

# Active Safety System for Semi-Autonomous Teleoperated Vehicles

Smit Saparia<sup>1</sup>, Andreas Schimpe<sup>2</sup> and Laura Ferranti<sup>3</sup>

**Abstract**—Autonomous cars can reduce road traffic accidents and provide a safer mode of transport. However, key technical challenges, such as safe navigation in complex urban environments, need to be addressed before deploying these vehicles on the market. Teleoperation can help smooth the transition from human operated to fully autonomous vehicles since it still has human in the loop providing the scope of fallback on driver. This paper presents an Active Safety System (ASS) approach for teleoperated driving. The proposed approach helps the operator ensure the safety of the vehicle in complex environments, that is, avoid collisions with static or dynamic obstacles. Our ASS relies on a model predictive control (MPC) formulation to control both the lateral and longitudinal dynamics of the vehicle. By exploiting the ability of the MPC framework to deal with constraints, our ASS restricts the controller’s authority to intervene for lateral correction of the human operator’s commands, avoiding counter-intuitive driving experience for the human operator. Further, we design a visual feedback to enhance the operator’s trust over the ASS. In addition, we propose an MPC’s prediction horizon data based novel predictive display to mitigate the effects of large latency in the teleoperation system. We tested the performance of the proposed approach on a high-fidelity vehicle simulator in the presence of dynamic obstacles and latency.

## I. INTRODUCTION

From the introduction of the seat belt to Automatic Emergency Braking (AEB) systems, vehicles have become increasingly safer. Despite the technological improvements, more than 25 thousand fatalities due to road accidents were reported just in the EU in 2017 [1]. These reports also highlighted that  $\approx 95\%$  of the fatalities were caused by human errors. Fully autonomous vehicles might present the solution to reduce such accidents by removing the human from the loop. Fully autonomous vehicles, however, may still fail in traffic situation a human could easily handle, such as parking places or pedestrian crossing areas. To smooth the transition from human-driven vehicles to fully autonomous vehicles, teleoperated driving can play a fundamental role.

In Teleoperated Driving (ToD), a human operator drives the vehicle from a workstation equipped with steering wheel and gas/brake pedals. The generated control commands are transmitted to the vehicle via a mobile network for execution.

<sup>1</sup> S. Saparia was with the Institute for Automotive Technology at the Technical University of Munich (TUM), Germany and Department of Cognitive Robotics, Delft University of Technology, Delft, The Netherlands [smit.saparia@outlook.com](mailto:smit.saparia@outlook.com)

<sup>2</sup> A. Schimpe is with the Institute for Automotive Technology at the Technical University of Munich (TUM), 85748 Garching bei München, Germany. [andreas.schimpe@tum.de](mailto:andreas.schimpe@tum.de).

<sup>3</sup> L. Ferranti is with Department of Cognitive Robotics, Delft University of Technology, Delft, The Netherlands [l.ferranti@tudelft.nl](mailto:l.ferranti@tudelft.nl). Her work is supported by the NWO VENI grant (n. 18165).

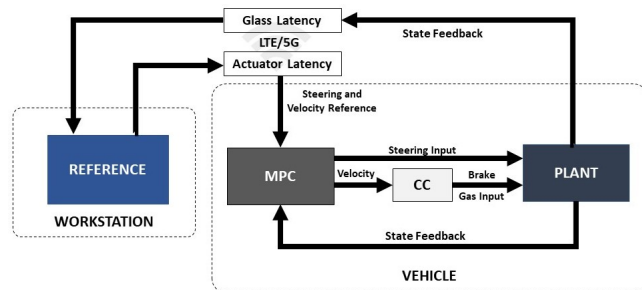


Fig. 1: System architecture for Teleoperated Driving showing major components in the closed loop.

The operator can view the vehicle’s environment on the monitors as captured by vehicle’s cameras. This transmission of signals via the mobile network introduces latency. As Fig. 1 shows, there are two types of latency to consider, that are, *actuator latency* and *glass latency*. The actuator latency is the time taken by the signals to travel from the input devices at workstation to the actuators on the vehicle via the mobile network. The glass latency is the time taken by camera information to travel from camera on the vehicle to the monitors at the workstation via the mobile network. These latency definitions account for the time spent by signals at various stages in hardware as well as in the network.

ToD has several challenges, such as large latency, complete connection loss, or reduced situation awareness. Large latency in the network can lead to unstable real time control of the vehicle [2]. To mitigate the effects of latency, techniques like predictive display can be employed. In worst cases, a complete connection loss poses a threat to make teleoperation useless and dangerous at the same time. Nevertheless, strategies like redundant mobile networks are deployed to overcome such an issue [3]. Another challenge in ToD is reduced situational awareness. It becomes difficult to judge velocity and location of the obstacles and this problem aggravates in the peripheral region of the view. In tight driving scenarios, the chances of collision increase and hence a driver assistance system to help avoid collision proves beneficial as shown in [4]. As Fig. 1 shows, the human operator generates steering and velocity commands that are then manipulated by the controller to generate safe collision free control inputs for the vehicle. Since human operator (from the workstation) and the system (with the final authority) control the vehicle, it can be stated as a case of *uncoupled shared control* [5]. To mitigate the issue of conflict of control interest between human operator and the system, a feedback is necessary to

communicate the intent of the system to the human operator.

### A. Contributions

To address the above challenges, this paper presents an active safety system for collision avoidance using Model Predictive Control (MPC) in a shared control framework specifically for a teleoperated road vehicle, building on the approach presented in [4]. First, we propose a driver assistance system (Sec. II), based on MPC, for a teleoperated semi-autonomous road vehicle. The primary objective is to track the input references as generated by the human operator and intervene when deemed necessary to avoid collision with static or dynamic obstacles. As compared to [4], the approach presented in this paper controls both, the longitudinal and lateral dynamics of the vehicle. In addition, our design introduces a restriction on the controller's authority to override human operator to improve the operator trust on the automation. Furthermore, we introduce potential fields in the MPC formulation to efficiently achieve a collision free manoeuvre. Second, we propose a visual feedback (Sec. III) to communicate system's intent and enhance trust in the human operator over the system to mitigate conflict of control interest. Third, we propose an MPC data-based predictive display technique (Sec. IV) to mitigate effects of high latency in a teleoperation setup. Finally, the proposed approach is validated using a high-fidelity vehicle simulator with the simulated human operator (Sec. V).

### B. Related Work

Our work spans across different areas. For this reason, this section provides an overview of the relevant literature on shared control, obstacle avoidance, visual feedback and predictive display with a main focus on MPC-based approaches.

*Shared Control:* Several shared control approaches have been proposed in the literature, such as model-based coupled shared control, model-based uncoupled shared control, model-free uncoupled shared control (as defined in [5]). In uncoupled shared control, automation has final authority to implement the input to the vehicle unlike in coupled shared control. A model-based shared control models the shared control behaviour, mostly using a driver model. A model-based coupled shared control approach is followed in [6], where human driver and steering system models are incorporated into a lateral controller design. Modelling the driver requires actual driving tests as shown in [7] and the driver model becomes specific to a particular driving style.

A model-free uncoupled shared control approach is followed in [8], where a final steering command as a weighted combination of driver and automation input according to a tailored threat metric is computed. Tuning of this threat metric is not trivial and only lateral control was showcased in this work. Finally, a model-based uncoupled shared control approach is followed in [9], [10] and [4]. In [9], a model predictive contouring control (MPCC) framework is used for automation where center of the lane is considered as a reference path. Another optimization problem considers driver inputs and a decaying function is used to compute a

weighted combination of driver and automation inputs. This approach considers both lateral and longitudinal control, but cannot be used for the case of teleoperation where availability of information of lane center via perception module is not reliable. The authors in [10] and [4] directly incorporate driver's steer commands into the objective function of MPC for only lateral control, where the objective is to match the driver inputs.

*Obstacle Avoidance:* Obstacles and ego vehicle can be directly modelled in the constraints of the MPC formulation as *geometrical shapes*, based on which a collision free criterion can be incorporated into the problem formulation of controller, as proposed, for example, in [11] and [12] in the context of fully autonomous vehicles. Reactive methods, such as potential fields, can also be incorporated in the MPC formulation instead of explicitly modelling obstacles in the constraints. The authors of [13] represented crossable and non-crossable obstacles with different potential fields. The authors of [4] modelled the obstacles as higher-order ellipses and formulated a potential function such that the front vehicle corners are pushed away from the obstacles. This potential function is incorporated into the objective of the MPC problem formulation. The advantage of using potential function over spatial constraint is that the size of the optimization problem does not grow with the number of obstacles in the scenario [9], [11]. Therefore, the solution time of the problem is not expected to increase with the increase in the number of obstacles.

*Visual Feedback:* A feedback is necessary for the controller to communicate its intent to the human driver. A haptic feedback on steering wheel is used in [14], to assist human operator in challenging situations and helps prevent collision of teleoperated vehicle. However, haptic feedback on steering wheel can be used only for lateral control. The authors in [15] showcased the benefit of having visual feedback along with haptic shared control for obstacle avoidance. Haptic feedback on steering wheel cannot communicate future vehicle state that might affect current driver inputs. This is supposedly compensated by a visual feedback. Monitors in teleoperation can be used to illustrate future vehicle positions. This advantage is exploited in this work.

*Predictive Display:* As presented in [16], predictive display can be used to predict ego vehicle position, to compensate for the delay in signal transmission. This involves projecting the predicted vehicle position images on the video stream. However, these techniques assume constant vehicle velocity and road wheel angle in the predictions.

## II. SHARED CONTROL APPROACH

### A. Prediction Model

The teleoperated vehicle is expected to be driven at slow speeds with moderate driving. Hence, the prediction model chosen in this approach for the MPC is the following

kinematic bicycle model [17]:

$$\dot{x} = v \cos(\psi + \beta) \quad (1a)$$

$$\dot{y} = v \sin(\psi + \beta) \quad (1b)$$

$$\dot{\psi} = \frac{v}{l_r} \sin(\beta) \quad (1c)$$

$$\dot{v} = a \quad (1d)$$

$$\beta = \tan^{-1} \left( \frac{l_r}{l_f + l_r} \tan(\delta) \right) \quad (1e)$$

where  $x$  and  $y$  are coordinates of the center of mass (CoM) of the vehicle,  $\psi$  is the heading angle of the vehicle in the inertial frame,  $v$  is the vehicle velocity,  $l_f$  and  $l_r$  are the distances between CoM and front and rear axles respectively.  $\beta$  is the angle that the velocity vector makes with the longitudinal axis of the vehicle,  $a$  is the acceleration of the CoM and  $\delta$  is the road wheel angle (RWA) of the front wheels. To simplify the notation, we define the following states and input quantities, that are,  $z := [x, y, \psi, \delta, v]$ , and  $u := [\dot{\delta}, a]$  respectively.

### B. Modelling Ego Vehicle and Rectangular Obstacles

The collision avoidance criteria are incorporated in the MPC problem formulation by approximating the ego vehicle and the rectangular obstacle as geometric shapes. The ego vehicle is represented with four longitudinally shifted circles of radius  $r$ , along its longitudinal axis [12]. The benefit of representing the ego vehicle with circles instead of just using front vehicle corners as in [4] for collision avoidance is that it ensures that the ego vehicle stays collision free from the obstacles with a downside of an increase in the computational burden. A rectangular obstacle is represented with a higher order ellipse. The implicit function, as proposed in [4], that describes the shape of the obstacle in inertial frame is given by,

$$e_i^m(x_{c_i}, y_{c_i}) = \left( \frac{R(\phi)(x_{c_i} - x_{obs}^m(t))}{\alpha_{maj}} \right)^n + \left( \frac{R(\phi)(y_{c_i} - y_{obs}^m(t))}{\beta_{min}} \right)^n - 1 \quad (2)$$

for  $i \in \{1, 2, 3, 4\}$ . Here,  $(x_{c_i}, y_{c_i})$  represents the center of the  $i^{th}$  ego vehicle circle and  $(x_{obs}^m(t), y_{obs}^m(t))$  represents the center of the  $m^{th}$  obstacle.  $R(\phi)$  is the rotation matrix, depending on the heading  $\phi$  of the obstacle. The semi-axis of the  $\alpha$  and  $\beta$  and the even order  $n$  of the ellipse help define the size of the bounding ellipse. Therefore, we define

$$\alpha_{maj} = a + r \quad \text{and} \quad \beta_{min} = b + r, \quad (3)$$

where  $a := f \cdot L/2$ ,  $b := f \cdot B/2$ ,  $f = \sqrt[3]{2}$  and  $r$  is the radius of the ego vehicle.  $L$  and  $B$  are the length and breadth of the obstacle respectively.

### C. Potential Field for Collision Avoidance

Building on [4], we rely on a repulsive potential field approach for collision avoidance. The potential field  $P$  is a sum of individual potential functions for  $m$  obstacles for all the  $i$  circles of the ego vehicle, that is,

$$P = \sum_m \sum_i P_i^m(x, y), \quad (4)$$

where the potential function, for the  $i^{th}$  circle for the  $m^{th}$  obstacle is as follows

$$P_i^m(x, y) = \frac{\tau}{(e_i^m(x_{c_i}, y_{c_i}) + 1)^\rho}. \quad (5)$$

In the equation above,  $\tau$  and  $\rho$  are the design parameters that specifies the strength and slope of the potential function. It should be noted that  $e^m$  is the implicit function that represents the shape of the  $m^{th}$  obstacle, which is already given by (2). When the ego vehicle is in collision with the obstacle, the value of  $e^m$  becomes zero and hence  $P_i^m$  equates to  $\tau$ . To make the obstacle as non-crossable for the vehicle, this information is used to formulate a constraint in MPC formulation.

The potential field approach for collision avoidance ensures that the size of MPC problem stays constant irrespective of the number of obstacle, an advantage over the spatial constraint approach.

### D. Model Predictive Control Formulation

We rely on MPC to achieve the navigation objectives. The MPC controller repeatedly solves a constrained optimization problem to compute an optimal sequence of control commands that minimize a desired cost function  $J$  over a finite horizon  $N$ . At every sampling instance, only the first element of this control sequence is applied in closed loop. This section details our MPC problem formulation.

The objectives of the controller are (i) to match the human operator's inputs (i.e.,  $\delta$  and  $v$ ), (ii) ensure collision avoidance, and (iii) restrict the controller authority to intervene to avoid surprises for the human operator. To achieve the objectives above we define the following MPC:

$$\min_{z, u, S} \sum_{k=1}^N J_k(z_k, u_k, S_k) \quad (6a)$$

$$\text{subject to } z_{k+1} = f(z_k, u_k) \quad (6b)$$

$$z_0 = z(t) \quad (6c)$$

$$\delta_{min} \leq \delta_{k+1} \leq \delta_{max} \quad (6d)$$

$$v_{min} \leq v_{k+1} \leq v_{max} \quad (6e)$$

$$\dot{\delta}_{min} \leq \dot{\delta}_k \leq \dot{\delta}_{max} \quad (6f)$$

$$a_{min} \leq a_k \leq a_{max} \quad (6g)$$

$$\delta_{dev \ min} - s_{k+1}^\delta \leq \delta_{k+1} - \delta_{ref}(t) \leq \delta_{dev \ max} + s_{k+1}^\delta \quad (6h)$$

$$P_{k+1, i} \leq \tau + s_{k+1}^P \quad (6i)$$

$$k \in (0, N - 1),$$

where  $z_k$ ,  $u_k$ , and  $S_k$  are the state, command, and slack variables (discussed below) at prediction step  $k$ . The MPC controller minimizes the cost (6a) subject to the constraints (6b)-(6i). The cost  $J_k$  is defined as follow:

$$J_k := J_k^P + J_k^\delta + J_k^v + J_k^S. \quad (7)$$

The first term of the cost function, namely  $J^P$ , is associated with the potential field to ensure a collision free motion and it is defined as follows:

$$J_k^P = W_P P_k(x_k, y_k), \quad (8)$$

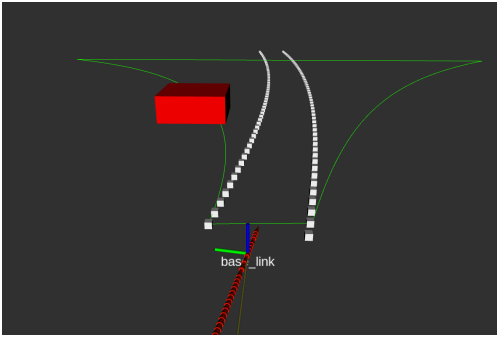


Fig. 2: Rviz<sup>1</sup> illustration of visual feedback to be added on top of video stream on the screens at workstation. The maximum and minimum vehicle trajectory due to the corresponding controller intervention is shown by the green controller authority cone. Obstacles are represented by red cubes.

where  $W_P$  is a weighting factor and  $P$  is defined according to (4), for all the circles representing the ego vehicles and for all the obstacles.

The second term of the cost function penalizes the deviation of controller input  $\delta$  from the reference or human operator's input  $\delta_{\text{ref}}$  with the penalty  $W_\delta$ ,

$$J_k^\delta = W_\delta (\delta_{\text{ref}}(t) - \delta_k)^2 \quad (9)$$

The third term of the cost function penalizes the deviation of controller input  $v$  from the reference or human operator's input  $v_{\text{ref}}$  with the penalty  $W_v$

$$J_k^v = W_v (v_{\text{ref}}(t) - v_k)^2 \quad (10)$$

The last term in the cost function is used to penalize, with a penalty  $W_s$ , slack variables  $S_k := [s_k^\delta, s_k^p]^T$  which are used to soften constraints (6h) and (6i):

$$J_k^S = W_s S_k^2. \quad (11)$$

Constraint (6b) represents the dynamic coupling. In addition, note that  $f(z_k, u_k)$  represents the discretized kinematic bicycle model equations presented in equations (1). Constraints (6d) and (6f) limits the RWA  $\delta$  and its rate of change  $\dot{\delta}$ , respectively. Furthermore, Constraints (6e) and (6g) limit the velocity  $v$  and acceleration  $a$  of the vehicle, respectively. Constraint (6h) restrict the controller's authority to deviate from the reference RWA at time  $t$ . This makes the controller action moderate in the lateral sense and helps avoid larger mismatch between human operator's intentions and the controller's action. Finally, together with the potential field in the cost function, constraint (6i) helps ensure collision avoidance by making the obstacles uncrossable. The upper limit of this constraint is governed by  $\tau$  as found in (5). Constraint (6i) is softened to practically avoid infeasible solutions, while a high penalty on the slack variables in the cost function limits such a constraint violation.

<sup>1</sup><http://wiki.ros.org/rviz>

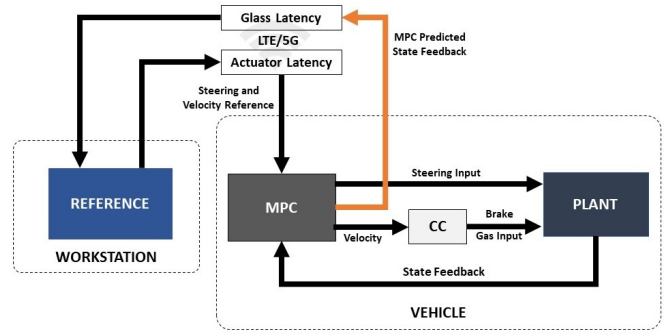


Fig. 3: System Architecture for MPC Data-Based Predictive Display.

### III. MPC DATA-BASED VISUAL FEEDBACK

To further raise human operator's trust and understanding of the system behaviour, a feedback can be used to communicate the intention of system to the human operator as discussed in I-B. Our proposed visual feedback contains information on the region of future vehicle positions due to controller authority, as defined by constraint (6h), and future vehicle path, shown by white tracks, due to complete sequence of steering wheel angle and velocity inputs by controller. This feedback is proposed to be added as graphical illustration on top of the video stream on screens at workstation. The visual feedback is expected to help the human operator better understand the possible positions of the vehicle according to both the operator's and controller's inputs. It aims at avoiding any surprises or conflict if the controller intervenes. This concept is illustrated in Fig. 2.

### IV. MPC DATA-BASED PREDICTIVE DISPLAY

This section discusses our proposed predictive display method. Our predictive display exploits the MPC predictions to mitigate the effects a high network latency can have on the teleoperation. Recall that the reference module in Fig. 1 generates reference inputs based on the state feedback it receives, which is delayed by time equal to glass latency. These reference inputs are further delayed by a time equal to the actuator latency during the transmission to the MPC module. During this time the ego vehicle would have travelled by time equal to this total round trip latency. The goal of our predictive display is to attempt predicting the ego vehicle position during the time frame of a total round trip latency. This predicted vehicle position can be used to enrich the camera images before they are displayed on the screens at the workstation.

Authors in [16] propose a *full prediction* method, which relies on a bicycle model (as presented in (1)) for prediction assuming constant vehicle velocity and RWA. This method is taken as baseline for comparison with the proposed MPC data based Predictive Display in section V-B.

In our approach, the idea is that MPC's predicted state corresponding to time equal to total round trip latency could be used as state feedback. This simplifies the process and needs no new calculations as in [16]. MPC already uses a

model of the system and sets of constraints and objective to calculate optimal evolution of states and inputs. This method also eliminates the assumptions of constant velocity and RWA. Fig. 3 illustrates this concept.

## V. SIMULATION RESULTS

The validation of the approach was carried out with simulations in MATLAB/Simulink<sup>®</sup>. For simulations, a high fidelity car model from IPG CarMaker [18] was used as plant to test the designed controller. A Volvo XC90 T6 AWD was used as the testing vehicle. The optimization of MPC problem is carried out through *acados*, an open source software package with a collection of solvers for fast embedded optimization intended for fast embedded applications [19]. Sequential Quadratic Programming (SQP) is used as the solver by *acados*. The algorithm runs well below 50 ms on an Intel Core i7-4600 2.10 Ghz 2 Core CPU with 8 GB of RAM and hence it can be expected to perform even better when run on a powerful hardware of the experimental vehicle. The system architecture for ToD is shown in Fig. 1. The architecture consists of two main blocks, that are, (i) the workstation, where the human operator generates reference inputs of velocity and RWA, (ii) the teleoperated vehicle equipped with the proposed MPC controller. In a real setup, the signals between these two blocks are transmitted over a commercial 4G LTE network and hence delayed due to network latency. In a simulation setup, delay blocks are added separately to artificially introduce this latency in simulations. For simulations, the reference block generates reference velocity, whose values are predefined as a function of time. In addition to this, the reference block simulates the human operator using a Feedback Linearized Path Tracking Controller (FBLT), taken from [20]. The control law used to simulate the human operator is defined as follows

$$\delta_{\text{FBL}}(t) = \arctan\left(\frac{-\gamma_1 e_L(t) - \gamma_2 v \sin(e_H(t))}{v^2 \cos(e_H(t))}\right) \quad (12)$$

where, the lateral and heading errors are  $e_L$  and  $e_H$ , respectively, and  $v$  is the current vehicle velocity. The correction for heading error and lateral error can be prioritized using the gains  $\gamma_1$  and  $\gamma_2$ . The errors,  $e_L$  and  $e_H$  are computed with respect to the reference tracking point along the trajectory, which is again a function of lookahead distance. To include the effect of visual feedback to the human operator, we added another feedback term. This term penalizes the deviation between the operator SWA (RWA) or the calculated  $\delta_{\text{FBL}}$  and the current SWA (RWA), with the proportional gain  $\gamma_3$ . Thus, the overall simulated human operator's RWA input becomes:

$$\delta_{\text{ref}}(t) = \delta_{\text{FBL}}(t) + \gamma_3(\delta(t_0) - \delta_{\text{FBL}}(t)) \quad (13)$$

The CarMaker plant cannot take velocity input directly and hence a Cruise Control (CC) is designed to translate the desired velocity from MPC into corresponding Gas and Brake pedal values. A basic CC is formulated based on a proportional-integral controller.

The sampling time of MPC is set to  $t_s = 50$  ms since the solution times are well within this limit. The order of

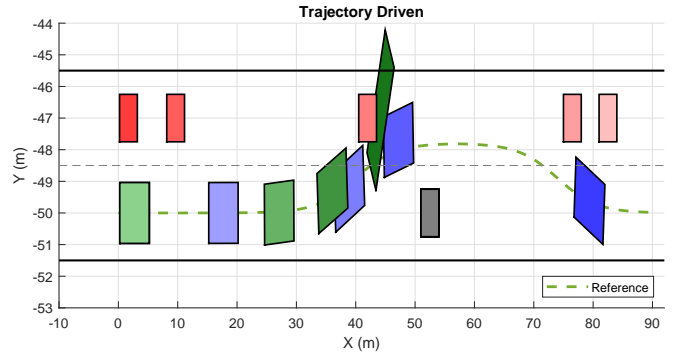


Fig. 4: Trajectory Driven for Overtake Scenario. Progression of ego vehicle with ASS is shown in shades of blue. Progression of ego vehicle with Baseline controller is shown in shades of green. Progression of oncoming obstacle (dynamic) is shown in shades of red. Stationary obstacle is shown in grey color. The shades go from light to dark with the progression of time.

the ellipse  $n$  is taken as 4. The potential function parameters are tuned to  $\tau = 0.1$  and  $\rho = 2$ . This tuning results in a close manoeuvre of the vehicle around the obstacles. ToD is expected to be performed at low speed due to the perils of remote driving at high speed. Therefore, these simulations consider only low speed (3 m/s) scenarios where the effect of latency is not so crucial and hence a constant latency in the simulation is considered. In the following, the performance of the proposed controller, namely, our Active Safety System (ASS), is compared with a baseline approach which does not have longitudinal control capability and restriction on deviation of RWA from its corresponding reference. Section V-A compares the two approaches in a scenario that involves overtaking a stationary obstacle with another dynamic obstacle in the scene. In addition, Section V-B illustrates our MPC-data-based predictive display concept.

### A. Overtake Scenario

This scenario considers the ego vehicle trying to overtake a stationary obstacle along its path (e.g., a parked car). While overtaking, the ego vehicle has to deal with an upcoming vehicle proceeding in the opposite lane. At every sampling time, the dynamic obstacles are elongated in size over the prediction horizon assuming their current heading and velocity stays constant. Effectively converting them into large static obstacles with changing locations and dimensions. Thus, collision avoidance of the dynamic obstacles is accounted for in a conservative manner, with a possibility of reducing this conservatism by accounting for predicted obstacle positions along the horizon by a perception module. Fig. 4 shows the behavior of the ego vehicle with our ASS approach. The reference path (green dashed line) of the ego vehicle's CoM is purposefully designed in such a way that the ego vehicle collides with the stationary obstacle while going around. The ASS-based ego vehicle brakes when encountering the oncoming obstacle, and then overtakes the stationary obstacle when there is free space.

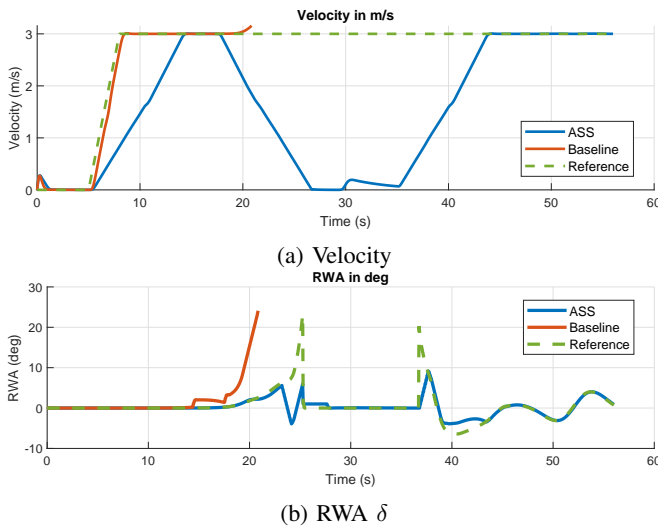


Fig. 5: Longitudinal velocity and RWA of the ego vehicle in the overtake scenario.

This behavior is reflected in the velocity and RWA results in Fig. 5, which compares our method (blue lines) with the baseline controller (red lines). The velocity plot in Fig. 5a highlights the contribution of our ASS on the control of the vehicle. As the figure shows, our system reduces the vehicle velocity to prevent collisions, in contrast with the constant reference signal provided by the operator. As Fig. 5b highlights, our ASS avoids collision by correcting the RWA at time 40s, where the RWA differs from reference RWA. Thus, longitudinal and lateral control capabilities of the ASS are highlighted here. The vehicle with Baseline controller fails to stop after encountering the oncoming obstacle and collides with it since it does not have any longitudinal control capability. The end of its trajectory represents the point where the vehicle leaves the road boundary and hence the simulation gets aborted by CarMaker. This highlights the drawback of not having longitudinal control capability. In addition to this, the RWA plot in Fig. 5b also shows a large deviation of RWA from reference RWA. This steering behaviour is highly counter-intuitive to what the human operator wishes to do, potentially raising trust issues over the system. In comparison to this, even though the ASS steering profile is showing considerable deviation from the reference RWA, it remains well around the range of reference and follows the reference back again. This highlights the importance of restricting the controller’s authority to deviate from the reference RWA by using the constraint (6h).

### B. Predictive Display

The predictive display is expected to nullify the effect of latency and produce results equivalent to that for no latency in system. Fig. 6 compares the performance of our ASS with our proposed predictive display (dashed purple line) with the baseline approach [16] (solid light blue line) and our ASS without predictive display (solid red line). The performance of our method with no latency is used as a benchmark. In

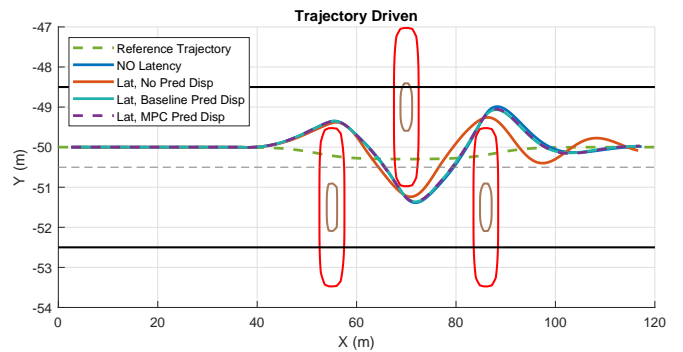


Fig. 6: Trajectory of the CoM of the ego vehicle driving through closely placed obstacles with latency.

addition, the stationary obstacles are represented by higher order ellipse (brown) and they are inflated by the radius of the ego circle as represented by the large ellipses (red). The reference trajectory (green dashed line) is purposefully designed for a collision with obstacles. In this simulation, we consider a total round trip latency of 500 ms. The trajectory driven by the CoM of the ego vehicle for both the cases with predictive displays (Baseline and MPC data-based) overlap the trajectory with no latency in the system. This proves that both predictive display techniques equally nullify the effect of latency. Nevertheless, the advantage of the proposed MPC-data-based predictive display method over the baseline method lies in the fact that it does not assume constant velocity and RWA for predicting vehicle position. It also removes extra calculation step of predicting vehicle position using a bicycle model as done in [16].

It should be noted that, even in the absence of any latency mitigation technique, the vehicle trajectory stays collision free, verifying that the controller is efficient to avoid collision even with delayed (outdated) reference inputs.

## VI. CONCLUSION

This paper presented an ASS approach for teleoperated vehicles based on model predictive control. The ASS controls both the lateral and longitudinal vehicle’s dynamics allowing the vehicle to perform a large range of maneuvers and avoid collision by both braking and steering intervention. Collision avoidance is achieved by using artificial potential fields in the MPC controller. Our ASS also has an added restriction on the controller’s authority to prevent large deviations from the human operator’s RWA inputs. In addition, we proposed a visual feedback in the form of graphical presentation on the monitors of workstation. Finally, to mitigate the effects of large latency in a teleoperation setup, we proposed a novel MPC data-based predictive display method. We tested our approach by using a high fidelity vehicle model for plant and a realistic environment. The simulation results highlighted the advantages of our ASS for teleoperation compared to existing baselines. Future work includes experimental validation of the proposed work.

## APPENDIX

The MPC penalties and FBLC gains for the corresponding scenarios are summarized in the following table.

TABLE I: MPC penalties and FBLC gains.

	$W_P$	$W_\delta$	$W_v$	$W_s$	$\gamma_1$	$\gamma_2$	$\gamma_3$
Overtake Scenario	0.1	$10^3$	1	$10^5$	1	2	0.25
Predictive Display	0.1	$10^2$	1	$10^5$	1	2	0.25

## ACKNOWLEDGMENT

This work was presented at the workshop for Road Vehicle Teleoperation (WS09), IV2021.

## REFERENCES

- [1] EPRS European Commission 2017. (2019) Road fatality statistics in the eu (infographic). [Online]. Available: <https://www.europarl.europa.eu/news/en/headlines/society/20190410STO36615/road-fatality-statistics-in-the-eu-infographic>
- [2] H. Pongrac, "Gestaltung und evaluation von virtuellen und telepräsenzsystemen an hand von aufgabenleistung und präsenzempfinden," 2011.
- [3] R. Baldwin. (2018) Phantom auto will drive your autonomous car if it gets confused. [Online]. Available: <https://www.engadget.com/2018-04-05-phantom-auto-will-drive-your-autonomous-car-if-it-gets-confused.html>
- [4] A. Schimpe and F. Diermeyer, "Steer with me: A predictive, potential field-based control approach for semi-autonomous, teleoperated road vehicles," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020, pp. 1–6.
- [5] M. Marcano, S. Díaz, J. Pérez, and E. Irigoyen, "A review of shared control for automated vehicles: Theory and applications," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 6, pp. 475–491, 2020.
- [6] Z. Ercan, A. Carvalho, S. Lefèvre, F. Borrelli, H. E. Tseng, and M. Gokasan, "Torque-based steering assistance for collision avoidance during lane changes," 2016.
- [7] P. Falcone, M. Ali, and J. Sjöberg, "Predictive threat assessment via reachability analysis and set invariance theory," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1352–1361, Dec 2011.
- [8] S. Anderson, S. Peters, T. Pilutti, and K. Iagnemma, "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios," *International Journal of Vehicle Autonomous Systems*, vol. 8, pp. 190–216, 10 2010.
- [9] W. Schwarting, J. Alonso-Mora, L. Paull, S. Karaman, and D. Rus, "Safe nonlinear trajectory generation for parallel autonomy with a dynamic vehicle model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2994–3008, Sep. 2018.
- [10] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Shared steering control using safe envelopes for obstacle avoidance and vehicle stability," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 441–451, Feb 2016.
- [11] L. Ferranti, B. Brito, E. Pool, Y. Zheng, R. M. Ensing, R. Happee, B. Shyrokau, J. F. P. Kooij, J. Alonso-Mora, and D. M. Gavrila, "Safevru: A research platform for the interaction of self-driving vehicles with vulnerable road users," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, June 2019, pp. 1660–1666.
- [12] B. Brito, B. Floor, L. Ferranti, and J. Alonso-Mora, "Model predictive contouring control for collision avoidance in unstructured dynamic environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4459–4466, Oct 2019.
- [13] Y. Rasekhipour, A. Khajepour, S. Chen, and B. Litkouhi, "A potential field-based model predictive path-planning controller for autonomous road vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1255–1267, May 2017.
- [14] A. Hosseini, F. Richthammer, and M. Lienkamp, "Predictive haptic feedback for safe lateral control of teleoperated road vehicles in urban areas," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016, pp. 1–7.
- [15] W. Vreugdenhil, S. Barendswaard, D. A. Abbink, C. Borst, and S. M. Petermeijer, "Complementing haptic shared control with visual feedback for obstacle avoidance," *IFAC-PapersOnLine*, vol. 52, no. 19, pp. 371 – 376, 2019, 14th IFAC Symposium on Analysis, Design, and Evaluation of Human Machine Systems HMS 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2405896319319263>
- [16] F. Chucholowski, S. Büchner, J. Reicheneder, and M. Lienkamp, "Prediction methods for teleoperated road vehicles," in *Conference on Future Automotive Technology - Focus Electromobility, Garching, Bayern Innovativ GmbH*, 01 2013.
- [17] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1094–1099.
- [18] Ipg automotive gmbh. [Online]. Available: <https://ipg-automotive.com/products-services/simulation-software/carmaker/>
- [19] R. Verschuere, G. Frison, D. Kouzoupis, N. van Duijkeren, A. Zanelli, B. Novoselnik, J. Frey, T. Albin, R. Quirynen, and M. Diehl, "acados: a modular open-source framework for fast embedded optimal control," *arXiv preprint*, 2019. [Online]. Available: <https://arxiv.org/abs/1910.13753>
- [20] K. Burnett, A. Schimpe, S. Samavi, M. Gridseth, C. W. Liu, Q. Li, Z. Kroeze, and A. P. Schoellig, "Building a winning self-driving car in six months," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9583–9589.