

Anticipatory Adaptation of Signalisation Based on Traffic Flow Forecasts within a Self-organised Traffic Control System

Matthias Sommer and Jörg Hähner

Organic Computing Group, University of Augsburg, 86150 Augsburg, Germany

Abstract. Autonomously adapting signalling strategies to changing traffic demand in urban areas have been frequently used as application scenario for self-adapting systems. Striving for the ability to cope with the dynamic behaviour of traffic and to react appropriately to unforeseen conditions, such solutions dynamically adapt the signalisation to the monitored traffic demands. The Organic Traffic Control (OTC) system is one of the most prominent representatives in this domain. OTC implements a multi-layered observer-/controller architecture. In this paper, we extend OTC's observer with a time series forecast component to create forecasts of future traffic developments for turning movements. These forecasts are then used to proactively adapt signalisation parameters. We demonstrate the benefit of the developed approach in terms of reduced travel times, and vehicle emissions within near-to-reality simulations of realistic traffic conditions from Hamburg, Germany.

1 Introduction

The vehicular traffic domain is a vivid research field, both for industrial and academic research institutions. Novel trends, such as self-driving cars [1], car-to-car communication [2], and traffic-adaptive control systems [3, 4], have the overall goal to optimise the existing traffic infrastructure towards a more efficient utilisation of road networks. However, we have to face certain negative impacts on the environment due to the increase of mobility, especially in urban areas. Consequently, this leads to an increase in pollution, a rising number of incidents, and an inefficient use of the transportation system.

Urban road networks are characterised by a large amount of signalised intersections in vicinity that need to be monitored and optimised. Due to the mathematical complexity, the online optimisation of a single signalised intersection is not feasible with analytical methods. Above all, the dynamic characteristics of traffic, the unpredictable behaviour of humans, and the highly complex dependencies between several streams throughout the road network make it an interesting and challenging field for self-adapting and self-organising solutions.

Traffic management systems can be divided into two categories: fixed-loop systems and adaptive systems. The first category relies on static time-based signalisation with a limited set of predefined signal plans for each signalised intersection. These signal plans are developed by traffic experts based on historic data. In consequence, fixed-time signal plans are not responsive to dynamic and

changing traffic demands, and become outdated over time. In contrast, adaptive systems alter the signal timing based on the observed state of traffic. The traffic demands are monitored in real-time by sensors along the road, and serve as input for the signal time optimisation at each intersection. Some traffic management systems react to the traffic measured in the previous interval [4] while others adjust the signalisation proactively to the estimated traffic demand in the near future [5]. This continuous adaptation leads to a better system performance, even in case of changing traffic demands.

We propose and evaluate a highly self-adaptive technique to relieve an engineer from complex, design-time-situated tasks related to signal plan design, forecast selection, and configuration. In this work, we use time series forecasts of traffic flows to proactively adapt the signal plans at signalised intersections to current and future demands. The forecasting process follows the idea of multi-model ensemble forecasting, combining individual forecasts of several independent forecast methods.

The remainder of this paper is structured as follows. First, we provide a brief overview of the related work in the fields of traffic control and short-term traffic forecasting. We move on, presenting OTC; a fully decentralised, self-organised traffic control system. Furthermore, we introduce our forecast module for time series ensemble forecasting. We then elaborate on our approach using short-term forecasts of traffic flow to adapt the green times of traffic lights. We present the evaluation of our approach that is based on near-to-reality

simulations of an intersection in Hamburg, Germany. We conclude with a summary of our findings.

2 Related work

Our approach to predictive traffic control introduced in this paper touches a number of topics of different research disciplines.

2.1 Traffic-adaptive systems

In the last decades, several traffic-adaptive systems have emerged. Traffic-adaptive control systems such as OPAC [6], SCATS [7], and SCOOT [8] adapt green times, cycle times, and/or phase sequences according to the recently monitored traffic demands. Often, the adjustment is executed every five minutes, respectively after one to three full cycles [5]. This description of the current traffic demands can be represented by the number of vehicles passing a sensor at the respective signalised intersection, the occupancy level of a detector, or the current queue length in front of a traffic light. These responsive adaptive systems [3] adjust signal timings based on recent data collected over a certain time span (several minutes or cycles) before implementing a timing change. This responsive process has the disadvantage that the adaptations lag behind current traffic demands [3]. In contrast, we try to omit this drawback by making forecasts of the short-term developments of traffic flow, and by adapting signal timings to future demands.

2.2 Short-term traffic forecasting

Making forecasts of future traffic developments is an important aspect of Intelligent Traffic Management Systems. Typically, forecast techniques take a time series of the recent traffic flows to forecast the estimated traffic conditions of the near future. A time series can be formalized as $ts = X_1, \dots, X_n$, where n determines the length of ts . Forecasting strives to look ahead the current value of ts and to predict the value m steps in the future X_{t+m} . There are also some techniques that rely on historical information (e.g. Daily Load Curves [9]) or a combination of both. Furthermore, there are advanced forecast techniques that allow to add more input variables such as traffic density, weather conditions, or data from adjacent roads (e.g. Kalman filters [10] or Artificial Neural Networks [11]). A collective consideration of several forecast models has proven to be a more powerful approach than just relying on one individual technique [12]. Consequently, we follow the idea of ensemble forecasting, combining the forecast of several independent techniques.

3 Architecture

This section presents the Organic Traffic Control (OTC) system for the self-optimising and self-adaptive control of signalised intersections [4]. Furthermore, we briefly introduce our forecast module for time series forecasting

[13] which we later use to make short-term forecasts of traffic flows of turning movements.

3.1 Organic traffic control

Most urban road traffic networks work with fixed-time signal plans defining the green times for each turning movement. A traffic engineer has to define a signal plan for each signalised intersection. This time-consuming task needs a fair amount of expert knowledge. Additionally, these static signal plans are prone to obsolescence due to changing traffic demands [14], and cannot offer ideal signal timing for every situation and traffic demand. Learning the best situation-to-signalisation mapping replaces the problem of which technique to prefer by the question of how to describe the particular traffic condition, and how to automatically learn this mapping.

The self-organised traffic control system OTC follows this idea. OTC creates and selects signal plans matching demands without human interaction, following a reinforcement learning approach. Fig. 1 illustrates the architecture of the OTC system.

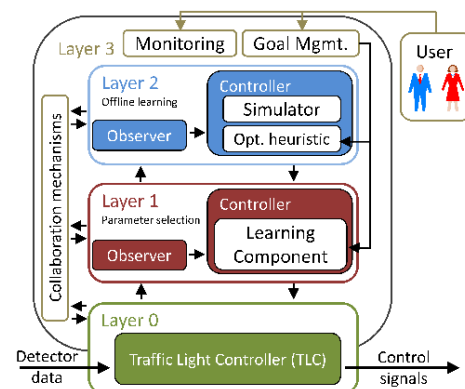


Figure 1. Multi-layered architecture of the OTC system.

An industry-standard, parametrisable traffic light controller (TLC) is extended by the OTC logic. The TLC is responsible for adapting phase durations (i.e. green times of traffic lights) according to the currently observed traffic conditions at the particular intersection. OTC works in fully decentralised mode – meaning that one individual instance of OTC deployed on a TLC controls one intersection only. Thereby, a safety-oriented self-learning concept is applied to allow for a continuous self-optimisation process, while simultaneously staying within controllable boundaries of system behaviour.

This optimisation is done in a two-step fashion. Layer 1 is responsible for the online learning process. The Layer 1 observer processes the input received from sensors at the local intersection. The observer passes a situation description of the current traffic conditions on to the Layer 1 controller which is represented by a learning component. Here, this component is realised as a learning classifier system. In case that no matching rule was found (rules resemble mappings of situation descriptions to signal plans), the offline learning process on Layer 2 is activated. Here, new rules (signal plans) are generated,

simulated, and evaluated. The plan with the lowest estimated average delay is returned to the controller on Layer 1. The simulation is executed by the traffic simulator Aimsun [15], combined with an evolutionary algorithm, based on the intersection’s actual topology, and the current traffic demands.

3.2 Time series forecast module

The observer’s task on Layer 1 (Fig. 2) is to create a situation description of the current traffic demands of the underlying intersection. First, raw data recorded by sensors is received by the *Monitor*. Raw data is often noisy and might be subject to disturbances (e.g. because of malfunctioning sensors). Therefore, the received raw data is pre-processed, e.g. it can be filtered or smoothed. Second, the processed information is passed on to the *Situation Analyser* which builds a situation description representing the current system conditions. Therefore, the measurements from the sensors are converted into a rolling detection interval. Third, the *Performance Analyser* retrieves statistical information about the system’s performance, derived from the current monitored data and the history of previous measurements. Fourth, the *Forecast Module* generates forecasts based on historical and recent data for future points in time. Several forecasting techniques each make a forecast, which are combined by a combination strategy into one comprehensive forecast. Fifth, the *Situation Descriptor* receives the current situation description, and the forecasts for future time steps, and creates one overall situation description. Finally, this description is passed on to the controller, where it serves as basis for the signal plan selection process.

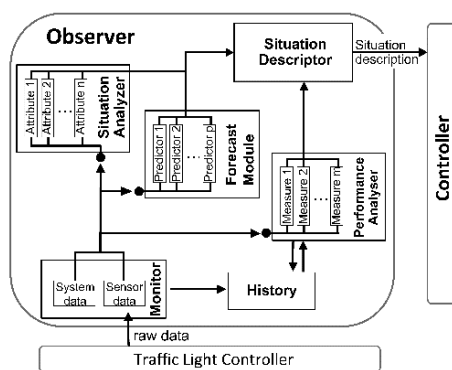


Figure 2. Detailed view of the observer on Layer 1.

4 Proactive adaptation of green times

OTC runs in cycles – meaning that a review of the current control strategy takes place at the beginning of each second or third cycle of the currently active phase plan (due to safety reasons and to allow the current strategy to show its performance). Taking into account a standard duration of such a phase plan of about 90 seconds, OTC decides about an adaptation in intervals of three to five minutes. Hence, an appropriate forecast would alter the adaptation strategy of OTC: Instead of selecting the most promising signal plan for the current traffic conditions,

selecting the one for the estimated traffic demands in the near future can lead to a better anticipatory adaptation strategy. Historical and current traffic flows of turning movements are used to estimate the most probable traffic state at a certain future point in time. This process involves the following four steps:

- 1) Create a vector describing the current traffic demands.
- 2) Create short-term forecasts for future traffic demands.
- 3) Combine these vectors into one situation description.
- 4) Use the new vector to select the next signal plan.

4.1 Representation of the current traffic demands

We estimate the intersection’s traffic flows based on a real-valued vector $\vec{M} = (M_1, \dots, M_n)$ containing the estimated hourly flows for each of the intersection’s signal groups. We make no limitations concerning the type of detection hardware to measure these traffic flows. Fig. 3 depicts a class diagram with the entities involved in the creation of \vec{M} and their relations among each other.

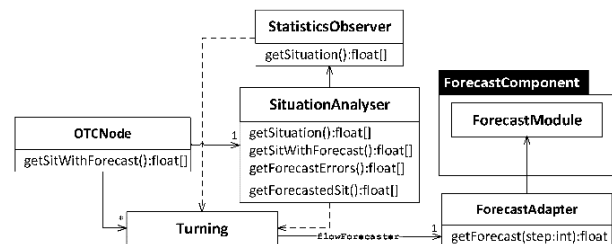


Figure 3. Class diagram showing a partial view of OTC.

An *OTCNode* resembles OTC’s abstracted internal model of the physical intersection. Therefore, it knows all its related turnings and detectors, its approaching and outgoing sections, and the current signal parameters. A description of the currently monitored traffic flows for each turning of the respective intersection can be derived via the associated *SituationAnalyser*. Each turning’s traffic demands are available as time series via a *StatisticsObserver* which retrieves its data from physical sensors, such as inductive loop detectors.

4.2 Forecasting the traffic flows of turning movements

Analogously to vector \vec{M} describing the current traffic demands, we have to derive forecasts for each of the intersection’s turning movements. Therefore, we extend each *Turning* by a *ForecastAdapter* which triggers the *ForecastModule* to generate forecasts of the future traffic flow based on time series of the recently monitored traffic flows (Fig. 3).

Our *ForecastModule* follows the idea of ensemble forecasting, resembling a weighted combination of forecasts calculated by several independent forecast methods. First, several forecasting methods P_1 to P_n independently generate forecasts based on their individual model, and historic and recently monitored data. The forecasts are generated for the point in time t , when the

next signal plan change can be executed. These forecasts F_1 to F_n are then combined based on the selected combination strategy by the *Forecast Combiner*. This module calculates the accumulated forecast as $\hat{F} = f(\vec{F}, \vec{w}) = \sum_{i=1}^n F_i * w_i$. The according weights w_1, \dots, w_n can be calculated by a basic approach, such as the simple average of all forecast values, or more complex ones, such as machine learning techniques that learn the optimal weights over time. The associated weights are managed by the *Weight Learner*. They are iteratively updated based on the forecast errors from the previous time step $t - 1$ derived from the actual value v_{t-1} and the respective forecasts. Finally, the combined forecast \hat{F} is calculated based on \vec{w} and the respective forecasts \vec{F} .

4.3 Weighted Combination of Traffic Flows and Forecasts

Having generated a vector $\vec{F} = (F_1, \dots, F_n)$ consisting of the forecast for each of the intersection's turning movements, we combine \vec{F} with the current traffic flow \vec{M} as

$$\vec{C} = \alpha * \vec{M}_t + (1 - \alpha) * \vec{F}_{t+m} \quad (1)$$

One crucial point is the selection of the optimal weight factor α . This can be a static weight or it can be dynamically calculated based on the accuracy of the latest forecasts. Consequently, a performance measure is needed that defines the forecast accuracy of each individual turning's forecast module. The higher the accuracy, the more likely it is that future forecasts will be equally accurate. Therefore, higher accuracy leads to more influence of the forecasts on the combined situation vector.

Algorithm 1 depicts the combination process. The vector *sit* describes the current traffic conditions at the local intersection (one value for each turning) and *F* contains a forecast value for each turning. The forecasts are created with the previously introduced forecast framework. These two vectors are combined based on the vector of forecast errors *e*. Each forecast error is calculated independently for each turning's forecast component in a sliding window fashion with the Mean Absolute Scaled Error (MASE). The resulting MASE depicts a reliable forecast accuracy, in case its value is lower than one. The lower the MASE value, the higher the influence of the forecasts on the final combination (see lines 5 to 7). For example, if the MASE is zero for a certain turning, the current value is neglected and only the respective forecast value is used at the according position in *sit_{adj}*. In case the MASE is higher than a previously defined maximal error, the currently monitored sensor values are taken without adaptation (line 9). Otherwise, the forecast error accounts proportionally to the final result. Finally, the adjusted situation vector *sit_{adj}* is returned. The result is an adjusted traffic flow vector combining the forecast and the current traffic flow of each turning according to the estimated forecast accuracy of the respective forecast module.

Algorithm 1. Build a situation description combining traffic flow forecasts and the current situation.

Data: situation *sit*, forecast vector *F*, forecast error vector *e*
Result: Vector with adjusted situation *sit_{adj}*

```

1  sitadj ← ∅;
2  errormax = 1.2;
3  for i = 0; i < sit.length; i++ do
4    error = |e[i]|
5    if error ≤ errormax then
6      α = error/errormax
7      sitadj[i] = α * sit[i] + (1 - α) * F[i]
8    else
9      sitadj[i] = sit[i]
10   end
11 end
12 return sitadj

```

Figure 4. Algorithm 1

4.4 Representation of the current traffic demands

Based on this weighted combination of the current and the forecasts of the turning's future traffic flows, the new situation is then passed-on to the controller on Layer 1. Finally, the controller selects a new signal plan matching the input vector provided by the observer. The new signal plan is activated once the currently active signal plan's cycle is finished.

5 Evaluation

In order to demonstrate the benefit of our approach, we conduct a simulation study to evaluate the impact of traffic flow forecasts within the learning-based traffic control strategy of OTC. The evaluation is done with the traffic simulation toolkit Aimsun 8.0 [15] which is a widely used tool in the field of professional traffic engineering. We compare three approaches: a) the reference solution running the fixed-time control strategy installed in reality, b) the standard OTC system as introduced in Section 3.A, and c) the new proactive adaptation of the signal timings as introduced in Section 4. We perform three independent runs for each approach and simulation model.

5.1 Experimental setup

1) *Traffic Simulation:* We developed near-to-reality simulation models of urban areas in Hamburg, Germany that reflect the actual topology, traffic conditions as derived by a census, and the actual control strategies running in reality. Due to space restrictions, we only present the results of one intersection. K3 (Fig. 5) is a four-armed intersection with eleven turnings and part of the federal highway 433 connecting Alsterkrugchaussee, Deelboge, and Borsteler Chaussee. K3 has 25 signal groups, ten of which serve motorised traffic movements. Six inductive loop detectors serve as input to the traffic-actuated control.

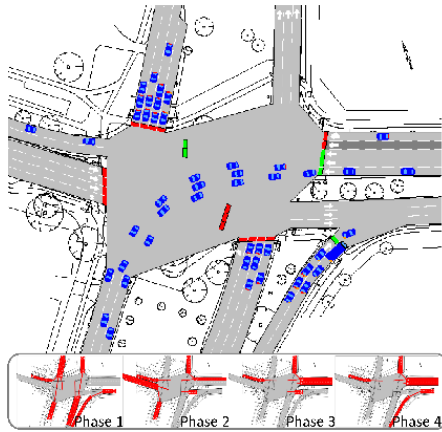


Figure 5. Aimsun's simulation model of intersection K3 at Hamburg, Germany.

The traffic data was collected at Tuesday, May 4 2004. In this census, cars and trucks passing the intersection were counted with a time resolution of 15 minutes. The simulation duration ranges from 5.30 a.m. to 12.30 p.m. including a typical morning rush hour scenario (6 a.m. to 8 a.m.). OTC starts without further knowledge, only the standard fixed-time signalisation is known in advance.

2) *Traffic Demand:* Fig. 6 illustrates K3's traffic profiles for three different replications with different random simulation seeds. The horizontal axis shows the time, the vertical axis depicts the traffic flow in vehicles per hour. The simulated period starts with low traffic density before it drastically raises up to 8000 vehicles per hour between 6 a.m. and 8 a.m. After 9 a.m., the traffic density slowly decreases to a medium level. The ratio of passing trucks to cars lies between 2% and 3%.

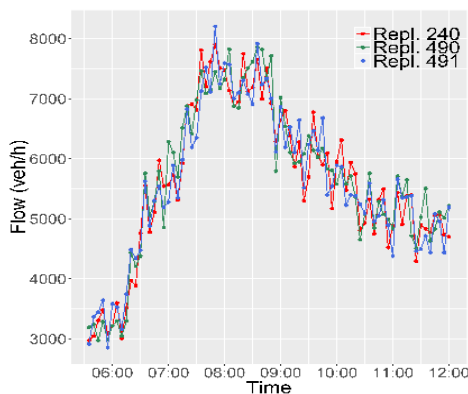


Figure 6. Vehicular traffic flow profile for K3.

3) *Signal Plan:* K3's simulation model (Fig. 5) depicts the real topology including the actual fixed-time signal programs that have been developed by traffic engineers. Table I shows the static fixed-time signal timings for the deployed traffic light controller which is later-on used as a reference controller in the simulation study. The signal plan has four phases and defines the green times and the interphase durations in between green phases. The duration of yellow light is three seconds, and the signal plan's cycle time is 90 seconds.

Table 1. Fixed-time signal timings in seconds (IP=Interphase, GT= Green time)

Phase 1		Phase 2		Phase 3		Phase 4			Cycle time
IP	GT	IP	GT	IP	GT	IP	GT	IP	
10	33	13	9	4	3	7	9	2	90

4) *Forecast Module:* Parameters for the forecast module are set as follows. The forecast methods used are Cubic Spline, Croston, Double Exponential Smoothing (DES), and Exponential Smoothing (ES). They are provided by the forecast package implemented in R [16]. They calculate their forecasts based on the last 10, 15, 15, and 10 observations. DES and ES automatically estimate their smoothing value α . Croston's α parameter is set to the default value 0.1. The simple average is used as combination strategy.

5.2 Experimental results

We evaluated our approach by means of several metrics:

- Mean queue of vehicles in front of red traffic lights.
- Average stop time in seconds per kilometre.
- Vehicle's emissions: mono-nitrogen oxide (NOx), carbon dioxide (CO₂), and particulate matter (PM).

The emission of these three pollutants has been estimated with the help of Aimsun's internal pollution emission model which is based on [17]. Each vehicle's emission is estimated based on its current speed and acceleration using non-linear multiple regression techniques. To remove random factors, the results are averaged over three independent replication runs with different seeds.

1) *Forecast Accuracy:* Fig. 7 presents the time plot of the actual and the forecasted traffic flow for a random turn of K3. The forecast horizon equals twice the signal plan's cycle time, here $2 \cdot 90s = 180s$. The horizontal axis displays the simulated time step, the vertical axis the vehicular flow in vehicles per hour. Although the actual traffic flow (dotted line) shows heavy fluctuations, the forecasts (solid line) are rather precise. The time plot shows that the forecasts tend to be too low in certain situations. Usually, this happens when traffic flow decreases.

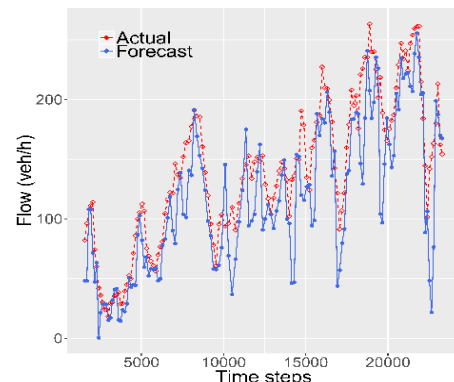


Figure 7. Actual vs. forecasted traffic flow in vehicles per hour.

The histogram of the forecast errors in Fig. 8 supports this observation. Still, this plot shows that the distribution of forecast errors is roughly centred around zero, more or

less normally distributed, and slightly skewed to the left compared to a normal curve (solid line).

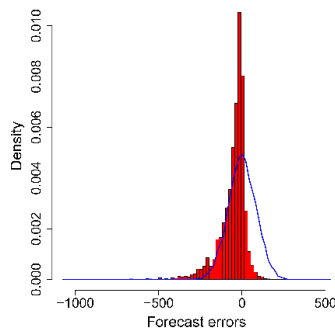


Figure 8. Forecast error histogram for 3 replications.

2) *Simulation Results:* The results of the simulation runs are given in Table 2. Compared to the standard OTC (fixed-time signal plan), the forecast-augmented OTC-control significantly reduces the mean queue by 5.3% (16.3%), and the average stop time by 3.4% (12.0%). Consequently, OTC with forecasts reduces vehicle emissions, such as CO₂ (FTC: 1.4%, OTC: 0.5%), and NO_x (2.5%, 0.6%), having slightly higher values for PM (-2.7%,-0.6).

Table 2. Average simulation results (3 replications)

	FTC	OTC	OTC+Forec.
Mean queue [veh]	52.4	46.4	43.9
Stop time [s/km]	152.4	137.0	131.6
CO ₂ [g/veh]	565.3	560.2	557.0
PM [g/veh]	0.329	0.336	0.338
NO _x [g/veh]	1.60	1.57	1.56

Fig. 9 shows the time series for the total average delay in seconds per kilometre. Using the static fixed-time signal plans, vehicles are faced with an average delay of 171.7 seconds per kilometre. The use of OTC-adapted signal plans reduces this value by 8.9% to 156.3 seconds. Compared to the standard OTC (fixed-time signal plan), the additional consideration of forecasts results in further improvement by 3.4% (12.0%), reducing the average delay to 151.0 seconds. It flattens the steeply rising peak in the morning that occurs from 6 a.m. to 8 a.m. before developing to a normal level until 10 a.m. The time, until a normal level of delay is reached, can be reduced.

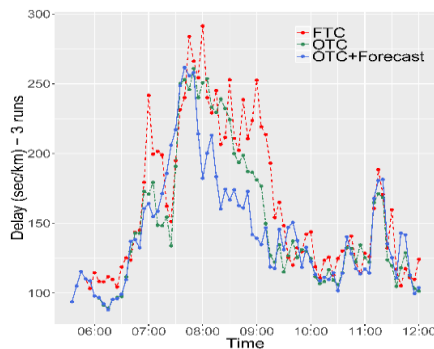


Figure 9. Average network-wide delay (3 replications).

6 Conclusions

The Organic Traffic Control (OTC) system is a self-adaptive feedback controller for adapting the green durations of traffic lights at intersections to the changing demands of traffic behaviour. We compared this approach to our forecast-augmented solution and to a real-world controller that is actually implemented in Hamburg, Germany. Using this setup, we demonstrated that our developed technique outperforms the standard OTC and the reference solution, especially in highly demanding conditions such as the morning rush-hour. This is especially promising since peak load poses the most severe challenge to traffic control.

References

1. M. Birdsall, *Google and ITE: The road ahead for self-driving cars*, ITE Journal **84**, 5, pp. 36–39 (2014)
2. N. Navet, Y. Song, F. Simonot-Lion, C. Wilwert, *Trends in automotive communication systems*, Proc. of the IEEE **93**, no. 6, pp. 1204–1223 (2005)
3. M. Selinger, L. Schmidt, *Adaptive traffic control systems in the U.S.: Updated summary and comparison*, HDR Engineering, Tech. Rep. (2010)
4. H. Prothmann, *Organic Traffic Control*. KIT Scientific Publishing (2011)
5. A. Stevanovic, National Research Council (U.S.), *Adaptive traffic control systems: domestic and foreign state of practice*, Synthesis of highway practice (2010)
6. N. H. Gartner, *OPAC Strategy for demand-responsive decentralized traffic signal control*, in Control, Comp., Communic. in Transp. (1989)
7. A. G. Sims, K. W. Dobinson, *The Sydney coordinated adaptive traffic (SCAT) system – Philosophy and benefits*, IEEE Trans. Veh. Techn. **29**, 130-137 (1980)
8. D. I. Robertson, R. D. Bretherton, *Optimizing networks of traffic signals in real time – the SCOOT method*, IEEE Trans. Veh. Technol. **40**, 11–15 (1991)
9. R. Chrobok, O. Kaumann, J. Wahle, M. Schreckenber, *Different methods of traffic forecast based on real data*, European Journal of Operational Research **155**, 3, pp. 558–568 (2004)
10. S. Garside, K. Lindveld, and J. Whittaker, *Tracking and predicting a network traffic process*. Intern. Journal of Forecasting **13**, pp. 51–61 (1997)
11. M. S. Dougherty, M. R. Cobbett, *Short-term inter-urban traffic forecasts using neural networks*, Intern. Journal of Forecasting **13**, 1, pp. 21–31 (1997)
12. R. Adhikari, R. K. Agrawal, *Performance evaluation of weights selection schemes for linear combination of multiple forecasts*, Artif. Intell. Rev., 529-548 (2014)
13. M. Sommer, S. Tomforde, J. Hähner, D. Auer, *Learning a Dynamic Re-combination Strategy of Forecast Techniques at Runtime*, Proc. of IEEE Int. Conf. Autonomic Computing, pp. 261–266 (2015)

-
14. M. C. Bell, R. D. Bretherton, *Ageing of fixed-time traffic signal plans*, 2nd Int. Conf. On Road Traffic Control, pp. 77–80 (1986)
 15. J. Barcelo, J. Casas, *Dynamic network simulation with AIMSUN*, Int. Symp. on Transp. Simulation. Kluwer, pp. 1–25 (2002)
 16. R. J. Hyndman, Y. Khandakar, *Automatic time series forecasting: the forecast package for R*, J. Statistical Software **26**, 3, pp. 1–22 (2008)
 17. L. I. Panis, S. Broekx, R. Liu, *Modelling instantaneous traffic emission and the influence of traffic speed limits*, Science of The Total Environment **371**, pp. 270–285 (2006)