A NEW ONLINE CONTROL STRATEGY FOR SIGNALIZED URBAN SUB-NETWORKS

Tobias Pohlmann, Technische Universität Braunschweig, Institute of Transportation and Urban Planning, Rebenring 31, 38106 Braunschweig, Germany, t.pohlmann@tu-braunschweig.de

Bernhard Friedrich, Technische Universität Braunschweig, Institute of Transportation and Urban Planning, Rebenring 31, 38106 Braunschweig, Germany, friedrich@tu-braunschweig.de

ABSTRACT

Different Adaptive Traffic Control Systems (ATCS) for traffic signal control in urban networks have been developed over the past years and even decades, e.g. SCOOT, SCATS, BALANCE and MOTION. The improvement of traffic modeling techniques and increasing computing power promote enhancement and further development of sophisticated ATCS systems. This paper presents the prototype of a newly developed ATCS and the results of a microsimulation study assessing its performance. The strategy is designed for use in interconnected urban sub-networks containing several signalized intersections. It optimizes signal plans and coordination patterns of consecutive time intervals of 15 minutes. These signal plans can then be used at intersection level as continuously updated fixed time plans or even as framework plans in traffic responsive controllers. Four modules are executed once per time interval: forecasting of detector counts, demand estimation, cycle length and green split optimization, and offset optimization. The modules are described in this paper with emphasis on the latter.

Keywords: adaptive traffic control system, signal plan optimization, offset optimization, genetic algorithm, cell transmission model

INTRODUCTION

Adaptive Traffic Control Systems (ATCS) aim at optimizing traffic signal control in real-time by continuously adapting signalization at intersections to the current traffic demand. In general, the major goal is to optimize a certain performance index (PI) such as minimization of overall delay, number of stops or fuel consumption or maximization of throughput or combinations thereof. Either arterials or (sub-)networks comprising several signalized intersections are considered. Well known ATCS (among others) are SCOOT and SCATS, which are well established nowadays. In Germany, mainly MOTION and BALANCE share the market. Many other ATCS with different philosophies and modes of operation exist or are about to be developed. Improvement of traffic modeling techniques and increasing computing power promote such an enhancement and further development of sophisticated ATCS.

This paper presents the prototype of a newly developed ATCS. It has been motivated by previous projects with promising results conducted at the authors' research institution, one dealing with traffic demand estimation (Friedrich/Wang, 2006, Wang, 2008), the other with model-based offset optimization for signalized networks (Almasri, 2006). The algorithms and ideas of both projects have been combined and enhanced to form the ATCS prototype.

After a literature review, the conceptual design of the ATCS is presented. All modules for traffic demand forecasting and estimation and signal setting optimization will be described. The main focus is on the offset optimization that utilizes the Cell Transmission Model (CTM) (Daganzo, 1994, 1995) as a macroscopic traffic flow model to evaluate the performance of different offset combinations and a Genetic Algorithm (GA) as optimization method. The second half of the paper presents a microsimulation-based evaluation of the ATCS using a real test network. This gives an idea of the potential of the ATCS, but also of some remaining weaknesses.

LITERATURE REVIEW

Even though the demand estimation modules will be roughly sketched, this paper emphasizes on the model-based offset optimization using a GA. Therefore, the literature mentioned below focuses on this topic.

Presumably the first experimenting with GAs in order to optimize signal timings were Foy et al. (1992). Using a simplistic traffic flow simulation model as fitness function to evaluate different signal settings, they optimized green allocation at four intersections of a simple grid network in order to reduce total delay. They highlighted that the purpose of the study was to assess the potential of GAs in traffic signal optimization. No comparison between the model and real traffic has been done. Park et al (1999, 2000) tested GA-based optimization of signal timings at oversaturated intersections. They used different objective functions for throughput maximization, average delay minimization and delay minimization with a penalty function for movements with volume-to-capacity ratios greater than 0.9. A mesoscopic simulator has been used to evaluate different signal settings within the GA framework. They optimized cycle length, offsets, green splits and phase sequences simultaneously and used an artificial arterial with four intersections. CORSIM was used to compare the optimized signal plans to a reference case created with TRANSYT 7F, the latter performing worse. Computation time was about 23 minutes for one optimization run, but it has to be considered that computer technology evolved tremendously since then.

The first one to use the CTM for modeling signalized intersections was Lo (1999, 2001). He transformed the model equations into a mixed integer programming approach to minimize total network delay. He optimized durations of signal phases and applied the method to an artificial, simple network with two intersections and only one-way streets without any turnings. Even though the method worked, solution time was very long, impeding its application on larger networks. Therefore, Lo et al (2001) tried to use the CTM as fitness function of a GA. They optimized green times for a real one-way street network with three

intersections. They compared the resulting signal timings to those produced by TRANSYT 7F by simulating both with the CTM and concluded that the performance of their method was better. Further studies of this method can be found in Lo et al. (2004).
Almasri (2006) used the CTM for assessing different offset combinations of signalized intersections in different networks of varying size. He, too, included the CTM as fitness function into a GA framework. He used different test networks and evaluated the optimized offsets by means of microsimulation, in this case with AIMSUN. Reference cases have been developed using other techniques including TRANSYT 7F, but they were all outperformed.
All of the studies mentioned so far investigated on offline optimization, i.e. signal settings of varying kind have been optimized for a given demand. Braun et al. (2008) are the first to report on implementation of GA into a real ATCS. They replaced the old hill-climbing optimization previously used in BALANCE by a GA and tested the enhanced ATCS in a real network in Ingolstadt, Germany. They state that travel time could be further reduced by 10 percent compared to the original BALANCE, as a field test revealed.
BALANCE uses a traffic model to assess signal settings that is different from the CTM. No real-time application using GA in combination with the CTM existed so far. Building on the work of Almasri (2006), this gap has been closed by implementing the prototypical ATCS that is presented in this paper.

CONCEPTUAL DESIGN

Overall system

This paragraph provides an overview of the general structure of the ATCS and its different modules and enables a basic understanding of how the system works (cp. figure 1). The following paragraphs explain the modules more in detail with clear emphasis on the model-based offset optimization. All modules have been implemented as an object-oriented program written in Java.

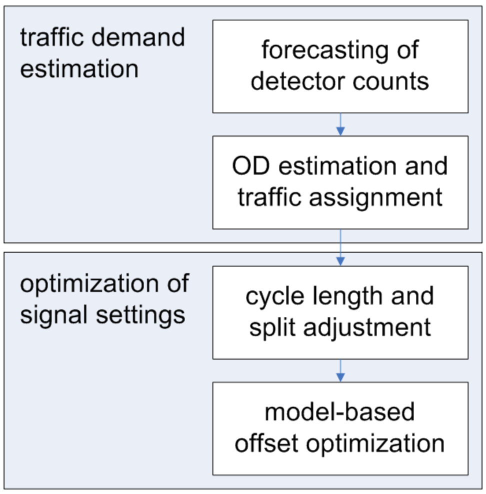

Figure 1 – modules of the system

The prototype of the ATCS is designed for use in urban sub-networks containing several signalized intersections. It can explicitly handle not only mere arterials but also arbitrary networks with intersecting or even meshed roads and competing traffic streams. The strategy optimizes signal settings for consecutive time intervals of 15 minutes, i.e. every 15 minutes a new optimization process is initiated.

The optimization algorithm needs the traffic demand that signal settings shall be optimized for as input data. Therefore the upcoming traffic demand has to be estimated. This task is divided into two steps. In a first step the expected detector counts of the next optimization interval are forecasted. Special detector locations are not required. However, some remarks on preferable detector locations will be made later. At this point it is only important to understand that the network is represented internally as a directed graph with separate links not only for sections connecting the intersections but also for all internal turning movements at all intersections (i.e. intersections are not represented as a single node only) and that at least some of the links have to be equipped with detectors measuring the flow on these links.

When a new optimization run is initiated, all available detector counts from the previous four time intervals are used to forecast the detector counts of the optimization interval. The optimization starts right after the previous interval has ended. Since the optimization consumes some computational time in the range of several minutes during which a portion of the next time interval elapses simultaneously the optimization has to be done for the subsequent time interval (cp. figure 2). Therefore, based on the counts of the four preceding time intervals the forecasting module has to look two intervals ahead instead of only one.

Once counts for all links equipped with a detector are forecasted, a module combining OD matrix estimation and traffic assignment uses these counts as constraints in a second step. Consistent traffic volumes on all links in the network can be derived as a result. Thus, an estimate of all source inflows into the network and of all turning percentages at all intersections is now available for the optimization interval and can be used as the desired input data for the optimization algorithms.

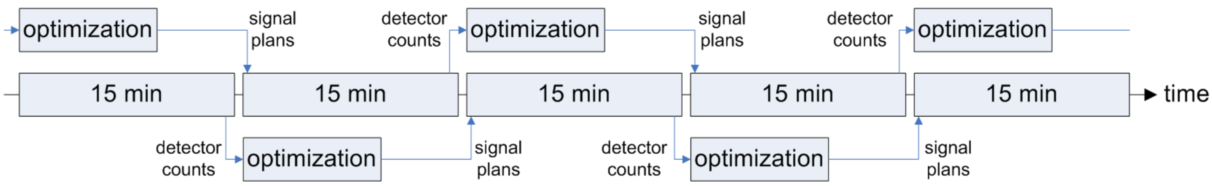

Figure 2: time flow of optimization of consecutive 15-min-intervals

The next module adapts cycle length and green splits to the forecasted traffic demand using a simple non-model-based approach. A common cycle length for all intersections is used depending on the most heavily loaded intersection. A more sophisticated module of the optimization process performs a model-based offset optimization in order to identify the currently best possible coordination pattern. The CTM in combination with a GA is applied.

The resulting signal plans are used as continuously updated fixed time plans. They are sent to the controllers at the end of each time interval or at the beginning of the next interval respectively. It is also conceivable to use these fixed time plans as framework plans in traffic responsive controllers. However, this option has not been tested yet.

Forecasting of detector counts

As has been sketched in the previous paragraph, the first module executes a forecast of detector counts to be expected during the next optimization interval. Detectors may be of any type as long as they provide vehicle counts. The currently still most common detectors in urban networks are loop detectors, but other technology becomes more and more widespread as well.

The presented ATCS method does not necessarily depend on special detector locations. However, directly detected turning flows at intersections are preferable because they contain the most precise constraints for the subsequent traffic demand estimation for the whole network. For the case of Germany and other countries as well, loop detectors can be expected to be located on all lanes approaching a signalized intersection. If lanes reserved for a specific turning movement (separate left-turn or right-turn lanes) exist, traffic volumes of these turning movements can be detected directly. In many cases there are only mixed lanes, and therefore only counts of combined thru and turning flows are available.

The forecasting technique employed in this work is based on a method proposed by Förster (2008) using current and reference space-time-patterns of detector counts. The method has been further enhanced and slightly adapted as described by Pohlmann/Friedrich (2009a).

Basically, two-dimensional reference space-time-patterns of average detector counts are used to forecast traffic counts of all detectors in a sub-network simultaneously. The reference patterns are derived from available data collected in the past. The data is clustered into several relevant groups such as different weekdays, Sundays, holidays etc. Each reference pattern consists of reference values for each detector and each 15-minute-interval of the day. In order to forecast the counts of the next optimization interval the real detector counts that have been observed during the last four preceding time intervals (covering one hour) are used as the currently observed traffic count pattern. Then, all reference patterns are scanned for a sub-pattern of the same size that matches best the current count pattern. A preferably high correlation coefficient r_{xy} in combination with a preferably low mean squared error MSE is used to identify the best matching sub-pattern within the reference patterns.

Once the best reference sub-pattern has been found the reference values of the time interval of interest following this sub-pattern are taken as forecasted values with some modification according to the remaining difference between the current pattern and the reference sub-pattern. As argued before, in this work the optimization interval is two time intervals ahead due to computing time issues, and therefore the interval in the reference pattern that has to be used must be the second interval following the identified sub-pattern.

The method turned out to be of satisfying quality in order to be used in the framework of the ATCS. For more details on the algorithm and its performance see Pohlmann/Friedrich (2009a).

OD demand, route and link volume estimation

Once the detector counts of the next optimization interval have been forecasted, they can be used to estimate the upcoming traffic demand (i.e. OD flows, route and links volumes) of this interval. The module is based on the work by Friedrich/Wang (2006) who employ the

information minimization (IM) model presented by van Zuylen/Willumsen (1980). The IM model uses available traffic counts on several links as constraints for the estimation.
For best estimation results the constraints should be as detailed as possible. The most detailed information would be forecasted traffic volumes for all turning flows (left, thru, right) at all approaches of all intersections. As has been mentioned before, however, not all turning flows at intersections are measured independently or at all. To enhance the information obtained from the available detector counts a simple iterative algorithm is proposed by Pohlmann/Friedrich (2009b) that derives as many missing counts on links without detectors as possible. Since the forecasted detector counts and the derived link volumes cannot be expected to be entirely consistent, another method proposed by van Zuylen/Branston (1982) is applied in a next step to overcome inconsistencies. An iterative algorithm balances the inconsistent traffic flows and finds a maximum likelihood estimate of consistent link volumes.
Wang (2008) dealt with the problem of a reduced performance of the IM model if redundant information is used as constraints, a drawback that had been described earlier by van Zuylen (1981). Several rules have been defined by Wang (2008) and Friedrich/Wang (2006) in order to eliminate redundant information while maintaining the most precise constraints possible. Applying these rules to the network before starting the IM model assures that no redundant information is used as constraints. A general algorithm covering all of these rules has been developed and used in this work in order to improve the demand estimation.
Another problem that Wang (2008) addressed is that for each link the IM model needs the portions of trips from each origin to each destination passing this link. This portion is generally not known precisely in practice. Therefore, Wang (2008) tried different traffic assignment techniques to estimate these portions and found that stochastic user equilibrium using the C-Logit model (cp. Cascetta et al., 1996) in combination with the method of successive averages (cp. Sheffi, 1985) performed well as assignment technique. This approach has also been used in the work presented in this paper.
The final implementation of the module integrates the IM model and its modifications into an iterative procedure as proposed by Wang (2008). Starting with a unit matrix in the first step, repeated traffic assignment and matrix estimation is executed until the estimated matrix converges against a stable solution. The final matrix and the assignment results of the last iteration step are used to determine the estimated flows on each predefined route and on each link in the network. A more detailed description of the module and its performance is given by Pohlmann/Friedrich (2009b). Again, the results suggested that the method performs well enough to be used in the framework of an ATCS.

Cycle length and green split optimization

After forecasting and demand estimation have been done, estimated traffic volumes are available for each link in the sub-network and thus also for each turning movement at each signalized intersection. These flow volumes can be used to determine an appropriate cycle length and adapted green splits. The signal phases as well as the phase sequences at each intersection are considered to be constant in this work. They have to be pre-planned offline.
While other research projects tried to use heuristics to optimize the whole range of signal plan settings (e.g. Braun et al., 2008) the authors of this abstract preferred to use an analytical method for the optimization of cycle length and green splits. It is assumed that at

least for undersaturated conditions the classic approach of splitting green times among phases according to the relevant ratios of flow to saturation flow is reasonable in order to reduce delays. The adjustment of cycle length has been implemented in two ways, so that either the Webster formula (Webster, 1958) or a saturation based approach as proposed in the Highway Capacity Manual can be used. Strictly speaking, the first can only be applied at isolated intersections, and thus the latter should be preferred for coordinated roads and networks. Therefore, in a first step, different cycle lengths t_c and green splits (phase duration $t_{g,i}$ of phase i) are determined for each intersection separately according to the following well-known formulas:

$$t_c = \frac{T_L}{1 - B/x} \qquad (1)$$

$$t_{g,i} = \frac{b_i}{B} \cdot \left(t_c - T_L\right) \qquad (2)$$

where T_L = total lost time,
$b_i = v_i / s_i$ = relevant ratio of flow to saturation flow of phase i,
B = sum of b_i,
x = desired degree of saturation.

If any of these initial phase durations is lower than the minimum green time of this phase, the cycle length is corrected upwards by using the following modified equation:

$$t_c = \frac{T_L + T_{g,\min}}{1 - B'/x} \qquad (3)$$

where $T_{g,min}$ = sum of minimum green times of all phases shorter than minimum green time,
B' = sum of b_i of all phases exceeding minimum green time.

If the final cycle length is shorter than a defined minimum cycle length, the latter is used instead. Once a cycle length has been determined for each intersection separately, the longest of these cycle lengths is chosen as common cycle length for the next optimization interval. A common cycle length has to be used to enable network-wide coordination of the intersections, which will be established in the following module.
Finally, phase durations of each intersection will be recalculated for the common cycle length according to equation (2), subject to minimum green times.

Offset optimization

Problem and approach

The main focus of the presented ATCS as well as of this paper is on the offset optimization, which poses the biggest challenge. A sophisticated module has been developed to deal with this problem.
While choosing offsets for arterials might be less difficult, it is more complex in interconnected networks. In a network with n intersections there are n-1 independent offsets (one intersection generally serves as reference with a fixed offset $t_{off} = 0$). If a common cycle length t_c is used for the whole network, offsets of each intersection can take a value between 0 and t_c-1. Therefore the solution space comprising all possible offset combinations has a size of $t_c^{(n-1)}$.
In order to assess the potential of a specific offset combination in terms of vehicle delay, an appropriate traffic model can be used. Since this module is based on the work by Almasri (2006), the CTM is used for this purpose. Due to the huge and highly irregular solution space the problem of offset optimization cannot be solved analytically or by brute force (trying every possible solution). Heuristic GAs are designed for this sort of problem because they can find solutions that are close the global optimum.
Almasri’s work resulted in an offline method identifying the presumably best offset combination (or at least a good one) for a given traffic demand. In the work presented in this paper, his method has been transformed into an online application. A flow chart of the developed module for offset optimization is shown in figure 3. Details are given in the following paragraphs.

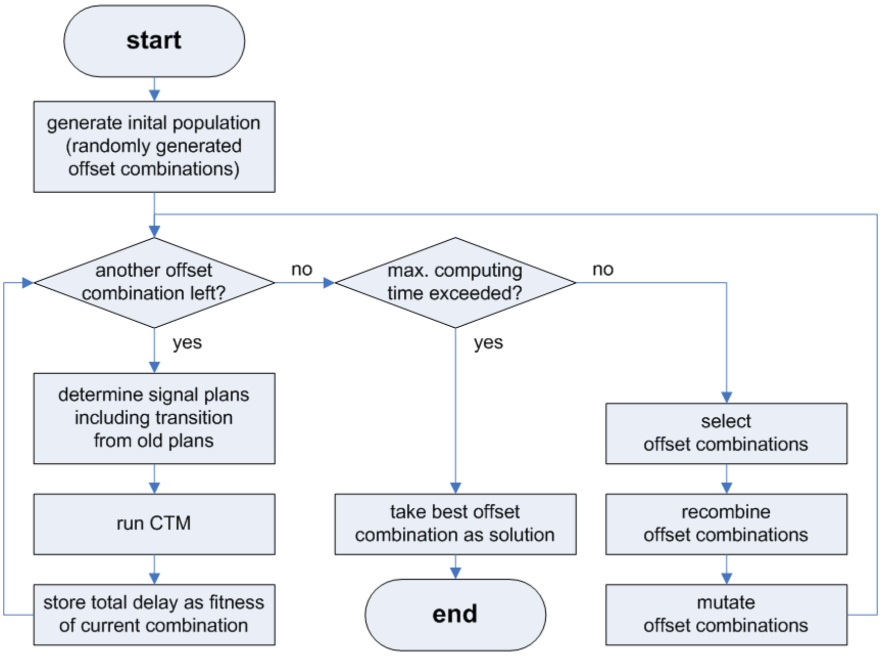

Figure 3: flow chart of offset optimization

Cell Transmission Model

The CTM serving as fitness function in this work has been initially proposed by Daganzo (1994, 1995) for highway applications. However, it can also be used to model urban traffic. It is a space and time discrete version of the macroscopic, hydrodynamic LWR model using a trapezoidal simplification of the fundamental diagram. Each link is divided into a finite number of cells that can contain a limited number of vehicles. The length of a cell is the ratio of free flow speed v_f [m/sec] and duration t_{sim} [sec] of a simulation step. This guarantees that no vehicle can pass more than one cell during one simulation step. With v_f = 13.8 m/sec (50 km/h) and t_{sim} = 1 sec (as used in this study), each cell has a length of 13.8 m which gives an idea of the spatial resolution of the model. The maximum number of vehicles a cell can contain is determined by the ratio of cell length and jam density.

Without presenting all the model equations in detail (see Daganzo, 1994 for further insights), basically, in one simulation step t the inflow $q_{in}(t)$ and the outflow $q_{out}(t)$ of each cell are calculated and the number of vehicles n(t) in the cell is updated afterwards. Besides ordinary cells the original CTM also provides special cells to diverge from one into two cells (1:2) or to merge from two cells into one (2:1). Flötteröd/Nagel (2005) proposed a more general algorithm that allows to connect any number of cells to any other number of cells (n:n). However, this algorithm must be considered to be relatively time consuming. Since the ATCS is supposed to operate in real-time, a short computing time is a crucial requirement. Therefore, only a modest extension of the CTM allowing 1:3-diverges and 3:1-merges has been implemented in this study, but even complex intersections can be modeled by means of this extension. Figure 4 shows an example of such an intersection. However, it might be necessary to further extend the model for some special cases that have not been considered yet.

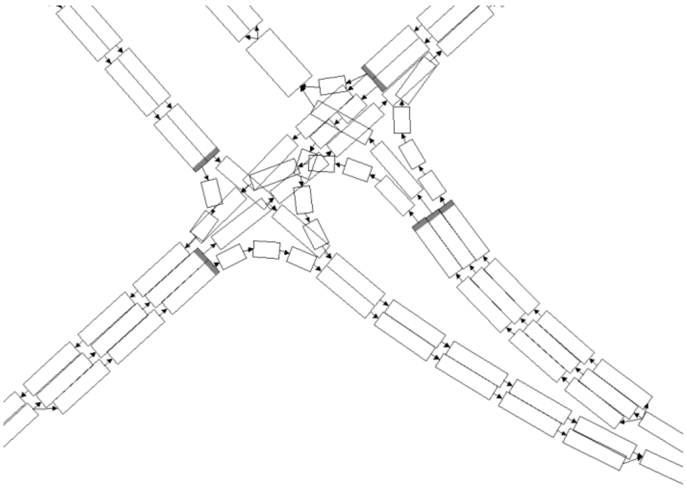

Figure 4: a complex intersection modeled with the CTM

In order to model signalized intersections, special cells have been included whose outflow is set to zero if the according traffic signal is red during the current simulation step. If the signal is green, the regular model equations are applied. Further enhancement of the CTM has been done by implementing simple rules to consider permitted left turning movements, i.e. the outflow of a cell belonging to a minor traffic stream is reduced according to the current traffic flow of cells having priority. This extension can also be used to model mere priority junctions without any traffic signals at all.

As has been said before, the CTM is used to estimate the delay imposed on the traffic by the current signal settings or, more precisely, the offsets that create a certain coordination pattern.
With a simulation step of 1 second the total delay of the vehicles in a cell during one step is

$$d(t) = n(t-1) - q_{out}(t) \qquad (4)$$

This is equivalent to the number of vehicles that have already been in the cell at the beginning of the simulation step but could not leave it during the step. By summing up this value for all simulation steps of a simulation run and for all cells of the network the total delay is obtained in veh·sec and can be used to indicate the fitness of the currently examined offset combination. The smaller the total delay the better the offset combination.

Genetic Algorithm

The CTM has been integrated as fitness function into a GA. The open source Java Genetic Algorithms and Genetic Programming Package (JGAP) provided by Meffert et al. (2009) has been used. GAs attracted major interest in traffic related applications during the last years. They mimic the evolutionary process by starting with a randomly generated set of solutions (cp. figure 3). These solutions are assessed, selected (with a higher chance for solutions having a better fitness), recombined into a next generation and mutated randomly (with a rather low probability). The whole process is repeated with this new generation of solutions and the following generations and so on until the stop criterion is met. This stop criterion might be that the best solution within a generation does not improve anymore over a certain number of generations, that a maximum number of generations has been reached or that the available computing time is exceeded. The last option has been used in this study.
In the context of this work, a solution, also referred to as chromosome, is a combination of different offsets for each intersection. Each offset can be considered as a gene of the chromosome that can take any integer value between zero an t_c-1. The signals in the CTM have been programmed in such a way that they switch during a run according to the previously adapted cycle length and green splits and the currently assessed offset combination. The traffic demand is taken from the estimation for the next optimization interval. The implementation of the CTM turned out to be sufficiently fast to enable the GA to test hundreds of different offset combinations within only a few minutes. Details on the runtime will be given later after the test network has been presented.

Traffic Signal Transition

Since the ATCS repeatedly sends new signal plans to the controllers for each optimization interval, a signal plan transition at each intersection has to occur at the beginning of each time interval in order to switch from the previous to the new signal plans. If intersections are not coordinated, this task is rather simple. The previous signal plan can be left at a break point and the new plan can be started immediately at an according point showing exactly the same signal indication. If intersections are coordinated, however, the new signal plan of a controller has to be shifted to an amount of seconds in the range of zero to t_c-1 in order synchronize with the other intersections. Otherwise, the new coordination pattern cannot be established.

A study assessing different transition techniques has been conducted by Pohlmann/Friedrich (2010) explicitly for the case of a meshed network. The results are in perfect accordance with previous studies (Shelby et al., 2006, Cohen et al., 2007) investigating on transition effects along coordinated arterials. Simple dwell methods that simply hold the coordinated phase as long as necessary to get the intersection in sync may induce major peaks in delay especially for non-coordinated side street traffic. Smooth methods using several shortened or lengthened transition cycles perform far better. The finally recommended transition method which is also used in the ATCS decides for each intersection individually whether to shorten or lengthen cycles, whichever finishes transition faster subject to a maximum shortening or lengthening per cycle of 20 percent of the current cycle length. Thus, no more than three transition cycles are needed, guaranteeing a smooth and quick transition into the new coordination pattern.

Of course, this transition may also influence the overall performance of a certain offset combination. If transition is not considered when testing an offset combination, the combination may perform better in the CTM than it finally does in reality because the necessary transition may reduce its otherwise good performance. Therefore, the CTM has been programmed in such a way that it also simulates the necessary transition from the old signal plan to the new one, shifted according to the currently tested offsets. So, the fitness or total delay of an offset combination also contains possible additional delays due to transition effects. Two solutions may perform equally well if transition is not considered, but if one of these solutions suffers from major transition disturbances, the GA is less likely to choose this solution over the other if transition effects are taken into account.

As has been said, the offset of one controller is typically kept at zero and serves as a reference for the other offsets. If transition is considered, however, all offsets should be free to take any valid value. A specific offset combination might have a detrimental effect during transition. The same combination, only shifted some seconds (i.e. the same value is added to all offsets which does not change the coordination pattern at all), might have a completely different and maybe less disruptive effect during transition. However, if one offset is always fixed at zero, this second combination is not part of the solution space. To overcome this effect, in this work the GA is allowed to modify the offsets of all controllers without exception. No reference offset at one intersection is used.

EVALUATION

Simulation setup

An extensive simulation study has been conducted in order to assess the newly developed ATCS prototype. The microsimulation software AIMSUN NG 5.1.8 has been used for this purpose. Its convenient application programming interface (API) in C++ allows the user to implement and test own control strategies. Among other things the API offers functionalities to read detector data at regular intervals and to directly change the current state of any signal group. Therefore, the ATCS could be integrated into the simulation environment. A special module has been programmed in C++ that can communicate with the main application of the ATCS, which is written in Java as mentioned before.

An urban sub-network in Hanover, Germany, with eight signalized intersections, two pedestrian lights and two non-signalized intersections has been chosen for testing. The network as modeled in AIMSUN is shown in figure 5 (left side). A total of 55 detectors are located on the lanes in front of traffic signals. Their data can be used for traffic demand forecasting and estimation as described earlier. Since there are also mixed lanes for more than one turning movement as well as entirely non-signalized intersections, some turning flows are not detected directly.

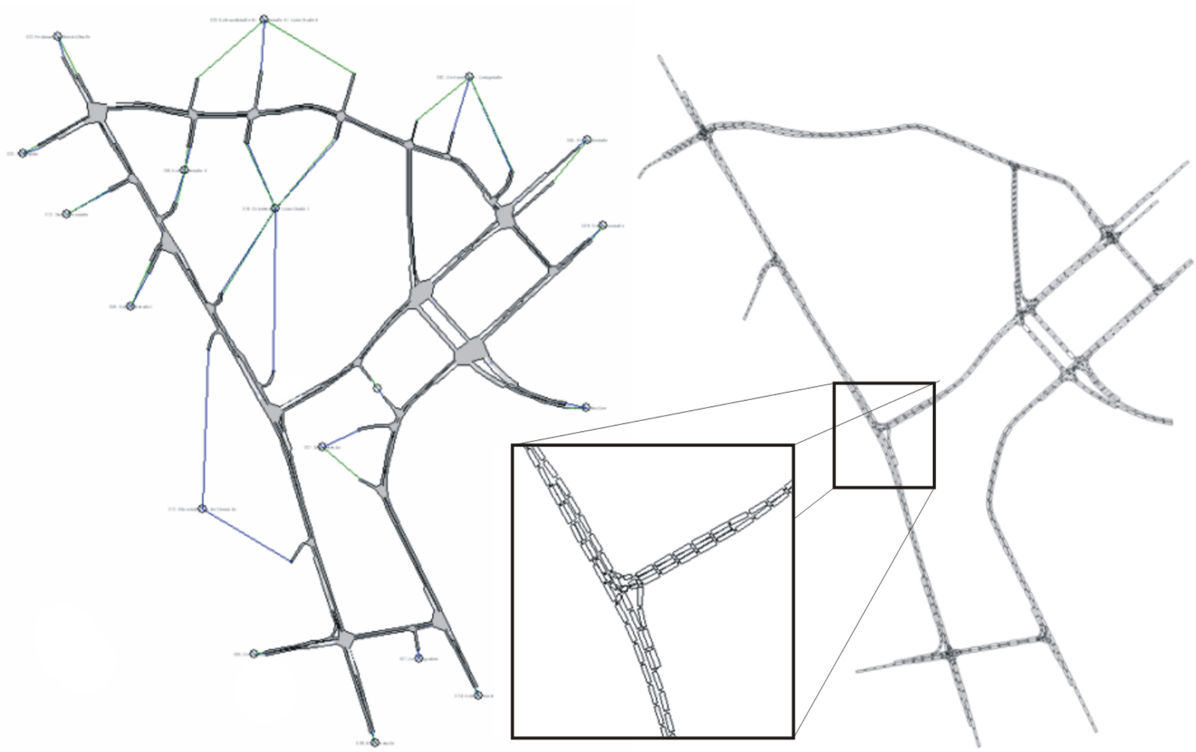

Figure 5: test network in AIMSUN NG (left) and as CTM representation

The simulation comprises the time period between 6 am and 8 pm on a regular weekday, i.e. 14 hours have been simulated in a row. Within this time period two peak hours from 8 to 9 am and from 5 to 6 pm occur with periods of lower traffic demand before, between and after those peak hours.

The traffic demand finally used for the study is semi-fictitious. For each of the 15-minute-intervals of the simulation, comprising a total of 56 intervals, a separate OD matrix has been fed to the simulation. This has been done in order to reflect the changes of traffic demand over a day that the ATCS is supposed to react to. The matrices have been deduced from real measurements on-site. These measurements include loop detector counts from a specific day, kindly provided by the city of Hanover, as well as data from six radar detectors that have been additionally installed at strategic positions on that day. Since the data from these detectors was not entirely consistent due to measuring errors and sporadic complete failure of single loops, some assumptions had to be made afterwards to correct the data.

The available data contained only traffic flows at detector locations whereas the input data of the simulation are OD matrices. Therefore, the detector data had to be transformed into OD matrices. 108 plausible alternative routes between 71 reasonable OD relations connecting 9 origins and destinations have been selected. The percentages of usage of these alternative routes have been set manually in such a way that they appeared reasonable and at the same time reproduce the real measurements at detector locations as well as possible when assigning the OD matrices to the network accordingly. To facilitate the process of OD matrix generation, only one matrix for the morning peak interval of 15 minutes and one for the afternoon peak interval of the same duration have been deduced in the first place. The OD matrices for all other intervals are linear combinations of these two peak interval matrices and reflect the overall profile of traffic demand over time as observed on-site in an acceptable way.

The main streams during morning peak hour are those going to the city centre (i.e. relations North-South and East-South) whereas the afternoon peak hour traffic rather tends to leave the city centre. However, the respective opposite directions also have high traffic volumes during both peak hours.

For first test runs and for calibration purposes the fixed time signal plans as used in reality at the modeled intersections have been implemented in the simulation. It turned out that with the given demand none of the intersections came even close to saturation even during peak hours. Therefore, the entire daily demand has been increased successively up to 25 percent. This leads to intersections close to saturation during both peak hours. However, oversaturated time periods have been avoided. The prototype in its present form is not able to handle oversaturated traffic conditions. Instead of the real demand, the loop detectors would only measure lane capacity during oversaturated periods and this would compromise the demand estimation.

Figure 6 shows the profile of the overall traffic demand that has been finally used for simulation. It displays the average number of vehicles entering the network during each of the 15-minute-intervals and has been derived from 30 simulation runs with different random seeds and fixed time traffic signal control according to the real signal plans.

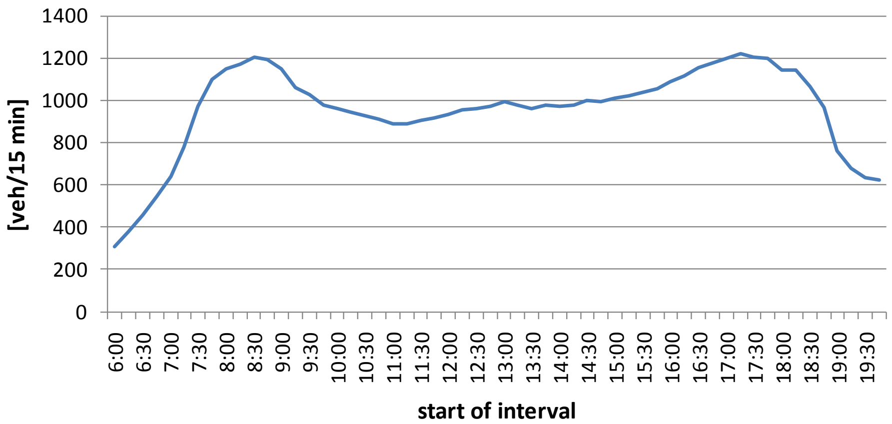

Figure 6: overall traffic demand used in simulation study

Reference case

When assessing a new ATCS, an adequate and incontestable reference case is needed. On the one hand, it would be best to compare the prototype to an existing ATCS. On the other hand, the software cores of existing ATCS are generally not publicly available. Therefore, fixed time signal plans are often used as reference case. When doing so, the fact that a fixed time control does not react to a changing traffic demand must be born in mind when discussing the comparative results.
In this work, a fixed time control has been used as reference. However, the existing fixed time control plans could not be used because they might be either outdated or optimized for a traffic demand that differs more or less severely from the one used in the simulation. As has been said, the traffic demand of the simulation is semi-fictitious and may not represent the real traffic demand in all details. Therefore, two fixed time control plans for both morning and afternoon peak hour have been generated using the established TRANSYT 7F tool. They have been optimized for traffic demand during peak hours as used in the simulation. The original phases and phase sequences as used in the real network have been used in most cases with only slight adaptations at some intersections. Intersections in the network have either two or three phases. TRANSYT 7F was allowed to optimize the common cycle length, green splits and offsets of all controllers including the two pedestrian lights. The inbuilt GA has been used for optimization, and total delay has been used as performance index to assess possible solutions.
Both resulting signal plans have a cycle length of 90 seconds. As in the real network, transition from morning to afternoon signal plans takes place at 1 pm, i.e. the first plan is used from 6 am to 1 pm and the second plan from 1 pm to 8 pm.

Specifics of offset optimization

Since the network contains 8 signalized intersections and 2 pedestrian lights which are also included into the coordination, the ATCS has to optimize 10 offsets simultaneously. A chromosome thus consists of 10 integer genes taking values between zero and the common cycle length minus 1, the latter determined by the previously executed module for cycle length and green split adjustment.
A CTM representation of the network had to be generated to be used as fitness function. It is shown in figure 5 (right side). It consists of 841 cells and can simulate a time period of 15 minutes in 52 ms on average on an Intel Core i7-920 quad-core processor with 2.67 GHz using only one processor. (The prototypical implementation has not been designed for parallel computing on two or more processors.) Neglecting the overhead of the GA itself, approximately 5.700 different offset combinations, each taking a single run of the CTM, can be tested within 5 minutes.
The GA library JGAP that has been used in this project allows altering different parameters according to the needs of the respective application. A variety of different settings has been tested in order to achieve a good performance of the GA when optimizing offsets. Different selectors selecting the chromosomes or solutions to be passed to the next generation according to their fitness have been tried out. Tournament selection with three competitors turned out to perform best. This selector randomly chooses three offset combinations from

the candidate pool, i.e. the current generation. The one with the best fitness is transferred to the next generation, and the other two solutions are reinserted into the candidate pool.

The crossover rate specifies the percentage of solutions in the next generation that are produced by mating two parent solutions from the previous generation. Single crossover has been used, i.e. the new offset combination starts with a series of offsets from the first parent and ends with a series of offsets from the second parent. The split point is chosen randomly. The crossover rate has been set to 0.35. This implies that 65 percent of the solutions of a generation are mere clones or copies of solutions from the previous generation.

The mutation rate has been set to 12. In JGAP the mutation rate is an integer value that indicates that every n^{th} gene is mutated randomly. Population size, i.e. the number of solutions in one generation, has been set to 50. The so called elitism functionality has been enabled. It guarantees that the best solution of a generation is always passed to the next generation. Most of the finally chosen settings turned out to be those recommended by Meffert et al. (2009). All in all the optimization was not highly sensitive to different parameter settings.

Figure 7 shows some examples of how the fitness of the best individual per generation evolves over time. Since GAs show a random behavior by nature, each optimization run is different. Therefore, several runs have to be considered to assess the overall performance. All six runs displayed in figure 7 optimized offsets for the same demand and used the same signal plans that have been determined before.

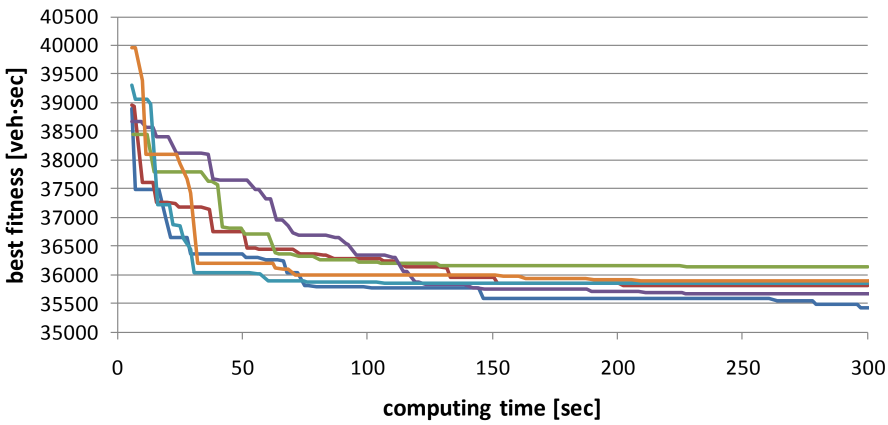

Figure 7: evolution of best fitness during six optimization runs

It can be seen that after about 150 seconds no further significant improvement of the best offset combination can be achieved. In order to be on the safe side, the final maximum computing time has been limited to 300 seconds. A further extension of the computing time does not lead to further improvement of the final solution.

The offset optimization is the most time consuming of all modules of the ATCS. Computing time for forecasting, demand estimation and cycle length and green split adjustment is in the range of only a few seconds. Therefore, executing all modules of the ATCS for one optimization interval takes about 5 minutes, whereas the available computing time would be 15 minutes which is the duration of each optimization interval (cp. figure 2). So, available resources are not entirely exhausted and larger networks with three times more cells in the CTM representation can also be handled. Another alternative would be to optimize signal plans not every 15 but every 5 minutes. However, this is not advisable due to the necessary signal plan transitions at the beginning of each time interval which may also take a few minutes. This might lead to a network that is more or less constantly in transition.

Results

Optimization for exact demand

Different approaches have been tested by simulation. In the first test case the exact traffic demand of each time interval has been used as input data for optimization. The motivation behind this was to assess the capabilities of the two optimization modules without any impairment caused by a possibly imprecise demand estimation.
Firstly, 30 simulation runs using the real fixed time signal plans have been executed. During each run the traffic volumes on all routes have been logged for each 15-minute-interval. Based on this data, average route volumes for each interval have been calculated afterwards and stored in a file.
Secondly, the optimization modules have been executed repeatedly for each of the 56 optimization intervals. During each execution the average route volumes of the respective time interval have been read and used to calculate average link volumes. Based on these volumes the optimization modules adapted the signal settings and stored the resulting switching sequences of all signal groups in another file. Optimizing signal settings for all time intervals one after another took about 4:40h (5 minutes per interval). This process had to be done only once. The phase sequences used for optimization were the same as those of the reference signal programs generated with TRANSYT 7F. This also applies to the other two test cases described later.
Thirdly, 30 simulation runs using the stored, optimized switching sequences have been carried out. AIMSUN has been programmed via the API to read the relevant switching sequences at the beginning of each time interval and to execute them accordingly.

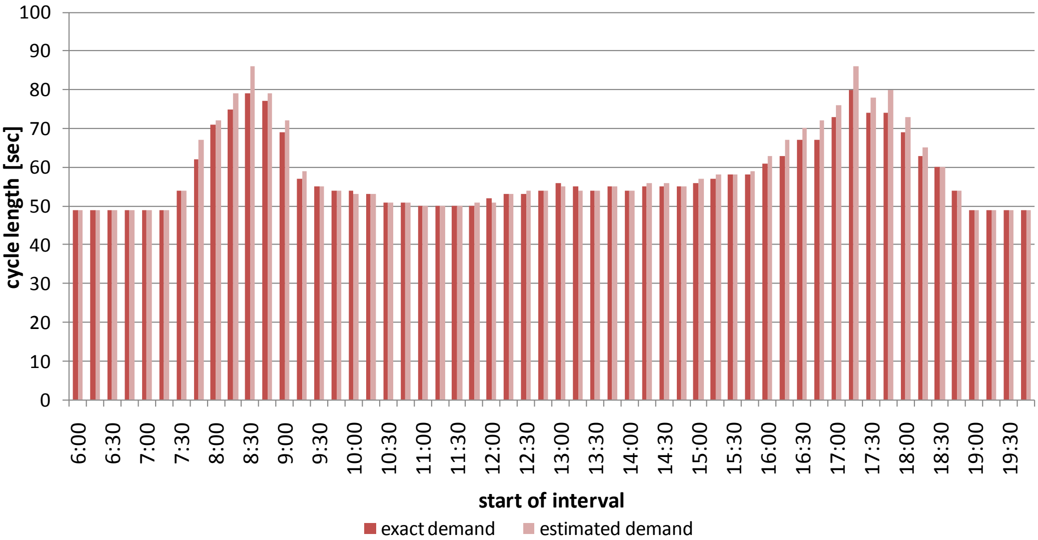

Figure 8: optimized cycle lengths for exact and estimated demand

Figure 8 shows the adapted cycle lengths of each time interval for this test case in dark red. The constant cycle lengths of 49 seconds during the first six and last four intervals correspond to the minimum cycle length dictated by the intersection with the largest sum of minimum phase durations and total lost time. As traffic increases, the cycle length is extended to up to 79 and 80 seconds respectively at peak intervals. This is still 10 seconds below the cycle length of the TRANSYT 7F optimized fixed time signal plans.

Figure 9 compares the network-wide average travel times (expressed in seconds per vehicle and km) of the test case (blue line) to the reference case (purple line). The peak at the 1pm-interval in the reference case is caused by a rather abrupt transition from the morning to the afternoon signal plans at the beginning of this interval. The ATCS produces smaller travel times than the reference case. For off-peak intervals the ATCS has the clear advantage of adapted cycle lengths which must be considered to be the main cause of reduced travel times. Nevertheless, offset optimization further contributes to this reduction. If offset optimization is turned off, which has also been tested, travel times are 5 to 15 seconds higher, depending on the interval. In this case, the TRANSYT 7F signal plans perform better during morning peak hour and comparably well during afternoon peak hour. It can be assumed that using a lower cycle length for the reference case instead of the obviously rather generous 90 seconds would reduce the discrepancy between the test case and reference case during peak hours.

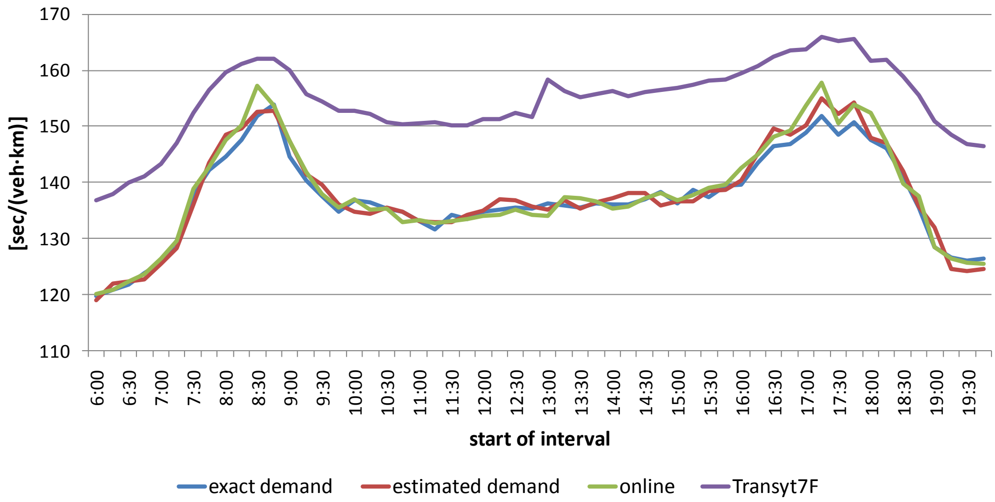

Figure 9: comparison of average overall travel times between test cases and reference case

A closer look on differences of travel time between test case and reference case is given in figure 10. Each row corresponds to one time interval. Morning and afternoon peak hour intervals are framed. Each column corresponds to one of the 108 routes. Routes are ordered by traffic volumes from left to right, starting with the highest volumes. This is illustrated by the upper three-rowed bar. The first row shows traffic volumes of the morning peak interval, the second of the afternoon peak interval, and the third shows the average traffic volumes of both. The pattern below shows improvements and degradations of travel times for each route and time interval in green and red respectively. Gray cells indicate that no statistically significant difference could be observed between test case and reference case for this route and interval.

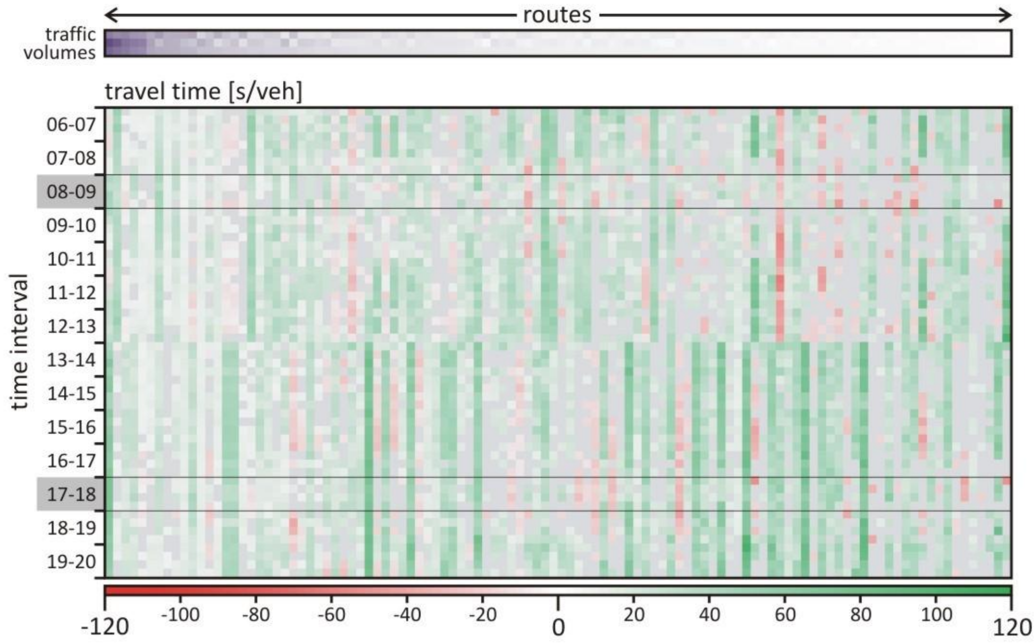

Figure 10: comparison of average travel times on all routes between test case 1 and reference case

As can be seen, improvements of travel time prevail, which is in accordance with the observations on overall travel times in figure 9. However, improvements on the heavier loaded routes are less intense than on less important routes. Obviously, the coordination pattern of the reference case is already of acceptable quality and the ATCS has not much room left for further improvement. But since volumes on the less heavily loaded routes add up to an important share of the total traffic demand in the network, major improvements on these routes also contribute largely to an improvement of the overall performance.

The faint horizontal line that can be seen in the middle of the pattern is again due to the change of signal plans in the reference chase, which also induces a change of the coordination pattern.

Optimization for estimated demand based on average detector counts

In the next considered case, the forecasting and demand estimation modules have been used in addition to the optimization modules. As has been described earlier, they rely on detector data as input data. In a first step, average detector data of each time interval has been used that has been produced in the same way as the average route volumes for the previous test case.

Again, the modules of the ATCS have been executed 56 times. For each interval, the detector counts of the four preceding time intervals have been read, serving as space-time-pattern to forecast the detector counts of the current optimization interval. Since the forecasting algorithm needs data from four previous time intervals and moreover looks two intervals ahead, the first optimization interval whose counts could be forecasted is the sixth time interval. For intervals 1 to 5 the average detector counts of these intervals have been used directly without any forecasting. The reference pattern needed for forecasting has also been derived from 30 simulation runs using the real fixed time plans. This implies that in this test case the quality of forecasting must be considered to be perfect.

Based on the forecasted counts the demand estimation and subsequent optimization of signal settings has been done. As for the first test case, switching sequences for all intervals have been stored and were read by AIMSUN at the beginning of each interval during each of the 30 simulation runs.

Figure 8 shows that due to imprecisions of the demand estimation the cycle length tends to be a little bit higher, especially during peak hours. Figure 9 reveals that in this test case (red line) travel times are slightly increased at some intervals. However, no highly relevant differences between test case 1 and 2 occur. The resulting travel time pattern is very similar to the one displayed in figure 10 and therefore not set out separately in this paper.

Real online optimization

This final test case represents an online application of the ATCS just as it would operate in a real network. No average data is used, but each simulation run in AIMSUN is optimized individually.

In this test case, AIMSUN and the ATCS exchange data directly during each simulation run. At the beginning of each time interval, AIMSUN gets the optimized switching sequences for each signal group from the ATCS. At the end of each time interval, AIMSUN provides the ATCS with recent detector counts of this interval. The ATCS stores these counts internally in order to generate space-time-patterns covering the four previous intervals. Again, the problem remains that no forecasting is possible for the first five intervals. For these five intervals the same average detector counts as in the second test case are used instead of forecasted counts.

The simulation in AIMSUN and the ATCS are executed in parallel. As has been said, execution of all four ATCS modules takes about 5 minutes for one optimization interval. AIMSUN simulates a 15-minute-interval in just a few seconds. This implies that AIMSUN spends most of the time waiting. The duration of a simulation run is thus dictated by the ATCS and amounts to the previously mentioned 4:40h. Therefore, only 10 simulation runs have been carried out in this case.

Since all simulation runs are optimized individually, the cycle length of single intervals varies, depending on the achieved quality of demand estimation. Figure 11 shows box plots including outliers (circles) and arithmetic means (black diamonds) of these variations. While variations are small during off-peak intervals, they are partly very severe during peak hours. No average detector counts are used as input data in this test case but those taken from single simulation runs with higher variations. This makes it harder for the forecasting module to identify the correct reference sub-pattern, and therefore the quality of the forecasted detector counts is reduced. This effect is more distinct, when traffic demand changes more rapidly. The reduced quality directly influences the performance of the subsequent demand estimation and the optimization.

On the average, however, subsuming all 10 simulation runs, the overall performance of the ATCS is still good and comparable to the other two test cases as figure 9 (green line) reveals. Again, a travel time pattern comparable to figure 10 has been produced.

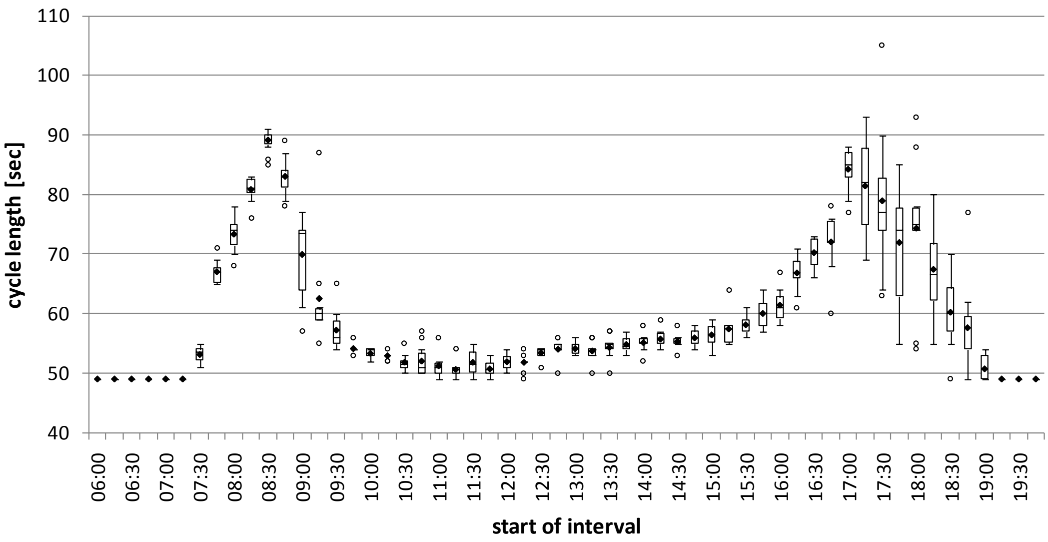

Figure 11: variation of cycle lengths for online optimization

CONCLUSIONS

The results of different test runs using microsimulation reveal that the prototype of the newly developed ATCS as presented in this paper has some potential to improve travel times in a sub-network. The overall improvement of travel times is clearly visible, whereas major improvements are more easily achieved on the less heavily loaded routes. Both cycle length adjustment and offset optimization contribute to this improvement and should be used as complementary modules. When offset optimization is switched off, the overall travel time is much higher which shows that the concept of combining GA with the CTM works well to optimize offsets.

The prototypical implementation of the ATCS proves that it is real-time capable. Optimizing signal settings for one time interval takes 5 minutes in the case study, whereas one time interval has a duration of 15 minutes, leaving an additional buffer of 10 minutes. The method can therefore also be used to optimize larger networks.

Some critical remarks have to be made as well. The reference case consisting of only two signal plans optimized for morning and afternoon peak hour respectively must be regarded with caution. On the one hand, the cycle length of 90 seconds seems to be a bit generous, since 80 seconds as chosen by the ATCS obviously do not provoke a breakdown during peak hour either. And on the other hand, major travel time improvements during off-peak hours had to be expected because the reference signal plans have not been optimized for this demand. In this regard, creating reference cases containing at least one or even more additional signal plans for the off-peak hours should be considered for future tests. The achievable performance of the ATCS in other networks should be examined as well.

The performance of the ATCS stands or falls on the quality of the demand estimation. If the estimated demand is of poor quality, even the best optimization algorithm will produce inferior results. The demand estimation used in this work obviously works well if average detector counts are used as input data. When single simulation runs are optimized, however, the variation of cycle length reveals that some partly major imprecisions remain. It also has to be highlighted that the method can only forecast detector counts according to the available reference space-time-patterns. Therefore, reference patterns for various days and special situations must be used and updated regularly. Future research will also have to find solutions to deal with oversaturated conditions that the demand estimation in its present form cannot cope with.

REFERENCES

Almasri, E. (2006). A New Offset Optimization Method for Signalized Road Networks. Dissertation at the Institute of Transport, Road Engineering and Planning, Leibniz Universität Hannover. Hannover, Germany.

Braun, R., C. Kemper and F. Weichenmeier (2008). TRAVOLUTION – Adaptive Urban Traffic Signal Control with an Evolutionary Algorithm. Proceedings 4th International Symposium Networks for Mobility. Stuttgart, Germany, September 25-26.

Cascetta, A., F.R. Nuzzolo and A. Vitetta (1996). A modified logit route choice model overcoming path-overlapping problems. Proceedings 13th International Symposium on Transportation and Traffic Theory. Lyon, France

Cohen, D., L. Head, and S. G. Shelby (2007). Performance Analysis of Coordinated Traffic Signals During Transition. In: Transportation Research Record (2035), 19-31

Daganzo, C. (1994). The Cell Transmission Model: A dynamic Representation of Highway Traffic consistent with the Hydrodynamic Theory. In: Transportation Research B 28 (4), 269-287

Daganzo, C. (1995). The Cell Transmission Model, Part II: Network Traffic. In: Transportation Research B 29 (2), 79-93

Flötteröd, G. and K. Nagel (2005). Some practical extensions to the Cell Transmission Model. Proceedings 8th International IEEE Conference on Intelligent Transportation Systems. Vienna, Austria, September 13-16

Förster, G. (2008). Kurzfristprognose auf Basis von Raum-Zeit-Mustern. Proceedings Heureka ’08. Stuttgart, Germany, 5 & 6 March.

Foy, M. D., R. F. Benekohal and D. E. Goldberg (1992). Signal Timing Determination Using Genetic Algorithms. In: Transportation Research Record (1365), 108-115

Friedrich, B. and Y. Wang (2006). Optimizing O-D Estimation with Respect to Redundant Information and Route Choice. Proceedings 11th IFAC Symposium on Control in Transportations Systems. Delft, Netherlands.

Lo, H. K. (1999). A novel traffic signal control formulation. In: Transportation Research A 33, 433-448

Lo, H. K. (2001). A Cell-Based Traffic Control Formulation: Strategies and Benefits of Dynamic Timing Plans. In: Transportation Science 35 (2), 148-164

Lo, H. K., E. Chang and Y. C. Chan (2001). Dynamic network traffic control. In: Transportation Research A 35, 721-744

Lo. H. K. and A. H. F. Chow (2004). Control Strategies for Oversaturated Traffic. In: Journal of Transportation Engineering 130 (4), 466-478

Meffert, Klaus et al. (2009): JGAP - Java Genetic Algorithms and Genetic Programming Package. URL: http://jgap.sf.net.

Park, B., C. J. Messer and T. Urbanik (1999). Traffic Signal Optimization Program for Oversaturated Conditions – Genetic Algorithm Approach. In: Transportation Research Record (1683), 133-142

Park, B., C. J. Messer and T. Urbanik (2000). Enhanced Genetic Algorithm for Signal-Timing Optimization of Oversaturated Intersections. In: Transportation Research Record (1727), 32-41

Pohlmann, T. and B. Friedrich (2009a). Combined Short-Term Forecasting and Traffic Demand Estimation for Online Control Strategies. Proceedings XIII Euro Working Group on Transportation Meeting. Padua, Italy, 23-25 September.

Pohlmann, T. and B. Friedrich (2009b). Short Term Estimation of OD Matrices and Traffic Streams in Urban Networks. Proceedings mobil.TUM 2009 International Scientific Conference on Mobility and Transport. Munich, Germany, May 12-13.

Pohlmann, T. and B. Friedrich (2010). Traffic Signal Transition in Coordinated Meshed Networks. Proceedings Transportation Research Board 89th Annual Meeting. Washington D.C., USA, January 10-14

Sheffi, Y. (1985). Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Shelby, S. G., D. M. Bullock and D. Gettman (2006). Transition Methods in Traffic Signal Control. In: Transportation Research Record (1978), 130-140

van Zuylen, H. J. (1981). Some Improvement in the Estimation of an OD Matrix from Traffic Counts. Proceedings 8th International Symposium on Transportation and Traffic Theory. University of Toronto Press, Toronto, Canada

van Zuylen, H. J. and D. M. Branston (1982). Consistent Link Flow Estimation From Counts. In: Transportation Research B 16 (6), 473-476

van Zuylen, H. J. and L. G. Willumsen (1980). The Most Likely Trip Matrix Estimated from Traffic Counts. Transportation Research B 14, 281-293

Webster, F. V. (1958). Traffic Signal Settings. Road Research Technical Paper No. 39. Department of Scientific and Industrial Research, Road Research Laboratory, London, UK

Wang, Y. (2008). Optimierung der Quelle-Ziel-Matrixschätzung hinsichtlich Redundanzstörung sich verändernder Verkehrszustände (Optimizating Origin-Destination Matrix Estimation with regard to Changing Traffic States) Dissertation at the Institute of Transport, Road Engineering and Planning, Leibniz Universität Hannover. Hannover, Germany.