Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · June 2010

DOI: 10.1109/ROBOT.2010.5509799 · Source: IEEE Xplore

4 authors, including:

Moritz Werling
BMW
33 PUBLICATIONS **496** CITATIONS
SEE PROFILE

Sören Kammel
Robert Bosch GmbH
37 PUBLICATIONS **622** CITATIONS
SEE PROFILE

2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenét Frame

Moritz Werling, Julius Ziegler, Sören Kammel, and Sebastian Thrun

Abstract— **Safe handling of dynamic highway and inner city scenarios with autonomous vehicles involves the problem of generating traffic-adapted trajectories. In order to account for the practical requirements of the holistic autonomous system, we propose a semi-reactive trajectory generation method, which can be tightly integrated into the behavioral layer. The method realizes long-term objectives such as velocity keeping, merging, following, stopping, in combination with a reactive collision avoidance by means of optimal-control strategies within the Frenét-Frame [12] of the street. The capabilities of this approach are demonstrated in the simulation of a typical high-speed highway scenario.**

I. Introduction

A. Motivation

The past three decades have witnessed ambitious research in the area of automated driving. As autonomous vehicles advance toward handling realistic road traffic, they face street scenarios where dynamics of other traffic participants must be considered explicitly. This includes every day driving maneuvers like merging into traffic flow, passing with oncoming traffic, changing lanes, or avoiding other vehicles. Under simplified conditions, such as during the *2007 DARPA Urban Challenge*[1], this can be tackled with fairly simple heuristics and conservative estimates [18]. However, these approaches quickly reach their limits in nose-to-tail traffic and at high driving speeds resulting in poor performance or even accidents [5]. This is where trajectory concepts come into play, which explicitly account for the time t on the planning and execution level.

The presented method embarks on this strategy and sets itself apart from previous work in that it is especially suitable for highway driving, as it generates velocity invariant movement[2] and transfers velocity and distance control to the planning level. Additionally, the algorithm provides for reactive obstacle avoidance by the *combined* usage of steering and breaking/acceleration.

B. Related work

Several methods for trajectory planning have been proposed [11], [19], [2], [4] that find a global trajectory connecting a start and a - possibly distant - goal state. However, these methods fail to model the inherent unpredictability of other traffic, and the resulting uncertainty, given that they rely on precise prediction of other traffic participant's motions over a long time period. Other approaches taken towards trajectory planning follow a discrete optimization scheme (e. g. [16], [1], [7]): A finite set of trajectories is computed, typically by forward integration of the differential equations that describe vehicle dynamics. From this set, the trajectory is chosen that minimizes a given cost functional. For generation of the trajectory set a parametric model is chosen, like curvature polynomials of arbitrary order. While this reduces the solution space and allows for fast planning, it may introduce suboptimality. We will show in Sec. II that this can lead to both overshoots and stationary offsets in curves. In [9], a tree of trajectories is sampled by simulating the closed loop system using the rapidly exploring random tree algorithm [10]. The system incorporates many heuristics in the form of sampling biases to assert well behaved operation. An approach that is in a similar spirit to our method but only considers the free problem that is not constrained by obstacles has been taken by [17]. Here, the optimal control trajectory for an aero dynamic system is found within a function space that is spanned by a Galerkin base.

For the above mentioned reasons and to, at least partly, overcome the limitations of the approaches described in the literature, we propose a local method, which is capable of realizing high-level decisions made by an upstream, behavioral layer (long-term objectives) and also performs (reactive) emergency obstacle avoidance in unexpected critical situations. One aspect that sets our method especially apart from other schemes is the guaranteed stability (temporal consitency) of the non-reactive maneuvers that follows directly from Bellman's principle of optimality. Within this work we adhere with the strategy of strictly decoupling feedback from planning. We demonstrated before that it is advantagous to separate the navigation task into real time trajectory generation and subsequent local stabilization through trajectory tracking feedback control. This is in contrast to some other approaches that close the control loop by feeding the observed state of the system directly back into the planning

M. Werling is with the Department of Applied Computer Science and Automation (AIA), University of Karlsruhe, 76131 Karlsruhe, Germany moritz.werling@iai.fzk.de

J. Ziegler is with the Department of Measurement and Control (MRT), University of Karlsruhe ziegler@mrt.uka.de

S. Kammel is with Robert Bosch LLC Research and Technology Center, Palo Alto, California 94304 soeren.kammel@us.bosch.com

S. Thrun is with Stanford Artificial Intelligence Laboratory, Stanford University, Stanford, California 94305 thrun@stanford.edu

The authors gratefully acknowledge the cooperation between the "Valley rally" project of Stanford University and the German Transregional Collaborative Research Center 28 *Cognitive Automobiles*. Both projects cross-fertilized each other and revealed significant synergy.

[1]The DARPA Urban Challenge is a research program conducted in a competitive format to address the challenges of autonomous driving, see http://www.darpa.mil/grandchallenge.

[2]It is highly desirable to generate lane change and merging maneuvers, which are timed completely independently from the absolute travelling speed.

stage. The focus of this work will be on the trajectory generation phase, i.e. generating the nominal input required to safely operate the vehicle in specific maneuvering modes.

II. Optimal control approach

Applying optimal control theory to trajectory generation is not new. In contrast to the well known works [13], [3], our main focus is not on the optimization of a certain cost functional. We instead formulate the problem of trajectory tracking in an optimal sense to take advantage of the theory asserting consistency in the choice of the best feasible trajectory over time. With this, we seek to make sure that once an optimal solution is found, it will be retained (Bellman's Principle of Optimality). For the car, this would mean that it follows the remainder of the previously calculated trajectory in each planning step and therefore temporal consistency is provided.

This is in contrast to methods such as [16], [1], [7], where the trajectories are represented parametrically, e. g. by assuming system inputs or curvature to be polynomials, and the set is generated by sampling from the parameter space [1] or by optimizing on it in order to meet certain end constraints [7], [16]. In general, the optimal trajectory - in terms of the cost functional - is *not* part of the function space spanned by the parameters. Consequently, Bellman's principle of optimality does not hold, and on the next iteration a trajectory will be chosen that is slightly different. Figure 1 illustrates that this temporal consistency can lead to overshoots or even instabilities.

While our main criterion in choosing a cost functional is compliance with Bellman's principle of optimality, trajectories minimizing it must still be close to the desired traffic behavior of the autonomous car. Therefore, let us verbally describe the "ideal" behavior of an autonomous car moving along a street: Suppose the car has a certain lateral offset to the desired lane, say due to a recently initiated lane change or an obstacle avoidance maneuver. The car should then return within its driving physics to the desired lane making the best compromise between the ease and comfort perceived in the car and the time it takes to get to the desired lane position.

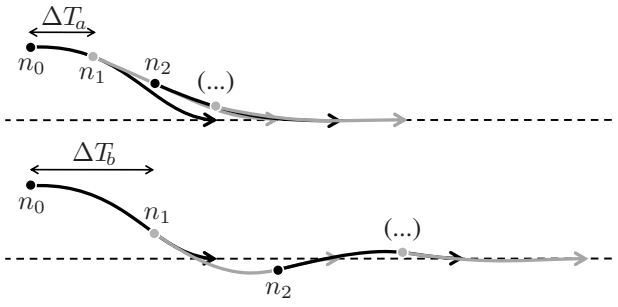

Fig. 1. Two different transient behaviors of the same planning strategy depending on the replanning frequency: (top) High replanning frequency with tolerable transient; (bottom) low replanning frequency causes overshoots. ΔT_a and ΔT_b are the inverse planning frequencies and n_i the starting points of subsequent planning steps.

At the same time, the best compromise has to be found in the longitudinal direction in an analog manner: Assuming the car drives too fast or too close to the vehicle in front, it has to slow down noticeably but without excessive rush. Mathematically speaking, *ease and comfort* can be best described by the *jerk*, which is defined by the change of acceleration over time, where *needed time* is simply $T = t_{\text{end}} - t_{\text{start}}$ of the maneuver.

As the solution to the general restricted optimization problem is not limited to a certain function class[3], the problem becomes highly complicated and can be solved numerically at best. This is why our approach searches, as a reasonable approximation for the restricted optimization problem, only within the set of optimal solutions to the *unrestricted* (free) optimization problem and chooses the best solution, which fulfills the restrictions. This in turn means that as soon as the best solution is valid (restrictions are then not active) temporal consistency of the non-reactive trajectories is assured. The verification of the reactive heuristic is yet to be shown simulatively.

III. Motion Planning in the Frenét Frame

A well known approach in tracking control theory is the *Frenét Frame method*, which asserts invariant tracking performance under the action of the special Euclidean group $SE(2) := SO(2) \times \mathbb{R}^2$.Here, we will apply this method in order to be able to combine different lateral and longitudinal cost functionals for different tasks as well as to mimic human-like driving behavior on the highway. As depicted in Fig. 2, the moving reference frame is given by the tangential and normal vector $\vec{t}_r, \vec{n}_r$ at a certain point of some curve referred to as the *center line* in the following. This center line represents either the ideal path along the free road, in the most simple case the road center, or the result of a path planning algorithm for unstructured environments [20]. Rather than formulating the trajectory generation problem directly in Cartesian Coordinates $\vec{x}$, we switch to the proposed dynamic reference frame and seek to generate a one-dimensional trajectory for both the root point $\vec{r}$ along the center line and the perpendicular offset d with the relation

$$\vec{x}(s(t), d(t)) = \vec{r}(s(t)) + d(t)\,\vec{n}_r(s(t)), \tag{1}$$

as shown in Fig. 2, where s denotes the covered arc length of the center line, and $\vec{t}_x, \vec{n}_x$ are the tangential and normal vectors of the resulting trajectory $\vec{x}(s(t), d(t))$.

Human perception obviously weights lateral and longitudinal changes of acceleration differently. Since the vector pairs $\vec{t}_x, \vec{n}_x$ and $\vec{t}_s, \vec{n}_s$ almost align at higher speeds, we consider the previously introduced jerk in these Frenét coordinates as $\dddot{d}$ and $\dddot{s}$. From [15] we also know that quintic polynomials are the jerk-optimal connection between a start state $P_0 = [p_0, \dot{p}_0, \ddot{p}_0]$ and an end state $P_1 = [p_1, \dot{p}_1, \ddot{p}_1]$ within the time interval $T := t_1 - t_0$ in a one-dimensional problem. More

[3]This becomes clear if you imagine the autonomous car being trapped between four moving cars, one in each direction, forcing the car's motion into a single possible solution, e. g. a sinusoidal.

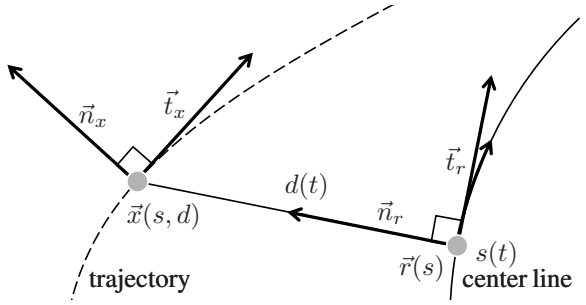

Fig. 2. Trajectory generation in a Frenét-frame

precisely, they minimize the cost functional given by the time integral of the square of jerk

$$J_t(p(t)) := \int_{t_0}^{t_1} \dddot{p}^2(\tau)\mathrm{d}\tau.$$

We will use this result also for our approach:

Proposition 1: Given the start state $P_0 = [p_0, \dot{p}_0, \ddot{p}_0]$ at t_0 and $[\dot{p}_1, \ddot{p}_1]$ of the end state P_1 at some $t_1 = t_0 + T$, the solution to the minimization problem of the cost functional

$$C = k_j J_t + k_t g(T) + k_p h(p_1)$$

with arbitrary functions g and h and $k_j, k_t, k_p > 0$ is also a quintic polynomial.

Proof:[4] Assume the optimal solution to the proposed problem was not a quintic polynomial. It would connect the the two points P_0 and $P_1(p_{1,\text{opt}})$ within the time interval T_{opt}. Then a quintic polynomial through the same points and the same time interval will always lead to a smaller cost term $\int_{t_0}^{t_1} \dddot{p}^2(\tau)$ in addition to the same two other cost terms. This is in contradiction to the assumption so that the optimal solution has to be a quintic polynomial. ■

IV. Generation of Lateral Movement

A. High Speed Trajectories

Since we seek to minimize the squared jerk of the resulting trajectory, we choose the start state of our optimization $D_0 = [d_0, \dot{d}_0, \ddot{d}_0]$ according to the previously calculated trajectory, s. Sec. VI, so that no discontinuities occur. For the optimization itself, we let $\dot{d}_1 = \ddot{d}_1 = 0$ (the *target manifold* in the optimal control lingo) as we want to move parallel to the center line. In addition, we choose $g(T) = T$ and $h(d_1) = d_1^2$ so that we get the cost functional

$$C_d = k_j J_t(d(t)) + k_t T + k_d d_1^2, \tag{2}$$

since we want to penalize solutions with slow convergence and those, which are off from the center $d = 0$ at the end. Notice, that this cost functional and the ones used in the sequel do not depend on the velocity of the vehicle (except for Sect. IV-B). As we know from Prop. 1 that the optimal solution is a quintic polynomial, we could calculate its coefficients and T minimizing (2) (rather lengthy expressions) and check it (in combination with the best longitudinal trajectory $s(t)$) against collision. If we are lucky, it is valid and we are done. If it is not, we would have to find a collision-free alternative, some kind of "second best" trajectory, by slightly modifying T along with the coefficients of $d(t)$ (and $s(t)$) and check for collision again, and so on.

Instead of calculating the best trajectory explicitly and modifying the coefficients to get a valid alternative, we generate in the first step, such as in [16], a whole trajectory set: By combining different end conditions d_i and T_j

$$[d_1, \dot{d}_1, \ddot{d}_1, T]_{ij} = [d_i, 0, 0, T_j]$$

for the polynomials, as shown in Fig. 3 at simulation time $t = 0$, all possible maneuvers are sufficiently covered. In the second step we pick the valid trajectory with the lowest cost. Notice that, as we continue in each step along the optimal trajectory (non-reactive, long-term goals), the remaining trajectory will be, in contrast to Fig. 1, the optimal solution in the next step. This is contributed, on the one hand, to the fact that we choose the discrete points in *absolute* time (in the simulation of Fig. 3 every full second), so that in each step the previously optimal trajectory is available in the next step, on the other, that we are in the correct (optimal) function class for the unrestricted problem.

B. Low Speed Trajectories

At higher speeds, $d(t)$ and $s(t)$ can be chosen independently[5], as proposed in the last section. At extreme low speeds, however, this strategy disregards the non-holonomic property of the car, so that the majority of the trajectories has to be rejected due to invalid curvatures (s. Sec. VI). For this reason the behavioral layer can switch below a certain velocity threshold to a slightly different trajectory mode generating the lateral trajectory in dependence on the longitudinal movement, that is

$$\vec{x}(s(t), d(t)) = \vec{r}(s(t)) + d(s(t))\,\vec{n}_r(s(t)).$$

Remember that our focus is not on the minimization of a certain cost functional, but we take advantage of optimization theory in order to rate the generated trajectories consistently. As quintic polynomials for $d(s)$ (defined over the center line arc length s) lead to clothoid-spline-like parallel maneuvers for orientation deviations from the center line smaller than $\frac{\pi}{2}$, we stick to the polynomials also for low speeds and modify the cost functional to

$$C_d = k_j J_s(d(s)) + k_t S + k_d d_1^2,$$

with $S = s_1 - s_0$ and with $(\cdot)' := \frac{\partial}{\partial s}(\cdot)$

$$J_s(d(s)) := \int_{s_0}^{s_1} d'''^2(\sigma)\mathrm{d}\sigma.$$

According to Prop. 1, the quintic polynomials over s belong to the optimal function class. The set generation can then

[4] From an optimal control's perspective this is directly clear, as the end point costs $g(T)$ and $h(p_1)$ do not change the Euler-Lagrange equation.

[5] excluding extreme maneuvers, where the combined lateral and longitudinal forces on the car play an important roll

carried analogously out to $d(t)$ with the start point $D_0 = [d_0, d_0', d_0'']$ and the various end points

$$[d_1, d_1', d_1'', T]_{ij} = [d_i, 0, 0, T_j].$$

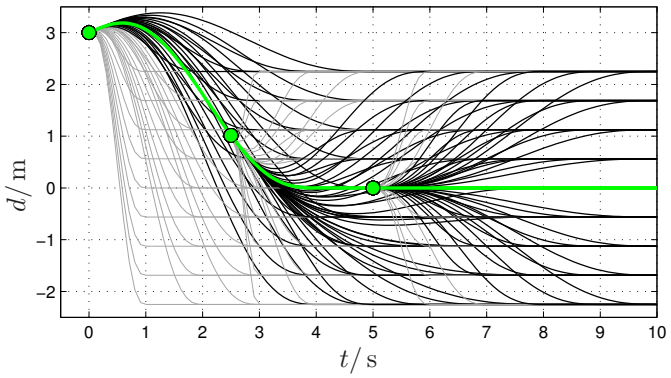

Fig. 3. Optimal lateral movement resulting from cyclic replanning with green being the optimal trajectory, black the valid, and gray the invalid alternatives

V. Generation of Longitudinal Movement

In contrast to previous works where time or travelled distance was the key criterion, we will focus here on comfort and contribute at the same time to safety at high speeds, as smooth movements adapt much better to the traffic flow. For that reason, we also take the longitudinal jerk into account in our optimization problem.

A. Following, Merging, and Stopping

Since distance keeping, merging, and stopping at certain positions require trajectories, which describe the transfer from the current state to a longitudinal, possibly moving, target position $s_{\text{target}}(t)$, we generate a longitudinal trajectory set, analogously to the lateral trajectories, starting at $S_0 = [s_0, \dot{s}_0, \ddot{s}_0]$ and vary the end constraints by different Δs_i and T_j according to

$$[s_1, \dot{s}_1, \ddot{s}_1, T]_{ij} = [[s_{\text{target}}(T_j) + \Delta s_i], \dot{s}_{\text{target}}(T_j), \ddot{s}_{\text{target}}(T_j), T_j]$$

as depicted for $t = 0$ in Fig. 4, and finally evaluate for each polynomial the cost functional

$$C_t = k_j J_t + k_t T + k_s [s_1 - s_d]^2.$$

Following

For following, the moving target point can be derived from international traffic rules, e. g. [14], requiring a certain temporal safety distance to the vehicle ahead, known as the *constant time gap law*, so that the desired position of the following vehicle along the lane is given by

$$s_{\text{target}}(t) := s_{lv}(t) - [D_0 + \tau\, \dot{s}_{lv}(t)],$$

with constants D_0 and τ and the position s_{lv} and velocity $\dot{s}_{lv}$ of the leading vehicle along the lane. As we would like to derive alternative trajectories to the vicinity of this point, the movement of the leading vehicle has to be predicted and we reasonably assume $\ddot{s}_{lv}(t) = \ddot{s}_{lv}(t_0) = \text{const}$. Time integration leads us to

$$\dot{s}_{lv}(t) = \dot{s}_{lv}(t_0) + \ddot{s}_{lv}(t_0)[t - t_0]$$
$$s_{lv}(t) = s_{lv}(t_0) + \dot{s}_{lv}(t_0)[t - t_0] + \frac{1}{2}\ddot{s}_{lv}(t_0)[t - t_0]^2,$$

which we need in the required time derivatives

$$\dot{s}_{\text{target}}(t) = \dot{s}_{lv}(t) - \tau \ddot{s}_{lv}(t),$$
$$\ddot{s}_{\text{target}}(t) = \ddot{s}_{lv}(t_1) - \tau \dddot{s}_{lv}(t) = \ddot{s}_{lv}(t_1).$$

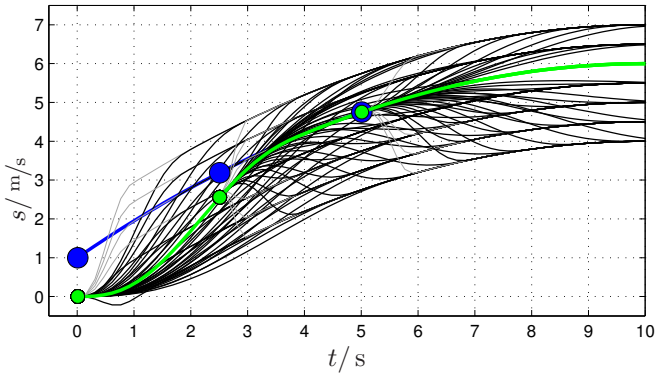

Fig. 4. Optimal longitudinal tracking of a target position in blue with green being the optimal trajectory, black the valid, and gray the invalid alternatives in each replanning step

Merging and Stopping

In the same fashion as above, we can define the target point

$$s_{\text{target}}(t) = \frac{1}{2}[s_a(t) + s_b(t)], \tag{3}$$

which enables us to position the autonomous car next to a pair of vehicles at $s_a(t)$ and $s_b(t)$, before squeezing slowly in between during a tight merging maneuver.

For stopping at intersections due to a red light or a stop sign, we define $s_{\text{target}} = s_{\text{stop}}$, $\dot{s}_{\text{target}} \equiv 0$, and $\ddot{s}_{\text{target}} \equiv 0$.

B. Velocity Keeping

In many situations, such as driving with no vehicles directly ahead, the autonomous car does not necessarily have to be at a certain position but needs to adapt to a desired velocity $\dot{s}_d = \text{const}$. given by the behavioral level. Analog to the calculus of variations in [15] (with the additional so-called *transversality condition* for s_1) and Prop. 1, quartic polynomials can be found to minimize the cost functional

$$C_v = k_j J_t(s(t)) + k_t T + k_{\dot{s}} [\dot{s}_1 - \dot{s}_d]^2$$

for a given start state $S_0 = [s_0, \dot{s}_0, \ddot{s}_0]$ at t_0 and $[\dot{s}_1, \ddot{s}_1]$ of the end state S_1 at some $t_1 = t_0 + T$. This means, that we can generate an optimal longitudinal trajectory set of quartic polynomials by varying the end constraints by $\Delta \dot{s}_i$ and T_j according to

$$[\dot{s}_1, \ddot{s}_1, T]_{ij} = [[\dot{s}_d + \Delta \dot{s}_i], 0, T_j],$$

as depicted in Fig. 5.

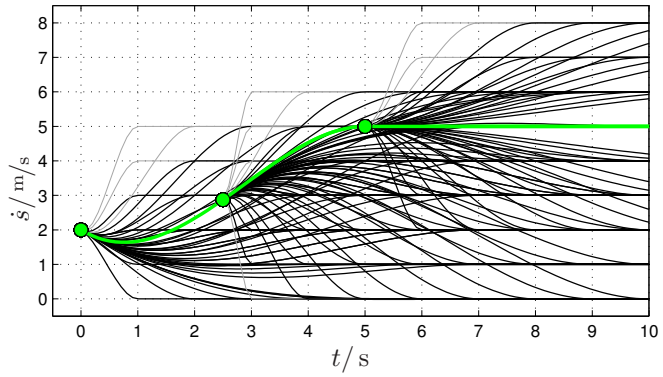

Fig. 5. Optimal velocity adaption to $\dot{s}_d = 5.0\,\mathrm{m/s}$ with green being the optimal trajectory, black the valid, and gray the invalid alternatives in each replanning step

VI. Combining Lateral and Longitudinal Trajectories

Before combining the lateral and longitudinal trajectory sets, denoted as $\mathcal{T}_{\text{lat}}$ and $\mathcal{T}_{\text{lon}}$ in the sequel, each one is checked against outsized acceleration values of $\ddot{s}$ and $\ddot{d}$ or d'' (gray trajectories in the figures of the previous section). As we do not concentrate in this contribution on maxing out the vehicle's physics, we choose them fairly conservative, leaving enough safety margin to the feedback controller. The remainders in each set are then brought together in every combination $\mathcal{T}_{\text{lat}} \times \mathcal{T}_{\text{lon}}$, as shown in Fig. 6.

Since the best valid trajectory describes the tracking reference for a feedback controller, we need to derive the higher order information of $\vec{x}(t)$, that is the heading $\theta_x(t)$, curvature $\kappa_x(t)$, velocity $v_x(t)$, and acceleration $a_x(t)$. As for most setups the center line is assumed not to be available in a closed form but represented by presampled curve points with orientation $\theta_r(s)$, curvature $\kappa_r(s)$, and change of curvature over arc length $\kappa_r'(s)$, the required interpolation makes it impossible to derive the higher order information numerically. The derivations of the required closed form transformations can therefore be found in App. I. The curvature $\kappa_x(t)$ is then used for excluding trajectories exceeding the maximum turn radius of the car. In a last step, the conjoint costs of each trajectory is calculated as the weighted sum $C_{\text{tot}} = k_{\text{lat}} C_{\text{lat}} + k_{\text{lon}} C_{\text{lon}}$.

As for collision detection, we would like to avoid adding heuristic penalty terms to the cost functionals in the vicinity of other obstacles, as they tend to lead to complex parameter adjustments as well as unpredictable behavior. Instead we add a certain safety distance to the size of our car on each side and make a hierarchical zero/one decision in terms of interference with other obstacles similar to [16]. Our solution to preventing the car from passing other obstacles *unnecessarily* closely without increasing the safety distance in general, is as simple as effective: The collision-checked contour is continuously expanded a little bit towards the time horizon, so obstacles of any kind seem to continuously back off as we get closer.

Every time we utilize a new reference as the center line, such as during initialization and lane changes, or when we switch between low and high speed trajectories, we have to project the current end point $(x, \theta_x, \kappa_x, v_x, a_x)(t_0)$ on the new center line and determine the corresponding $[s_0, \dot{s}_0, \ddot{s}_0, d_0, \dot{d}_0, \ddot{d}_0]$ or $[s_0, \dot{s}_0, \ddot{s}_0, d_0, d_0', d_0'']$ respectively. For this reason, the transformations in the appendix can easily be inverted in closed form, except for s_0, as we do not restrict the center line $\vec{r}(s)$ to a certain shape[6]. However the inversion can be restated as the minimization problem $s = \underset{\sigma}{\operatorname{argmin}} \|x - r(\sigma)\|$, for which efficient numerical methods exist.

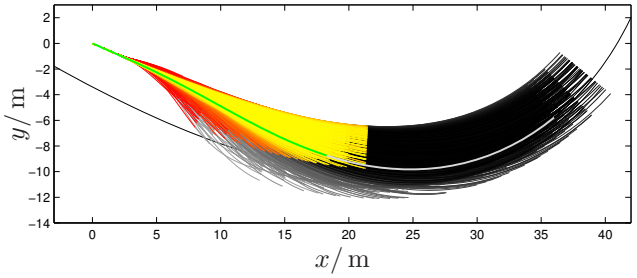

Fig. 6. Resulting trajectory set in global coordinates for velocity keeping: The color map visualizes the increasing costs of both the reactive layer with 3.0 s lookahead from red to yellow and the alternatives for the long-term objectives form gray to black. As there are no obstacles within the 3.0 s horizon, the optimal trajectory of the free problem is chosen (green, light gray), which leads the vehicle back to the center line and to the desired speed.

VII. Choosing the Right Strategy

As far as our experience goes, it is sufficient for highway trajectory generation to classify all traffic scenarios as *merging*, *following* another car, *keeping* a certain *velocity*, *stopping* at a certain point, and all combinations thereof, which are conflicting most of the time. In control theory, *override control* [6] is a well-known technique, which chooses among multiple control strategies according to a scheme, prevalently the most conservative one via a max- or a min-operator. This technique has been successfully implemented in numerous autonomous cars on the control level (Adaptive Cruise Control), but, to our best knowledge, not on the trajectory generation level, as we propose here.

At any time, the lateral trajectory set is combined with the ones of every active longitudinal trajectory generation mode according to Sec. VI. Then the collision-free trajectory with the lowest conjoint cost functionals C_{tot} of each active mode is compared to the other ones, and the trajectory with the smallest initial jerk value $\dddot{s}(t_0)$ is finally put through to the tracking controller. Typical combinations of active modes are *velocity keeping* and *following* (Adaptive Cruise Control, lane changes in sparce traffic), *merging* (lane changes in dense traffic), and *velocity keeping* and *stopping* (intersection with traffic lights).

VIII. Experiments

A first version of the algorithm was implemented and tested without obstacles (long-term objectives) on the au-

[6]For a straight line or a circular arc closed-form solutions exist.

tonomous vehicle JUNIOR with a planning cycle of 100 ms. The trajectories generated in combination with a tracking controller [8] a smooth, controlled ride with velocity invariant timing and the guaranteed temporal consistency as already shown in Fig. 6.

Due to the associated risk of the real world highway scenario shown in Fig. 7, we tested the reactive capabilities of the trajectory generation method in simulation and hold off on the practical validation. In order to show the functionality of the algorithm, we *disabled* the behavioral layer, which is otherwise meant to prevent the vehicle from solving critical situations at the reactive layer as often as possible. As a consequence, only the center line of the middle lane and a desired velocity $5.0\,\mathrm{m/s}$ higher than the avarage traffic flow is permanently put through to the algorithm: With the cost weights $k_{\mathrm{lat}} \approx k_{\mathrm{lon}}$, the car always drives well-behaved right behind the leading car (not shown), so, for the sake of clearness, we used $k_{\mathrm{lat}} \ll k_{\mathrm{lon}}$. With this, the car prefers passing the much slower vehicles, such as between $t = 0.0\,\mathrm{s}$ and $7.75\,\mathrm{s}$, to slowing down for them, but also provokes reckless driving between $t = 29.07\,\mathrm{s}$ and $50.0\,\mathrm{s}$, demonstrating the combined usage of steering and breaking/acceleration.

IX. Conclusion and Future Works

In order to handle dynamic street scenarios with an autonomous vehicle we proposed an optimal control based solution to the trajectory generation problem, which we illustrated by means of a conclusive experiment. The derived strategy realizes effectively all necessary maneuvers for on-road driving in the presence of dynamic and static obstacles. The resulting maneuvers are characterized by a consistent, effective, comfortable as well as safe integration into the permanently changing traffic flow and can be tuned by a small set of intuitive, orthogonal parameters. The trajectory generation can directly be embedded into the behavioral layer commanding abstract inputs to the algorithm, such as desired speed, the position of the car to follow, or lane change intents.

Due to the short optimization horizon, however, the approach is not intended to relieve the behavior layer of making farsighted decisions. New suitable heuristics, such as the constant-time-gap-law or conservative merge checks, can minimize the number of critical situations handled on the reactive level. The better the algorithm gets integrated into the behavioral layer and the more precisely other traffic participants can be predicted, the further we can advance with our test vehicle to open road traffic, which will reveal the level of acceptance there.

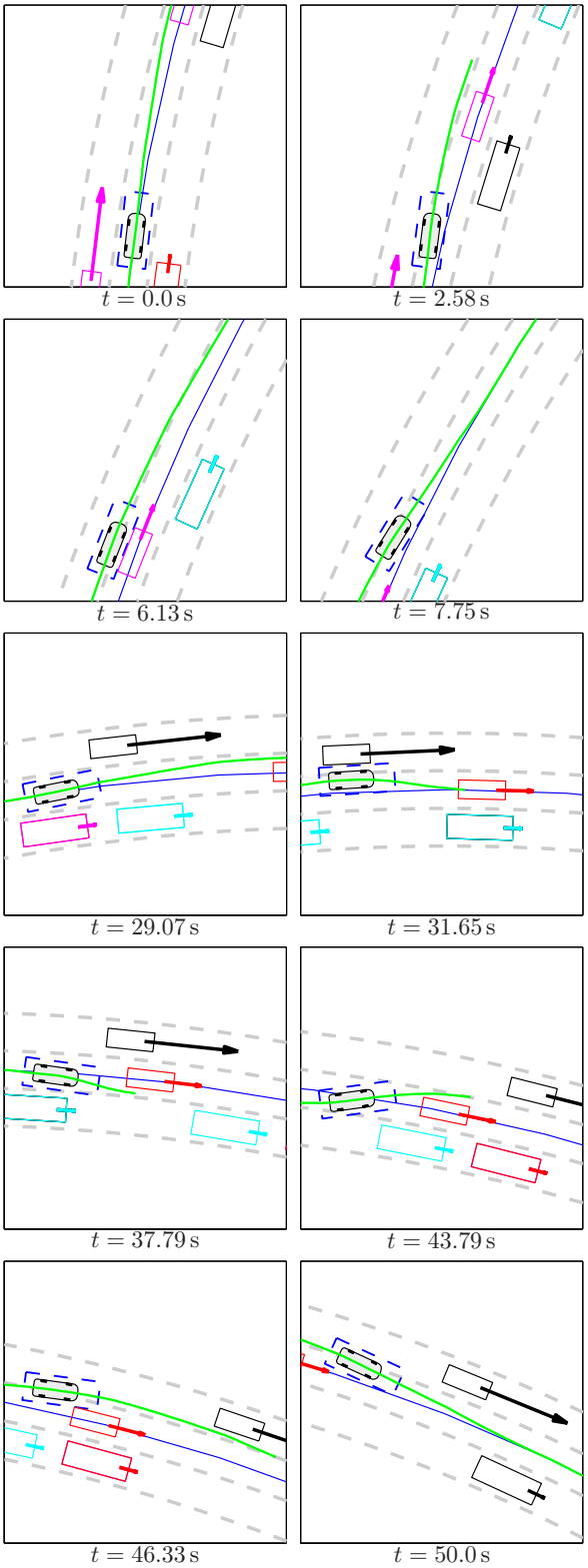

Fig. 7. Simulated highway scenario: The center line is given by the middle lane and the desired speed is significantly higher than the traffic flow. Two completely reactive passing maneuvers are shown between $t = 0.0\,\mathrm{s}$ and $7.75\,\mathrm{s}$, as well as between $t = 29.07\,\mathrm{s}$ and $50.0\,\mathrm{s}$ with predicted obstacle movements on $T_{\mathrm{hor}} \leq 3.0\,\mathrm{s}$.

References

[1] A. Bacha, C. Bauman, R. Faruque, M. Fleming, C. Terwelp, C. Reinholtz, D. Hong, A. Wicks, T. Alberi, D. Anderson, et al. Odin: Team VictorTango's entry in the DARPA Urban Challenge. *Journal of Field Robotics*, 25(8), 2008.

[2] B. Donald, P. Xavier, J. Canny, and J. Reif. Kinodynamic motion planning. *Journal of the ACM*, 40:1048–1066, November 1994.

[3] L.E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, pages 497–516, 1957.

[4] Paolo Fiorini and Zvi Shiller. Time optimal trajectory planning in dynamic environments. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1553–1558, 1996.

[5] L. Fletcher, S. Teller, E. Olson, D. Moore, Y. Kuwata, J. How, J. Leonard, I. Miller, M. Campbell, D. Huttenlocher, et al. The MIT-Cornell collision and why it happened. *Journal of Robotic Systems*, 25(10):775–807, 2008.

[6] A.H. Glattfelder, W. Schaufelberger, and H. Fässler. Stability of override control systems. *International Journal of Control*, 37(5):1023–1037, 1983.

[7] Alonzo Kelly and Bryan Nagy. Reactive nonholonomic trajectory generation via parametric optimal control. *International Journal of Robotics Research*, 22(7-8):583–602, 2003.

[8] J.Z. Kolter, C. Plagemann, D.T. Jackson, A.Y. Ng, and S. Thrun. A probabilistic approach to mixed open-loop/closed-loop control, with application to extreme autonomous driving. In *International Conference on Robotics and Automation*, 2010.

[9] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How. Motion planning for urban driving using RRT. In *International Conference on Intelligent Robots and Systems*, 2008.

[10] M. LaValle. Rapidly-exploring random trees: A new tool for path planning. Technical report, Computer Science Dept., Iowa State University, 1998.

[11] M. LaValle and J. J. Kuffner. Randomized kinodynamic planning. In *International Conference on Robotics and Automation*, 1999.

[12] P. Manfredo. *Differential geometry of curves and surfaces*. Prentice-Hall Englewood Cliffs, NJ, 1976.

[13] J.A. Reeds and L.A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2):367–393, 1990.

[14] A. Schwarzenegger. *California Driver Handbook*. Department of Motor Vehicles, 2007.

[15] A. Takahashi, T. Hongo, Y. Ninomiya, and G. Sugimoto. Local path planning and motion control for AGV in positioning. In *IEEE/RSJ International Workshop on Intelligent Robots and Systems' 89. The Autonomous Mobile Robots and Its Applications. IROS'89. Proceedings.*, pages 392–397, 1989.

[16] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, MN Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, et al. Autonomous driving in urban environments: Boss and the Urban Challenge. *Journal of Field Robotics*, 25(8), 2008.

[17] Michiel van Nieuwstadt and Richard M. Murray. Real time trajectory generation for differentially flat systems. *Int. Journal of Robust and Nonlinear Control*, 8:995–1020, 1996.

[18] M. Werling, T. Gindele, D. Jagszent, and L. Gröll. A robust algorithm for handling moving traffic in urban scenarios. In *IEEE Intelligent Vehicles Symposium 2008, Eindhoven, The Netherlands*, 2008.

[19] J. Ziegler and C. Stiller. Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.

[20] J. Ziegler, M. Werling, and J. Schröder. Navigating car-like robots in unstructured environments using an obstacle sensitive cost function. In *IEEE Intelligent Vehicles Symposium 2008, Eindhoven, The Netherlands*, 2008.

APPENDIX I
TRANSFORMATIONS FROM FRENÉT COORDINATES TO GLOBAL COORDINATES

In addition to (1), we seek for transformations

$$[s, \dot{s}, \ddot{s}; d, \dot{d}, \ddot{d}/d, d', d''] \mapsto [\vec{x}, \theta_x, \kappa_x, v_x, a_x]$$

The mayor challenges at this is to handle the singularity $v_x = 0$. Therefore we introduce $\vec{t}_r(s) := \begin{bmatrix} \cos\theta_r(s) & \sin\theta_r(s) \end{bmatrix}^{\mathrm{T}}$ and $\vec{n}_r(s) := \begin{bmatrix} -\sin\theta_r(s) & \cos\theta_r(s) \end{bmatrix}^{\mathrm{T}}$, where $\theta_r(s)$, $\vec{t}_r(s)$ and $\vec{n}_r(s)$ are the orientation, the tangential and normal vectors of the center line in s. In addition, we denote the curvature as κ_r and assume that we travel along the center line excluding extreme situations, such that $\|\Delta\theta\|_2 < \frac{\pi}{2}$, with $\Delta\theta := \theta_x - \theta_r$, and $1 - \kappa_r d > 0$ at all times. As we can derive the transformation needed for higher speeds from the one associated with lower speeds, we will start with the latter. With (1) we get

$$d = [\vec{x} - \vec{r}(s)]^{\mathrm{T}} \vec{n}_r. \tag{4}$$

Time derivative yields with $\dot{\vec{n}}_r = -\kappa_r \vec{t}_r$

$$\begin{aligned} \dot{d} &= [\dot{\vec{x}} - \dot{\vec{r}}(s)]^{\mathrm{T}} \vec{n}_r + [\vec{x} - \vec{r}(s)]^{\mathrm{T}} \dot{\vec{n}}_r \\ &= v_x \vec{t}_x^{\mathrm{T}} \vec{n}_r - \dot{s} \underbrace{\vec{t}_r^{\mathrm{T}} \vec{n}_r}_{=0} - \kappa_r \underbrace{[\vec{x} - \vec{r}(s)]^{\mathrm{T}} \vec{t}_r}_{=0} = v_x \sin\Delta\theta. \end{aligned} \tag{5}$$

Therefore, we calculate

$$\begin{aligned} v_x = \|\dot{x}\|_2 &= \left\| \begin{bmatrix} \vec{t}_r & \vec{n}_r \end{bmatrix} \begin{bmatrix} 1-\kappa_r d & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{s} \\ \dot{d} \end{bmatrix} \right\|_2 \\ &= \left\| \begin{bmatrix} 1-\kappa_r d & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{s} \\ \dot{d} \end{bmatrix} \right\|_2 = \sqrt{[1-\kappa_r d]^2 \dot{s}^2 + \dot{d}^2} \end{aligned}$$

$$\begin{aligned} \text{and} \quad d' := \frac{\mathrm{d}}{\mathrm{d}s} d &= \frac{\mathrm{d}t}{\mathrm{d}s} \frac{\mathrm{d}}{\mathrm{d}t} d = \frac{1}{\dot{s}} \dot{d} = \frac{1}{\dot{s}} v_x \sin\Delta\theta \\ &= \sqrt{[1-\kappa_r d]^2 + d'^2} \sin\Delta\theta \\ d'^2 &= [[1-\kappa_r d]^2 + d'^2] \sin^2\Delta\theta \\ d'^2 [1 - \sin^2\Delta\theta] &= [1-\kappa_r d]^2 \sin^2\Delta\theta, \end{aligned}$$

$$\text{so that we get} \quad d' = [1-\kappa_r d] \tan\Delta\theta. \tag{6}$$

Additionally, we know that $[\vec{x} - \vec{r}]^{\mathrm{T}} \vec{t}_r = 0$ at all times, so that differentiating with respect to time gives us analog to (5) $\frac{v_x}{\dot{s}} \cos\Delta\theta - 1 + \kappa_r d = 0$ and we can solve for the velocity

$$v_x = \dot{s} \frac{1-\kappa_r d}{\cos\Delta\theta}. \tag{7}$$

With this and s_x being the arc length of the trajectory $\vec{x}$, we can conclude that

$$\frac{\mathrm{d}}{\mathrm{d}s} = \frac{\mathrm{d}s_x}{\mathrm{d}s} \frac{\mathrm{d}}{\mathrm{d}s_x} = \frac{\mathrm{d}s_x}{\mathrm{d}t} \frac{\mathrm{d}t}{\mathrm{d}s} \frac{\mathrm{d}}{\mathrm{d}s_x} = \frac{v_x}{\dot{s}} \frac{\mathrm{d}}{\mathrm{d}s_x} = \frac{1-\kappa_r d}{\cos\Delta\theta} \frac{\mathrm{d}}{\mathrm{d}s_x}, \tag{8}$$

so that we calculate the second derivative of d to be

$$\begin{aligned} d'' &= -[\kappa_r d]' \tan\Delta\theta + \frac{1-\kappa_r d}{\cos^2\Delta\theta} \left[\frac{\mathrm{d}}{\mathrm{d}s} \theta_x - \theta_r' \right] \\ &= -[\kappa_r' d + \kappa_r d'] \tan\Delta\theta + \frac{1-\kappa_r d}{\cos^2\Delta\theta} \left[\kappa_x \frac{1-\kappa_r d}{\cos\Delta\theta} - \kappa_r \right]. \end{aligned} \tag{9}$$

Equations (6) and (9) can be solved for θ_x and κ_x, including $v_x = 0$. Time differentiating the velocity once more yields the last unknown in our transformation

$$\begin{aligned} a_x := \dot{v}_x &= \ddot{s} \frac{1-\kappa_r d}{\cos\Delta\theta} + \dot{s} \frac{\mathrm{d}}{\mathrm{d}s} \frac{1-\kappa_r d}{\cos\Delta\theta} \dot{s} = \\ &\ddot{s} \frac{1-\kappa_r d}{\cos\Delta\theta} + \frac{\dot{s}^2}{\cos\Delta\theta} \left[[1-\kappa_r d] \tan\Delta\theta \Delta\theta' - [\kappa_r' d + \kappa_r d'] \right]. \end{aligned}$$

For high speed driving we calculate $\dot{d} = \frac{\mathrm{d}}{\mathrm{d}t} d = \frac{\mathrm{d}s}{\mathrm{d}t} \frac{\mathrm{d}}{\mathrm{d}s} d = \dot{s}\, d$ and $\ddot{d} = d'' \dot{s}^2 + d' \ddot{s}$. As $\dot{s} \neq 0$ holds for higher speeds, subsequently solving these equations for d' and d'' enables us to use the previously calculated transformations. Notice, that the center line $\vec{r}(s)$ needs to have a continuous change of curvature κ_r' in order to provide for a trajectory $\vec{x}(t)$ with a continuous κ_x.