

Optimal Velocity Profile Generation for Semi-Autonomous Vehicles

Christoph Winter; Dr. Ricardo de Castro, Deutsches Zentrum für Luft- und Raumfahrt (DLR), Oberpfaffenhofen

Kurzfassung

Diese Arbeit beschreibt die Optimierung von Geschwindigkeitsprofilen für (semi-) autonome Fahrzeuge. Der Fokus der Optimierung liegt auf dem Kompromiss zwischen zeit- und energieoptimalen Geschwindigkeitsprofilen. Dieser Kompromiss kann direkt durch den Fahrer gewählt werden. Mit Hilfe von dynamischer Programmierung wird das globale Optimum der komplexen Optimierung mit nichtlinearen Nebenbedingungen, die auf den physikalischen Beschränkungen des Fahrzeugs basieren, gefunden. Die Planung der Geschwindigkeitsprofile zusammen mit einem online Bahnplaner dient als Vorgabe für die Pfadfolgeregelung des Fahrzeugs, welche automatisiertes Fahren ermöglicht. Die Effektivität des vorgestellten Ansatzes wird durch numerische Simulationen gezeigt.

Abstract

This work describes the optimization of velocity profiles for the usage in (semi-) autonomous driving. The focus of the optimization lies on the trade-off between time and energy optimal velocity profiles. This trade-off can be directly chosen by the driver. With help of a dynamic programming framework the global optimum of the complex optimization with nonlinear constraints based on the vehicles physical limits is found. The velocity profile generation together with an online path planner serves as an input to the vehicle's path following control, allowing for automated driving. The effectiveness of the proposed approach is shown via numerical simulations.

1. Introduction

The main focus of this work consists in the optimal generation of velocity profiles for semi-autonomous vehicles, e.g. the DLR's robotic electric vehicle research platform ROboMObil (ROMO) [1]. We are particularly interested in investigating how the driver's longitudinal demands, i.e. his preference on how fast the car should be driven along the road, can be integrated into the optimal generation of the velocity profile. In manually operated vehicles, the throttle and braking pedals have been the dominant mechanism to interpret the driver's

longitudinal demands. For example, the displacement of the throttle or braking pedal produced by the driver is normally mapped into an engine torque or braking pressure setpoint. More recently, with the advent of (Adaptive) Cruise Control technologies, the driver started to have the possibility to digitally pre-define velocity setpoints for the vehicle [2], which then can be automatically modified accordingly to the velocity of preceding vehicles. With the increased automation of the vehicle, new ways to interpret the longitudinal driving demands are emerging. Mostly visual, acoustic or haptic feedback for the driver is used to influence the driving behaviour towards a calculated optimal velocity setpoint [3].

In this work, we tackle this problem from a different perspective. We assume that the speed profile for the vehicle is generated by an optimization framework. This means that an optimization problem is formulated, containing two main performance metrics in the cost function: on the one hand the journey time J_T and on the other hand the energy consumption J_E . These two terms are then combined into an overall cost function J using the trade-off factor $\epsilon \in [0,1]$:

$$J = \epsilon J_T + (1 - \epsilon) J_E . \quad (1)$$

Our main proposal consists in asking the driver to directly define the trade-off factor ϵ . For example, if the driver selects $\epsilon = 1$, the vehicle should travel as fast as possible throughout the journey. In contrary, by selecting $\epsilon = 0$, minimization of energy consumption is requested, which would make the vehicle eventually come to a standstill without further constraints. A value in the range $(0,1)$ provides a trade-off between the two performance goals, which the driver can directly explore to adapt the vehicle's speed profile to his preferences.

A similar approach to let the driver choose between fast and slow trajectories was presented in [4]. There the driver manipulates the overall travel time which is incorporated as an additional constraint in the optimization problem. The travel time chosen by the driver then indirectly influences the energy consumption. The resulting optimization problem from [4] is

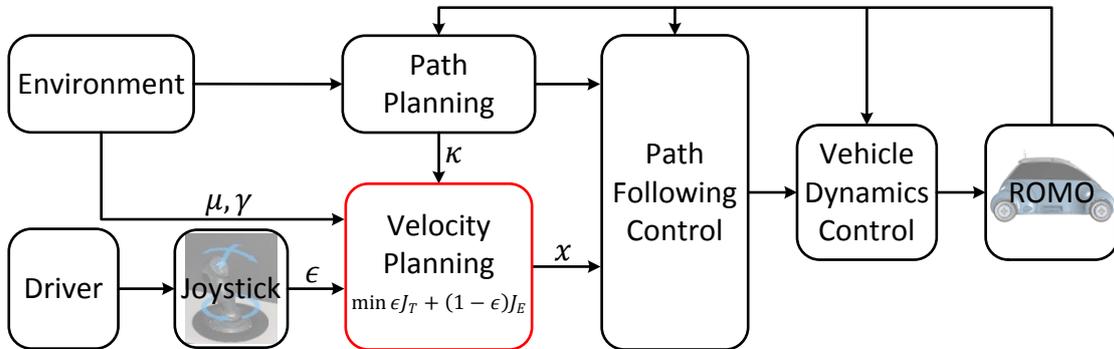


Fig. 1: Schematic showing the structure of the entire motion planning framework of the ROboMObil, with tyre-road friction coefficient μ , road slope γ , curvature κ and velocity x .

nonlinear and the numerical solver is not able to ensure global optimal solutions. This contrasts with the approach investigated in this work, based on dynamic programming, which provides global optimal results for this nonlinear problem.

Fig. 1 shows a schematic of the motion planning framework of the ROboMObil, where the velocity profile generation (marked in red) together with the online path planner [5] serves as an input to the vehicle's path following control [6]. Thereby, the velocity planning module receives the driver's preference ϵ via joystick input, the path curvature κ from the path planner, and environmental information, such as the tyre-road friction coefficient μ and the slope angle of the path γ . Next, the optimized velocity profile x , as well as the planned path, serves as a demand for the path following control, which determines the motion demands of the vehicle dynamics control [7]. The proposed speed profile generation mechanism can also be seen as an advanced driver assistance system (ADAS), which would forbid the driver from requesting unfeasible or potentially dangerous velocity demands. It can also be used in fully autonomous driving as an interface to acquire the driver's velocity preferences which might vary over the journey.

2. Method Description

2.1. Mathematical Problem Formulation

In this section we discuss the formulation of the optimization problem for the velocity profile generation. It includes the cost function and constraints resulting from modelling of the physical interaction between the vehicle and its environment. Let us consider the vehicle's longitudinal dynamics in the time-domain as

$$m\dot{x} = mu - F_{\text{roll}} - F_{\text{down}} - F_{\text{air}}, \quad (2)$$

where x is the vehicle longitudinal velocity, u the acceleration provided by the powertrain (i.e. electric motors and braking system), m the vehicle mass and the driving resistance containing: F_{roll} the rolling resistance, F_{down} the downhill force and F_{air} the air drag.

For the velocity profile generation we consider the arc length s of the path as the independent variable instead of time t . Since environmental information as well as the path description is described with respect to the arc length this simplifies the path and velocity planning. The generation of the desired vehicle's path is described in [5] and is not addressed in this work. With help of the following change of the independent variable

$$\dot{x} = \frac{dx}{dt} = \frac{dx}{ds} \cdot \frac{ds}{dt} = \frac{dx}{ds} x \quad (3)$$

the longitudinal dynamics from (2) can be expressed in space-domain as

$$mx \frac{dx}{ds} = mu - F_{\text{roll}} - F_{\text{down}} - F_{\text{air}}. \quad (4)$$

The motion of the vehicle is constrained by arc-length-dependent upper and lower limits. Actuation constraints are expressed as $u_{\min}(s) \leq u(s) \leq u_{\max}(s)$, where u_{\min} and u_{\max} are the minimum and maximum longitudinal acceleration achievable by the powertrain. Those limits also can be set lower than the physical limitation to satisfy comfort requirements. Additional information about the upcoming path is important to determine suitable velocity limits along the path. Therefore, the slope angle of the path $\gamma(s)$, speed limits along the road, and the path curvature $\kappa(s)$ in dependency of the arc length s are assumed to be known in advanced from the path planning module. The velocity constraints due to physical limits of the vehicle origin from two relations: On the one hand, the maximum lateral and longitudinal acceleration is constrained by the available tire-road friction according to the well-known friction circle [8]

$$a_{\text{lat}}(s)^2 + u(s)^2 \leq \left(\mu(s)g\cos(\gamma(s))\right)^2, \quad (5)$$

where the combination of lateral acceleration a_{lat} and longitudinal acceleration u has to be less or equal to the physical maximum acceleration a_{max} with a given friction coefficient μ , slope angle γ and the gravitational force g . On the other hand, the lateral acceleration is determined by the velocity x and the curvature κ along the path as in $a_{\text{lat}}(s) = \kappa(s) \cdot x(s)^2$. Combining those two relations leads to the following velocity-dependent constraint:

$$\left(\kappa(s) \cdot x(s)^2\right)^2 \leq \left(\mu(s)g\cos(\gamma(s))\right)^2 - u(s)^2. \quad (6)$$

Furthermore, there are velocity limits based on traffic rules and comfort criterions, which are given by $x_{\min}(s) \leq x(s) \leq x_{\max}(s)$, with the minimum velocity x_{\min} and the maximum velocity x_{\max} , both with respect to legal limits or the drivers preference. The optimization problem has additional boundary conditions for the start and end values of the velocity. The initial speed is set to $x(0) = x_{\text{start}}$ and the desired final speed $x(s_f)$ at arc length s_f needs to stay in the interval $x(s_f) \in [x_{f,\min}, x_{f,\max}]$.

As explained in Section 1, we are interested in investigating trade-offs between travel time and energy consumption. Therefore, the overall travel time J_T as cost function is obtained as

$$J_T = \int_0^{t_f} dt = \int_0^{s_f} \frac{1}{x} ds, \quad (7)$$

with the final time of the velocity profile t_f and the overall arc length $s(t_f) = s_f$. The energy consumption as the second part of the cost function is determined as follows:

$$J_E = \int_0^{s_f} \frac{mu^2}{\eta(u, x)} ds, \quad (8)$$

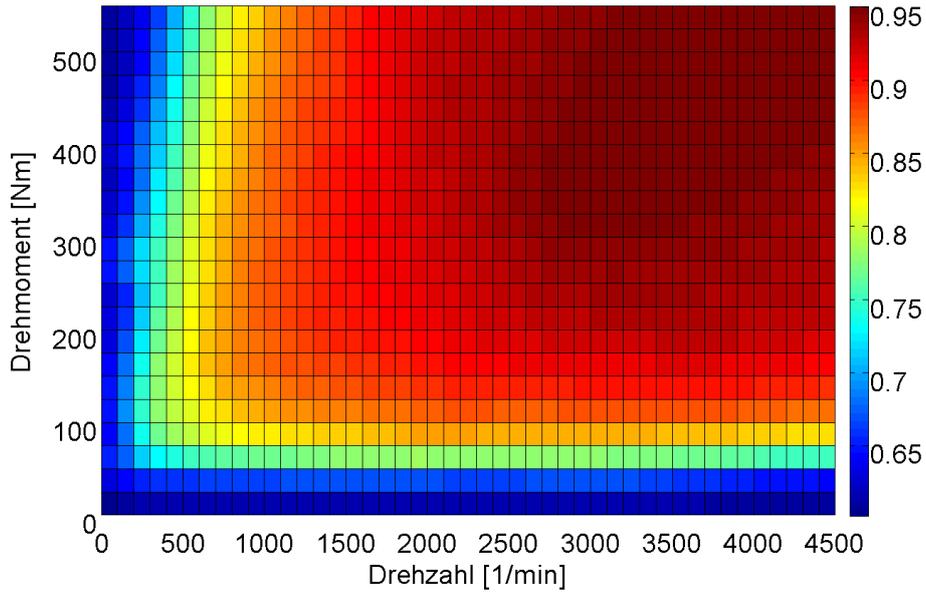


Fig. 2: Efficiency map of a permanent magnet synchronous motor (PowerPhase 100 Traction System, UQM).

where η is the motor efficiency in dependency of the desired acceleration u and vehicle speed x . Note that to (indirectly) penalize the energy, a quadratic acceleration term was used in the cost function as applied in the field of robotics [9] as well as in autonomous driving [10]. This quadratic term prevents chattering problems of the DP solver. The cost function is weighted with the efficiency of the traction motor to penalize inefficient operating points. The efficiency of the motor is represented by an efficiency map of a permanent magnet synchronous motor (see Fig. 2). The motor torque τ and the rotational speed ω required to calculate the efficiency from the map are determined from x and u . Here it is assumed that only the electric motor and no friction brake is utilized for braking. In addition, a single electric motor is considered, otherwise a control allocation describing the distribution of the desired acceleration to individual motor torques has to be taken into account for the cost function calculation.

The above relations were discretized using a forward Euler approximation and a constant acceleration between arc length nodes is assumed. The resulting discrete optimization problem of finding an optimal velocity profile for the vehicle can be formulated as follows:

$$\begin{aligned}
 & \min_u \epsilon J_T + (1 - \epsilon) J_E \\
 \approx & \min_{u_k} \sum_{k=0}^N \underbrace{\epsilon \frac{1}{x_k} \Delta s_k + (1 - \epsilon) \frac{m u_k^2}{\eta_k(u, x_k)} \Delta s_k}_{g_k(u_k, x_k)} \quad (9a)
 \end{aligned}$$

$$\text{s.t. } x_{k+1} = x_k + \frac{\Delta s_k}{m x_k} (m u_k - F_{\text{roll},k} - F_{\text{down},k} - F_{\text{air},k}) = f(u_k, x_k) \quad (9b)$$

$$u_{\min,k} \leq u_k \leq u_{\max,k}, \quad x_{\min,k} \leq x_k \leq x_{\max,k}$$

$$(|\kappa_k| x_k^2)^2 \leq (\mu_k g \cos(\gamma_k))^2 - u_k^2$$

$$x_0 = x_{\text{start}}, \quad x_N \in [x_{f,\min}, x_{f,\max}]$$

where $x_k = x(s_k)$, $u_k = u(s_k)$ etc. state the discretised variable at node k of the overall $N + 1$ discrete arc length nodes with a spacing of Δs_k . Thereby the cost function $g_k(u_k, x_k)$ represents the driver defined trade-off (1) and $f(u_k, x_k)$ describes the state space difference equation.

As mentioned before, the simultaneous reduction of both cost terms J_T and J_E is contradictory to each other and therefore the optimization (9) yields a point on a Pareto-front, which is defined by a varying ϵ from 0 to 1. In our case the optimization problem requires only one state variable x since for example no gear shifting has to be taken into account for an electric drivetrain and only one input variable u . This reduced number of states and inputs is important to ensure that the problem can be solved efficiently in real-time.

2.2. Dynamic Programming

The optimization problem of the previous section is already formulated in a discrete way, which makes it well suitable for the use of dynamic programming (DP) [11] as a numerical solver. An important advantage of DP lies in its capability to handle nonlinear cost-functions and constraints, while guaranteeing global optimal solutions. The dynamic optimization problem is solved with the help of the following recursive equation starting from the end state x_N and $k = N - 1, \dots, 0$

$$J_N(x_N) = g_N(u_N, x_N)$$

$$J_k(x_k) = \min_{\text{s.t. (9b)}} \{g_k(u_k, x_k) + J_{k+1}(f(u_k, x_k))\}. \quad (10)$$

The computational effort for solving the problem with dynamic programming strongly depends on the number of states and the number of nodes along the considered path length that is optimized. To enable the driver to command the trade-off factor ϵ it is important to achieve a real-time capable implementation of the DP solver.

For an efficient solving of the optimization problem a size reduction is important. The state space as well as the input space can be reduced by constraining it to only feasible values regarding the constraints in (9b). An example of the resulting state space restricted by various constraints is depicted in Fig. 3. These constraints include: 1. maximum longitudinal acceleration u_{\max} , 2. maximum lateral acceleration $a_{\text{lat,max}}$, 3. maximum velocity x_{\max} , 4. minimum longitudinal acceleration u_{\min} and 5. minimum velocity x_{\min} (e.g. 40 % of x_{\max}).

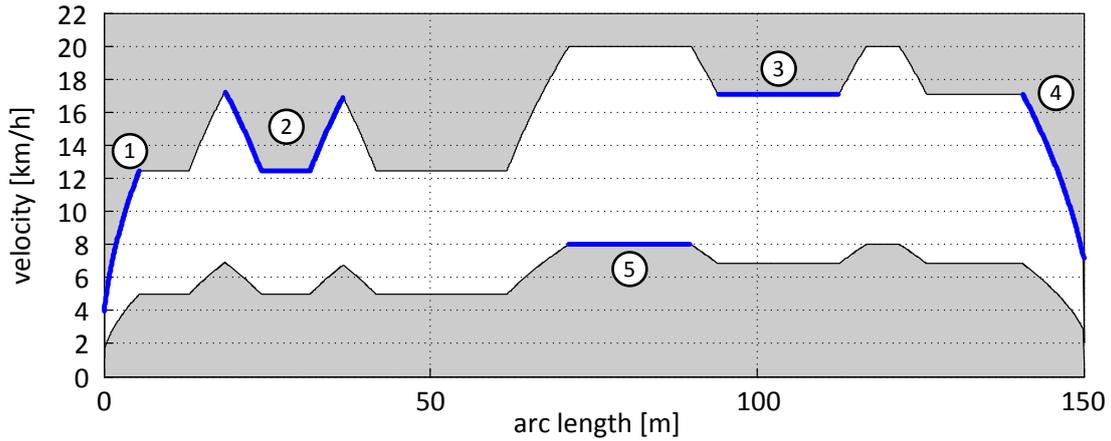


Fig. 3: Reduced state space with consideration of velocity and longitudinal and lateral acceleration constraints.

Furthermore, the presented velocity profile generation is integrated with an online path planning framework [5], [12] which is able to react to environmental changes. The possibility to easily apply nonlinear constraints to the optimization problem can be used to include those changes into the velocity optimization problem. One possible environmental condition that directly affects the velocity generation is the tyre-road friction coefficient μ . The change in the tyre-road friction coefficient $\mu(s)$ in dependency of arc length s directly influences the maximum feasible velocity $x_{\max}(s)$ in the DP algorithm. With the help of (6) the upper bound of the state space $x_{\max}(s)$ can be lowered if an area with a low friction coefficient μ is detected. In the results in Section 3.3 an example of this adaption of the velocity profile to environmental changes is shown.

3. Results

3.1. Real-Time Capability

In a (semi-) autonomous vehicle the motion demand has to be determined while the vehicle is moving to be able to react to environmental changes, in contrast to a pre-planned motion. Consequently, a real-time implementation of the velocity planning algorithm is necessary. Similar to [13], we consider a computation time of 0.1 s as a threshold for real-time capability of the algorithm. This threshold is also constrained by the vision based environment perception of the ROboMObil [14] and is marked with a red dashed line in Fig. 4. The DP problem as described in Section 2 is implemented in Matlab and solved using the DP Toolbox [15], which we modified to handle constraints in form of $x \leq x_{\max}(s, u)$ that are not only dependent on the arc length but also on the input u . Fig. 4 shows the computation time needed to solve the DP problem depending on individual problem parameters (step size along the arc length Δs , number of arc length steps N , number of steps in the states grid N_x

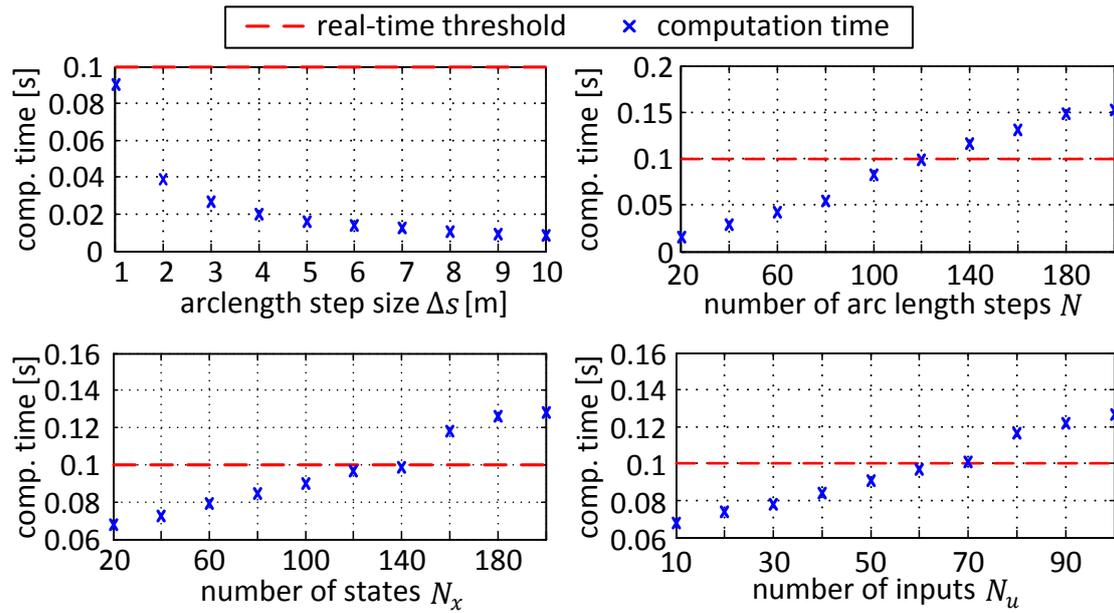


Fig. 4: Computation time of the DP algorithm depending on different parameters as the step size along the arc length Δs , the number of arc length steps N , the number of steps in the states grid N_x and the number of steps in the input grid N_u (default values: $\Delta s = 1$ m, $N = 110$, $N_x = 100$ and $N_u = 50$).

and number of steps in the input grid N_u) while the other quantities are held constant. The default setup for measuring the computation time is $\Delta s = 1$ m, $N = 110$ (regarding a 110 m long path), $N_x = 100$ and $N_u = 50$ on an Intel i7 quad-core processor with 2.8 GHz.

As can be seen in Fig. 4, the step size Δs along the arc length of the given path has the highest impact on the computation time with an exponential dependency. All other parameters have approximately a linear dependency in the computation time. For a real-time implementation it is critical to choose an appropriate step size Δs to get a high accuracy of the computed velocity profile along the arc length and a sufficiently low computation time. The DP problem remains feasible within our computation time threshold for $\Delta s \geq 1$ m. This is valid even with the high density of the state-space discretisation.

3.2. Trade-Off Velocity Profiles

After discussing the computational effort, in this section we show the results of the velocity profile generation using the DP method. To show the impact of the driver's choice on the trade-off parameter, various velocity profiles for the same test track (see Fig. 6a) are determined. Thereby, the trade-off parameter is varied from $\epsilon = 0$ to $\epsilon = 1$ in steps of 0.1. The upper plot of Fig. 5 shows the resulting family of curves of the velocity for the test track. The lower and upper bounds of the velocity are marked with black dashed lines, as well as the start and end values of the velocity profile are marked in red in Fig. 5. The corresponding

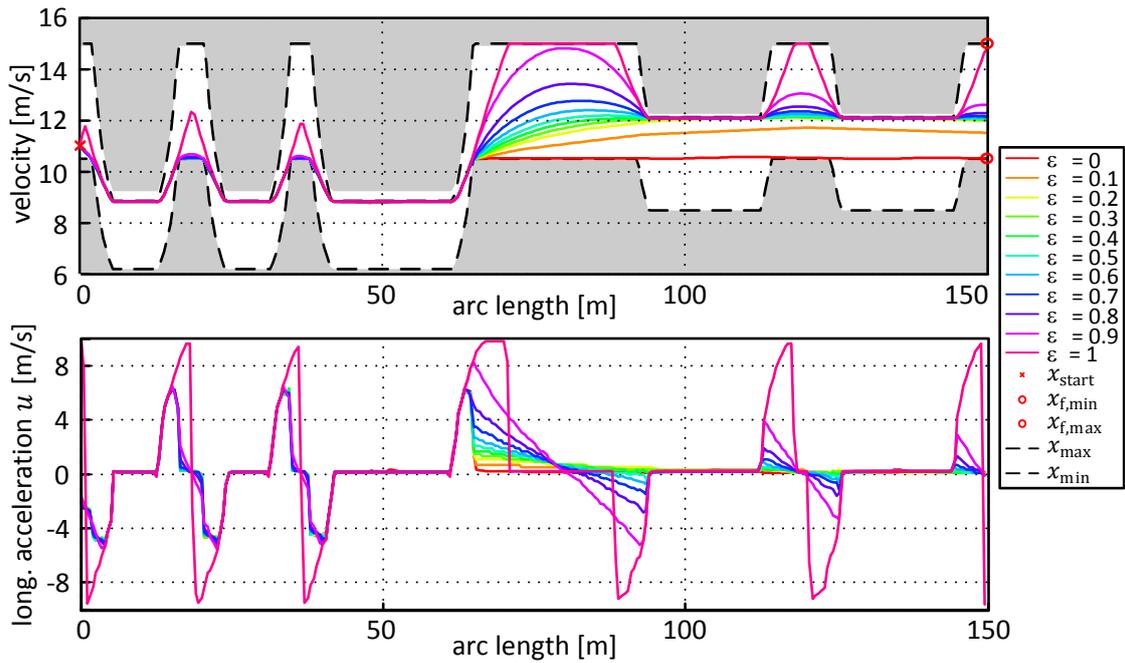


Fig. 5 Velocity profiles for a varying trade-off parameter ϵ within the constrained state-space and the corresponding longitudinal accelerations.

longitudinal acceleration u is shown in the lower plot of Fig. 5 for the varying trade-off parameter. During the optimization all constraints of the problem are satisfied.

Fig. 6b depicts the resulting Pareto-front of the two cost criteria, the normalized journey time $J_T/J_T(\epsilon = 0)$ and the normalized energy consumption $J_E/J_E(\epsilon = 1)$, for the different trade-off parameters. Our proposal is that the driver can freely choose the trade-off parameter along this Pareto-front while driving. The extreme cases of a time optimal velocity profile ($\epsilon = 1$) and an energy optimal profile ($\epsilon = 0$) constrain the possible trade-off solutions.

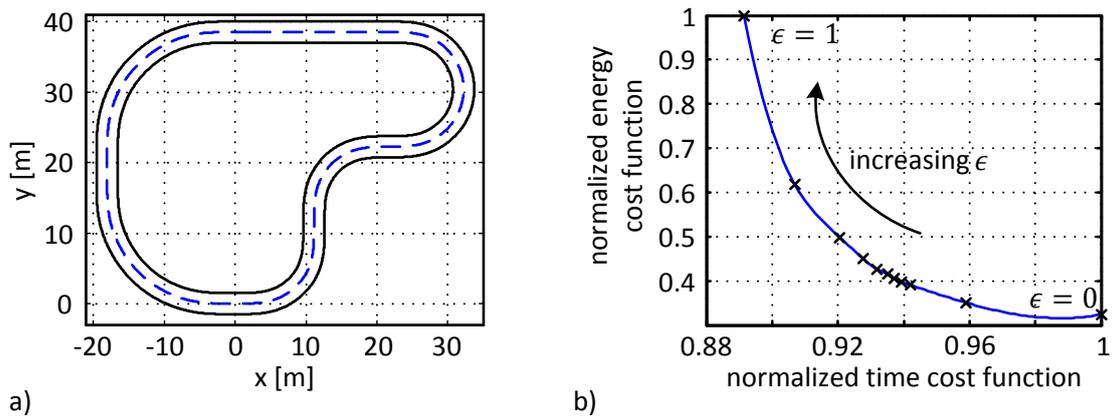


Fig. 6: a) Test track for the velocity profile generation. b) Pareto-front of the two normalized energy and time cost criterions.

3.3. Varying tyre-road friction coefficient

As introduced in Section 2.2, the presented framework is able to adjust the constraints of the velocity profile to environmental changes. This is helpful to always maintain the computed velocity profile within the physical limits of the vehicle, also under varying road conditions. In the following investigation a road turn with variable friction coefficient μ , e.g. resulting from wet or icy road conditions, is used (see Fig. 7a). The turn consists of straight segments at the beginning and the end, a circular arc with a radius of 37 m in the middle and clothoids joining the straight segments and the arc segment.

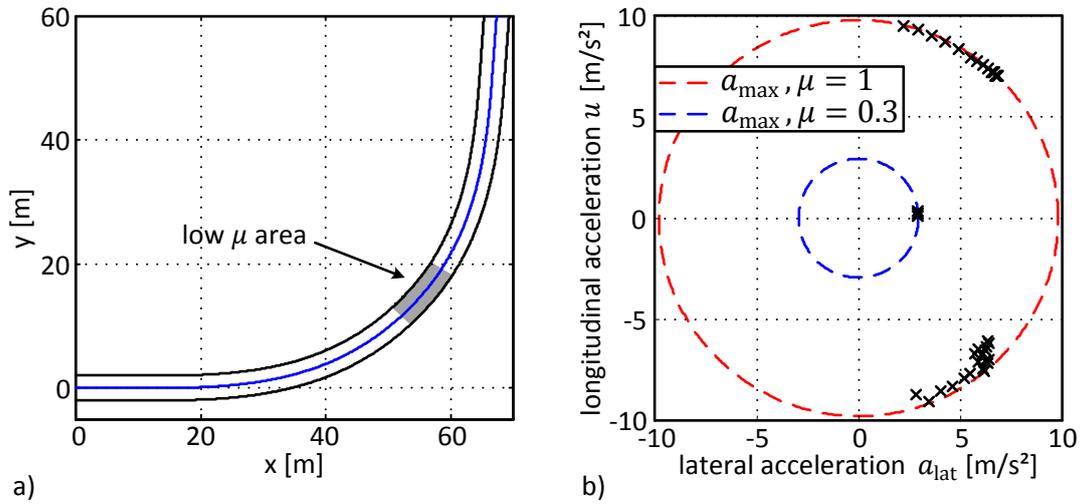


Fig. 7: a) Visualisation of the path of a single road turn (minimum radius of 37 m) with the low μ area marked grey. b) Friction circle of the acceleration while driving through the turn (for $s \in [26 \text{ m}, 82 \text{ m}]$) with the physical acceleration limits a_{\max} as dashed lines.

The upper plot in Fig. 8 shows the optimal velocity profile (marked with a solid black line) corresponding to the path depicted in Fig. 7a. It can be seen clearly that the lower friction coefficient in the turn lowers the maximum lateral acceleration and thereby the maximum stable velocity x_{\max} within the turn (marked with a red dashed line in the upper plot of Fig. 8). The DP algorithm shows an anticipatory behaviour by braking approximately 20 m before the low μ area to avoid excessive lateral acceleration during this slippery section. The step in the upper bound of the velocity profile is handled well by the DP algorithm and a smooth velocity profile for $\epsilon = 1$ within its acceleration bound is computed (see the lower plot of Fig. 8). The DP algorithm fully exploits the physical limits (marked as dashed lines for different μ values in Fig. 7b) to achieve a time optimal solution, as indicated by the friction circle in Fig. 7b. The points corresponding to the part of the turn with constant velocity (before $s = 26 \text{ m}$ and after $s = 82 \text{ m}$) are left out to focus on the non-zero acceleration phases. In summary, a safe velocity profile is always generated even under changing road conditions. Although not addressed in this work, the required data of the road surface friction coefficient could be

estimated using on-board sensors, e.g. with the help of vision sensors, or acquired via external sources such as car-to-infrastructure communications.

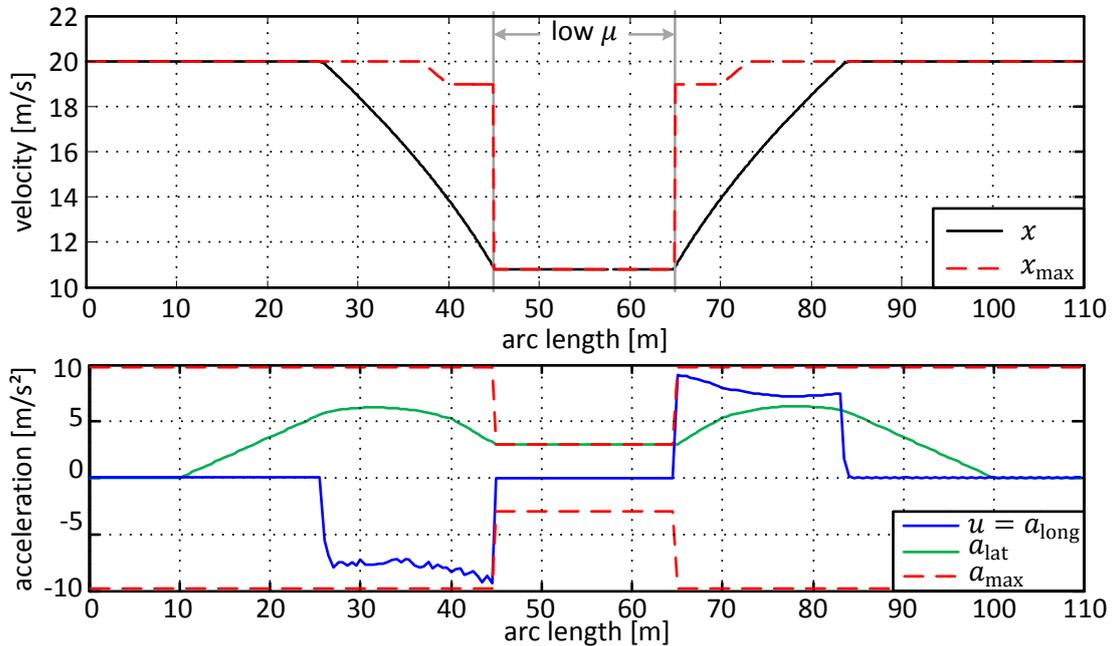


Fig. 8: Velocity profile and upper velocity bounds of a road turn with a low μ area and a friction coefficient of $\mu = 0.3$ in the upper plot with the corresponding longitudinal and lateral acceleration u and a_{lat} and its combined limit a_{max} in the lower plot.

4. Conclusion and Future Work

The presented velocity profile generation algorithm is able to determine the global optimum of a complex optimization problem with nonlinear constraints with the help of dynamic programming. Thereby, the optimization considers a trade-off between time and energy optimal velocity profiles. This trade-off is directly chosen by the driver. Furthermore, the velocity planning is able to take environmental changes, e.g. a variation of the tyre-road friction coefficient, into account by incorporating an adapted maximum velocity into the optimization constraints. As a result, a safe velocity profile is always determined regarding the physical limits of the vehicle.

Future work will include the implementation of the velocity profile generation on the real-time system of the ROboMObil and its experimental validation. In addition, the combination of the velocity profile generation and the online path planning into a single optimization problem will be investigated.

Bibliography

- [1] Jonathan Brembeck et al., "ROMO – The Robotic Electric Vehicle," in *International Symposium on Dynamics of Vehicle on Roads and Tracks*, 2011.
- [2] Bernhard Knauder, Michael Karner, and Markus Schratter, "Predictive longitudinal vehicle control based on vehicle-to-infrastructure communication," in *International Conference on Connected Vehicles and Expo*, Vienna, 2014, pp. 258-263.
- [3] Carina Fors, Katja Kircher, and Christer Ahlström, "Interface design of eco-driving support systems – Truck drivers' preferences and behavioural compliance," *Transportation Research Part C: Emerging Technologies*, vol. 58, pp. 706 - 720, 2015.
- [4] Thijs van Keulen, Bram de Jager, Darren Foster, and Maarten Steinbuch, "Velocity trajectory optimization in hybrid electric trucks," in *American Control Conference (ACC), 2010*, 2010, pp. 5074-5079.
- [5] Christoph Winter, Peter Ritzer, and Jonathan Brembeck, "Experimental Investigation of Online Path Planning for Electric Vehicles," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Rio de Janeiro, Brazil, 2016, pp. 1403-1409.
- [6] Peter Ritzer, Christoph Winter, and Jonathan Brembeck, "Experimental validation of geometric path following control with demand supervision on an over-actuated robotic vehicle," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, 2016, pp. 539-545.
- [7] Ricardo de Castro, Tilman Bünthe, and Jonathan Brembeck, "Design and Validation of the Second Generation of the Robomobil's Vehicle Dynamics Controller," in *Proceedings of the 24th Symposium of the International Association for Vehicle System Dynamics (IAVSD 2015)*, 2016.
- [8] Hans Pacejka, "Tyre and Axle Characteristics," in *Tire and Vehicle Dynamics*.: Elsevier, 2006, p. 5.
- [9] Diederik Verscheure, Bram Demeulenaere, Jan Swevers, Joris De Schutter, and Moritz Diehl, "Time-Optimal Path Tracking for Robots: A Convex Optimization Approach," *IEEE Transactions on Automatic Control*, vol. 54, no. 10, pp. 2318-2327, 2009.
- [10] Ricardo de Castro, Mara Tanelli, Rui Esteves Ara, and Sergio Matteo Savaresi, "Minimum-Time Path-Following for Highly Redundant Electric Vehicles," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 2, pp. 487-501, March 2016.
- [11] Richard Bellman and Stuart Dreyfus, *Applied Dynamic Programming*. Princeton: Princeton Univ. Press, 1962.
- [12] Jonathan Brembeck and Christoph Winter, "Real-time capable path planning for energy management systems in future vehicle architectures," in *IEEE Intelligent Vehicles Symposium*, 2014, pp. 599-604.
- [13] Erik Hellström, Maria Ivarsson, Jan Aslund, and Lars Nielsen, "Look-ahead control for heavy trucks to minimize trip time and fuel consumption," *Control Engineering Practice*, vol. 17, no. 2, pp. 245 - 254, 2009.
- [14] Alexander Schaub, Juan Carlos Ramirez de la Cruz, and Darius Burschka, "Autonomous Parking using a Highly Maneuverable Robotic Vehicle," in *IFAC World Congress*, vol. 19, 2014, pp. 2640-2645.
- [15] Olle Sundstrom and Lino Guzzella, "A generic dynamic programming Matlab function," in *2009 IEEE Control Applications, (CCA) Intelligent Control, (ISIC)*, 2009, pp. 1625-1630.