Nabil Aouf

ASME Paper Title: A Comparison of Trajectory Planning and Control Frameworks for Cooperative Autonomous Driving

Authors: Viana, I. B., Kanchwala, H., Ahiska, K. & Aouf, N.

ASME Journal Title: Journal of Dynamic Systems, Measurement, and Control

Volume/Issue 143/7 Date of Publication (VOR* Online) 4/2/2021

ASME Digital Collection URL:

DOI: https://doi.org/10.1115/1.4049554

*VOR (version of record)

Journal of Dynamic Systems, Measurement and Control.
Received August 09, 2020;Accepted manuscript posted January 7, 2021.
doi:10.1115/1.4049554 Copyright (c) 2021 by ASME

Journal of Dynamic Systems, Measurement and Control
Volume 143, Issue 7, July 2021,
Article number DS-20-1318

A Comparison of Trajectory Planning and Control Frameworks for Cooperative Autonomous Driving

Icaro Bezerra Viana[*]
Centre for Electronic Warfare,
Information and Cyber,
Cranfield University,
Defence Academy of the UK,
Shrivenham, SN6 8LA
icaro.bezerra-viana@cranfield.ac.uk

Husain Kanchwala
Warwick Manufacturing Group,
University of Warwick, UK,
Coventry CV4 7AL
husain.kanchwala@warwick.ac.uk

Kenan Ahiska
Centre for Electronic Warfare,
Information and Cyber,
Cranfield University,
Defence Academy of the UK,
Shrivenham, SN6 8LA
k.ahiska@cranfield.ac.uk

Nabil Aouf
Dept. of Electrical and Electronic Engineering,
City, University of London,
Northampton Square, EC1V 0HB
nabil.aouf@city.ac.uk

This work considers the cooperative trajectory-planning problem along a double lane change scenario for autonomous driving. In this paper we develop two frameworks to solve this problem based on distributed model predictive control (MPC). The first approach solves a single non-linear MPC problem. The general idea is to introduce a collision cost function in the optimization problem at the planning task to achieve a smooth and bounded collision function and thus to prevent the need to implement tight hard constraints. The second method uses a hierarchical scheme with two main units: a trajectory-planning layer based on mixed-integer quadratic program (MIQP) computes an on-line collision-free trajectory using simplified motion dynamics, and a tracking controller unit to follow the trajectory from the higher level using the non-linear vehicle model. Connected and automated vehicles (CAVs) sharing their planned trajectories lay the foundation of the cooperative behaviour. In the tests and evaluation of the proposed methodologies, MATLAB-CARSIM co-simulation is utilized. CARSIM provides the high fidelity model for the multi-body vehicle dynamics. MATLAB-CARSIM conjoint simulation experiments compare both approaches for a cooperative double lane change maneuver of two vehicles moving along a one-way three-lane road with obstacles.

1 Introduction

Industrial as well as academic interest on advanced driver assistance systems (ADAS) increases significantly as

[*] Corresponding author.

they can decrease road fatalities [1]. A higher level autonomy such as crash mitigation, collision avoidance, autonomous driving and autonomous platooning attract more attention compared to conventional drive assist technologies such as anti-lock braking systems (ABS), power-steering, adaptive cruise controllers (ACC) and electronic stability controllers (ESC) [2]. In general a hierarchical control architecture is employed in a collision avoidance scheme: the higher level layer generates a collision-free path, and the lower level layer is a path tracking controller, and it follows the planned path by controlling the actuators [3],[4].

From inter-vehicle communication (IVC) systems point of view, the vehicle-to-everything (V2X) communication technologies [5] provide the autonomous vehicles with the capability to communicate with each other, and it is possible now to transfer sensor data between vehicles and plan cooperative maneuvers. The cooperative path-planning problem has been studied extensively in the literature [6],[7],[8].

Mixed-integer programming is proposed to generate the optimal multiple vehicle path-planning [6]. In the work of Frese and Beyerer [7], a comparison between several cooperative path planning algorithms is available. The authors considered computation time and the quality of the path planner in dangerous traffic situations as the comparison measures for the elastic band method, the tree search algorithm, a priority-based approach and mixed-integer linear programming. For a cooperative lane change, Wang et al. [8] used model predictive control (MPC) to attenuate the inauspicious effects during the lane change maneuver in traffic.

Alternative to the hierarchical schemes, an integrated trajectory planning and tracking controller approach is also possible. In such a scheme, the integrated controller will directly optimise the driving forces and the steering angles for cooperative collision avoidance. Yuan et al. proposed a unified path-planner and tracking controller and in the quadratic programming form, they used a simple formulation of MPC [9]. The collision avoidance is achieved simply by imposing constraints on the vehicle position. However, in the case of dynamic obstacles, as the scenario gets more complex, the performance of the MPC optimiser is usually not adequate. Furthermore, the approach is not applicable to the problem of multiple vehicles.

In [10], a proactive collision avoidance scheme was proposed. In this work, a mixed-integer quadratic programming (MIQP) formulation is utilised to achieve the collision avoidance via hard constraints imposed in the optimisation problem. In [11], a potential field-like term is introduced to the cost function to handle the collision avoidance requirements. This is an example of soft constraints where inequality constraints are defined indirectly with penalty functions [12]. In [13], it is stated that a penalty term is advantageous compared to an additional inequality constraint.

The objective of this paper is to extend the work proposed by the authors in [11] by comparing the unified control approach with a hierarchical scheme for cooperative trajectory-planning. This work defines a single constrained non-linear optimisation problem and the control commands of the actuators are attained directly instead of using a separate path-tracking controller. The baseline approach generates plans based on receding-horizon mixed-integer programming, similar to the approaches presented in [6]. This contribution compares both algorithms in terms of their functionalities and computation costs, discussing the advantages and limitations of the proposed methodologies.

The rest of the text is organized as follows: Section 2 presents an overview of the available literature in cooperative trajectory planning. Section 3 describes the vehicle model. Section 4 formulates the cooperative trajectory-planning problem using a non-linear model predictive controller. Section 5 presents the cooperative trajectory-planning approach based on receding-horizon mixed-integer quadratic programming. Section 6 includes the results from the MATLAB-CARSIM co-simulation framework in a two-vehicle obstacle avoidance scenario. Finally, Section 7 concludes the paper with future work suggestions.

2 Related Works

With increasing intelligent road and vehicle-to-vehicle (V2V) communication systems, cooperative path planning concept extensively studied in the literature has been validated in experimental works. Ji et al. used multi-constrained model predictive control (MMPC) for path planning and tracking control for vehicle collision avoidance problem and validated their approach in Simulink and Carsim simulations [14]. Shibata et al. used velocity potential fields to study steering-based collision avoidance solution in simulations [15].

Trajectory planning for autonomous lane changing manoeuvre in smart roads is among the most studied problems in vehicle control. You et. al used a polynomial and back-stepping approach to tackle trajectory planning and tracking problems, and with simulations and experimental results, they demonstrated the feasibility of their approach [16]. Nilsson and Sjoberg used model predictive control in decision making for benefits of lane change manoeuvre [17]. Schildbach and Borelli utilized model predictive control design for lane change in traffic [18]. Wang et al. used model predictive control in cooperative lane change problem to improve the traffic flow by reducing the deceleration of the following vehicle [8].

Alternative approaches based on elastic bands [19], a technique between global path planning and real-time motion control have been studied in vehicle motion control as well [20]. Gehriq and Stein used elastic bands in collision avoidance for vehicle-following systems along the lateral direction, and in simulations and real-world results they demonstrated the efficiency of their proposal [21]. As an interesting extension, cooperative trajectory planning for vehicles in the absence of the speed lanes using elastic strips is also available [22].

The literature of vehicle trajectory planning uses another widely-utilized technique: approaches based on tree-search. Lenz et al. used Monte-Carlo tree search in tactical cooperative planning with possible actions of lane change, accelerate, brake and stop [23]. Kurzer et al. showed the effectiveness of decentralized cooperative path planning achieved with continuous Monte Carlo tree search compared to egocentric planning [24].

Optimization based approaches have been utilized in solution of vehicle path planning and obstacle avoidance problems as well. Different optimization objective criteria such as minimum time [25], minimum kinetic energy [26], minimum covered distance [27], maximum safe distance [28] and maximum comfort [29] have been employed. Gabarron et al. used multi-objective vehicle trajectory optimization process along lateral direction only, for cooperative collision avoidance in high-speed vehicles [30].

Eilbercht and Strusberg formulated manoeuvres in cooperative motion plans as discrete time hybrid automation and used mixed integer quadratic programming (MIQP) to find a solution to the optimization problem [31]. Similarly, Burger and Lauer used MIQP in cooperative multiple vehicle trajectory planning and in numerical experiments they showed the advantages of this approach over priority-based and non-cooperative individual motion planning [32]. MIQP have been utilized in robot motion planning and trajectory optimization for goal assignment as well [33].

Branca and Fierro proposed heuristics to improve the efficiency of the hierarchical and decentralized optimization problem in cooperative trajectory planning of various number of vehicles equipped with mixed-integer linear programming (MILP) and MPC path planning schemes in an environment with different number of obstacles [34]. Schouwenaars et al. [6] used AMPL and CPLEX optimization tools for mixed integer programming in multiple vehicle path planning. Miller et al. benefited from decomposing the motion and used MILP in longitudinal and lateral motion planning and proposed solution to eliminate infeasible trajectories [35].

3 Preliminaries and Vehicle Model

3.1 Notation

The block diagram of the two cooperative planning frameworks is shown in Fig. 1. The first approach is a NMPC-based trajectory planner designed with a non-linear bicycle model. The second approach consists of a hierarchical architecture based on mixed-integer quadratic programming (MIQP) for trajectory-planning and a linear MPC for tracking purpose.

Assume $\mathcal{V}$ denotes the set of vehicles $\mathcal{V} \triangleq \{1, 2, ..., N_v\}$. The trajectory-planner generates two-dimensional trajectories $\mathbf{r}_p^{(e)} \triangleq [r_{p,x}^{(e)}\ r_{p,y}^{(e)}]^\mathrm{T} \in \mathbb{R}^2$ of an ego vehicle $e \in \mathcal{V}$ in accordance with a predetermined plan. The subscript p marks the trajectories generated by the trajectory-planner. To plan the collision-free trajectories, at each time step $k \in \mathbb{N}$, through V2V communications, vehicles share their planned trajectories over a finite horizon N: $\mathbf{r}_p^{(\nu)} \triangleq [r_{p,x}^{(\nu)}\ r_{p,y}^{(\nu)}]^\mathrm{T} \in \mathbb{R}^2$. Here, $\nu \in \frac{\mathcal{V}}{\{e\}}$ denotes the ν-th vehicle in communication. All the vehicles are permitted to implement the optimization procedure for planning concurrently at each time step.

The unified trajectory-planning computes the longitudinal force $F_{x,r}$ and δ_f the steering angle at the same time in which it generates its planned trajectories, while the hierarchical approach decomposes the control problem into a high-level trajectory-planning and lower-level trajectory-tracking problems. This lower-level layer is responsible to generate the actuators signals and track the trajectory $\mathbf{r}_p^{(e)}$ planned by the higher level trajectory-planning layer, in other words, the output of the higher level layer becomes the reference trajectory $\bar{\mathbf{r}}_t^{(e)}$ for the lower level tracking controller layer. The subscript t denotes the trajectories of the trajectory controller in the hierarchical unit.

Assumption 1. The route starts with the initial point; it continues to provide the information each time step over a prediction horizon, and it stops providing information at the final point.

Assumption 2. All of the network connections must be bidirectional. The MPC is a well-known method to handle the expected latencies, however, for simplification, we assume no latency in V2V communication.

Assumption 3. The vehicle state is estimated using the measurements from the on-board sensors and they can be used for feedback. A GPS-aided inertial navigation system is adequate to satisfy this assumption.

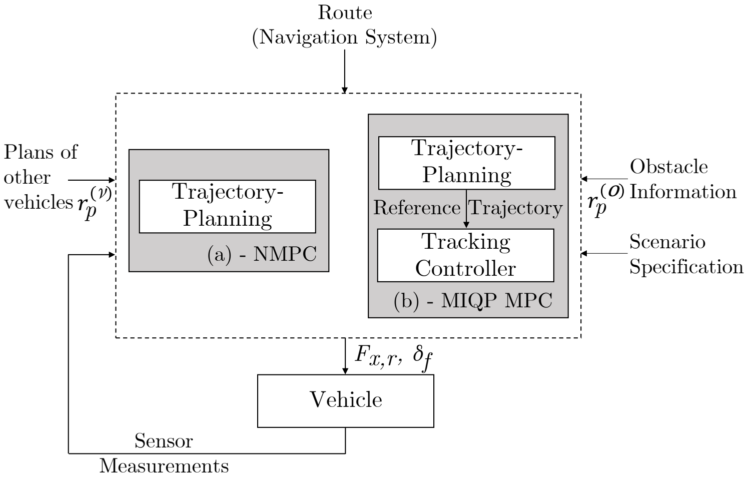


Fig. 1: Overview of the two cooperative planning and control frameworks: a) unified trajectory-planning using non-linear MPC; b) hierarchical unit with a mixed-integer quadratic programming trajectory-planning and a linear MPC for trajectory-tracking.

3.2 Bicycle Model

In inertial frame, the non-linear state-space representation of the bicycle dynamic model in continuous time [36] (see Fig. 2) is given in (1).

$$\dot{X} = f_c(X,U), \tag{1}$$

with,

$$\dot{X} = \begin{bmatrix} v_x \cos\psi - v_y \sin\psi \\ v_x \sin\psi + v_y \cos\psi \\ \omega \\ \frac{1}{m}(F_{x,r} - F_{y,f}\sin\delta_f + m v_y \omega) \\ \frac{1}{m}(F_{y,r} + F_{y,f}\cos\delta_f - m v_x \omega) \\ \frac{1}{\Theta}(F_{y,f} l_f \cos\delta_f - F_{y,r} l_r) \end{bmatrix}. \tag{2}$$

Let X denotes the vehicle state vector and U denotes the control input vector:

$$X \triangleq [r_x\ r_y\ \psi\ v_x\ v_y\ \omega]^{\mathrm{T}} \in \mathbb{R}^n,\ U \triangleq [F_{x,r}\ \delta_f]^{\mathrm{T}} \in \mathbb{R}^m. \tag{3}$$

In the vehicle model of (2), r_x and r_y denote the coordinates of the center of mass represented in the inertial frame, v_x and v_y are the vehicle velocities represented in the body-fixed coordinate frame. Moreover, ψ and ω denote the vehicle heading angle and its rate of change, respectively. l_r and l_f are the geometric parameters that describe the distance between the center of mass of the vehicle and the rear and front axles, respectively.

Assuming a rear-wheel drive, U, the control vector is composed by the steering angle δ_f of the front wheel and the longitudinal force $F_{x,r}$ at the rear wheel. $F_{y,f}$ and $F_{y,r}$ are the lateral tire forces acting at the front and rear wheels, respectively. A linear tire model is adopted by assuming small side-slip angles. The lateral tire forces $F_{y,i}$ can be approximated according to [37],

$$F_{y,i} = C_{\alpha_i}\alpha_i, \tag{4}$$

where $i \in \{f, r\}$ and C_{α_i} is the tire cornering stiffness.

The front and rear side-slip angles α_f and α_r are, respectively, given by

$$\alpha_f = \delta - \arctan\left(\frac{l_f\omega + v_y}{v_x}\right), \tag{5}$$

and,

$$\alpha_r = \arctan\left(\frac{l_r\omega - v_y}{v_x}\right). \tag{6}$$

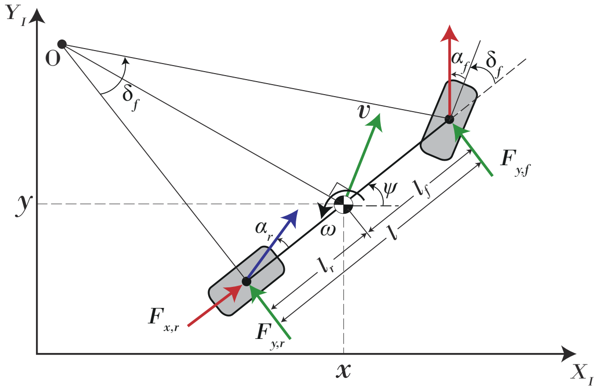


Fig. 2: fig2.eps

The continuous-time dynamics in (1) can be discretized using the Runge-Kutta method leading to a discrete-time dynamical system of the form,

$$X_{k+1} = f_k(X_k, U_k). \tag{7}$$

The model in (1)-(7) account for the non-linear vehicle dynamics and are used in the following NMPC trajectory-planner design.

In the hierarchical scheme, to design the lower-level layer MPC tracking controller, the vehicle dynamical model (2) is linearised around an equilibrium point (X_e, U_e):

$$\dot{X}_e + \Delta\dot{X} = f(X_e, U_e) + \frac{\partial f}{\partial X}\Delta X + \frac{\partial f}{\partial U}\Delta U. \tag{8}$$

The time-invariant linearised system matrices generated from the Jacobian matrix are,

$$A \triangleq \left.\frac{\partial f}{\partial X}\right|_{\substack{X=X_e \\ U=U_e}} \in \mathbb{R}^{n\times n},\ B \triangleq \left.\frac{\partial f}{\partial U}\right|_{\substack{X=X_e \\ U=U_e}} \in \mathbb{R}^{n\times m}. \tag{9}$$

3.3 CarSim Model

Regardless of vehicle model selected for the design of the NMPC trajectory-planner, to assess the performance of an architecture, the actuator commands are sent to a high fidelity model developed in CARSIM. This model emulates vehicle dynamics characteristics that can be encountered in real world with a 50+ degree of freedom simulation [38].

The vehicle model was built for a prototype Westfield Sports Car [39], the manufacturer of the vehicle used during trials carried out for the MuCCA project [40]. The mass of the vehicle m, and its yaw inertia, Θ center of gravity (CG) location of the vehicle as well as suspension and steering properties are obtained from measurements. The values for the parameters used to develop the 50+ degree of freedom model are found in [41].

The CARSIM model is then exported in SIMULINK and through the co-simulation interface, the performance of the designed cooperative trajectory-planning framework is assessed. In the co-simulation the steering angle of the front wheel δ_f and the longitudinal force $F_{x,r}$ are transformed to the control inputs of the CARSIM model: the engine torque T and the steering hand-wheel angle δ_{SW}. In our case, the engine torque T is equally distributed to both the rear wheels as the vehicle considered for test is a rear-wheel-driven car with an open differential of gear ratio $\tau = 1$. The steering ratio (SR), used to transform δ_f to δ_{SW}, is found to be equal to 13.

4 Unified Cooperative Trajectory-Planning Using NMPC

In this section, we present the unified trajectory-planner and controller architecture which generates the optimal trajectories as well as the control signals for the actuators directly, employing a non-linear model predictive controller NMPC that considers the non-linear bicycle dynamic model. The proposed method is designed for the cooperation of multiple vehicles.

4.1 Optimization Problem

In the design, soft constraints will be used to ensure multi-vehicle collision avoidance. Hence, at each time step k, the following objective function is to be minimized:

$$\begin{aligned}\min_{U,X} J(k) = &\sum_{j=1}^{N} [X_{k+j|k} - \bar{X}_{k+j|k}]^{\mathrm{T}} Q [X_{k+j|k} - \bar{X}_{k+j|k}] \\ &+ \sum_{j=1}^{M} [U_{k+j-1|k}]^{\mathrm{T}} R [U_{k+j-1|k}] + \sum_{j=1}^{N} J^{c}_{k+j|k} + \sum_{j=1}^{N} J^{o}_{k+j|k}\end{aligned}, \qquad (10)$$

subject to,

$$\begin{aligned} &X_{k+j|k} = f_k(X_{k+j-1|k}, U_{k+j-1|k}) \\ &U_{k+j-1|k} \in \mathcal{U} \\ &X_{k|k} = X_{t_k} \end{aligned}, \qquad (11)$$

where f_k is composed of (7). The index $k+j|k$ denotes a prediction of a quantity for time instant $k+j$, made at time instant k. The difference between the state sequence $X := \{X_{k+1|k}\ X_{k+2|k}\ \ldots\ X_{k+N|k}\}$ and the reference trajectory $\bar{X} := \{\bar{X}_{k+1|k}\ \bar{X}_{k+2|k}\ \ldots\ \bar{X}_{k+N|k}\}$ is obtained over a prediction horizon of length $N \in \mathbb{N}_{>0}$.

The matrices $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$ are the symmetrical state and control weighting matrices, respectively. Note that $Q \geq 0$ and $R > 0$. In this work, we take $Q = \eta \times \mathbf{I}_{6N}$ and $R = \rho \times \mathbf{I}_{2M}$.

The input sequence $U := \{U_{k|k}\ U_{k+1|k}\ \ldots\ U_{k+M-1|k}\}$ is the control input vector obtained along a control horizon of length M, where $U^{*}_{k|k}$ is the first control vector applied to the vehicle system at time instant k. At instant $k+1$, the optimization routine re-initiated for the problem over the shifted prediction horizon. Set $\mathcal{U}$ represents box constraints of the form $U = \{U_{\min} \leq U_{k+j-1|k} \leq U_{\max}\}$.

J^c and J^o are the collision cost and the obstacle avoidance cost, respectively. In the following, we describe in details these components of the cost function presented in (10).

4.2 Collision Avoidance Cost

A possible collision between the ego vehicle e and all others surrounding vehicles ν is avoided simply by adding a penalty term to cost function of NMPC in (10) at every time step k:

$$\text{for } j = \{1, \ldots, N_v\}: \quad J^{c}(X) = \sum_{j=1}^{N_v} \frac{\kappa_d}{1 + \exp^{\kappa_j}\left(\mathbf{d}_j(k) - r_{th,j}\right)}, \qquad (12)$$

where $\mathbf{d}_j(t)$ is the Euclidean distance between the ego-vehicle and the j-th surrounding vehicle given by $\mathbf{d}_j(k) = \|\mathbf{r}_p^{(e)}(k) - \mathbf{r}_p^{(\nu)}(k)\|_2$. κ_j is a positive constant defining the steepness of the sigmoid curve or the logistic growth rate. $r_{th,j}$ denotes the threshold distance between any vehicles, i.e. the safety region. Fig. 3 shows the logistic cost function for different κ parameters.

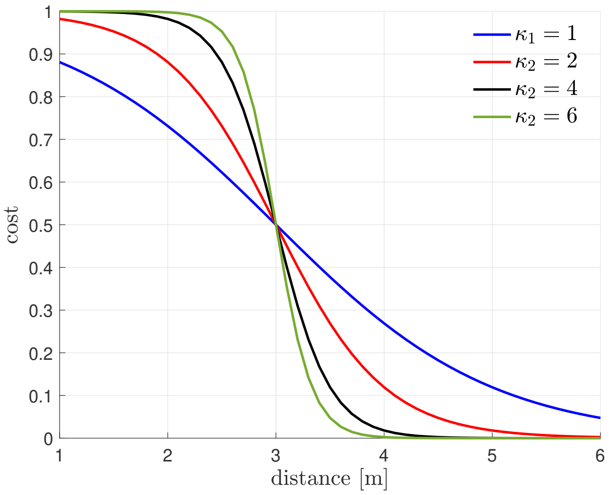


Fig. 3: Activation function for different steepness parameter κ.

Assumption 4. At each time instant k, the vehicle ν communicates its trajectory plans over a prediction horizon, using in the inter-vehicle communication network.

4.3 Obstacle Avoidance Cost

Similar to what is done for collision avoidance, the obstacle avoidance is achieved by adding a cost term to NMPC objective function. In (12), the planned trajectories $\mathbf{r}_p^{(\nu)}$ are simply replaced with the coordinates of the obstacles $O \in \mathbb{R}^2$ to generate the obstacle avoidance cost term:

$$\text{for } j = \{1, ..., N_o\} : \\ J^o(X) = \sum_{j=1}^{N_o} \frac{\kappa_d}{1 + \exp^{\kappa_j}\left(\mathbf{d}_j(k) - r_{th,j}\right)}, \qquad (13)$$

where $\mathbf{d}_j(k) = \|\mathbf{r}_p^{(e)}(k) - \mathbf{r}_p^{(o)}(k)\|_2$ denotes the Euclidean distance to the j-th obstacle to the vehicle.

Assumption 5. The positions of the obstacles on the road are known. Using on-board sensors such as cameras or light detection and ranging (LiDAR) systems with range at least 10 meters, it is possible to satisfy this assumption [42].

The cooperative trajectory-planning algorithm produces the optimal trajectories $\mathbf{r}_p^{(e)}(k)$ minimizing (10), using the planned trajectories of other vehicles $\mathbf{r}_p^{(\nu)}(k)$ through the cost term (12) and adding the obstacle coordinates $\mathbf{r}_p^{(o)}(k)$ as in (13). This non-linear optimization problem is tackled using the *fmincon* command in MATLAB, with the sequential quadratic programming algorithm. The distributed NPMC algorithm to solve this problem is summarized as follows:

Table 1: Algorithm for cooperative trajectory-planning using NMPC.

	Distributed NMPC
	Inputs: reference path $\bar{X}$, current states X, obstacle information $\mathbf{r}_p^{(o)}$ and plans of the surrounding vehicles $\mathbf{r}_p^{(\nu)}$ **Outputs:** the collision-free planned trajectories $\mathbf{r}_p^{(e)}$ and the control signals $F_{x,r}, \delta_f$
1:	Acquisition through sensors of the current states $X_{k\|k}$ and obstacle coordinates $\mathbf{r}_p^{(o)}(k)$.
2:	Obtain through the network the current planned trajectories of every surrounding vehicles $\mathbf{r}_p^{(\nu)}(k)$ over a prediction horizon N.
3:	Find the optimal solution $U^*_{k\|k}$ of the sub-problem (10) with non-linear optimisation to account for the non-linear dynamics (7).
4:	Apply the first control input $U^*_{k\|k}$ of the optimal control sequence to the ego vehicle e.
5:	Transmit the optimised planned trajectory $\mathbf{r}_p^{(e)}$(k) to all others connected vehicles ν.
6:	**repeat** (Steps 1-5) $k = k + 1$

5 Hierarchical Cooperative Trajectory-Planning Using MIQP

In this section, we present the cooperative trajectory-planning approach based on receding-horizon mixed-integer quadratic programming. This solution incorporates integer constraints into a linear MPC-based formulation. Once calculated the optimal solution, the planned trajectory is then converted to the reference trajectory $\bar{\mathbf{r}}_t^{(e)}$ for the tracking controller represented by another MPC. The proposed hierarchical scheme uses parallel optimization of both MPCs to improve global efficiency.

5.1 Point-Mass Model

Assumption 6. The curvature of the road is adequately small. Similar to what one encounters in highways, a straight road is considered.

According to Assumption 6, the trajectory can be planned easily with the following differential equation,

$$\dot{r}_x = v_x \ , \ \dot{v}_x = a_x, \ \dot{r}_y = v_y, \ \dot{v}_y = a_y. \qquad (14)$$

(14) describes the double integrator vehicle dynamics model. Accordingly, linear time-invariant state-space model in the discrete-time is taken into account,

$$\begin{aligned} X_{k+1} &= A_\mathrm{d} X_k + B_\mathrm{d} U_k \\ Y_k &= C_\mathrm{d} X_k \end{aligned}, \qquad (15)$$

where the state vector $X \triangleq [r_{p,x}\ v_{p,x}\ r_{p,y}\ v_{p,y}]^\mathrm{T} \in \mathbb{R}^4$ describes the position and velocity in both longitudinal direction x and lateral direction y of the road in the inertial frame. The control vector $U \triangleq [a_{p,x}\ a_{p,y}]^\mathrm{T} \in \mathbb{R}^2$ consists of the longitudinal and lateral acceleration.

5.2 Collision Avoidance Constraints

To achieve the collision avoidance among cooperative vehicles, consider any pair of a ego and surrounding vehicles, e and $\nu \in \mathcal{V}$. The collision-free planned trajectories can be ensured if, for every time step k, the coordinates $r_{p,x}^{(e)}$, $r_{p,y}^{(e)}$ of the ego vehicle are outside the bounding rectangles of all other vehicles ν. This can be described by the following set of logical constraints:

$$\forall k \in \{1, ..., t_f - 1\} : \\ |r_{p,x}^{(e)}(k) - r_{p,x}^{(\nu)}(k)| \geq l_\mathrm{safe} \qquad (16) \\ \text{OR } |r_{p,y}^{(e)}(k) - r_{p,y}^{(\nu)}(k)| \geq W,$$

with $l_{\text{safe}} = L + v^{(e)}_{p,x}(k)\Delta T$. This term includes a safety margin to the longitudinal dimension in dependency of the longitudinal speed of the ego-vehicle. The value of the parameter ΔT determines the length of the safety distance. Hereby, L and W are the longitudinal and lateral dimensions, respectively, that approximate the shape of a vehicle by a rectangle orientated along the x-axis.

The so-called Big-M method [43] is employed to transform the logical constraints (16) into a set of linear inequality non-convex constraints. This is achieved by introducing the binary variables $b^{(e,\nu)}_p(k) \in \{0,1\}$, yielding

$$
\begin{aligned}
&\forall k \in \{1,...,t_f-1\}:\\
&\qquad\qquad r^{(e)}_{p,x}(k) - r^{(\nu)}_{p,x}(k) \geq l_{\text{safe}} - Mb^{(e,\nu)}_1(k)\\
&\text{AND,}\ \ r^{(\nu)}_{p,x}(k) - r^{(e)}_{p,x}(k) \geq l_{\text{safe}} - Mb^{(e,\nu)}_2(k)\\
&\text{AND,}\ \ r^{(e)}_{p,y}(k) - r^{(\nu)}_{p,y}(k) \geq \ \ W - Mb^{(e,\nu)}_3(k)\\
&\text{AND,}\ \ r^{(\nu)}_{p,y}(k) - r^{(e)}_{p,y}(k) \geq \ \ W - Mb^{(e,\nu)}_4(k)\\
&\text{AND,}\quad \sum_{p=1}^{4} b^{(e,\nu)}_p(k) \leq 3,
\end{aligned}
\tag{17}
$$

where $M \gg 0$ is a large constant.

The binary variables $b^{(e,\nu)}_p(k)$ become the additional decision variables of the optimization problem. If $b^{(e,\nu)}_p(k) = 0$, this means the p-th constraint in (17) is activated. However, when $b^{(e,\nu)}_p(k) = 1$, this means that constraint is relaxed, because the constant M puts the upper bound outside of the solution space. The last constraint in (17) guarantees that at least one original OR-condition form of (16) is activated.

5.3 Obstacle Avoidance Constraints

Obstacle avoidance with static objects is modeled in the same way as shown in (17), introducing the binary variables $b^{(e,o)}_p(k)$ for each obstacle $o \in O$ and replacing the planned trajectories of the surrounding vehicle ν with the coordinates of the obstacles $r^{(o)}_{p,x}$, $r^{(o)}_{p,y}$. The minimal obstacle bounding rectangle is represented by $[r^{(o)}_{p,x} - L^{(o)}, r^{(o)}_{p,x} + L^{(o)}] \times [r^{(o)}_{p,y} - W^{(o)}, r^{(o)}_{p,y} + W^{(o)}]$.

This formulation can be also extended to include non-cooperative vehicles. A human driver model (HDM) integrated with the trajectory-planner is assumed to be capable to obtain a prediction motion of a human driver vehicle (HDV).

5.4 High-Level Trajectory-Planning

The high-level cooperative trajectory-planning acquires the optimal control vector $U^*_{k|k}$ by the minimization of the following quadratic cost function of the form,

$$
\begin{aligned}
\min_{U,X} J(k) = &\sum_{j=1}^{N} [X_{k+j|k} - \bar{X}_{k+j|k}]^{\mathrm{T}} Q [X_{k+j|k} - \bar{X}_{k+j|k}]\\
+ &\sum_{j=1}^{M} [U_{k+j-1|k}]^{\mathrm{T}} R [U_{k+j-1|k}]
\end{aligned}
\ \ ,
\tag{18}
$$

subject to,

$$
\begin{aligned}
X_{k|k} &= X_{t_k}\\
X_{k+j|k} &= A_{\mathrm{d}} X_{k+j-1|k} + B_{\mathrm{d}} U_{k+j-1|k},\\
U_{k+j-1|k} &\in \mathcal{U}
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
&\forall k \in \{1,...,t_f-1\}:\\
&\qquad\qquad \mathbf{r}_{\min} \leq \mathbf{r}^{(e)}_p(k) \leq \mathbf{r}_{\max}
\end{aligned}
\ ,
\tag{20}
$$

$$
\begin{aligned}
&\forall k \in \{1,...,t_f-1\}, \forall \nu \in \frac{\mathcal{V}}{\{e\}}:\\
&\qquad\qquad r^{(e)}_{p,x}(k) \leq r^{(\nu)}_{p,x}(k) - l_{\text{safe}} + Mb^{(e,\nu)}_1(k)\\
&\text{AND,}\ \ r^{(e)}_{p,x}(k) \geq r^{(\nu)}_{p,x}(k) + l_{\text{safe}} - Mb^{(e,\nu)}_2(k)\\
&\text{AND,}\ \ r^{(e)}_{p,y}(k) \leq r^{(\nu)}_{p,y}(k) - \ \ W + Mb^{(e,\nu)}_3(k)\\
&\text{AND,}\ \ r^{(e)}_{p,y}(k) \geq r^{(\nu)}_{p,y}(k) + \ \ W - Mb^{(e,\nu)}_4(k)\\
&\text{AND,}\quad \sum_{p=1}^{4} b^{(e,\nu)}_p(k) \leq 3
\end{aligned}
\ ,
\tag{21}
$$

$$
\begin{aligned}
&\forall k \in \{1,...,t_f-1\}, \forall o \in O:\\
&\qquad\qquad r^{(e)}_{p,x}(k) \leq r^{(o)}_{p,x}(k) - L^{(o)} + Mb^{(e,o)}_1(k)\\
&\text{AND,}\ \ r^{(e)}_{p,x}(k) \geq r^{(o)}_{p,x}(k) + L^{(o)} - Mb^{(e,o)}_2(k)\\
&\text{AND,}\ \ r^{(e)}_{p,y}(k) \leq r^{(o)}_{p,y}(k) - W^{(o)} + Mb^{(e,o)}_3(k).\\
&\text{AND,}\ \ r^{(e)}_{p,y}(k) \geq r^{(o)}_{p,y}(k) + W^{(o)} - Mb^{(e,o)}_4(k)\\
&\text{AND,}\quad \sum_{p=1}^{4} b^{(e,o)}_p(k) \leq 3
\end{aligned}
\tag{22}
$$

The linear convex inequality constraints in (20) are employed to model the straight road. The vectors $\mathbf{r}_{\min}$, $\mathbf{r}_{\max} \in \mathbb{R}^2$ contain boundaries of the road along each direction. This optimization problem in MIQP formulation is solved with the CPLEX IBM package [44].

5.5 Low-Level Trajectory-Controller

The linearized model in Section 3.2 is used in the design of the MPC in the lower layer of the hierarchy, namely in the tracking controller design. The linearization is necessary to satisfy the computational requirements of the real-time applications.

Control Constraints

The control constraints written in terms of U are incorporated in the scheme as given in (23):

$$\mathcal{U}_{\min} \leq U \leq \mathcal{U}_{\max}, \tag{23}$$

where $\mathcal{U}_{\min}$ and $\mathcal{U}_{\max}$ are the upper and lower boundary limits on the control commands as in (24):

$$\mathcal{U}_{\min} = [\underline{F}_{x,r}\ \underline{\delta}_f]^{\mathrm{T}},\ \mathcal{U}_{\max} = [\overline{F}_{x,r}\ \overline{\delta}_f]^{\mathrm{T}}. \tag{24}$$

MPC for Tracking

The optimal control vector $U^*_{k|k}$ in the tracking control layer is acquired with the same approach in (18), but this time without the mixed-integer constraints. Additional control constraints are incorporated in the optimization problem formulation in the form given in (25):

$$\min_{U,X} J(k) = \sum_{j=1}^{N} \|Y_{k+j|k} - \bar{Y}_{k+j|k}\|_{\mathrm{Q}} + \sum_{j=1}^{M} \|[U_{k+j-1|k}\|_{\mathrm{R}} \quad , \tag{25}$$

subject to (23).

In this case the reference is given by the vector $\bar{Y} \triangleq [\bar{r}^{(e)}_{t,x}, \bar{r}^{(e)}_{t,y}]^T$ obtained from the high-level trajectory-planning as shown in Fig. 1. This information is sufficient for the tracking purpose. The prediction model uses the control vector $U \in \mathbb{R}^2$ and the state vector $X \in \mathbb{R}^6$ defined in (3).

6 Computational Evaluation

In this section, the introduced cooperative trajectory-planning frameworks are applied to demonstrate the feasibility in general on-road scenarios. A challenging traffic scenario with two vehicles moving along a straight road with one obstacle is presented. The name of the autonomous vehicles here are MuCCA Electric Vehicles (MEVs) [45]. The MEVs (vehicles 1 and 2) have a driving speed of 36 km/h (10 m/s). The rectangular obstacle on the road is centered at $O = [20\ 4]^{\mathrm{T}}$ m. The problem of cooperative trajectory-planning is solved employing the techniques described in Sections 4 and 5.

The solver utilized in the simulations is Runge-Kutta 4 with the integration step of 0.001 s. Table I presents the parameters of the trajectory-planners. The tracking controller on the lower-level layer in the hierarchical approach runs at a high frequency with a sampling time of T = 0.01 s. A relatively large prediction horizon N = 20 and small control horizon M = 5 produces less aggressive control, and in turn a smoother vehicle motion. The actuator constraints are set according to the saturation limits. MATLAB/Simulink is the medium selected for the implementation of the proposed algorithms, and the experiments are conducted on a 16GB RAM, Intel Core i7-6700U CPU computer clocked at 3.40GHz.

Table 2: Trajectory-Planner Optimisation Parameters

Variables	Values
Sampling time	$T = 0.05$ s
Prediction horizon	$N = 20$
Control horizon	$M = 5$
Control constraints:	$\underline{\delta}_{\mathrm{sw}} = -630°$, $\underline{T} = 0$ N m
	$\overline{\delta}_{\mathrm{sw}} = 630°$, $\overline{T} = 400$ N m
(a) NMPC	
Controlled output weights	$\eta = [1\ 1\ 1\ 1\ 1\ 1]^{\mathrm{T}}$
Control input weights	$\rho = [0.1\ 0.1]^{\mathrm{T}}$
Tuning parameter	$\kappa_d = 1$
Parameter for smoothness	$\kappa_j = 2$
Threshold distance	$r_{th} = 2$ m
(b) MIQP MPC	
Controlled output weights	$\eta = [1\ 1\ 1\ 1]^{\mathrm{T}}$
Control input weights	$\rho = [20\ 20]^{\mathrm{T}}$
Longitudinal dimension	$L = 2.5$ m
Lateral dimension	$W = 2$ m
Safety parameter	$\Delta T = 0.5$

To asses the performances of the proposed two cooperative collision avoidance schemes, both algorithms were tested in a co-simulation environment including CARSIM 2016.1 [46]. The car model in CARSIM follows the vehicle parameters tabulated in Table II.

Table 3: Parameters of the Vehicle

Variables	Values
Mass, m	950 kg
Yaw inertia, Θ	1200 kg m^2
Wheelbase, l	2.5 m
Distance from front axle to CG, l_f	1 m
Distance from rear axle to CG, l_r	1.5 m
Cornering stiffness, c_f,c_r	36000 N m/rad
Steering ratio, SR	13 N m/rad
Wheel radius, R	325 mm

Fig. 4 illustrates the resulting planned trajectories along with the actual vehicle trajectory. The initial positions of the vehicles are $\mathbf{r}^{(1)} = [0\ 0]^{\mathrm{T}}$ m, $\mathbf{r}^{(2)} = [0\ 4]^{\mathrm{T}}$ m, and they move along a horizontal straight road. Both Fig.4(a) and Fig.4(b) demonstrates that the MEV-2 steers around the obstacle successfully, and returns to its original path. In accordance with the cooperative trajectory planning, the MEV-1 pre-emptively veers to allow a safe space for the MEV-2 during its obstacle avoidance maneuver.

The unified trajectory-planning approach generates an evasive trajectory in advance as can be seen in Fig.4(a). MEV-1 senses the intention of MEV-2 to overtake the obstacle, and it re-plans its trajectory accordingly. One drawback of this approach is that it cannot guarantee a collision avoidance at every time step along the prediction horizon. It is seen at beginning of the avoidance maneuver one sample of the MEV-2 planned trajectory crossing the obstacle. By employing the proposed cooperative MIQP approach Fig.4(b) yields a more precise double lane change maneuver. We observe that before MEV-2 passing the obstacle, the trajectory-planner re-plans its trajectory avoiding the obstacle by turning right and obeying the longitudinal safety margin. After passing the obstacle, the planned-trajectories converge to the reference. Fig. 5 shows the vehicle position as it overtakes the obstacle in three lane scenario. The results are obtained in the co-simulation framework of MATLAB-CARSIM.

Fig. 6 shows steering hand-wheel angle profiles for both approaches. In general, the MPC follows a smooth driving behaviour while it seers vehicle to complete the obstacle avoidance maneuver. First, the steering inputs turn right for collision avoidance, thereafter they turn left to safely return the vehicle to its original lane which is the determined reference trajectory. Looking at Fig. 6(a-b) one can see that using NMPC the vehicles have to take more corrective action to return back to their original lane. The MIQP approach as demonstrated in Fig. 6(c-d) outperforms with a smoother steering action and therefore increases the passenger comfort.

Fig. 7 shows the total rear wheel torque as input signal generated by the CarSim. Initially both vehicles have a torque of approximately 17 N-m, this value is required to maintain a constant speed of 36 km/h. MEV-2 decelerates for a period as it sees the obstacle first as a result the torque decreases. Once it overtakes the obstacle by turning towards the right, the torque increases again to avoid the obstacle quickly. This allows MEV-2 performs the maneuver without a harsh braking. The MEV-1 on other hand just turns to the right and its torque increases to avoid colliding MEV-2.

To evaluate the performance in a scenario at highway speeds, we have simulated the hierarchical scheme for MEV-2 with a speed of 80 km/h, carrying out the obstacle avoidance maneuver. The obstacle is located now at $O = [150\ 4]^{\mathrm{T}}$ m. The results of the tracking controller are shown in Fig. 8. It is noted that the vehicle is able to follow sufficiently well the reference trajectory $\bar{\mathbf{r}}_t$, however its performance is inferior due the presence of actuator delay. The enlarged display shows the profile of the longitudinal velocity v_x.

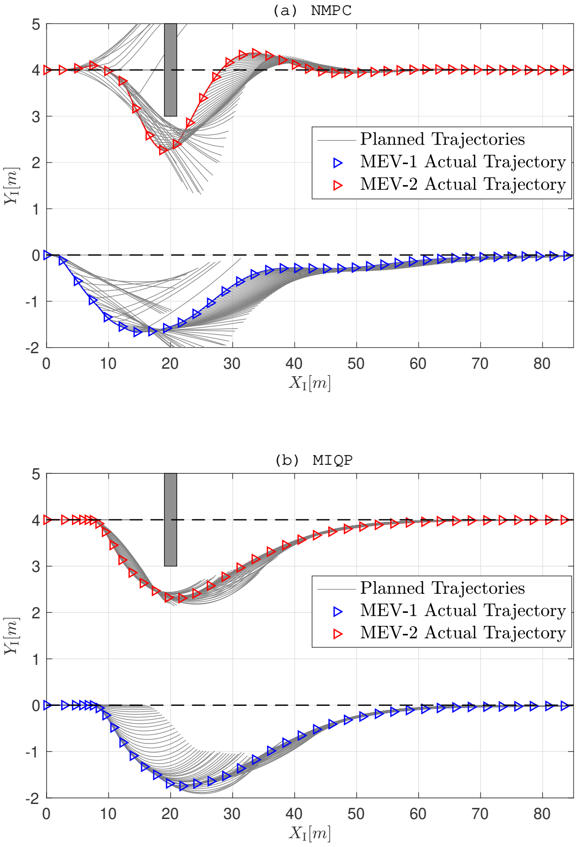


Fig. 4: The trajectories of the vehicles generated by the trajectory-planner during the collision avoidance for a straight reference (dashed line) and obstacles: (**a**) NMPC trajectory-planning; (**b**) MIQP trajectory-planning.

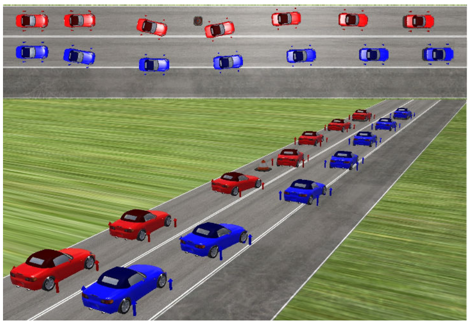


Fig. 5: The animation interface of CARSIM showing maneuver to avoid collision.

Computing Times

The computation times to obtain the control signal for the different algorithms of the performed simulations are compared in Fig. 9. Looking for the MEV-1 Fig.9a, it is observed that the time of getting the control signal using NMPC is around 50 ms in 79% of the iterations, while for the MIQP approach this value reaches up to 175 ms for the majority of the samples. Regarding the MEV-2 Fig.9b,

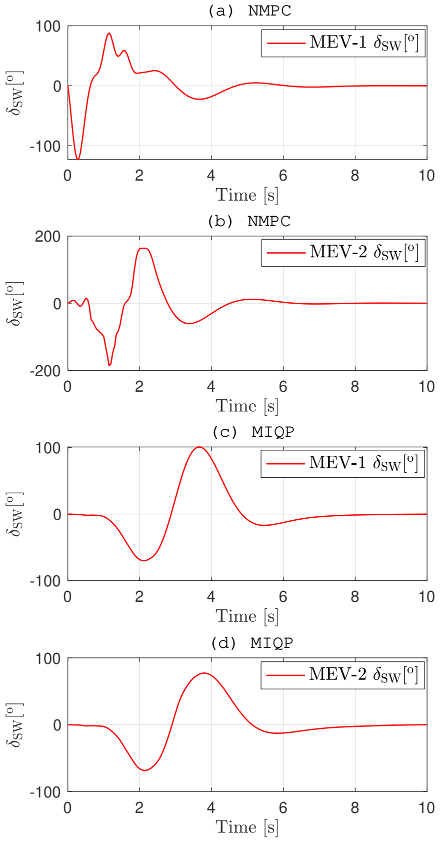


Fig. 6: Profiles of the steering hand-wheel angle.

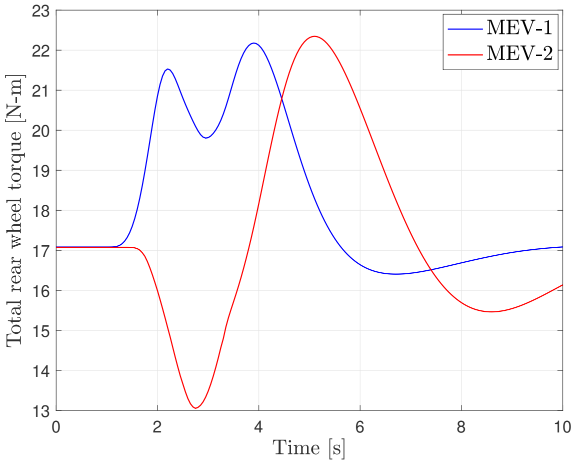


Fig. 7: Total rear wheel torque.

the average computation time with the NMPC framework is 50 ms, while for the MIQP scheme it is 190 ms. Notice that the MIQP trajectory-planner presents in the worst case 700 ms, as the computational load increases drastically when the vehicles approach the obstacles due to the activation of the binary variables. For the NMPC framework, the

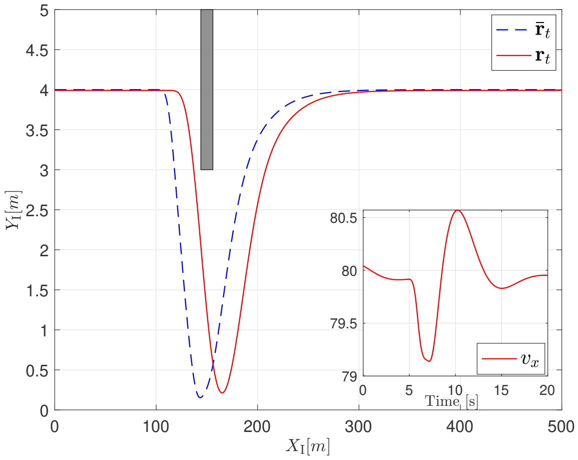


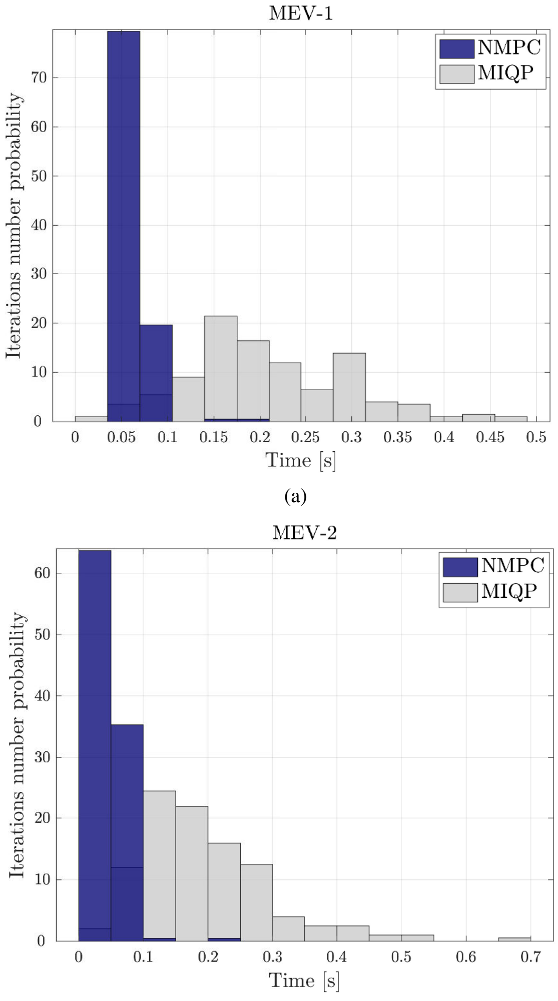
Fig. 8: Performance evaluation of the tracking controller at high speed (80 km/h).

computation time in the worst case is 200 ms, that occurs when the controller decides to relax the soft constraints to improve feasibility. Overall the computation burden during the double lane change manoeuvre execution is lighter for the unified NMPC approach when compared to the MIQP trajectory-planner, which include collision avoidance as hard constraints. For the latter approach, the time to get a solution becomes prohibitive for real-time applications in complex scenarios where a bigger ensemble of cooperative vehicles are employed.

7 Conclusions

This paper evaluated two different trajectory-planning algorithms on a cooperative collision avoidance scenario. To assess the performances of the proposed methods, numerical simulations were performed using a high fidelity non-linear vehicle model in CARSIM to attain the on-road autonomous driving environment. The unified approach based on non-linear MPC have the potential for real-time application and it is shown to be a practical solution in the overtaking scenario. While the hierarchical approach using MIQP allows a high-precision planning, however, its computational cost is high as well because of the binary variables added to the decision variables of the optimisation problem. It is considered that timing behavior of MIQP is prohibitive for real-time use. Both frameworks account for the trajectories of other cooperating vehicles planned over a finite horizon to satisfy the collision avoidance requirement.

For a future work, the human driver vehicles (HDVs) will be incorporated in the traffic scenario. This will allow to test the cooperative trajectory-planning algorithms in the experiments including both human-driven and autonomous vehicles. An HDV prediction model will be integrated to assist the trajectory-planner; and therefore it will be possible for the human-driven vehicle to influence the behaviour of the autonomous vehicles. Another improvement of the



(a)

(b)

Fig. 9: Running times to compute the control signal in each iteration: (**a**) MEV-1; (**b**) MEV-2.

work will focus in handling the conflict resolution problem that emerges when we consider the trajectory-planning of multiple vehicles. This work is also planned to be implemented and tested on real vehicles according to the scope of the MuCCA project founded by Innovate UK [45].

Acknowledgements

The authors thanks for the financial support of Innovate UK under the project grant MuCCA. The consortium of the project is comprised of Cosworth Electronics, Applus IDIADA, Westfield Sportscars, Connected Places Catapult and Cranfield University.

References

[1] Geiger, A., Lenz, P., and Urtasun, R., 2012. "Are we ready for autonomous driving? the kitti vision benchmark suite". In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361.

[2] Yuan, H., Sun, X., and Gordon, T., 2018. "Unified decision-making and control for highway collision avoidance using active front steer and individual wheel torque control". *Vehicle System Dynamics*, pp. 1–18.

[3] Kelly, A., and Nagy, B., 2003. "Reactive nonholonomic trajectory generation via parametric optimal control". *Int J Rob Res.,* **22**(78), pp. 583–601.

[4] Hassanzadeh, M., Lidberg, M., Keshavarz, M., and L., B., 2012. "Path and speed control of a heavy vehicle for collision avoidance manoeuvres". In 2012 IEEE Intelligent Vehicles Symposium (IV), IEEE.

[5] Sichitiu, M. L., and Kihl, M., 2008. "Inter-vehicle communication systems: A survey". *IEEE Communications Surveys Tutorials,* **10**(2), p. 88105.

[6] Schouwenaars, T., Demoor, B., Feron, E., and How, J., 2001. "Mixed integer programming for multi-vehicle path planning". In Proceedings..., European Control Conference, pp. 2603–2608.

[7] Frese, C., and Beyerer, J., 2011. "A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles". In 2011 IEEE Intelligent Vehicles Symposium (IV), pp. 1156–1162.

[8] Wang, D., Hu, M., Wang, Y., Wang, J., Qin, H., and Bian, Y., 2016. "Model predictive controlbased cooperative lane change strategy for improving traffic flow". *Advances in Mechanical Engineering,* **8**(2), p. 1687814016632992.

[9] Yuan, H., Sun, X., and Gordon, T., 2018. "Unified decisionmaking and control for highway collision avoidance using active front steer and individual wheel torque control, vehicle system dynamics". *Vehicle System Dynamics*, pp. 1744–5159.

[10] Viana, . B., and Aouf, N., 2018. "Distributed cooperative path-planning for autonomous vehicles integrating human driver trajectories". In 2018 International Conference on Intelligent Systems (IS), pp. 655–661.

[11] Viana, . B., Kanchwala, H., and Aouf, N., 2019. "Cooperative trajectory planning for autonomous driving using nonlinear model predictive control". In 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE), pp. 1–6.

[12] Zeilinger, M. N., Morari, M., and Jones, C. N., 2014. "Soft constrained model predictive control with robust stability guarantees". *IEEE Transactions on Automatic Control,* **59**(5), pp. 1190–1202.

[13] Kuriki, Y., and Namerikawa, T., 2015. "Formation control with collision avoidance for a multi-uav system using decentralized mpc and consensus-based control". *SICE Journal of Control, Measurement, and System Integration,* **8**, pp. 285–294.

[14] Ji, J., Khajepour, A., Melek, W. W., and Huang, Y., 2017. "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints". *IEEE Transactions on Vehicular Technology,* **66**(2), pp. 952–964.

[15] Shibata, N., Sugiyama, S., and Wada, T., 2014. "Collision avoidance control with steering using velocity po-

tential field". In IEEE Intelligent Vehicles Symposium Proceedings, pp. 438–443.

[16] You, F., Zhang, R., Lie, G., Wang, H., Wen, H., and Xu, J., 2015. "Trajectory planning and tracking control for autonomous lane change maneuver based on the cooperative vehicle infrastructure system". *Expert Systems with Applications,* **42**(14), pp. 5932 – 5946.

[17] Nilsson, J., and Sjberg, J., 2013. "Strategic decision making for automated driving on two-lane, one way roads using model predictive control". In IEEE Intelligent Vehicles Symposium (IV), pp. 1253–1258.

[18] Schildbach, G., and Borrelli, F., 2015. "Scenario model predictive control for lane change assistance on highways". In IEEE Intelligent Vehicles Symposium (IV), pp. 611–616.

[19] Quinlan, S., and Khatib, O., 1993. "Elastic bands: connecting path planning and control". In IEEE International Conference on Robotics and Automation, Vol. 2, pp. 802–807.

[20] Hilgert, J., Hirsch, K., Bertram, T., and Hiller, M., 2003. "Emergency path planning for autonomous vehicles using elastic band theory". In IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003), Vol. 2, pp. 1390–1395.

[21] Gehrig, S. K., and Stein, F. J., 2007. "Collision avoidance for vehicle-following systems". *IEEE Transactions on Intelligent Transportation Systems,* **8**(2), pp. 233–244.

[22] Kala, R., and Warwick, K., 2013. "Planning autonomous vehicles in the absence of speed lanes using an elastic strip". *IEEE Transactions on Intelligent Transportation Systems,* **14**(4), pp. 1743–1752.

[23] Lenz, D., Kessler, T., and Knoll, A., 2016. "Tactical cooperative planning for autonomous highway driving using monte-carlo tree search". In IEEE Intelligent Vehicles Symposium (IV), pp. 447–453.

[24] Kurzer, K., Engelhorn, F., and Zllner, J. M., 2018. "Decentralized cooperative planning for automated vehicles with continuous monte carlo tree search". In 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 452–459.

[25] Venkatraman, A., and Bhat, S. P., 2006. "Optimal planar turns under acceleration constraints". In 45th IEEE Conference on Decision and Control, pp. 235–240.

[26] Shamir, T., 2004. "How should an autonomous vehicle overtake a slower moving vehicle: design and analysis of an optimal trajectory". *IEEE Transactions on Automatic Control,* **49**(4), pp. 607–610.

[27] Anisi, D. A., Hamberg, J., and Hu, X., 2003. "Nearly time-optimal paths for a ground vehicle". *Journal of Control Theory and Applications,* **1**(1), pp. 2–8.

[28] Dingle, P., and Guzzella, L., 2010. "Optimal emergency maneuvers on highways for passenger vehicles with two- and four-wheel active steering". In American Control Conference, pp. 5374–5381.

[29] Tomas-Gabarron, J.-B., Egea-Lopez, E., and Garcia-Haro, J., 2011. "Evaluating communications and idm in a context of chain cca application for vanets". In 3rd International Conference on Road Safety and SimulationPurdue UniversityTransportation Research Board.

[30] Tomas-Gabarron, J., Egea-Lopez, E., and Garcia-Haro, J., 2013. "Vehicular trajectory optimization for cooperative collision avoidance at high speeds". *IEEE Transactions on Intelligent Transportation Systems,* **14**(4), pp. 1930–1941.

[31] Eilbrecht, J., and Stursberg, O., 2019. "Reducing computation times for planning of reference trajectories in cooperative autonomous driving". In IEEE Intelligent Vehicles Symposium (IV), pp. 146–152.

[32] Burger, C., and Lauer, M., 2018. "Cooperative multiple vehicle trajectory planning using miqp". In 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 602–607.

[33] Salvado, J., Krug, R., Mansouri, M., and Pecora, F., 2018. "Motion planning and goal assignment for robot fleets using trajectory optimization". In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7939–7946.

[34] Branca, C., and Fierro, R., 2006. "A hierarchical optimization algorithm for cooperative vehicle networks". In American Control Conference, pp. 4225–4230.

[35] Miller, C., Pek, C., and Althoff, M., 2018. "Efficient mixed-integer programming for longitudinal and lateral motion planning of autonomous vehicles". In IEEE Intelligent Vehicles Symposium (IV), pp. 1954–1961.

[36] Pacejka, H., 2005. *Tire and Vehicle Dynamics*. Elsevier.

[37] Rajamani, R., 2012. *Vehicle Dynamics and Control*. Springer.

[38] Zhang, K., Sprinkle, J., and Sanfelice, R. G., 2015. "A hybrid model predictive controller for path planning and path following". In Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems, ICCPS '15, ACM, pp. 139–148.

[39] Westfield. *Sports cars*.

[40] MuCCA. *Multi-Car Collision Avoidance project*.

[41] H. Kanchwala, I. B. Viana, N. A., 2020. "Cooperative path-planning and tracking controller evaluation using vehicle models of varying complexities". *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*.

[42] Eckelmann, S., Trautmann, T., Ubler, H., Reichelt, B., and Michler, O., 2017. "V2v-communication, lidar system and positioning sensors for future fusion algorithms in connected vehicles". *Transportation Research Procedia,* **27**, pp. 69–76.

[43] Williams, H. P., and Brailsford, S. C., 1996. *Computational Logic and Integer Programming*, j. e. beasley ed. Oxford University Press, Inc.

[44] IBM, 2009. *IBM ILOG CPLEX V12.1: user's Manual for CPLEX*.

[45] Wartnaby, C. E., Nam, D., and Viana, I. B., 2018. "Multi-car collision avoidance". In ITS World Congress.

[46] Corporation, M. S., 2016. *CarSim user reference manual*.

List of Tables

List of Figures