Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Optimal Kinodynamic Risk Aversive Behavior Planning using Predictive Risk Maps

Florian Damerow, Julian Eggert

2015

Preprint:

This is an accepted article published in Intelligent Vehicles Symposium (IV) 2015. The final authenticated version is available online at: https://doi.org/[DOI not available]

Balancing Risk against Utility: Behavior Planning using Predictive Risk Maps

Florian Damerow[1] and Julian Eggert[2]

Abstract— **This paper addresses the problem of future behavior evaluation and planning for ADAS in general traffic situations. Complex traffic situations require the estimation of future behavior alternatives in terms of predictive risks. Based on the predicted future dynamics of traffic scene entities, we present an approach where a continuous, probabilistic model for future risks is used to build so-called predictive risk maps. These maps indicate how risky a certain ego-car trajectory will be at different predicted times so that they can be used to directly plan the best possible future behavior. Since this optimization problem is highly non-convex we combine the risk maps with sampling-based planning algorithms of the RRT*-type to obtain future trajectories which minimize risk and maximize utility. We apply our approach to multiple risk types and various different scenarios, including inner city and highway situations.**

I. Introduction

The target of future Advanced Driver Assistance Systems (ADAS) is to relief the strain on the driver, starting from simple warning systems and comfort functions such as blind spot information system or parking assistance up to partially or fully automated driving. We aim for a system that is generally able to securely support driving, especially in inner city scenarios with multiple behavior alternatives. Inner city scenes are highly complex, especially if more than a few traffic participant are involved. Current ADAS systems targeting at inner-city scenarios are either mainly reactive or designed to work under very narrowly defined conditions and therefore not applicable as a general approach for risk-based behavior control.

In complex situations, it is unfeasible to evaluate all possible state evolution alternatives of the involved traffic participants. A way to restrict the alternatives is to guide them by the behavioral needs of one of the entities. Still, predictions of the "behaviorally relevant" future dynamics of the regarded entity and of the other traffic participants as well as their relations are required. This then leads to an evaluation of the possible future behavior in terms of behavioral risk and utility for the entity in the context of the surrounding traffic participants. Based on such an evaluation the best future behavior can be planned.

In [1] a probabilistic scheme for continuously and consistently modeling different types of future risk is presented, which provides the basis of the risk evaluation from this paper. In [2] the idea of predictive risk maps is introduced. Predictive risk maps indicate how critical a certain behavior will be in the future and can thus be used to plan future behavior. The model for predictive risk used in [2] is a special case of the one introduced in [1]. It is based on characteristic risk indicators such as time of closest encounter (TCE), which can be seen as a generalization of the well known time to collision (TTC). In this work, we combine the full probabilistic approach from [1] with the risk map idea from [2] and a sampling-based algorithms for trajectory evaluation to efficiently plan future behavior in a way that minimizes predictive risk and maximizes utility of the resulting trajectories for the chosen entity.

Due to a lack of sufficiently general behavior evaluation measures in complex traffic scenarios, the area of risk assessment gained a lot of attention in recent years. The most common risk indicators used so far are the time-to-X (TTX) type indicators, such as time-to-collision (TTC) [3], [4]. Further similar measures are the headway time [5] and the required deceleration to avoid an incident [6]. The great benefit of these approaches is their simplicity and the very low computational costs. However they are generally restricted to well-defined scenarios such as car following or longitudinal collisions and not straightforwardly applicable to other scenarios. In [1] however, it could be shown how the TTX-like indicators are special cases that can be derived from a general, continuous risk theory. Furthermore, it could be shown that several different types of risk (e.g., longitudinal collision, close approaches, curve speeding, etc.) can be incorporated in similar ways into risk calculations.

Several other, "sampling-based" approaches concentrate on a specific type of risk, as e.g. [7], [8] on collision risk, by building a large variation of predicted future trajectories for the ego-car and the other traffic scene entities and then checking for spatial overlap. In such approaches, the result strongly depends on the decision on which trajectories to choose. In addition, the expected damage is usually not taken into account, which is e.g. of high importance if a collision is inevitable and the best behavior has to be selected to reduce an upcoming damage (see however [9] for an approach that takes this into consideration). A drawback of these approaches is the very high computational costs due to the extensive trajectory generation, the bias in the trajectory selection, the limitation to collision risks, and the neglection of the expected damage.

To deal with the complexity of trajectory generation, rapidly exploring random tree (RRT) - based approaches [11] are used to efficiently search through the space of

[1]Florian Damerow is with the Control Methods and Robotics Lab, Technical University of Darmstadt, 64283 Darmstadt, Germany florian.damerow@rmr.tu-darmstadt.de

[2]Julian Eggert is with the Honda Research Institute (HRI) Europe, Carl-Legien-Str. 30, 63073 Offenbach, Germany julian.eggert@honda-ri.de

future trajectories. RRT-based approaches can easily handle trajectory search problems with obstacles and kinematic and dynamic constraints, including constraints given by risk considerations. E.g., in [10], the set of reachable surrounding moving entities at an intersection is calculated and then the overall earliest time to collision is taken as a threat/risk measure. In [12] RRT's are used to guide a wheelchair through an environment with multiple other moving entities also including some risk function, the collision risk, into the path planning algorithm. However, we have seen that in order to select the most appropriate future behavior planning in complex traffic situations, a more general risk evaluation is necessary.

For this purpose, a risk-based constraint scheme for trajectory evaluation (and RRT's in particular) is necessary. In this paper, starting with section II, we therefore focus on a continuous, probabilistic, future risk estimation scheme, which also takes the expected damage related to future critical events into account. The estimation is based on the probability that the entity actually remains "safe" until an event happens ("survival probability") and a model for the event to actually happen at a future point in time ("event rate"). We use this risk estimation to predict several different kinds of future risks such as a continuous distance-based collision risk from each other traffic participant, applicable at e.g. intersections and highways, and the risk caused by high centrifugal forces when driving too fast around a curve. The estimation of the combined, total risk is then used in section III to build so-called predictive risk maps, which indicate how critical a certain behavior will be in the future. In section IV, we apply an RRT*-based algorithm to efficiently find a globally optimal trajectory through the risk map, which minimizes risk and maximizes utility. In section V the approach is applied to different scenarios like curve driving and basic intersections, as well as in more complex intersection scenarios, where we show how to combine curve driving with intersection behavior in a single behavior planning scheme.

II. Predictive Risk Estimation

Risk is in general defined as the expectation value of the cost or benefit related to future critical events [13], [14]. As stated in [1] this can be expressed in a probabilistic way where $< ... >$ represents the expectation value,

$$risk =< damage >$$
$$= \int_{\text{states}} \text{damage}\, P(\text{damage}|\text{states})\, P(\text{states})\, d\text{states}, \quad (1)$$

with

$$P(\text{damage}|\text{states})$$
$$= P(\text{damage}|\text{states}, \text{event} = \text{true})\, P(\text{event} = \text{true}|\text{states}). \quad (2)$$

Probabilistic risk includes a prediction of *future* critical events and the prediction of damage in case that the related event actually happens, in the time interval $[t, t+s]$.

$$P(\text{damage}_{t+s}|\text{states}_t) \quad (3)$$
$$= P(\text{damage}_{t+s}|\text{states}_{t+s}, \text{event}_{t+s} = \text{true}) \quad (4)$$
$$P(\text{event}_{t+s} = \text{true}|\text{states}_t), \quad (5)$$

where the first term represents the probability that a certain damage will occur, assuming that the event occurs at future time $t+s$. The second term represents the probability that the event happens at time $t+s$, given the known states at the current time t.

In order to advance from the general definition of risk to an actual risk measure we have to model 1) the prediction of future states using a proper prediction model, 2) the damage probability using a damage approximation model, and 3) the probability of the future event happening at time $t+s$, which can be done using a model for a stochastic process similar to a Poisson process. As shown in [1] the future event probability $P(\text{event}_{t+s} = \text{true}|\text{states}_t)$ can be modeled as in (6) using a so called inhomogeneous survival function $S(s;t)$ in combination with a total event rate $\tau^{-1}(\text{states}(t+s'))$,

$$P_E(s;t,\delta t) = \{\tau^{-1}(\text{states}(t+s))\delta t\} S(s;t), \quad (6)$$

with

$$S(s;t) = \exp\{-\int_0^s \tau^{-1}(\text{states}(t+s'))\, ds'\} \quad (7)$$

and

$$\tau^{-1}(\text{states}(t+s')) = \sum_i \tau_i^{-1}(\text{states}_i(t+s)). \quad (8)$$

Every single event rate τ_i^{-1} can be modeled using appropriate risk indicators as shown in the following on the example of car-2-car collision and high centrifugal force in curves.

The survival probability is the probability that a certain entity "survives" during the continuous time interval $[t, t+s]$ [1]. This probability decreases monotonically with time. Additionally if the total event rate increases, the survival probability decreases faster, because the probability that the entity gets involved in a risky event also increases. The survival probability has the effect that critical events that are predicted to occur further away in the future are usually considered as less probable. This is a desired property which matches with the intuition that, with more distant time, the likelihood to escape a critical event increases because (i) the state prediction gets more inaccurate and (ii) a driver may have more time to start a reaction that leads to the avoidance of the event. The future predicted risks therefore get "discounted" by the survival function, which at the same time serves to incorporate all the uncertainties (explicit and implicit) in the prediction process.

Assuming that critical events are usually triggered by a single cause (a car either collides *or* drifts out of the curve), different risk sources are superposed in the total event rate (8).

The car-2-car collision risk is modeled by an event rate that depends on the distance between two traffic participants,

$$\tau_d^{-1} = \tau_{d,0}^{-1} \exp\{-\beta_d(d - d_{min})\}, \tag{9}$$

where d is the distance between the ego-car and another traffic participant and d_{min} the minimally allowed distance corresponding to a physical overlap.

The accident risk of loosing control in a curve can be modeled as

$$\tau_c^{-1} = \tau_{c,0}^{-1} \exp\{-\beta_c(v - v_{c,max})\}, \tag{10}$$

where v is the longitudinal velocity while driving along the curve, $v_{c,max} = \sqrt{a_{c,max} R}$ the maximal longitudinal velocity with the maximal centrifugal acceleration $a_{c,max}$, and R the curve radius at the driving point.

The parameters $\tau_{d,0}$, $\tau_{c,0}$ define the event rate at minimal distance resp. maximal longitudinal velocity in the curve. β_d, β_c define the steepness of the event rates. Together, (9) and (10) describe a continuous, instantaneous event rate for two different types of risk, whose parameters (esp. the steepness parameters) are used to describe uncertainties in the involved context variables d, v and R. Further risk sources can be modeled in a similar way and superposed in the total event rate (8).

Starting now from eqs. (1) and (3), in the following we assume equidistributed $P(\text{states}) = \text{const}$. We set $t = 0$ at the actual / starting time for the risk estimation and $\delta t = \text{const}$. In addition, we model the damage deterministically, assuming that if an event happens/does not happen, the damage occurs/does not occur. This leads to the risk model as a function of predicted time s,

$$\text{risk}_s(s) = \text{damage}(\text{states}(s))\, P_E(s; 0, \delta t) \tag{11}$$

where $\text{damage}(s) = f(\text{states}(s))$ is an empirical damage model as a function of the predicted states at future time s, so that $\text{damage}(s)$ quantifies the severity of the occurring damage if the event (here: the accident) actually happens.

For the collision risk, as a simple approximation we consider a 2D inelastic collision model (more accurate damage models can be applied here in similar ways), so that

$$\text{damage}(s) \sim \frac{1}{2} \frac{m_1 m_2}{m_1 + m_2} [\hat{v}_1(s) - \hat{v}_2(s)]^2, \tag{12}$$

where m_1, m_2 are the masses and $\hat{v}$,$\hat{v}_2$ the velocity components of the entities involved in the collision risk estimation along the line of sight between the two entities.

For the risk in a curve, we model the damage based on kinetic energy $\text{damage}(s) \sim \frac{1}{2} m_1 |v_1|^2$.

Since we focus on the future risk evaluation of the ego car along selected predicted spatio-temporal trajectories, instead of using a time-dependent risk function $\text{risk}_s(s)$, we can also calculate the risk as a function of the longitudinally traveled distance along the ego-car trajectory (see [2]), so that we will exchangeably also write $\text{risk}(x_e)$ ($\text{risk}_s(s) := \text{risk}(x_e(s))$), where x_e is the future longitudinal path of the ego car.

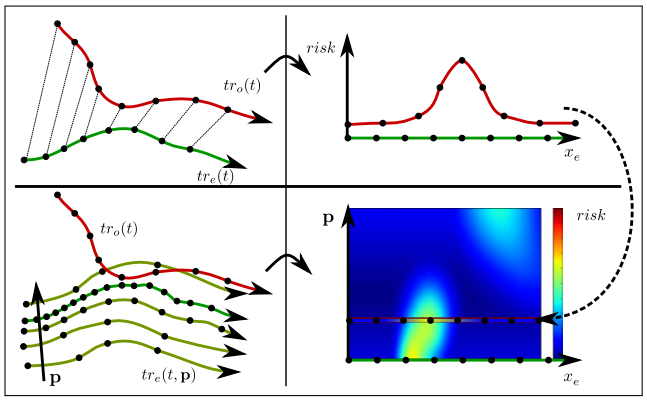


Fig. 1: Predictive Risk Map - Top: evaluation of risk (top right) based on predicted ego car (green) and other car (red) trajectories (top, left) - Bottom: Generation of predictive risk map (bottom, right) based on risk evaluation of a variation of ego car trajectories and other car trajectory (bottom, left).

III. Predictive Risk Maps

In [2] a model for continuous predictive risk, based on heuristic risk indicators such as "time to closest encounter" (TCE) and "distance of closest encounter" (DCE), is used to generate so called "predictive risk maps" .

In this paper we use a more general risk model (11) to evaluate the ego cars' future risk based on the predicted trajectories of the ego car and the other cars, tr_e and tr_o, as shown in figure 1 (top).

For simplicity, we use a constant velocity motion model as e.g. also used in [15], [2] for the prediction of future ego and other trajectories,

$$\begin{aligned} \text{tr}_e &= \{x_e(s), v_e(s), s \,|\, s = 0, ..., S\}\ , \\ \text{tr}_o &= \{x_o(s), v_o(s), s \,|\, s = 0, ..., S\}\ , \end{aligned} \tag{13}$$

where S is the prediction time horizon.

The risk function used in [2] based on risk indicators, such as TCE and DCE, can be seen as a special case of the more general risk function introduced in [1], which we will use in the following.

If we now build a variation of ego car trajectories using parameters p, and evaluate the risk according to section II for each ego-car trajectory alternative together with the predicted other cars' trajectory we arrive at a predictive risk map $\text{Risk}_p(x_e, p)$ in the (x_e, p)-plane, as shown in figure 1. In the following we use the ego velocity as a variation parameter $p = v_e$ and get a predictive risk map in the (x_e, v_e)-plane as $\text{Risk}(x_e, v_e)$ The predictive risk map indicates how risky a certain behavior (in our example: a trajectory with a certain longitudinal velocity profile) will be in the future. Predictive risk maps can then be employed to plan future behavior/future velocity profiles, in a way that minimizes risk and maximizes utility. In general the planned velocity profile varies only smoothly and thus does not differ strongly from the velocity profiles used to generate the risk map. In this case the timing error and its influence on the risk map is sufficiently small. Additionally predictive risk maps are

highly beneficial to illustrate our approach, but can also be easily exchanged by any other risk evaluation scheme.

IV. Behavior Planning using RRT*

Once we have evaluated the risk of possible behavior alternatives and composed them into a predictive risk map the target is to plan the best future behavior minimizing risk and maximizing utility. Here we use a globally optimal sampling based approach called RRT* to obtain the best possible velocity profile through the risk map.

In [2] a gradient descent behavior selection approach was introduced to gather a risk minimizing future trajectory. The drawback of such an approach is the lack of global optimality, which means that in certain constellations the gradient descent approach might run into local minima.

As an example let us have a look at a simple turning behavior at intersections. Here the velocity has to be reduced in order to reduce the risk resulting from a high centrifugal acceleration in the curve for high velocities. At the same time the velocity could be increased in order to pass with reduced risk in front of a crossing car. This can end up in a constellation where either the risk of high centrifugal acceleration or the collision risk can not be kept sufficiently small.

In those cases a globally optimal planner is necessary to find a satisfying plan for future behavior. In the example, slowing down to let the crossing car pass and then performing the turning with reduced velocity, would be a plan for future behavior that considers both risks adequately.

A. General Cost Function

The target of the behavior planning stage is to find a trajectory/velocity profile that minimizes risk and maximizes utility at the same time. In order to accomplish this we combine the predictive risk map with an utility cost function, to arrive at a differential cost

$$\mathrm{DCost}(x_e, v_e) = \mathrm{Risk}(x_e, v_e) + \mathrm{TC}(x_e, v_e) \ , \tag{14}$$

with the travel cost TC. The travel cost can be used to describe soft constraints and optimization criteria such as time and smoothness of travel. As a simple travel cost function, here we set $\mathrm{TC}(v_e, x_e)$ to penalize deviations from a desired travel velocity $v_{e,des}$,

$$\mathrm{TC}(v_e, x_e) = \mathrm{TC}_0 + m\,|v_{e,des} - v_e| \ , \tag{15}$$

with a slew rate m and the minimal travel cost at the desired velocity TC_0. In this way, we force the system to move away from the (usually) lower risk solutions for zero velocity, arriving at solutions that tradeoff travel costs and overall risk.

The solutions that we search for planning the ego-vehicle behavior should minimize both risk and travel cost, so that they should minimize the integrated cost along their trajectory,

$$\mathrm{Cost}(x_e, v_e) = \int_0^{x_e} \mathrm{DCost}(x, v_e)\mathrm{d}x \ . \tag{16}$$

In the following, we will present a sampling based approach to find these solutions.

Algorithm 1 RRT

```
1: procedure RRT
2:     G.init(z_init)
3:     while i < N do
4:         z_rand ← Sample(i);
5:         G ← Extend(G, z_rand);
6:         i ← i + 1;
7:     return CheapestTrajectory(G, target_region)
```

Algorithm 2 Extend RRT

```
1: procedure EXTEND(G,z)
2:     z_nearest ← Nearest(G, z);
3:     z_new ← Steer(z_nearest, z);
4:     G.add_vertex(z_new);
5:     G.add_edge(z_nearest, z_new);
6:     return G
```

B. Rapidly exploring Random Trees (RRTs)

The rapidly exploring random tree (RRT) algorithm was first introduced in [11]. It is an efficient algorithm to search non-convex spaces constructing a space-covering tree by randomly sampling and forward simulation using the systems' dynamic model and kinematic constraints. Beginning from an initial state as a root vertex the RRT constructs open-loop trajectories for any kind of non-linear systems with state constraints. The complexity of the dynamic model with kinematic constraints is not a restriction for the used RRT algorithm. Similar to [16] we use longitudinal double integrator dynamics with input and state constraints (maximal acceleration, deceleration a_{max} and maximal velocity v_{max}),

$$\dot{x_e} = v_e, \quad \dot{v_e} = a_e, \tag{17}$$

with $|v_e| \leq v_{max}$ and $|a_e| \leq a_{max}$.

The general RRT algorithm is meant to rapidly cover the configuration space and is not directly intended to be used as a planning algorithm. By defining a target region the RRT can be used for planning purposes, because every branch of the constructed tree reaching the target region defines a solution trajectory for the system from the starting region to the target region. The basic procedure is shown in algorithm 1, where G is the RRT tree containing *vertices* (here in the (x_e, v_e)-plane) and *edges*. The main part of the algorithm is the **Extend** procedure in algorithm 2, which defines how the tree is extended towards the sampled vertex z_{rand}.

A so-called **Steer** function is used to connect the nearest vertex $z_{nearest}$ of the RRT tree with the next randomly sampled vertex z_{new} by internal forward simulation using the dynamic model of the system. In [16] a time optimal controller is used. The resulting trajectory of the RRT would then be a sequence of maximal acceleration and deceleration, which is not suitable for our purposes. Thus we use a P controller to steer from a start vertex towards a target vertex,

$$a_e = \beta \Delta v_e, \tag{18}$$

where a_e is the acceleration used as the input signal for the dynamic car model, and Δv_e the velocity difference between the start and the target vertex.

Algorithm 3 Extend RRT*

```
1:  procedure EXTEND(G,z)
2:      z_nearest ← Nearest(G, z);
3:      z_new ← Steer(z_nearest, z);
4:      G.add_vertex(z_new);
5:      z_min ← z_nearest;
6:      Z_nearby ← NearVertices(G, z_new);
7:      for all z_near ∈ Z_nearby do
8:          z_new,temp ← Steer(z_near, z_new);
9:          if State(z_new,temp) = State(z_new) then
10:             if Cost(z_near,temp) < Cost(z_new) then
11:                 z_min ← z_near;
12:     G.add_edge(z_min, z_new);
13:     for all z_near ∈ Z_nearby \ {z_min} do
14:         z_near,temp ← Steer(z_new, z_near);
15:         if State(z_near,temp) = State(z_near) then
16:             if Cost(z_near) > Cost(z_near,temp) then
17:                 z_parent ← Parent(z_near,temp);
18:                 G.remove_edge(z_parent, z_near);
19:                 G.add_edge(z_new, z_near,temp);
20:     return G
```

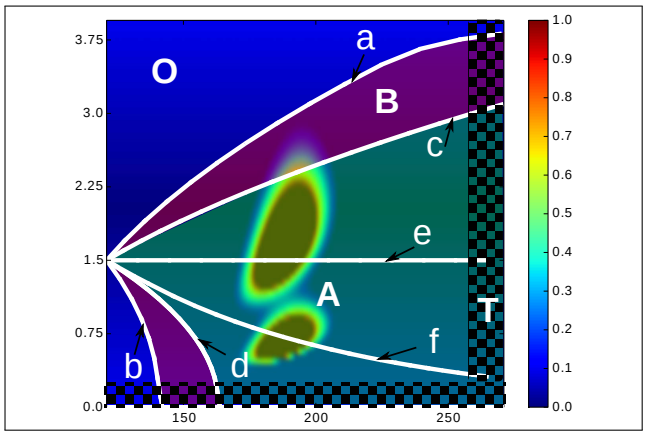


Fig. 2: Predictive Risk Map. Predefined trajectories: (a) emergency acceleration, (b) emergency braking, (c) comfort acceleration, (d) comfort braking, (e) constant velocity, (f) coasting down. Regions: Comfort region (A, green), emergency region (B, red/violet), not reachable region (O, clear blue), target region (T, checkerboard), extended target region for low velocities and stopping (T*, checkerboard).

C. RRT*

As stated in [16] the RRT algorithm used for kinodynamic planning under consideration of a cost function does not guarantee asymptotic optimality. Since we are searching for a trajectory through the predictive risk map to minimize a cost function, we use the RRT* extension [16] of the RRT, which enables asymptotic optimality by re-wiring the constructed tree based on a given cost function. This is done by enhancing the **Extend** function as shown in algorithm 3. From lines 6 to 11 all connections from nearby vertices Z_{nearby} to the new vertex z_{new} are checked and the connection with minimal cost z_{min} is added to the tree G. Additionally from line 13 to 19 all connections from the new vertex z_{new} to nearby vertices Z_{nearby} are checked and if a connection with costs less than the original cost is found, z_{new} is made the new parent of z_{near}. The general RRT* also checks for collision while extending the tree. As collisions are represented in a continuous cost function through risk, we do not include collision checking into the algorithm.

D. Adaptation of RRT* to the given Problem

In order to improve the general RRT/RRT* algorithm for behavior/velocity profile planning we include some prior knowledge as shown in algorithm 4. This is done on one hand by incorporating predefined "typical" trajectories, which should always be considered by the planner as a possible solution, e.g. the safety solution of full braking at a largest desired deceleration. On the other hand we include a bias in the sampling procedure towards low risk areas and additionally adapt the target region of the planner to also allow "stop and wait" as a solution (i.e., regions with zero or near-zero velocities).

1) Predefined Trajectories: As the RRT/RRT* algorithm is based on random sampling and internal forward simulation we can not ensure to always find a suitable trajectory / velocity profile. However some trajectories, such as full braking, should always be considered in the planning process. Thus we ensure the presence of such trajectories by adding them to the initial tree before building the entire tree, as indicated in algorithm 4. This is done using the initial state and the dynamic model by internal forward simulation. There are two categories of predefined trajectories, emergency trajectories and comfort trajectories. We usually plan in the comfort region and use $a_{max} = a_{max,comfort}$ for the dynamic model used in forward simulation. But as we want to allow certain emergency trajectories we set $a_{max} = a_{max,total}$, with $a_{max,total} > a_{max,comfort}$, to allow the full solution space of the dynamic system for those trajectories. The predefined trajectories are shown in figure 2.

a) Emergency Braking and Emergency Acceleration: The emergency braking and emergency acceleration trajectories enable the planner to always find full braking and full acceleration as a solution. Since emergency trajectories are no longer in the comfortable acceleration region we set the costs higher, $\mathrm{DCost}_{emergency}(x_e, v_e) = b \cdot \mathrm{DCost}_{comfort}(x_e, v_e)$ with $b > 1$, in order to avoid unnecessarily using those trajectories.

b) Comfort Braking, Comfort Acceleration, Constant Velocity and Coasting Down: The constant velocity trajectory enables the planner to always consider a straightforward solution. Without the constant velocity trajectory, keeping the velocity constant at the beginning, is not always a solution and high frequent accelerations and deceleration might occur from time to time. A similar is given by the coasting down trajectory which helps the planner to consider a coast-down for slow braking. The comfort braking and comfort acceleration trajectories define the borderline of the comfort zones, as shown in figure 2, and enable the planner to faster cover the whole comfort region with comfortable solution trajectories. As the RRT* rewires its connections according to the cost function a predefined comfort trajectory is usually modified for a better fit, avoiding risky regions.

Algorithm 4 Predictive-Risk-RRT

```
1: procedure PREDICTIVE-RISK-RRT
2:     G.init(z_init);
3:     G.add_init_trajectories(z_init);
4:     while i < N do
5:         z_rand ← RiskBiasedSample(RiskMap);
6:         G ← Extend(G, z_rand);
7:         i ← i + 1;
8:     return CheapestTrajectory(G, target_region)
```

2) Biased Sampling: In each planning phase a predictive risk map is calculated, before running the actual RRT-based planning procedure. The predictive risk map indicates how risky certain risk map areas will be. Knowing that we want to minimize risk, we reduce the probability of sampling in risky areas by introducing a bias to the sampling procedure in a way that areas with low risk are sampled more often than areas with high risk. As a result we gain a better coverage of low risky areas by the tree with the same computational costs. Additionally we sample more often in the target region, which in general speeds up the solution finding.

3) Target Region: In general the target of the algorithm is to find a trajectory from the initial state (on the left of the risk map) through the risk map to the furthest point on the future path (on the right of the risk map) with minimal cost. But if we consider the case of a crowded intersection or a red traffic light, where the ego-vehicle has to stop, there is no direct trajectory through the predictive risk map with sufficiently low costs. The favored solution in terms of risk and utility would be to stop and wait until the environment changes, and then enable a trajectory further along the future path. Thus the target area, as shown in figure 2, is the complete right area of the risk map representing the furthest predicted point along the longitudinal path and the complete area at the bottom of the risk map, representing all possible stop location along the future path. This can be extended further to include preferred stopping zones, etc.

4) Extended Cost Function: By including the stopping area as a target region we encounter the problem that the traveled way may be shorter if the trajectory is finalized at the stopping area, leading to overall lower costs. To overcome this problem, for all trajectories that reach a target region, we normalize the cost function by the traveled way x_e

$$\text{Cost}_{norm}(x_e, v_e) = \frac{1}{x_e} \int_0^{x_e} [\text{Risk}(x, v_e) + \text{TC}(x, v_e)] \text{d}x. \quad (19)$$

V. Applications of Predictive Risk Evaluation and Behavior Planning

Based on the predicted future dynamics of scene entities, expressed in terms of spatio-temporal trajectories, we can now predict the future cost (comprising risk and other additional travel costs) for certain behavior alternatives (here shown for ego car velocity profiles). By generating predictive risk maps we enable our system to plan a cost-optimal future behavior using a sampling-based globally optimal planning algorithm. As we act in a dynamically changing environment we have to re-evaluate and re-plan from time to time. In order to validate the approach we apply our formalism to a set of different scenarios, including inner-city and highway scenarios.

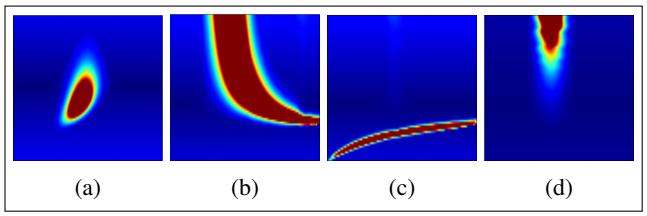


Fig. 3: Basic Risk Shapes: (a) Ego-car crossing the path of other car, (b) following another car, (c) driving in front of another car, (d) passing through a narrow curve. The risk maps in figure 4, 5, 6 and 7 can be interpreted using those basic risk shapes.

A. Basic Risk Shapes

From the calculation of the predictive risk maps, we noticed that some basic risk patterns in form of characteristic shapes reappear frequently in different complex scenarios. Depending on the real constellation, these shapes are usually distorted. First, if another car is predicted to cross the predicted ego car path an ellipsoidal risk spot, as shown in figure 3a, will occur on the risk map with its peak approximately at the ego car velocity and path position where the cars would actually crash. Second, the risk pattern of a car driving in front of the ego car has the shape shown in figure 3b. High ego car velocities result in high risk, since this would lead to a collision. If the ego car velocity decreases towards the velocity of the leading car the point of collision will be further away in the future and once the ego car velocity is below the velocity of the leading car there is no predicted collision anymore and the risk drops to zero. Third, if a car is approaching very fast from behind, there is a similar risk spot as for a leading car. But this time the high risky area is at velocities lower than the other cars' velocity, as shown in figure 3c. And finally the general risk shape for driving through a curve is shown in figure 3d. Here we have high risk for high velocities at the curve locations with highest curvature, since the risk is caused by centrifugal forces.

B. Risk Aversive Curve Driving

In the scenario shown in figure 4, the ego car drives along a very curvy road, which consists basically of four main curves with different radii. As described in the previous section, the risk for driving through a curve is based on the predicted radial acceleration. Each curvy segment generates risky areas raising at a certain velocity. As shown in figure 4, the behavior planner chose a velocity profile that reduces risk in all of the upcoming curves in order to drive safely, but still efficiently along the road. Main risk sources are marked by circled numbers. Each risk map has a horizon range up to 150m.

C. Risk Aversive Intersection Behavior

Next we apply our approach to the scenario shown in figure 5, where the ego car (yellow trajectory) is intended

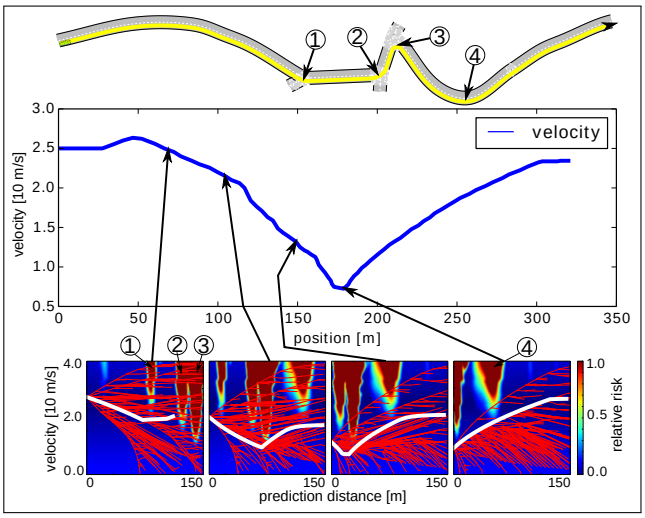


Fig. 4: Curve Driving: general setting (top) with recorded velocity profile (middle) and risk maps at different planning points in time (bottom), including the planned future trajectories (white).

to drive safely straight over an intersection with multiple other crossing traffic participants. The predictive risk map shows a risk peak for each traffic participant and the behavior planner constructs a velocity profile through the risk map while minimizing risk and maximizing utility. It can be seen in the velocity profile that the ego car slows down to let the first car pass and then speeds up to pass between the cars, as there is enough space to pass safely.

D. Risk Aversive Motorway Access Behavior

In this scenario the ego car is approaching the access of a motorway. Each approaching car on the highway generates a risk sport in the predictive risk map. The behavior planner determines if the gap between the first and the second car is sufficiently large for a low-risk motorway access. As shown

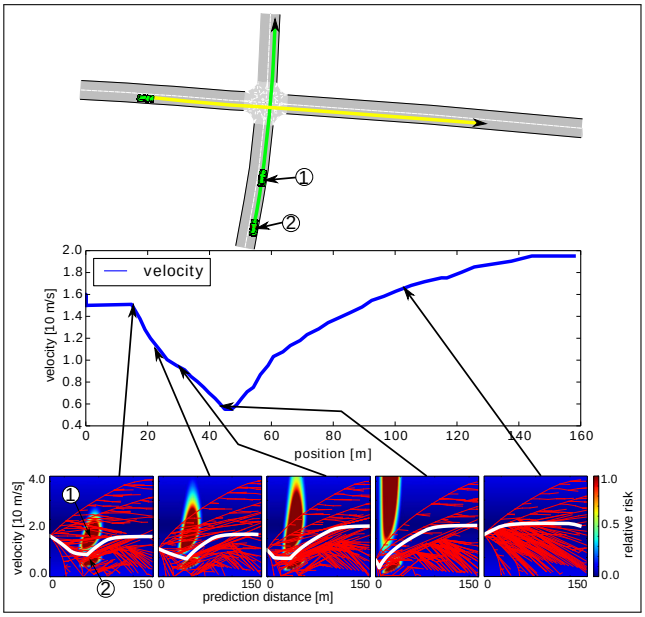


Fig. 5: Basic Intersection Behavior: 2 other cars (green trajectories) as risk sources (1,2).

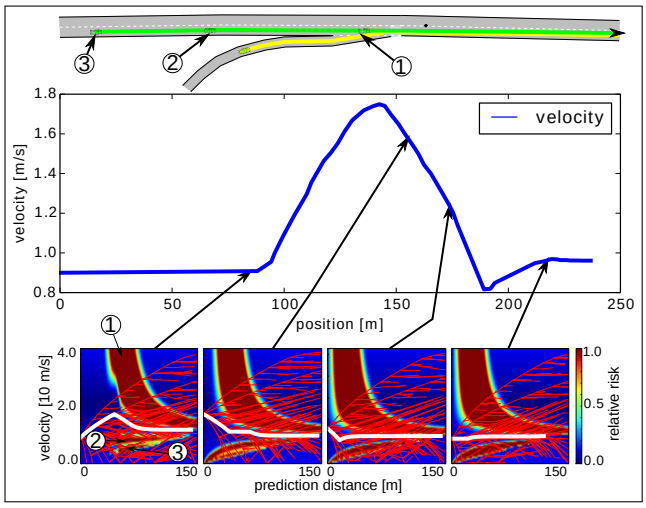


Fig. 6: Highway Accessing.

in figure 6, the resulting velocity profile is speeding up to reach that gap, then slowing down to the same velocity of the other cars on the highway to follow the general traffic flow, even though the desired velocity target is much higher.

E. Risk Aversive Turning Behavior for Complex Intersections with Multiple other Traffic Participants

The scenario from figure 7 shows a crossing, where the ego car is planning to turn left. There are four other traffic participants approaching the crossing. One car approaching from the opposite direction planning to turn right onto the same lane as targeted by the ego car and three cars approaching from the right, one driving straight and two turning left. Each of those cars generate a risk spot in the risk map. The turning process combines intersection driving with curve driving, thus a further risk spot for high velocities is caused by the curve. The behavior planner determines the velocity profile shown in figure 7, which means stopping at the intersection to let the other cars cross, then driving with reduced velocity through the curve, and finally speeding up again to reach the desired travel velocity.

VI. Outlook

The future risk estimation for the ego car depends mainly on the choice of the correct prediction models for the other traffic participants. As a simple approximation, here we assumed that they continue traveling with constant velocity, an assumption which is of course violated as soon as the other cars would have to give way at intersections, or decelerate to turn around curves with low speed.

Therefore, in a next extension of the model, we intend to incorporate simplified risk evaluation also for the driving models of other traffic participants, which would give us a much more realistic and long-time stable trajectory prediction and as a result also better planning capabilities.

Furthermore, the current model assumes that the ego car knows the route that the other traffic participants will take. However, using only sensor measurements available at the ego-car, this is difficult to assess with certainty, e.g. it is difficult to estimate in advance if another traffic participant

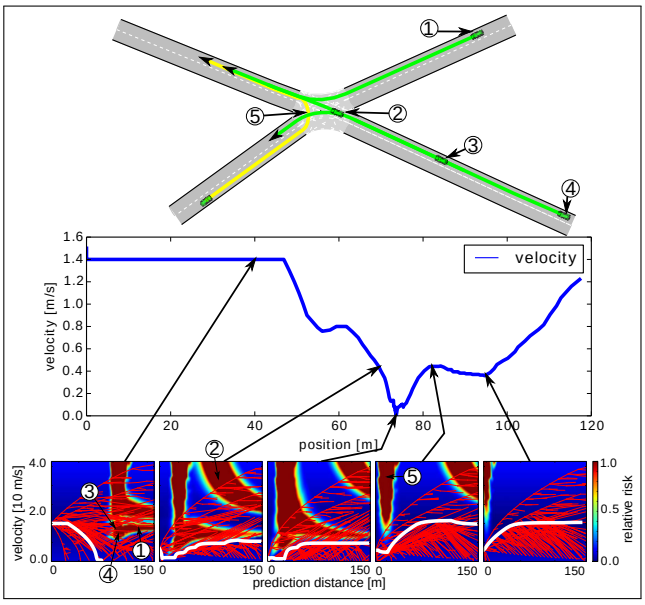


Fig. 7: Complex Turning Behavior: other cars as risk sources (1-4), curve as risk source (5).

will continue straight or turn at an intersection. For tackling this problem we plan to integrate an intention classification step in order to cope with multiple uncertain behavior alternatives of the other traffic participants.

VII. Conclusion

This paper focused on the problem of risk-based behavior evaluation in dynamic scenarios. First a continuous, probabilistic risk assessment is described, which is then used to generate so-called predictive risk maps. Those risk maps are then used to plan future behavior. Here an RRT* algorithm is adapted to the presented problem and used to generate future velocity profiles, which minimize risk and maximize utility.

The modeling of continuous probabilistic risk is based on the expected damage given predicted spatiotemporal trajectories for the ego car and other traffic participants. The expected damage is then modeled using a so called event rate combined with a survival function, which takes into account, that trajectories passing high risk spots are discarded during the search of good and safe trajectory candidates.

The probabilistic model for future risk can be applied to various different types of risk by using different types of event rates. Here we modeled two typical risks encountered in complex scenarios: The car-2-car collision risk (considering arbitrary many other traffic participants) and the risk caused by high centrifugal forces on curvy roads.

With the risk model we built a variation of ego car trajectories, evaluated the risk for each variation and combined them into predictive risk maps, which indicated how risky a certain behavior will be in the future.

Combining the predictive risk map with a utility measure we used a sampling based globally optimal planner (RRT*) to find cost-optimal velocity profiles through the future risk landscape. By integrating a-priori knowledge in form of typical trajectories into the algorithm we increased both the efficiency and the quality of the planning results.

Finally we applied our approach to different scenarios such as inner-city intersection, curve and turning scenarios interacting with other traffic participants and an outer-city highway accessing scenario. Here we could show the high generality of our approach. Even for complex scenarios, where a gradient-descent-like approach would encounter problems of local minima, the RRT* behavior planner in combination with the general predictive risk estimation scheme is able to find safe and still cost efficient velocity profiles.

References

[1] J. Eggert, "Predictive risk estimation for intelligent adas functions," in *Intelligent Transportation Systems-(ITSC), 2014 17th International IEEE Conference on*. IEEE, 2014.

[2] F. Damerow and J. Eggert, "Predictive risk maps," in *Intelligent Transportation Systems-(ITSC), 2014 17th International IEEE Conference on*. IEEE, 2014.

[3] J. Leonard, J. How, S. Teller, M. Berger, S. Campbell, G. Fiore, L. Fletcher, E. Frazzoli, A. Huang, S. Karaman *et al.*, "A perception-driven autonomous urban vehicle," *Journal of Field Robotics*, vol. 25, no. 10, pp. 727–774, 2008.

[4] J. Hillenbrand, A. M. Spieker, and K. Kroschel, "A multilevel collision mitigation approachits situation assessment, decision making, and performance tradeoffs," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 4, pp. 528–540, 2006.

[5] A. Polychronopoulos, M. Tsogas, A. Amditis, U. Scheunert, L. Andreone, and F. Tango, "Dynamic situation and threat assessment for collision warning systems: The euclide approach," in *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, 2004, pp. 636–641.

[6] R. Karlsson, J. Jansson, and F. Gustafsson, "Model-based statistical tracking and decision making for collision avoidance application," in *American Control Conference, 2004. Proceedings of the 2004*, vol. 4. IEEE, 2004, pp. 3435–3440.

[7] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 334–341.

[8] E. Käfer, "Situationsklassifikation und Bewegungsprognose in Verkehrssituationen mit mehreren Fahrzeugen," 2013.

[9] C. Braeuchle, F. Flehmig, W. Rosenstiel, and T. Kropf, "Maneuver decision for active predestrian protection under uncertainty," in *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*. IEEE, 2013, pp. 646–651.

[10] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *The International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.

[11] G. S. Aoude, B. D. Luders, K. K. Lee, D. S. Levine, and J. P. How, "Threat assessment design for driver assistance system at intersections," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 1855–1862.

[12] J. Rios-Martinez, A. Spalanzani, and C. Laugier, "Understanding human interaction for probabilistic autonomous navigation using risk-rrt approach," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 2014–2019.

[13] ISO31000, "Risk management–principles and guidelines," *International Organization for Standardization, Geneva, Switzerland*, 2009.

[14] Wikipedia, "Risk — Wikipedia, the free encyclopedia," 2014, [Online; accessed 12-May-2014]. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Risk&oldid=608166233

[15] R. Miller and Q. Huang, "An adaptive peer-to-peer collision warning system," in *Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th*, vol. 1. IEEE, 2002, pp. 317–321.

[16] S. Karaman and E. Frazzoli, "Optimal kinodynamic motion planning using incremental sampling-based methods," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 7681–7687.