

Optimal Trajectories

Moritz Werling

Zusammenfassung New active driver assistance systems that work at the road- and navigation level as well as automated driving face a challenging task. They must permanently calculate the vehicle input commands (such as those for the steering, brakes, and the engine/powertrain) to realize a desired future vehicle movement, a driving trajectory. This trajectory has to be optimal in terms of some optimization criterion (in general a trade-off between comfort, safety, energy effort, and travelling time), needs to take the vehicular dynamics into account, and must incorporate lane boundaries or the predicted free space amidst (possibly moving) obstacles. This kind of optimization can be mathematically formulated as a so-called optimal control problem. To limit the calculation effort, the optimal control problem is usually solved only on a limited prediction interval (starting with the current time) leading to a receding horizon optimization. The chapter illustrates this practically proven approach in detail. Furthermore, the three general principles of dynamic optimization known from control theory and robotics are presented, namely calculus of variations, direct optimization, and dynamic programming. Furthermore, their application to driver assistance systems and automated driving is exemplified and the high practical relevance is supported by the given literature. Finally, the respective advantages and limitations of the optimization principles are discussed in detail proposing their combination for more involved system designs.

1 Introduction

Advanced collision avoidance systems, lane keeping support, traffic jam assistance, and remote valet parking; they all operate on the actuators to relieve the drivers of the lateral and/or longitudinal vehicle control or make it safer for them. Well-

Moritz Werling
BMW Group, München and Karlsruhe Institut of Technology, Karlsruhe, E-mail: moritz.werling@bmw.de

defined tasks, such as staying in the middle of the marked lane while following the vehicle ahead, can be handled by a set-point controller [4]. And yet standard automated parking maneuvers already require a calculated trajectory¹ that must be adapted to the available parking space. As systems need to cover more and more situations, the number of degrees of freedom increase, which makes a trajectory parameterization very complex, especially when the vehicle must take numerous obstacles into account. This calls for a systematic approach based on mathematical optimization (as opposed to heuristic approaches such as potential field and elastic bands methods, see e.g. [6], [1], with their inherent limitations, cf. [5]). In the chapter at hand, we address real-time trajectory optimization², a task that an automated vehicle faces when it travels through its environment, also referred to as motion planning in robotics [7]; [8]. The focus will be on methods that engage with the longitudinal and lateral vehicle movement. However, the results can be transferred to novel warning systems that can also benefit from an optimal trajectory prediction, see e.g. [3]. Speaking most generally, a trajectory optimization method is sought that can handle both structured (e.g. streets) and unstructured environments (parking lots), one that works amongst cluttered static obstacles and in moving traffic as well as exhibits a natural, human-like, anticipatory driving behavior. Using more technical terms, the method should be easy to implement, parameterize, adapt, scale well with the number of vehicle states and the length of the optimization horizon, incorporate nonlinear, high fidelity vehicle models, combine the lateral and longitudinal motion, be complete³, allow for both grid maps and object lists representations (with predicted future poses) of the obstacles, be numerically stable, and transparent in its convergence behavior (if applicable). Also, the calculation effort must be low to allow for short optimization cycles on (low performance) electronic control units so that the vehicle can quickly react to sudden changes in the environment. Unfortunately, there is no such single method that has all these properties. And, most likely, there will never be one. However, different optimization methods can be combined to get as close as possible to the above requirements. The next section therefore gives a closer look into the basic principles of trajectory optimization and their application.

2 Dynamic Optimization

When engineers speak about optimization, they usually refer to static optimization, in which the optimization variables \mathbf{p} are finite, also called parameters (e.g., finding the most efficient operating point of an engine). Then optimal refers to some well-defined

¹ More precisely: a path, which does not have any time dependency

² Notice, that in control theory the term trajectory planning usually implies that there is no feedback of the actual system states on the trajectory. The dynamical system is then only stabilized by a downstream trajectory tracking controller, which is not always advisable. We therefore use the term trajectory optimization instead to be independent of the utilized stabilization concept.

³ A complete algorithm always finds the solution if it exists.

optimization criterion, usually the minimization of a cost function $J(\mathbf{p})$ (e.g., fuel consumption per hour). Trajectory optimization is different in that the optimization variables are functions $\mathbf{x}(t)$ of an independent variable t , usually time. It is also called dynamic or infinite-dimensional optimization. Evaluating $\mathbf{x}(t)$ therefore requires a cost functional (a “function of a function”), which quantifies the “quality” of the trajectory $\mathbf{x}(t)$ by a scalar value. Due to the vehicular focus, a special case will be considered, one that requires the trajectory $\mathbf{x}(t)$ to be consistent with some dynamical system model which has an input \mathbf{u} . Without such model, the optimization cannot incorporate the inherent properties and physical limitations of the vehicle. This special case of dynamic optimization is called an optimal control problem (e.g., [9]).

2.1 Optimal Control Problem

A fairly general formulation of the optimal control problem (OCP) reads:

Minimize the cost functional

$$J(\mathbf{u}(t)) = \int_{t_0}^{t_f} l(\mathbf{x}(t), \mathbf{u}(t), t) dt + V(\mathbf{x}(t_f), t_f) \quad (1a)$$

subject to the system dynamics

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (1b)$$

as well as the equality and inequality constraints

$$\mathbf{g}(\mathbf{x}(t_f), t_f) = \mathbf{0} \quad (1c)$$

$$\mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), t) \leq \mathbf{0}, \quad \forall t \in [t_0, t_f] . \quad (1d)$$

In other words, for our (possibly nonlinear and time variant) system with state $\mathbf{x} \in \mathbb{R}^n$ and input $\mathbf{u} \in \mathbb{R}^m$ we seek on the interval $t \in [t_0, t_f]$ the input trajectory $\mathbf{u}(t)$ that minimizes the cost functional J while steering (in the truest sense of the word) the system from its initial state \mathbf{x}_0 to an end state $\mathbf{x}(t_f)$, so that the equality and inequality constraints are fulfilled at all times. The optimal input trajectory is usually denoted by $\mathbf{u}^*(t)$ and the resulting state trajectory by $\mathbf{x}^*(t)$, see Fig. 1. Notice that J comprises integral costs l and end point costs V and is not only a functional of the input $\mathbf{u}(t)$ but also of the states $\mathbf{x}(t)$. Furthermore, if the final time t_f is not given, it then becomes part of the optimization, so that the length of the trajectory will also be optimized.

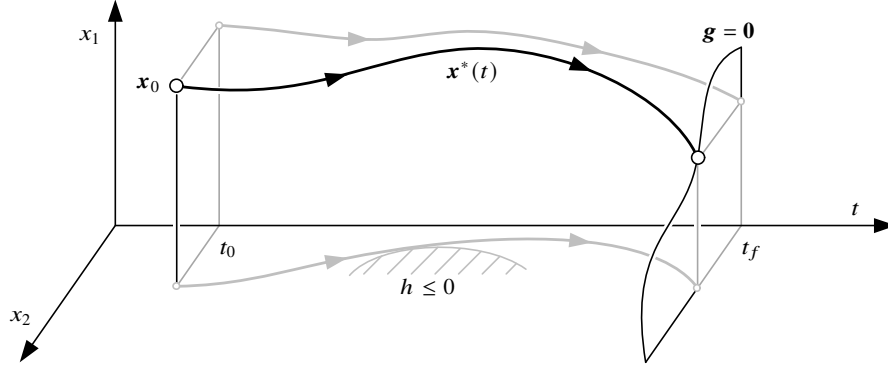


Abb. 1 Example of an optimal trajectory $x^*(t) \in \mathbb{R}^2$ with end constraints $g = 0$ at a fixed final end time t_f and inequality constraint $h \leq 0$ for x_2

2.2 Problem Formulation for DAS and Automated Driving

Coming back to the automotive application, equation (1) of the previous section describes the dynamics of the vehicle. This state space model also includes the planar motion, either relative to some reference such as the lane center, see Sect. 3.1.2, or relative to a stationary origin, see Sect. 3.3.7. Undesired vehicle motion such as deviations from the lane center, detours, dangerous vehicle states (e.g., large slip angles), or uncomfortable jerks, e.g., caused by hectic steering actions will be penalized in the cost functional (1). The free space prediction between the generally moving obstacles can be described by the inequality constraints (4). And the end constraints (3) can be utilized to require that the optimized vehicle trajectory will be aligned to the road at the end of the optimization interval. As will be explained in Sect. 5, this final state plays an important role for the stability of the “replanning” algorithm, therefore special costs $V(x(t_f), t_f)$ can be introduced in (1).

3 Solving the Optimal Control Problem

All known approaches to the OCP can be assigned to one of the following three principles, see e.g. [2].

3.1 Approach I: Calculus of Variations

The classical approach to the OCP is *calculus of variations*, which delivers valuable insight in the solution.

3.1.1 Theoretical Background: Hamilton Equations

Static optimization problems can be tackled by *differential calculus*. It is well known that the first derivative of a function $J(\mathbf{p})$ is equal to zero at a minimum (or any other stationary point). For multivariate problems we can write $\nabla J(\mathbf{x}) = \mathbf{0}$ which leads to a set of (algebraic) equations that the optimum \mathbf{p}^* must satisfy. The extension to problems with equality constraints requires the method of so-called *Lagrange-multipliers*, yielding first order necessary conditions for optimality. Analogously for dynamic optimization, variational calculus requires that the first variation of the functional $J(\mathbf{u}(t))$ vanishes for the optimal control function \mathbf{u}^* which is often written as $\delta J(\mathbf{u}(t)) = 0$.

In order to incorporate the system dynamics in the OCP, which constitute (differential) equality constraints, we can also apply the Lagrange-multiplier method. This yields a set of differential equations, the so-called Hamilton equations:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \quad (2a)$$

$$\dot{\lambda} = -\frac{\partial l}{\partial \mathbf{x}} - \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right]^T \lambda \quad (2b)$$

$$\mathbf{0} = \frac{\partial l}{\partial \mathbf{u}} + \left[\frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right]^T \lambda \quad (2c)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0 \quad (3)$$

They are *first order necessary conditions* for our OCP, when there are no inequality constraints (4) involved. The function $\lambda(t)$ constitutes the Lagrange-multipliers, here called *co-states*. Besides the initial condition

$$\mathbf{x}(t_0) = \mathbf{x}_0 \quad (4)$$

the optimal trajectory must fulfill the (algebraic) *transversality conditions*, depending on whether the end state $\mathbf{x}(t_f)$ is constraint by TODO (3) and/or the final time t_f is given (see e.g. [9]).

The simplest condition requires a fixed end state \mathbf{x}_f at a given end time t_f so that

$$\mathbf{x}(t_f) = \mathbf{x}_f \quad (5)$$

Either way, this results in a boundary value problem, which in general needs to be solved numerically. This so-called indirect approach is very accurate but not as flexible as the direct approach that we will introduce in Sect. 3.2. However, for simple OCPs the resultant boundary value problem can be solved analytically, leading to fast computable optimal trajectory primitives with broad applications (see Sect. 4).

3.1.2 Example Application: Automated Lane Change

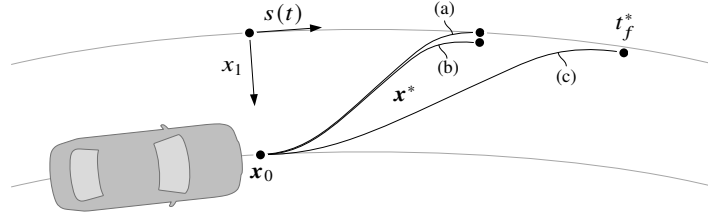


Abb. 2 Optimal lane changes for (a) a given end time and end state; (b) a given end time and a free end state; (c) a free end time and a free end state

3.1.3 Further Readings

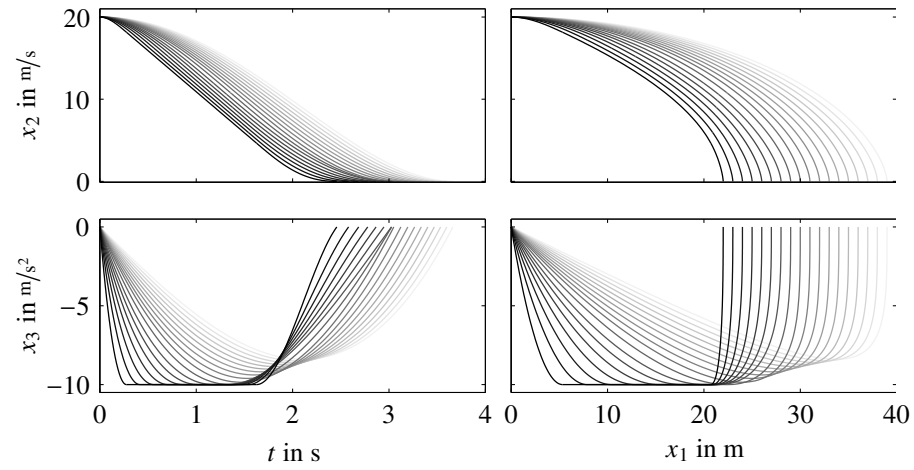


Abb. 3 Minimal jerk-square minimal time stopping trajectories with acceleration constraint with $a \geq -10\text{m/s}^2$

3.2 Approach II: Direct Optimization Techniques

3.3 Theoretical Background: Finite Parameterization Approximation

Finite parameterization of the input and sampling of the constraints

3.3.1 Example Application: Emergency Obstacle Avoidance

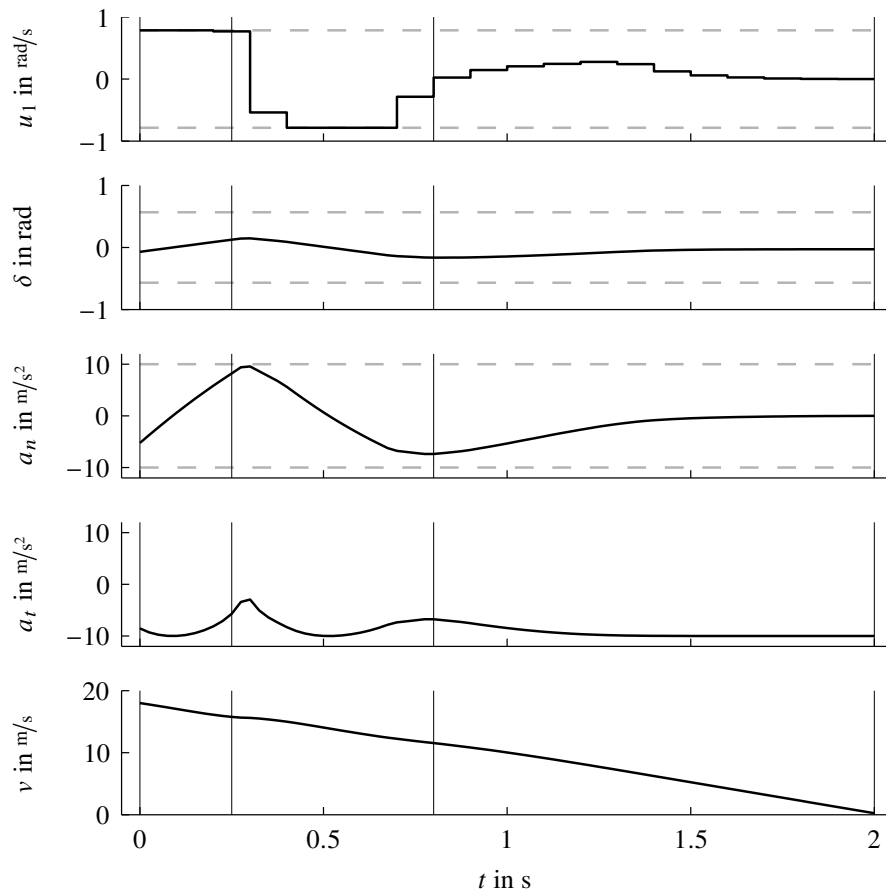


Abb. 4 Signalverläufe zu Abb. 6 mit optimaler Steuerung u_1 , Lenkwinkel δ , Quer- und Längsbeschleunigung a_t , a_n sowie Geschwindigkeit v ; Die vertikalen Linien markieren die Zeitpunkte der Draufsichten, die horizontalen gestrichelten Linien die Nebenbedingungen [10].

Automatic obstacle avoidance with combined braking and steering at the friction limit; the safety margin d_{safety} is shown as a grey circle around the stationary pedestrian depicted as a black dot; the lane boundaries given by d^{max} are drawn as grey dotted lines.

3.3.2 Further Readings

3.3.3 Approach III: Dynamic Programming

3.3.4 Theoretical Background: Bellman's Principle of Optimality

3.3.5 Value Iteration

3.3.6 Example Application: Optimal Overtaking

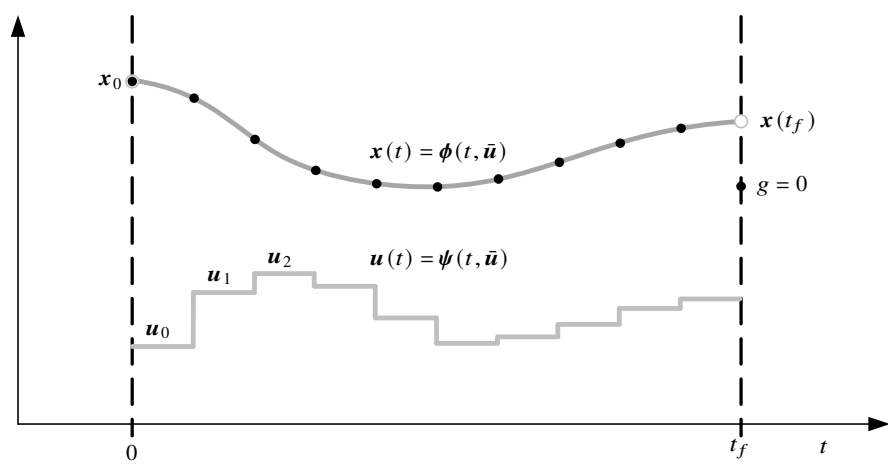
3.3.7 Further Readings

4 Comparison of the Approaches

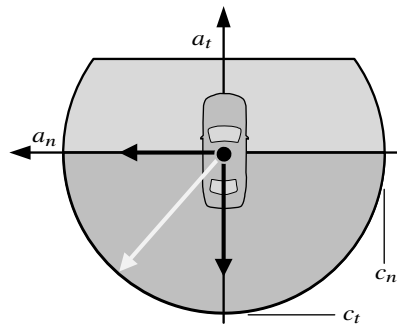
5 Receding Horizon Optimization

6 Conclusion

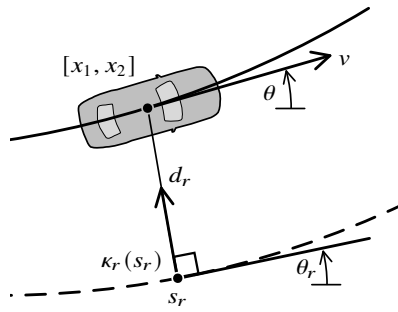
[Finite parameterization of the input]



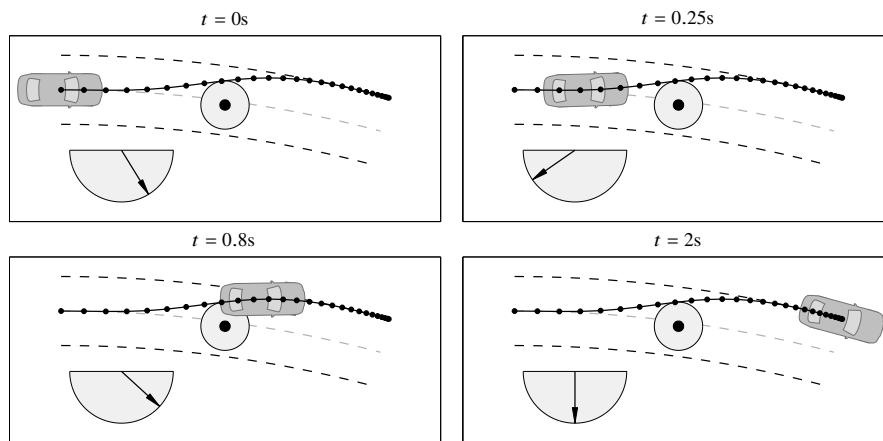
[Physical acceleration limits]



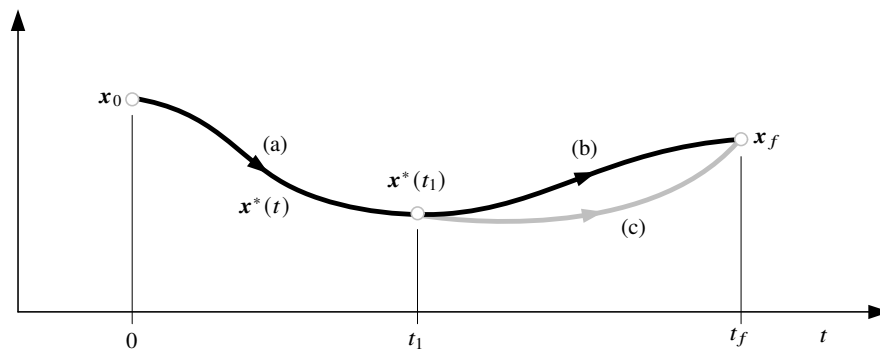
[System states]



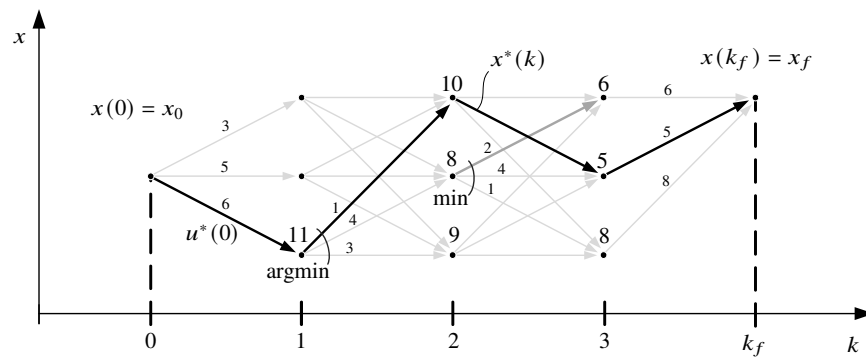
[Combined braking and steering]



[Bellman's principle of optimality]



[Discrete decision process]



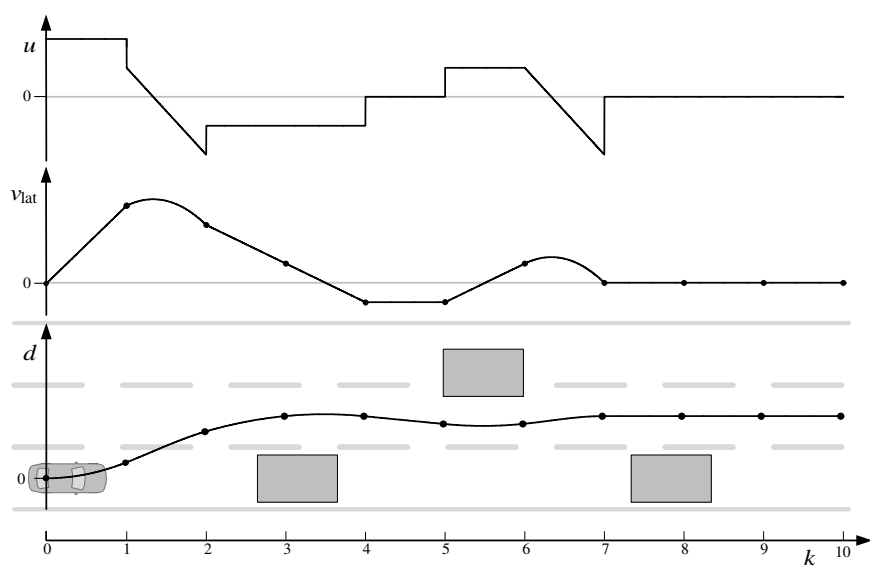
[Value iteration]

```

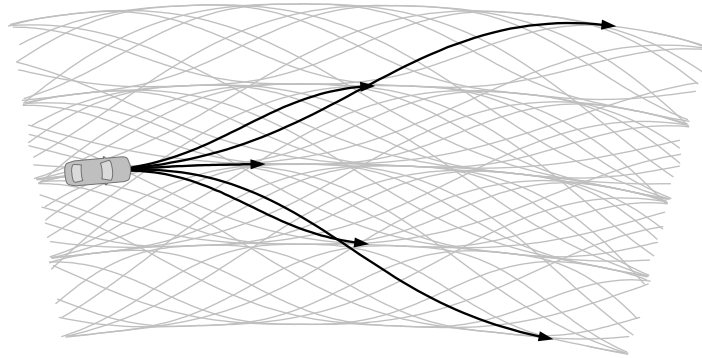
1:  $G^*(\mathbf{x}_f, k_f) \leftarrow 0$ 
2: for  $k = k_f - 1$  to 0 do
3:   for all  $\mathbf{x} \in \mathcal{X}$  do
4:      $G^*(\mathbf{x}(k), k) = \min_{\mathbf{u}(k)} \{ l(\mathbf{x}(k), \mathbf{u}(k), k) + G^*(f(\mathbf{x}(k), \mathbf{u}(k), k), k+1) \}$ 
5:   end for
6: end for

```

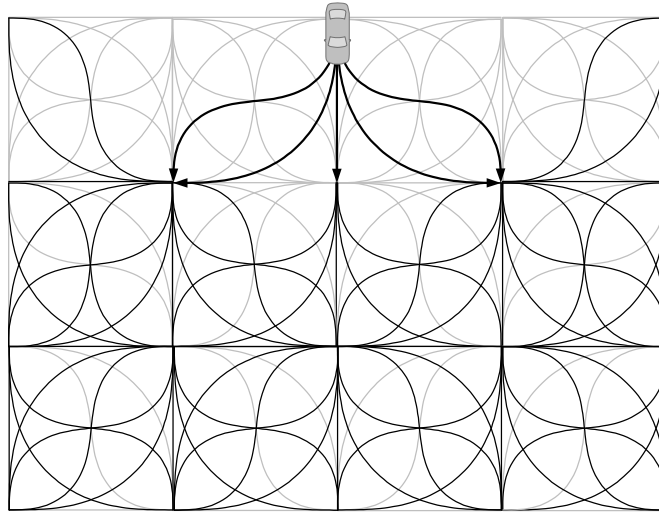
[Dynamic programming example]



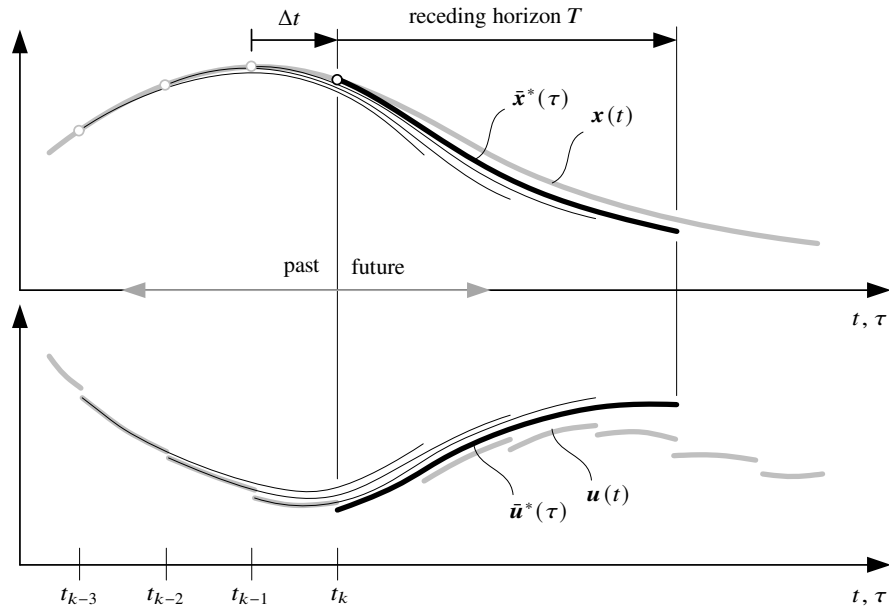
[State lattice for structured environments]



[State lattice for unstructured environments]



[Receding horizon optimization]



Literaturverzeichnis

- [1] BRANDT, T. : *A predictive potential field concept for shared vehicle guidance*, Universität Paderborn, Diss., 2008
- [2] DIEHL, M. ; BOCK, H. ; DIEDAM, H. ; WIEBER, P. : Fast Direct Multiple Shooting Algorithms for Optimal Robot Control. In: *Fast Motions in Biomechanics and Robotics* 340 (2006), S. 65–93
- [3] EICHHORN, A. ; WERLING, M. ; ZAHN, P. ; SCHRAMM, D. : Maneuver Prediction at Intersections using Cost-to-go Gradients. In: *International Conference on Intelligent Transportation Systems* IEEE, 2013
- [4] GAYKO, J. : Lane Keeping Support. In: WINNER, H. (Hrsg.) ; HAKULI, S. (Hrsg.) ; WOLF, G. (Hrsg.): *Handbuch Fahrerassistenzsysteme*. Wiesbaden : Vieweg + Teubner Verlag, 2012, S. 554–561
- [5] KOREN, Y. ; BORENSTEIN, J. : Potential field methods and their inherent limitations for mobile robot navigation. In: *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on IEEE*, 1991, S. 1398–1404
- [6] KROGH, B. : A generalized potential field approach to obstacle avoidance control. In: *International Robotics Research Conference*. Bethlehem, Pennsylvania, 1984
- [7] LATOMBE, J. : *Robot Motion Planning*. Springer Verlag, 1990. – ISBN 0792391292
- [8] LAVALLE, S. : *Planning Algorithms*. Cambridge University Press, 2006
- [9] LEWIS, F. ; SYRMOS, V. : *Optimal Control*. John Wiley & Sons, Inc. <http://de.scribd.com/doc/49575744/Opti-Control-Lewis>

- [10] WERLING, M. ; LICCARDO, D. : Automatic collision avoidance using model-predictive online optimization. In: *Conference on Decision and Control* IEEE, 2012, S. 6309–6314