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

Spatiotemporally Consistent Smooth Speed Profiles for Autonomous Driving

Benjamin C. Heinrich, Alexander Frericks and Hans-Joachim Wuensche

Abstract— **Graph-based trajectory planning algorithms such as A*, Dijkstra or RRT use motion primitives to generate obstacle-free trajectories. A common method is the use of piecewise constant acceleration profiles along these primitives to avoid high computational complexity. However, the resulting trajectories can be uncomfortable to drive for the passengers due to the abrupt changes in the acceleration.**

In this paper, we present three different methods to smoothen given trajectories which are discontinuous in their acceleration profile, while complying with the given lengths as well as times. Due to the latter, safety is still guaranteed while the driving comfort is increased. Our approach allows to decrease the complexity of the trajectory-generation problem by relaxing the driving-comfort requirements on the initial creation and shifting it to a second smoothing step.

I. Introduction

For autonomous driving, it is safety critical to react to changes in the environment as fast as possible. One time-consuming step from sensing to acting is the planning of trajectories. This is especially the case in dynamically changing environments.

We speak of static environments when the occupancy of the vehicle's surrounding does not change over time. There, comfortable driving can be guaranteed by planning a path and then determining the speeds at each path point in a second step. This is called a path-velocity decomposition [1].

In contrast, a dynamic environment describes a surrounding where the occupancy changes over time. Here, both static and dynamic obstacles have to be avoided. This is only possible when the planner decides where the ego vehicle is situated at which point in time.

It is a challenge to determine which homotopy to use. A homotopy is a solution set, e. g., overtake before or after the oncoming car passed or pass the crossing before or after the cross traffic. Thus, often simple speed profiles, such as constant speed or constant acceleration, are used in order to reduce the computational complexity. This comes at the cost of driving comfort. Hence, we propose to split the trajectory planning into two parts:

- Find a trajectory using simple motion primitives and provide times for important sample points.
- Ensure comfortable driving by adapting the speed profile without altering the given times.

This can be seen as a variation of the classic path-velocity decomposition. The added constraint is that we need to be *spatiotemporally consistent*. I. e., the times at which important path points are reached may not be changed.

All authors are with the Institute for Autonomous Systems Technology (TAS) of the Universität der Bundeswehr Munich, Neubiberg, Germany. Contact author email: benjamin.heinrich@unibw.de

The main motivation for the proposed decomposition is to increase the driving comfort without slowing down the overall trajectory generation. Therefore, it is important that the speed-profile generating algorithm is as fast as possible. Note that every millisecond the algorithm runs adds to the overall reaction time of the system.

This paper is structured as follows: First, we give a short overview regarding speed-profile generation literature in section II. Then, we give the prerequisites used throughout the paper and a simple speed-planning heuristic as performance baseline in section III and section IV, respectively. Furthermore, we will present two optimization-based spatiotemporally consistent speed-profile planners in section V and section VI. Finally, the given planners are evaluated in section VII and then discussed in the concluding section VIII.

II. Related Work

There exists a sizable literature on speed-profile planning. For reasonable control performance (e. g., to not loose traction) and for increased driving comfort, it is important to consider bounds on the possible jerk and maximal allowed accelerations. An example for time-optimal speed profile computation can be found in [2]. Computation times varied but averaged 200 ms for different trajectories of about 14 m lengths. As the time is the minimization objective here, those speed profiles are not spatiotemporally consistent.

Another popular topic is energy minimization. For these approaches, models of the engine and track [3] or environmental conditions [4] are used in order to find an optimal solution. However, this usually does not consider temporal factors.

For obvious reasons, dynamic environments are handled when it comes to planning traffic at roundabouts or intersections. An example for speed-profile rather than trajectory planning is [5], where safety is considered for two vehicles crossing at an intersection.

Driving comfort was explicitly considered in [6]. Here, quintic Bézier curves where iteratively optimized for generating a smooth speed profile. The results show a high degree of smoothness and consider lateral as well as longitudinal acceleration limits. Though no planning times are given, due to the simplicity of the algorithm it should perform reasonably fast. However, again, the algorithm is only useful in a classic path-velocity decomposition, as it is not spatiotemporally consistent.

An optimization approach for spatiotemporal speed-profile planning was given in [7]. Here, the assumption is that a certain path has to be followed. In addition, bounds on the path-point times are given regarding a predetermined

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

homotopy. The objective of the proposed algorithm is to optimize the times on the given path. In a way, this is an extension of the proposed decomposition, as it leaves the algorithm more freedom to obtain optimality. This freedom comes at a computational cost: The computation times, using Matlab, were in the hundreds of milliseconds.

A similar problem was solved using an A*-based approach in [8]. The computation times were highly dependent on the planning horizon, but for a 10 s horizon, about 10 ms are given as a worst case scenario. Due to the nature of the A* algorithm, the proposed algorithm should be able to find the best homotopy by itself. The major drawback of this algorithm is the generation of only piecewise constant-acceleration profiles, which are not comfortable to drive. Worth mentioning is also [9], were the problem is solved time-optimal and even faster, but again, only constant-acceleration profiles are generated.

None of the above mentioned publications find spatiotemporally consistent speed profiles. Hence, to the best of our knowledge, this kind of trajectory smoothing has not been attempted in the field of autonomous driving.

III. Prerequisites

Our goal in this paper is to smooth a given trajectory in order to increase the driving comfort for the passengers, while still remaining spatiotemporally consistent. In our case, the original trajectory is computed by an A*-based planning algorithm (see [10], [11]). The given trajectory consists of N concatenated clothoid segments and a set of desired time points. The latter are generated using a piecewise constant acceleration profile with one value per arc (see fig. 1 for an illustration). Henceforth, the original trajectory is given as a sequence of the 3-tuples

$$\mathbf{r}_{[i]} = (l_i^*, t_i^*, c_i^*), \qquad i \in \{0, \dots, N\} \,, \tag{1}$$

where:

l_i^* length at beginning of i^{th} clothoid segment,
t_i^* time when to reach the beginning of the segment,
c_i^* initial curvature of the clothoid segment.

Our premise is that constant acceleration profiles are not comfortable to drive, due to their discontinuities in the acceleration. Our main objective is to generate a new smooth speed profile with at least continuous acceleration. For this, we use piecewise polynomials $l_i(t)$, $v_i(t)$ and $a_i(t)$ for the path length, speed and acceleration at time t on segment i, respectively.[1] Thus, we required $\forall i \in \{0, \dots, N-2\}$:

$$l_i(t_{i+1}^*) = l_{i+1}(t_{i+1}^*) = l_{i+1} \tag{2}$$
$$v_i(t_{i+1}^*) = v_{i+1}(t_{i+1}^*) = v_{i+1} \tag{3}$$
$$a_i(t_{i+1}^*) = a_{i+1}(t_{i+1}^*) = a_{i+1} \,. \tag{4}$$

The provided original trajectory $\mathbf{r}$ is planned to be collision-free considering other dynamic objects in the environment.

[1] Note the slight abuse of notation: On the one hand, v_i^* is the reference value of the speed at $\mathbf{r}_{[i]}$ and v_i the actual value there. On the other hand, $v_i(t)$ is the function of the speed on the i^{th} segment, i. e., for $t \in [t_i^*, t_{i+1}^*]$.

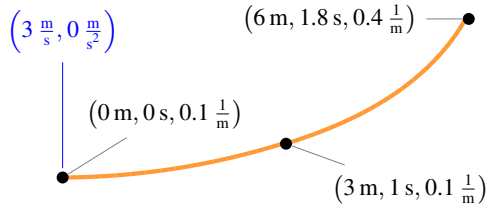

Fig. 1: Illustration of a reference sequence (orange). The reference points are a (length, time,curvature)-tuple (black). Additionally, starting speed and acceleration are given (blue)

Thus, it is required for the resulting smoothed trajectory to maintain the original spatiotemporal mapping

$$l(t_i^*) = l_i^* \quad \forall i \in \{0, \dots, N\} \,. \tag{5}$$

Furthermore, the two presented optimization approaches additionally minimize the resulting jerk as well as the steering rate. Utilizing the kinematic bicycle model and the small-angle approximation, the steering rate $\dot{\delta}$ for a vehicle moving along a path can be computed as

$$\dot{\delta} = \mathring{c} v w \,, \tag{6}$$

where:

$\mathring{c}$ change of curvature c w. r. t. the curve's length,
w vehicle's wheelbase.

As our given trajectory consists of clothoid arcs, the change in curvature on each segment is constant and simply computed as

$$\mathring{c}_i = \frac{c_{i+1}^* - c_i^*}{l_{i+1}^* - l_i^*} \,. \tag{7}$$

Moreover, we use the common simplified lateral acceleration

$$a_{\text{lat}} = c v^2 \,. \tag{8}$$

IV. A Simple Heuristic

As a performance baseline, we introduce a spatiotemporally consistent speed-profile generation heuristic. Given the original trajectory $\mathbf{r}$, this sub-optimal heuristic provides a continuous-acceleration speed profile that respects the reference at virtually no computation cost. For notational convenience, let

$\Delta l_i = l_{i+1}^* - l_i^*$ length of segment i,
$\Delta t_i = t_{i+1}^* - t_i^*$ duration of a that segment and
$\tau_i = t - t_i^*$ elapsed time on that segment.

Continuity in the acceleration $a(t)$ is guaranteed when we use a polynomial for the jerk $j(t)$, and deduce $a(t)$, $v(t)$ as well as $l(t)$ from integration. Let

$$j_i(t) = \alpha_i \tau_i + \beta_i \tau_i^2 + \gamma_i \tau_i^3 \,, \tag{9}$$

where we skipped a constant term, as we enforce zero jerk at all segment borders, i. e.,

$$j_i(t_i^*) = j_i(t_{i+1}^*) = 0 \,. \tag{10}$$

Hence, the resulting speed profile is also continuous in the jerk. Using eq. (4) and eq. (10), we solve for the parameters

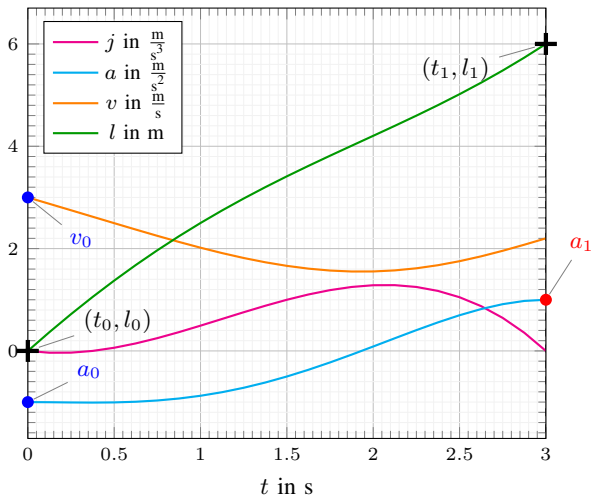

Fig. 2: Typical transition from proposed baseline. Given points from $\mathbf{r}$ are marked with black crosses. The starting condition v_0, a_0 are shown as blue dots. The heuristically chosen value a_1 is shown as red dot.

$\alpha_i, \beta_i, \gamma_i$ and get

$$\alpha_i = -12\frac{-10\Delta l_i + \Delta t_i(4a_i\Delta t_i + a_{i+1}\Delta t_i + 10v_i)}{\Delta t_i^4} \quad (11)$$

$$\beta_i = 12\frac{-30\Delta l_i + \Delta t_i(11a_i\Delta t_i + 4a_{i+1}\Delta t_i + 30v_i)}{\Delta t_i^5} \quad (12)$$

$$\gamma_i = -12\frac{-20\Delta l_i + \Delta t_i(7a_i\Delta t_i + 3a_{i+1}\Delta t_i + 20v_i)}{\Delta t^6}\,. \quad (13)$$

Note that l_0, v_0 and a_0 are assumed given. Hence, starting at the 0^{th} segment, we can deduce v_i for $i \in \{1, \dots, N\}$, if we choose a_i.[2]

Given $\mathbf{r}$, we first obtain

$$\tilde{v}_i = 2\frac{\Delta l_i}{\Delta t_i} - \tilde{v}_{i-1}\,, \quad (14)$$

i. e., the speeds a constant-acceleration profile would have. Then, we choose

$$a_i = \begin{cases} \frac{\tilde{v}_{i+1} - \tilde{v}_{i-1}}{\Delta t_{i+1} + \Delta t_i} & \forall i \in \{1, \dots, N-1\} \\ \frac{\tilde{v}_i - \tilde{v}_{i-1}}{\Delta t_i} & i = N \end{cases}\,. \quad (15)$$

A depiction of a typical baseline transition can be found in fig. 2.

A drawback of this approach is that it cannot be guaranteed that

$$\forall t : v(t) > 0\,. \quad (16)$$

I. e., unintended back and forth maneuvering may result in fringe cases, for instance at very small speeds and large desired acceleration transitions. Also, neither steering-rate nor lateral-acceleration restrictions are considered. In our case, where the reference $\mathbf{r}$ is generated considering those constraints including a small margin, this is not an issue.

[2]Note that we have one more parameter than condition. Thus, we have to choose either v_i or a_i to be able to compute the coefficients.

V. Basic Optimization Approach

We use an optimization approach to determine more optimal values for a_i. This approach allows to introduce bounds on the speed, lateral acceleration and steering rate on the trajectory. Our problem definition results in a quadratic optimization problem with linear inequality constraints.

Let our optimization variables be

$$\mathbf{x} = \begin{bmatrix} a_1 & a_2 & \dots & a_N \end{bmatrix}^\mathsf{T}\,. \quad (17)$$

Applying eqs. (11) to (13), the coefficients of the polynomials can be computed as functions of the optimization vector $\mathbf{x}$:

$$\alpha_i = \alpha_i(\mathbf{x})\,, \quad \beta_i = \beta_i(\mathbf{x})\,, \quad \gamma_i = \gamma_i(\mathbf{x}) \quad (18)$$

Furthermore, let the objective function be

$$J = \frac{\sum_{i=0}^{N-1} \int_{t_i^*}^{t_{i+1}^*} k_j j_i^2(t, \mathbf{x}) + k_\delta \mathring{c}_i^2 v_i^2(t, \mathbf{x})\,\mathrm{d}t}{t_N^* - t_0^*}\,, \quad (19)$$

where

k_j weighting factor for longitudinal jerk,
k_δ weighting factor for steering rate

and we applied the steering-rate approximation eq. (6).

Since the sample points t_i^* are fixed, the objective is a function of the optimization variables $\mathbf{x}$ only. In addition, the resulting objective function can be expressed in the following quadratic form after the evaluation of the integral terms:

$$J = \frac{1}{2}\mathbf{x}^\mathsf{T}\,\mathbf{H}(t_0^*, t_1^*, \dots, t_N^*)\,\mathbf{x} + \mathbf{f}^\mathsf{T}(t_0^*, t_1^*, \dots, t_N^*)\,\mathbf{x}\,. \quad (20)$$

Note that for a fixed number of sample points t_i^*, the expressions for $\mathbf{H}$ and $\mathbf{f}$ can can be derived offline. Therefore, the online evaluation of the cost functional, given $\mathbf{r}$, becomes very fast.

To guarantee eq. (16), i. e., in order to avoid undesired back and forth maneuvers, we introduce the constraint

$$v(t_k, \mathbf{x}) \geq 0\,, \quad (21)$$

where we equidistantly sample $t_k \in [0, t_N^*]$.[3]

In addition, we require the resulting steering rate and the resulting lateral acceleration to bounded. For the steering rate, we use

$$\left|\dot{\delta}(t_k, \mathbf{x})\right| \leq \dot{\delta}_{\max} \quad (22)$$

$$\Rightarrow w\,|\mathring{c}|\,v(t_k, \mathbf{x}) \leq \dot{\delta}_{\max} \quad (23)$$

$$\Leftrightarrow v(t_k, \mathbf{x}) \leq \frac{\dot{\delta}_{\max}}{w\,|\mathring{c}|}\,. \quad (24)$$

Note, that we already bounded the minimum speed to zero (see eq. (21)). Thus, only the upper bound for the speed has to be considered for the limit of the steering rate.

Similarly, we define the following inequality constraints

[3]Note that for simplicity, we only consider forward driving in this paper. However, the approaches presented here can easily be adapted for the case of backward driving.

to bound the lateral acceleration:

$$|a_{\text{lat}}(t_k, \mathbf{x})| \leq a_{\text{lat,max}} \tag{25}$$

$$\Rightarrow |c(l(t_k, \mathbf{x}))|\, v^2(t_k, \mathbf{x}) \leq a_{\text{lat,max}} \tag{26}$$

$$\Leftrightarrow v(t_k, \mathbf{x}) \leq \sqrt{\frac{a_{\text{lat,max}}}{|c(l(t_k, \mathbf{x}))|}}\,. \tag{27}$$

Equation (27) depicts a nonlinear inequality constraint, which would drastically increase the complexity of the optimization problem. Thus, we use the following simplification to linearize our problem:

$$v(t_k, \mathbf{x}) \leq \sqrt{\frac{a_{\text{lat,max}}}{|c_{i,\text{max}}|}}\,, \tag{28}$$

where $c_{i,\text{max}}$ is the maximum curvature of the i^{th} segment.

Equation (24) and eq. (28) can be merged into one single inequality constraint

$$v(t_k, \mathbf{x}) \leq \min\left(\frac{\dot{\delta}_{\text{max}}}{w\,|\mathring{c}|}, \sqrt{\frac{a_{\text{lat,max}}}{|c_{i,\text{max}}|}}\right)\,. \tag{29}$$

Due to the linearity of polynomials in their coefficients, the inequality constraints can be written as the following matrix inequality:

$$\mathbf{A}_{\text{ineq}} \cdot \mathbf{x} \leq \mathbf{b}_{\text{ineq}}\,. \tag{30}$$

Equation (19) and eq. (30) form a quadratic optimization problem with linear inequality constraints. This can be solved efficiently with a variety of algorithms and tools.

VI. Generalized Polynomial Parameter Optimization

In a further step, we generalize the optimization approach from the previous section. We define a polynomial of arbitrary degree M for the length parameter in each segment. From there, speed, acceleration and jerk can directly be derived as:

$$l_i(t) = \sum_{k=0}^{M} p_{i,k}\tau_i^k \tag{31}$$

$$\Rightarrow v_i(t) = \sum_{k=1}^{M} k \cdot p_{i,k}\tau_i^{k-1} \tag{32}$$

$$\Rightarrow a_i(t) = \sum_{k=2}^{M} k \cdot (k-1) \cdot p_{i,k}\tau_i^{k-2} \tag{33}$$

$$\Rightarrow j_i(t) = \sum_{k=3}^{M} k \cdot (k-1) \cdot (k-2) \cdot p_{i,k}\tau_i^{k-3}\,. \tag{34}$$

Similar to the optimization approach from the previous section, we define the coefficients of the polynomials as the optimization variables:

$$\mathbf{x}_{\text{g}} = \begin{bmatrix} p_{0,0} & p_{0,1} & \dots & p_{i,k} & \dots & p_{N,M} \end{bmatrix}^{\mathsf{T}}\,. \tag{35}$$

Let the objective function be the same as above, i. e., eq. (19). Written in quadratic form (see eq. (20)), we get:

$$J = \frac{1}{2}\mathbf{x}_{\text{g}}^{\mathsf{T}}\mathbf{H}_{\text{g}}\mathbf{x}_{\text{g}} + \mathbf{f}_{\text{g}}^{\mathsf{T}}\mathbf{x}_{\text{g}}\,. \tag{36}$$

However, in contrast to the previous approach, we have to define equality constraints to satisfy the continuity requirements at the sample points (see eqs. (2) to (4)). These can be written as a linear matrix equation:

$$\underbrace{\begin{bmatrix} 1 & \Delta t_1 & \Delta t_1^2 & \cdots & 1 & 0 & \dots \\ 0 & 1 & 2\Delta t_1 & \cdots & 0 & -1 & \dots \\ 0 & 0 & 2 & \cdots & 0 & 0 & \dots \\ & & & \ddots & & & \end{bmatrix}}_{\mathbf{A}_{\text{g,eq}}} \cdot \mathbf{x}_{\text{g}} = \mathbf{0}\,. \tag{37}$$

Again, we bound the resulting steering rate $\dot{\delta}$ and the resulting lateral acceleration a_{lat} to some maximum values using eqs. (24) and (28) and get:

$$\mathbf{A}_{\text{g,ineq}} \cdot \mathbf{x}_{\text{g}} \leq \mathbf{0}\,. \tag{38}$$

Similar to the previous section, eqs. (36) to (38) form a constrained quadratic optimization problem with additional equality constraints.

VII. Evaluation

All results were produced on an Intel i7-3770K CPU 3.50 GHz with 24 GB of RAM. The code was written and executed using MATLAB 2017b, the solver for the constrained optimization problem was `quadprog` with no further parameters.

The following parameters were used for all tests:

$$w = 2.855\,\text{m} \qquad a_{\text{lat,max}} = 2.3\,\tfrac{\text{m}}{\text{s}^2} \qquad k_a = 1.0$$

$$T_s = 0.2\,\text{s} \qquad \dot{\delta}_{\text{max}} = 0.65\,\tfrac{\text{rad}}{\text{s}} \qquad k_\delta = 1.0\,,$$

where T_s is the sample distance for the inequality constraints eqs. (21) and (29).

We evaluate two scenarios (see fig. 3):

- In *case 1*, a simple 90° left turn, where we start at 3 $\frac{\text{m}}{\text{s}}$ and try to accelerate as much as possible.
- In *case 2*, a randomly-generated more challenging chicane, where we start at 6 $\frac{\text{m}}{\text{s}}$ and stop at the end.
- In *subcase a*, we used the trajectory kink points (the points where the $\mathring{c}$ changes) for determining $\mathbf{r}$.
- In *subcase b*, we set a maximum on Δl of 1 m.

Additionally, we evaluated the computational cost of activating the inequality constraints eqs. (24) and (28). Note that in our case, the reference already respects those values, with a 15 % margin.

In figs. 4 and 5, we show plots for both cases. Polynomial degrees greater 5 were not shown as the difference in the curves is minor. In table I, we show numerical results up to 6^{th} order, as the cost reduction begins to stagnate there.

The choice of sample points determines how tight the initial speed profile is followed. The smoothing potential, however, is greater with fewer samples (see fig. 5 for a side-by-side comparison). An increased number of samples is, of course, also reflected in the computation time. Thus, it is recommended to only use critical points, for instance those directly at an intersection, as reference and not oversample uncritical parts of the trajectory.

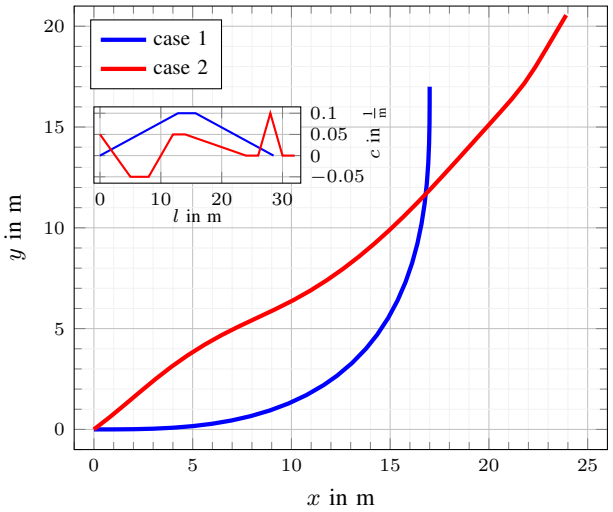

Fig. 3: Test scenarios: A left turn (case 1, drawn blue) and a chicane (case 2, red). The inlay shows the curvatures over the path length.

However, too few points can result in an infeasible problem, as can be seen from table I (lower right quadrant). This is due to the inequality constraints using conservative approximations on the lateral acceleration.

The differences between the algorithms can best be seen in the jerk plots. Here, the heuristic and the basic optimization approach both return to zero at the sample points. Furthermore, the maximum needed jerk decreases with higher degrees.

TABLE I: Computation time vs optimality

Case 1

	time (ms)	cost J	time (ms)	cost J
unconstrained	case $1a: N = 29$		case $1b: N = 4$	
heuristic	0.014	25.52	0.002	2.79
basic	2.002	19.70	2.000	0.84
poly_4	5.463	17.09	2.402	0.72
poly_5	5.736	16.90	2.584	0.50
poly_6	6.660	16.90	2.512	0.50
constrained	case $1a: N = 29$		case $1b: N = 4$	
basic	4.893	36.679	–	–
poly_4	11.123	20.367	–	–
poly_5	12.861	19.206	–	–
poly_6	15.684	18.309	–	–

Case 2

	time (ms)	cost J	time (ms)	cost J
unconstrained	case $2a: N = 33$		case $2b: N = 10$	
heuristic	0.015	307.24	0.004	49.74
basic	2.288	192.05	2.001	21.87
poly_4	3.989	170.85	2.527	20.24
poly_5	4.618	167.04	2.697	19.16
poly_6	5.082	167.04	2.836	19.16
constrained	case $2a: N = 33$		case $2b: N = 10$	
basic	4.714	210.48	3.464	21.87
poly_4	18.780	184.06	7.466	20.24
poly_5	20.660	174.65	7.728	19.16
poly_6	24.190	173.27	8.408	19.16

Further experiments showed that additional polynomial degrees sometimes allow for feasible solutions where lower degrees fail. However, in those cases, often very large accelerations are reached and hence it is preferable to just use a discontinuous acceleration profile.

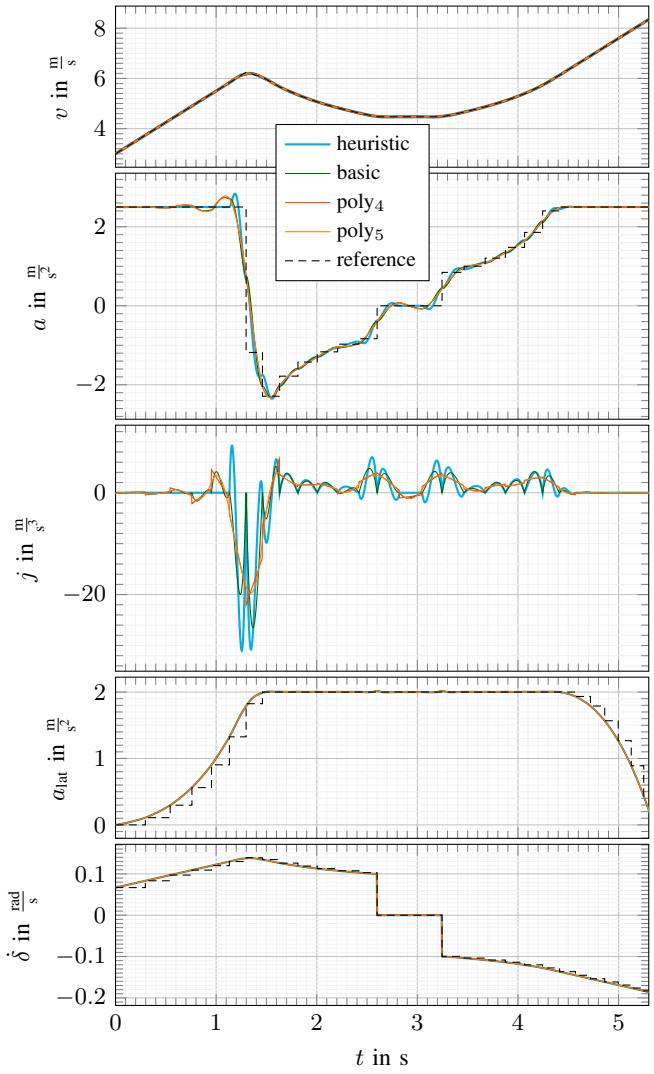

Fig. 4: Results for test case 1b

VIII. Conclusion

In this paper, we presented three algorithms for spatiotemporally consistent speed-profile smoothing. This means, we smooth a speed profile while respecting a sequence of given reference length-time tuples.

We want to speed up trajectory generation by decomposing the process into two phases: A first phase, where a best trajectory (e. g., in terms of the best homotopy) is found without emphasis on the driving comfort. And a second phase, where we smooth its speed profile using one of the proposed methods and thus achieve a smoother driving experience.

The methods are

- a simple heuristic as baseline, which is instantaneously calculable but sub-optimal and does not respect constraints
- the optimization of the heuristic, which is a trade-off of speed vs optimality and
- a generalized polynomial parameter optimization, which, due to the problems structure, provides an optimum w. r. t. the chosen cost functional.

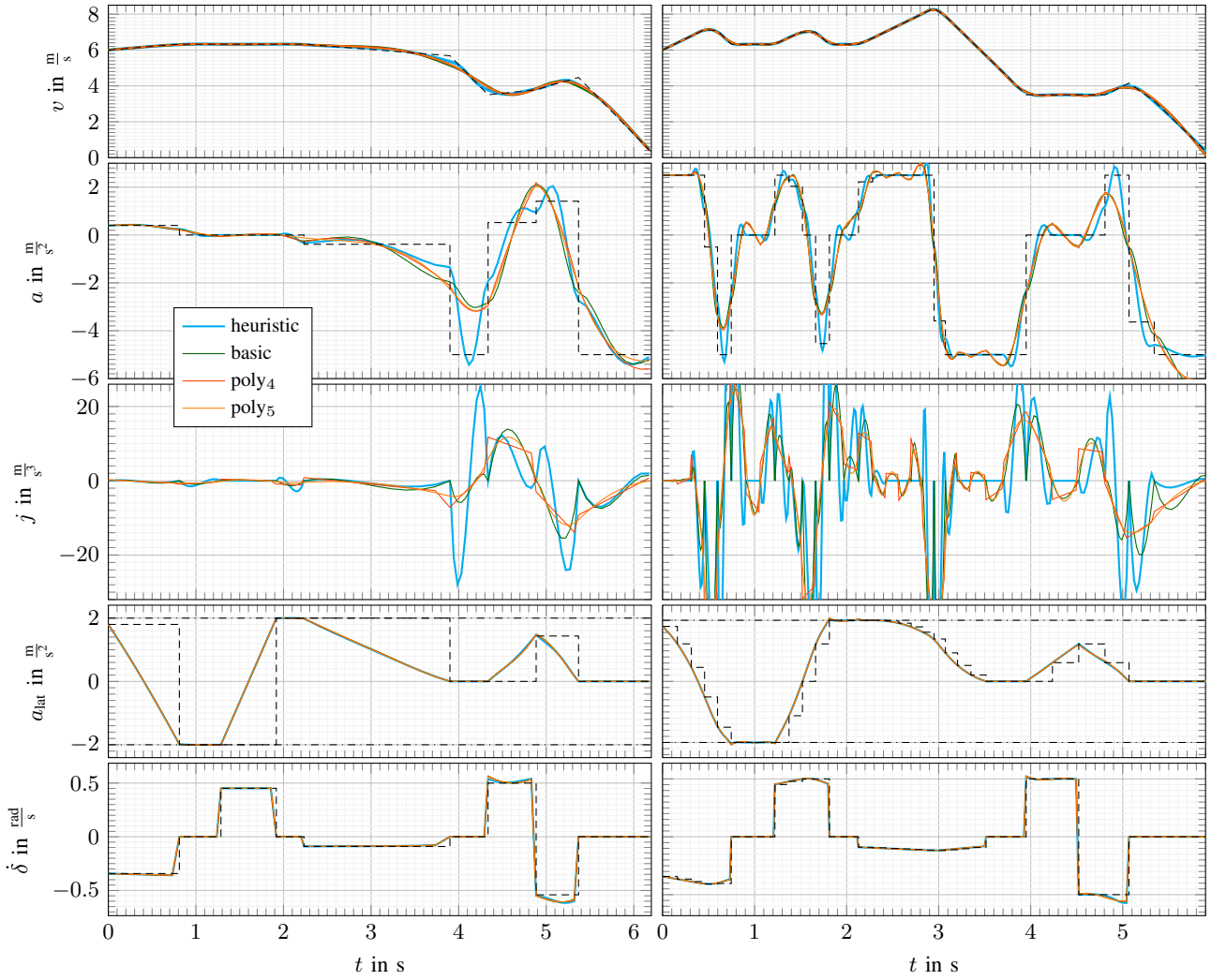

Fig. 5: Results for test case 2, subcase a on the left, b on the right.

To give some intuition, we evaluated two scenarios, a handmade left turn maneuver and a randomly generated sequence. Which method to use depends on the users requirement on the computation time. It also depends on whether the first planning algorithm considers bounds on the steering rate/lateral acceleration.

In the near future, we will extend this approach to work not only with exact times per path point but using a lower and upper bound in time for each sample. This way, higher driving comfort is achievable at the cost of computational complexity. Our work will focus on achieving the best trade-off between comfort and computation time, where the latter ultimately is a measure for safety.

References

[1] K. Kant and S. W. Zucker, "Toward Efficient Trajectory Planning: The Path-Velocity Decomposition," *Int. J. Robotics Research*, vol. 5, no. 3, pp. 72–89, 1986.

[2] S. Perri, C. G. L. Bianco, and M. Locatelli, "Jerk Bounded Velocity Planner for the Online Management of Autonomous Vehicles," in *Proc. IEEE Int. Conf. Automat. Science and Eng. (CASE)*, 2015, pp. 618–625.

[3] F. Ding and H. Jin, "On the Optimal Speed Profile for Eco-Driving on Curved Roads," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2018.

[4] Z. Yi and P. H. Bauer, "Optimal Speed Profiles for Sustainable Driving of Electric Vehicles," in *Proc. IEEE Conf. Vehicle Power and Propulsion (VPPC)*, Oct 2015, pp. 1–6.

[5] A. C. Charalampidis and D. Gillet, "Speed Profile Optimization for Vehicles Crossing an Intersection Under a Safety Constraint," in *Proc. IEEE European Control Conf. (ECC)*, June 2014, pp. 2894–2901.

[6] D. González, V. Milanés, J. Pérez, and F. Nashashibi, "Speed Profile Generation Based on Quintic Bézier Curves for Enhanced Passenger Comfort," in *Proc. IEEE Intelligent Transportation Syst. Conf. (ITSC)*, 2016, pp. 814–819.

[7] C. Liu, W. Zhan, and M. Tomizuka, "Speed Profile Planning in Dynamic Environments via Temporal Optimization," in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, 2017, pp. 154–159.

[8] C. Hubmann, M. Aeberhard, and C. Stiller, " A Generic Driving Strategy for Urban Environments ," in *Proc. IEEE Intelligent Transportation Syst. Conf. (ITSC)*, Rio de Janeiro, Brazil, 2016, pp. 1010–1016.

[9] J. Johnson and K. Hauser, "Optimal Acceleration-bounded Trajectory Planning in Dynamic Environments Along a Specified Path," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2012, pp. 2035–2041.

[10] D. Fassbender, A. Mueller, and H.-J. Wuensche, "Trajectory Planning for Car-Like Robots in Unknown, Unstructured Environments," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, Chicago, IL, USA, Sep. 2014.

[11] D. Fassbender, B. C. Heinrich, and H.-J. Wuensche, "Motion Planning for Autonomous Vehicles in Highly Constrained Urban Environments," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, Daejeon, South Korea, Oct. 2016.