2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

Fast Maneuver Planning for Cooperative Automated Vehicles

Daniel Heß[1], Ray Lattarulo[2], Joshue Pérez[2], Julian Schindler[1], Tobias Hesse[1] and Frank Köster[1]

Abstract— **A lane following and lane changing maneuver planning method for automated vehicles is investigated, which is capable of evaluating and incorporating cooperative agreements between several automated vehicles. An application level cooperation protocol is discussed, which allows vehicles to negotiate space time reservations in conflict areas via Vehicle-to-Vehicle communication. The planning method is based on decoupling of longitudinal and lateral movement directions, formulation of convex quadratic programming problems and input-output linearization for recovery of a full state reference trajectory and feed-forward controls. Several different lane following and merging maneuvers can be planned in one update cycle in order to support an informed selection of the currently best driving strategy. We demonstrate and evaluate the communication protocol and the maneuver planning method on cooperative lane changing scenarios with a physical automated vehicle as well as in a real time simulation.**

I. Introduction

People's mobility, especially terrestrial, is one of the most important topics in the last decades. New technologies, in all these transportation areas, are looking to improve life quality of societies with the inclusion of reduced mobility persons, increase people comfort and reducing the safety risks, for example during driving. A study by the US National Highway Traffic Safety Administration [1] attributed 94% of vehicle crashes to be related with human behaviors, and 33% of them for mistakes or wrong decisions during maneuvering. A later study revealed drowsy driving represents 2.4% of the total fatalities across the US [2]. Many people hope that human related risk factors can be mitigated with the advent of the automated driving technology.

In the last decades a number of advances have been achieved in the field of automated driving, such as improvement of trajectory and speed tracking controllers [3], path planning [4], wireless communication [5] and perception. Many challenges remain due to the difficulty of introducing the technology into a complex, unstructured and unpredictable environments, which is often governed by social norms and interactions between a multitude of participants [6]. Related with this topic, the authors of [7] have conducted a study based on information given by the Bureau of Transportation Statistics in United States (2015), where it is demonstrated that a fleet of 100 automated vehicles working 24 hours all the year will require around 8 years of testing, to demonstrate a rate of 77 injuries per 100 million persons; and 500 years to obtain a rate of 1.09 fatalities per 100 million (nominal data obtained from the injuries and fatalities caused during vehicle crashes reported in United States 2013).

Many see the application of formal methods in automated driving as one possible remedy to the problematic situation of validation by test drive. In [8] a formal approach was proposed to verify that distributed vehicle control systems based on Adaptive Cruise Control (ACC) satisfy safety objective. In [9] a method for automated vehicles safety verification based on on-line reachability analysis has been presented. Optionally some of the verification tasks can be executed off-line and the results provided as safe maneuver automata [10] to an on-line planning module, which relates the results to the vehicle's current situation. Some other methods used for enforcing automated vehicles safety considered the environment as a cooperative scenario where the interaction between the actors is coordinated all the time. In [11], the authors apply probabilistic methods based on Hidden Markov Models (HMM) to risk assessment of the ego vehicle under certain traffic condition models. The approach considered the collisions as stochastic risks and it was implemented in a simulation environment to ensure free-collision situations. Other authors have proposed a minimally restrictive supervisor (centralized approach) for the problem of an intersection with multiple participants and the goal of avoiding conflicts in all the participants considering their future states [12].

With relation to our work on on-line verification for cooperative, automated vehicles in the EU project UnCoVerCPS, we have to slightly digress and investigate in this paper into a typical trajectory planning and decision module, which can be used as the module under test, or more precisely the module verified on-line in our framework. We require a module, which plans typical lane following, lane changing and merging maneuvers and supports cooperation between multiple automated vehicles. Even more importantly, the planner has to be fast enough to allow sufficient time for both planning and verification in one update cycle. An overview of merging maneuvers, with a focus on Cooperative ACC can be found in [13]. Nilsson et. al. [14] demonstrate how to formulate automated lane changing in a Model Predictive Control framework, deriving low complexity Quadratic Problems (QP) that can be solved efficiently.

In the following, sec. II will give an overview of the cooperative on-line verification framework in which we are conducting the investigation, shortly describe in sec. III the employed vehicle model and controller, with the main sec. IV detailing the concept of our maneuver planner, including

[1]Deutsches Zentrum für Luft- und Raumfahrt, Lilienthalpl. 7, 38108 Braunschweig, Germany. {daniel. hess; julian. schindler; tobias. hesse; frank. koester}@dlr.com

[2]Tecnalia Research and Innovation, 48160, Derio, Vizcaya, Spain. {rayalejandro. lattarulo; joshue. perez}@tecnalia.com

978-1-7281-0322-8/18/$31.00 ©2018 IEEE

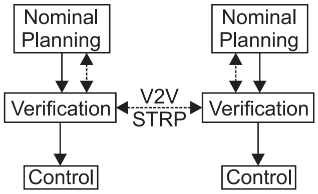


Fig. 1. Framework architecture for safe, cooperative driving.

integration of the space time reservation protocol for cooperation of automated vehicles. Sections VI and VII explain the setup and results of test drive and simulation experiments, followed by sec. VIII discussion and conclusion.

II. Framework and Approach

We have chosen an overall approach to verification of the automated driving process, which is based on an on-line analysis of each vehicle's actions. This allows to frame much simpler verification problems, albeit which have to be solved in real time. An overview of the proposed system architecture is given in fig. 1. A nominal planning module defines the driving skills required to autonomously navigate typical day to day traffic situations. Given a frequent update of filtered, labeled and tracked environment information is received from a set of perception modules, the nominal planning module creates predictions of the likely evolutions of traffic situations and determines actions, which befit its benign models and lead in almost all situations to the safe satisfaction of the vehicle's goals. Typical actions are the execution of such maneuvers as driving along a lane, following a vehicle, merging into a gap in traffic and stopping at intersections.

Obviously, there might be some situations in which the benign models do not apply and the nominal action becomes unsafe: Therefore a verification module analyses the safety of each action, before it is dispatched for execution by a controller [15]. The verification module employs worst-case models for other traffic participants as well as bounded error models for ego vehicle state sensors and ego vehicle dynamics. It then tries to plan an emergency maneuver, which brings the ego vehicle to a standstill after partial execution of the action and under exclusion of any collisions of the ego vehicle with other traffic participants or static obstacles. This has to be true for every combination of behaviors of traffic participants as well as all possible ego measurement errors and disturbances, as defined by the worst-case models. If a safe emergency maneuver is found, the current action is forwarded to the control module for execution. Otherwise the previous emergency maneuver is executed.

Let us assume the nominal planning module and the verification module operate at a fixed rate of $1/T_p$, i.e. receiving new sensor information for the i-th update at time t_i, computing first a nominal action and then the emergency maneuver in a time interval $[t_i, t_i + T_p]$. The resulting nominal action would take place during the time interval $[t_i + T_p, t_i + 2 \cdot T_p]$ and the $i-th$ emergency maneuver could start at $t_i + 2 \cdot T_p$. Akin to the reaction time of human drivers, the delay of $2 \cdot T_p$ influences the degree of conservatism of the approach: Generally speaking, with higher delay, more distance has to be kept to preceding vehicles and tighter merges are not possible. We want to answer the question, how much computation time must be foreseen for nominal maneuver planning and emergency maneuver planning, in order to asses whether the proposed on-line verification framework is practically feasible for the automated driving application. Furthermore, we require a nominal planning module for future integrated tests of our framework under realistic operation conditions. In the following sections we therefore formulate an approach to motion planning of lane following, lane changing, vehicle following, merging and cooperative merging maneuvers.

III. Vehicle Model and Stabilization

A standard linear bicycle model [16] is employed for planning and simulation. The six-dimensional state vector $x = [X, Y, \psi, v_x, v_y, \dot{\psi}]^{\mathrm{T}}$ consists of the position in Cartesian coordinates X, Y, the orientation ψ, the translational velocity vector $[v_x, v_y]^{\mathrm{T}}$ given in vehicle-fixed coordinates and the turn rate $\dot{\psi}$. The control input $u = [a_x, \delta]^{\mathrm{T}}$ consists of longitudinal acceleration and steering angle. The model is parametrized by m/I_z the ratio of translational and rotational inertia (around up axis), a and b the distance between center of gravity and front and rear axle, g the gravity constant, as well as c_f, c_r the relative cornering stiffness of front and rear axle respectively. Using $k_f := \mu g c_f b/(a+b)$, $k_r := \mu g c_r a/(b+a)$, as well as the slip angles $\beta_f = \tan^{-1}\left(\frac{v_y + a\dot{\psi}}{v_x}\right)$ and $\beta_r = \tan^{-1}\left(\frac{v_y - b\dot{\psi}}{v_x}\right)$, the following differential equation describes a bicycle model with linear tire forces:

$$\dot{X} = \cos(\psi) v_x - \sin(\psi) v_y \tag{1}$$

$$\dot{Y} = \sin(\psi) v_x + \cos(\psi) v_y \tag{2}$$

$$\dot{v}_x = a_x + v_y \dot{\psi} \tag{3}$$

$$\dot{v}_y = k_f (\beta_f - \delta) + k_r \beta_r - v_x \dot{\psi} \tag{4}$$

$$\ddot{\psi} = m/I_z \left(a k_f (\beta_f - \delta) - b k_r \beta_r\right) \tag{5}$$

In our framework, a controller is tasked to execute the trajectories computed by the nominal planner. The reference trajectory is specified at each point of time by the reference vehicle state $x^r = [X^r, Y^r, \psi^r, v_x^r, v_y^r, \dot{\psi}^r]^{\mathrm{T}}$ and the reference input (feed-forward control) $u^r = [a_x^r, \delta^r]^{\mathrm{T}}$. The following error terms describe the vehicle's deviation from the reference trajectory:

$$e_x := \cos(\psi^r)(X - X^r) + \sin(\psi^r)(Y - Y^r) \tag{6}$$

$$e_y := -\sin(\psi^r)(X - X^r) + \cos(\psi^r)(Y - Y^r) \tag{7}$$

$$e_v := v_x - v_x^r \tag{8}$$

$$e_\psi := \psi - \psi^r \tag{9}$$

$$e_\omega := \dot{\psi} - \dot{\psi}^r \tag{10}$$

A linear controller equivalent to the one proposed in [9] computes the vehicle input based on the reference input, feed-back of the above error terms and the control parameters $k_y, k_\psi, k_\omega, k_x, k_v$ in order to regulate the error terms:

$$a_x := a_x^r - k_x \cdot e_x - k_v \cdot e_v \tag{11}$$
$$\delta := \delta^r - k_y \cdot e_y - k_\psi e_\psi - k_\omega e_\omega, \tag{12}$$

IV. Nominal Maneuver Planning

The overall lane following and lane changing optimization problem is intrinsically non-convex and non-linear. We employ several approximations and partitionings of the original problem in order to gain linear-quadratic, convex sub-problems that can be efficiently solved: We restrict the solution space to lane following and single lane changing maneuvers, i.e. the solution should contain at most one transition to another lane. Furthermore, the vehicle is not allowed to overtake into head on traffic.

An integral decision variable $D = [D_L, D_G]$ is defined. $D_L \in \{-1, 0, 1\}$ encodes the target lane, with -1 for a transition to the right neighboring lane, 1 for the left neighboring lane and 0 for the current lane. $D_G \in \mathbb{N}$ is the index of the object, in front of which the ego vehicle should merge. (If there is no such object, a virtual one can be inserted). The problem is then decoupled into linear, convex quadratic optimization problems for the longitudinal and lateral dynamics of a point mass. In a last step the solutions are combined and a full state reference trajectory $x_r(t)$ and feed-forward control inputs $a_x^r(t)$ and $\delta^r(t)$ for the linear bicycle model eqs. (1)-(5) are reconstructed using input-output linearization [17].

In the following, we describe the underlying environment model, longitudinal planning, lateral planning and the reconstruction of the full state trajectory. The overall construction of a combined longitudinal and lateral motion plan is quite similar to [18], except that we employ a gradient based planning method instead of a sampling based and thus have to put more emphasis on formulation of convex constraints.

A. Preliminaries

Both the longitudinal and the lateral motion plan will be described as a solution of an optimal control problem for a third-order integrator chain. Basically the same problem type, the following notation will be used in both sections. Let us assume the dynamic optimization problem is discretized with equidistantly spaced points of time $t_0, t_1, ..., t_N$, with $T_p = t_N - t_0$. Let the integrator chain be described by the state vector $y_i^{(0:2)} := [y(t_i), y^{(1)}(t_i), y^{(2)}(t_i)]^\mathrm{T}$, the input $y_i^{(3)}$ and the differential equation $y_i^{(0:2)} = A \cdot y_{i-1}^{(0:2)} + B y_{i-1}^{(3)}$, $A \in \mathbb{R}^{3,3}, B \in \mathbb{R}^{3,1}$. Given a weight vector $w \in \mathbb{R}^4$, an initial condition $y_0^{(0:2)}$, lower and upper bounds for each point of time i and each derivative j, $y_{lb|ub,i}^{(j)}$, as well as a reference $\tilde{y}_i^{(j)}$, a Quadratic Programming (QP) problem with a solution $Y^* = \left[[y_0^{(0:2)}, y_0^{*(3)}]^\mathrm{T}, ..., y_N^{*(0:3)}\right]^\mathrm{T}$ is defined:

$$\begin{aligned} Y^* = \arg\min \quad & \sum_{i=0}^{N} \sum_{j=0}^{3} w_j \left(y_i^{(j)} - \tilde{y}_i^{(j)}\right)^2 \\ \text{s. t.} \quad & \forall_{i=0}^{N} \forall_{j=0}^{3} : y_i^{(j)} \in [y_{lb,i}^{(j)}, y_{ub,i}^{(j)}] \\ & \forall_{i=1}^{N} : y_i^{(0:2)} = A \cdot y_{i-1}^{(0:2)} + B \cdot y_{i-1}^{(3)} \end{aligned} \tag{13}$$

Let $y^*(t)$ denote the appropriately interpolated continuous function based on the optimization result. Consequently, the longitudinal and lateral solution will be defined by the number of points, a set of constraints $y_{lb|ub}^{(0:3)}$, a reference $\tilde{y}^{(0:3)}$ and a weight vector $w \in \mathbb{R}^4$. Let the variable y in these cases stand for the longitudinal variable s or the lateral variable d.

B. Environment Model

In order to decouple the motion planning problem into longitudinal and lateral planning, a smooth, bijective mapping from a longitudinal coordinate s and a lateral coordinate n to the Euclidean space covered by the current and the adjacent lanes is required.

Assume the center of the current lane is described as shown in fig. 2 by $N_c \in \mathbb{N}$ sampling points $c_1, \ldots, c_{N_c}$, $c_i \in \mathbb{R}^2$, spaced equidistantly at Δs, from a distance s_{lbh} behind the ego vehicle, up to a distance s_{lah} ahead of the ego vehicle. Non-linear regression is employed to gain $N_q \in \mathbb{N}, N_q \cdot k = N_c$ cubic, two-dimensional polynomials $q_1, \ldots, q_{N_q}$, $q_i \in (\mathbb{R}_3[s])^2$, $q_i : [s_i, s_{i+1}] \to \mathbb{R}^2$, which guarantee smoothness up to second derivative and which minimize the squared distance to the sampling points:

$$\begin{aligned} q_{1..N_q} := & \arg\min \sum_{i=1}^{N_q} \sum_{j=1}^{k} \| q_i(s_i + j\Delta s) - c_{(i-1)\cdot k + j} \|_2^2 \\ \text{s. t. :} \quad & q_i^{(d)}(s_{i+1}) = q_{i+1}^{(d)}(s_{i+1}) \\ & d \in \{0, 1, 2\} \end{aligned} \tag{14}$$

The overall piecewise polynomial $q : [s_1, s_{N_q}] \to \mathbb{R}^2$, $q(s) = q_{i(s)}(s)$ with $i(s) = \lceil (s - s_1)/(k\Delta s) \rceil$ is used as a baseline for the required mapping. The baseline normal $n(s)$ and curvature $\kappa(s)$ are computed as:

$$n := \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right) q' / \|q'\|_2 \tag{15}$$
$$\kappa := (q_Y'' q_X' - q_X'' q_Y') / \|q'\|_2^2 \tag{16}$$

A transformation $T : \mathbb{R}^2 \to \mathbb{R}^2$ from a road-relative coordinate (s, d) to a Euclidean coordinate (X, Y) is defined as:

$$T : [X, Y]^\mathrm{T} = q(s) + n(s) \cdot d \tag{17}$$

Using an implicitly realized approximation of the inverse transformation $T^{-1} \approx \Gamma(p) = [\arg\min (\|q(s) - p\|), n(s)^\mathrm{T} p]^\mathrm{T}$, the scalar functions $b_i : s \mapsto d, i \in \{0, \ldots, 3\}$ are computed, so that $T(s, b_0(s))$ describes the right border of the right lane, $T(s, b_1(s))$ the right border of the current lane and so forth as shown in fig. 2.

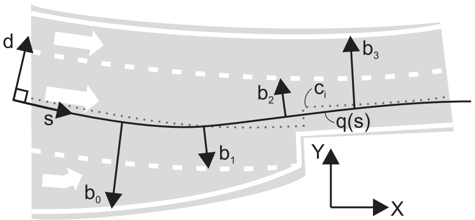


Fig. 2. Representation of the current and both adjacent lanes in a non-linear coordinate system defined by piecewise polynomial$q$.

For each detected object k in the ego vehicle's environment, assume the X, Y position of the center of its rear bumper, $p_{k,0}$, its velocity v_k, as well as its length L_k and width w_k are given. For each stop line, waiting position and lane closure a similar object with $v_k = 0$ is added to the object set, to simplify notation. For each object the transformation $(s_{k,0}, d_{k,0}) = \Gamma(p_{k,0})$, e.g. a lane matching, is computed. For each lane $l \in \{-1, 0, 1\}$, a set of indices of all objects which intersect the lane is created:

$$O_l := \left\{ k \,\middle|\, \begin{array}{rl} & d_{k,0} + w_{k,0}/2 \geq b_{l+1}(s_{k,0}) \\ \wedge & d_{k,0} - w_{k,0}/2 \leq b_{l+2}(s_{k,0}) \end{array} \right\}. \tag{18}$$

We employ a linear prediction for other traffic participants and given a length of the ego vehicle L_e a time gap Δt_g and a constant safety distance Δs_g, the forward and rearward safety boundaries are defined for an object k:

$$s_k^+(t) := s_{k,0} + v_k \cdot t + L_k + v_k \cdot \Delta t_g + \Delta s_g \tag{19}$$

$$s_k^-(t) := s_{k,0} + v_k \cdot t - L_e - v_k \cdot \Delta t_g - \Delta s_g \tag{20}$$

One has to consider, which objects are relevant for the planning task. If the selected maneuver is lane following, e.g. $D_{LC} = 0$, only objects on the current lane in front of the ego vehicle are relevant. The lower bound on these vehicle positions will be designated by the term *front (f)*. For lane changing, $D_{LC} \neq 0$, additionally the lower bound of vehicles preceding the gap in the target lane, designated *lead (l)*, and the upper bound on vehicles following the gap in the target lane, designated *chase (c)*, are of relevance to the maneuver. These position bounds are defined as:

$$s_f(t) := \min\left\{ s_k^-(t) \,|\, k \in O_0 \wedge s_{k,0} > s_{e,0} \right\} \tag{21}$$

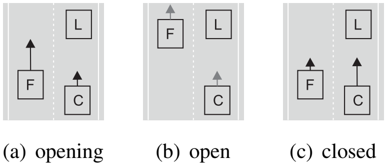
$$s_l(t) := \min\left\{ s_k^-(t) \,|\, k \in O_{D_{LC}} \wedge s_{k,0} > s_{D_{gap},0} \right\} \tag{22}$$

$$s_c(t) := \max\left\{ s_k^+(t) \,|\, k \in O_{D_{LC}} \wedge s_{k,0} \leq s_{D_{gap},0} \right\} \tag{23}$$

The longitudinal planner is supplied with the environment information encoded in the position constraints $s_f(t)$, $s_l(t)$, $s_c(t)$, the speed limit $\hat{v}(s)$ and the initial state $[s_{e,0}, \dot{s}_{e,0}, \ddot{s}_{e,0}]^{\mathrm{T}}$. The lateral planner is additionally supplied with the initial state $[d_{e,0}, \dot{d}_{e,0}, \ddot{d}_{e,0}]^{\mathrm{T}}$ and the lane boundaries $b_0(s), ..., b_3(s)$.

C. Longitudinal Planning

Consider the window of opportunity for execution of a merge into a target gap, as shown in fig. 3. A merge is only possible at a point of time, when the front vehicle in the ego vehicle's own lane is in front of the chase vehicle in the target lane. Depending on the relative velocities of front and chase vehicle, the window of opportunity opens in the future, fig. 3 (a) leads to (b), or closes in the future, fig. 3 (b) leads to (c). The points of time for transition from *opening* to *open* and from *open* to *closed* are:



(a) opening (b) open (c) closed

Fig. 3. Possibility to merge into a gap between lead (L) and chase (C) vehicles depending on configuration of front (f) and chase vehicle.

$$t_{g0} := \min\{t \,|\, s_c(t) < s_f(t)\}, \tag{24}$$

$$t_{g1} := \max\{t \,|\, s_c(t) < s_f(t)\}. \tag{25}$$

Let us assume t_{g0} and t_{g1} are appropriately bounded for the case that s_f and s_c only intersect at $-\infty$ or ∞ and that $t_{g0} = t_{g1} = \infty$ for $D_{LC} = 0$. The upper and lower bounds for the ego vehicle position are defined using these gap switching times:

$$s_{ub}(t) := \begin{cases} s_f(t) & t < t_{g0} \\ \max(s_f(t), s_l(t)) & t_{g0} \leq t < t_{g1} \\ s_l(t) & t_{g1} < t \end{cases} \tag{26}$$

$$s_{lb}(t) := \begin{cases} -\infty & t < t_{g1} \\ s_l(t) & t_{g1} \leq t \end{cases} \tag{27}$$

A speed limit is by its nature a position dependent constraint, so it would be a non-linear constraint for the optimization problem. In order to approximate the velocity constraint, an upper bound on the vehicle position $\hat{s}(t)$ is estimated. The approximate velocity constraint is defined by the speed limit and the curvature at the estimated position:

$$s_{ub}^{(1)}(t) := \min\left(\hat{v}(\hat{s}(t)), \sqrt{\hat{a}_y \kappa^{-1}(\hat{s}(t))} \right) \tag{28}$$

The lower velocity bound is set $s_{lb}^{(1)} = 0$ to prevent the vehicle from traveling against the movement direction of the lane. The bounds for longitudinal acceleration are constant parameters, $s_{lb}^{(2)} := \hat{a}_{lb}$, $s_{ub}^{(2)} := \hat{a}_{ub}$ and the jerk is set to be unconstrained, $s_{lb}^{(3)} := -\infty$, $s_{ub}^{(3)} := \infty$.

In order to provide a reference for the longitudinal vehicle position, as a first step a target position $\bar{s}(t)$ is defined, which we chose as the foremost position attainable in the target gap, so that $\bar{s}(t) := s_f(t)$ for $D_{LC} = 0$ and $\bar{s}(t) := s_l(t)$ for $D_{LC} \neq 0$. As a second step, a reference is computed, which maintains the forward acceleration bound and the maximum speed whenever possible, in order to prevent unrealistically big position differences. The velocity and position reference are recursively defined, using the initial condition $\tilde{s}_0 := s_{e,0}$, $\tilde{s}_0^{(1)} = s_{e,0}^{(1)}$, the target position $\bar{s}$, the position based speed limit $\hat{v}$ and $\Delta t_{si} = (t_{si+1} - t_i)$:

$$\tilde{s}_{i+1}^{(1)} := \min\left(\hat{v}(\tilde{s}_i), \tilde{s}_i^{(1)} + \Delta t_{si} \cdot a_{max} \right) \tag{29}$$

$$\tilde{s}_{i+1} := \min\left(\bar{s}(t_{si+1}), s_{ub}(t_{si+1}), \tilde{s}_i + \Delta t_{si} \cdot \tilde{s}_i^{(1)} \right) \tag{30}$$

The reference jerk and acceleration are defined to be zero $\tilde{s}^{(3)} \equiv 0$, $\tilde{s}^{(2)} \equiv 0$.

Using the QP for an integrator chain (13), above definitions of bounds $s^{(0:3)}_{lb|ub}(t)$ and reference $\tilde{s}^{(0:3)}(t)$ under consideration of the decision variable D, as well as the parameters plan duration $T_{p,s}$, number of discretization points N_s and weight vector w_s, the longitudinal motion plan $s^*(t)$ is computed.

D. Lateral planning

The lateral motion plan is always computed after the longitudinal planning and the availability of the longitudinal plan $s^*(t)$ allows to handle all time and position based constraints as constant constraints. In order to prevent collisions with surrounding vehicles, we specify lateral position constraints, which allow the vehicle to leave its own lane only, if it is executing a lane change and if it is correctly aligned with the target gap. Furthermore, as soon as the longitudinal position s^* is overtaking the front safety margin s_f, the lateral position must be contained in the target lane. Using indices $lo = 1$, $mid = 2$ and $hi = 3$ for lane changes to the left with $D_{LC} = 1$ and $lo = 2$, $mid = 1$ and $hi = 0$ for lane changes to the right with $D_{LC} = -1$, the lateral position $d(t)$ must be adhere to the following bounds:

$$d^+(t) := \begin{cases} b_{hi}(s^*(t)) & s_c(t) < s^*(t) < s_l(t) \\ b_{mid}(s^*(t)) & \text{else} \end{cases} \tag{31}$$

$$d^-(t) := \begin{cases} b_{lo}(s^*(t)) & s^*(t) < s_f(t) \\ b_{mid}(s^*(t)) & \text{else} \end{cases} \tag{32}$$

$$d_{ub}(t) := \max(d^-(t), d^+(t)) - w/2 \tag{33}$$

$$d_{lb}(t) := \min(d^-(t), d^+(t)) + w/2 \tag{34}$$

We leave the higher derivatives unconstrained, $d^{(1)}_{ub} := \infty$, etc. The lateral reference position is switched between the center of the current lane $b_{cu}(s) := (b_{lo}(s) + b_{mid}(s))/2$ and the center of the target lane $b_{ta}(s) := (b_{mid}(s) + b_{hi}(s))/2$:

$$\tilde{d}_i := \begin{cases} b_{ta}(s^*(t_i)) & s_c(t_i) < s^*(t_i) < s_l(t_i) \\ b_{cu}(s^*(t_i)) & \text{else} \end{cases} \tag{35}$$

The lateral reference speed is set $\tilde{d}^{(1)}(t) \equiv 0$. Setting the lateral acceleration and jerk reference to zero would induce the ego vehicle to abidingly track all motions of the base line polynomial q inside the given bounds. If instead it is desirable to compensate unnecessary lateral motions of the base line, the compensating lateral acceleration and jerk can be approximated using the curvature and curvature derivative of q:

$$\tilde{d}^{(2)}_i := -\left(s^{*(1)}(t_i)\right)^2 \cdot \kappa\left(s^*(t_i)\right) \tag{36}$$

$$\tilde{d}^{(3)}_i := -\left(s^{*(1)}(t_i)\right)^3 \cdot \kappa'\left(s^*(t_i)\right) \tag{37}$$

The lateral motion plan $d^*(t)$ is gained by solving the QP (13), with the above definition of lateral constraints $d^{(0:3)}_{ub|lb}(t)$ and reference $\tilde{d}^{(0:3)}(t)$, as well as the constant parameters plan duration $T_{p,d}$, number of discretization points N_d and weight vector w_d.

E. Reference Trajectory and Feed-Forward Controls

The longitudinal motion plan $s^*(t)$ and the lateral profile $d^*(t)$ are combined into a reference trajectory τ represented in Cartesian coordinates, $\tau(t) = [\tau_x(t), \tau_y(t)]^{\mathrm{T}}$:

$$\tau(t) := \rho\left(s^*(t)\right) + n\left(s^*(t)\right) \cdot d^*(t) \tag{38}$$

The application of input-output linearization to motion planning is well known [18] and the zero dynamics of a vehicle model tracking τ can be acquired by solution of an initial value problem. We here express the second order zero dynamics in the variables $[\psi^r, \dot{\psi}^r]$ and solve the following equation:

$$\ddot{\psi}^r = m/I_z \left(a k_f \left(\beta^r_f - \delta^r \right) - b k_r \beta^r_r \right). \tag{39}$$

For each time step of the initial value problem, the remaining state variables and the feed-forward controls are computed by inversion of the system dynamics (1)-(5):

$$X^r = \tau_x, \; Y^r = \tau_y \tag{40}$$

$$v^r_x = \cos(\psi^r)\dot{\tau}_x + \sin(\psi^r)\dot{\tau}_y \tag{41}$$

$$v^r_y = -\sin(\psi^r)\dot{\tau}_x + \cos(\psi^r)\dot{\tau}_y \tag{42}$$

$$a^r_x := \cos(\psi^r)\ddot{\tau}_x + \sin(\psi^r)\ddot{\tau}_y \tag{43}$$

$$\begin{aligned} \delta^r := {} & \beta^r_f + k_r {k_f}^{-1} \beta^r_r \\ & + k_f^{-1}\left(\sin(\psi^r)\ddot{\tau}_x - \cos(\psi^r)\ddot{\tau}_y\right). \end{aligned} \tag{44}$$

V. V2V Cooperation Protocol

Vehicle to vehicle communication gives automated vehicles the opportunity to exchange intentions and negotiate cooperations in a very explicit form of representation [19]. Instead of using an indicator, which communicates a general intend to change to a certain lane, radio communication allows vehicles to discuss explicitly, where and when a lane change will take place and whether other vehicles will support such a maneuver. Some approaches transmit the specific trajectories [20] of each cooperating vehicle, which can be data intensive. Instead, it may suffice to exchange constraints, which decouple the trajectories of all involved entities. Based on [15] we propose a simple space-time reservation protocol, which allows vehicles to negotiate constraints in order to resolve conflicting traffic situations. A cooperative exchange involves one requesting and any number of confirming vehicles and is based on the two corresponding types of broadcast messages, Request and Confirm. Per exchange, one Request message is sent and one Confirm message per answering vehicle is sent back.

The cooperative exchange is initiated by an imminent conflict in vehicle positions: When a vehicle V_R intends to execute a maneuver such as changing to or crossing over a higher priority lane, it must take into account possible conflicts with future actions of vehicles V_P on the targeted higher priority lane. If V_R's intended maneuver is not feasible under given predictions, it broadcasts a Request message containing its station id R_{ID} and a running number M_{ID}, which uniquely identifies the negotiation session. Most importantly it specifies a lane area to be reserved for V_R, as indicated in fig. 4: A part of the lane starting at a coordinate

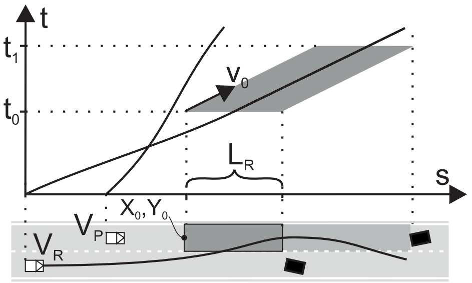


Fig. 4. Reserved area in s and t for a cooperative lane change. Requesting vehicle V_R and cooperating vehicle V_P.

X_0, Y_0, with a length L_R along the lane is reserved for a time interval $[t_0, t_1]$. The reservation moves in the lane's direction with a speed v_0, starting at t_0.

Any vehicle V_P receiving the request determines whether the request has acceptable impact on the vehicle's objectives and whether it contradicts internal constraints. Coming to a negative conclusion, no answering message is sent, otherwise consent is given by broadcasting a Commit message, which relates to the original request's constraints by specifying R_{ID}, M_{ID} and identifies the consenting vehicle by its station id. The confirmation message on the one hand obliges V_P to avoid any overlap with the reserved area. This can be achieved by adding corresponding constraints to V_P's maneuver planner. On the other hand, V_P's confirmation allows V_R to relax the constraints imposed by the prediction of V_P's intent, according to the negotiated reservation. Interpreted for the example given in fig. 4, V_R may enter the left lane in the reserved area between t_0 and t_1 due to a confirmation of V_P. The number of vehicles, which actively change their original plan due to the negotiated cooperation, depends on the situation: A positive outcome of the negotiation certainly influences V_R, as it facilitates an action otherwise unfeasible. Vehicles V_P may be obliged to change their plan, when their original plan is invalidated by the newly accepted constraint.

The space time reservation protocol is integrated into the above nominal maneuver planning as follows: The committing vehicle adds a virtual vehicle of length L_R to the targeted lane object set during the time interval $[t_0, t_1]$, starting at an s position given by $\Gamma(X_0, Y_0)$ with a velocity v_0. The requesting vehicle removes all vehicles, which sent a commit message, from the object set of the targeted lane during the same time interval and adds upper and lower bounds according to the reserved area to the s_l and s_c constraints.

VI. Experiment Setup

A nominal maneuver planner is implemented according to the above description in c++, using qpOASES [21] to solve the quadratic problems for longitudinal and lateral planning. At a rate of 10 Hz, the nominal planner sends and receives cooperation messages, computes the environment model according to above description, sequentially computes lane following and lane changing maneuvers, selects a valid maneuver for execution and sends a reference trajectory to a control process. Vehicle model, controller and nominal planner are parametrized with the values given in tab. I.

Fig. 5. DLR automated vehicle FASCarE used in test drive

vehicle	g	μ	a	b	I_z/m	c_f	c_r
	$9.81\,\frac{\mathrm{m}}{\mathrm{s}^2}$	0.8	1.01m	1.68m	$1.57\,\frac{\mathrm{m}^2}{\mathrm{s}^2}$	10.8	17.8
controller	k_y	k_ψ	k_ω	k_x	k_v		
	0.05	0.4	0.25	0.5	1.4		
env. repr.	N_c	N_q	s_{lbh}	s_{lah}			
	200	20	50m	200m			
long. plan.	$T_{p,s}$	N_s	v_{max}	$\hat{a}_x$	$\hat{a}_y$	w_s	
	10s	20	$16.7\,\frac{\mathrm{m}}{\mathrm{s}}$	[-3,2] $\frac{\mathrm{m}}{\mathrm{s}^2}$	[-1,1] $\frac{\mathrm{m}}{\mathrm{s}^2}$	$(0.5, 1, 1, 1)$	
lat. plan.	$T_{p,d}$	N_d	w_d				
	10s	20	$(2, 1, 3, 3)$				

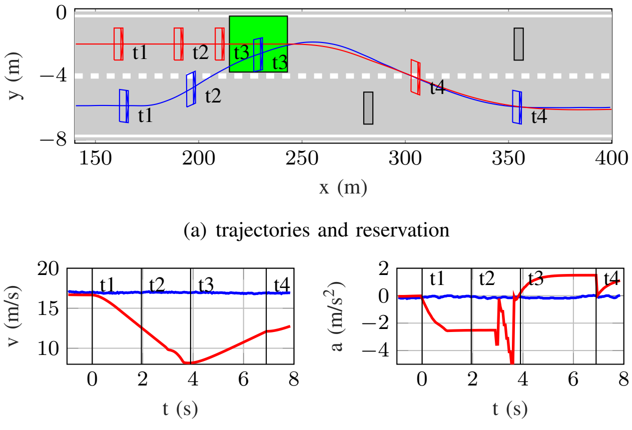
TABLE I

Three experiments have been carried out: Experiment A and B use the same scenario of a 500m long two lane road, with the same driving direction for both lanes and two static obstacles at distance 280m and 350m. Both vehicles start ca. at 20m with $v_x = 0$ on adjacent lanes and accelerate up to the maximum velocity of 16.7 m/s. Experiment A is a combination of physical test drive and simulation: The FASCarE automated vehicle, fig. 5, is equipped with a Novatel differential GPS/INS, which provides state estimates at 20 Hz, a control PC, a control interface, which allows to set desired acceleration a_x and steering angle δ, (as well as a set of environmental sensors, which are not used in this work). The PC executes two sets of automated vehicle processes: One set for the physical vehicle itself and one set for a simulated vehicle, which cooperates with the physical vehicle. In experiment A, the physical vehicle starts in the right lane. In experiment B, both vehicles are simulated.

Experiment C is a simulation experiment for an urban scenario with high lane curvatures and a varying number of lanes. As in A and B, two static obstacles are placed at different positions to induce lane changes. The admissible top speed of one vehicle is reduced to 7 m/s. The slow vehicle is placed 35 m ahead of of the fast vehicle, which leads to the fast vehicle catching up to the slow vehicle right in front of the first blocking obstacle. Both experiment B and C use a real time simulation environment, e.g. the timing acquired in simulation is directly comparable to test drive results.

VII. Results

The results of experiment A are shown in fig. 6: Vehicle positions for four points of time, before during and after the cooperation are shown. The physical vehicle 0 on the right lane and the simulated vehicle 1 on the left lane arrive at their top speed with only a minor deviation in x-distance. At t_1 vehicle 0 sends a cooperation request to vehicle 1 in order to

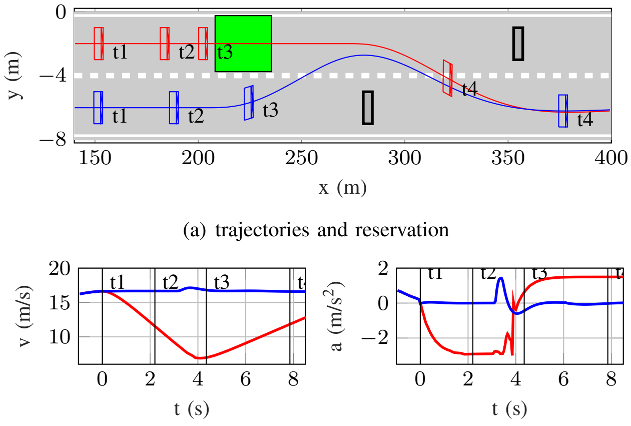


(a) trajectories and reservation

(b) velocity

(c) acceleration

Fig. 6. Experiment A - Cooperation of a physical and a simulated vehicle

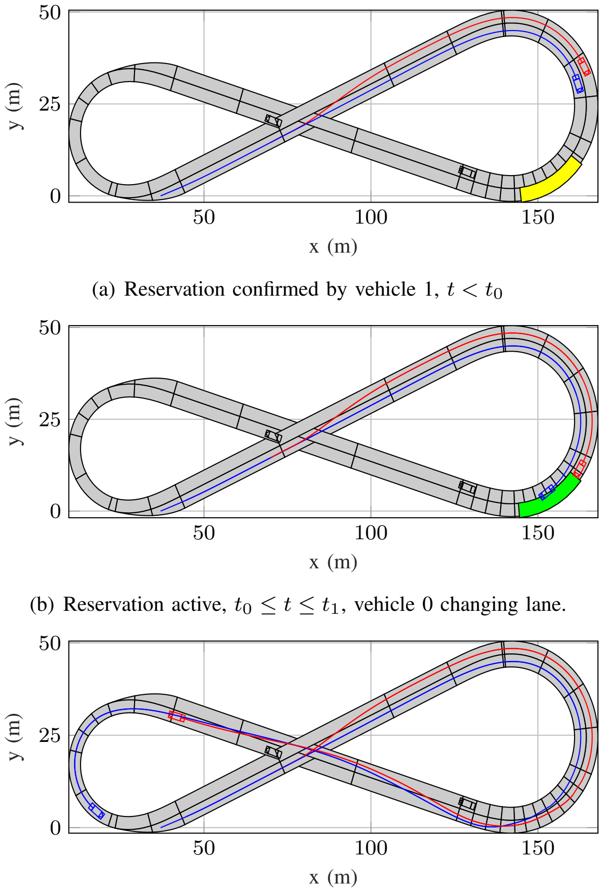


(a) trajectories and reservation

(b) velocity

(c) acceleration

Fig. 7. Experiment B - Cooperation of two simulated vehicles



(a) Reservation confirmed by vehicle 1, $t < t_0$

(b) Reservation active, $t_0 \leq t \leq t_1$, vehicle 0 changing lane.

(c) Overall paths taken by cooperating vehicles

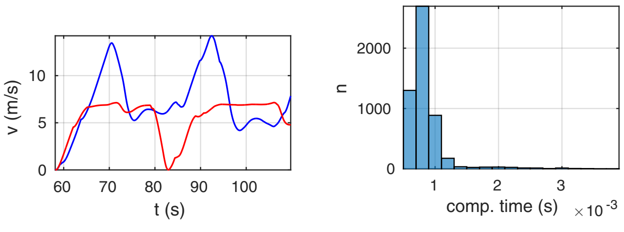
Fig. 8. Experiment C - Cooperation of two simulated vehicles on a more complex road graph.

reserve the green area on the left lane for a reservation start time equaling t_3 and directly receives a promise message. Vehicle 1 starts braking at t_1 to avoid the reserved area, so that vehicle 0 is ahead of vehicle 1 at t_2. Vehicle 0 changes lanes and enters the reserved area in front of vehicle 1. At t_3, vehicle 1 starts to accelerate back to top speed. Both vehicles then non-cooperatively execute a lane change to the right lane, to avoid the second obstacle.

During experiment A, a programming error became obvious: Vehicle 0 started the lane change too early, because we initially removed the promising vehicles completely from the constraint set, not only during the reservation time interval $[t_{R0}, t_{R1}]$. The error was corrected and simulation experiment B was carried out, which is shown in fig. 7: One can see that vehicle 0 initiates the lane change just slightly ahead of the reservation activation time $t_{R0} = t_3$, so that vehicle 0 fully remains inside its own lane until at least t_3.

The results of experiment C are shown in figs. 8-9(b): The slow vehicle 1 (red) initially changes to the left lane at the entrance of the curve, as it is already aware of the blocked left lane at the exit of the curve. Vehicle 0 (blue) intends to drive faster and tries to overtake vehicle 1 by pursuing the inner, right lane. Yet due to the upper bound on lateral acceleration, it is limited to a similarly low speed inside the curve. At the point of time $t = 80$s depicted in fig. 8(a), vehicle 0 reserves a portion of the outer lane in front of vehicle 1. The reservation is shown in yellow, as it subsequently becomes active at $t = 83$s. Vehicle 1 reduces its velocity (coming to a standstill at $t = 83$s due to the low initial velocity), in order to avoid the reserved area. As shown in fig. 8(b) for $t = 83$s, vehicle 0 changes lanes into the center of the active (green) reservation area. With the faster vehicle 0 ahead, both vehicles go on to non-cooperatively change lanes to avoid the second obstacle on the left lane as well as the closure of the right lane (fig. 8(c) for $t = 107$s), while approaching admissible top speeds on the straight parts of the road and reducing the speed appropriately during curves.

The computation time for maneuver planning is evaluated in experiment C. The simulation is carried out on a PC with an Intel Core i7-6820HQ CPU, Windows 7 operating system and sources compiled with the Visual Studio 2010 c++ compiler. The duration for computing each maneuver was recorded, where the computation steps for one maneuver are formulating and solving one longitudinal QP, solving one lateral QP and solving an initial value problem for



(a)

(b)

Fig. 9. Experiment C - Velocity and computation time

reconstruction of the complete reference state. The time required for preparation of the environment model is not included in the measurement, as it is associated with multiple plan computations for different decision variable values. Furthermore, it could be easily outsourced to a parallel computation process. During each 10 Hz cycle of the two nominal maneuver planners between 1 and 3 maneuvers are computed, depending on the availability of target lanes and target gaps. Fig. 9(b) gives a histogram with the number of occurrences for a maneuver computation taking between 0.5 ms and 4 ms with 0.2ms wide bins. The highest observed computation time was 3.5ms, with most computations taking ca. 1ms. The computation time limit per QP was set to 10ms, so each feasible problem was solved to the specified degree of precision.

VIII. Discussion and Conclusion

We describe and evaluate a method for planning of nominal maneuvers for automated vehicles, which is based on longitudinal and lateral decoupling, solution of Quadratic Programs and reconstruction of a full state vehicle trajectory according to the idea of input-output linearization. The planner provides lane following and lane changing maneuvers, as well as cooperative lane changing maneuvers based on a space time reservation protocol, which allows automated vehicles to negotiate cooperation for execution of future maneuvers in critical situations. Our experiments show through physical test drive and simulation that a real time application with 10Hz update rate, which includes planning of multiple, alternative nominal maneuvers as well as computation of emergency maneuvers similar to [22] as a verification step is feasible. Our next steps will be to integrate nominal planning and emergency maneuver planning for a cooperative, automated driving use case.

Acknowledgment

The authors gratefully acknowledge financial support from the European Commission project UnCoVerCPS under grant number 643921 and from Deutsche Forschungsgemeinschaft (DFG) SPP 1835 CoInCar and subproject CoInCiDE under grant number KO 1990/3-1.

References

[1] US Department of Transportation. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. In *Traffic Safety Facts Crash Stats*. NHTSA, 2015.

[2] US Department of Transportation. A brief statistical summary: Drowsy driving. In *Traffic Safety Facts Crash Stats*. NHTSA, 2011.

[3] Alireza Khodayari, Ali Ghaffari, Sina Ameli, and Jamal Flahatgar. A historical review on lateral and longitudinal control of autonomous vehicle motions. In *2nd Int. Conf. on Mechanical and Electrical Technology (ICMET)*, pages 421 – 429. IEEE, 2010.

[4] David González, Joshué Pérez, Vicente Milanés, and Fawzi Nashashibi. A review of motion planning techniques for automated vehicles. In *Transactions on Intelligent Transportation Systems*, volume 17(4), pages 1135 – 1145. IEEE, 2016.

[5] Haijian Bai, Jianfeng Shen, Liyang Wei, and Zhongxiang Feng. Accelerated lane-changing trajectory planning of automated vehicles with vehicle-to-vehicle collaboration. *Journal of Advanced Transportation*, 2017.

[6] Hadas Kress-Gazit and George J. Pappas. Automatically synthesizing a planning and control subsystem for the darpa urban challenge. In *4th Conf. on Automation Science and Engineering*, page 766 771. IEEE, 2008.

[7] Nidhi Kalra and Susan M. Paddock. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? In *Transportation Research Part A: Policy and Practice*, volume 94, pages 182 – 193. ELSEVIER, 2016.

[8] Sarah M. Loos, André Platzer, and Ligia Nistor. Adaptive cruise control: Hybrid, distributed, and now formally verified. In *International Symposium on Formal Methods*, pages 42 – 56. Springer, Berlin, 2011.

[9] Matthias Althoff and John M. Dolan. Online verification of automated road vehicles using reachability analysis. In *IEEE Transactions On Robotics*, volume 30, pages 903 – 918. 2014.

[10] Daniel Heß, Matthias Althoff, and Thomas Sattel. Formal verification of maneuver automata for parameterized motion primitives. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 1474–1481. IEEE, 2014.

[11] Christian Laugier, Igor E. Paromtchik, Mathias Perrollaz, Mao Yong, John-David Yoder, Christopher Tay, Kamel Mekhnacha, and Amaury Négre. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. In *Intelligent Transportation Systems Magazine*, pages 4 – 19. IEEE, 2011.

[12] Alessandro Colombo and Domitilla Del Vecchio. Least restrictive supervisors for intersection collision avoidance: A scheduling approach. In *IEEE Transactions on Automatic Control*, volume 60, pages 1515 – 1527. IEEE, 2014.

[13] David Bevly, Xiaolong Cao, Mikhail Gordon, Guchan Ozbilgin, David Kari, Brently Nelson, Jonathan Woodruff, Matthew Barth, Chase Murray, Arda Kurt, et al. Lane change and merge maneuvers for connected and automated vehicles: A survey. *IEEE Transactions on Intelligent Vehicles*, pages 105–120, 2016.

[14] Julia Nilsson, Mattias Brännström, Erik Coelingh, and Jonas Fredriksson. Lane change maneuvers for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1087–1096, 2017.

[15] Daniel Heß, Christian Löper, and Tobias Hesse. Safe cooperation of automated vehicles. In *Proc. of the AAET*. ITS automotive nord e.V., 2017.

[16] Rajesh Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[17] G Conte, CH Moog, and AM Perdon. Input/output linearization. *Nonlinear control systems: An algebraic setting*, pages 79–82, 1999.

[18] Moritz Werling, Julius Ziegler, Sören Kammel, and Sebastian Thrun. Optimal trajectory generation for dynamic street scenarios in a frenet frame. In *Robotics and Automation (ICRA), 2010 IEEE Int. Conf. on*, pages 987–993, 2010.

[19] Laurens Hobert, Andreas Festag, Ignacio Llatser, Luciano Altomare, Filippo Visintainer, and Andras Kovacs. Enhancements of v2x communication in support of cooperative autonomous driving. *IEEE communications magazine*, 53(12):64–70, 2015.

[20] F Visintainer, L Altomare, A Toffetti, A Kovacs, and A Amditis. Towards manoeuver negotiation: Autonet2030 project from a car maker perspective. *Transportation Research Procedia*, 14:2237–2244, 2016.

[21] H.J. Ferreau, C. Kirches, A. Potschka, H.G. Bock, and M. Diehl. qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363, 2014.

[22] Joao Salvado, Luís MM Custódio, and Daniel Heß. Contingency planning for automated vehicles. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 2853–2858, 2016.