2014 IEEE Intelligent Vehicles Symposium (IV)
June 8-11, 2014. Dearborn, Michigan, USA

Discrete Dynamic Optimization in Automated Driving Systems to Improve Energy Efficiency in Cooperative Networks*

P. Themann, R. Krajewski and L. Eckstein

Abstract—**Predictive and energy efficient driving styles considerably reduce fuel consumption and emissions of vehicles. Vehicle-to-vehicle and vehicle-to-infrastructure (V2X) communication provide information useful to further optimize fuel economy especially in urban conditions. This work summarizes an optimization approach integrating V2X information in the optimization of longitudinal dynamics. Besides the dimensions distance and velocity also the dimension time is reflected in discrete dynamic programming, which is based on a three-dimensional state space. Upcoming signal states of traffic signals are reflected in the optimization to implement an efficient pass through at intersections. Furthermore, simulated average driving behavior defines a reference for optimized velocity trajectories. This excludes optimization results strongly deviating from average behavior. The approach is implemented in a vehicle in a real-time capable way. In a field test the vehicle approaches a V2X traffic light and the optimization reduces fuel consumption by up to 15 % without increasing travel time.**

I. Introduction and Motivation

The escalating shortage of fossil fuels accompanied by steadily increasing energy costs strongly emphasizes the need to optimize energy efficiency of traffic. Driver assistance systems support drivers in this aspect by automating driving in an efficient way. As vehicle-to-vehicle and vehicle-to-infrastructure (V2X) communication gain market share the availability of cooperative information increases. The information exchange enables systems to also account future traffic light states in the optimization of energy efficiency. Driver acceptance of the optimization results is crucial.

II. State of the Art systems optimizing efficiency

Different system approaches to increase efficiency are described in literature. For the scope of this work the approaches are split into two groups. The first group of assistance systems decreases fuel consumption based on mostly static road information. The second group focuses on situations in which a vehicle approaches a traffic light and signal change information is available by V2I communication.

A. *Vehicle systems mainly based on static data*

Model Predictive Control is used in [1] to optimize fuel consumption while maintaining a reasonable distance to a preceding vehicle and assessing the state of traffic signals.

Another approach is dynamic programming. Porsche developed in 2011 a system called "Porsche InnoDrive" [2]. Static map data of the road ahead are used to optimize velocity trajectories calculated from a two dimensional discretized horizon in the vehicle. The optimization itself decreases accelerations, preserves speed in curvatures and makes use of coasting and cut-off to maximize the effect on fuel consumption. The system saves up to 10% of fuel on rural roads, while travel times do not change. A similar system described in [3] additionally implements an adjustable weight function enabling drivers to weight up fuel consumption versus travel time relative to a simulated average driver. Thus, resulting trajectories not only save fuel, but also closely resemble the average behavior of drivers. Simulations identify maximum fuel reductions by 12% and decreases in travel time by 2.5%.

An approach to reduce fuel consumption while simultaneously reaching a defined position after a fixed travel time is presented in [4]. The discretization is extended to the third dimension time. Due to high computational costs of this approach, optimizations are introduced in [5]. The main idea is to look at only two instead of three dimensions. The desired travel time is approximated by iterative adjustments to the cost function. Comparing to the not improved version the computational time is cut from "more than half a day" to 56 seconds. In contrast to that, another method to decrease computation time is described in [6]. Instead of executing the full trajectory optimization online, computational demanding parts of it are done offline. Rules are deduced from the computations used online as an approximation to the accurate calculations. The results presented show a neglectable difference between both variants.

B. *Dynamic information from traffic lights*

Traffic light information available from V2I communication is used in different ways to minimize idle time at traffic lights. One possibility to optimize traffic light approaches is the use of acceleration and deceleration profiles, which are adjusted to each individual situation minimizing fuel consumption or travel-time. Examples are given in [7], [8] and [9]. Likewise in [10] the optimal speed adjustment to arrive during green phases is calculated by optimizations minimizing the acceleration. In [11] a genetic algorithm determines the ideal constant speed between two traffic lights to minimize either travel-time or fuel consumption. Model Predictive Control optimizes traffic light approaches in real-time [12] [13], which allows including safe headway distances as boundary conditions. Another system includes traffic lights into a graph representation solved by shortest-path algorithms [14]. In many situations this results in pulse-and-glide maneuvers, in which the vehicle constantly accelerates and decelerates. This deviates strongly from average driving behavior and may affect user acceptance negatively.

*Research is funded by Deutsche Forschungsgemeinschaft (DFG) in the post graduate program "Integrated Energy Supply Modules for Roadbound E-Mobility" (mobileEM) www.mobilem.rwth-aachen.de

P. Themann is with the driver assistance department of the Institut für Kraftfahrzeuge (ika), RWTH Aachen University, (phone: +49-241-80-25673; fax: +49 241-80-22147; e-mail: themann@ika.rwth-aachen.de).

R. Krajewski is with the Institut für Kraftfahrzeuge (ika), RWTH Aachen University, (e-mail: krajewski@ika.rwth-aachen.de).

L. Eckstein is the head of the Institut für Kraftfahrzeuge (ika), RWTH Aachen University, (e-mail: eckstein@ika.rwth-aachen.de).

III. Research Approach and Methodology

The analysis of the state of the art shows that several approaches aim to improve fuel economy by automating the longitudinal dynamics of vehicles and that V2X communication provides useful information to vehicles. The approach presented in this work includes cooperative information in the optimization to fully exploit the potential of automated systems. To take into account both static (curvature, slope, speed limit, etc.) and time-dependent environmental information (traffic light status and timing, etc.) available to the vehicle, the optimization algorithm not only has to handle information about position and velocity, but also about time. This enables the algorithm to properly consider traffic light signal state information available from V2X communication. Additionally, the optimization algorithm presented here evaluates the average behavior of drivers in each traffic situation as a reference, as proposed in [15]. All systems described in the state of the art do not consider average driver behavior.

Discrete dynamic programming with the dimensions velocity, distance and time is applied extending state of the art approaches as V2X information and detailed driver models are included. A utility function based upon the driving behavior of a reference driver (see [15]) is used in the optimization. The utility function defines the usefulness of an optimized velocity profile and decreases it in case of strong deviations from the simulated reference driver. A standard Adaptive Cruise Control is extended by a controller using the resulting trajectory as input.

A. *Optimization approach*

To optimize velocity trajectories all given information about the driving environment ahead have to be accessible during the whole process of optimization. Thus, an adequate structure for the representation of this information is needed which includes all horizon information of a certain distance ahead and allows executing algorithms in a very time-efficient manner. Hence, a graph is used comprising a set of vertices (V) and edges (E) between these vertices. Each vertex represents a unique state of the vehicle on the horizon, while each edge represents the set of possible changes between two states. This approach results in a graph with a starting vertex and at least one target vertex. By assigning weights to each edge, the problem of optimizing a velocity trajectory is reduced to a shortest-path problem.

B. *Reference vehicle and driver model*

A Volkswagen Passat CC vehicle with a power of 224 kW and a semi-automatic 6-speed direct-shift gearbox is used within the scope of this research. The longitudinal dynamics as well as corresponding fuel consumption of the vehicle are modeled in Matlab/Simulink and the model is validated amongst others for the NEDC cycle. Interfaces to the CAN-Bus and several controllers allow controlling the longitudinal and lateral dynamics of the vehicle.

The model is executed offline to pre-process fuel consumption, driving time and target velocities for all driving strategies considered in the optimization. Results are stored in a multidimensional lookup table for each start velocity. This avoids complex and time consuming calculations of the vehicle model during the optimization.

The velocity of the vehicle is discretized by $\Delta v = 1$ km/h, which is also chosen in [3], for a maximum speed of 150 km/h. Gradients are discretized with $\Delta g = 1\%$ in a range of +/- 12%. To depict different driving strategies accelerations intervals of 0.1 m/s^2 are used within a range from -4 to +4 m/s^2 (40 variants for acceleration and 40 variants for deceleration). Additionally, the driving modes coasting in neutral, coasting in gear, fuel cut-off and also constant driving are reflected in the lookup table.

C. *Reference driver behavior in prediction module*

In order to assess the velocity of the vehicle for an upcoming situation, a prediction module is used. For the scope of this work the so-called "ecoSituational Model" (eSiM) prediction model is utilized which is developed in the research project eCoMove [16]. The eSiM model provides a short term prediction in form of a velocity profile versus distance and is based upon a detailed driver model. Average driver behavior is identified from data available from field operational tests (FOT) [17]. The parameters of the driver model are adapted to reflect the average behavior in an optimal way, which is not the scope of this work. The velocity profile available from the prediction module is used as a reference for the optimization as the utility function assesses optimized trajectories relative to this reference profile.

IV. Design of Optimization Algorithm

The graph representation of the horizon contains vertices representing unique states of the vehicle. These describe the position of the vehicle along the horizon, its current speed and also the specific moment in time. The corresponding gear is automatically chosen by the vehicle due to the automatic gearbox and hence implicitly included in the speed.

Edges between vertices are needed as transitions between valid states. Each edge also carries different values, but in contrast to vertices, edges are not unique in general. One value is the driving mode as described above. This information is needed as between vertices there are several edges with different driving modes. Furthermore, the edges contain the amount of fuel and time corresponding to the particular transition. A utility function assesses these values to determine the utility or weight of the edge. Based upon the utility of an edge the most suitable edges are chosen by the optimization algorithm. Each utility u results from weighting up the fuel consumption f and time t of the optimized trajectory (f_{eco}, t_{eco}) relative to the average trajectory (f_{avg}, t_{avg}), as described in (1). The parameter $\mathrm{w_E} \in [0,1]$ reflects the preference of the driver and weights between minimizing travel time ($\mathrm{w_E} = 0$) or minimizing fuel consumption ($\mathrm{w_E} = 1$).

$$\mathrm{u} = \mathrm{k_1} - \left(\mathrm{k_2}\, \mathrm{w_E} \frac{f_{eco}}{f_{avg}} + (1 - \mathrm{w_E}) \frac{t_{eco}}{t_{avg}} \right) \qquad (1)$$

The two constant factors $\mathrm{k_1}$ and $\mathrm{k_2}$ in (1) are determined from offline analysis in order to normalize the bandwidth of the decision alternatives [3], which is out of scope of this work.

The graph expands from the start vertex, which is the current vehicle state (position=0, speed = current speed, time=0). All driving modes are evaluated to create the edges starting from this first vertex, which leads to a set of reachable vertices. The repetition of this process results in the total graph.

The final position and the final velocity of the simulated average driver define the target position of the graph in the position-speed space. Thus, the reference trajectory is comparable to optimization results with respect to the distance traveled and the final velocity. The value for the arrival time is intentionally left open as it is affected by the optimization. Consequently, there are many target vertices defining suitable solutions. In order to find the optimal trajectories paths with minimal summed edge costs between the start and each target vertex have to be identified. This is the so called shortest-path problem. Because of its efficiency Dijkstra's algorithm [18] providing a loglinear time complexity is used, which allows optimizing graphs with a large number of vertices. The algorithm determines optimal trajectories to each target vertex and the best trajectory is chosen afterwards by comparing the utility values of each trajectory.

A. *Exclusion of variants in the graph setup*

Certain vertices and edges are excluded and not considered in the setup of the graph in order to avoid driving modes of the optimized trajectory that strongly deviate from the simulated average driver behavior. A range of acceptable speed and time values is set up relative to the reference trajectory forming a tube in the velocity-distance-time state space.

The diameter of the tube varies as a function of the reference speed at each position and the weighting factor w_E [15]. Higher factors result in higher accepted deviations from the reference speed. Additionally, speed limits and maximal velocities in curves influence upper bounds. The resulting tube of acceptable states so far affects only vertices and includes all boundaries given by the reference trajectory and static environmental properties.

In a second step time-dependent information is assessed to exclude certain states. The prediction module (see section III C) provides the predicted velocity profile (distance, velocity and time) of each vehicle that is detected by the radar sensor of the vehicle or that is known from V2X communication. This velocity profile of the front vehicle defines states in the graph that may not be chosen by the optimizer. An appropriate distance safety gap is added to the predicted velocity profile and all corresponding states are excluded from the tube. This way the front vehicle's predicted velocity profile cuts out a certain area in the optimization state space.

Additionally, V2X information of an arbitrary number of traffic lights is respected in the tube. Traffic lights allow passing during green phases and waiting during red phases which results in acceptable and excluded areas in the state space. To integrate this, vertices at velocities greater than zero are only allowed during green phases while on the other hand vertices with speed zero are only permitted during a red phase. Vertices at speed zero may only have an edge to the next waiting vertex at the next discrete time as long as the traffic light is red. To further shrink down the number of accepted driving modes, the reference trajectory is assessed and driving maneuvers that are unexpected and thus inconvenient for a driver are excluded.

Finally, the graph is set up starting with the first vertex. All permitted edges within the tube are evaluated. Only if the target vertex on the next position lies within the tube, the edge is added to the graph. If the target vertex does not already exist, it is added to the graph and also enqueued to a buffer of newly created vertices. After examining all possible edges of a vertex, the next one is dequeued from the buffer. This procedure is continued until the buffer is empty. As the order in which vertices and edges are added to the graph does not influence the result, the performance of this process is easily improved by examining multiple vertices in parallel.

B. *Implementation approach*

The optimization is implemented in C++ due to the higher performance in comparison to interpreted languages. Moreover, the library Boost [19] was used. Boost and especially its sub-library Boost Graph Library (BGL) provide several implementations of standard algorithms and efficient structures for graphs. In order to increase the performance during the graph setup, this process is run in parallel on different threads working on shared data. For this purpose, OpenMP [20] was chosen because of the easy implementation.

The optimization is executed on a quad-core i7 notebook with a 2.9 GHz processor. Results of the optimization are sent via UDP to a microcontroller which connects to the vehicle's interfaces via the CAN-bus. The microcontroller hosts a standard adaptive cruise control (ACC) that evaluates the distance to front vehicles. Additionally, a PID controller derives a required acceleration from the optimized velocity profile. To ensure safe distances to front vehicles, the minimal acceleration of both controllers is forwarded to the vehicle. This way the standard ACC algorithm intervenes only in situations where the optimized trajectory would result in low distances to front vehicles. The optimization requires a calculation time of roughly one second while the time delay of the standard ACC algorithm is smaller and hence it is able to react faster to changes in the behavior of front vehicles.

C. *Definition of appropriate discretization steps*

The application of discrete dynamic programming requires defining suitable discretization step size for each of the three dimensions time, distance and velocity. On the one hand, this affects the size of the lookup table for the vehicle model as well as the size of the graph. Thus, a smaller discretization step size results in higher memory usage and calculation time. On the other hand, with increasing discretization step size also the discretization errors increase. Previous research identifies a distance discretization step of $\Delta d = 50$ m to be useful in combination with a velocity discretization step size of $\Delta v = 1$ km/h [3]. This combination on the one hand provides sufficient discretization accuracies and on the other hand results in acceptable calculation times. Traffic lights have ideally to be positioned exactly on a discretized position as the system otherwise could make the vehicle stop in the middle of an intersection in the worst case. Consequently, the position of traffic lights is chosen as a starting point for the distance discretization which avoids discretization errors at the position of the traffic light.

State of the art systems do not consider the third dimension time in the discretization for time-dependent information (see section II.), and hence, the impact of the time discretization step size needs to be investigated. This especially affects all dynamic properties of the horizon such as preceding vehicles and traffic lights.

The discretization error in time sums up over distance and this could result in a green phase within the discretized time scale while the traffic light is still red in the continuous time scale without discretization errors. Similarly, the safety distance gap to a front vehicle is affected by the discretization error. To avoid these negative aspects, a suitable discretization step size for the dimension time needs to be defined and discretization errors at the traffic light position are monitored.

To define a suitable time discretization step size, the discretization error is assessed for a typical situation where a traffic light is placed at 900 m on the digital horizon and the start position of the vehicle is varied from 0 to 100 m in 20 m steps. The vehicle starts with 45 km/h and the traffic light is shifting from red to green in 70, 75 or 80 seconds. So there are 18 simulations that are repeated for every time discretization step. The outcome of the analysis is visualized in figure 1. The resulting discretization time error at the distance of the traffic light is assessed as well as the calculation time for the optimization. The evaluation shows that with increasing time discretization step size the calculation time decreases, but the cumulated discretization time errors at the traffic light increase. As the system needs to avoid red light violations, time errors at the traffic light position need to be controlled. For a time discretization step size of $\Delta t = 0.25$ s the median time error comes up to 0.43 s and the median total calculation time results in 0.81 s. Thus, in the following this step size is chosen. The optimization checks the time error of each trajectory at the point of the traffic signal to prevent red light violations.

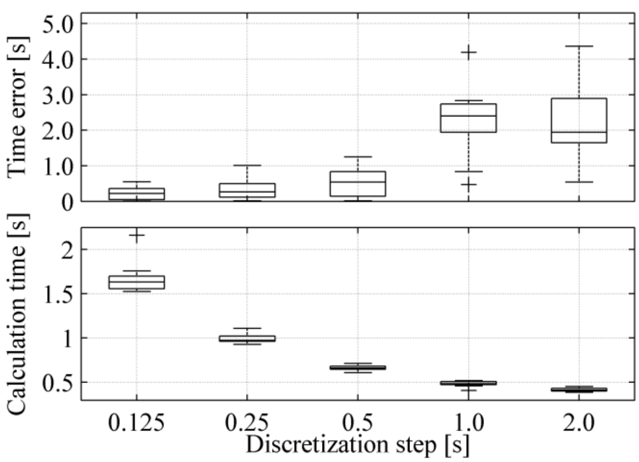

Figure 1. Cumulated discretization time errors and calculation times

Time discretization affects the size of the graph, which causes different calculation demands. In the example above there are 1690645 edges and 153859 vertices for a time discretization step of $\Delta t = 0.125$ s which results in a calculation time of 1528 ms for the optimization. The prediction module requires another 104 ms in this situation and this results in a total computation time of 1632 ms. The size of the graph shrinks down to 861163 edges and 78427 vertices for a time discretization step of $\Delta t = 0.25$ s which results in a calculation time for the optimization of 882 ms and consequently in a total calculation time of 986 ms. The prediction module therefore is responsible for roughly 10% of the total calculation time for the chosen time discretization step.

Besides discretization also the dimensioning of the considered time range is important. While a short time range could exclude the optimal result from the state space, a long range potentially includes neglectable arrival times and causes also higher computation time. To overcome these problems, the travel time of the reference trajectory is evaluated to define the maximal accepted arrival time.

V. Validation of Energy Efficiency

A. *Simulated approach of cooperative intersection*

One main advantage of the optimization approach introduced above is the consideration of traffic signal phase information in the optimization. As the states of the traffic signal change over time, the optimization needs to be validated for different scenarios. In case the traffic light is green and stays green for a sufficient amount of time, the vehicle is able to pass the intersection without deceleration. The optimizer in this scenario does not provide any reduction in fuel consumption. If the traffic light is red while the vehicle approaches, the optimizer is able to identify energy efficient deceleration strategies and hence reduces fuel consumption. The vehicle and driver models described above allow determining fuel and time demand for average and also for optimized approaches. External factors such as the signal switch time or the velocity of the vehicle are fed into the simulation.

To analyze the effect of the optimization, a total simulation distance of 1.5 km is conservatively chosen. The distance from the vehicle to the traffic light is varied from 0 m to 600 m in steps of 50 m while it is assumed that V2X communication provides connectivity for up to 600 m.

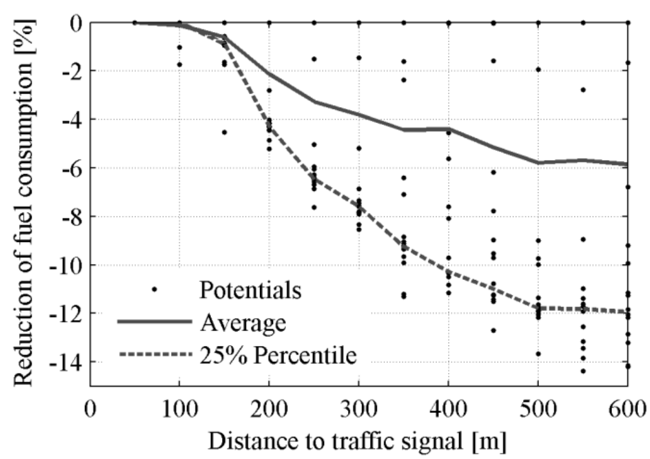

Figure 2. Reduction of fuel consumption for a weighting factor $w_E = 1$

For this simulation a flat road without inclination is assumed. The traffic signal is modeled to have equal switching times for the green and the red phase of 60 s each. This assumption heavily affects the number of situations for which the optimization is actually able to find more efficient deceleration strategies. In green phases there are no decelerations and the optimizer cannot improve the driving style. The assumption of an equal distribution of red and green phases thus underestimates the impact of the optimization for traffic signals with considerably longer red phases. The start time of the traffic light phase is changed from 0 to 120 s in steps of 5 s to model different situations. Moreover, the driver preference factor is chosen to be 0, 0.5 or 1 to evaluate different optimizer parameterizations. All in all this sums up to 864 traffic situations. For each situation the driver model provides a reference velocity trajectory. The optimization algorithm using the vehicle model calculates fuel consumption and travel time for the reference and the optimized approaching.

TABLE I. POTENTIAL CHANGE IN FUEL CONSUMPTION RELATIVE TO REFERENCE DRIVER FROM FOT DATA

Distance	Average	25% Percentile	Maximum
100 m	-0,1	0,0	-1,7
200 m	-2,1	-4,3	-5,2
300 m	-3,8	-7,6	-8,5
400 m	-4,4	-10,3	-11,2
500 m	-5,8	-11,8	-13,7
600 m	-5,9	-11,9	-14,2

Figure 2 visualizes the resulting reduction of fuel consumption of the optimized velocity profiles in comparison to the reference trajectory. The reductions are plotted versus the distance from the vehicle to the traffic light. The evaluation reviles an average reduction in fuel consumption of 5.8% for a distance of 600 m to the traffic light. For this distance the maximal reduction of fuel consumption comes up to 14% and the 25th percentile over all traffic signal start times at this distance results in a reduction of fuel consumption of 12%. Details are summarized in Table I. The analysis shows a decrease of the reduction potential with decreasing distance to the traffic light. At distances to the traffic light below 150 m the impact of the optimization is marginal and there are only few situations where actual reductions in fuel consumption are achieved. The distance discretization within the setup of the optimization algorithm is set to 50 m, which explains the limited reduction of fuel consumption close to the traffic light. As there are only few vertices before the traffic light, the optimizer in many situations does not find a suitable solution and so the reduction potential diminishes.

B. *Test drive approaching cooperative intersection*

The optimization approach is implemented in the test vehicle Passat CC and real world test drives are performed in the cooperative test site in the city of Aachen. Four intersections in the city are equipped with routers sending out traffic signal phase information to vehicles. In-vehicle routers receive and decode the messages that are used in the optimization algorithm. As the test site is a part of the public traffic network, the optimization algorithm and the reaction of other traffic participants to the system are evaluated in real world conditions. Compared to dedicated tests on a controlled and closed test site this allows to evaluate external influences.

The traffic signal phases of the different intersections are synchronized to each other by the city traffic management. To ensure comparability, all test drives are started at intersection number 1 when the traffic signal turns green. The vehicle approaches intersection number 2 in a distance of 730 m, with identical signal phases (current state and shifting time to next phase) in all test drives. To compare vehicle states with comparable velocities, a distance of 50 m behind the traffic light is included in the evaluation.

The field operational test (FOT) executed in the euroFOT project provides valuable data to describe naturalistic driving behavior [17]. FOT data of more than 90 drivers on more than 790,000 km is evaluated to tune the prediction module (see section III.C) in a way that it reflects average driving behavior. In a test drive the system implements this average behavior and represents the reference the optimization is compared to. Figure 3 visualizes the velocity profile of the system configured to the average behavior (marked as “Reference”) as well as the average and standard deviation extracted from the FOT data in equal traffic situations. This comparison proves that the reference configuration results in a velocity trajectory which stays within the standard deviation of the FOT data. The minor deviations of the velocity profile from 50 km/h in the constant drive phase at -450 m and -200 m stem back from shifts of the automated gear box and slope changes.

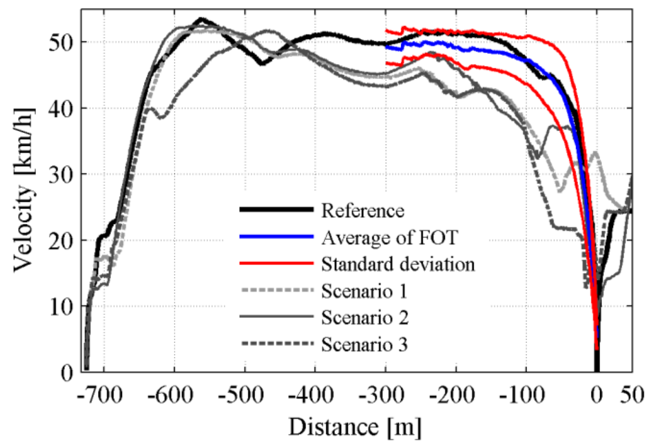

Figure 3. Comparison of different real-world scenarios for $w_E = 0.5$

In this scenario the V2X communication provides connectivity at about 280 m in front of the traffic light. Up to this point per default the system assumes the traffic signal to be red, and hence for this scenario communication range does not affect the results.

TABLE II. COMPARISON OF DIFFERENT REAL-WORLD SCENARIOS

	Reference	Scenario 1	Scenario 2	Scenario 3
Fuel consumption	105.89 ml	98.15 ml (-7.3 %)	98.44 ml (-7.0 %)	98.32 ml (-7.2 %)
Travel time	79.73 s	78.18 s (-1.9 %)	83.59 s (+4.8%)	84.88 s (+6.5 %)

The optimization approach is evaluated in three scenarios with different impacts of front vehicles. As the tests are performed on public roads, the front vehicles are uninfluenced. In scenario 1 one front vehicle stopped at the traffic signal and accelerates quickly which results in only very limited intervention of the standard ACC algorithm. In the second scenario the front vehicle accelerates not as quickly as before, and hence the standard ACC has to intervene for a longer amount of time. In the third scenario there are three vehicles waiting in front of the traffic light and accelerate slowly after it turns green. None of the front vehicles is equipped with V2V communication and hence is not considered in the optimization. The Radar sensor of the vehicle detects the front vehicles and the standard ACC algorithm reacts on this. Resulting travel time and fuel consumption for all scenarios are summarized in Table II. From scenario 1 to 3 reductions in fuel consumption decrease as the standard ACC algorithm intervenes more often. However, also in scenario 3 fuel consumption is decreased while travel time only increases slightly compared to the reference drive. The system thus proves to be robust with respect to the number of front vehicles. Furthermore, it implements efficient driving strategies such as fuel cut-off or gear shift to neutral and accounts the slope of the road which is visualized in figure 4 for scenario 1.

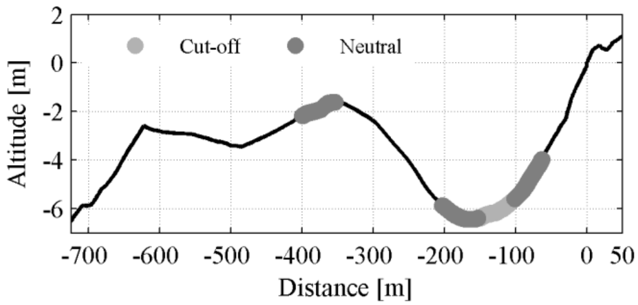

Figure 4. Altitude profile and efficient driving strategies applied

The system optimizes especially the approach and deceleration phase and does not strongly affect the acceleration phase. Evaluating a distance from 500 m in front to 50 m behind the signal hence focuses on the most important part of the situation. For this part fuel consumption is reduced by 15.1 % while also time travel time decreases by 7.3 % in scenario 1 compared to the reference. Also for scenario 3 with three front vehicles fuel consumption reduces by 10.6 % while travel time increases only by 3.8 %. This underlines that the system is useful especially in approach and deceleration situations. Front vehicle equipped with V2X communication could be included in the optimization and the system would further reduce fuel consumption.

C. *Driver Acceptance*

Driver acceptance of the assistance system approaching cooperative traffic lights is assessed with questionnaires. The system was presented to 13 international subjects. Due to legal issues the subjects were not allowed to drive the test vehicle themselves, but could experience the system activated as a passenger. In average the usefulness according to van der Laan [21] was rated with 1.02 (standard deviation of 0.4) and the satisfaction to 0.67 (standard deviation of 0.37). This reflects a generally positive attitude towards the system.

VI. Conclusion and Outlook

This paper summarizes an approach to directly consider dynamic time-dependent information from traffic signals and front vehicles in the optimization of the vehicle's longitudinal dynamics which extends state of the art systems. A method to define suitable step sizes for time discretization is described and parameters are derived for the depicted implementation. Simulations assess the performance of the system, which takes less than 1 second for the optimization of 1.5 km ahead. The chosen parameterization of the system reduces fuel consumption in average for different situations by 6 % in case the signal information is available 500 m in front of the signal. Further research aims to parameterize the system to also effectively reduce fuel consumption at smaller V2X communication ranges. In this situation e.g. the distance discretization could be reduced while a shorter optimization horizon of less than 1.5 km would be sufficient. The authors prove the usability of the system in a real world driving test where fuel consumption decreases by around 15 % in deceleration situations approaching cooperative traffic signals. This verifies the approach to be valid and the implementation concept to be real-time capable. Additional research aims to further reduce the calculation demand of the algorithm. Also the effect on multiple vehicles needs to be examined. In first tests the system is very well accepted by subjects, but driver acceptance needs to be examined in dedicated field operational tests that also need to analyze limited communication ranges.

References

[1] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, "Development of ecological driving system using model predictive control," in *ICROS-SICE*, Japan, 2009.

[2] M. Roth, T. Radke, and M. Lederer, "Porsche InnoDrive - An Innovative Approach for the Future of Driving," in *20th Aachen Colloquium Automobile and Enginge Technology*, Aachen, 2011.

[3] P. Themann, J. Bock, and L. Eckstein, "Energy efficient adaptive cruise control utilizing V2X information," in *9th ITS European Congress*, Dublin, 2013.

[4] F. Mensing, R. Trigui, and E. Bideaux, "Vehicle trajectory optimization for application in ECO-driving," in *2011 IEEE Vehicle Power and Propulsion Conference*, Chicago, 2011, pp. 1–6.

[5] F. Mensing, E. Bideaux, R. Trigui, and B. Jeanneret, "Trajectory optimisation for eco–driving - an experimentally verified optimisation method," *International Journal of Vehicle Systems Modelling and Testing*, vol. 8, no. 4, pp. 295–315, 2013.

[6] F. Aparicio, J. L. López-Covarrubias, W. Cabrera, and F. Jiménez, "Real-time speed profile calculation for fuel saving considering unforeseen situations and travel time," *IET Intelligent Transport Systems*, vol. 7, no. 1, pp. 10–19, 2013.

[7] M. Barth, S. Mandava, K. Boriboonsomsin, and H. Xia, "Dynamic ECO-driving for arterial corridors," in *2011 IEEE Forum on Integrated and Sustainable Transportation Systems*, 2011, pp. 182–188.

[8] H. Rakha and R. K. Kamalanathsharma, "Eco-driving at signalized intersections using V2I communication," in *14th IEEE Conference on Intelligent Transportation Systems*, Washington, 2011, pp. 341–346.

[9] P. Schuricht, O. Michler, and B. Baker, "Efficiency-increasing driver assistance at signalized intersections using predictive traffic state estimation," in *2011 14th International IEEE Conference on Intelligent Transportation Systems*, Washington, 2011.

[10] S. Mandava, K. Boriboonsomsin, and M. Barth, "Arterial velocity planning based on traffic signal information under light traffic conditions," in *12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, 2009.

[11] M. Seredynski, W. Mazurczyk, and D. Khadraoui, "Multi-segment Green Light Optimal Speed Advisory," in *27th IEEE International Parallel and Distributed Processing Symposium*, 2013.

[12] B. Asadi and A. Vahidi, "Predictive Cruise Control: Utilizing Upcoming Traffic Signal Information for Improving Fuel Economy and Reducing Trip Time," *IEEE Trans. Contr. Syst. Technol.,* vol. 19, no. 3, pp. 707–714, 2011.

[13] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, Eds, "On board eco-driving system for varying road-traffic environments using model predictive control," in *2010 IEEE International Conference on Control Applications*, Yokohama, 2010.

[14] S. Gausemeier, A. Trächtler, and K.-P. Jäker, "Fahrerassistenzsystem für energie- und zeitoptimales Fahren durch prädiktive Geschwindigkeitsprofil-Planung," in *13. Braunschweiger Symposium AAET*, 2012.

[15] P. Themann and L. Eckstein, "Modular approach to energy efficient driver assistance incorporating driver acceptance," in *2012 IEEE Intelligent Vehicles Symposium (IV)*, Madrid, 2012, pp. 1023–1028.

[16] N.N, *eCoMove project.* Available: http://www.ecomove-project.eu/.

[17] N.N, *euroFOT – Bringing intelligent vehicles to the road.* Available: http://www.eurofot-ip.eu.

[18] E. W. Dijkstra, "A note on two problems in connexion with graphs," in *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[19] N.N, *Boost C++ Libraries.* Available: http://www.boost.org/.

[20] N.N, *OpenMP.* Available: http://openmp.org/.

[21] J. D. Van Der Laan, A. Heino, and D. de Waard, "A simple procedure for the assessment of acceptance of advanced transport telematics," *Transportation Research Part C: Emerging Technologies*, vol. 5, no. 1, pp. 1–10, 1997.