Minimizing Long Vehicles Overhang Exceeding the Drivable Surface via Convex Path Optimization

Conference Paper · October 2017

DOI: 10.1109/ITSC.2017.8317754

4 authors, including:

Pedro Lima
Scania
25 PUBLICATIONS **232** CITATIONS
SEE PROFILE

Rui Oliveira
KTH Royal Institute of Technology
11 PUBLICATIONS **69** CITATIONS
SEE PROFILE

Jonas Mårtensson
KTH Royal Institute of Technology
118 PUBLICATIONS **1,766** CITATIONS
SEE PROFILE

Minimizing Long Vehicles Overhang Exceeding the Drivable Surface via Convex Path Optimization

Pedro F. Lima[1], Rui Oliveira[2], Jonas Mårtensson[1], and Bo Wahlberg[1] .

***Abstract*— This paper presents a novel path planning algorithm for on-road autonomous driving. The algorithm targets long and wide vehicles, in which the overhangs (*i.e.*, the vehicle chassis extending beyond the front and rear wheelbase) can endanger other vehicles, pedestrians, or even the vehicle itself. The vehicle motion is described in a road-aligned coordinate frame. A novel method for computing the vehicle limits is proposed guaranteeing feasibility of the planned path when converted back into the original coordinate frame. The algorithm is posed as a convex optimization that takes into account the exact dimensions of the vehicle and the road, while minimizing the amount of overhang outside of the drivable surface.**

The results of the proposed algorithm are compared in a simulation of a real road scenario against a centerline tracking scheme. The results show a significant decrease on the amount of overhang area outside of the drivable surface, leading to an increased safety in driving maneuvers. The real-time applicability of the method is shown, by using it in a receding-horizon framework.

Fig. 1. Modified Scania bus used as experimental and research platform.

I. Introduction

A. Background and motivation

Public transportation in urban cities is gaining more attention everyday, however, the barometer [1] performed by the European Commission showed that only 16% of of EU27 citizens use public transport as their main mode of transport. The main reasons for the low usage of public transport range from high ticket cost to irregular travel times. One of the main public transportation modes, the bus, could greatly benefit from automated solutions. Autonomous vehicles achieve better fuel consumption, which leads to reduced expenses. In the case of an autonomous bus fleet, the expenses related to the driver staff would also vanish, and running times would be more predictable. These factors would greatly reduce the final cost, making public transportation a more attractive option.

Buses are often characterized by their big dimensions, being both long and wide (see Fig. 1). These characteristics seek to optimize the fuel consumption and driver staff cost, per passenger. However, this hardens the driving task, as it is difficult to maneuver the vehicle in tight roads. This difficulty also reflects itself on automated path planning algorithms, which are often not able to deal with scenarios considering large vehicles in roads with small clearance. In some situations, it is even impossible for a bus to travel completely within its own lane. Consequently, these vehicles are typically forced to perform maneuvers in which parts of the vehicle overhang go outside of the drivable surface. This is particularly noticeable in vehicles with lengthy overhangs. The overhang is defined as the length of a vehicle, in both the front and rear directions, which exceeds its wheelbase, and it can reach several meters in the case of buses.

Current automated motion planning algorithms often have hard constraints that do not allow any part of the vehicle chassis to leave its assigned driving corridor. Thus, it is necessary to relax these constraints in order to find feasible traveling paths. However, this must be done in a way that minimizes the vehicle area that exits the lane while still ensuring that the paths are collision free with other existing road users.

B. Main contributions

In this paper, we extend the work presented in [2]. There, a sequential convex programming (SCP) algorithm is proposed to address the problem of spatial-based path planning under actuator, obstacle avoidance, and vehicle dimension constraints. The main benefit of the proposed approach is the ability of posing the problem in a convex optimization format, even when considering the vehicle dimension constraints, which maximizes the path planning performance in very constrained environments. The vehicle is modeled in the space-domain and in a road-aligned frame along the reference path to exclude time and speed from the equations of the vehicle dynamics. Also, it allows the formulation of

[1]Integrated Transport Research Lab and ACCESS Linnaeus Centre, Department of Automatic Control, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden. pfrdal@kth.se, rfoli@kth.se, jonas1@kth.se, bo@kth.se

[2]Research and Development, Scania CV AB, 151 87 Södertälje, Sweden. rui.oliveira@scania.com

This work has been partially funded by the Swedish government and automotive industry within the FFI program - Strategic Vehicle Research and Innovation under the project iQMatic 2012-04626 and by the Wallenberg Autonomous Systems and Software Program (WASP).

We would like to thank George Dibben, a development engineer at Scania CV AB, for the valuable insight about long vehicles maneuvering.

978-1-5386-1526-3/17/$31.00 ©2017 IEEE

obstacle constraints as simple state bounds. However, in [2], the proposed methodology does not deal with the fact that, in the road-aligned coordinate frame, long vehicles become extremely distorted, which makes the proposed computation of vehicle boundaries unusable, as it poorly describes the vehicle dimension constraints.

The main contributions presented in this paper are:

1) a novel method to compute the vehicle dimension constraints taking into account the distortion caused by the coordinate frame transformation;
2) the vehicle modeling into three distinct parts: rear overhang, between wheels, and front overhang. This modeling allows the formulation of hard constraints to ensure that the vehicle wheels are inside the drivable surface, and soft constraints to minimize the amount of front/rear overhang outside the drivable surface;
3) the feasibility of applying the proposed path optimization method in real-time using it in a receding-horizon framework;
4) the significant decrease on the amount of overhang area outside of the drivable surface, leading to an increased safety in driving maneuvers, when compared to a centerline tracking scheme in a real driving scenario.

In this paper, we address on-road path optimization situations, where there is a known drivable surface, normally the area between two lane borders, and a known traversable surface, typically outside the lane boundaries (e.g., adjacent lanes or a low sidewalk), where the overhang can be without risking safety. Moreover, in [2], the proposed convex program is sequentially solved until a certain termination criterion is met. There, obstacle avoidance could require large steering maneuvers, which could affect the vehicle dynamics and the vehicle's dimension constraints linearization due to their dependence on the underlying reference trajectories and input. In the case of on-road planning, there is no need for sequentially solving the optimization problem, as the planning can be seen as an optimal refinement of the road centerline.

C. Literature review

To the best knowledge of the authors, there are currently no proposed path planning solutions that address the problem of minimizing the amount of overhang area outside the road lane during long-vehicle maneuvers. However, in this section, we perform a brief literature review of path planning methods (see [3] and references therein) comparing them to our proposed approach.

Graph-search approaches (e.g., [4]–[9]) are common in practical applications, as they yield real-time algorithms for discrete state spaces. To satisfy the non-holonomic constraints of the vehicle, those algorithms can be based on an offline created lattice, which is used as a search graph to span over the drivable environment. However, the search graph only represents a discretized version of the drivable environment and so, those algorithms tend to produce paths that are non-smooth.

Alternatively, incremental tree-based approaches (e.g., [10]–[13]) ensure kinematic feasibility of the path in a forward search in continuous coordinates. However, practical implementations usually require an efficient choice of heuristics, which can be scenario-dependent. Some of these methods are also known for their sub-optimality and randomness.

In this paper, the preferred approach is to directly formulate the path-planning problem as an optimization problem (e.g., [14]–[22]). Optimization-based approaches have recently gained attention, due to the ever-increasing computing power and broader offer of numerical optimization tools. These approaches are attractive due to the possibility of explicitly enforcing vehicle constraints and easily defining optimization objectives. The main challenge with this type of approaches is to guarantee real-time performance and the convergence to a global minimum. Typically, that is not possible when formulating the problem in a nonlinear and nonconvex framework (e.g., [14], [16], [18], [21]). To overcome these limitations, the hierarchical division of the planning/control problem in two sequential stages has been proposed (e.g., [15], [23]). The first layer is assumed to be computationally more expensive and is performed less frequently, and the second layer uses simplified vehicle models and is computationally more efficient. An alternative is the linearization of the vehicle dynamics and constraints to reformulate the problem in a convex and simpler-to-solve framework (e.g., [20], [22]). However, in these, vehicle dimension constraints are typically discarded, either assuming point mass vehicle modeling or resorting to coarse approximations of the vehicle limits. In [14], the vehicle is decomposed into several circles of fixed radius. This approach leads to quadratic constraints and thus, the problem is solved by means of a sequential quadratic programming (SQP) method, which is typically more expensive than convex problems. Within the optimization based approaches, spatial-based vehicle models and problem formulations are recurrent in the literature (e.g., [15], [16], [22]).

D. Outline

The rest of this paper is organized as follows. In Section II, the problem is formulated, and we introduce the notion of road-aligned coordinate frame and derive the space-based vehicle model; in Section III, we present a method to compute the vehicle dimension constraints, and formulate the problem as a convex optimization; in Section IV, we provide simulation results showing the effectiveness of the proposed method; finally, in Section IV, we provide some concluding remarks and outline future work.

II. Problem Formulation

In this section, we setup the problem addressed in the remainder of the paper and review the derivation of the linearized spatial-based kinematic model. We follow the notation used in [2].

A. Road-aligned coordinate frame and spatial-based vehicle model

As in [2], the vehicle model and the path optimization problem constraints are derived in the road-aligned frame. Such representation eliminates the time-dependency by introducing the variable s representing the distance along

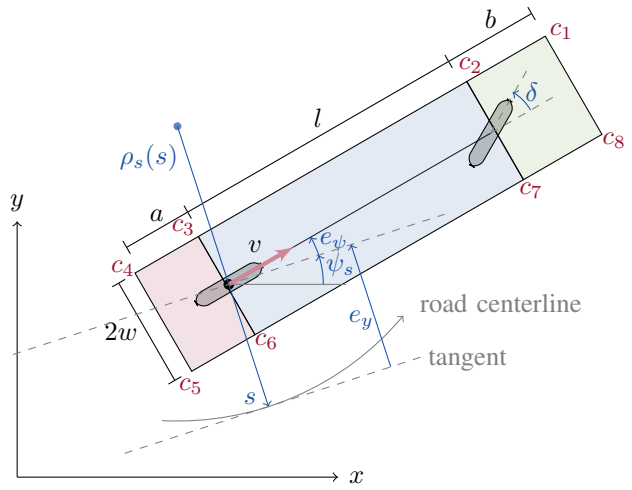

Fig. 2. Illustration of a bicycle model including the representation of the road-aligned coordinate frame. The vehicle model is separated in three different parts, namely, rear overhang (a), wheelbase (l), and front overhang (b).

the road centerline. We model the lateral displacement e_y and heading displacement e_ψ between the vehicle and the road as a function of space s. The relation between the global frame and the road-aligned frame is illustrated in Fig. 2. The parameters a, b, l, and w indicate front and rear overhang length, vehicle wheelbase, and vehicle half-width, respectively. This paper, focuses on the generation, at each time instant t, of a smooth spatial-based discrete path defined by

$$\mathcal{T}_t(\eta_n) = \begin{bmatrix} s_t(\eta_n) & e_{y,t}(\eta_n) & e_{\psi,t}(\eta_n) \end{bmatrix}^\intercal, \tag{1}$$

where η_n is an N-point discretization of the distance traveled along the planned path and $n \in \{1, 2, \dots, N\}$. Note that the distance along the planned path η is, in general, not the same as the distance s along the centerline.

We target on-road path optimization, where the road consists of a convex corridor characterized by bounds of the form

$$e_y^{\min}\left(s_t(\eta_n)\right) \leq e_{y,t}(\eta_n) \leq e_y^{\max}\left(s_t(\eta_n)\right).$$

We assume forward motion at constant speed. When modeling the vehicle in the space domain, we eliminate the velocity dependency and, therefore, vehicle acceleration constraints are not considered.

The nonholonomic car-like kinematic model is particularly suitable for low-speed (*i.e.*, when the lateral dynamics have little influence) applications. Its time-domain kinematic equations [24] are given by

$$\begin{aligned} \dot{x} &= \frac{\partial x}{\partial t} = v\cos(\psi), \\ \dot{y} &= \frac{\partial y}{\partial t} = v\sin(\psi), \\ \dot{\psi} &= \frac{\partial \psi}{\partial t} = \frac{v}{l}\tan(\delta), \end{aligned} \tag{2}$$

where (x, y, ψ) represent the vehicle pose in the global frame, l is the wheelbase, v is the longitudinal velocity in the vehicle coordinate system, and δ is the steering angle of the front wheels. We can relate the vehicle curvature κ with the vehicle steering angle δ by $\kappa = \frac{\tan(\delta)}{l}$.

According to Fig. 2, we can geometrically derive the relations

$$\begin{aligned} \dot{e}_y &= v\sin(e_\psi), \\ \dot{e}_\psi &= \dot{\psi} - \dot{\psi}_s, \\ \dot{s} &= \frac{\rho_s v\cos(e_\psi)}{\rho_s - e_y}, \end{aligned} \tag{3}$$

where ρ_s is the radius of curvature of the road (assumed to be bigger than e_y) and ψ_s is the road heading angle.

We use the chain rule to express the spatial derivative as function of the time derivative, namely $\frac{\partial(\cdot)}{\partial s} = \frac{\partial(\cdot)}{\partial t}\frac{\partial t}{\partial s} = \frac{\partial(\cdot)}{\partial t}\frac{1}{\dot{s}}$. Hence, the spatial-based representation of (3) can then be derived as

$$\begin{aligned} e'_y &= \frac{\dot{e}_y}{\dot{s}} = \frac{\rho_s - e_y}{\rho_s}\tan(e_\psi), \\ e'_\psi &= \frac{\dot{e}_\psi}{\dot{s}} = \frac{(\rho_s - e_y)\kappa}{\rho_s\cos(e_\psi)} - \psi'_s. \end{aligned} \tag{4}$$

Let the road centerline be uniformly discretized such that $\{s_j\}_{j=0}^N = \{s_0, s_1, \dots, s_N\}$. Also, let the state vector be $z = \begin{bmatrix} e_\psi & e_y \end{bmatrix}^\intercal$, and (4) be described by $z' = f(z, u)$. We linearize and discretize the vehicle model around the road centerline $\{e_{\psi,j}^{\text{ref}}\}_{j=0}^N = 0$, $\{e_{y,j}^{\text{ref}}\}_{j=0}^N = 0$, and $\{\kappa_j^{\text{ref}}\}_{j=0}^{N-1} = \{\frac{1}{\rho_{s,j}}\}_{j=0}^{N-1}$ using a first-order Taylor approximation and then apply exact discretization using [25].

III. Convex Path Optimization

In this section, we formulate the path optimization problem in the form of a convex optimization problem. To that end, we describe a novel method for computing the vehicle dimension constraints and their corresponding linearization.

A. Linear vehicle dimension constraints

The method presented in [2] assumes that the vehicle is short and the road has low curvature. Therefore, representing the vehicle as a rectangle in the road-aligned frame is, in fact, a fair approximation. However, as the vehicle length increases and the scenarios consider large road curvatures, that approximation loses its validity. Fig. 5 illustrates the vehicle distortion when converting from the global coordinate frame to the road-aligned frame.

In the following, we derive the vehicle dimension constraints for the vehicle edges between the front and rear axles. The derivation of the vehicle edges for the other parts (*i.e.*, the rear or the front overhang) is done analogously. We want to ensure that the vehicle wheelbase is always inside the drivable surface. Also, to evaluate the amount of vehicle chassis outside the drivable surface, we need to derive expressions that allow us to compute the position of the vehicle edges.

According to Fig. 3, at every s_j, we define the set of s-coordinates that include the s-coordinates of the vehicle corners in between any two grid points. Thus, we compute the s-coordinates of the four vehicle corners and approximate them conservatively. Formally, we define the set $\tilde{\mathcal{S}}_j$, which

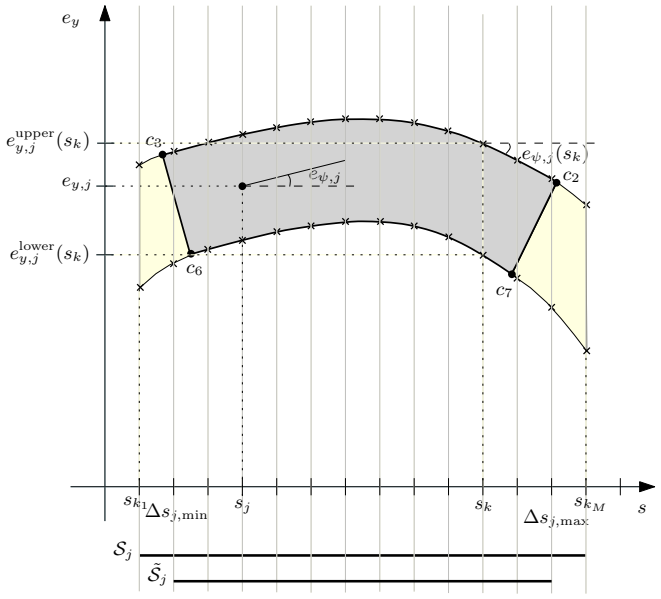

Fig. 3. Vehicle dimension constraints computation. It is illustrated the actual (distorted) vehicle geometry (gray) and the conservative envelop (yellow).

contains all the s-coordinates between the outer-most vehicle corners,

$$\begin{aligned}\tilde{\mathcal{S}}_j &= \{s \in \{s_k\}_{k=0}^{N} : \Delta s_{j,\min} \le s \le \Delta s_{j,\max}\} \\ &= \{s_{\tilde{k}_1}, s_{\tilde{k}_2}, \dots, s_{\tilde{k}_{\tilde{M}}}\},\end{aligned} \tag{5}$$

with $\tilde{M}$ being the number of points in the set $\tilde{\mathcal{S}}_j$ and

$$\begin{aligned}\Delta s_{j,\min} &= \min(s_{j,c_3}, s_{j,c_6}),\\ \Delta s_{j,\max} &= \max(s_{j,c_2}, s_{j,c_7}),\end{aligned}$$

where s_{j,c_i} for $i = 2, 3, 6, 7$ represent the four vehicle wheel-base corners when the rear-axle center is at position s_j. The s-coordinate of the corners $c_i,\ i = 1, \dots, 8$ in Fig. 2, can be expressed as

$$s_{c_i} = s + \xi_{c_i}\cos(e_\psi) + \zeta_{c_i}\sin(e_\psi), \tag{6}$$

with

$$\begin{aligned}\xi_{c_i} &\in \{l+b, l, 0, -a, -a, 0, l, l+b\},\\ \zeta_{c_i} &\in \{-w, -w, -w, -w, w, w, w, w\}.\end{aligned} \tag{7}$$

Afterwards, we take the conservative approximation of the vehicle envelope by prepending and appending the closest s-coordinates, defining the new set $\mathcal{S}_j$ as

$$\mathcal{S}_j = \{s_{\tilde{k}_1-1}, s_{\tilde{k}_1}, \dots, s_{\tilde{k}_{\tilde{M}}}, s_{\tilde{k}_{\tilde{M}}+1}\} = \{s_{k_1}, \dots, s_{k_M}\}, \tag{8}$$

where $M \ge \tilde{M}$ is the number of points in the set $\mathcal{S}_j$.

Thus, we describe lateral vehicle boundaries affine in s_j and nonlinear in $e_{\psi,j}$ as

$$\begin{aligned}e^{\text{lower}}_{y,j}(s_k) &= e_{y,j} + \tan(e_{\psi,j}(s_k))(s_k - s_j) - \frac{w}{\cos(e_{\psi,j})},\\ e^{\text{upper}}_{y,j}(s_k) &= e_{y,j} + \tan(e_{\psi,j}(s_k))(s_k - s_j) + \frac{w}{\cos(e_{\psi,j})},\end{aligned} \tag{9}$$

where $s_k \in \mathcal{S}_j$.

The inclusion of the s_k dependency in the term $\tan(e_{\psi,j}(s_k))$ captures the deformation caused by the transformation from global coordinate frame to road-aligned coordinate frame.

The linearization of (9) yields

$$\begin{aligned}e^{\text{lower}}_{y,\text{lin},j}(s_k) &= \begin{bmatrix} g^{\text{lower}}(s_k) & 1\end{bmatrix} z_j + h^{\text{lower}}_{\text{lin},j}(s_k),\\ e^{\text{upper}}_{y,\text{lin},j}(s_k) &= \begin{bmatrix} g^{\text{upper}}(s_k) & 1\end{bmatrix} z_j + h^{\text{upper}}_{\text{lin},s_k}(s_k),\end{aligned} \tag{10}$$

with $g^{\text{lower}}(s_k)$, $h^{\text{lower}}_{\text{lin},j}(s_k)$, $g^{\text{upper}}(s_k)$, and $h^{\text{upper}}_{\text{lin},j}(s_k)$ parameterized by s_j, $e^{\text{ref}}_{\psi,j}$, and $e^{\text{ref}}_{y,j}$. The later vehicle edges (10) are only evaluated at the discrete grid points s_{k_1} and s_{k_M}. In other words, we are only interested in evaluating the corners position of the outer envelope of each one of the vehicle parts. This can be expressed as the set of inequalities

$$\begin{aligned}\begin{bmatrix} e^{\text{lower}}_{y,\text{lin},j}(s_{k_1})\\ e^{\text{lower}}_{y,\text{lin},j}(s_{k_M})\end{bmatrix} &\ge \begin{bmatrix} e^{\min}_{y}(s_{k_1})\\ e^{\min}_{y}(s_{k_M})\end{bmatrix},\\ \begin{bmatrix} e^{\text{upper}}_{y,\text{lin},j}(s_{k_1})\\ e^{\text{upper}}_{y,\text{lin},j}(s_{k_M})\end{bmatrix} &\le \begin{bmatrix} e^{\max}_{y}(s_{k_1})\\ e^{\max}_{y}(s_{k_M})\end{bmatrix}.\end{aligned} \tag{11}$$

Inequalities (11) are linear in state z_j at position s_j and can be compactly summarized as

$$Q^{\text{lower}}_j z_j \ge q^{\text{lower}}_j \text{ and } Q^{\text{upper}}_j z_j \le q^{\text{upper}}_j,$$

with $Q^{\text{lower}}_j,\ Q^{\text{upper}}_j \in \mathbb{R}^{2\times 2}$, $q^{\text{lower}}_j,\ q^{\text{upper}}_j \in \mathbb{R}^2$ dependent on references $e^{\text{ref}}_{\psi,j}$ and $e^{\text{ref}}_{y,j}$.

We have a vector of inequalities (11) for each one of the vehicle parts. Thus, we construct

$$\begin{aligned}\bar{Q}^{\text{lower}}_j &= \begin{bmatrix} Q^{\text{lower,roh}}_j & Q^{\text{lower,bw}}_j & Q^{\text{lower,foh}}_j\end{bmatrix}^\intercal,\\ \bar{Q}^{\text{upper}}_j &= \begin{bmatrix} Q^{\text{upper,roh}}_j & Q^{\text{upper,bw}}_j & Q^{\text{upper,foh}}_j\end{bmatrix}^\intercal,\\ \bar{q}^{\text{lower}}_j &= \begin{bmatrix} q^{\text{lower,roh}}_j & q^{\text{lower,bw}}_j & q^{\text{lower,foh}}_j\end{bmatrix}^\intercal,\\ \bar{q}^{\text{upper}}_j &= \begin{bmatrix} q^{\text{upper,roh}}_j & q^{\text{upper,bw}}_j & q^{\text{upper,foh}}_j\end{bmatrix}^\intercal,\end{aligned} \tag{12}$$

where the subscripts roh, bw, and foh stand for rear overhang, between wheels, and front overhang, respectively.

B. Objective function

The objective function has two distinct terms. The first term is $J_{\text{smooth}}(\kappa)$, which is a convex function (e.g., $\max|\kappa|$) that is used to enforce smoothness of the planned path. The second term is $J_{\text{overhang}}(\sigma_{e_y}, \lambda) = \sigma^\intercal_{e_y}\lambda\sigma_{e_y}$, which is another convex function that accounts for the vehicle dimension constraints violations. The parameter λ is used to penalize vehicle dimension constraints violations (e.g., lane boundaries or obstacles). Since there are three different vehicle parts to be considered, we choose $\lambda = \text{diag}(\lambda_{\text{foh}}, \lambda_{\text{bw}}, \lambda_{\text{roh}})$, to penalize the front overhang, between wheels, and rear overhang outside the drivable surface, separately. We choose $\lambda_{\text{bw}} \gg \lambda_{\text{foh}} = \lambda_{\text{roh}}$ as we want to prevent the wheels from exiting the lane boundaries but allow the overhang(s) to surpass the lane limit. The introduction of a slack variable vector $\sigma_{e_y} = [\sigma_{e^{\text{foh}}_y}, \sigma_{e^{\text{bw}}_y}, \sigma_{e^{\text{roh}}_y}]^\intercal$ ensures feasibility of the optimization problem.

C. Convex optimization problem formulation

We formulate the path optimization problem as a convex optimization based on the linearized vehicle model and vehicle dimension constraints as

$$\min_{\kappa,\sigma_{e_y}} \quad J_{\text{smooth}}(\kappa) + J_{\text{overhang}}(\sigma_{e_y}, \lambda) \tag{13a}$$

$$\text{s.t.} \quad z_{j+1} = A_j z_j + B_j \tilde{\kappa}_j,\ j = 0, \dots, N-1, \tag{13b}$$

$$e_{\psi,N} = e^{\text{ref}}_{\psi,N},\ e_{y,N} = e^{\text{ref}}_{y,N}, \tag{13c}$$

$$e_{\psi,0} = e^{\text{ref}}_{\psi,0},\ e_{y,0} = e^{\text{ref}}_{y,0}, \tag{13d}$$

$$e^{\min}_{y,j} \leq e_{y,j} \leq e^{\max}_{y,j},\ j = 1, \dots, N, \tag{13e}$$

$$\kappa^{\min} \leq \kappa_j \leq \kappa^{\max},\ j = 0, \dots, N-1, \tag{13f}$$

$$\Delta\kappa^{\min} \leq \kappa_j - \kappa_{j-1} \leq \Delta\kappa^{\max},\ j = 0, \dots, N-1, \tag{13g}$$

$$\kappa_{-1} = \kappa(0) \tag{13h}$$

$$\bar{Q}^{\text{lower}}_j z_j \geq \bar{q}^{\text{lower}}_j - \sigma_{e_y},\ j = 1, \dots, N, \tag{13i}$$

$$\bar{Q}^{\text{upper}}_j z_j \leq \bar{q}^{\text{upper}}_j + \sigma_{e_y},\ j = 1, \dots, N, \tag{13j}$$

$$\sigma_{e_y} \geq 0, \tag{13k}$$

with decision variables $\{\kappa_j\}_{j=0}^{N-1}$ and σ_{e_y}. The initial state is denoted by $z(0) = [e^{\text{ref}}_{\psi,0}\ e^{\text{ref}}_{y,0}]^\intercal$, initial curvature $\kappa(0)$, and $\tilde{\kappa}_j = \kappa_j - \kappa^{\text{ref}}_j$. Constant upper and lower bounds are $\kappa^{\min}$, $\kappa^{\max}$, $\Delta\kappa^{\min} = \dot{\kappa}^{\min} T_s$, and $\Delta\kappa^{\max} = \dot{\kappa}^{\max} T_s$, where time T_s is related heuristically to the discretization grid taking reference speed into account. The vehicle dynamics model imposed under the form of equality constraint (13b) is presented in Section II-A. Vehicle dimension constraints (13i) and (13j) are discussed in Section III-A. The objective function (13a) is discussed in Section III-B.

IV. Simulation Results

In this section, we discuss the performance of the proposed path optimization algorithm described in III-C. The performance evaluation is based on the vehicle overhang amount exiting the drivable surface (*i.e.*, the road lane limits). We measure this amount in two situations, namely, when tracking the road centerline and the optimized path. To be able to provide performance bounds, we derive geometrically the minimum road width necessary depending on the vehicle geometry and road curvature.

A. Minimum road width

Fig. 4 depicts the geometry of the problem of computing the minimum road width depending on the road curvature $\kappa_s = \rho_s^{-1}$ and the vehicle dimensions. Using this information, we can derive the minimum road width necessary for the vehicle to travel with all the wheels inside the lane limits. Moreover, we can also derive an expression for an estimate of the amount of vehicle overhang outside the drivable surface. From Fig. 4 we get

$$R_1 = \sqrt{R_2^2 + l^2}, \tag{14a}$$

$$R_3 = \sqrt{R_2^2 + (l+b)^2}, \tag{14b}$$

Furthermore,

$$R_1 = \rho_s + W, \tag{15a}$$

$$R_2 = \rho_s - W + 2w. \tag{15b}$$

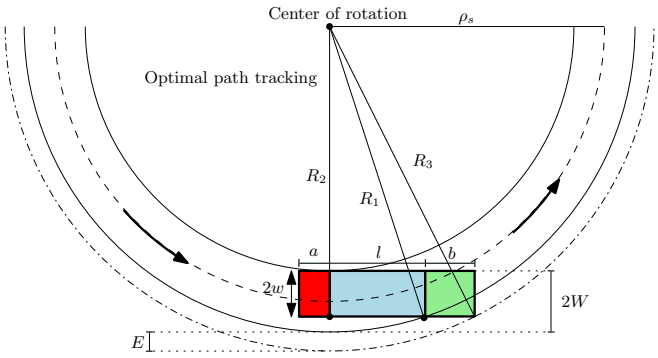

Fig. 4. Geometric relations used in the computation of the minimum road width.

From the above relations, we determine the minimum road width as

$$W = \frac{l^2 + 4w^2 + 4\rho_s w}{4(\rho_s + w)}, \tag{16}$$

and the estimation of the amount of vehicle overhang outside the drivable surface

$$E = R_3 - R_1. \tag{17}$$

Note that this estimate considers that the vehicle is traveling with the rear wheels, and, consequently, with the vehicle center of rotation, tangentially to the road inner-lane. If the road width is bigger than the minimum computed in (16), the vehicle does not necessarily need to travel tangentially to the road inner-lane to minimize the amount of overhangs outside the drivable surface. We will come back to this discussion in the next subsections.

B. Path optimization

For prototyping the optimization problem, we use the MATLAB toolbox CVX [26] with Gurobi as a solver [27]. All simulations are conducted on a desktop PC running Windows 7 equipped with an Intel Xeon CPU E5-2670 v2 @2.50GHz, 16.0GB of memory, and using MATLAB R2016a.

The scenario presented in Fig. 5 consists of a road with tight lane boundaries and large curvature. The reference path data points are recorded from Scania's test facilities near Södertälje, Sweden. The space discretization is chosen to be 0.25 m. The maximum path curvature is 0.117 m^{-1}, which, using (16), corresponds to a minimum lane width of 2.19 m. However, we use a lane width of 2.5 m, otherwise, the optimal path would be to travel tangentially to the road inner-lane as shown in Fig. 4 and discussed in Sec. IV-A. The vehicle dimensions used and input constraints are based on the Scania bus shown in Fig. 1. The dimensions are $a = 2.66$ m, $b = 3.34$ m, $l = 6$ m, and $w = 1.27$ m. The input constraints are $\kappa^{\max} = 0.18$ m^{-1} and $\dot{\kappa}^{\max} = 0.03$ m^{-1}/s.

1) Offline path optimization using the whole reference path: Fig. 5 depicts the outcome of (13). The first term of the cost function is chosen as $J_{\text{smooth}}(\kappa) = \max(|D_1\kappa|) + \max(|\kappa|)$, where D_1 is the matrix operator that calculates the first-order differences of

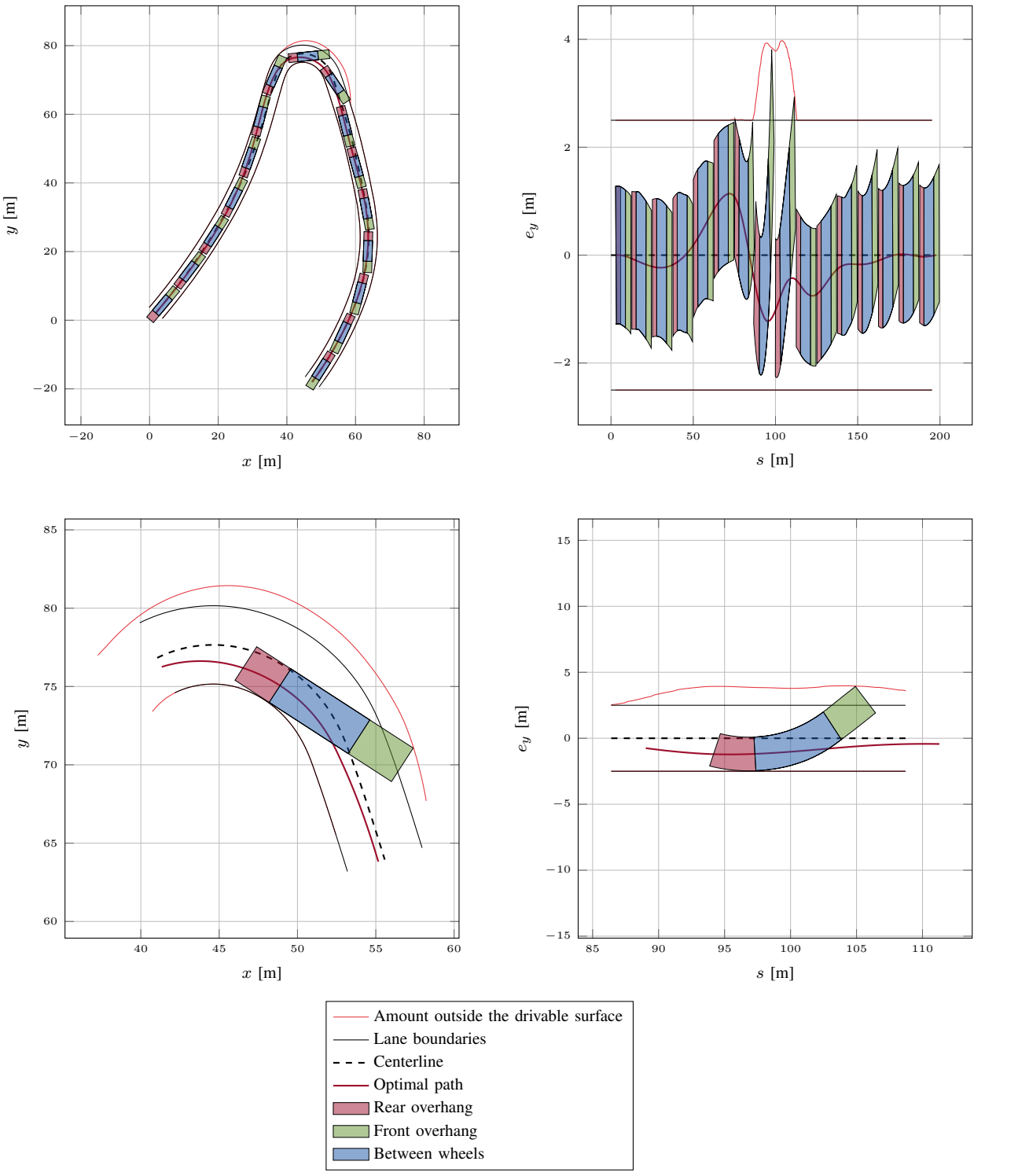

Fig. 5. Resulting vehicle path in the global frame (upper-left plot) after the retransformation from the road-aligned frame (upper-right plot), in which computations are conducted. The plot depicts vehicle dimensions (displayed every 50th sample), where we distinguish the three different vehicle parts. According to the problem optimization, the vehicle wheelbase is always inside the lane boundaries, while minimizing the amount of the overhang that exceeds them. Also, the amount of the vehicle overhang exceeding the lane limits is shown. The lower plots depict one instant when the overhang is outside of the drivable surface.

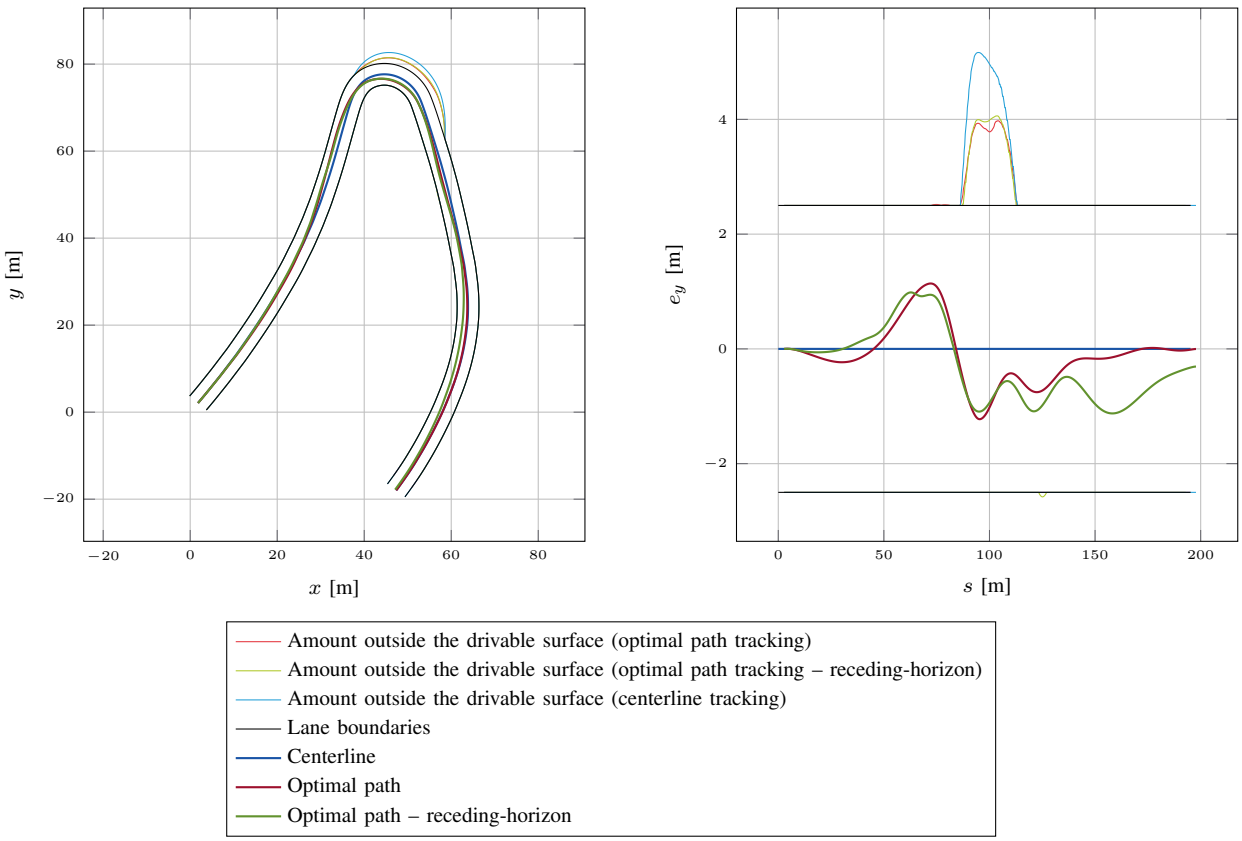

Fig. 6. Path comparison in the global frame (left plot) after the retransformation from the road-aligned frame (right plot), in which computations are conducted. Also, the amount of the vehicle overhang exceeding the lane limits is shown.

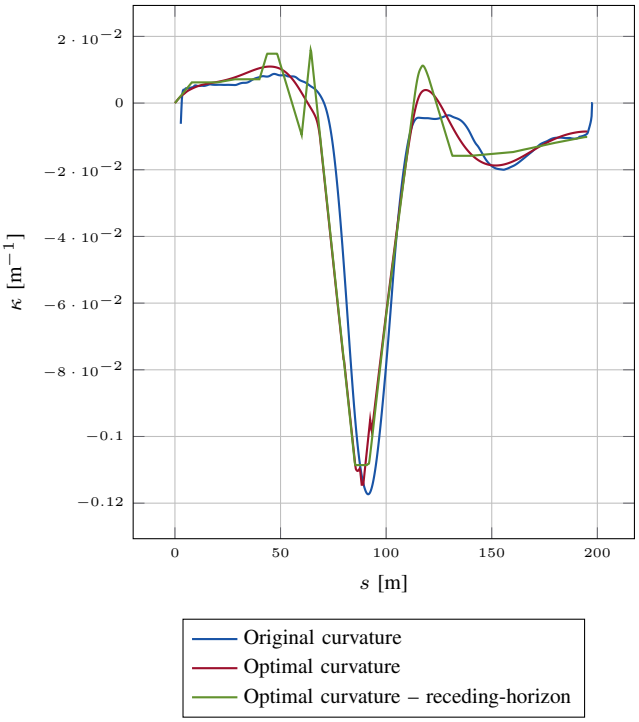

Fig. 7. Vehicle curvature (the control input) comparison.

a vector. Moreover, $\lambda_{\text{foh}} = \lambda_{\text{roh}} = 1$ and $\lambda_{\text{bw}} = 10^3$. It can be seen that the vehicle wheels always lie inside the drivable surface. Note that the obtained shape of the optimal path *cuts the corner* to use the drivable surface more efficiently. Moreover, Fig. 6 compares the optimal path with the centerline. When tracking the centerline, the amount of (front) overhang outside the drivable surface is 2.67 m, while when tracking the optimal path it is only 1.47 m. Therefore, our proposed method reduces the amount of overhang outside the drivable surface by 45%. Estimating the amount of overhang outside the drivable surface using (17), we get 1.66 m. Since the road width chosen in this scenario is bigger than the minimum road width (16), the optimal path is not tangent to the road inner-lane, leading to a smaller amount of overhang outside the drivable surface.

2) Online path optimization using receding-horizon framework: The major drawback with the proposed method is its high computation burden. The scenario presented has 195 m and, thus, 780 sampling points. The optimization solver takes 2.16 s to find the solution. A possible alternative is to make use of the receding-horizon framework. Instead of using the whole reference path, the optimization can be performed over shorter subsets of the path, while moving along it. Moreover, the solutions computed at each step can be used as new references to improve the linearization accuracy and reduce the solver execution time in the next step. An optimal path is computed every 5 m, using a prediction horizon of 10 m. Fig. 6 compares the outcome of (13) in

a receding-horizon fashion with optimizing over the whole reference path. When tracking the receding-horizon optimal path, the amount of (front) overhang that exceeds lane limits is 1.56 m, only 0.09 m more than when tracking the optimal path. Moreover, the average computation time is 0.30 s, with a maximum of 0.55 s. These execution times demonstrate the feasibility of applying the method in real-time, since a vehicle would not drive faster than 5 m/s in a road with curvature exceeding 0.11 m^{-1} [28]. Therefore, at 5 m/s solutions are computed every 1 s, which comfortably fits the execution time of the solver. Fig. 7 depicts the input signals computed using the offline and the receding-horizon approach. As expected, due to the inherent suboptimality of the receding-horizon approach, the computed curvature is not as smooth as in the offline case, which considers the whole path. Note that, for comparison purposes, the cost function used in the receding-horizon approach is exactly the same as in the offline case. We can get smoother solutions by tuning the cost function by, for instance, penalizing more non-smooth curvature signals. In any case, it can be seen that, either in the road-aligned and in the global coordinate frame, the computed optimal path is smooth.

V. Conclusions and Future Work

In this paper, we proposed a spatial-based path optimization method for actuator and vehicle dimension constraints using convex optimization. The vehicle motion is predicted using a linearized and discretized spatial-based vehicle model in a road-aligned coordinate frame. We propose a novel method to compute the vehicle dimension constraints, since long vehicles are severely deformed in the road-aligned frame. In simulation, the effectiveness of the method was evaluated using a real road with tight lane constraints. The amount of overhang that surpassed the lane limit was reduced by 45% when compared with driving the same road tracking the centerline. Moreover, we demonstrated the real-time applicability of our method by using it in a receding-horizon framework, with almost no loss of performance.

As future work, we would like to compare our method with the path driven by professional drivers and, eventually, implement such algorithm in a real vehicle. Also, we believe that our method is easily extendable to other applications involving long vehicles.

References

[1] European Commission, "Attitudes of europeans towards urban mobility," Electronic, 2013.

[2] M. G. Plessen, P. F. Lima, J. Mårtensson, A. Bemporad, and B. Wahlberg, "Spatial-based trajectory planning under vehicle dimension constraints using sequential linear programming," in *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, June 2017.

[3] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, March 2016.

[4] M. Pivtoraiko, R. Knepper, and A. Kelly, "Differentially constrained mobile robot motion planning in state lattices," *Journal of Field Robotics*, vol. 26, no. 3, pp. 308–333, 2009.

[5] M. Cirillo, "From videogames to autonomous trucks: A new algorithm for lattice-based motion planning," in *Proceedings of the Conference on Automated Planning and Scheduling*, June 2016, pp. 6–12.

[6] T. Ersson and X. Hu, "Path planning and navigation of mobile robots in unknown environments," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2. IEEE, 2001, pp. 858–864.

[7] S. Koenig and M. Likhachev, "Improved fast replanning for robot navigation in unknown terrain," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 2002, pp. 968–975.

[8] D. Ferguson and A. Stentz, "Field D*: An interpolation-based path planner and replanner," in *Robotics research*. Springer, 2007, pp. 239–253.

[9] K. Daniel, A. Nash, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," *Journal of Artificial Intelligence Research*, vol. 39, pp. 533–579, 2010.

[10] S. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Iowa State University, Tech. Rep., 1998.

[11] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime motion planning using the RRT," in *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1478–1483.

[12] J. Hwan Jeon, S. Karaman, and E. Frazzoli, "Anytime computation of time-optimal off-road vehicle maneuvers using the RRT*," in *Proceedings of the IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*. IEEE, 2011, pp. 3276–3282.

[13] E. Plaku, L. E. Kavraki, and M. Y. Vardi, "Discrete search leading continuous exploration for kinodynamic motion planning." in *Robotics: Science and Systems*, 2007, pp. 326–333.

[14] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha – a local, continuous method," in *Proceedings of the IEEE Intelligent Transportation Systems*. IEEE, 2014, pp. 450–457.

[15] Y. Gao, A. Gray, J. Frasch, T. Lin, E. Tseng, J. Hedrick, and F. Borrelli, "Spatial predictive control for agile semi-autonomous ground vehicles," in *Proceedings of the International Symposium on Advanced Vehicle Control*, September 2012.

[16] J. V. Frasch, A. Gray, M. Zanon, H. J. Ferreau, S. Sager, F. Borrelli, and M. Diehl, "Aq," in *Proceeding of the IEEE European Control Conference*, 2013, pp. 4136–4141.

[17] M. Plessen, D. Bernardini, H. Esen, and A. Bemporad, "Spatial-based predictive control and geometric corridor planning for adaptive cruise control coupled with obstacle avoidance," *IEEE Transactions on Control Systems Technology*, 2017.

[18] X. Qian, F. Altché, P. Bender, C. Stiller, and A. de La Fortelle, "Optimal trajectory planning for autonomous driving integrating logical constraints: An MIQP perspective," in *Proceedings of the International IEEE Intelligent Transportation Systems Conference*. IEEE, 2016, pp. 205–210.

[19] 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.

[20] G. P. Bevan, H. Gollee, and J. O'reilly, "Trajectory generation for road vehicle obstacle avoidance using convex optimization," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 224, no. 4, pp. 455–473, 2010.

[21] M. Werling and D. Liccardo, "Automatic collision avoidance using model-predictive online optimization," in *Proceedings of the IEEE Conference on Decision and Control*. IEEE, 2012, pp. 6309–6314.

[22] A. Carvalho, Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, "Predictive control of an autonomous ground vehicle using an iterative linearization approach," in *Proceedings of the International IEEE Intelligent Transportation Systems Conference*. IEEE, 2013, pp. 2335–2340.

[23] T. Gu, J. Snider, J. M. Dolan, and J.-w. Lee, "Focused trajectory planning for autonomous on-road driving," in *Proceedings of the IEEE Intelligent Vehicles Symposium*. IEEE, 2013, pp. 547–552.

[24] A. De Luca, G. Oriolo, and C. Samson, *Feedback control of a nonholonomic car-like robot*. Springer Berlin Heidelberg, 1998, vol. 229, ch. 4, pp. 171–253.

[25] C. Van Loan, "Computing integrals involving the matrix exponential," *IEEE Transactions on Automatic Control*, vol. 23, no. 3, pp. 395–404, 1978.

[26] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[27] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2016. [Online]. Available: http://www.gurobi.com

[28] P. F. Lima, M. Trincavelli, J. Mårtensson, and B. Wahlberg, "Clothoid-based speed profiler and control for autonomous driving," in *Proceedings of the International IEEE Intelligent Transportation Systems Conference*, Sept. 2015, pp. 2194 – 2199.