow at these measured links is equal to 429. The MAPE for a link  $l$  at prediction time 30 minutes ago is defined as follows:



$$MAPE_{i,t=-30} = \sum_{i=1}^N \frac{|P_{t=-30,i,l} - M_{i,l}|}{M_{i,l}} \tag{3}$$

The overall MAPE is the average of the MAPE's of all considered links.

Table 1. Overall RMSE and MAPE values at measured links (veh/hour).

Time (min)		RMSE	MAPE
t= -30	Prediction of 30 minutes ago	119	23%
t= -25	Prediction of 25 minutes ago	106	20%
t= -20	Prediction of 20 minutes ago	95	18%
t= -15	Prediction of 15 minutes ago	89	16%
t= -10	Prediction of 10 minutes ago	86	15%
t= -5	Prediction of 5 minutes ago	89	15%
t= 0	Prediction first 5 minutes (i.e. time interval 0-5)	90	16%

As Table 1 shows, and as may be expected, the result of the prediction slowly improves when the prediction horizon is closer by. But it is also clear that the quality of the prediction of 5 minutes ago is not extremely better than the prediction of 30 minutes ago. The results also indicate that the largest improvements can be found in improving the current state estimate, which in the current approach can be further improved by adjusting based on current measurements (see Section 5). Similar results are found if we analyze the peak period only, which means that the prediction of this often more important period for traffic management is of similar accuracy. Figure 7 shows the distribution of the MAPE's over the measured links are shown for the predictions of 30, 20 and 10 minutes before and the actual estimate. Because of readability of this figure the forecasts of 25, 15 and 5 minutes ago are not shown here. This figure shows that the distribution of the MAPE's moves to the left if the prediction time is shorter.

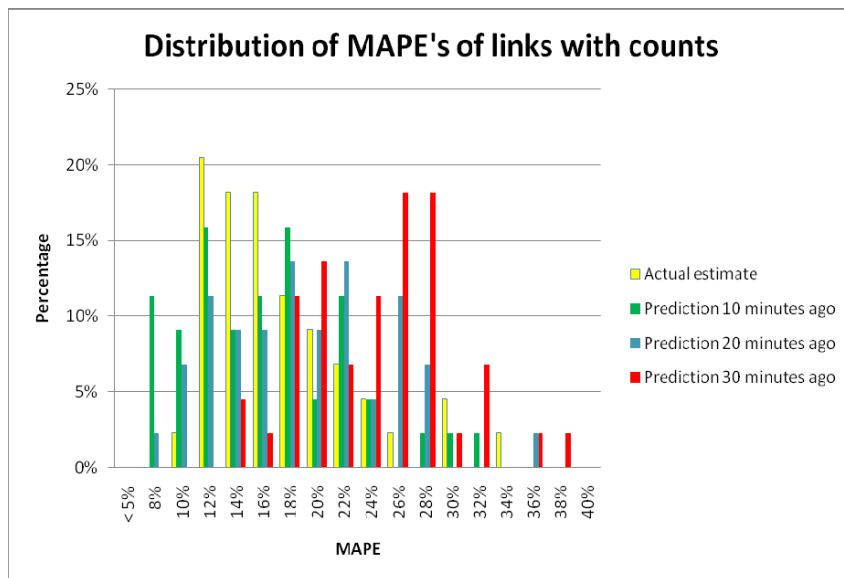


Fig. 7. Distribution of MAPE's.

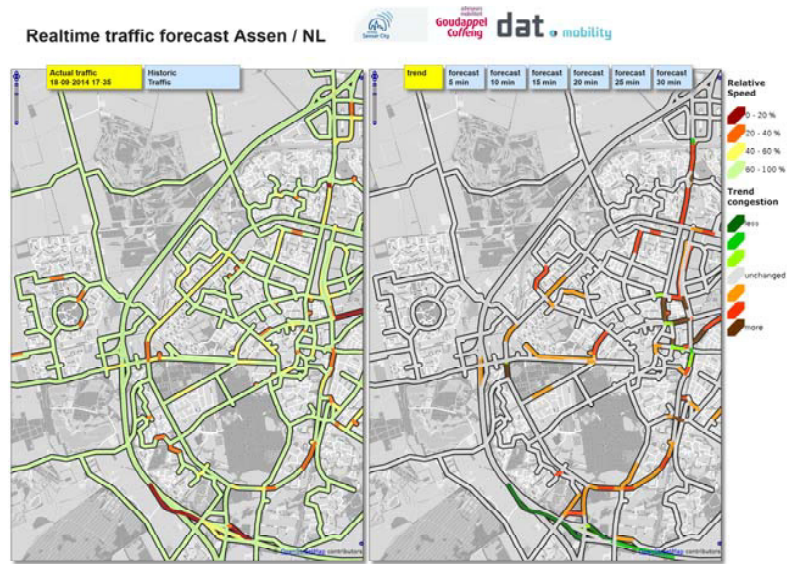


Fig. 8. Online viewer real time model.

An online viewer has been developed where the model results are presented in different ways. In Figure 8 at the left the current traffic state is presented by showing the relative speed (actual speed vs. free flow speed) on the whole network. This directly shows where the largest delays are at the actual moment. In the figure on the right the trend is presented where the green colored links show where it is expected that the travel speeds will increase in the next half an hour, where the red colored links show where the speeds will decrease. The user can also select to check the expected trend for other 5 minutes periods ahead.

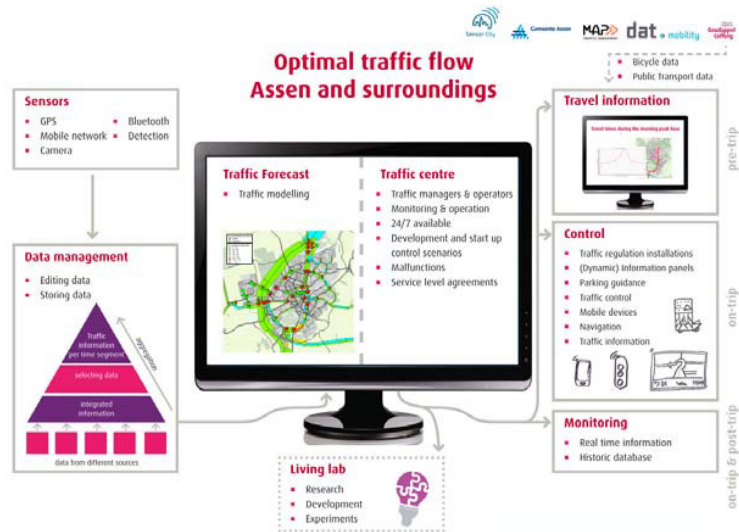


Fig. 9. Position of real time model in Sensor City Assen.

Figure 9 shows a complete overview of the position and the possibilities of the real time model in a setting for optimal traffic flows in urban areas. As shown on the right side of the figure the model can be used as a pre-trip information system, as well as on-trip or off-trip.

### 5. Improvements in state estimation

The accuracy of the prediction heavily depends on an accurate demand estimation and current state estimation, because possible errors in these estimations will be propagated through the network. The presented framework already contains algorithms to calibrate the a-priori demand matrices based on measurements. However, it takes time, especially for longer routes, until the calibration of the a-priori matrices improves the predictions, while only the inflows at origins are adapted. In Figure 10, this is illustrated for a simple network in which all link lengths are 1 km, the free speed is 35 km/h and there is one measured link (i.e. link 11). In this network the demand from A to V is in reality 500 veh/h, while the a-priori matrices assumed a demand of 200 veh/h. After calibrating the OD-matrix towards exactly 500 veh/h, we still find flow errors on links after 20 minutes in the simulation. Because in the current framework, the OD-matrix is calibrated every time interval, the errors are less. However, the example stresses the importance of an accurate current state estimation for real time traffic models used for short term prediction.

In Yuan (2013) an extensive literature review can be found on traffic state estimation. In our presented framework the current state estimation is fully based on the prediction made in the previous time step. In the OPTIMA framework<sup>11</sup> the current measurements are used to improve the state estimation. In their approach only the state of measured links are adapted. This means that the improved current state estimation of these links will be propagated through the network, which improves the predictions. However, the current state estimation can possibly be further improved by adapting upstream and downstream non-measured links as well. In our research, we investigated both the correction of measured links as well as non-measured links using exponential smoothing. In addition, the method was extended to incorporate non-measured links beyond junctions and the effect of two measured links near each other.

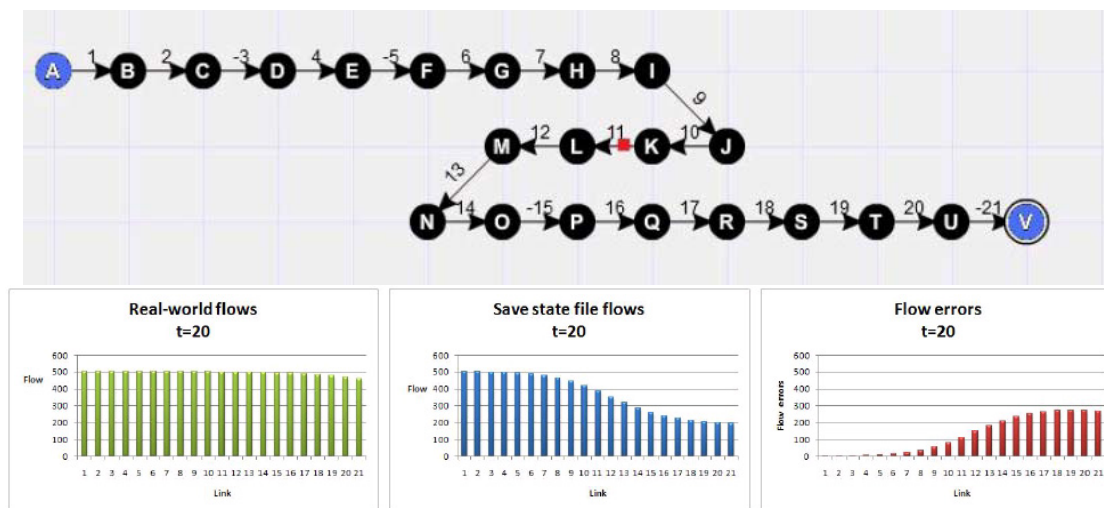


Fig. 10. Example of delay in improved prediction depending on OD calibration only.

5.1. Corrections on monitored links only

When performing the state estimation based on the current measurements of measured links only, this correction will propagate through the network improving the state in future time steps. In figure 11 the effect of correcting the flows on the measured link only is illustrated, by presenting the absolute errors for all links for different time steps. The reference situation is a framework in which there is no calibration of the a-priori OD matrix and no correction of current state estimation. The current framework presented in section 3 is with calibration of the a-priori OD matrix and no current state estimation. The improved state estimation is called Safe State File (SSF) manipulation.

At the start the reference situation shows obviously the same absolute error as the current framework and only the SSF approach shows an improvement at link 11, because the flow on this link is corrected for the measured flow. In future time steps the correction propagates downstream improving the state estimation of downstream links, but also the calibrated a-priori OD matrices improves the state estimation on upstream links starting at the origin.

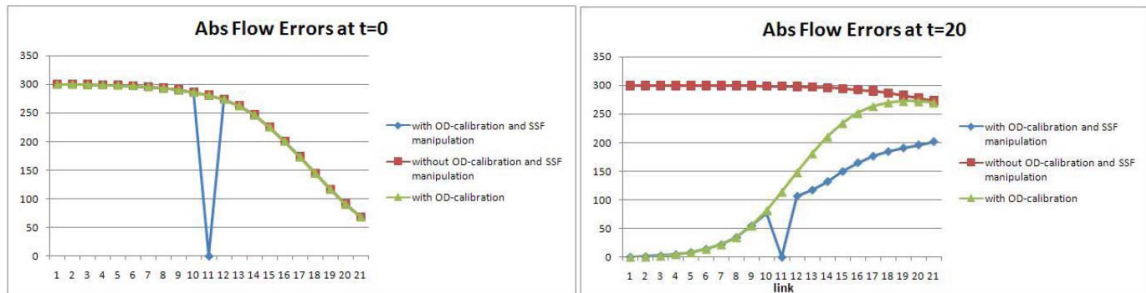


Fig. 11. Improvement in absolute error when correction of state estimation measured link (link 11).

5.2. Corrections on upstream and downstream links between junctions

If there is an error in the current state estimate of a certain link when compared with a measured link, it is safe to assume that there might be a similar error on links upstream and downstream of this link, as long as there are no junctions (i.e. locations at which traffic merges or diverges) involved. Therefore, we investigated the use of exponential smoothing to correct the flows on these links. The correction of a link is calculated using the following equation:

$$q_a^{new} = q_a^{org} + err_{a,m} \exp(-\alpha L_{a,m}^\beta) \tag{4}$$

in which  $q_a^{org}$  is the originally estimated flow of link  $a$ ,  $q_a^{new}$  is the corrected flow of link  $a$ ,  $err_{a,m}$  the absolute error between measured and originally estimated flow on link  $m$ ,  $L_{a,m}^\beta$  is the distance of link  $a$  to link  $m$  and  $\alpha$  and  $\beta$  are scale parameters. The absolute error is used, because using relative errors will result in wrong correction in methods presented in section 5.3 when we use exponential smoothing beyond junctions. The scale parameters determine to what extent the error measured at a certain link is also used to correct upstream and downstream links. Because the calibration of the OD matrix also improves the prediction, we distinguish two different sets for upstream and downstream links in which the corrections are lower when distance increases for upstream links than for downstream links. Using  $\alpha_{up} = 0.005$ ,  $\beta_{up} = 2.9$ ,  $\alpha_{down} = 0.01$ ,  $\beta_{down} = 2.4$  results in the absolute errors presented in figure 12 for the same network as used earlier.

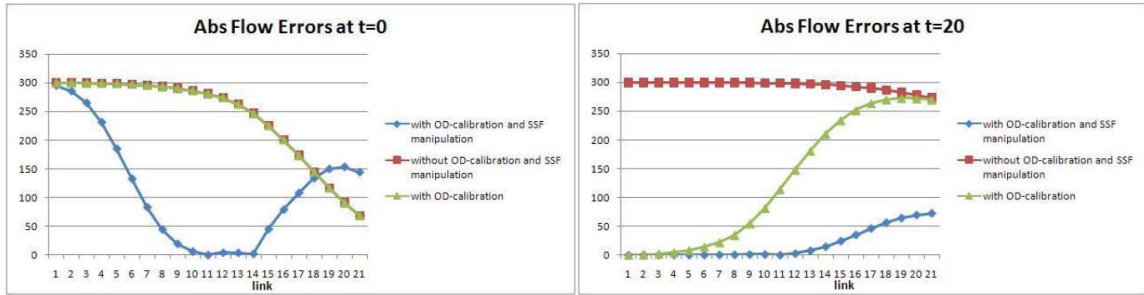


Fig. 12. Improvement in absolute error when correction of state estimation measured and non-measured links.

The results show significant improvements of the state estimation compared to only correcting the measured link.

### 5.3. Corrections with multiple measured links

When multiple links are measured on a stretch of road two situations can occur to estimate the initial state of a link. First, the measured links are both up or both downstream of the link. In that case the exponential smoothing suggested in equation 4 can still be used with the additional heuristic rule that only the nearest measured link is used. The second situation occurs when there are measurements both up and downstream of the link. In that case both corrections are weighted, using a weight depending on the distance to the link. If we assume that the measured link upstream of link  $a$  is link  $m$  and the measured link downstream of link  $a$  is link  $n$ , the corrected flow of link  $a$ ,  $q_a^{new}$ , can be calculated using the following equation:

$$q_a^{new} = q_a^{org} + \frac{L_{a,m}}{L_{a,m} + L_{a,n}} err_{a,m} \exp(-\alpha_{up} L_{a,m} \beta_{up}) + \frac{L_{a,n}}{L_{a,m} + L_{a,n}} err_{a,n} \exp(-\alpha_{down} L_{a,m} \beta_{down}) \tag{5}$$

Using the same network but now assuming an additional measured link (i.e. link 17), this method results in the following absolute errors.

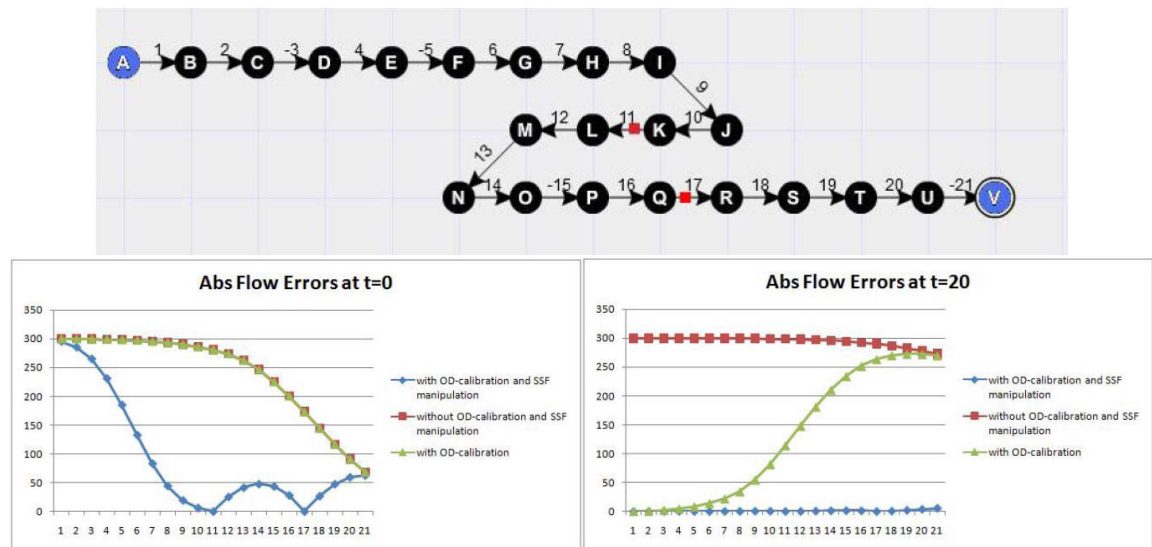


Fig. 13. Improvement in absolute error when correction of state estimation measured and non-measured links, multiple measurements.

If we compare the root mean squared error (RMSE) and largest absolute error (LAE) of the three presented methods (i.e. 1, correction of measured link, 2, correction of measured and non measured links and 3, usage of multiple measurements), the exponential smoothing shows significant improvements compared to only correcting the measured link.

Table 2. RMSE and LAE values between junctions.

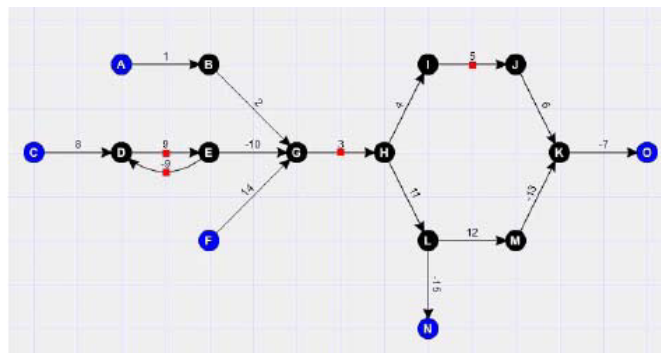
Time	Method	RMSE	LAE
t=0	1. Correction of measured link	243.9	299.9
	2. Correction of measured and non measured links	147.4	294.6
	3. Correction of measured and non measured links, multiple measurements	132.7	294.6
t=20	1. Correction of measured link	115.9	201.2
	2. Correction of measured and non measured links	32.1	72.6
	3. Correction of measured and non measured links, multiple measurements	1.6	5.6

#### 5.4. Corrections on upstream and downstream links beyond junctions

In real life urban networks there are many junctions in which traffic merges or diverges. In that case we can not simply correct upstream or downstream links beyond those junctions with the earlier presented equations, because this depends on the route choice of traffic. Therefore, the correction should be *path based* taking route choice into account. For this a correction term is introduced which equals the partial flow of link *a* which also uses the measured link *m*,  $q_a^{new}$  divided by the total flow of measured link *m*. This means that the corrected flow on link *a*, can be calculated using the following equation.

$$q_a^{new} = q_a^{org} + err_{a,m} \exp(-\alpha L_{a,m}^\beta) \frac{q_{a,m}^{org}}{q_m^{org}} \tag{6}$$

in which  $q_{a,m}^{org}$  equals the flow passing link *a* which also passes link *m* and  $q_m^{org}$  equals the total flow of the measured link. The quality of this correction obviously highly depends on the accuracy of the estimated route choice in a network. If we assume these to be correct, the results of correcting beyond junctions compared to only correcting results between junctions is presented in figure 14. For this case another network is used containing multiple junctions and a few measured links indicated with a red square.





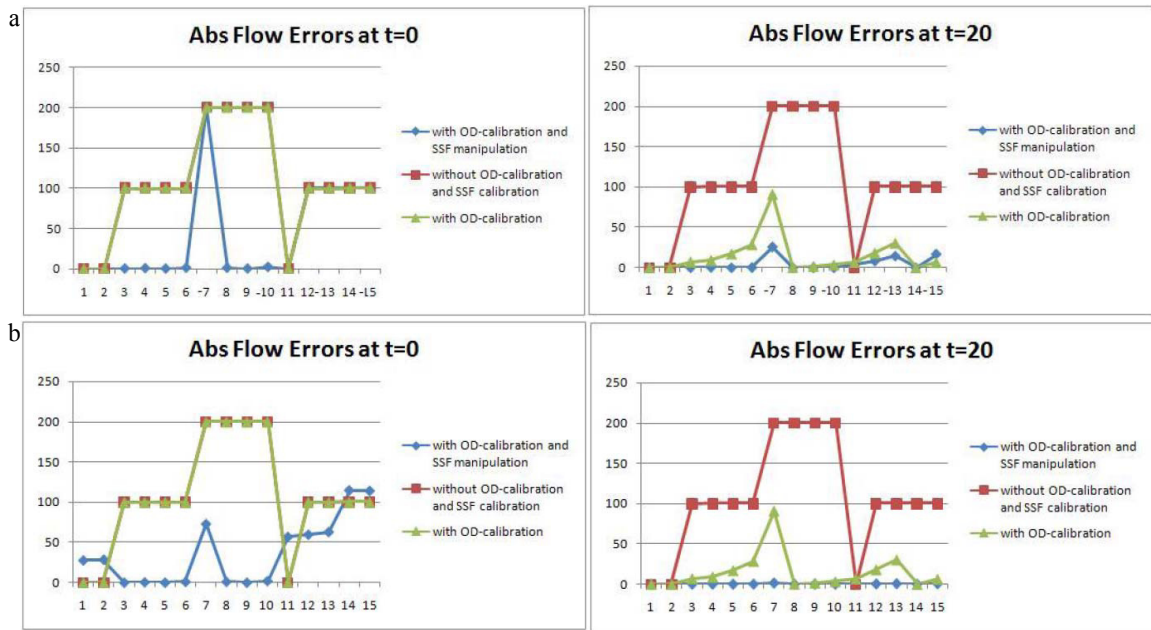


Fig. 14. Corrections in network, a) only between junctions, b) also beyond junctions.

The results in terms of root mean squared error (RMSE) and largest absolute error (LAE) of both methods are presented in table 3. In this example the method beyond junctions shows again a significant improvement. Although errors in route choice estimations in the real time model will have some impact on the predictions as well, these errors will also influence the propagation of corrections in state estimation if no smoothing would be applied beyond junctions.

Table 3. RMSE and LAE values between and beyond junctions.

Time	Method	RMSE	LAE
t=0	4. Correction of measured link and non measured links, between junctions	73.0	199.8
	5. Correction of measured link and non measured links, beyond junctions	53.8	114.2
t=20	4. Correction of measured link and non measured links, between junctions	9.0	25.6
	5. Correction of measured link and non measured links, beyond junctions	1.8	5.6

### 6. Conclusions and recommendations

Traffic management measures are deployed in many countries to better utilize the existing road network. The current state of practice is that decisions regarding this deployment are made reactively. This means decisions are based on available information of the current traffic state only. Significant improvement can be expected if those decisions could be made based on accurate predictions of *future* traffic states (i.e. pro-active). In this paper a framework is presented of a real-time traffic model, which can deliver predictions and is applied in a real life case in the Assen region within the “Sensor City Assen” project. Although the current approach shows promising results using a smoothing procedure for traffic counts near traffic lights, it is easy to demonstrate the importance of an accurate current state estimation when the approach uses a rolling horizon. In the current approach this state estimation is assumed to be the predicted state of that time-period in the previous prediction. Comparing different methods shows that using exponential smoothing to determine corrections of upstream and downstream links of measured links, improves the current state estimation significantly and as a result the prediction.

Further research is needed on testing these state estimation methods in the real life case, as well as the influence of errors in predicted route choice behavior and calibrated OD matrices. In addition, further calibration of model parameters, especially calibration effects applied on short-term prediction matrices as well as the a-priori matrices and usage of more available data to improve predictions. Next, improvements in the dynamic network loading model concerning for example dynamics in available capacity of the network (e.g. due to heavy rain fall or accidents), but also reducing computation times. Finally, the approach should be tested on more cases with different characteristics.

## Acknowledgements

We acknowledge the Stichting Sensor City for giving us the opportunity to work on this project and the city of Assen of providing us with real-time data. The Sensor City project is being facilitated by the European Union, the European Regional Development Fund, the Ministry of Economic Affairs, Agriculture & Innovation and The Northern Netherlands Provinces, KOERS NOORD.

## References

1. Wismans LJJ. Towards sustainable traffic management. *Doctoral Thesis*, University of Twente, 2012.
2. Raadsen MPH, Mein HE, Schilpzand MP, Brandt F. Implementation of a single dynamic traffic assignment model on mixed urban and highway transport networks including junction modeling. In: *Proceedings of the third international symposium on dynamic traffic assignment*. Takayama, Japan, 2010.
3. Klunder G, Chen YS, Zantema J, Friso K. Consistent traffic Modeling and calibration at different resolution levels based on various sensor data for real-time traffic management. In: *proceedings second MT-ITS conference*, Leuven, June. 22-24, 2011.
4. Aimsun online, [http://www.aimsun.com/wp/?page\\_id=33](http://www.aimsun.com/wp/?page_id=33), 2014
5. Gentile G, Meschini L. Using dynamic assignment models for real-time traffic forecast on large urban networks. In: *proceedings second MT-ITS conference*, Leuven, Jun. 22-24, 2011.
6. Lighthill M, Witham G. On Kinematic waves II: a theory of traffic flow on long crowded roads. In: *Proceedings of the royal society of London*. Part A 229 (1178), 1955, p. 281-345.
7. Richards P. Shock waves on the highway. *Operations Research* 4; **1956**: 42-51.
8. OViN, *Onderzoek Verplaatsingen in Nederland*, 2013, <http://www.cbs.nl/nl-NL/menu/informatie/deelnemers-enquetes/personen-huishoudens/ovin/doel/default.htm>
9. van Zuylen HJ, Willumsen LG. The most likely trip matrix estimated from traffic counts. *Transportation Research Part B: Methodological* 1980; **14**(3): 281-293.
10. Friso K, Zantema J, Mein E. On-line Traffic Modelling In Assen: The Sensor City. In *proceedings third MT-ITS conference*, Dresden, 2013.
11. Yuan, Y. Lagrangian multi-class traffic state estimation. *Doctoral dissertation*, Delft University of Technology, 2013