Enabling GLOSA for Adaptive Traffic Lights

Robert Bodenheimer*, Alexej Brauer†, David Eckhoff‡ and Reinhard German‡
*AUDI AG, Germany
†Technische Universität München, Germany
‡Computer Networks and Communication Systems, Dept. of Computer Science, University of Erlangen, Germany
robert.bodenheimer@audi.de, alexej.brauer@tum.de,
{david.eckhoff, reinhard.german}@fau.de

***Abstract*—Green Light Optimized Speed Advisory (GLOSA) systems aim at giving ideal target speed recommendations to the driver when approaching a traffic light to lower CO_2 emissions (and fuel consumption) and to reduce the number of unnecessary stops. These systems have been shown to work well with static traffic light programs, unfortunately, a large portion of traffic lights in inner cities are adaptive and can change their behaviour with almost no lead time.**

This paper presents and validates (using field tests and simulation) a method to help overcome this problem and forecast fully and semi-adaptive traffic lights. First, we transformed the state graph of the traffic light controller into a transition graph focusing on signal changes and their occurrence probability. We then reduced routing possibilities within the graph using real life observations and recorded detector data of the traffic light. We further optimized our system in terms of needed storage and computationally efficiency. Our results show that in 80 % of all cases we could predict signal changes 15 s in the future with a high enough accuracy to enable GLOSA for adaptive traffic lights.

I. Introduction

Green Light Optimized Speed Advisory (GLOSA) systems, that is, the recommendation of an optimal speed to pass a traffic light just after it turned green, have been shown to be able to reduce CO_2 emissions and fuel consumption by up to 13 % [1], [2]. The basic principle is to calculate this speed recommendation based on the distance to the traffic light and the time to the next signal change. The latter is wirelessly transmitted to the vehicle using IEEE 802.11p [3] communication or cellular network technology such as UMTS or LTE. These messages are referred to Signal Phase and Timing (SPaT) messages [4] and have also been standardized for ETSI ITS G5 [5].

Forecasting and thereby enabling GLOSA for non-adaptive traffic lights is trivial as they run their static program in an endless loop. These traffic lights only need a power supply and a clock with no additional communication technology as the signal changing times could be transmitted via cellular networks from a centralized server that has knowledge of all traffic light programs. However, to improve traffic flows more and more traffic lights become adaptive and are able to react to certain traffic conditions. As an example, in the city of Hamburg, Germany over 95 % of all traffic lights are adaptive. These traffic lights can 'decide' to change their signal with lead times as short as 1 s and are therefore complex to forecast. Because the traffic light controller is stimulated by external inputs it cannot forecast own future behaviors itself. Not being able to predict these signal changes would have a considerable impact on the success and applicability of GLOSA systems in future intelligent transportation systems.

Adaptive controllers can vary in semi adaptive and fully adaptive behavior. While semi-adaptive controllers do not alter the order of signals but only their length, fully adaptive controllers can also change the order of signals and even leave unneeded ones out. Adaptive traffic lights can be stimulated by many different inputs. The most important ones are detectors in or above the street to measure the number of approaching and departing vehicles. Detectors for pedestrians, strategic control signals from traffic management centers, or the prioritization of emergency vehicles can also be relevant for the behavior of the traffic light.

This requires adaptive traffic lights to be equipped with a communication device to transmit their current detector occupation in order to be forecast reliably. This information can either be used locally to predict the signal behavior and disseminated in an ad-hoc fashion using IEEE 802.11p or it can be sent to a central service provider where it is evaluated and then sent back to approaching vehicles using cellular communication.

In this paper, we present a method to also enable GLOSA for fully adaptive traffic lights. For this, the traffic light program (represented by a graph where each node is a distinct state of the traffic light and transitions represent the signal changes) is transformed to a transition graph focusing on the signal changes. Each transition is assigned a certain probability, based on real life observations and the use of detector data provided by the traffic light. These probabilities can then be used to predict the signal change and give the driver a speed recommendation. Our system is validated by means of simulation and is already installed in large test beds in the city of Ingolstadt. We were able to show that it works with high accuracy and therefore provides a significant contribution to the successful operation of GLOSA systems.

The remainder of this paper is organized as follows: In Section II we discuss related work in the field. Section III describes the basic approach to forecast adaptive traffic lights which is then further optimized in Section III-C. In Section IV we discuss our simulation results followed by a short overview where our system is already installed (Section V). Section VI concludes the paper and discusses directions for future work.

II. Related Work

GLOSA systems date back to as early as 1983 when Volkswagen introduced the *Wolfsburger Welle*, a driver assistance system that informed approaching drivers about the signals of infrared communication enabled traffic lights [6]. Although the project was discontinued at the time, the concept was picked up again in 2009 by the Travolution project [7] that instead of infrared used IEEE 802.11p communication and showed that GLOSA is a promising approach to lower CO_2 emissions and fuel consumptions. Since then GLOSA systems have been a part of many projects and field operational tests such as PREDRIVE C2X [8] or sim^{TD} [9].

Impact and potentials of Green Light Optimized Speed Advisory have been the focus of many publications [1], [2], [8]. Based on extensive simulations, these papers offer valuable clues to what extent CO_2 emissions can be lowered, fuel be saved, and how many stops could be avoided. The publications more or less agree that a potential benefit of about 13 % of emission reduction can be reached when all traffic lights are static and all vehicles are equipped with the GLOSA device.

Focusing on optimizing the system itself, for example by recommending not only the speed but different driving strategies or considering the current queue length at the traffic light was covered in [1], [10]–[12]. It has been shown that when these parameters are considered, the system can even further decrease the number of unnecessary stops and reduce consumption and emissions.

Unfortunately, almost all literature is based on the assumption of traffic light programs to be static allowing a trivial forecast of the signal changes. In the 10 biggest German cities, about 73 % of all traffic lights are fully or semi-adaptive. This fact would therefore considerably reduce the benefits introduced by GLOSA systems that only work for static traffic lights.

In his mathematical prognosis approach for adaptive traffic lights Weisheit describes the usage of Support Vector Machines (SVM) to recognize similar traffic conditions to improve the prediction accuracy [13]. This requires a large knowledge base to provide enough sample data and was shown to reach a forecast quality of 83 % using the observations from 40 operation hours. He focuses on only one direction of signal changes, that is, from green to red. However, GLOSA systems mainly build on the prognosis from red to green to inform drivers by how much lower the recommended speed has to be to arrive shortly after the traffic light turns green.

In this paper we show how to bridge the gap of missing prognoses at adaptive traffic lights to help ensure that speed recommendations can be given at every traffic light to fully use the potential of GLOSA systems.

III. Approach

A traffic light program for all traffic control signals at a given intersection can be described by the controller graph. This graph is the most important input parameter for our approach and needs to be provided by the operator or can be generated by observing the report messages of the traffic light itself. One node in this graph represents an entire state of the traffic light, that is, the signals for all lanes. An edge between two nodes means that there is a possible transition from one state to another. (See Appendix, Figure 7 for a real controller graph for a traffic light in the city of Ingolstadt provided by the operator.)

The transition is a short static program that starts with the signaling of the source node (e.g., lanes 1 & 3 green, 2 & 4 red) and ends with the signals of the destination node (e.g., lanes 1 & 3 red, 2 & 4 green). All signals must be changed in a save order, including amber phases and sufficiently long blocking times for the intersection to be cleared. Each time a transition is triggered the traffic light reports this event to the operator. These reports are also used as sample data in our approach. Depending on the level of adaptivity a prediction can start with sample data of 75 min from comparable day classes and time windows.

To account for highly dynamic traffic conditions we introduce day classes and time windows as described in [14]. This means that each reported signal change is categorized by the type of day (e.g. Monday, Vacation) and by the time window it is in (e.g. 11:00 am till 12:00 pm). The recorded data can then be used to predict signal changes more accurately. This requires a complete calendar including holidays, school vacation, or even sports events for the city in which the traffic light is located. Especially for rare day classes a longer learning phase is necessary to collect enough data for the forecast to function properly.

A. Graph Transformation

A controller graph consists of nodes for all allowed signal combinations and edges for possible transitions between them. As this graph focuses on the signals rather than the signal changes we transform this graph to a transition graph to be able to reliably predict the order and the time offset of the transitions. The transformation of the original controller graph G into a transition graph G' (as illustrated in Figure 1) can be done using the following equations:

$$G = (V, E, f, g) \tag{1}$$

$$= \sum_{\forall i} C_i \quad \text{with } C \subseteq G \text{ and } C = x_0 \dots x_{k-1} x_0 \tag{2}$$

$$\varphi : V \to V' \quad \text{with } xy \in E \Leftrightarrow \varphi(x)\varphi(y) \in E' \tag{3}$$

$$\varphi : \varphi(x) = g(wx) \quad \text{with } wx \in E \tag{4}$$

$$G \to G' \tag{5}$$

$$G' = \sum_{\forall i} C'_i \quad \text{with } C'_i = \varphi(C_i) \tag{6}$$

The controller graph (G) is a node (f) and edge labeled (g), finite and strongly connected digraph (1). Therefore it can also be displayed as a sum of all its cycles (2). Each circle is one possible path through the controller graph with no branches. It is also possible to create an isomorphism (φ) that does not change the topology of each circle (3) but transforms the label of the nodes by replacing them by the label of the incoming edge (4). The transformation of the controller graph into the transition graph (5) is the sum of all transformed cycles (6).

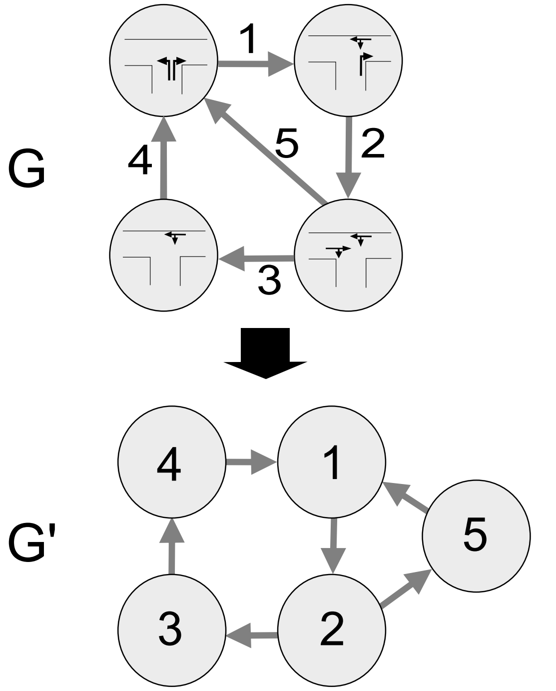

Figure 1. Graph transformation

Nodes in the transition graph represent the transition and the edges show which transition can possibly follow up. The costs of the edges can be used to store data needed to forecast the next transition correctly.

B. Probability Assignment

To forecast a traffic light the occurrence of its transitions has to be predicted. Therefore the correct order of transitions and their length have to be determined.

If there are multiple possible transitions, that is, one node in the transition graph has more than one successor, each transition has to be assigned a probability. These probabilities can be calculated based on empiric data collected from the traffic light using Equation (7). The equation is also used to calculate the occurrence probabilities between two consecutive transitions for all possible offset times.

$$P\left\{\Gamma_t = (\mu, i, j) \middle| \left(\Lambda_t = (\mu, i)\right) \wedge \left(Z_{(t-\mu)} = h\right)\right\} \quad (7)$$

The equation describes the probability for the occurrence Γ of a distinct transition j at a certain point in time t and with i to be the last observed transition μ seconds earlier. The first constraint is transition i to be the direct predecessor of transition j with the predecessor running for μ seconds till time t. The second condition takes the day class Z into account, that is, all considered sample data requires the same time classification h as it was present at time $t - \mu$.

All required information to forecast traffic signal phases can now be assigned to the edges in the transition graph. An example final transition graph is shown in figure Figure 2. The sum of all outgoing edges from a transition must always be 100 %, i.e. at branches the assigned probability indicates which transition is more likely to follow. Additionally, every edge also holds information about the minimum, the maximum and the most likely offset time. These values are required for the optimal operation of a GLOSA system: As vehicles approaching an intersection are often bound to surrounding traffic, it is not always possible to drive at the exact recommended speed. In these cases, minimum and maximum times can be used to determine whether the current speed is also acceptable. Also, when operating on low confidence forecasts, on-board applications can decide to aim for the earliest or latest change time instead.

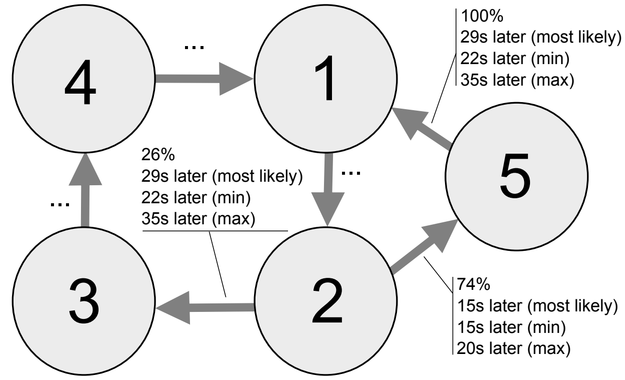

Figure 2. Final transition graph

C. Graph Extension

A critical section in the transition graph are nodes with more than one possible successor. For the forecast system to work properly edges are desired to be assigned a high probability. Assume a main road with a simple traffic light program alternating to allow the merging of traffic at a crossroads. In the transition graph this would lead to two outgoing branches, each with a probability of 50 %. In this case, a prediction solely based on the assigned probabilities cannot meet the quality requirements for a GLOSA prognosis.

One possible approach to counter this problem is to also account for transitions before the current one. By looking back in the transition graph it can become considerably easier to forecast the next transition. Information about previous transitions could also be merged into the edge costs, but instead of bloating the information assigned to the edges we decided to clone affected nodes to represent this additional knowledge. It is not required to clone every node in the transition graph, but only nodes between branching and merges because for these nodes the path from the last merge needs to be taken into account.

The benefit of this method is shown in Figure 3. While in the upper, non-extended graph the probabilities for transition 2 and 3 to follow transition 1 are equally 50 %, cloning nodes 1 and 4 to store one step of history creates edges with 100 %

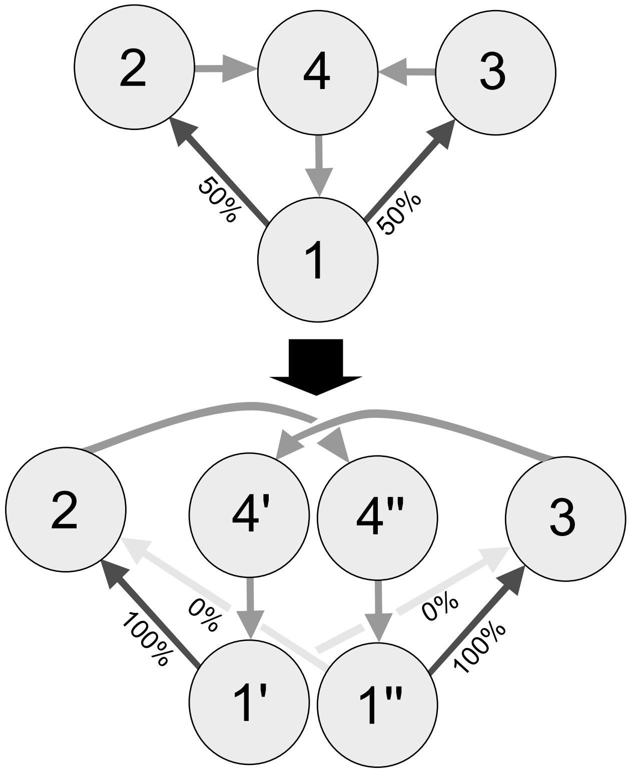

Figure 3. Graph extension to eliminate low probability edges

probability allowing the lower transition graph to be used for GLOSA prognosis.

In theory, we could create even more clones to store longer histories and thereby more complex alternations, however, in real life most traffic lights do not toggle between more than two paths. All other and more complex decisions of traffic lights are made with the help of detectors.

D. Using Detectors

Fully and semi-adaptive traffic lights are stimulated by detectors and triggers. These sensors have a major impact on the signal phases as they make it possible to consider current situations to improve traffic flows. Therefore, a prediction of adaptive traffic lights of good enough quality to be used in a GLOSA system is not possible without taking detectors into account.

Detectors and triggers can be categorized into three basic classes:

- vehicle, bus and pedestrian detectors
- emergency vehicle triggers
- traffic management triggers (used by the operator)

To include the current state of a detector in the prognosis, traffic lights need to be equipped with communication devices to transmit this information to the back-end of the forecast system, where all edges in the transition graph are assigned probabilities. Each transition can depend on a different set of detectors. To keep the complexity at a manageable level we ran a χ^2 analysis to only consider detectors for a transition that really have a considerable influence. Extending Equation (7) to account for detector states yields:

$$P\left\{\Gamma_t = (\mu, i, j) \middle| \left(\bigwedge_{y=1}^{\kappa}(Dy_t = d_y)\right) \wedge \Big(\Lambda_t = (\mu, i)\Big) \wedge \Big(Z_{(t-\mu)} = h\Big)\right\} \tag{8}$$

In addition to the constraints from Equation (7), we now only consider sample data with the same detector occupation at the intersection for all κ relevant detectors $\left(\bigwedge_{y=1}^{\kappa}(Dy_t = d_y)\right)$. This extension allows the consideration of the first class of detectors, that is, vehicle, bus and pedestrian detectors.

The second class of detectors, the emergency vehicle triggers, sets the traffic light in a special mode to give way for said vehicles. While the traffic light could forward the trigger signal to the prediction service as long as it is present, the traffic light does not know how long it will remain in this special mode. In these circumstances the forecast mechanism is likely to have a low accuracy. Additionally, warning sirens and emergency lights of the emergency vehicle would overrule the traffic light anyway, therefore we decided to stop the prognosis service while emergency triggers are active.

The last type of detectors are steering commands or preset parameters from a traffic management center. These commands are not yet standardized and are incorporated into each traffic light controller program individually. At this point, we decided to ignore this class of triggers and to not adjust every transition graph separately but to aim for developing a generic architecture for all traffic lights. This will be the focus of future work to further improve the prognosis quality.

E. Simplification

The use of detector data increases the number of data sets required for an accurate forecast enormously. Even after eliminating all irrelevant detectors for a transition, the number of relevant detectors can still be large. As edges in the transition graph have to cover all possible combinations of detector occupations, this may cause a problem in computational efficiency possibly introducing latencies. Furthermore, this can become a difficult requirement for the database storage and the amount of sample data needed to run a proper prognosis.

As no vehicle can trigger more than one detector when driving through an intersection, we treat all detectors mounted to the traffic light as statistically independent to each other. This allows us to compute the signal change probability as:

$$P\left\{\Gamma_t = (\mu, i, j) \middle| \left(\bigwedge_{y=1}^{\kappa}(Dy_t = d_y)\right) \wedge \Big(\Lambda_t = (\mu, i)\Big) \wedge \Big(Z_{(t-\mu)} = h\Big)\right\} = \frac{\prod_{y=1}^{\kappa}\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| (Dy_t = d_y) \wedge (Z_{(t-\mu)} = h) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| (Z_{(t-\mu)} = h) \wedge (\Lambda_t = (\mu, i))\right\}\right)^{\kappa-1}} \tag{9}$$

Assume a traffic light with 4 detectors (Dy_t), that can either be occupied (=1) or free (=0). The detector state for the traffic light can then be coded as a 4 bit vector, e.g. (1010). To compute the transition probability with an occupied detector 1,

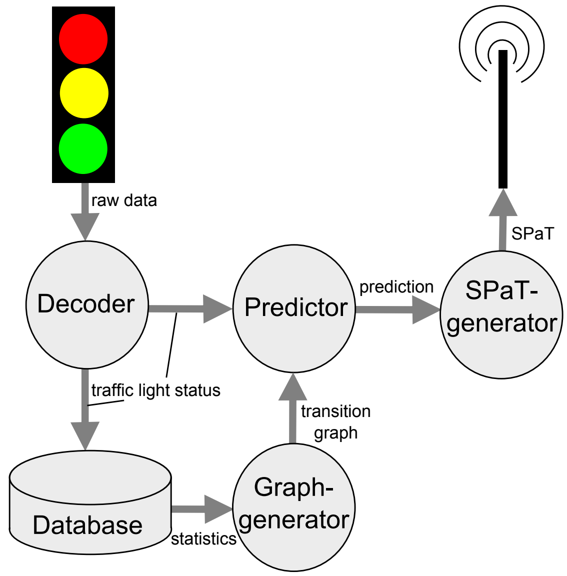

Figure 4. Data Flow

we'd need one edge in the transition graph for each possible permutation (1000), (1001), ... (1111). The simplification allows us to compute the transition probability for one specific detector set as the product of each single detector probability, yielding a transition probability for the state (1xxx) as stated in Equation (9). The full proof can be found in the Appendix.

This simplification introduces several benefits: For the GLOSA system to work accurately, now considerably less empiric data is needed as each recorded data set can now be analyzed independently for each detector. This makes it easier to start the prognosis service with fewer sampled data (e.g. in the roll-out phase) and allows for an easy alteration of relevant detectors for certain transitions. Furthermore, the system performance itself can benefit from Equation (9) as the computations become less complex and the needed storage in the database back-end can be reduced. Although the quality of the prediction does not benefit from this, this performance optimization can have a positive impact on the roll-out of GLOSA systems.

F. System Architecture

Figure 4 shows the data flow and the final architecture of our approach. Raw data collected from the traffic light (e.g. detector states and current signals) is decoded and stored to a database to continuously enlarge the knowledge base. Based on this knowledge base a transition graph including transition probabilities can be generated. Taking into account the current detector states, the time of day and so on, a prediction of the next signal change can be made. Part of every prediction is the confidence of the forecast so every receiver can judge the usefulness of the prediction separately. This forecast is put into a SPaT message that is transmitted to approaching vehicles via IEEE 802.11p or cellular communication. The on-board

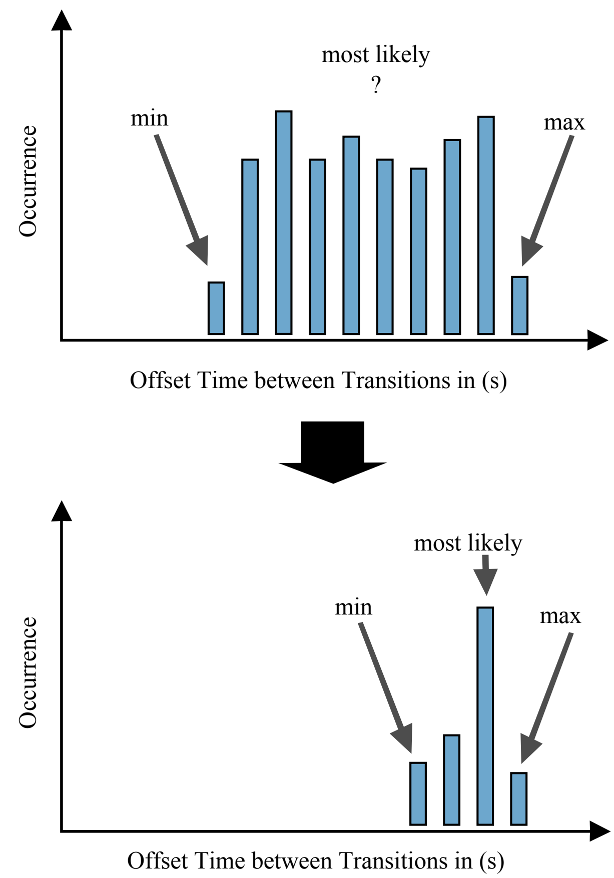

Figure 5. Example for increasing the prediction accuracy using historic and detector data

unit of the vehicle can then give out a speed recommendation to the driver, for example by displaying them on the head up display.

The enhancements described in this paper all aim at narrowing down the time window to the next possible signal change. Without the usage of cloned nodes, detectors, or historic data a possible distribution of signal change times would be rather wide as illustrated in the upper part of Figure 5. Using said enhancements these distributions can be made considerably narrower allowing for the transmission of minimum, maximum, and most likely signal changing times. This is illustrated in the lower part of Figure 5.

Unfortunately, not all traffic lights can be sufficiently optimized this way. Therefore, in addition to the prediction of the signal change time, the prognosis also requires a value to reflect its confidence. This value needs to be assessed carefully and included in the transmitted SPaT messages. One possible approach to obtain confidence values is to look at the quality of past predictions as described in the next section. By transmitting the confidence of every forecast each receiver can judge separately whether to use this information or to wait for another forecast with higher confidence but less time to react.

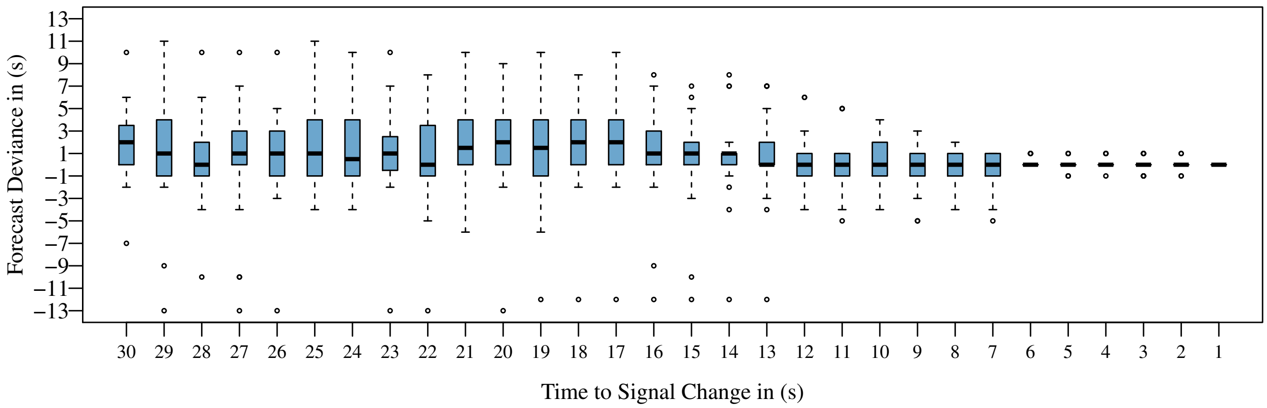

Figure 6. Prediction accuracy of our system for signal changes up to 30 s in the future

IV. Trace Driven Simulation

Although our method is applicable for every traffic light, each traffic light is unique and needs some individual parameters such as the number of required data samples or the set of relevant detectors for a transition. This knowledge is built up in a learning phase by a trace driven simulation that iteratively improves the quality of the forecast. The simulation output is continuously compared to the real signal phases allowing to assess the current quality of prediction. Only after reaching the required accuracy the system will go live for a particular traffic light.

A. Simulation Setup

For the evaluation presented in this paper we predefine a specific assessment time slot of about 1 hour. All real recorded data (i.e., detector states, current signal) from this time slot is then used as a simulation input. In our simulation, the learning phase has already been concluded and the transition graph has been fully build. Every second our systems tries to predict the time to the next signal change. This prediction is logged and, after the simulation has finished, compared to the real value at the time.

This is the same process used in the learning phase to identify the most important parameters for a traffic light.

The output data can be used to rate the prognosis itself and determine the confidence value. By storing these values the prognosis service can determine how many forecasts were absolutely correct or by how much they deviated and can therefore give a confidence for the prognosis that is transmitted to the approaching vehicle. This is important, as SPaT messages include an extra field for the prognosis confidence. This information can be used by the on-board unit to judge whether the confidence of the received forecast is high enough to inform the driver or if it is preferable to wait for another forecast with less reaction time to the traffic light change but with a higher confidence.

In general, quality assessment for the prognosis algorithm can also be done for just one signal change direction, e.g., from red to green. Application development requires separate confidence values for each direction to determine whether the confidence values are high enough for a given lead time. For example, GLOSA systems mainly focus on red to green changes (by how much does the driver need to decelerate to arrive when the traffic light turns green) while the main interest of autonomous driving is the forecast of green to red changes.

B. Results

Figure 6 shows the results for our trace-driven simulation. The forecast deviance is the time by how much our prediction algorithm was wrong. Predictions for signal changes more than 30 seconds in the future are not transmitted because they are likely to be inaccurate and even if not, to not have a positive effect on traffic efficiency as the vehicle is too far away [2]. For each data set, a box is drawn from the 25% to the 75% quantile; the thick line is the median. Whiskers extend from the edges of the box towards the minimum and maximum of the data set, but no further than 1:5 times the interquartile range. Data points outside the range of whiskers are drawn separately.

When the signal change is longer than 15 s in the future, the prediction accuracy of the system is good most of the times, but tends to over/underestimate the remaining time. For longer forecast times a higher deviance is acceptable because the time to react is also high. At times ≤ 15 s the accuracy noticeably increases with a deviance of less then 2 s in about 80 % of all cases. Our algorithm was able to reliably predict signal changes 6 s or less in the future with an accuracy of over 95%. The variance of the data can also be used to determine the confidence of the information which is then handed to the receiving on-board unit to decide whether to recommend a speed the driver.

Close to the signal change the traffic light has less degrees of freedom due to amber phases and blocking times. This results in the low deviance for the last seconds of the forecast.

V. Field Testing

Our main testbed is located in Ingolstadt (Germany), where we tested both ad-hoc and cellular communication based GLOSA systems. For the IEEE 802.11p [3] approach, we equipped 10 traffic lights with Roadside Units and connected to them the traffic light controller. For these traffic lights SPaT messages were locally available on an ITS channel in the 5.9 GHz from the years 2012 to 2013.

A centralized approach was evaluated using more than 50 traffic lights in Ingolstadt that were modified to continuously report their status to a central server, where the prognosis service can be queried via internet. Compared to the ad-hoc method this system is more cost efficient and can be rolled out easier. Querying the central server did not introduce a problematic latency as the included timestamps could be used to properly evaluate the contents of the received SPaT messages.

For our testbed we use different deviance thresholds to determine when a forecast should be used to inform the driver. In the last 10 s prior to a signal change the deviance has to be less or equal 1 s. Between 10 and 20 s before a traffic light change a deviance of 2 s is acceptable. For predictions of up to 30 s prior to a change a deviance of 3 s is considered a good prognosis.

Additionally to our experiments in Ingolstadt we brought our expertise to several other FOTs like simTD [9] in Germany or DRIVE C2X [15] and several other all over Europe. We also started the testing of GLOSA applications in the US where we had a first presentation on this year's Consumer Electronics Show in Las Vegas.

VI. Conclusion and Future Work

Green Light Optimized Speed Advisory (GLOSA) systems have been shown to be a promising approach to improve traffic flows and reduce CO_2 emissions. Until now, almost all related research assumed static traffic light programs. Unfortunately a large portion of traffic lights are adaptive and can start signal transitions with lead times as short as 1 s.

In this paper we presented a methodology to enable GLOSA also for fully and semi-adaptive traffic lights. Based on a knowledge base containing empiric data such as signals at a given time of day and the state of detectors for vehicles, buses, or pedestrians we were able to predict a traffic light change 15 s in the future with an accuracy of over 80 %. We further optimized our approach to reduce computational complexity and reduce the amount of storage needed in the back-end. Our approach was validated in both simulations and real life experiments in a large test bed in the city of Ingolstadt, Germany.

Future work includes the enhancement of our approach to further increase the prediction accuracy by, e.g., also considering traffic management commands by the operator.

VII. Acknowledgments

We want to thank the city of Ingolstadt, Germany for providing all necessary traffic light information and supporting us in building our testbed.

References

[1] D. Eckhoff, B. Halmos, and R. German, "Potentials and Limitations of Green Light Optimal Speed Advisory Systems," in *5th IEEE Vehicular Networking Conference (VNC 2013)*. Boston, MA, USA: IEEE, December 2013, pp. 103–110.

[2] T. Tielert, M. Killat, H. Hartenstein, R. Luz, S. Hausberger, and T. Benz, "The Impact of Traffic-Light-to-Vehicle Communication on Fuel Consumption and Emissions," in *Internet of Things 2010 Conference (IoT2010)*, Tokyo, Japan, November 2010, pp. 1–8.

[3] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE, Std 802.11-2012, February 2012.

[4] "Dedicated Short Range Communications (DSRC) Message Set Dictionary," SAE Int. DSRC Committee, Std 2735-200911, November 2010.

[5] "Intelligent Transport Systems (ITS); European Profile Standard for the Physical and Medium Access Control Layer of Intelligent Transport Systems Operating in the 5 GHz Frequency Band," ETSI, Std ES 202 663, November 2009.

[6] W. Zimdahl, "Guidelines and some Developments for a new Modular Driver Information System," in *34th IEEE Vehicular Technology Conference (VTC1984)*, vol. 34. Pittsburgh, PA, USA: IEEE, May 1984, pp. 178–182.

[7] R. Braun, F. Busch, C. Kemper, R. Hildebrandt, F. Weichenmeier, C. Menig, I. Paulus, and R. Presslein-Lehle, "Travolution – Netzweite Optimierung der Lichtsignalsteuerung und LSA-Fahrzeug-Kommunikation," *Strassenverkehrstechnik*, vol. 53, pp. 365–374, June 2009.

[8] K. Katsaros, R. Kernchen, M. Dianati, D. Rieck, and C. Zinoviou, "Application of Vehicular Communications for Improving the Efficiency of Traffic in Urban Areas," *Wireless Communications and Mobile Computing*, vol. 11, no. 12, pp. 1657–1667, November 2011.

[9] H. Stübing, M. Bechler, D. Heussner, T. May, I. Radusch, H. Rechner, and P. Vogel, "simTD: A Car-to-X System Architecture for Field Operational Tests," *IEEE Communications Magazine*, vol. 48, no. 5, pp. 148–154, May 2010.

[10] C. Raubitschek, N. Schütze, E. Kozlov, and B. Baker, "Predictive Driving Strategies Under Urban Conditions for Reducing Fuel Consumption Based on Vehicle Environment Information," in *2011 IEEE Forum on Integrated and Sustainable Transportation System (FISTS)*. Vienna, Austria: IEEE, July 2011, pp. 13 –19.

[11] M. Seredynski, W. Mazurczyk, and D. Khadraoui, "Multi-Segment Green Light Optimal Speed Advisory," in *Parallel and Distributed Processing Symposium Workshops and PhD Forum (IPDPSW), 2013 IEEE 27th International*. Cambridge, MA: IEEE, May 2013, pp. 459 – 465.

[12] C. Dittrich, R. Stahlmann, and F. Busch, "Refinement of Urban Traffic State Estimation by Using Queue Length Information," in *5th International Symposium Networks for Mobility 2010*, Stuttgart, Germany, September 2010.

[13] T. Weisheit, "Algorithmenentwicklung zur Prognose von Schaltzeitpunkten an Verkehrsabhängigen Lichtsignalanlagen," in *HEUREKA'14*, Stuttgart, Germany, April 2014.

[14] C. Dittrich, J. Lüßmann, F. Busch, R. Mänz, R. Stahlmann, and C. Menig, "Greener Roads by Talking Traffic Lights: Knowledge about Queue Length and Upcoming Traffic Light Signal," in *2nd International Conference on Sustainable Automotive Technologies (ICSAT 2010)*, Chieming, Germany, February 2010.

[15] R. Stahlmann, A. Festag, A. Tomatis, I. Radusch, and F. Fischer, "Starting European Field Tests for Car-2-X Communication: the DRIVE C2X Framework," in *18th ITS World Congress and Exhibition*, Orlando, FL, USA, October 2011.

Appendix

A. Real Controller Graph

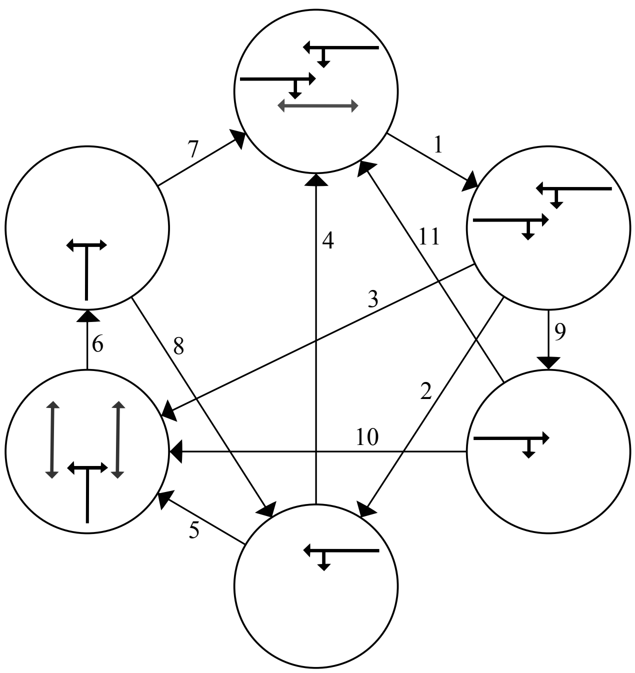

Figure 7. A traffic controller graph as provided by the operator

B. Proof of Equation 9

$$P\left\{\Gamma_t = (\mu, i, j) \middle| \left(\bigwedge_{y=1}^{\kappa}(Dy_t = d_y)\right) \wedge \left(\Lambda_t = (\mu, i)\right) \wedge \left(Z_{(t-\mu)} = h\right)\right\}$$

with $P\{A|B\} = \frac{P\{A\cap B\}}{P\{B\}}$:

$$= \frac{P\left\{(\Gamma_t=(\mu,i,j))\wedge\left(\bigwedge_{y=1}^{\kappa}(Dy_t=d_y)\right)\wedge(\Lambda_t=(\mu,i))\wedge\left(Z_{(t-\mu)}=h\right)\right\}}{P\left\{\left(\bigwedge_{y=1}^{\kappa}(Dy_t=d_y)\right)\wedge(\Lambda_t=(\mu,i))\wedge\left(Z_{(t-\mu)}=h\right)\right\}} =$$

with $P\{A \cap B\} = P\{A|B\} \cdot P\{B\}$:

$$= \frac{P\left\{\left(\bigwedge_{y=1}^{\kappa} Dy_t = d_y\right) \middle| (\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}{P\left\{\left(\bigwedge_{y=1}^{\kappa} Dy_t = d_y\right) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}} \cdot \frac{P\left\{(\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}{P\left\{\left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}$$

with Dy_t independent:

$$= \frac{\prod_{y=1}^{\kappa} P\left\{(Dy_t = d_y) \middle| (\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}{\prod_{y=1}^{\kappa} P\left\{(Dy_t = d_y) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}} \cdot \frac{P\left\{(\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}{P\left\{\left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}$$

with $P\{A|B\} = \frac{P\{A\cap B\}}{P\{B\}}$:

$$= \frac{\prod_{y=1}^{\kappa}\left(\frac{P\left\{(Dy_t=d_y)\wedge(\Gamma_t=(\mu,i,j))\wedge\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}}{P\left\{(\Gamma_t=(\mu,i,j))\wedge\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}}\right)}{\prod_{y=1}^{\kappa} P\left\{(Dy_t = d_y) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}} \cdot \frac{P\left\{(\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}{P\left\{\left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}}$$

$$= \frac{\frac{\prod_{y=1}^{\kappa}\left(P\left\{(Dy_t=d_y)\wedge(\Gamma_t=(\mu,i,j))\wedge\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}\right)}{\left(P\left\{(\Gamma_t=(\mu,i,j))\wedge\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}\right)^{\kappa-1}\cdot P\left\{\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}}}{\prod_{y=1}^{\kappa} P\left\{(Dy_t=d_y)\middle|\left(Z_{(t-\mu)}=h\right)\wedge(\Lambda_t=(\mu,i))\right\}}$$

with $P\{A \cap B\} = P\{A|B\} \cdot P\{B\}$:

$$= \frac{\prod_{y=1}^{\kappa}\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| (Dy_t = d_y) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{P\left\{\left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\} \cdot \prod_{y=1}^{\kappa} P\left\{(Dy_t = d_y) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}} \cdot \frac{\prod_{y=1}^{\kappa}\left(P\left\{(Dy_t = d_y) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{\left(P\left\{(\Gamma_t = (\mu, i, j)) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)^{\kappa-1}}$$

with $P\{A \cap B\} = P\{A|B\} \cdot P\{B\}$:

$$= \frac{\prod_{y=1}^{\kappa}\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| (Dy_t = d_y) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{\left(P\left\{\left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)^{\kappa} \cdot \prod_{y=1}^{\kappa}\left(P\left\{(Dy_t = d_y) | \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)} \cdot \frac{\prod_{y=1}^{\kappa}\left(P\left\{(Dy_t = d_y) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)^{\kappa-1}}$$

with $P\{A|B\} \cdot P\{B\} = P\{A \cap B\}$:

$$= \frac{\prod_{y=1}^{\kappa}\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| (Dy_t = d_y) \wedge \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)}{\left(P\left\{(\Gamma_t = (\mu, i, j)) \middle| \left(Z_{(t-\mu)} = h\right) \wedge (\Lambda_t = (\mu, i))\right\}\right)^{\kappa-1}}$$

■