SCISPACE
formerly Typeset

Open access • Proceedings Article • DOI:10.1109/ITSC.2016.7795555

Optimal trajectory planning for autonomous driving integrating logical constraints: An MIQP perspective — Source link

Xiangjun Qian, Florent Altché, Philipp Bender, Christoph Stiller ...+1 more authors

Institutions: PSL Research University, Karlsruhe Institute of Technology

Published on: 01 Nov 2016 - International Conference on Intelligent Transportation Systems

Topics: Trajectory optimization, Obstacle avoidance, Quadratic programming, Continuous optimization and Trajectory

Related papers:

- Trajectory planning for Bertha — A local, continuous method
- The combinatorial aspect of motion planning: Maneuver variants in structured environments
- A Review of Motion Planning Techniques for Automated Vehicles
- Mixed integer programming for multi-vehicle path planning
- A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles

Share this paper:

View more about this paper here: https://typeset.io/papers/optimal-trajectory-planning-for-autonomous-driving-7t0ub67qmo

Optimal Trajectory Planning for Autonomous Driving Integrating Logical Constraints: An MIQP Perspective

Xiangjun Qian, Florent Altché, Philipp Bender, Christoph Stiller, Arnaud de la Fortelle

▶ **To cite this version:**

Xiangjun Qian, Florent Altché, Philipp Bender, Christoph Stiller, Arnaud de la Fortelle. Optimal Trajectory Planning for Autonomous Driving Integrating Logical Constraints: An MIQP Perspective. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC 2016), Nov 2016, Rio de Janeiro, Brazil. hal-01342358v2

HAL Id: hal-01342358
https://hal.archives-ouvertes.fr/hal-01342358v2

Submitted on 30 Aug 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Optimal Trajectory Planning for Autonomous Driving Integrating Logical Constraints: An MIQP Perspective

Xiangjun Qian[1] Florent Altché[1,3] Philipp Bender[2] Christoph Stiller[2] Arnaud de La Fortelle[1]

***Abstract*— This paper considers the problem of optimal trajectory generation for autonomous driving under both continuous and logical constraints. Classical approaches based on continuous optimization formulate the trajectory generation problem as a nonlinear program, in which vehicle dynamics and obstacle avoidance requirements are enforced as nonlinear equality and inequality constraints. In general, gradient-based optimization methods are then used to find the optimal trajectory. However, these methods are ill-suited for logical constraints such as those raised by traffic rules, presence of obstacles and, more generally, to the existence of multiple maneuver variants. We propose a new formulation of the trajectory planning problem as a Mixed-Integer Quadratic Program. This formulation can be solved efficiently using widely available solvers, and the resulting trajectory is guaranteed to be globally optimal. We apply our framework to several scenarios that are still widely considered as challenging for autonomous driving, such as obstacle avoidance with multiple maneuver choices, overtaking with oncoming traffic or optimal lane-change decision making. Simulation results demonstrate the effectiveness of our approach and its real-time applicability.**

I. Introduction

Autonomous driving has been gaining impetus in the last few years, thanks to its foreseen potential of increasing traffic efficiency and reducing the number of road accidents. Recent studies suggest that up to 50% of vehicles may be automated by 2030 [1].

A challenging research task for autonomous driving is to generate optimal trajectories for certain criteria such as comfort or energy efficiency, while satisfying various constraints arising both from traffic rules and operational limits of the vehicles, some of them being highly specific to the context of driving. A large number of approaches for the *motion planning* or *trajectory planning* problem can be found in the literature [2], among which one important category is the sampling-based methods. These methods [3]–[5] typically generate a large set of candidate trajectories by deterministically or stochastically sampling a *state space*, before executing the best solution among those candidates. The optimality of the resulting trajectory relies on generating a high number of samples, which may not allow real-time applications.

[1]Xiangjun Qian, Florent Altché and Arnaud de La Fortelle are with MINES ParisTech, PSL - Research University, Centre for Robotics. {xiangjun.qian, florent.altche, arnaud.de_la_fortelle}@mines-paristech.fr

[2]Philipp Bender and Christoph Stiller are with Department of Measurement and Control, Karlsruhe Institute of Technology. {philipp.bender,stiller}@kit.edu

[3]Florent Altché is also with Ecole des Ponts ParisTech.

Recently, new approaches based on model predictive control (MPC) [6]–[8] have attracted increased attention due to their ability to systematically handle system constraints and quickly find the optimal trajectory. MPC-based approaches rely on iteratively solving a constrained, finite horizon optimal control problem generally using nonlinear optimization techniques. These approaches are intrinsically more efficient than sampling-based ones in finding optimal trajectories, as they exploit useful information like gradient fields to orient the search of optimal trajectories. Real-time implementations of these methods on actual vehicles can be found, *e.g.*, in [6], [7], [9].

Even though MPC-based methods have yielded good results for problems with continuous constraints, we argue that there are constraints that are intrinsically *logical*, in that they are naturally formulated as logical propositions. First, on-road driving is by nature highly constrained by traffic rules and expected driving behaviors, which can be represented naturally as logical propositions, for instance:

(R.1) A vehicle must slow down when approaching a speed bump.

(R.2) A vehicle can only overtake from the left side.

(R.3) A vehicle needs to be on the exit lane before exiting on the highway.

Moreover, driving generally involves discrete decisions among multiple maneuver variants, which makes trajectory planning combinatorial in nature [10], [11]. Indeed, it has been shown that each maneuver variant can be mapped to a unique homotopy class of trajectories [10], [11]. It is generally straightforward to describe these homotopy classes using logical propositions. For example, an obstacle in the middle of the road brings two distinct maneuver choices, as follows:

(R.4) A vehicle should avoid an obstacle, either in a clockwise or in a counterclockwise manner.

By nature, sampling-based approaches can accommodate logical constraints. However, as stated earlier, these methods can yield sub-optimal trajectories even with a large number of samples. Most current MPC-based approaches are ill-suited to take these logical rules into account, as most of them require all constraints to be continuous and differentiable. Moreover, most continuous solvers use local, gradient-based optimization methods which can be trapped in a local optimum inside a certain homotopy class. Some heuristics have been proposed to approximate logical constraints by nonlinear functions [7], [9], [10] and to initialize the algorithm in a homotopy class that is likely to be the best one [7], but they provide no guarantee regarding the global optimality

of the generated trajectory.

A possible way of handling logical constraints is to formulate the problem as a Mixed-Integer Programming (MIP) problem. General MIP problems are hard to solve, but efficient algorithms exist for special instances of MIP, notably Mixed-Integer Linear Programming (MILP) and Mixed-Integer Quadratic Programming (MIQP). Both have successfully been used for Unmanned Aerial Vehicle (UAV) trajectory planning [12], [13] and multi-vehicle collision avoidance problem [14]. MILP methods have also been used to design automatic lane change controllers that are proposed in [15], [16], but they require a simplified vehicle model. To the best of our knowledge, an efficient MIP formulation using realistic vehicle dynamics and capable of handling multiple on-road driving scenarios is not yet available.

The major contribution of this paper is to propose a novel approach for the trajectory generation problem using Mixed Integer Quadratic Programming. This approach allows to seamlessly treat continuous and logical constraints. Furthermore, we apply our framework to several situations which are still widely recognized as challenging for autonomous driving, for instance obstacle avoidance, overtaking in presence of oncoming traffic or optimal lane-change decision making. We demonstrate that our method can intuitively handle these problems and generate globally optimal trajectories.

II. Optimal Trajectory Generation as MIQP

This section proposes an MIQP formulation for the problem of generating optimal trajectories for autonomous driving. We consider that the ego vehicle is driving on a road with lane markings. The ego vehicle uses a Model Predictive Control (MPC) scheme to perform trajectory planning, in which the planned trajectory is updated in successive stages. However, in this paper, we only discuss in detail a single MPC stage. We assume that the road curvature is sufficiently small to consider the road as straight within the horizon of one MPC stage, so that we can model the road in a Cartesian frame (x, y) with x the longitudinal direction of the road and y the lateral direction.

A. Vehicle dynamics and constraints

The ego vehicle is modeled as a third-order point-mass system, with state $\mathbf{x}$ and control vector $\mathbf{u}$ defined as

$$\mathbf{x}(t) = [x(t), v_x(t), a_x(t), y(t), v_y(t), a_y(t)]^T,$$
$$\mathbf{u}(t) = [j_x(t), j_y(t)]^T$$

where x and y denote longitudinal and lateral position in the inertial frame, v the speed and a the acceleration, with subscript x or y respectively indicating their longitudinal and lateral components. The vehicle dynamics are ruled by the following differential equation:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} A & \mathbf{0} \\ \mathbf{0} & A \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} B & \mathbf{0} \\ \mathbf{0} & B \end{bmatrix} \mathbf{u}(t),$$
$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \tag{1}$$

where $\mathbf{0}$ denotes zero matrices of proper dimensions.

For practical purposes, we assume that the control $\mathbf{u}$ is a piecewise constant function with a time step of τ. The discretized vehicle dynamics can then be approximated as

$$\mathbf{x}(k+1) = \begin{bmatrix} A^d & \mathbf{0} \\ \mathbf{0} & A^d \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} B^d & \mathbf{0} \\ \mathbf{0} & B^d \end{bmatrix} \mathbf{u}(k),$$
$$A^d = \begin{bmatrix} 1 & \tau & \frac{1}{2}\tau^2 \\ 0 & 1 & \tau \\ 0 & 0 & 1 \end{bmatrix}, \quad B^d = \begin{bmatrix} \frac{1}{6}\tau^3 \\ \frac{1}{2}\tau^2 \\ \tau \end{bmatrix}. \tag{2}$$

where $\mathbf{x}(k)$ and $\mathbf{u}(k)$ are respectively the state and the control of the vehicle at the beginning of the time interval $[k\tau, (k+1)\tau)$.

Remark that some previous work [8], [17] has used a first or second order point-mass description. Here, we use a third-order model to ensure that the second derivative of the generated trajectory is continuous even with a piecewise constant input. As a result, the yaw rate of the planned trajectory is continuous, and the vehicle can thus smoothly track the trajectory.

Due to the dynamic limitations of the vehicle, bound constraints are enforced on the state and control signals as

$$\mathbf{x} \in [\underline{\mathbf{x}}, \overline{\mathbf{x}}], \mathbf{u} \in [\underline{\mathbf{u}}, \overline{\mathbf{u}}], \tag{3}$$

with the bounds defined as

$$\underline{\mathbf{x}} = [0, 0, \underline{a}_x, \underline{y}, \underline{v}_y, \underline{a}_y]^T, \overline{\mathbf{x}} = [\text{free}, \overline{v}_x, \overline{a}_x, \overline{y}, \overline{v}_y, \overline{a}_y]^T, \tag{4a}$$
$$\underline{\mathbf{u}} = [\underline{j}_x, \underline{j}_y]^T, \overline{\mathbf{u}} = [\overline{j}_x, \overline{j}_y]^T. \tag{4b}$$

The above formulation does not consider the nonholonomic constraints of the vehicle, and the longitudinal and lateral dynamics are fully decoupled. The exact coupling between these dynamics involves nonlinear relations [7]; therefore, we approximate it by two additional constraints which ensure that the generated trajectory is dynamically feasible.

The vehicle heading θ can be reconstructed from the state $\mathbf{x}$ as $\theta = \arctan(v_y/v_x)$. We model the coupling of longitudinal and lateral dynamics by enforcing condition $\theta \in [\underline{\theta}, \overline{\theta}]$, with

$$v_y \in [v_x \tan(\underline{\theta}), v_x \tan(\overline{\theta})]. \tag{5}$$

B. Logical constraints for driving

Winston et al [18] have shown that propositional logic can be further reformulated as a set of linear inequalities with integer variables. Here, we briefly present this approach in the context of autonomous driving.

We define a *literal* as an atomic statement corresponding to a linear mathematical condition on one of the state variables, for instance: *the longitudinal position of the vehicle is larger than* $30\,\mathrm{m}$. Literals can be combined using connectors, namely $\wedge$ (and), $\vee$ (or), $\neg$ (negation); implications ($\Rightarrow$) and equivalences ($\Leftrightarrow$) can be formed using the first three connectors.

To illustrate, we consider a speed bump covering the range of longitudinal positions $x \in [30\,\mathrm{m}, 50\,\mathrm{m}]$, with a speed limit of $10\,\mathrm{m/s}$ in this range. Let us define three literals $P_1 =$

$[x(k) \geq 30]$, $P_2 = [x(k) \leq 50]$ and $P_3 = [v_x(k) \leq 10]$; rule (R.1) can then be expressed in the form: $\forall k \geq 0, (P_1 \wedge P_2) \Rightarrow P_3$, *i.e.* if $30 \leq x(k) \leq 50$, then the vehicle's longitudinal speed should be lower than or equal to 10 m/s.

The so-called Big-M method (see [18]) allows to force a binary variable $\delta(k)$ to be equal to 1 when a given literal is true, and equal to 0 when the literal is false, using only linear inequalities by introducing a large constant M. In the previous example, we can let $\delta_i(k) = 1 \Leftrightarrow P_i$ for $i \in \{1, 2, 3\}$ so that (R.1) can be expressed equivalently as, $\forall k \geq 0$,

$$\begin{aligned} &\delta_1(k) = 1 \Leftrightarrow x(k) \geq 30, \\ &\delta_2(k) = 1 \Leftrightarrow x(k) \leq 50, \\ &\delta_3(k) = 1 \Leftrightarrow v_x(k) \leq 10, \\ &-\delta_1(k) + \delta_3(k) \leq 0, \\ &-\delta_2(k) + \delta_3(k) \leq 0, \\ &\delta_1(k) + \delta_2(k) - \delta_3(k) \leq 1. \end{aligned} \tag{6}$$

Note that these conditions could also be approximated using nonlinear constraints in a continuous solver. However, current state-of-the-art solvers for general nonlinear optimization problems can only find local optima, whereas our mixed-integer formulation ensures global optimality.

C. MIQP formulation

We introduce a new variable $\boldsymbol{\delta}$ such that $\boldsymbol{\delta}(k) = \{0, 1\}^m$ is a collection of all binary variables resulting from the reformulation of relevant literals as mixed integer linear inequalities. Let $\mathbf{x}_r$ be the reference trajectory for the vehicle, which can be time-dependent, state-dependent, or dependent on propositions. We also introduce $\boldsymbol{\delta}_r$ as the reference trajectory for the binary variables, so that we can also express preferences on some binary states, for instance to specify a preferred lane in a multi-lane road. Let $K = T/\tau$ be the number of time steps in the prediction horizon. The optimization problem can now be formulated, in its generic form, as

$$\begin{aligned} \min_{\mathbf{u},\boldsymbol{\delta}} \sum_{k=0}^{K} &||\mathbf{x}(k) - \mathbf{x}_r(k)||_Q^2 \\ &+ ||\boldsymbol{\delta}(k) - \boldsymbol{\delta}_r(k)||_S^2 + ||\mathbf{u}(k)||_R^2, \end{aligned} \tag{7}$$

subject to

$$\text{known } \mathbf{x}(0), \tag{8a}$$

$$\text{vehicle dynamics (2)}, \tag{8b}$$

$$\text{dynamic constraints (3) and (5)}, \tag{8c}$$

$$C \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_r \\ \boldsymbol{\delta} \end{bmatrix} \leq D. \tag{8d}$$

where Q, S, R are positive weighting matrices of proper dimensions. Constraint (8d) is the set of all linear inequalities written in matrix form with two matrices C and D of proper dimensions, incorporating all mixed-integer constraints used to enforce driving rules.

$\underline{\mathbf{x}} = [0, 0, -4\,\mathrm{m/s^2}, 0, -2\,\mathrm{m/s}, -1\,\mathrm{m/s^2}]^T$,
$\overline{\mathbf{x}} = [\text{free}, 20\,\mathrm{m/s}, 3\,\mathrm{m/s^2}, 5\,\mathrm{m}, 2\,\mathrm{m/s}, 1\,\mathrm{m/s^2}]^T$,
$\underline{\mathbf{u}} = [-3\,\mathrm{m/s^3}, -2\,\mathrm{m/s^3}]^T, \overline{\mathbf{u}} = [3\,\mathrm{m/s^3}, 2\,\mathrm{m/s^3}]^T$,
$\underline{\theta} = -0.4\,\mathrm{rad}, \overline{\theta} = 0.4\,\mathrm{rad}, \underline{\omega} = -0.26\,\mathrm{rad/s}, \overline{\omega} = 0.26\,\mathrm{rad/s}$,
$q_1 = 1, q_2 = 2, q_3 = 1, q_4 = 2, q_5 = 4, r_1 = 4, r_2 = 4.$

TABLE I: Parameters used for case study

The cost function (7) is quadratic if $\mathbf{x}_r$ and $\boldsymbol{\delta}_r$ are independent from other variables or are linearly dependent on other variables. The constraints (8) are linear. Therefore, the above optimization problem is an instance of a mixed-integer quadratic program (MIQP). Exact resolution algorithms are known to solve such problems, leading to a globally optimal trajectory which can then be fed to low-level controller charged of tracking this trajectory.

Remark 1: The proposed MIQP formulation is able to cover a vast majority of on-road driving scenarios. For example, Rules (R.1)-(R.4) can be naturally incorporated in our framework. If we must consider non-quadratic cost function, nonlinear vehicle dynamics or constraints, we can formulate a Mixed Integer Nonlinear Program, which, however, is much harder to solve.

III. Case study

The previous section proposes an MIQP framework that integrates the continuous dynamic of the vehicle and propositional logic. In this section, we explain how this framework can be used to effectively handle various on-road driving scenarios. Note that, although the major purpose of this section is to demonstrate the universality of the framework, the detailed strategies for different driving scenarios are by themselves contributions to some challenging problems for on-road autonomous driving.

Throughout this section, we use the following cost function

$$\begin{aligned} J = \sum_{k=0}^{K} &q_1(v_x(k) - v_r)^2 + q_2 a_x^2(k) + q_3(y(k) - y_r(k))^2 \\ &+ q_4 v_y^2(k) + q_5 a_y^2(k) + r_1 j_x^2(k) + r_2 j_y^2(k), \end{aligned} \tag{9}$$

such that the vehicle tracks a constant desired speed profile and a potentially time-varying desired lateral deviation, while trying to minimize the control effort. We do not assume any desired binary state, and therefore the term $||\boldsymbol{\delta}(k) - \boldsymbol{\delta}_r(k)||_S^2$ in the generic formulation is ignored.

In all cases, the ego vehicle is assumed to start at position $(0, 2.5)$ in the rightmost lane. The initial and desired speeds are equal, with $v_x(0) = v_r = 15\,\mathrm{m/s}$. The width of each lane is 5 m. Other parameters that are common in all cases are recapitulated in Table I. Scenario-specific parameters will be presented respectively in each case study.

We use the commercial solver Gurobi [19] to compute solutions to our MIQP formulation. Simulation codes are written in Python, and experiments are performed on a laptop with Intel Core i5-5300U CPU clocked at 2.30GHz with 8GB RAM.

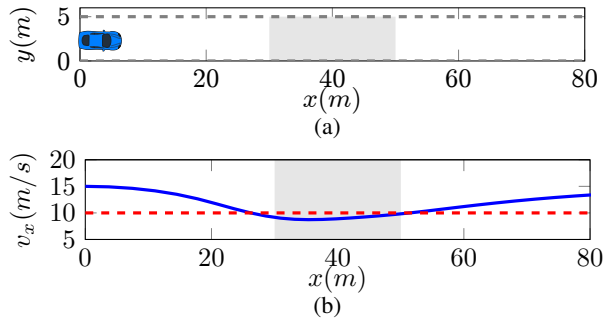

Fig. 1: Speed bump scenario: (a) illustration of the scenario, (b) longitudinal speed profile w.r.t. the longitudinal offset.

A. Speed bump

The first case study considers the speed bump scenario (Fig. 1a) that is used as an example in II-B.

The speed bump conditions are given in (6). In the simulation, we use a prediction horizon $T = 5\,\mathrm{s}$ and $\tau = 0.25\,\mathrm{s}$. Fig. 1b illustrates the longitudinal speed profile of the planned trajectory with respect to the traveled distance. We observe that vehicles effectively reduces its speed to less than $10\,\mathrm{m/s}$ within the interval of $[30, 50]$.

B. Obstacle avoidance

We now consider an obstacle avoidance scenario during on-road driving. The irregular shapes of obstacles are approximated using minimal bounding rectangles. A more complex polygonal modeling is also possible, at the cost of increased computational complexity. For an obstacle ν with bounding rectangle $[x^\nu(k) - L^\nu, x^\nu(k) + L^\nu] \times [y^\nu(k) - W^\nu, y^\nu(k) + W^\nu]$, the set of constraints for collision avoidance is then given as $\forall k \geq 0$,

$$\delta_1^\nu(k) = 1 \Leftrightarrow x(k) \leq x^\nu(k) - L^\nu, \tag{10a}$$
$$\delta_2^\nu(k) = 1 \Leftrightarrow x(k) \geq x^\nu(k) + L^\nu, \tag{10b}$$
$$\delta_3^\nu(k) = 1 \Leftrightarrow x(k) \leq y^\nu(k) - W^\nu, \tag{10c}$$
$$\delta_4^\nu(k) = 1 \Leftrightarrow x(k) \geq y^\nu(k) + W^\nu, \tag{10d}$$
$$\delta_1^\nu(k) + \delta_2^\nu(k) + \delta_3^\nu(k) + \delta_4^\nu(k) = 1. \tag{10e}$$

Note that the formulation allows both moving and still obstacles. The conditions (10) state that the vehicle must be separated from the obstacle, either by a longitudinal distance L^ν or laterally by W^ν.

For illustration, we consider two identical obstacles centered at $(80, 1.5)$ and $(160, 3.5)$. Parameters are $T = 15\,\mathrm{s}$, $\tau = 1\,\mathrm{s}$, $L^\nu = 10\,\mathrm{m}$ and $W^\nu = 2\,\mathrm{m}$. The trajectory labeled "opt" in Fig. 2a is the global optimum found by the MIQP planner; for comparison purposes, we also plot all locally optimal trajectories "nopt-1,2,3" with their respective costs. We observe that our method can effectively find the globally optimal trajectory. Fig. 2b and 2c respectively illustrate the longitudinal and lateral speed and acceleration profiles of the ego vehicle.

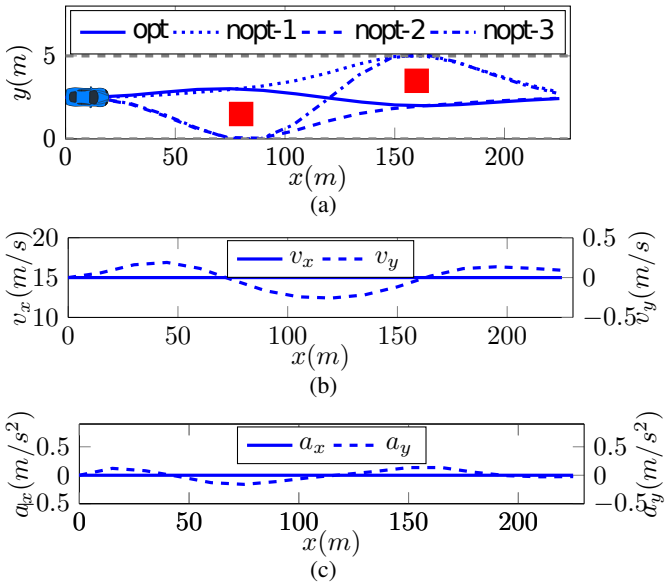

Fig. 2: Obstacle avoidance scenario: (a) the globally optimal trajectory and locally optimal trajectories (respective costs: opt 3.16, nopt-1 36.5, nopt-2 37.2, nopt-3 84.67), (b) speed profiles, (c) acceleration profiles. Obstacles are marked in red.

C. Overtaking in a two-lane road

This case study considers an overtaking scenario on a two-lane road with oncoming traffic. Fig. 3a shows the initial configurations of nearby vehicles, all driving with a constant speed of $10\,\mathrm{m/s}$. This scenario is considered as difficult for both human drivers and autonomous vehicles [8], [17]. Reference [10] shows the existence of multiple homotopy classes in this scenario and proposes to exhaustively search for the globally optimal solution. Here, we show that we can find the globally optimal solution without explicitly enumerating all homotopy classes.

It is possible to model surrounding vehicles as rectangles as in the previous case study, thus requiring four integer variables for each vehicle and each time step k. However, by introducing the so called ramp barrier [8], [17], the problem can be further simplified by approximating the rectangular obstacle region by a triangle only using two linear constraints as shown in Fig. 3a.

Let ν be a surrounding vehicle; if ν is in the same lane as the ego vehicle, the constraints are given as

$$\delta^\nu(k) = 0 \Leftrightarrow -\frac{x(k) - x^\nu(k)}{L^\nu} + \frac{y(k) - y^\nu(k)}{W^\nu} \geq 1, \tag{11a}$$
$$\delta^\nu(k) = 1 \Leftrightarrow \frac{x(k) - x^\nu(k)}{L^\nu} + \frac{y(k) - y^\nu(k)}{W^\nu} \geq 1, \tag{11b}$$

where L^ν and W^ν are minimal longitudinal and lateral separations during lane change. Similarly, the constraints for an oncoming vehicle ν can be modeled as

$$\delta^\nu(k) = 0 \Leftrightarrow \frac{x(k) - x^\nu(k)}{L^\nu} + \frac{y(k) - y^\nu(k)}{W^\nu} \leq -1, \tag{12a}$$
$$\delta^\nu(k) = 1 \Leftrightarrow -\frac{x(k) - x^\nu(k)}{L^\nu} + \frac{y(k) - y^\nu(k)}{W^\nu} \leq -1. \tag{12b}$$

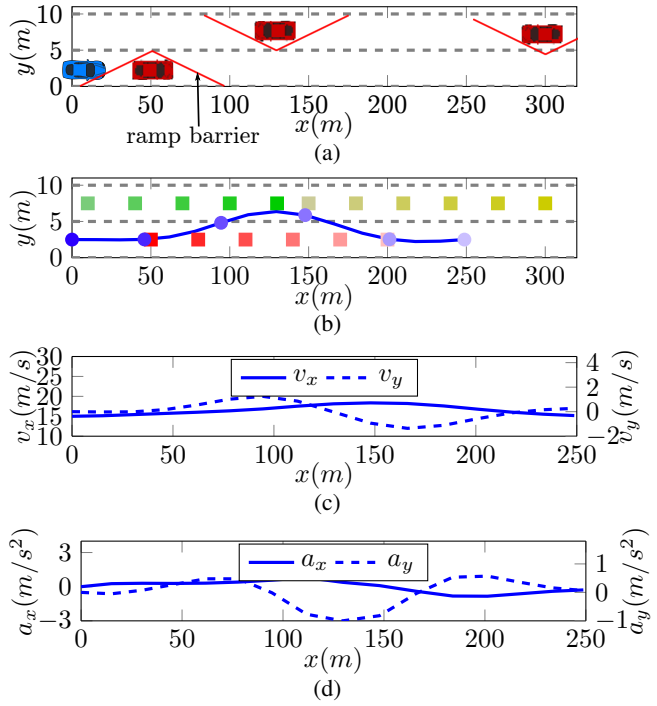

Fig. 3: Overtaking scenario: (a) illustration of the scenario and the ramp barrier methods, (b) the overtaking trajectory, (c) speed profiles, (d) acceleration profiles. The position of the ego vehicle is shown as blue dots, and rectangles with different colors mark the positions of surrounding vehicles. Time dimension is color-coded, with lighter colors corresponding to instants further away in time.

Using this formulation, only one integer variable δ^ν is required per vehicle and per time step.

In the simulation, we adopt a prediction horizon of $T = 15\,\mathrm{s}$ and we let $\tau = 1\,\mathrm{s}$. To allow the ego vehicle to temporarily cross the lane border, the upper limit $\overline{y}$ is relaxed. Fig. 3b illustrates the trajectory of overtaking. We observe that the ego vehicle decides to accelerate slightly so that it can use the space between the first oncoming vehicle and the second oncoming vehicle to perform the overtaking.

For comparison purposes, we reduce the penalty on speed deviation q_1 to 0.5 in the cost function (7). The resulting trajectory is shown in Fig. 4: in this case, the ego vehicle chooses not to overtake, as the cost of this maneuver is higher than that of following the slower car, due to the small penalty on the speed deviation. This demonstrates the flexibility of the MIQP formulation: different driving styles can be configured simply by modulating the weighting terms.

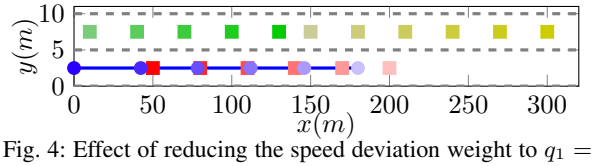

Fig. 4: Effect of reducing the speed deviation weight to $q_1 = 0.5$ in the overtaking scenario. The position of the ego vehicle is shown as blue dots, and rectangles with different colors mark the positions of surrounding vehicles. Time dimension is color-coded, with lighter colors corresponding to instants further away in time.

D. Lane change

This last case study considers the problem of decision making and trajectory generation for a lane change maneuver: the ego vehicle must decide the objective lane as well as the optimal trajectory to reach this lane, without colliding with surrounding vehicles. The complexity of this problem lies in the multiple discrete choices raised from multiple lanes and multiple vehicles on each lane. References [15], [16] have considered this problem using MILP formulations; however, their modeling cannot ensure that trajectories are dynamically feasible due to important simplifications of the vehicle dynamics.

We consider a road with N lanes, labeled by $\gamma \in \{1, ..., N\}$. We introduce a binary variable $\delta^\gamma(k)$ that equals 1 if the ego vehicle is on lane γ at time step k. Let V be the set of surrounding vehicles and V^γ be the set of vehicles inside lane γ. We introduce the following logical constraint: $\forall k > 0$,

$$\delta^\gamma(k) = 1 \Leftrightarrow \left(y_r(k) = y^\gamma \wedge y(k) \in [\underline{y}^\gamma, \overline{y}^\gamma]\right), \tag{13}$$

such that, if the ego vehicle is in lane γ, then the vehicle should be within the boundary of lane γ and the reference centerline should be set to the centerline of the lane.

Moreover, we add the following collision avoidance constraints: $\forall k \geq 0$,

$$\begin{aligned}\delta^\gamma(k) = 1 \Leftrightarrow \forall \nu \in V^\gamma,\ \delta^\nu(k) = 1 \Leftrightarrow x(k) \leq x^\nu(k) - L^\nu,\\ \delta^\nu(k) = 0 \Leftrightarrow x(k) \geq x^\nu(k) + L^\nu,\end{aligned} \tag{14}$$

such that the ego vehicle must avoid collisions with all the vehicles in lane γ.

The ego vehicle is only allowed to be in one lane at any given time, thus we add the following constraint: $\forall k \geq 0$,

$$\sum_{\gamma=1}^{N} \delta^\gamma(k) = 1. \tag{15}$$

Fig. 5a shows a highway with three lanes. The ego vehicle starts in the rightmost lane with a speed of $15\,\mathrm{m/s}$. Surrounding vehicles are distributed over three lanes. The vehicle on the leftmost lane drives at a speed of $15\,\mathrm{m/s}$ while other surrounding vehicles drive at a speed of $10\,\mathrm{m/s}$. Constraints (13), (14) and (15) are enforced along with other constraints on the formulated MIQP problem. The horizon is set to $T = 15\,\mathrm{s}$ and the time step $\tau = 1\,\mathrm{s}$. The constraint on lane boundary is temporarily deactivated. We observe in Fig. 5b that the ego vehicle chooses the left-most lane as the objective lane and plans a dynamically feasible and collision-free trajectory to reach the lane within the prediction horizon.

Table II presents the computation times for the four case studies, demonstrating the real-time capability of the proposed formulation; future work will focus on further reducing this computation time.

IV. Discussion and Conclusion

We have addressed the problem of optimal trajectory generation integrating both continuous and logical constraints

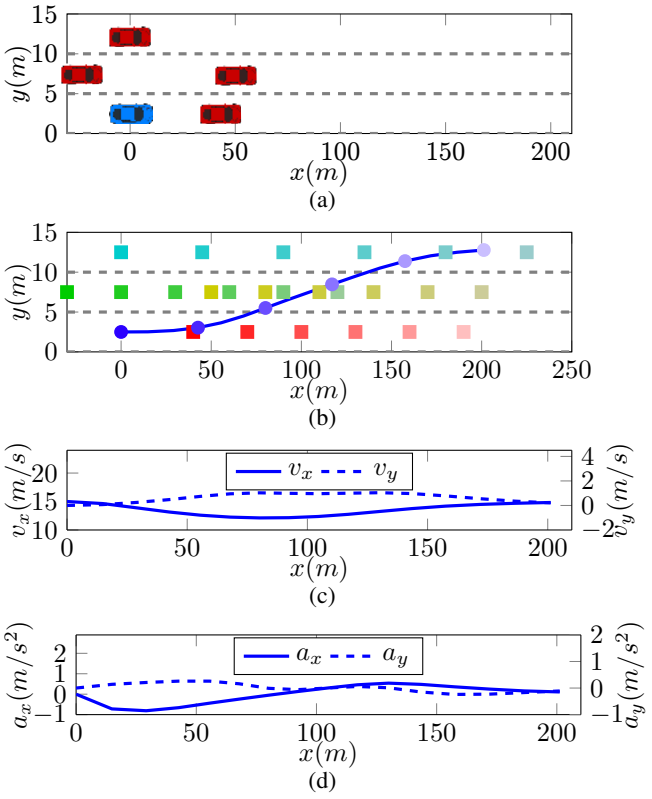

Fig. 5: Lane change scenario: (a) illustration of the scenario, (b) trajectory of the ego vehicle and the surrounding vehicles, (c) speed profiles, (d) acceleration profiles. The position of the ego vehicle is shown as blue dots, and rectangles with different colors mark the positions of surrounding vehicles. Time dimension is color-coded, with lighter colors corresponding to instants further away in time.

Speed bump	Obstacle avoidance	Overtaking	Lane change
17 ms	73 ms	81 ms	228 ms

TABLE II: Time statistics

using an MIQP based approach. This approach is well-suited to the on-road autonomous driving environment as logical constraints can arise from different aspects of driving: traffic rules, on-road obstacles and the existence of multiple maneuver variants. We have showcased the universality and the efficiency of the method by applying it to various challenging driving scenarios.

This paper uses a third-order linear vehicle model for trajectory generation. This model is suitable if the longitudinal motion dominates the lateral one, for example when driving on highways or on urban arterial roads. However, for application to low-speed driving, a nonlinear vehicle model might be more desirable. Future work will investigate the applicability of combining the feedback linearization with the MIQP formulation. Another important assumption in this paper is that the vehicle is driving on road segments with small curvature. For large road curvatures, the current model can be imprecise. Future work will investigate this issue.

In the future, we will also integrate the proposed method into a receding horizon framework so that the ego vehicle re-plans regularly to incorporate new information. The algorithm will be implemented in the AnnieWay [20] autonomous vehicle to study the performance of our approach in a real-world setting.

References

[1] A. de La Fortelle, X. Qian, S. Diemer, F. Moutarde, and S. Bonnabel, "Network of Automated Vehicles: the AutoNet2030 vision," in *ITS World Congress*, 2014.

[2] B. Paden, M. Cap, S. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control," *IEEE Transactions on Intelligent Vehicles*, vol. PP, no. 99, pp. 1–1, 2016.

[3] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. P. How, and G. Fiore, "Real-time motion planning with applications to autonomous urban driving," *Control Systems Technology, IEEE Transactions on*, vol. 17, no. 5, pp. 1105–1118, 2009.

[4] M. Werling, S. Kammel, J. Ziegler, and L. Groll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *The International Journal of Robotics Research*, vol. 31, no. September 2015, pp. 346–359, 2012.

[5] T. Gu, J. Snider, J. M. Dolan, and J.-W. Lee, "Focused trajectory planning for autonomous on-road driving," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pp. 547–552, IEEE, 2013.

[6] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for bertha; a local, continuous method," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 450–457, June 2014.

[7] Y. Gao, *Model Predictive Control for Autonomous and Semiautonomous Vehicles*. PhD thesis, University of California, Berkeley, 2014.

[8] J. Nilsson, P. Falcone, M. Ali, and J. Sjöberg, "Receding horizon maneuver generation for automated highway driving," *Control Engineering Practice*, vol. 41, pp. 124–133, 2015.

[9] T. Weiskircher and B. Ayalew, "Predictive adas: A predictive trajectory guidance scheme for advanced driver assistance in public traffic," in *Control Conference (ECC), 2015 European*, pp. 3402–3407, July 2015.

[10] P. Bender, O. S. Tas, J. Ziegler, and C. Stiller, "The combinatorial aspect of motion planning: Maneuver variants in structured environments," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pp. 1386–1392, IEEE, 2015.

[11] Ö. Ş. Taş, "Integrating Combinatorial Reasoning and Continuous Methods for Optimal Motion Planning of Autonomous Vehicles," Master's thesis, Karlsruhe Institute of Technology, Germany, September 2014.

[12] T. Schouwenaars, É. Féron, and J. How, "Safe receding horizon path planning for autonomous vehicles," in *Proceedings of the Annual Allerton Conference on Communication, Control and Computing*, vol. 40, pp. 295–304, The University; 1998, 2002.

[13] J. Bellingham, A. Richards, and J. P. How, "Receding horizon control of autonomous aerial vehicles," in *American Control Conference, 2002. Proceedings of the 2002*, vol. 5, pp. 3741–3746, IEEE, 2002.

[14] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat, "Mpc-based approach to active steering for autonomous vehicle systems," *International Journal of Vehicle Autonomous Systems*, vol. 3, no. 2-4, pp. 265–291, 2005.

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

[16] Y. Du, Y. Wang, and C.-Y. Chan, "Autonomous lane-change controller via mixed logical dynamical," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pp. 1154–1159, IEEE, 2014.

[17] N. Murgovski, J. Sj, *et al.*, "Predictive cruise control with autonomous overtaking," in *2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 644–649, IEEE, 2015.

[18] W. L. Winston, M. Venkataramanan, and J. B. Goldberg, *Introduction to mathematical programming*, vol. 1. Thomson/Brooks/Cole, 2003.

[19] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2015.

[20] S. Kammel, J. Ziegler, B. Pitzer, M. Werling, T. Gindele, D. Jagzent, J. Schröder, M. Thuy, M. Goebl, F. v. Hundelshausen, *et al.*, "Team annieway's autonomous system for the 2007 darpa urban challenge," *Journal of Field Robotics*, vol. 25, no. 9, pp. 615–639, 2008.