Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Highway Entrance Merging Assistant for Minimal Traffic Disturbance

Amin Assadi * **Florian Meier** * **Luigi del Re** *

* *Institute for Design and Control of Mechatronical Systems, Johannes Kepler University Linz, Altenberger Strasse 69, A-4040 Linz, Austria (e-mail: {amin.assadi, florian.meier, luigi.delre}@jku.at)*

Abstract: Merging of traffic flows is a potentially dangerous situation frequently associated with important impacts on traffic fluidity. This paper examines the possibility of improving merging from an entrance of a highway by adapting the speed of the incoming vehicles reducing as much as possible the need for actions by the drivers on the main lane. Multi-layer traffic models are used to describe and simulate the interactions between drivers. A predictive controller is then designed to minimize the probability that the vehicles on the incoming roads need to change the speed or the lane. The proposed approach is shown to strongly enhance traffic fluidity for a wide range of traffic densities on the main flow without compromising safety.

Keywords: Traffic merging, traffic modeling, vehicle control

1. INTRODUCTION

Traffic merging in general (e.g. highway entrances) is one of the most critical maneuvers potentially leading to dangerous situations which demand actions on all traffic participants, in particular the correct adaptation of the speed of the vehicles in the main line, or even a lane change, whenever possible. However, these actions tend to affect negatively the traffic fluidity (see e.g. by Laval and Daganzo [2006], Tang et al. [2007]).

Such scenarios and consequential chain reactions by following drivers, regularly lead to traffic disturbances and in further consequence to jams near merging zones as examined in several studies such as Nagel and Herrmann [1993], Nagatani [1995], Jerath and Brennan [2012].

All this could be avoided if the speed of the incoming vehicle was adapted to prevent the need for a reaction of the vehicles in the main lane, at least in the cases in which the traffic density allows it.Wei et al. [2013] considers algorithms to estimate the intention of other traffic participants and to correspondingly choose an adapted policy for the the entering vehicle. The authors demonstrate that cooperative behavior outperforms regular adaptive cruise controllers (ACC) in merging scenarios.

Most effort has been put into the development of synchronization methods for automated highway systems (AHS). Several publications about coordinating highway entrance maneuvers have been presented over the last decades e.g.Yang and Kurami [1993], Kachroo and Li [1997], Kato and Tsugawa [2001], Lu et al. [2004], Milanés et al. [2011], Rios-Torres and Malikopoulos [2017], Letter and Elefteriadou [2017].

The majority of all approaches relies on some kind of communication such as inter-vehicle (V2V) or vehicle to infrastructure (V2I) in order to synchronize maneuvers. A comprehensive overview of merging algorithms using intelligent communicating traffic can be found in Scarinci and Heydecker [2014]. Information is exchanged in order to calculate a merging trajectory, in a centralized or decentralized way, for the entering vehicle. This coordination and cooperation are beneficial in terms of increasing traffic capacity and energy efficiency, however, it requires all vehicles to be connected.

However, this ideal case, which would solve several traffic issues at a time, cannot be expected within the next years for all vehicles on the road. As a possible alternative this paper focuses on optimizing the trajectory of the entering vehicle so to disturb as little as possible the traffic on the main road. No exchange of information is needed and no assumptions about the equipment of other traffic participants have to be made, only relative speed and position measurements are needed as they can be obtained with Lidars now available for many vehicles.

In order to minimize the reactions of the drivers on the main road, we first need a model of these interaction, for which we propose a double-layer stochastic system. A classifier determines the lateral and longitudinal maneuver based on current states of the vehicle itself and the surrounding traffic. Bayesian networks are then used to model the specific maneuver in a stochastic framework Barber [2012].

2. PROBLEM STATEMENT

The considered scenario is depicted in Fig. 1, where an entering vehicle S wants to merge onto a highway in front of the vehicle P on the main lane. Our control tasks consists in determining the longitudinal velocity of S so that it merges along the merging lane without forcing P to change its speed or lane. Based on measurement data, conditions for longitudinal velocity difference and longitudinal distance were derived, under which decelerations or lane changes are likely to be triggered.

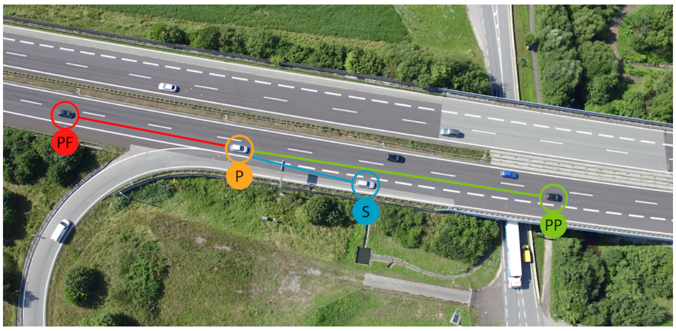

Fig. 1. Highway entrance - general merging scenario (P: Primary Veh. S: Secondary Veh. PP: Primary's Preceding Veh. PF: Primary's Following Veh.)

In this work, for simplicity we consider only the situation presented in Fig. 2, with one vehicle mering in the right lane of the freeway and no vehicle in the left lane, but the method is extendable to more complex cases as well.

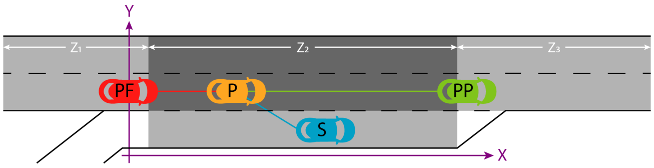

Fig. 2. Surrounding vehicles.

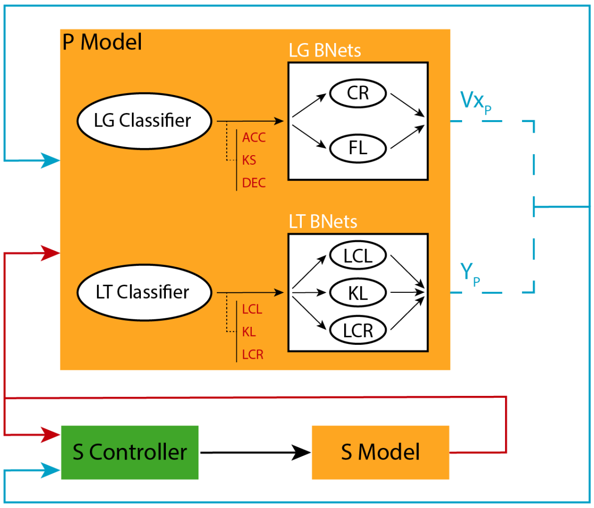

Fig. 3. Descriptive scheme of the system

3. METHODS

3.1 Methodology and criteria of assessment

The task to be solved requires three components

(1) a model of the behavior of the vehicles P on the main road when another vehicle S is entering the the lane from an entrance
(2) an inverse model, i.e. conditions under which the vehicles P will not change their path or speed
(3) a control approach able to steer the path of the vehicle S respecting these results

Models and simulation were performed using real traffic data from the highD set Krajewski et al. [2018]. The simulation setup as shown in Fig.3 was used to test the performance of the controller. This setup includes a block for the controller of the S vehcile, a block for its dynamics (a double integrator) and a complex model for the behavior of the Ps vehicles derived from the highD data. As initial conditions for all traffic participants, random scenarios were taken from the traffic data. At each time step, the controller calculates the complete optimal trajectory of the entering vehicle S and applies the first sample. To calculate an optimal trajectory, predictions of the future traffic situation are needed.

Testing was performed repeating the simulation S reaches a feasible and safe merging point and successfully performs a merging maneuver. The algorithm fails, if S reaches the end of the merging lane without having successfully performed a safe merge. The algorithm was assessed using a Monte Carlo Simulation with randomly chosen traffic out of the 147 hours of recorded real traffic data.

The performance of the controller in terms of safety and traffic impact was assessed by comparison to a baseline of an inattentive, inexperienced or reckless driver, who does not consider the impact of its actions on others. He just merges once reaching usually forcing others to react harshly in order to avoid hazardous situations or even accidents.

3.2 Traffic interaction model P

Typically, the movement of the vehicles can be categorized into five general categories:

- "Cruising" and "Following" for the longitudinal movements.
- "Lane Change Left", "Keep Lane" and "Lane Change Right" for the lateral movement.

For each of them, specific Bayesian Networks have been designed and trained using discretized trajectories with a suitable sampling time, which following specifications;

(1) **Cruising Bnet**: is a discrete model which calculates the next differential longitudinal velocity depending on the current one and the longitudinal velocity.
$$CR : Pr(\Delta Vx_{(k+1)} \,|\, \Delta Vx_{(k)},\, Vx_{(k)})$$
where $\Delta Vx_{(k)} = Vx_{(k)} - Vx_{(k-1)}$.
(2) **Following Bnet**: is a discrete model which calculates the next differential longitudinal velocity depending on the current one and the relative velocities between the vehicle and its preceding.
$$FL : Pr(\Delta Vx_{(k+1)} \,|\, \Delta Vx_{(k)},\, (Vx_{PP(k)} - Vx_{(k)}))$$
(3) **LaneChangeLeft Bnet**: is a discrete model which calculates the next lateral position depending on the current one.
$$LCL : Pr(y(k+1) \,|\, y(k))$$
(4) **KeepLane Bnet**: is a discrete model which calculates the next lateral position depending on the current one.
$$KL : Pr(y(k+1) \,|\, y(k))$$

(5) **LaneChangeRight Bnet**: is a discrete model which calculates the next lateral position depending on the current one.

$$LCR : Pr(y(k+1) \mid y(k))$$

Only two BNets (one longitudinal & one lateral) are used to calculate the next position of each vehicle at each time-step. The decision of selecting the correct BNets for each vehicle is taken by two classifiers, one for longitudinal & one for lateral movement. Since the average perception-reaction time for human drivers is in the order of 1 second(Gartner et al. [2001]) and the drivers tend to be more attentive near to a junctions, the execution time for each classifier has been set at 500 milliseconds.

In driving, each vehicle is influenced by the events which happen around it. Each driver decides its future movements based on the event's type and its importance level. For simplification, the decision of the drivers is classified into two general classes named as Longitudinal and Lateral. Each class has three labels.

- Accelerating (ACC), Keeping the Speed (KS) and Decelerating (DEC) are the labels for the longitudinal movements.
- Lance Change Left (LCL), Keep Lane (KL) and Lane change Right (LCR) for the lateral movements.

Although the maneuver's decisions of the P vehicle are influenced by the events in all its surrounding zones, for simplicity we shall consider here only events in two "Preceding" and "following" zones.

Useful information (like: the existence of surrounding vehicles in the stated zones, time headways, relative longitudinal distances) are extracted from the dataset. In addition, the relative conditions between P and S have been considered. The details of all the engaged parameters in this modeling are reported in Tables 1 and 2. These parameters were considered as inputs for the classifiers.

Table 1. Inputs of the Longitudinal classifier.

abbreviation	Description
Vx_P	Longitudinal Velocity of the Primary Vehicle
Vx_S	Longitudinal Velocity of the Secondary Vehicle
$PS_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Secondary Vehicle
$PP_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Preceding Vehicle
$PF_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Following Vehicle

Table 2. Inputs of the Lateral classifier.

abbreviation	Description
$PS_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Secondary Vehicle
$PP_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Preceding Vehicle
$PF_{Rel\,LG\,D}$	Relative Longitudinal Distance between Primary and Following Vehicle

In this research, various classification methods (like: Decision Trees, Naive Bayes, Nearest Neighbors, Support Vector Machines, Ensemble, etc.) have been studied to find the most suited method based on the studied-case and dataset. After several tests, the KNN classification method has shown the best results. The misclassification rate of the longitudinal classifier is less than 9%, mainly because it is particularly difficult and also senseless to distinguish between marginal accelerations/decelerations and speed keeping. Lane change is easier to distinguish from keeping the lane and the misclassification rate is about 1% (Table 3).

Table 3. Misclassification Rate.

	Dataset 1	Dataset 2	Dataset 3	Total
LT	0.9 %	0.63 %	0.17 %	0.58 %
LG	6.02 %	11.9 %	7.4 %	8.73 %

Although the misclassification rate of the longitudinal classifier is much higher than the lateral one, their impact ratio is the opposite. In other words, negative consequences caused by a wrong classification of the lateral classifier are much worse than the longitudinal one.

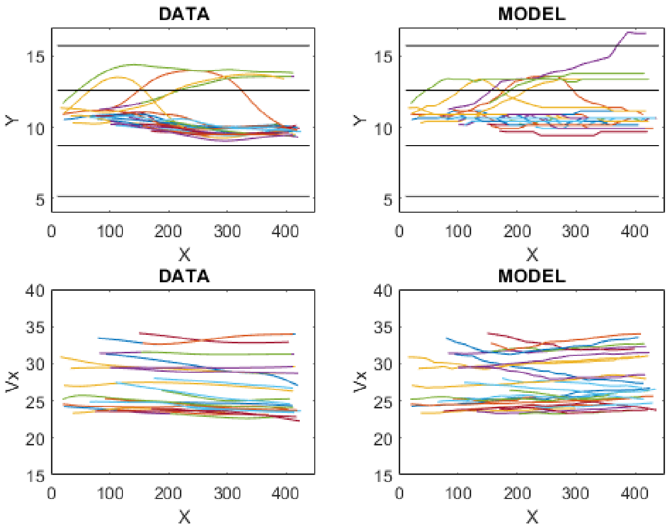

Fig. 4. Dataset 1 Vs Model Simulation.

This discrepancy in the importance level of the two classifiers is caused by the differences in the "Lane Changing" performing time with the other maneuvers. As stated before, the execution time of each classifier is 500 milliseconds. This means in every 500 milliseconds a new decision is taken and implemented (for Longitudinal movement: accelerating, decelerating, keeping speed. - for lateral movement: keeping lane, lane changing to left or right). However, if a lane-change decision is taken, all the next lateral decisions are not considered until the lane-change maneuver is completed. Accordingly, the errors resulting from a wrong decision by the lateral classifier are heavier than those caused by wrong decisions of the longitudinal classifier. (Fig. 4.)

In order to evaluate the model, the values of longitudinal velocities and Lateral positions in the dataset have been compared with the result of the simulation, and their differences have been calculated and reported in Fig. 5 & 6.

3.3 CONDITIONS FOR IMPACTING TRAFFIC

Through analysis of the classified data, as depicted in Fig. 7, four conditions were identified under which forced lane

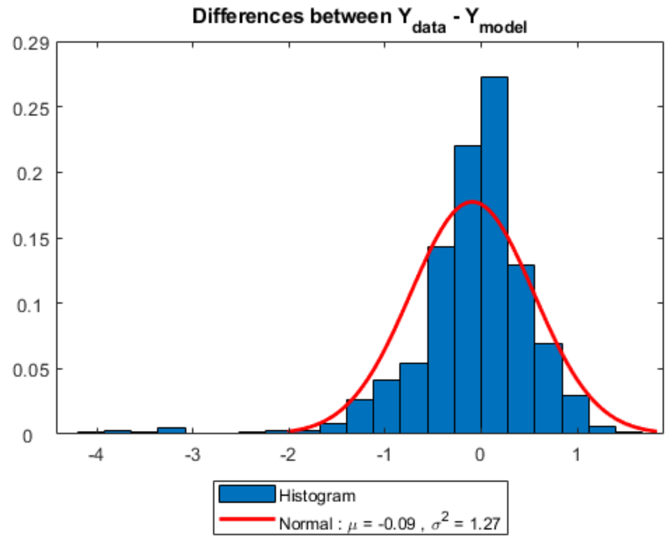

Fig. 5. Model evaluation - Lateral movement.

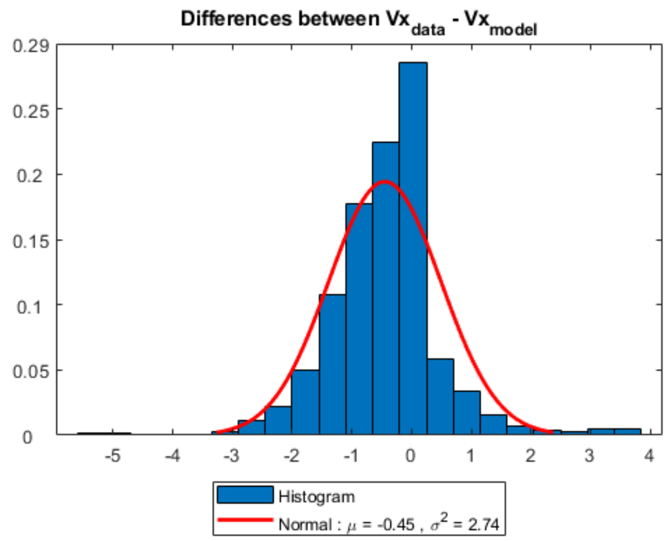

Fig. 6. Model evaluation - Longitudinal movement.

changes occur. For all triggered lane changes conditions (1)-(4) (depicted by the red border) were true. The conclusion is, that triggered lane changes most likely happen if the combined states of P and S are below the red box in Fig. 7. One constraint on the controller is to avoid those conditions whenever physically feasible in order to prevent situations that are likely to result in a lane change of any P_i. The conditions where lane changes were recorded are:

$$0 \leq \Delta x \leq 160 \quad (1)$$
$$v_s \leq v_p - 5 \quad (2)$$
$$\Delta x \leq -10 \cdot \Delta v \quad (3)$$
$$0 \leq x_p \leq 230 \quad (4)$$

The longitudinal distance $\Delta x = x_s - x_p$ between S and P is between 0 and 160m. P's speed v_p exceeds S' v_s by at least 5m/s. The time-to-collision (TTC $= -\Delta x/\Delta v$) is smaller than 10s and P is close to or beside the merging lane, which in our scenario goes from 60 to 230m.

3.4 DYNAMIC VEHICLE AND PREDICTION MODEL

In order to achieve fast computation times so that the algorithm could be implemented in reality, a simplified model was aimed at. Therefore the ego vehicle S is modeled as a double integrator with the acceleration as input. Since the acceleration limits (a_{lb} = -4m/s^2 a_{ub} = 2m/s^2) are derived from real measurement data with the test car it can be presumed, that the engine is able to deliver the torque demand and the vehicle follows the requested trajectory. S' velocity is restricted to the interval (v_{lb} = 12m/s and v_{ub} = 40m/s) in accordance with the data in the highD dataset.

To achieve a set of linear constraints simple prediction models for all the other traffic participants P are required. For a baseline the cruising mode is chosen, where all vehicles on the highway basically keep their speed for the next few seconds. To improve prediction precision initial accelerations are considered as well but assumed to converge towards 0 (which means constant speed) as in Eq.5. This is the most commonly recorded scenario for undisturbed traffic flows on highways. As the aim is to influence other traffic participants as little as possible with S' actions this can be presumed as a valid assumption in accordance with the measurements.

$$\hat{a}_{p,i}(k) = a_{p,i}(0) \cdot 0.5^k \; \forall i = 1..n \quad (5)$$

For longer prediction horizons, of course, this might lead to relevant prediction errors for unsteady traffic conditions. However, for short term predictions it proves of sufficient accuracy. Since the controller updates the optimal action at each time step based on the latest future predictions it adapts to deviations between the predicted and actual behavior of other traffic participants. As the results show, using even a trivial prediction model enables satisfying results for a wide range of traffic conditions when using the controller in a feedback loop.

3.5 CONTROLLER

The controller optimizes S' acceleration in order to guarantee a safe merging maneuver without impeding the traffic flow. The traffic flow is impeded if vehicles already on the highway are forced to brake or perform a lane change in order to maintain safety distances or even avoid collisions with the merging vehicle. S starts at the initial position $x_{s,0}$=0m with a random initial velocity $v_{s,0} \in$ [15,25]m/s^2 and is allowed to merge onto the main lane within the whole merging lane between 60 and 230m. If S' position exceeds 230m without having found a safe merging time, the scenario is considered to be failed. This might occur due to very dense traffic, where safe merging without cooperative behavior of the other traffic participants is just physically impossible. A safe merging maneuver is defined, so that the distance between S and all Ps is at least 20m and the TTC is bigger than 5s.

The objective of the controller besides safety is a comfortable merging so it minimizes quadratic jerk. Merging time is minimized as well in order to encourage accelerations to the nominal highway speed and avoid optimal solutions that schedule the merging at the very end of the merging lane, where small disturbances can easily lead to infeasibility of the control problem.

The controller calculates the optimal acceleration trajectory $a_{s,opt}$ for S at each timestep according to:

$$a_{s,opt} = \arg \min_{a_s, k_m} \sum_{k=0}^{k_m} (\Delta a_s)^2 + c \cdot k_m \quad (6)$$

$$s.t.$$

$$a_{lb} \leq a_s(k) \leq a_{ub} \quad (7)$$

$$v_{lb} \leq v_s(k) \leq v_{ub} \quad (8)$$

$$x_{m,0} \leq x_s(k_m) \leq x_{m,end} \quad (9)$$

$$TTC(k_m) \geq TTC_{min} \ \forall i = 1..n \quad (10)$$

$$x_s(k_m) \geq \hat{x}_{p,i}(k_m) + SD || x_s(k_m) \leq \hat{x}_{p,i}(k_m) - SD \quad (11)$$

$$\Delta\hat{x}_i(k) \leq d_{min} || \Delta\hat{x}_i(k) \geq d_{max} \text{ OR} \quad (12)$$

$$v_s(k) \geq \hat{v}_{p,i} + v_{diff} \text{ OR} \quad (13)$$

$$\Delta\hat{x}_i(k) \geq -10 \cdot \Delta\hat{v}_{p,i}(k) \text{ OR} \quad (14)$$

$$\hat{x}_{p,i} \leq x_{min} || \hat{x}_{p,i}(k)| \geq x_{max} \quad (15)$$

$$\forall i = 1..n$$

The acceleration and velocity constraints must be satisfied during the whole maneuver as well as at least one of the conditions to avoid negatively influencing the traffic by forcing any P_i to either decelerate or perform a lane change. This can be achieved with a high probability by avoiding that all conditions (1)-(4) become true at the same time for any P_i. d_{min} and d_{max} are the range of relative longitudinal distances between S and others for which lane changes were recorded. v_{diff} is the speed offset lane changing cars were at least faster than the entering one and x_{min} respectively x_{max} limit the area around the merging lane where lane changes occurred in the measurements.

The conditions of a safe merging, satisfying the safety distance SD and the time to collision limit TTC_{min}, must only be satisfied at the time of merging k_m. $\hat{x}_{p,i}$ and $\hat{v}_{p,i}$ are the predicted position and velocity of the ith vehicle, where i is between 1 and n the number of vehicles in the scenario.

This problem can be brought in to a mixed integer quadratic programming structure which is solved effectively by a solver such as Gurobi Optimization [2019].

4. RESULTS

The controller is evaluated using a Monte Carlo simulation (N=10000) with random traffic scenarios from highD. For comparison purpose, a 'blind' or very unexperienced driver is used, which is not aware of the consequences caused by his maneuvers. He just merges at a given point not considering any other vehicles forcing others to severe actions in order to avoid accidents. The controller proves to reduce the number of dangerous merging scenarios drastically. In fact the only critical maneuvers using the controller occur, when there is no physically possible safe merging solution, so the cooperation of others is required in order to let S merge. For example such a scenario occurs, if all P on the highway already fall below the defined safety gaps, which is a regular case during highspeed following.

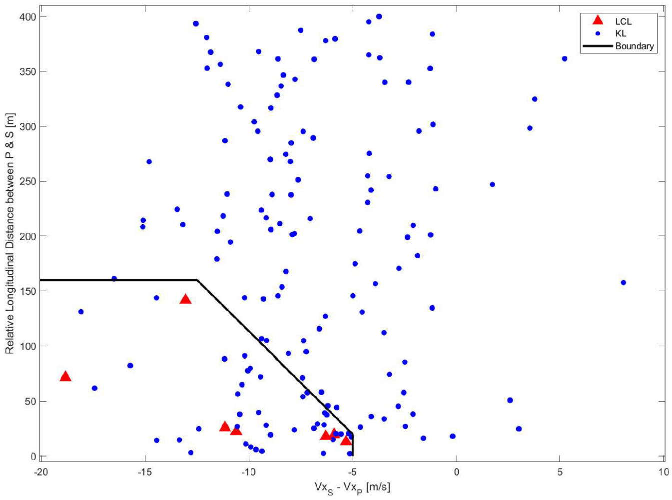

Fig. 7. Initial conditions for measured lateral movements (KL=blue, LCL=red)

Table 4. Results

	Controlled	Blind merging
Safety distance violations	0.3%	16.4%
min TTC violactions	6.6%	27.6%
triggered lane changes	1.2%	8.8%
mean deceleration <-0.5m/s	4.6%	20.4%
min deceleration <-1m/s	11.1%	28.5%

As the results in Tab.4 show, succeeds the controller in 99.7% of all cases to achieve a merging maneuver satisfying at least the safe distances between S and the surround Ps. The TTC constraint cannot be fulfilled in 6.6% of the case simply because there is no physically feasible solution for some scenarios. In terms of minimizing the impact on the traffic flow, it reduces the number of triggered lane changes by a factor 8 and the deceleration maneuvers by a factor 3-4 compared to an uncooperative driver. Mean deceleration refers to an overall reduction of velocity, averaged for all vehicles. With an uncooperative entering vehicle, mean traffic velocity is reduced by more than 0.5m/s in 20% of all scenarios, with the optimal controller only in less than 5% of all cases. In comparison min deceleration refers to the vehicle in a scenario, that reduces its velocity most compared to the original measurement data. The proportion of scenarios where at least one vehicle needs to drop its speed by more than 1m/s is reduced by roughly a factor 3 when using the optimized trajectory.

5. CONCLUSIONS

This proof of concept demonstrates that the presented novel models are able to accurately and realistically describe the complex interactions during a merging maneuver. Additional measurement data can be used in the future to further improve the precision and quality of the simulations. The controller proves, that considering the impact of its own actions onto the other traffic participants can help to drastically enhance challenging traffic situations such as merging onto a crowded highway. Even without any information exchange between drivers smooth, safe maneuvers are possible without negative consequences for the traffic flow. The controller is able to optimize trajectories up to the physically possible limits in order to optimally merge into existing gaps instead of relying on others to support the maneuver.

ACKNOWLEDGMENT

This work has been supported by the COMET-K2 "Center for Symbiotic Mechatronics" of the Linz Center of Mechatronics (LCM) funded by the Austrian federal government and the federal state of Upper Austria.

REFERENCES

David Barber. *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

Nathan Gartner, C. Messer, and A. Rathi. *Traffic Flow Theory: A State-of-the-Art Report*. 01 2001.

LLC Gurobi Optimization. Gurobi optimizer reference manual, 2019. URL `http://www.gurobi.com`.

Kshitij Jerath and Sean N Brennan. Analytical prediction of self-organized traffic jams as a function of increasing acc penetration. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1782–1791, 2012.

Pushkin Kachroo and Zhijun Li. Vehicle merging control design for an automated highway system. In *Proceedings of Conference on Intelligent Transportation Systems*, pages 224–229. IEEE, 1997.

Shin Kato and Sadayuki Tsugawa. Cooperative driving of autonomous vehicles based on localization, inter-vehicle communications and vision systems. *Jsae Review*, 22 (4):503–509, 2001.

Robert Krajewski, Julian Bock, Laurent Kloeker, and Lutz Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

Jorge A Laval and Carlos F Daganzo. Lane-changing in traffic streams. *Transportation Research Part B: Methodological*, 40(3):251–264, 2006.

Clark Letter and Lily Elefteriadou. Efficient control of fully automated connected vehicles at freeway merge segments. *Transportation Research Part C: Emerging Technologies*, 80:190–205, 2017.

Xiao-Yun Lu, Han-Shue Tan, Steven E Shladover, and J Karl Hedrick. Automated vehicle merging maneuver implementation for ahs. *Vehicle System Dynamics*, 41 (2):85–107, 2004.

Vicente Milanés, Jorge Godoy, Jorge Villagrá, and Joshué Pérez. Automated on-ramp merging system for congested traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):500–508, 2011.

Takashi Nagatani. Creation and annihilation of traffic jams in a stochastic asymmetric exclusion model with open boundaries: a computer simulation. *Journal of Physics A: Mathematical and General*, 28(24):7079, 1995.

Kai Nagel and Hans J Herrmann. Deterministic models for traffic jams. *Physica A: Statistical Mechanics and its Applications*, 199(2):254–269, 1993.

Jackeline Rios-Torres and Andreas A Malikopoulos. Automated and cooperative vehicle merging at highway on-ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(4):780–789, 2017.

Riccardo Scarinci and Benjamin Heydecker. Control concepts for facilitating motorway on-ramp merging using intelligent vehicles. *Transport reviews*, 34(6):775–797, 2014.

Tieqiao Tang, Haijun Huang, SC Wong, and Rui Jiang. Lane changing analysis for two-lane traffic flow. *Acta Mechanica Sinica*, 23(1):49–54, 2007.

Junqing Wei, John M Dolan, and Bakhtiar Litkouhi. Autonomous vehicle social behavior for highway entrance ramp management. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 201–207. IEEE, 2013.

Chang Yang and K Kurami. Longitudinal guidance and control for the entry of vehicles onto automated highways. In *Proceedings of 32nd IEEE Conference on Decision and Control*, pages 1891–1896. IEEE, 1993.