Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Manoeuvre generation and control for automated highway driving

Julia Nilsson, [*] **Yiqi Gao,** [**] **Ashwin Carvalho,** [**]
Francesco Borrelli [**]

[*] *Active Safety and Chassis, Volvo Car Corporation, Sweden.*
Signals and Systems, Chalmers University of Technology, Sweden.
(e-mail: julia.nilsson@volvocars.com)
[**] *Mechanical Engineering, University of California, Berkeley, USA.*

Abstract: A hierarchical, two-level architecture for manoeuvre generation and vehicle control for automated highway driving is presented. The high-level planner computes a manoeuvre in terms of a (X,Y)-trajectory as well as a longitudinal velocity profile, utilizing a simplified point-mass model and linear collision avoidance constraints. The low-level controller utilizes a non-linear vehicle model in order to compute the vehicle control inputs required to execute the planned manoeuvre. Both the high-level planner and low-level controller are formulated based on the model predictive control methodology. Simulation results demonstrates the ability of the high-level planner to compute appropriate, traffic-dependent manoeuvres, that can be tracked by the low-level controller in real-time.

1. INTRODUCTION

Within the automotive industry, automated functionality is ever increasing and advanced driver assistance systems (ADAS) such as adaptive cruise control (ACC), lane keeping aid (LKA) [Pohl et al., 2007], and parking assistance systems [Pohl et al., 2006] are now standard in many production vehicles. These systems aid the driver in difficult and tedious tasks in order to increase safety and comfort. It is expected that an even higher level of automation will provide further benefits, especially in relation to traffic safety by reducing the impact of human factors [Neale et al., 2005].

An area where a high level of autonomy is both desirable and feasible is highway driving. Highways are structured environments with relatively simple and easily maintainable traffic rules. As such, the driving task is quite straightforward, i.e. maintain a desired velocity while avoiding collisions with surrounding vehicles and respecting the rules of the road. However, due to the high velocity and intense traffic flow, violating safety constraints can have severe consequences.

Because of its ability to systematically handle system constraints and non-linearities, model predictive control (MPC) is an attractive choice for control design of ADAS for highway driving. In MPC, at every time instance a finite-time constrained optimal control problem is solved in receding horizon, i.e. every time instance the optimal control problem is reformulated and solved based on the current state, over a shifted time horizon [Mayne et al., 2000]. Non-linear MPC has been used for combined steering and braking control [Falcone et al., 2008] as well as for obstacle avoidance manoeuvres [Gao et al., 2012], [Gray et al., 2012]. However, non-linear MPC requires solving complex non-linear optimizations problems which normally results in high computational complexity. This drawback can render non-linear MPC unsuitable for applications with limited computing resources and real-time demands [Borrelli et al., 2005].

In this paper a two-level hierarchical control scheme, similar to the approach presented in [Gao et al., 2012], [Gray et al., 2012] is employed. The high-level planner computes a collision free manoeuvre in terms of a (X,Y)-trajectory and a longitudinal velocity profile which are fed to a low-level controller which computes the required vehicle inputs to track the planned manoeuvre. In the paper we try to overcome the problem of prohibitive computational complexity in the high-level planner resulting from collision avoidance constraints formulated as mixed-integer inequalities [Borrelli et al., 2006] or non-linearities in the vehicle model. To this aim the structured environment of highways is exploited in order to formulate linear collision avoidance constraints. The high-level MPC planning problem can thus be written as an equivalent quadratic program (QP), utilizing a simplified point-mass model to represent the vehicle, allowing the planner to optimize lateral and longitudinal control simultaneously. The resulting manoeuvre is processed by a low-level controller which tracks the planned (X,Y)-trajectory and longitudinal velocity profile using a four wheel vehicle model and computes the required vehicle inputs as the solution of a low complexity non-linear MPC problem. The paper thus presents a hierarchical, two-level architecture for real-time manoeuvre generation and vehicle control for automated highway driving.

The paper is structured as follows: in Section II the four wheel vehicle model is presented along with the simplified point-mass vehicle model and the linear collision avoidance constraints. The predictive control problem of the high-level planner and low-level controller is introduced in Section III, and simulation results are shown in Section IV. Finally, conclusions and directions for future work is given in Section V.

2. MODELLING

2.1 Four wheel vehicle model

The low-level controller uses the six state non-linear vehicle model, presented in [Falcone et al., 2008]. The model captures the main lateral and longitudinal dynamics and computes the tire forces as a function of the road friction coefficient, and longitudinal and lateral wheel slip angles, using a non-linear Pacejka tire model [Bakker et al., 1987]. By considering Fig. 1, the following set of differential equations to describe the vehicle dynamics is derived using the equations of motion about the vehicle's center of gravity (CoG) and coordinate transformation between the inertial- and vehicle body-frame,

$$m\ddot{x} = m\dot{y}\dot{\psi} + \sum_{i=1}^{4} F_{x_i}, \tag{1a}$$

$$m\ddot{y} = -m\dot{x}\dot{\psi} + \sum_{i=1}^{4} F_{y_i}, \tag{1b}$$

$$J_z\ddot{\psi} = d_f(F_{y_1} + F_{y_2}) - d_r(F_{y_3} + F_{y_4}) + \frac{w_t}{2}(-F_{x_1} + F_{x_2} - F_{x_3} + F_{x_4}), \tag{1c}$$

$$\dot{X} = \dot{x}\cos\psi - \dot{y}\sin\psi, \tag{1d}$$

$$\dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi, \tag{1e}$$

where m and J_z denote the vehicle mass and yaw inertia respectively, d_f and d_r denote the distances from the vehicle CoG to the front and rear axles respectively, and w_t denotes the track width. $\dot{x}$ and $\dot{y}$ denote the vehicle's longitudinal and lateral velocities in the body frame respectively, and $\dot{\psi}$, X, and Y denote the yaw rate, longitudinal, and lateral vehicle coordinates in the inertial frame respectively. F_{y_i} and F_{x_i} are tire forces acting along the vehicle's lateral and longitudinal axis, respectively.

The vehicle dynamics can be compactly written as

$$\dot{\xi}^{4w}(t) = f(\xi^{4w}(t), u^{4w}(t)), \tag{2}$$

where $\xi^{4w} = \left[\dot{x}, \dot{y}, \psi, \dot{\psi}, X, Y\right]^T$ and $u^{4w} = [\delta, F_L, F_R]^T$, δ is the front steering angle, and F_L, F_R denote the left and right braking/acceleration forces respectively. Note that it is assumed that only the steering angle at the front wheels can be controlled and the steering angles at the right and left wheels of each axle are the same.

F_{y_i} and F_{x_i} are computed as

$$F_{y_i} = f_{x_i}\sin\delta + f_{y_i}\cos\delta, \quad i \in \{1,2,3,4\}, \tag{3a}$$

$$\sum F_{x_i} = F_L, \quad i \in \{1,3\}, \tag{3b}$$

$$\sum F_{x_i} = F_R, \quad i \in \{2,4\}, \tag{3c}$$

where f_{x_i} and f_{y_i} respectively denote the longitudinal and lateral tire forces along the tire axis

$$f_{x_i} = f_l(\alpha_i, s_i, \mu, F_{z_i}), \quad i \in \{1,2,3,4\}, \tag{4a}$$

$$f_{y_i} = f_c(\alpha_i, s_i, \mu, F_{z_i}), \quad i \in \{1,2,3,4\}, \tag{4b}$$

where f_l and f_c are non-linear functions described by a Pacejka tire model [Bakker et al., 1987]. In Eq. (4), α is the tire slip angle defined as

$$\alpha_i = \arctan\frac{v_{c_i}}{v_{l_i}}, \quad i \in \{1,2,3,4\}, \tag{5}$$

where v_c and v_l respectively are the lateral (cornering) and longitudinal wheel velocities, computed as

$$v_{c_i} = v_{y_i}\cos\delta - v_{x_i}\sin\delta, \quad i \in \{1,2,3,4\}, \tag{6a}$$

$$v_{l_i} = v_{y_i}\sin\delta + v_{x_i}\cos\delta, \quad i \in \{1,2,3,4\}, \tag{6b}$$

where

$$v_{y_i} = \dot{y} + d_f\dot{\psi}, \quad i \in \{1,2\}, \tag{7a}$$

$$v_{y_i} = \dot{y} - d_r\dot{\psi}, \quad i \in \{3,4\}, \tag{7b}$$

$$v_{x_i} = \dot{x} - \frac{w_t}{2}\dot{\psi}, \quad i \in \{1,3\}, \tag{7c}$$

$$v_{x_i} = \dot{x} + \frac{w_t}{2}\dot{\psi}, \quad i \in \{2,4\}. \tag{7d}$$

The slip ratio, s, is defined as

$$s_i = \begin{cases} \dfrac{r\omega_i}{v_{l_i}} - 1 \text{ if } v_{l_i} > r\omega_i,\ v \neq 0 \text{ for deceleration,} \\ 1 - \dfrac{v_{l_i}}{r\omega_i} \text{ if } v_{l_i} < r\omega_i,\ \omega \neq 0 \text{ for acceleration,} \end{cases} \quad i \in \{1,2,3,4\}, \tag{8}$$

where r and ω are the wheel radius and angular speed respectively. However, as the wheel speed is not captured by the presented model it is assumed to be measured at each sampling time and is kept constant until next available update. Likewise, the friction coefficient μ is assumed to be a known constant. Further, a constant normal tyre load, F_z, is assumed distributed between the front and rear wheels as following

$$F_{z_i} = \frac{d_r mg}{2(d_f + d_r)}, \quad i \in \{1,2\}, \tag{9a}$$

$$F_{z_i} = \frac{d_f mg}{2(d_f + d_r)}, \quad i \in \{3,4\}, \tag{9b}$$

where g is the gravitational acceleration.

2.2 Point-mass vehicle model

Considering the scenario sketched in Fig. 2, the following set of equations is used to model the motion of the ego vehicle, E, with respect to the surrounding vehicles S_j, $(j = 1, \ldots, q)$, and the road boundaries, in a road aligned coordinate frame,

$$\triangle\dot{x}_j = v_{s_j} - v_x, \tag{10a}$$

$$\dot{y} = v_y, \tag{10b}$$

$$\dot{v}_x = a_x, \tag{10c}$$

$$\dot{v}_y = a_y, \tag{10d}$$

where y, v_y, a_y, v_x, and a_x respectively denote the lateral position, velocity and acceleration, and the longitudinal velocity and acceleration of E. The longitudinal velocity of S_j is denoted by v_{s_j}, and $\triangle x_j = x_{s_j} - x_E$ denotes the relative distance between E and S_j along the x-axis.

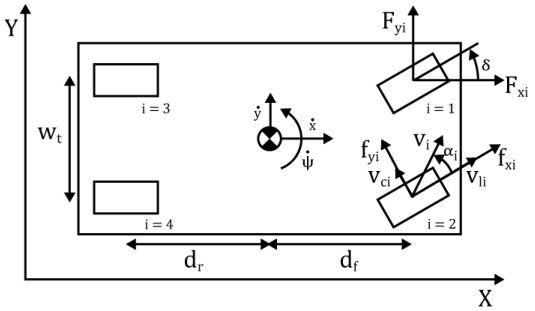

Fig. 1. Model notation for the four wheel vehicle dynamical model.

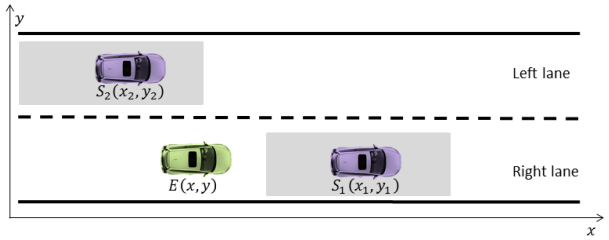

Fig. 2. Vehicles travelling on a road with two lanes. The gray boxes around the surrounding vehicles (S_1 and S_2) indicate safety critical regions which the ego vehicle (E) should not enter.

The dynamics of the point-mass model can compactly be written as

$$\dot{\xi}^{pm}(t) = f(\xi^{pm}(t), u^{pm}(t)) \tag{11}$$

where $\xi^{pm} = [\triangle x_j, y, v_x, v_y]^T$ and $u^{pm} = [a_x, a_y]^T$.

Remark 1. A target position can be reached if a feasible path connecting the initial and the target positions exists. On the other hand, the lateral and longitudinal positions in equation (10) are independent, i.e. not subject to the vehicle nonholonomic constraints. For instance, $a_x = 0$, $a_y \neq 0$ would generate a lateral movement, at a constant longitudinal velocity, that is infeasible for a real vehicle. Nevertheless, equation (10) can generate a path in a Cartesian coordinate system that can be followed by a vehicle, by limiting the side *slip angle* of the vehicle defined as $\beta = \arctan \frac{v_y}{v_x}$. In particular, by assuming $|\beta| \leq 10°(\approx 0.17\text{rad})$, small angle approximation leads to

$$-0.17 v_x \leq v_y \leq 0.17 v_x. \tag{12}$$

2.3 Collision avoidance constraints

By restricting E to stay outside safety critical regions, as illustrated in Fig. 2, it is possible to ensure that the vehicle stays on a collision free path. However, as indicated in Fig. 2, the area outside such regions is non-convex. In order to formulate the planning problem as a QP, the safety constraints must be expressed as convex sets, e.g. linear inequality constraints. Therefore, for each S_j two constraints are introduced, called the Forward Collision Constraint (FCC) and the Rear Collision Constraint (RCC). The purpose of the FCC is to keep E from colliding with its front while the purpose of the RCC is to avoid collisions with its rear, as shown in Fig. 3 and 4 respectively.

Whether the constraints should be active depends on the relative position of E and S_j. The relative position is both explicitly and implicitly included in the formulation of the FCC and the RCC in terms of Δx_j and Δy_j as well as in the slack variables depending thereof.

The FCC is defined as

$$\frac{\Delta x_j}{L_f} \pm \frac{\Delta y_j}{W} + \vartheta \varepsilon_{x_{j_f}} + \frac{\varepsilon_{y_j}}{\varphi} + \varepsilon_{j_f} \geq 1, \tag{13}$$

where the sign of the second term depends on which lane S_j is in (+ if left lane, − if right lane) and,

$$\begin{aligned} \Delta y_j &= y_{s_j} - y, \\ \varepsilon_{y_j} &= -\Delta y_j - \sigma, \\ \varepsilon_{x_{j_f}} &\geq 0, \\ \varepsilon_{j_f} &\geq 0, \end{aligned}$$

where y_{s_j} is the lateral position of S_j, and σ denotes the lane center adjacent to S_j's travelling lane. Note that

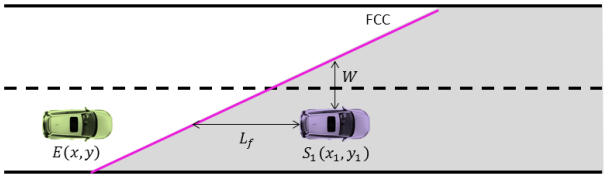

Fig. 3. The FCC enforced for a surrounding vehicle (S_1). The infeasible area generated by the FCC is displayed in gray.

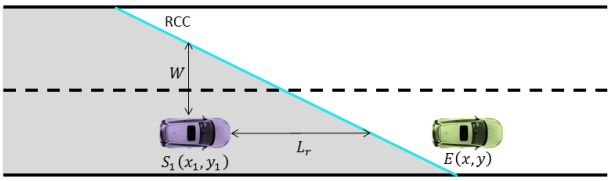

Fig. 4. The RCC enforced for a surrounding vehicle (S_1). The infeasible area generated by the RCC is displayed in gray.

$\varepsilon_{x_{j_f}}$ will only relax the constraint if E has passed S_j rendering $\Delta x_j < 0$. Further, $\varepsilon_{y_j} \geq 0$ if and only if $y_j \geq \sigma$ and the slack variable ε_{y_j} will thus only relax the FCC if E has changed lane. The slack variable ε_{j_f} should be heavily penalized in the cost function in order to only affect condition (13) if no other feasible options exist. The constant parameters are initialized before each optimization cycle as following

$$\begin{aligned} L_f &= v_x \theta_f + L_c, \\ W &= \frac{1}{2} W_L + W_c, \\ \vartheta &= -\Delta x_j, \\ \varphi &= \max(\psi, |\Delta x_j|), \end{aligned}$$

where θ_f is the desired front time gap to S_j. L_c, W_c are the length and width of S_j respectively, and W_L is the lane width. φ is set as $\max(\psi, |\Delta x_j|)$, $(\psi \geq 0)$ in order to reduce the impact of $\varepsilon_{y_j} \leq 0$ when $|\Delta x_j|$ is large.

The RCC can likewise be formulated as

$$\frac{\Delta x_j}{L_r} \pm \frac{\Delta y_j}{W} - \vartheta \varepsilon_{x_{j_r}} - \frac{\varepsilon_{y_j}}{\varphi} + \varepsilon_{j_r} \leq -1, \tag{14}$$

where the sign of the second term depends on which lane S_j is in (− if left lane, + if right lane),

$$\begin{aligned} \varepsilon_{j_r} &\leq 0, \\ \varepsilon_{x_{j_r}} &\leq 0, \end{aligned}$$

and the constant parameter is initialized before each optimization cycle as

$$L_r = v_x \theta_r + L_c,$$

where θ_r is the desired rear time gap to S_j. Further details on the formulation of the collision avoidance constraints are provided in [Nilsson et al., 2013].

Remark 2. The main drawback of formulating the collision avoidance constraints in this manner, is that during an optimization cycle a full overtake manoeuvre cannot be planned. This is because over the prediction horizon, the FCC is only relaxed when E is in the adjacent lane. Thus, only when E has actually passed S_j can the constraint be appropriately relaxed by $\varepsilon_{x_{j_f}} > 0$, and E is allowed to return to its original lane. The main advantage is that forward collisions are avoided if a collision free trajectory exists.

3. PREDICTIVE CONTROL PROBLEM

The problem of highway manoeuvre generation and control is modelled through a two-level hierarchical decomposition [Gao et al., 2010] as illustrated in Fig. 5. The high-level planner utilizes the point-mass model (11) and collision avoidance constraints (13)-(14) in order to compute a collision free manoeuvre. The resulting manoeuvre $[\dot{x}_{\text{ref}}(X), Y_{\text{ref}}(X)]$ is fed to the low-level controller which computes the optimal input required to execute the planed manoeuvre using the four wheel vehicle model (2).

The braking/acceleration forces F_L and F_R computed by the low-level controller is distributed to braking/acceleration torques T_i, $i \in \{1,2,3,4\}$, by using braking/acceleration logic as described in [Falcone et al., 2008]. Both the high-level planner and low-level controller are formulated as MPC problems where at each time instance an optimal input sequence is calculated by solving a constrained finite time optimal control problem. The optimization problem is solved in receding horizon i.e. the computed input sequence is only applied during the following sampling interval after which the problem is reformulated and resolved based on the current state and sensor measurements.

3.1 High-level planner

The goal of the high-level planner is to plan the path of E with the objective of *(i)* maintaining the vehicle at the centreline of its preferred lane, y_{ref}, and *(ii)* travelling at a desired velocity, $v_{x_{\text{des}}}$, while *(a)* avoiding collisions with S_j, *(b)* keeping E within the road boundaries, and *(c)* fulfilling E's physical and design constraints.

In order to achieve objective *(i)*-*(ii)* while allowing smooth ride comfort, the performance index i.e. cost function is defined as

$$J = \sum_{k=0}^{H_p^{pm}-1} \alpha(v_{x_k} - v_{x_{\text{des}}})^2 + \kappa(y_k - y_{\text{ref}})^2 + \gamma v_{y_k}^2 + \nu a_{x_k}^2 + \varrho a_{y_k}^2 + \chi \varepsilon_{j_f}^2 + \Xi \varepsilon_{j_r}^2, \quad (15)$$

where H_p^{pm} denotes the prediction horizon and $\alpha, \kappa, \gamma, \nu, \varrho$, χ, and Ξ are positive scalar weights. Objective *(ii)* is achieved by the $\alpha(v_{x_k} - v_{x_{\text{des}}})^2$ term, while the $\kappa(y_k - y_{\text{ref}})^2$ term satisfies objective *(i)* and implies a cost associated with a lane change manoeuvre, whereas the $\gamma v_{y_k}^2, \nu a_{x_k}^2$, and $\varrho a_{y_k}^2$ terms allow for ride comfort.

Since the sets of feasible states, $\mathcal{X}$, and control inputs, $\mathcal{U}$, are convex the MPC problem can be formulated as a standard QP problem

$$\min_w J = \frac{1}{2} w^T H w \quad (16a)$$

subject to (16b)

$$H_{eq} w = K_{eq}, \quad (16c)$$

$$H_{in} w \leq K_{in}, \quad (16d)$$

with $w = [\xi_k^{pm}, u_k^{pm}]$, (16a) is the cost function (15), (16c) is the system dynamics (11) discretized with sampling time T_s^{pm}, and (16d) includes the collision avoidance constraints (13)-(14) as well as the following physical and design constraints

$$\xi_{\min}^{pm} \leq \xi_k^{pm} \leq \xi_{\max}^{pm}, \quad (17a)$$

$$u_{\min}^{pm} \leq u_k^{pm} \leq u_{\max}^{pm}, \quad (17b)$$

$$\Delta u_{\min}^{pm} \leq \Delta u_k^{pm} \leq \Delta u_{\max}^{pm}, \quad (17c)$$

where $u_k^{pm} = \Delta u_k^{pm} + u_{k-1}^{pm}$.

Remark 3. The optimization problem (16), has $(2+q)H_p^{pm} + 4q$ optimization variables i.e. control inputs and slack variables, and $(11+4q)H_p^{pm}$ linear constraints corresponding to system dynamics (11), physical and design constraints (17a)-(17c) as well as collision avoidance constraints (13)-(14).

3.2 Low-level controller

The low-level MPC controller is formulated as follows

$$\min_{\mathcal{U}_t} \sum_{k=0}^{H_p^{4w}-1} \|\eta_{t+k,t} - \eta_{\text{ref}_{t+k,t}}\|_Q^2 + \|u_{t+k,t}^{4w}\|_R^2 + \|\Delta u_{t+k,t}^{4w}\|_S^2 \quad (18a)$$

subject to (18b)

$$\xi_{t+k+1,t}^{4w} = f^d(\xi_{t+k,t}^{4w}, u_{t+k,t}^{4w}),\ k = 0, \ldots, H_p^{4w} - 1 \quad (18c)$$

$$u_{t+k,t}^{4w} = \Delta u_{t+k,t}^{4w} + u_{t+k-1,t}^{4w},\ k = 0, \ldots, H_p^{4w} \quad (18d)$$

$$u_{\min}^{4w} \leq u_{t+k,t}^{4w} \leq u_{\max}^{4w},\ k = 0, \ldots, H_p^{4w} \quad (18e)$$

$$\Delta u_{\min}^{4w} \leq \Delta u_{t+k,t}^{4w} \leq \Delta u_{\max}^{4w},\ k = 0, \ldots, H_c^{4w} - 1 \quad (18f)$$

$$\Delta u_{t+k,t}^{4w} = 0, k = H_c^{4w}, \ldots, H_p^{4w} \quad (18g)$$

$$u_{t-1,t}^{4w} = u^{4w}(t-1), \quad (18h)$$

$$\xi_{t,t}^{4w} = \xi^{4w}(t), \quad (18i)$$

where t denotes the current time instance and $\xi_{t+k,t}^{4w}$ is the predicted state at time $t+k$ obtained by applying the control sequence $\mathcal{U}_t = [u_{t,t}, \ldots, u_{t+k,t}]$ to the discrete time version f^d of (2) with $\xi_{t,t}^{4w} = \xi^{4w}(t)$. The cost function considers the deviation of the tracking state $\eta_{t+k,t} = [\dot{x}_{t+k,t}, Y_{t+k,t}]$ from the state reference $\eta_{\text{ref}_{t+k,t}} = [\dot{x}_{\text{ref}_{t+k,t}}, Y_{\text{ref}_{t+k,t}}]$, and penalizes large control inputs and changes thereof over the prediction horizon H_p^{4w}, according to the weighting matrices Q, R, and S. H_c^{4w} denotes the control horizon and the control input is kept constant during the prediction time beyond H_c^{4w}. Further details on the low-level controller is given in [Gao et al., 2010].

Remark 4. The optimization problem (18), has $3H_c^{4w}$ optimization variables i.e. control inputs, $5H_p^{4w}$ non-linear constraints corresponding to system dynamics (2), and $15H_p^{4w}$ linear constraints corresponding to physical and design constraints (18d)-(18g).

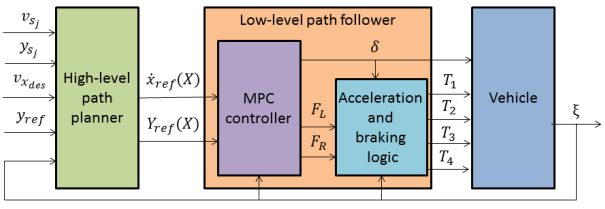

Fig. 5. Two-level hierarchical architecture for the decision and control algorithm.

4. SIMULATION RESULTS

In order to test the performance of the proposed planning and control scheme, a scenario where E is driving on a straight two-lane, one-way road with one surrounding vehicle S is considered. Three versions of this traffic scenario are created, where in each version E approaches S which is travelling at a velocity *just less*, *less*, and *much less* than E's desired velocity in scenario versions 1, 2, and 3, respectively. It is assumed that E initially travels at its desired velocity ($v_{x_{\text{des}}} = 20$ m/s), in its preferred lane i.e. the right lane ($y_{\text{ref}} = 0$). Further, it is assumed that S is travelling in the right lane at constant velocity over the prediction horizon. Although this is a strong assumption regarding the behaviour of S, it can still be considered reasonable since some changes in the behaviour of S can be handled by the replanning nature of the MPC algorithm that accounts for measurement updates. The signals v_s and y_s are treated as measurable exogenous disturbance signals which can be obtained using e.g. a doppler radar [Jansson, 2005].

For each of the described scenarios, the respective initial conditions are given in Table 1. The general design parameters for the high-level planner (16) and low-level controller (18) are given in Table 2 and 3 respectively. Note that for the parameters that do not have a physical counterpart the parameter values are selected through tuning. The closed loop system is simulated using Matlab where the two MPC optimization problems have been implemented as C-coded s-Functions. For the high-level planner (16) CVXGEN [Mattingley et al., 2012] is used. The commercial NPSOL software package [Gill et al., 2014] is used for solving the non-linear low-level control problem (18). The first element of the optimized control sequence is passed to an external block which uses a four wheel vehicle model and Pacjeka tire model to simulate the vehicle dynamics, and feeds the current state of the vehicle back to the high- and low-level optimization blocks, as shown in Fig. 5.

In Figs. 6-8 the (X,Y)-trajectory of E relative to S, as well as the longitudinal velocity profile of E are shown for each of the three scenarios respectively. In each figure both the planned and the resulting trajectories are shown, and it can be seen that low-level controller is able to follow the planned trajectories with only a slight mismatch. From the figures it can also be seen that depending on the relative velocity of E and S, the longitudinal velocity of E is reduced in order to perform an appropriate manoeuvre i.e. change lanes while keeping safe distance to S. The reduction in longitudinal velocity during the overtake manoeuvre is a consequence of the condition on lateral jerk (17c) and tuning parameters in the FCC (13). These conditions along with the selected values of the tuning parameters in Table. 2 is also the reason as to why the trajectories in Figs. 6-8 are slightly off-centred with respect to S.

Table 1. Initial conditions for the three considered scenarios with one surrounding vehicle.

	Δx_0	y_0	v_{x_0}	v_{y_0}	a_{x_0}	a_{y_0}	v_s	y_s
scenario 1	50	0	20	0	0	0	15	0
scenario 2	50	0	20	0	0	0	10	0
scenario 3	50	0	20	0	0	0	5	0

Table 2. Design parameters for the high-level planner.

$v_x \in [0, 22]$ m/s	$v_y \in [-5, 5]$ m/s	$y \in [-2.5, 7.5]$
$a_x \in [-4, 1]$ m/s^2	$a_y \in [-2, 2]$ m/s^2	$T_s^{pm} = 0.2$ s
$\Delta a_x \in [-3, 1.5]$ m/s^2	$\theta_f = 2$ s	$H_p^{pm} = 25$
$\Delta a_y \in [-0.5, 0.5]$ m/s^2	$\theta_r = 1$ s	$L_c = 5$ m
$W_L = 5$ m	$W_c = 2.5$ m	$\sigma = 4.5$
$\psi = 7$	$\alpha = 20$	$\kappa = 2$
$\gamma = 20$	$\nu = 1$	$\varrho = 1$
$\chi = 50000$	$\Xi = 50000$	

Table 3. Design parameters for the low-level controller.

$\delta \in [-10, 10]^\circ$	$\Delta\delta \in [-17, 17]^\circ$/s
$F_L \in [-1500, 1500]$ N	$F_R \in [-1500, 1500]$ N
$\Delta F_L \in [-1000, 1000]$ N/s	$\Delta F_R \in [-1000, 1000]$ N/s
$T_s^{4w} = 0.05$ s	$H_p^{4w} = 15$
$H_c^{4w} = 1$	$\mu = 0.3$
$Q = \text{diag}(30, 30)$	$R = \text{diag}(0.1, 0.1, 0.1)$
$S = \text{diag}(0.1, 0.1, 0.1)$	

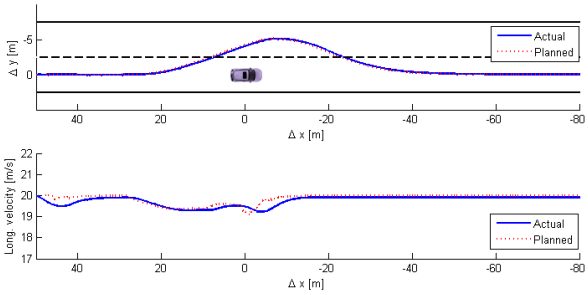

Fig. 6. The (X,Y)-trajectory of the ego vehicle relative to the surrounding vehicle, as well as the longitudinal velocity profile of the ego vehicle for scenario 1.

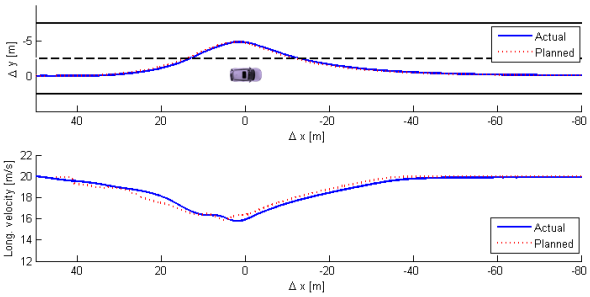

Fig. 7. The (X,Y)-trajectory of the ego vehicle relative to the surrounding vehicle, as well as the longitudinal velocity profile of the ego vehicle for scenario 2.

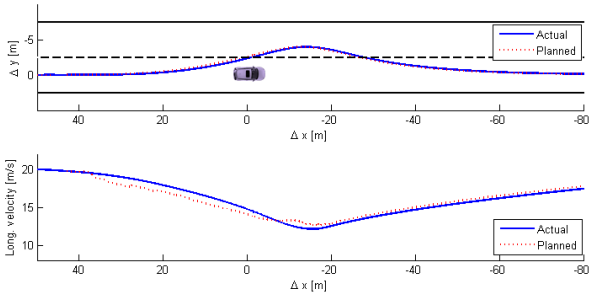

Fig. 8. The (X,Y)-trajectory of the ego vehicle relative to the surrounding vehicle, as well as the longitudinal velocity profile of the ego vehicle for scenario 3.

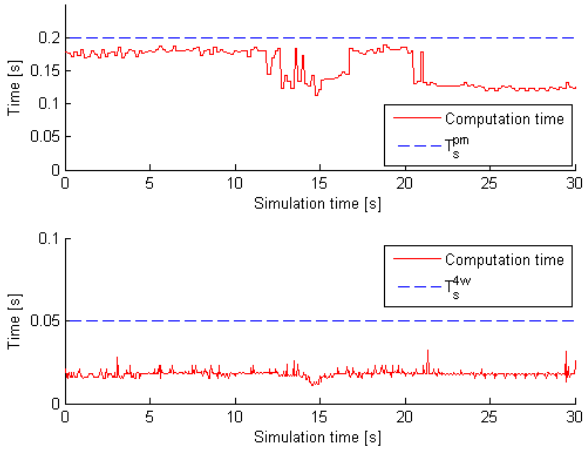

Fig. 9. Computational time for the high-level planner (top) and low-level controller (bottom).

To study the possibility of real-time implementation, the closed loop system was tested in a hardware-in-the-loop (HIL) simulation environment using a dSPACE Autobox system, equipped with a DS1005 processor board and a DS2210 I/O board. The required computational time for the high-level planner and low-level controller, for version 2 of the considered scenario is shown in Fig. 9. From these results it can be seen that real-time execution is possible, since the required computational time is less than the sampling time for both the high-level and low-level MPC optimization problems.

5. CONCLUSIONS

A two-level architecture for manoeuvre generation and vehicle control for automated highway driving has been presented. The high-level path planner computes a (X,Y)-trajectory along with a longitudinal velocity profile as the solution of a QP optimization problem. The low-level controller computes the required vehicle control inputs to follow the trajectories provided by the high-level planner by solving a low-complexity non-linear MPC problem.

Simulation results have shown the ability of the presented approach to plan appropriate manoeuvres depending on the traffic situation, in real-time.

These results motivate continued work in incorporating a dynamic prediction model of the surrounding vehicles, and also include sensor noise and uncertainty. In order to improve the traffic dependent trajectories, tuning functions for the problem parameters of the high-level planner should also be developed. Further, efforts should be made towards implementation and testing on a passenger car in real-world scenarios.

REFERENCES

Bakker, E., Nyborg, L., and Pacejka, H. B. (1987). Tyre modeling for use in vehicle dynamics control studies. *SAE tech. paper 870421*, pp. 2190-2198.

Borrelli, F., Falcone, P., Keviczky, T., Asgari, J., and Hrovat, D. (2005). MPC-based approach to active steering for autonomous vehicle systems. *International Journal of Vehicle Autonomous Systems*, vol. 3, pp. 265-291.

Borrelli, F., Subramanian, D., Raghunathan, A. U., and Biegler, L. T. (2006). MILP and NLP techniques for centralized trajectory planning of multiple unmanned air vehicles. *American Control Conference*, pp. 5764-5769.

Falcone, P., Tseng, H. E., Borrelli, F, Asgari, J., and Hrovat, D. (2008). MPC-based yaw and lateral stabilisation via active front steering and braking. *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 46, pp. 611-628.

Gao, Y., Lin, T., Borrelli, F., Tseng, H. E., and Hrovat, D. (2010). Predictive control of autonomous ground vehicles with obstacle avoidance on slippery roads. *ASME 2010 Dynamic Systems and Control Conference*, vol. 1, pp. 265-272.

Gao, Y., Gray, A., Frasch, J. V., Lin, T., Tseng, H. E., Hedrick, J. K., and Borrelli, F. (2012). Spatial predictive control for agile semi-autonomous ground vehicles. *11th International Symposium on Advanced Vehicle Control*.

Gill, P., Murray, W., Saunders, M., and Wright, M. (2014) *NPSOL-Nonlinear Programming Software*. Stanford Business Software, Inc., Mountain View, CA.

Gray, A., Gao, Y., Lin, T., Hedrick, J. K., Tseng, H. E., and Borrelli, F. (2012). Predictive control for agile semi-autonomous ground vehicles using motion primitives. *American Control Conference*, pp. 4239-4244.

Jansson, J. (2005). Collision avoidance theory with application to automotive collision mitigation. *Ph.D. Thesis, Linköping University*.

Mattingley, J. and Boyd, S. (2012). CVXGEN: A code generator for embedded convex optimization. *Optimization and Engineering*, vol. 13, no. 1, pp. 1-27.

Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. M. (2000). Constrained model predictive control : stability and optimality. *Automatica*, vol. 36, pp. 789-814.

Neale, V. L., Dingus, T. A, Klauer, S. G., Sudweeks, J., and Goodman, M. (2005). An overview of the 100 car naturalistic study and findings. *Proc. of the 19th International Technical Conference on the Enhanced Safety of Vehicles*, paper 05-0400.

Nilsson, J., Ali, M., Falcone, P., and Sjöberg, J. (2013). Predictive manoeuvre generation for automated driving. *IEEE Conference on Intelligent Transportation Systems*.

Pohl, J., Sethsson, M., Degerman, P., and Larsson, J. (2006). A semi-automated parallel parking system for passenger cars. *Journal of Automobile Engineering*, vol. 220, pp. 53-65.

Pohl, J., Birk, W., and Westervall, L. (2007). A driver-distraction-based lane-keeping assistance system. *Journal of Systems and Control Engineering*, vol. 221, pp. 541-552.