Comparison of Local vs. Global Optimization for Trajectory Planning in Automated Driving

M. Ruf*† J.R. Ziehn*‡ D. Willersinn* B. Rosenhahn‡ J. Beyerer*† H. Gotzig§

Abstract: Trajectory planning for automated driving has recently seen a shift from highly discrete models towards continuous models related to the calculus of variations. These allow intuitive, elegant and flexible formulations, however, a major challenge of these models lies in finding their optima. Current iterative solvers are prone to terminating in local optima, being sensitive to initial parameters, violating constraints and exceeding real-time restrictions. A possible solution is presented in [1], where such variational models are transformed into Hidden Markov Models (which provide global optimization) while retaining the original optimization goals. This paper evaluates the necessity of such global optimization by comparing local and global approaches on several realistic traffic scenarios.

Keywords: fully-automated driving, global optimization, path planning, trajectory planning

1 Background and Motivation

Trajectory planning for automated driving has been expressed through a wide variety of models. In the DARPA Grand Challenge [2], the models chosen by the successful participants are characterized by a high degree of discretization, in several ways: Firstly, most approaches chose to implement not a single trajectory planner, but multiple dedicated subsystems tailored to specific situations and activated by a higher-level logic when a specific situation is recognized. Secondly, most of the individual planners themselves applied combinatorial methods that can be ascribed to a school of artificial intelligence substantially influenced by [3], which lays a strong focus on graph-based models, such as graph searches or Bayesian networks. This general paradigm can be found again, more pronounced, in the subsequent DARPA Urban Challenge, where many teams refined their methods through the lessons learnt in the Grand Challenge.

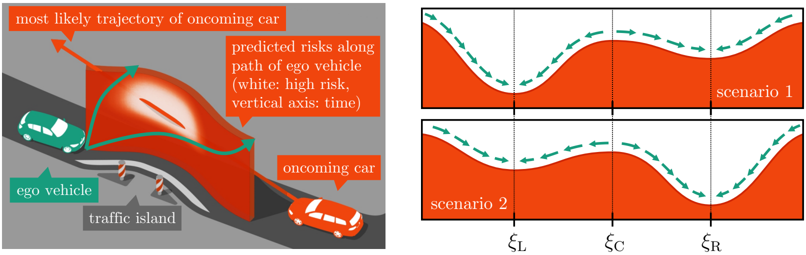
In more recent years, it may denote a paradigm shift that several publications picked up models from the continuous domain. [4] describes a model for evasive emergency maneuvers based on continuous non-linear model predictive control (NMPC); [5, 6] propose a model from the *calculus of variations* to represent trajectory planning for all traffic situations in a unified and continuous way; [7] presents a similar approach also using the

*{`miriam.ruf, jens.ziehn, dieter.willersinn, juergen.beyerer`}`@iosb.fraunhofer.de`, Fraunhofer IOSB, 76131 Karlsruhe

†Vision and Fusion Laboratory (IES), Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe

‡{`rosenhahn, ziehn`}`@tnt.uni-hannover.de`, Institut für Informationsverarbeitung (TNT), Leibniz Universität Hannover, 30167 Hanover

§`heinrich.gotzig@valeo.com`, Valeo Schalter und Sensoren GmbH



(a) A simple traffic scenario with two local optima: The ego vehicle (green), an oncoming vehicle (red) and a field of predicted occupancy risks (red: low; white: high). Local optima (lower: passing before the red car; upper: waiting) are indicated in green.

(b) Abstract view of two local optima in two different variants of the scenario. The red curve shows the penalty values for the different trajectories, the green arrows show the directions in which local optimizers will converge from an "initial guess".

Figure 1: A simple traffic scenario with two local optima. The ego vehicle (green) must pass a traffic island and cross the opposite lane. An oncoming car (red) will use the same lane. There exist two local optima (passing before the other car, or behind it), between which no continuous transformation of vehicle parameters exists, that also provides a safe solution. The optima are thus distinct, and considering the exact geometry of the scenario, traffic laws, safety and comfort requirements, one will usually be more desirable. Local iterative optimizers require an "initial guess", and this guess can have more bearing on the optimization result than the actual scenario. An intuitive representation is shown in (b), where the horizontal axis represents an abstract trajectory space, and the vertical axis the penalties assigned to each trajectory in a given scenario. ξ_L/ξ_R may represent passing before/after the oncoming car, which in scenario 1 is very far away, and very close in scenario 2. While ξ_L is the best trajectory in scenario 1 by far, and ξ_R in scenario 2, this circumstance is irrelevant to most iterative solvers. Instead, *regardless of the scenario*, any initial guess left of ξ_C will converge to ξ_L, and any initial guess right of ξ_C will converge to ξ_R[1]. This property is largely independent of concrete model choices, as both maneuvers exist in the real world, may represent safe trajectories and yet there is no way to continuously modify one maneuver into the other without traversing a higher-penalty maneuver.

calculus of variations, which proved successful in the acclaimed Bertha-Benz drive of 2013.

The continuous models provide several advantages over the earlier combinatorial models. They allow a formulation that sticks close to physical intuitions of vehicle dynamics; they do not require an abstract planning space that has to be equipped with a separate physical interpretation (such as building a search tree from a finite set of preselected maneuvers); they can easily be extended to consider a wide range of different goals within the same framework; and they are, at the same time, efficient to compute, as previous applications in other domains rendered them a well-understood topic.

One considerable challenge of these models stems from the fact that classical solution methods are iterative, such as gradient descents or SQP. These methods are known for problematic behavior that particularly concerns real-time applications: Convergence to an optimal solution can not be guaranteed in general; solutions may be local optima (see Fig. 1), stationary points or none of the before; convergence may not be attained in time at all; hard constraints may be violated in the numerical process, so the resulting trajectory is not even physically driveable. One possible solution is presented in [7],

[1] It must be noted here that this observation only applies to the interval and parameters shown in Fig. 1b. In general, it is very well possible that, for example, initial guesses right of ξ_R will converge to ξ_L, and vice versa, adding to the unpredictability of iterative solutions. Examples of this behavior will be given in Sec. 3.

where the environment model and optimization goals were explicitly set up to assure the existence of a single local *and* global optimum. However, the model choices therein do not immediately extend to general, dynamic traffic situations, as found, for example, in a left-turn maneuver through oncoming traffic. [8] uses a combinatorial approach to enumerate all possible *maneuver variants* and choose desirable candidates, but requires a finite set of predicted trajectories for other traffic participants; the solution is thus inapplicable to the models used in [6, 9], which use continuous probability densities to incorporate uncertain predictions and sensor uncertainties.

A solution for the latter was suggested in [1], which provides *global* optimization of general variational problems by transforming given variational models into equivalent *Hidden Markov Models* (HMMs), thereby establishing a bridge between combinatorial and continuous models. As the transformation preserves optimization goals and thus optima, it can be used in combination with iterative methods to efficiently improve the quality of the solution. This paper compares state-of-the-art local optimization with the approach presented in [1] on real-world traffic situations, to highlight risks associated with iterative solvers even in simple and common environments.

2 Euler–Lagrange and Hidden Markov Models

The scope of this paper allows but a brief recapitulation of the use of variational Euler–Lagrange Models in automated driving and of the transformation to Hidden Markov Models, as is given in this section. The reader is asked to seek more thorough descriptions of the underlying models and the transformation in [6, 7, 10] (for the ELMs) and in [1, 11] (for the transformation to HMMs).

2.1 Euler–Lagrange Models and the Calculus of Variations

Optimizing continuous trajectories in general is a problem from the *calculus of variations.* The calculus of variations considers *functionals* $\mathcal{P}$ (here representing *total penalty*), which map a function space Ξ to the real numbers, and aims to find the optimal function $\xi^* \in \Xi$ for which $\mathcal{P}$ takes its lowest value, formally:

$$\xi^* = \arg\min_{\xi \in \Xi} \mathcal{P}[\xi] \tag{1}$$

In automated driving, Ξ is the space of possible trajectories for the ego vehicle. In general, any computable $\mathcal{P}$ is conceivable, and any generalized coordinates and parametrization can be chosen for the trajectories ξ, since (1)—for automated driving—merely represents the general goal of finding the best continuous sequence of parameters that uniquely define a behavior for the ego vehicle. Usually, and in all models discussed here, the parameter of ξ is *time* t. Several recent works ([4, 5, 6, 7]) have considered a special form of $\mathcal{P}$ to be sufficient for the tasks of automated driving,

$$\mathcal{P}[\xi] = \int_{t_1}^{t_2} \mathrm{d}t \, p(\xi(t), \dot{\xi}(t), \ddot{\xi}(t), ..., \frac{\mathrm{d}^\omega}{(\mathrm{d}t)^\omega}\xi(t), t), \tag{2}$$

which is based on the choice of a single function p, the *Lagrangian.* The Lagrangian assigns a *local penalty* value dependent on time t, and on the current values of ξ and its first ω derivatives w.r.t. t, $\dot{\xi}(t)$, ..., ${}^{\mathrm{d}^\omega}/_{(\mathrm{d}t)^\omega}\xi(t)$. p can penalize collision risks or lane departures through ξ, deviations from speed limits through $\dot{\xi}$, and uncomfortable and ecologically inefficient maneuvers through $\ddot{\xi}$ and $\dddot{\xi}$ (for $\omega = 3$).

The key simplifications of (2), called here the *Euler–Lagrange model* (ELM), are:

▷ Trajectories can only be rated based on local properties, through ρ. It is generally impossible, for example, to optimize the relative position of several isolated points along a trajectory.

▷ Trajectories can only be rated based on their local, finite Taylor expansion. Properties not expressible in this form, as well as discontinuities up to ${}^{\mathrm{d}^\omega}/_{(\mathrm{d}t)^\omega}\xi$, cannot be considered.

The gain from the above limitations is that the optimization of ELMs is well-understood due to their long-standing applications in the physics of particle trajectories. In particular, they allow to analytically define the variational equivalent of the gradient $\nabla\mathcal{P}$, known as the *first variation*, such that $\nabla\mathcal{P}|_{\xi^*} \equiv 0$ (which is known as the *Euler–Lagrange equation*, hence the name). $\nabla\mathcal{P}$ can be used in iterative gradient descent methods, and, along with an analogous definition of a Hessian $\nabla^2\mathcal{P}$, with variants of sequential quadratic programming (SQP, [7, 12]). In this case, trajectories ξ are not considered analytically, but instead as a sequence of some $T+1$ discrete points, spaced with $\Delta t = (t_2 - t_1)/T$. However, as with common vector optimization, the iterative optimization of trajectories is prone to several pitfalls, detailed in [11]. The most relevant are, in short:

▷ Iterative optimization can get stuck in local optima that are not the global optimum (as shown in Fig. 1), or even just stationary points. In this case, the solution depends critically on the "initial guess".

▷ The optimization result can depend sensitively and counterintuitively on the choice of the "initial guess", as well as on optimization parameters such as step size or step direction.

▷ It can take arbitrarily long to converge or even diverge. This is particularly problematic in real-time applications that have a sharp limit on the computation time.

▷ Enforcement of hard constraints is generally possible, but cannot be guaranteed for arbitrary constraints.

Whether or not these issues arise in a given application depends very much on the problem layout. For the particular application of automated driving, this paper will show in Sec. 3, that even for simple and common traffic situations, and under rather general assumptions concerning the problem layout, local optima may be plentiful and can present a hazardous challenge for iterative optimizers.

2.2 From Euler–Lagrange Models to Hidden Markov Models

To address the issues of local, iterative optimization, [1] presents a method to transform ELMs to *Hidden Markov Models* (HMMs), which allows to apply methods from HMMs, in particular *global* optimization, to ELMs. [1] demonstrates that HMMs and ELMs are dual in the sense that both models make the same set of simplifying assumptions, and thus HMMs can be considered the discrete version of ELMs. This section will outline the key aspects of the transformation, and describe briefly, why the formulation as an HMM is considered advantageous.

HMMs are probabilistic graphical models, and an extension of *Markov chains*. They describe a probabilistic state machine that starts in a random state $x \in X$ based on a starting probability $p(x{\leftarrow}\!\dashv)$, and transitions—over discrete time steps τ—at random between states $x \in X$, based on *transition probabilities* $p(x_j \leftarrow x_i | x_i)$. While in each state, the automaton randomly emits a single *symbol* ς from a given set Σ, based on an *emission probability* $p(\varsigma|x_i)$ which differs between the states. Now, the most common application of this model is to consider a sequence $\sigma = [\sigma_1, ..., \sigma_T]^\top$ ($\sigma_t \in \Sigma$) of emitted symbols *known*, and the sequence of underlying states $\xi = [\xi_1, ..., \xi_T]^\top$ ($\xi_t \in X$) *unknown*. While in general no symbol σ_t uniquely defines ξ_t, the differing emission probabilities make certain states at a given time t more likely than others, given σ_t; the transition probabilities in turn decide whether a sequence is possible and likely in itself.

The most likely sequence of states ξ^* given a sequence of observed symbols σ is then the ξ to maximize the following expression:

$$p(\xi|\sigma) \propto p(\xi,\sigma) = \Big(\prod_{\tau=1}^{T} p(\sigma_\tau|\xi_\tau) \cdot p\big(\xi_\tau \leftarrow \xi_{\tau-1}\big|\xi_{\tau-1}\big)\Big) \cdot p(\sigma_0|\xi_0) \cdot p\big(\xi_0 \hookleftarrow\big), \tag{3}$$

The key simplification of the HMM is to assume the transition and emission probabilities to be dependent only on the current state (the *Markov property*), and thus to compose the total probability as an accumulation of local probabilities. [1] shows that the set of given probability distributions, in particular $p(x_j \leftarrow x_i|x_i)$ and $p(\varsigma|x_i)$, corresponds to the Lagrangian $\wp$ in an ELM. A particularly intuitive case of a transformation can be given for a Lagrangian of the form

$$\wp(x,\dot{x},t) = \wp_1(x,t) + \wp_2(x,\dot{x}), \tag{4}$$

which allows for the following transformation to the relevant probabilities of the HMM,

$$p(x_j \leftarrow x_i|x_i) = \frac{1}{Z_2}\exp\left(-\wp_2\left(x_j, \frac{x_j - x_i}{\Delta t}\right)\right) \quad \text{and} \quad p(\sigma_\tau|x_i) = \frac{1}{Z_1}\exp\left(-\wp_1\left(x_i, \tau\Delta t + t_1\right)\right), \tag{5}$$

which in turn directly yield the following correspondence between optima in the resulting HMM, and optima in the original ELM:

$$\arg\max_{\xi\in\Xi} p(\xi|\sigma) = \arg\min_{\xi\in\Xi} \mathcal{P}[\xi] = \xi^* \quad \text{and} \quad p(\xi^*|\sigma) = \frac{1}{Z}\exp(-\mathcal{P}[\xi^*]) \tag{6}$$

In the above transformation, which is based on the *Boltzmann distribution* from statistical mechanics, the constants Z, Z_1, Z_2 serve to establish proper probabilities and are uniquely determined by the problem; they also can be ignored in practical applications, as they do not affect the location of the minima in (6). The transformation connects the sequence of points in an ELM to the sequence of states in an HMM; the limited, local Taylor expansion used in the Lagrangian $\wp$ to the limited, local Markov memory; the time-dependence of $\wp$ to the time-dependence introduced by the HMM emissions; and the integral (or sum) of local Lagrangians $\wp$ to the multiplication of local probabilities. [1] shows that ELMs with arbitrary Lagrangians can be transformed into HMMs, however the models discussed here already lend to the simplification as in (4), if generalized states (consisting of locations, velocities and further derivatives) are used, as is common in variational optimization. The conversion of a given ELM to an HMM has the following main consequences:

▷ The original ELM can be optimized through the *Viterbi algorithm*, which finds the global optimal sequence of states (i.e. the trajectory ξ) in a fixed number of steps. As the search is exhaustive, the computation time of an HMM ($\mathcal{O}(T\cdot|X|^2)$ is generally slower than for the equivalent ELM (which is equally linear in T but does not depend on $|X|$) for large state spaces; however the computation time of iterative solvers depends on the concrete values in the problem and can grow arbitrarily large even for a fixed problem size ($\mathcal{O}(\infty)$). Furthermore, the fixed number of computation steps allows for an efficient hardware implementation of an HMM, while iterative ELM solvers require dynamic computation times.

▷ The free parameters to the Viterbi algorithm are the choice of a discretization of X (and of T, as in the ELM). Compared to the free parameters in iterative optimization ("initial guess", step size and direction, termination criteria, see Sec. 2.1), these parameters are considered considerably more intuitive and less delicate in their effect on the result. However, as the computation time scales with $|X|^2$, the state space must still be chosen carefully.

▷ The respective algorithms can be parallelized along a different problem dimension. An ELM can compute the next iteration step largely in parallel over all $T + 1$ time steps, but has to execute the iterations in sequence and does not scale with $|X|$. An HMM on the other hand must compute the T time steps in sequence, but can compute the entire state space X in parallel for each time step. The parallel capabilities available, and the size of the problem, thus can also affect the choice between ELM and HMM.

[1] shows HMMs to be real-time capable for automated driving tasks at a low resolution, and names three key applications of the transformation:

▷ HMMs can be used to directly optimize vehicle trajectories.

▷ HMMs can be used to provide initial guesses to an iterative optimizer at a low resolution, hopefully close to a global optimum. The iterative optimizer can then continue "smoothing" the trajectory at a finer resolution, based on the same optimization criteria.

▷ HMMs can be used to analyze given planning problems offline during research and development, when computation must not occur in real time. This is relevant since with iterative solvers, one can rarely be sure to have found the *global* optimum already, even with several initial guesses (see Fig. 5); the high-resolution HMM result can then be used to benchmark less expensive methods.

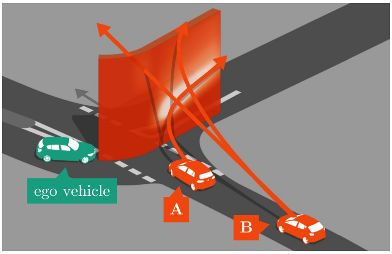
3 Practical Evaluation

For any of the aforementioned applications, two main considerations determine whether the transformation from a particular problem in ELM form to an equivalent HMM is worthwhile: Does the problem have very few and/or priorly known and/or very similar optima? Can the iterative optimization be guaranteed to terminate in time? If both answers are "yes", then there is little to be gained from a conversion to an HMM. [7] presents an approach that deliberately reduces the complexity of the situation to assure the existence of a single local and global optimum. However, we believe that arbitrary traffic situations cannot generally be reduced to this extent. This section will give several according examples. The focus is on an analysis of the local optima in traffic situations; some comments on execution times can be found in [1], but a serious comparison would require algorithms that are optimized to fully exploit their respective advantages; this debate is beyond the scope of this paper.

The Optimization Goals The basis for the evaluation is the ELM used in the *Situation Prediction and Reaction Control* (SPARC) approach (cf. [6]). We consider trajectories of world coordinates parametrized over time, and a Lagrangian of the form

$$p(\xi, \dot{\xi}, \ddot{\xi}, t) = p_{\text{in}}(\xi, \dot{\xi}, \ddot{\xi}) + p'_{\text{out}}(\xi, \dot{\xi}, t) + p''_{\text{out}}(\xi, \dot{\xi}, t), \tag{7}$$

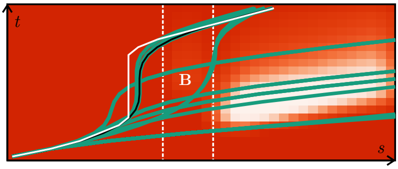
where p_{in} denotes the *inner penalties* related to the geometry of the trajectory (sharp accelerations, high speeds in sharp curves), p'_{out} denotes the *primary outer penalties* related to expected risks of collisions at location ξ and time t, and p''_{out} denotes the *secondary outer penalties* related to non-vital interactions with the environment, such as traffic rule



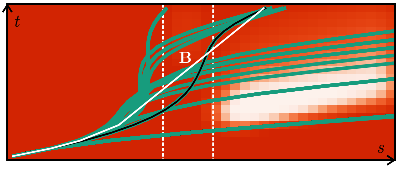
(a) Sketch of the situation with predicted risks.

(b) Aerial view of the situation.

(c) Ground view of the situation. The front car **A** is indicating a right turn, the rear car **B** is not.



(d) Results for narrow spacing between **A** and **B**.

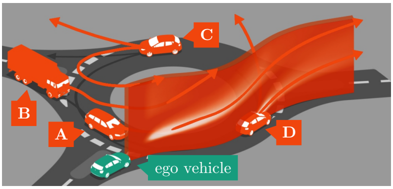


(e) Results for wide spacing between **A** and **B**.

Figure 2: Scenario I: A left turn through oncoming traffic from Bannwaldallee into Grießbachstraße, Karlsruhe. Two variants are compared: Narrow spacing between the vehicles in (d) and wide spacing in (e). For both variants, the iterative solutions (green lines) include local minimum passing in between the cars, even though the risk in the narrow spacing is high (white map entries). There also exist local minima that pass the intersection before *both* cars, which are safe but highly uncomfortable due to relatively high speeds in a sharp curve. The best solution in (d) however is to wait for both cars, while in (e) the gap is wide enough to cross, as suggested by the HMM (white) and subsequently refined iteratively (black).

violations (such as exceeding speed limits or overtaking a car on the wrong side) or potholes. p_{in} is computed in a space–time volume of expected risks, based on predicted occupancy probabilities for other traffic participants and objects. All penalty terms are smooth and and analytically differentiable, except for the predicted collision risks, which can be discontinuous. The only hard constraints are put on the physical limits of the ego vehicle, such as maximal and minimal accelerations and velocities. The prediction principles are based on the descriptions in [9]; however, as is argued here and indicated in Fig. 1, the existence of multiple optima is not an artifact of specific prediction methods, but rather the fundamental nature of and sufficiently complex traffic situations.

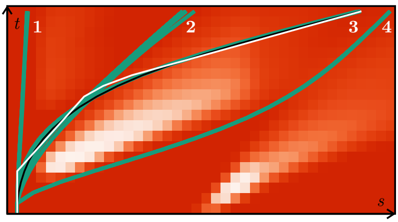
The Optimization Methods Each scenario compares the result of the HMM, based on the principles laid out in [1], to the results obtained from the MATLAB constrained optimization function *fmincon*, which is an iterative local optimizer based on sequential



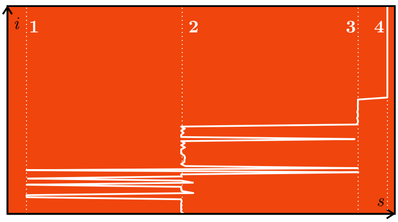
(a) Sketch of the situation with predicted risks.

(b) Aerial view of the situation.

(c) Ground view of the situation. The ego vehicle wants to leave at the first exit, following **D**.



(d) Results with numbered endpoint clusters.

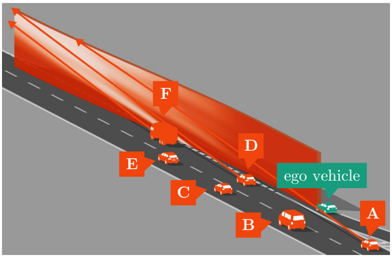


(e) Endpoint jumps over initial guess index.

Figure 3: Scenario II: A roundabout between Raffineriestraße, DEA-Sholven-Straße and Esso-Straße outside of Karlsruhe. In Tab. 1, this scenario stands out as having the highest J_{end}, indicating that the obtained local minima jump randomly with the initial guesses. In between initial guesses converging to endpoint **2**, there are many guesses converging to **1** and **3**. It is thus hard to predict the ξ^{i}_{ELM} from a given $\xi^{(i)}_{\text{ELM}}$, as very small changes can have significant impacts. Table 1 also shows that the discretization of the HMM in this scenario has the most notable effect: The ELM using the HMM result as an initial guess is able to improve the HMM result by about 30 %.

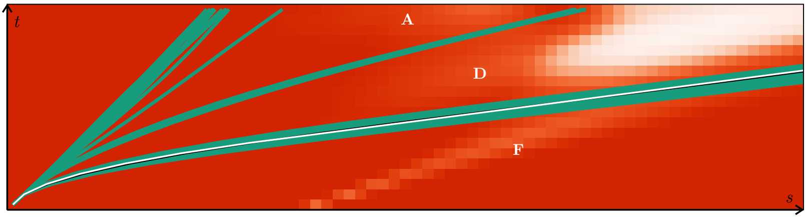
quadratic programming (SQP), using *line search* and *active set* or *interior point methods* for constraint enforcement (cf. [12] for details). The maximum number of iterations was set to 800, and the iterative optimization was started from a series of approximately 100 "initial guesses" (each accelerating the car to a different speed and keeping the speed constant from then on), which converged to several local optima. The iterative results should be considered in view of the fact that *fmincon* is an advanced optimizer; less sophisticated methods can get stuck in considerably more optima, and have trouble enforcing the imposed hard constraints.

The Scenarios The scenarios presented here are based on aerial images gathered in the vicinity of Karlsruhe. For easier reference, each scenario is provided with a ground view of the situation, with a sketch showing the involved participants (not to scale) and with the space–time map of the predicted occupancy probabilities.



(a) Sketch of the situation with predicted risks.

(b) Ground view of the situation.



(c) Results with car indices. Merging between **A** and **D** has an increased risk, as **B** might merge here. The optimal solution is to merge behind **F**. Several local optima suggest to not leave the ramp at all.

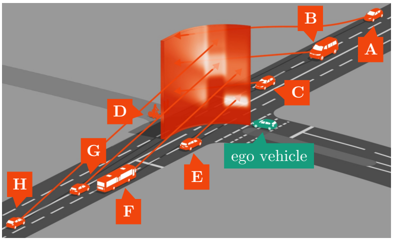
Figure 4: Scenario III: Entering the L605 highway from Oberreut, outside of Karlsruhe. Table 1 shows that this scenario has local optima of very different quality, yet, due to the very structured environment with all surrounding cars driving at a very similar speed, the local optima are continuously traversed and found by the initial guesses, with hardly any jumps on the endpoints. Coincidentally, the discretization of the HMM in this scenario matches the requirements of the situation so well that even the iterative solver can hardly improve its solution. This, however, does not hint at a general rule.

The scenarios provided in this section assume a fixed path, and thus only optimize the timing along the path. This simplification was chosen to allow for a more obvious visual representation, but presents no principal limitation to the approach (a general discussion of longitudinal and lateral planning can be found in [11]). Only those scenarios are shown where the lateral planning options are limited by the nature of the scenario; however, even these cases are challenging for iterative optimization. In multidimensional planning, the number and diversity of local optima increases further.

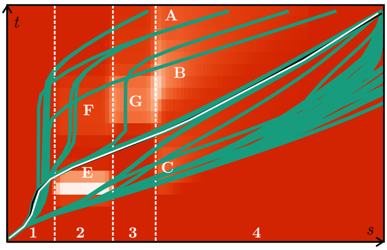
The Metrics To evaluate the scenarios in terms of the topology of local optima and of the effort for the respective solvers, we establish the following variables and metrics that will be given in the corresponding figures. To facilitate the understanding, we have normalized the time index to $t \in [0,1]$ and the space scale to $\xi \in [0,1]$ respectively, and the penalties such that the penalty of ξ_{HMM}, the trajectory obtained directly by the HMM, is $\mathcal{P}_{\text{HMM}} = 1$. Thus, an iterative ELM solution yielding $\mathcal{P}_{\text{ELM}} = 1.5$ will be 50 % worse in terms of optimality than the HMM solution.

- ▷ $\xi_{\text{ELM}}^{(i)}$ and ξ_{ELM}^{i}: The i-th “initial guess”, and the trajectory to which the iterative solver converges from that guess.
- ▷ $\xi_{\text{ELM}}^{(*)}$ and ξ_{ELM}^{*}: The best initial guess and the trajectory it converges to, which has the lowest $\mathcal{P}$ obtained from any ELM guess.

(a) Ground view (cont'd on the next page). The bus is expected to turn right with high probability.



(b) Sketch of the situation with predicted risks.



(c) Results with marked lane locations and cars.

Figure 5: Scenario IV: The intersection of Adenauerring and Am Fasanengarten in Karlsruhe. Strikingly in Tab. 1, this scenario poses the greatest challenge to the iterative optimizer. 90 % of the iterative trajectories are worse than the HMM result by 100 %, some even by more than 400 %. This is mostly due to the high complexity of the maneuver, crossing three lanes and including locations where collisions with more than one vehicle are expected (in between lanes). The scenario however also differs from the previous ones in that it is the only where trajectories with similar endpoints have significantly different penalty values. Hence, even though the moderate J_{end} indicates (correctly) that the trajectories mostly vary smoothly with the initial guesses, $J_{\mathcal{P}}$ is highest in this scenario by far. The reason for this is that here, more than in any other scenario, very undesirable local minima and relatively safe local minima are densely packed. There is, for example, a local minimum that chooses to likely collide with **G**, because it only sees the alternatives of colliding with **G** *and* **F** or **B**. None of the initial guesses exactly finds the global optimum found by the HMM.

▷ $\xi_{\text{HMM}\to\text{ELM}}$: The trajectory found by the iterative solver when using ξ_{HMM} as the initial guess.

▷ I^*_{ELM}: The number of iterations required to converge from $\xi^{(*)}_{\text{ELM}}$ to ξ^*_{ELM}.

▷ $I^{\min}_{\text{ELM}}$, $I^{\max}_{\text{ELM}}$, $\langle I_{\text{ELM}} \rangle$, $\text{med}(I_{\text{ELM}})$: The minimum, maximum, mean and median number of iterations from $\xi^{(i)}_{\text{ELM}}$ to ξ^i_{ELM}, over all i.

▷ $I_{\text{HMM}\to\text{ELM}}$: The number of iterations required to converge from ξ_{HMM} to $\xi_{\text{HMM}\to\text{ELM}}$.

▷ $\mathcal{P}^i_{\text{ELM}} = \mathcal{P}[\xi^i_{\text{ELM}}]$: The penalty value to which the iterative solver converged from the i-th initial guess.

▷ $\mathcal{P}_{\text{HMM}\to\text{ELM}}$: The penalty value obtained by using the solution found by the HMM as an initial guess to the iterative solver.

▷ $\min_i \mathcal{P}^i_{\text{ELM}}$, $\max_i \mathcal{P}^i_{\text{ELM}}$, $\langle \mathcal{P}_{\text{ELM}} \rangle$, $\text{med}(\mathcal{P}_{\text{ELM}})$, $\sigma(\mathcal{P}_{\text{ELM}})$: The minimum, maximum, mean, median and standard deviation of $\mathcal{P}^i_{\text{ELM}}$ over all i. This hints at the probability distribution for obtaining a certain penalty value by trying out initial guesses at random.

▷ $p_{n\%}$: The probability of the iterative solver achieving an n % worse penalty result than the direct HMM result.

▷ $J_{\mathcal{P}} = \sum_i |\mathcal{P}^i_{\text{ELM}} - \mathcal{P}^{(i-1)}_{\text{ELM}}|$: The absolute amount of jumps of penalty values obtained from consecutive initial guesses. As the initial guesses are continuous in trajectory space, a large number of jumps between con-

Figure 5a (cont'd from previous page).

secutive results indicates a very diverse arrangement of optima.

▷ $J_{\text{end}} = \sum_i |\xi(1)^i - \xi(1)^{(i-1)}| / (\max_j \xi(1)^j - \min \xi(1)^j)$: The absolute amount of jumps in the obtained endpoints from consecutive initial guesses, divided by the largest distance of endpoints. As most local optima vary in their endpoints in the examples provided here, jumps in these endpoints indicate that the solver jumped to another local optimum altogether. As the initial guesses are continuous in trajectory space, it would be expected that the total number of jumps equals the distance from the "leftmost" to the "rightmost" locally optimal endpoint, as the iterative optimizer slips from one local optimum to the next. Instead, however, the optimizer may not converge to an optimum close to the initial guess, but instead to more remote optima. If this metric is 1, the iterative solutions vary orderly with the initial guesses. The higher this metric, the less ξ^i_{ELM} can be predicted from $\xi^{(i)}_{\text{ELM}}$. An example of this case is found in Fig. 3 / scenario II.

3.1 Discussion

The evaluation of scenarios is given in detail in Figs. 2–5. In each case, the approximately 100 solutions found by the iterative optimizer are shown in green, the solution found by the HMM is shown in white, and the iteratively refined HMM solution is shown in black. A full presentation of the quantitative results is given in Tab. 1. The analysis confirms the basic assumption that the iterative optimizer has difficulty finding the global solution through initial guesses. The HMM approach reliably finds solutions in all four scenarios that are, given the discretization, as close as possible to the global optimum, and—with the exception of scenario II—always within 10 % of its value. If the solution of the HMM is further refined by the iterative solver, the result is in all cases the best solution that is found by any method applied here.

Even though the scenarios were intentionally chosen to be mostly common situations of low complexity, they reveal some of the main problems associated with iterative optimization: Plentiful optima of very different quality (scenarios I and IV), unpredictable convergence from the initial guess (scenario II) and lack of convergence even after many iterations (scenarios II and IV). An effect that was notably absent was the violation of hard constraints, which were enforced successfully in all cases by *fmincon*. On the other hand, it can be stated that scenario II posed a notable challenge to both optimization methods and certainly merits further attention. Detailed descriptions of the particular effects can be found in the respective figures.

4 Conclusion and Outlook

This paper has provided a deeper insight into the situation of local space–time trajectory optima in several real-world traffic situations. Maneuvers to which local iterative optimization methods, such as SQP, may converge, were compared to global optima which are obtained by transforming the variational formulations into an equivalent HMM. The results show that even for very simple and common traffic situations, the scattered distribution of local optima can play a significant role in trajectory optimization. Free parameters in iterative solvers have a significant impact on the solution—at times even more than the actual problem structure. The evaluation indicates that the HMM approach is able to find optima deterministically and in fixed time that are very close to the best solutions found by iterative solvers in unlimited time. It also suggests that global optimization of very general environment models of automated driving is feasible in real time. The most promising implementation of the HMM-based approach is a hardware implementation, such as on an FPGA (cf. [1]); it thus must be demonstrated that such an implementation is able to produce solutions in real time.

Several solutions were recently proposed to address the issue of local optima, in particular [8] and [1]. While the transformation to an HMM, as in [1], can provide global optimization for arbitrary ELMs, highly structured models as in [8] are certainly more efficiently solved by other means. The models proposed in [1, 11] and discussed here are based on the assumption that environment models may (need to) evolve and become more complex and more stochastic, in which case it is considered inefficient at a certain point to develop specialized solvers for every new environment model, and instead apply a solver that can handle arbitrary problems of a more general form. Whether abstract and structured environment models, or fuzzy, stochastic models as used here are better suited for automated driving remains an open question.

References

[1] J.R. Ziehn, M. Ruf, B. Rosenhahn, D. Willersinn, J. Beyerer, and H. Gotzig. Correspondence between Variational Methods and Hidden Markov Models. In *Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV), Seoul*, June 2015.

[2] M. Buehler, K. Iagnemma, and S. Singh. *The 2005 DARPA Grand Challenge: The Great Robot Race*. Springer Tracts in Advanced Robotics. Springer, 2007.

[3] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3rd edition, 2003.

[4] M. Werling, Ph. Reinisch, and K. Gresser. Kombinierte Brems-Ausweich-Assistenz mittels nichtlinearer modellprädiktiver Trajektorienplanung für den aktiven Fußgängerschutz. In K. Dietmayer et al., editors, *8. Workshop Fahrerassistenzsysteme – FAS2012*, pages 77–86, September 2012.

[5] J.R. Ziehn. Energy-based collision avoidance for autonomous vehicles. Master's thesis, Leibniz Universität Hannover, Germany, October 2012.

[6] M. Ruf, J.R. Ziehn, B. Rosenhahn, J. Beyerer, D. Willersinn, and H. Gotzig. Situation Prediction And Reaction Control (SPARC). In B. Färber et al., editors, *9. Workshop Fahrerassistenzsysteme – FAS2014*, pages 55–66, March 2014.

	Scenario Ia	Scenario Ib	Scenario II	Scenario III	Scenario IV
Penalty distribution (colored in relation to the *column* minima and maxima)					
$\mathcal{P}_{\text{HMM}}$	1	1	1	1	1
$\mathcal{P}_{\text{HMM}\to\text{ELM}}$	0.92	0.94	0.77	1.00	0.98
$\min_i \mathcal{P}^i_{\text{ELM}}$	0.92	0.94	0.77	1.00	1.26
$\max_i \mathcal{P}^i_{\text{ELM}}$	3.91	3.81	1.29	2.52	5.13
$\langle\mathcal{P}_{\text{ELM}}\rangle$	1.13	1.15	1.18	1.77	3.34
$\text{med}(\mathcal{P}_{\text{ELM}})$	1.01	1.04	1.23	2.52	2.73
$\sigma(\mathcal{P}_{\text{ELM}})$	0.54	0.48	0.18	0.76	1.09
Solutions inferior to HMM (colored in relation to *row* minima and maxima)					
$p_{0\%}$	0.81	0.85	0.84	0.51	1.00
$p_{10\%}$	0.08	0.11	0.84	0.51	1.00
$p_{30\%}$	0.08	0.11	0.00	0.51	0.90
$p_{50\%}$	0.07	0.09	0.00	0.51	0.90
$p_{100\%}$	0.05	0.04	0.00	0.51	0.90
Iteration counts (colored in relation to *row* minima and maxima)					
$I_{\text{HMM}\to\text{ELM}}$	66.00	91.00	105.00	15.00	185.00
$I^{\text{min}}_{\text{ELM}}$	33.00	33.00	52.00	25.00	79.00
$\langle I_{\text{ELM}}\rangle$	58.01	55.16	130.04	63.66	252.58
$\text{med}(I_{\text{ELM}})$	49.00	49.00	110.00	62.00	206.00
$I^{\text{max}}_{\text{ELM}}$	137.00	104.00	$>$ 800.00	169.00	$>$ 800.00
I^{*}_{ELM}	67.00	96.00	100.00	71.00	215.00
Predictability metrics (colored in relation to *row* minima and maxima)					
$J_{\mathcal{P}}$	9.54	10.06	6.71	5.05	27.67
J_{end}	2.57	2.46	7.17	1.44	2.15

Table 1: Metrics for the evaluation presented in Sec. 3; the metrics are defined there.

[7] J. Ziegler, P. Bender, T. Dang, and C. Stiller. Trajectory planning for Bertha – A local, continuous method. In *Proceedings of the 2014 IEEE Intelligent Vehicles Symposium (IV), Dearborn*, pages 450–457, June 2014.

[8] Ph. Bender, Ö. S. Tas, J. Ziegler, and Ch. Stiller. The combinatorial aspect of motion planning: Maneuver variants in structured environments. In *Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV), Seoul*, June 2015.

[9] M. Ruf, J.R. Ziehn, D. Willersinn, B. Rosenhahn, J. Beyerer, and H. Gotzig. A Continuous Approach to Autonomous Driving. In *Vehicle and Infrastructure Safety Improvement in Adverse Conditions and Night Driving (VISION), Versailles*, October 2014.

[10] B. Van Brunt. *The Calculus of Variations*. Springer, 2010.

[11] M. Ruf, J.R. Ziehn, D. Willersinn, B. Rosenhahn, J. Beyerer, and H. Gotzig. Global Trajectory Optimization on Multilane Roads. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, September 2015.

[12] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, 2006.