RAP: Risk-Aware Prediction for Robust Planning

Anonymous Author(s)
Affiliation
Address
email

Abstract: Robust planning in interactive scenarios requires predicting the uncertain future to make risk-aware decisions. Unfortunately, due to long-tail safety-critical events, the risk is often under-estimated by finite-sampling approximations of probabilistic motion forecasts. This can lead to overconfident and unsafe robot behavior, even with robust planners. Instead of assuming full prediction coverage that robust planners require, we propose to make prediction itself risk-aware. We introduce a new prediction objective to learn a risk-biased distribution over trajectories, so that risk evaluation simplifies to an expected cost estimation under this biased distribution. This reduces sample complexity during online planning for safe real-time performance. Evaluation results in a didactic simulation environment and on a real-world dataset demonstrate the effectiveness of our approach.

Keywords: Risk Measures, Forecasting, Safety, Human-Robot Interaction

1 Introduction

In safety-critical and interactive control tasks such as autonomous driving, the robot must successfully account for uncertainty of the future motion of surrounding humans. To achieve this, many contemporary approaches to safe human-robot interaction decompose the decision-making pipeline into prediction and planning modules [1–5] for maintainability, debuggability, and interpretability. A prediction module, often learned from data, first produces likely future trajectories of surrounding agents, which are then consumed by a planning module for computing safe robot actions. Recent works [6, 7] further propose to couple prediction with risk-sensitive planning for enhanced safety, wherein the planner computes and minimizes a risk measure [8] of its planned trajectory based on probabilistic forecasts of human motion from the data-driven predictor. A risk measure is a functional that maps a cost distribution to a deterministic real number, which lies between the expected cost and the worst-case cost [9].

Although combining data-driven forecasting and risk-sensitive planning has been shown to be effective, there exist several limitations to this approach. First, accurate risk evaluation of candidate robot plans remains challenging, due to inaccurate characterization of uncertainty in human behavior [10] and finite-sampling from the predictor. Although some existing methods promoting diversity of prediction (e.g., [11, 12]) may alleviate this issue, they are not explicitly designed for reliable risk estimation needed for robust planning. Second, endowing an existing planner with risk-sensitivity often requires non-trivial modifications to its internal optimization algorithm, since the process of optimization can differ significantly between risk-neutral and risk-sensitive planning [13–15]. This modification can be problematic, if, for example, an autonomy stack already has a dedicated and complex (risk-neutral) planner in use and cannot easily modify its internal optimization algorithms.

To address the above limitations, we propose to consider risk within the *predictor* rather than in the planner. We present a risk-biased trajectory forecasting framework, which provides a general approach to making a generative trajectory forecasting model risk-aware. Our novel method augments a pre-trained generative model with an additional encoding process. This modification changes the output of the prediction so that it purposefully and deliberately over-estimates the probability of

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

dangerous trajectories. This "pessimistic" forecasting model give *distributional robustness* [16, 17] to the planner against potential inaccuracies of the human behavior model.

We achieve the pessimistic risk-biased distribution using a novel prediction loss. This shifts the computational burden of drawing many prediction samples that capture rares events from online deployment to offline prediction training. The planner can still obtain an accurate estimate of the risk measure in real-time during deployment with fewer prediction samples required from the biased distribution. Furthermore, this approach also eliminates the need for modifications to the planner's optimization algorithm. Thus, one can achieve enhanced safety by simply replacing a conventional probabilistic motion forecaster with the proposed risk-biased model, while still using the same existing risk-neutral planner. This capability is intended for use in robotic applications where misestimation of risk could lead to injury, including autonomous vehicles and home robots that must operate safely in close proximity to humans.

2 Related Work

Trajectory forecasting from data. Early trajectory forecasting approaches defined hand-crafted dynamics models [18, 19], and incorporated rules that induce obstacle avoidance behavior [20] or mimic the overall traffic flow [21, 22]. More recently, data-driven, learning-based methods have gained popularity for their ability to better capture the complexity of human behavior [23], and typically use neural networks defining multi-modal trajectory distributions [12, 24–40].

Significant effort is directed toward increasing the coverage, or diversity, of motion forecasting models [11, 12, 35–43] in order to ensure that no critical events are missed. Diversity can be explicitly encouraged using a best-of-many loss [26], by replacing a mean-squared loss with a Huber loss [42], by choosing trajectory samples that maximize the distribution coverage [36], or by setting diverse anchors or target points [38–40]. Another strategy to increase mode coverage takes advantage of the latent distribution of CVAEs [5, 11, 43] or GANs [12]. Cui et al. [5] argue that besides coverage, sample efficiency is also an important factor. The authors trained a road-scene motion forecasting model to produce predictions of other agents that induce diverse reactions from the given robot planner. Similarly, McAllister et al. [44] train a model with a weighted loss giving a low weight to the predictions that do not affect the planner. Huang et al. [29] train a forecasting model that allows a simple optimization procedure to select the safest among a set of plans generated by a planner. While prior work considered task-awareness or planner-awareness, to the best of our knowledge, we are the first to use risk as a proxy to make forecasts more useful for the downstream task.

Subjective probability and prospect theory. Our pessimistic risk-biased prediction can be interpreted as a model of subjective probability (e.g., [45]), which is closely related to risk-awareness [46]. For instance, prospect theory [47] studies how humans make risk-aware decisions and introduces the notion of *probability weighting* [48]. Under this model, the distribution is "warped" so that the probabilities of unlikely events are always over-weighted. Recent robotics literature has used prospect theory to better model risk-awareness in human decision making, for example, in collaborative human-robot manipulation [49] and driver behavior modeling [50].

Prospect theory is a descriptive model of decision making that differs from our application to risk-aware autonomous robots. Moreover, our model only overestimates the probability of events that incur high-cost for the robot, unlike probability weighting that overestimates any unlikely outcome.

Risk-sensitive planning and control. Risk-sensitive planning and control dates back to the 1970s, as exemplified by risk-sensitive Linear-Exponential-Quadratic-Gaussian [51, 52] and risk-sensitive Markov Decision Processes (MDPs) [53]. More recent methods include risk-sensitive nonlinear MPC [6, 54], Q-learning [46, 55], and actor-critic [56, 57] methods, for various types of risk measures. Refer to a recent survey [58] for further details. Unlike those methods in which the policy directly optimizes a risk-measure, we propose to instead bias the prediction so that risk-sensitivity can be achieved by a risk-neutral planner that simply optimizes the expected value of the cost.

3 Background

3.1 Generative Probabilistic Trajectory Forecasting

Let x and y be the past and the future trajectories of an agent, and $Y|_x$ denote the random variable of the future trajectory conditioned on the observed past trajectory x. We would like to fit the distribution of $p(Y|_x)$ given a dataset $\mathcal{D}$ of i.i.d. samples of (x, y) pairs. To fit $p(Y|_x)$, we maximize the likelihood of future trajectories w.r.t. the model parameters θ, ϕ: $\text{maximize}_{\theta,\phi} \prod_{(x,y)\in\mathcal{D}} \mathcal{L}(\theta, \phi; y|_x)$, where $\mathcal{L}(\theta, \phi; y|_x)$ is the likelihood of the sample y knowing x. One method to fit this distribution is to learn a conditional variational auto-encoder, CVAE [59]. We focus on this approach because it produces a structured latent representation. The CVAE conditions its likelihood estimation on a latent random variable $Z|_{x,y}$ with a posterior $q_{\phi_2}(z|_{x,y})$, or $Z|_x$ with an inferred prior $q_{\phi_1}(z|_x)$ used in the joint likelihood $p_\theta(y|_x, z|_x)$. The marginal likelihood of the future trajectory (or "model evidence") is $p_\theta(y|_{x,z})$, and can be rewritten as:

$$\mathcal{L}(\theta, \phi; y|_x) \;=\; \int p_\theta(y|_{x,z})dz \;=\; \int p_\theta(y|_{x,z})\frac{q_{\phi_2}(z|_{x,y})}{q_{\phi_2}(z|_{x,y})}dz \;=\; \mathbb{E}_{q_{\phi_2}(z|_{x,y})}\left[\frac{p_\theta(y|_x, z|_x)}{q_{\phi_2}(z|_{x,y})}\right]. \quad (1)$$

Using Jensen's inequality, the logarithm of (1) is lower bounded by

$$L(\theta, \phi; x, y) \;=\; \mathbb{E}_{q_{\phi_2}(z|_{x,y})}\left[\ln(p_\theta(y|_{x,z}))\right] \;-\; \mathrm{KL}\big(q_{\phi_2}(z|_{x,y})||q_{\phi_1}(z|_x)\big), \quad (2)$$

called the evidence lower bound (ELBO). We model q_ϕ and p_θ using neural networks. The encoders assume a Gaussian prior with independent elements to produce the inferred prior $f_{\phi_1} : (x) \to (\mu|_x, \mathrm{diag}(\Sigma|_x))$, and the posterior $f_{\phi_2} : (x, y) \to (\mu_{x,y}, \mathrm{diag}(\Sigma|_{x,y}))$. The decoder makes the forecast $g_\theta : (x, z) \to y$. Every term in (2) can be either computed or estimated with Monte-Carlo sampling as established in [59, 60].

3.2 Risk Measures

A risk measure is defined as a functional that maps a cost distribution to a real number. In other words, given a random cost variable C with distribution p, a risk measure of p yields a deterministic number r called the risk. In practice, we often consider a class of risk measures that lie between the expected value $\mathbb{E}[C]$ and the highest value $\sup(C)$. The former corresponds to the risk-neutral evaluation of C, while the latter gives the worst-case assessment. Such risk measures often take a user-specified risk-sensitivity level $\sigma \in \mathbb{R}$ as an additional argument, which determines where the risk value r is positioned between $\mathbb{E}[C]$ and $\sup(C)$. Formally, let us define a risk measure as $\mathcal{R}_p : (C, \sigma) \to r \in [\mathbb{E}[C], \sup(C)]$, where the subscript p indicates that the risk is evaluated under the distribution p. Examples of such risk measures include entropic risk [52]: $\mathcal{R}_p^{\text{entropic}}(C, \sigma) = \frac{1}{\sigma}\log\mathbb{E}_p[\exp(\sigma C)]$ as well as CVaR [61]:

$$\mathcal{R}_p^{\text{CVaR}}(C, \sigma) \;=\; \inf_{t\in\mathbb{R}}\left\{t + \frac{1}{1-\sigma}\mathbb{E}_p\left[\max(0, C - t)\right]\right\}. \quad (3)$$

The rest of the paper assumes CVaR (3) as the underlying risk measure, but note that the proposed approach is not necessarily bound to this particular choice. For CVaR, the risk value r given risk-sensitivity level $\sigma \in (0, 1)$ can be interpreted as the expected value of the right $(1 - \sigma)$-tail of the cost distribution [62]. Thus, $\mathcal{R}_p(C, \sigma)$ tends to $\mathbb{E}_p[C]$ as $\sigma \to 0$ and to $\sup(C)$ as $\sigma \to 1$.

Another intriguing property of CVaR is its fundamental relation to distributional robustness. It is known that CVaR belongs to a class of risk measures called *coherent measures of risk* [63]. This class of risk measures has the following dual characterization ([63], Theorem 4a):

$$\mathcal{R}_p(C, \sigma) \;=\; \sup_{q\in\mathcal{Q}}\mathbb{E}_q[C], \quad (4)$$

where $\mathcal{Q}$ is a uniquely-determined, non-empty and closed convex subset of the set of all density functions. This suggests that CVaR is equivalent to a worst-case expectation of the cost C when the underlying probability distribution q is chosen adversarially from $\mathcal{Q}$. Therefore, an autonomous robot optimizing CVaR (or coherent measures of risk in general) obtains distributional robustness, in that the objective accounts for robustness to potential inaccuracies in the underlying probabilistic model. In this context, the set $\mathcal{Q}$ is often referred to as an *ambiguity set* in the literature [16, 64].

4 Problem Formulation

Suppose that a robot incurs cost C under a planned policy π or trajectory. This cost is given by a cost function J^π such that $C = J^\pi(Y)$ with Y being the human future trajectory random variable, which the robot predicts probabilistically. We assume that J^π is known and differentiable in y for each π. In an autonomous driving scenario, for example, one can design such a cost function so that $J^\pi(y)$ is high when the robot collides into the particular trajectory $Y = y$ of a human pedestrian. Details about the cost function that we use in this work are given in supplementary material E.

We begin with a pre-trained generative model that gives a predictive distribution $p(Y|_x) = \int p(Y|_{x,z})p(z)dz$ through an inferred latent distribution $p(Z|_x)$ or a prior latent distribution $p(Z)$. This latent is mapped to the trajectory space by a generator or decoder $y = g(z, x)$. We can use a generative model presented in Section 3.1 to learn the decoder (and optionally an encoder for the inference latent distribution $p(Z|_x)$). We omit the subscripts representing the network parameters since they are assumed pre-trained and fixed throughout this section. Under this (unbiased) model, the risk is given by $r = \mathcal{R}_p\left(J^\pi(g(Z, x)), \sigma\right)$ using the risk measure introduced in Section 3.2.

Given the unbiased model and the risk measure, we are interested in finding another distribution $q_\psi(Z)$ in the latent space with learnable parameters ψ, under which simply taking the risk-neutral expectation of the cost will yield the same risk value as given above. This can be achieved by enforcing the following equality constraint on this "biased" distribution $q_\psi(Z)$:

$$\mathbb{E}_{q_\psi}\left[J^\pi(g(Z, x))\right] \;=\; \mathcal{R}_p\left(J^\pi(g(Z, x)), \sigma\right). \tag{5}$$

We show that such a distribution exists in Section A.1 of the supplementary material. Comparing both sides in (5), we note that such q should be dependent on the risk-sensitivity level σ. We propose to optimize the parameters ψ of the risk-biased distribution $q_\psi(Z|_\sigma)$. In general, many distributions q can satisfy (5). We propose to pick a particular q that additionally minimizes the KL divergence from the prior p, to prevent the biased distribution from becoming too different from the original unbiased distribution. This leads to the following constrained optimization problem:

$$\underset{\psi}{\text{minimize}} \quad \mathrm{KL}\left(q_\psi(Z|_\sigma) \| p(Z)\right) \quad \text{subject to } \mathbb{E}_{q_\psi}\left[J^\pi(g(Z, x))\right] \;=\; \mathcal{R}_p\left(J^\pi(g(Z, x)), \sigma\right). \tag{6}$$

We show in the supplementary material A.2 that, under the assumption that the decoder g is volume preserving [65], the solution to the optimization problem (6) is unique. Volume preservation is a strong assumption that we do not attempt to meet but this uniqueness property is a good indication that our problem is well posed.

Connection to distributional robustness. When a coherent measure of risk is chosen as the underlying risk measure (such as CVaR), the right-hand side of (5) is always equivalent to a worst-case distribution q chosen out of an ambiguity set $\mathcal{Q}$ (4). In general, it is difficult to verify if the optimal distribution q_{ψ^*} is in $\mathcal{Q}$, since the specifics of $\mathcal{Q}$ depend on the choice of the risk measure as well as the risk-sensitivity level σ. Nevertheless, it holds true that any feasible distribution q_ψ for (6) yields the same worst-case expected cost as the most adversarial distribution from $\mathcal{Q}$. Therefore, a planner relying on q_ψ instead of p will possess distributional robustness.

5 Implementation Details

5.1 Learning to forecast

In section 6, we perform experiments that rely on specific implementations of trajectory forecasting models. We use two different architectures: the first is a small MLP-based model used in our proof-of-concept experiments (Sections 6.1 and 6.2), and the second resembles state-of-the-art forecasting architectures used in challenging, real-world scenarios (Section 6.3). As described in Section 4, our method relies on a latent space to bias the forecasts. We use CVAEs in this work, however, our method can be applied more broadly to any forecasting model that conditions on a latent sample.

Algorithm 1 Proposed Risk-Biasing Loss Estimation

INPUT: Trajectory $(x, y) \sim \mathcal{D}$, risk level $\sigma \sim p(\sigma)$, KL-loss weight β, risk weight α, robot motion y_{robot}

for $k \in \{1, \dots, K_1\}$ **do**
  Sample latent $z_k|_x \sim \mathcal{N}(\mu|_x, \Sigma|_x)$ with prior parameters $(\mu|_x, \Sigma|_x) = f_{\phi_1}(x)$
  Decode risk-neutral predictions $y_k = g_\theta(x, z_k|_x)$
Compute risk r using $\{y_1, \dots y_{K_1}\}$ and $J^{y_{\text{robot}}}$ with Monte Carlo estimation (e.g., [67])
for $k \in \{1, \dots, K_2\}$ **do**
  Sample biased latent $\hat{z}_k^{(b)} \sim \mathcal{N}(\mu^{(b)}, \Sigma^{(b)})$ with risk-biased parameters $(\mu^{(b)}, \Sigma^{(b)}) = f_\psi(x, \sigma, y_{\text{robot}})$
  Decode risk-biased predictions $\hat{y}_k = g_\theta(x, \hat{z}_k^{(b)})$
Compute expected cost $\hat{r} = \frac{1}{K_2} \sum_{k=1}^{K_2} J^{y_{\text{robot}}}(\hat{y}_k)$
Compute risk loss $L_{\text{risk}} = \rho(\hat{r} - r)$ and prior loss $L_{\text{prior}} = \text{KL}\left(\mathcal{N}(\mu^{(b)}, \Sigma^{(b)}) || \mathcal{N}(\mu|_x, \Sigma|_x)\right)$

OUTPUT: Loss value $\alpha L_{\text{risk}} + \beta L_{\text{prior}}$ to train ψ (θ and ϕ_1 are fixed)

The small model is composed of two multi-layer perceptron (MLP) encoders and one MLP decoder. The inference encoder takes the past trajectory x and outputs the parameters of a Normal distribution $\mu|_x$ and $\log(\text{diag}(\Sigma|_x))$. The posterior encoder takes the whole past and future trajectory x, y and outputs the parameters of a Normal distribution $\mu|_{x,y}$ and $\log(\text{diag}(\Sigma|_{x,y}))$. Finally, the decoder takes the past trajectory x and a latent sample z and outputs a prediction y. The second architecture is designed to model the interactions with both the other agents and map elements. It takes a similar form as the small model, but with additional context inputs and larger hidden dimensions. The social and map interactions are accounted for with a modified multi-context gating block [40]. Section B of the supplemental provides additional details on these forecasting models.

5.2 Learning to bias towards risk

We propose to solve problem (6) by learning a third neural network encoder to define a biased latent distribution that, in combination with the pre-trained decoder, produces biased forecasts. This biased encoder takes the past trajectory x, a risk-level σ, and the robot future trajectory y_{robot}. It outputs the parameters of a Normal distribution $\mu^{(b)}$ and $\log(\text{diag}(\Sigma^{(b)}))$.

In practice, optimizing a neural network under constraints is challenging. Therefore, we soften the hard constraint (5) by using the penalty method [66], which progressively increases the weight α of the risk-loss during training. We also leverage a user-defined sampling distribution $p(\sigma)$ to sample different risk-sensitivity levels during training, so that the risk estimate remains accurate at any reasonable value of σ at inference time. Finally, we encourage the model to overestimate the risk rather than underestimate it so we scale by the value s and define an asymmetric risk-loss that penalizes linearly the underestimation of the risk and logarithmically its overestimation:

$$\rho(x) \;=\; \begin{cases} s|x|, & \text{if } x \leq 1 \\ \log(sx), & \text{otherwise.} \end{cases} \tag{7}$$

We obtain the following loss function with α and β controlling the relative importance of the losses:

$$\mathcal{L}(\psi) \;=\; \mathbb{E}_{\sigma \sim p(\sigma)} \left[\alpha\, \rho\big(\mathbb{E}_{q_\psi}\left[J^\pi(g(Z, x))\right] - \mathcal{R}_p\left(J^\pi(g(Z, x)), \sigma\right)\big) + \beta\, \text{KL}\left(q_\psi(Z|_{\sigma,x}) \| p(Z|_x)\right) \right].$$

The expected values and the risk measure are approximated by Monte Carlo sampling. For computing CVaR ($\mathcal{R}_p\left(J^\pi(g(Z, x)), \sigma\right)$), we use the estimator proposed by Hong et al. [67]. Consistency and asymptotic normality of this estimator hold under mild assumptions [67].

Algorithm 1 lays out the procedure for training our proposed risk-aware prediction. It relies on a fully trained CVAE with the encoder $f_{\phi_1} : x \to (\mu|_x, \Sigma|_x)$ and decoder $g_\theta : x, z \to y$ that fits the distribution of $Y|_x$ from a dataset. We train a new latent-biasing encoder $f_\psi : x, \sigma, y_{\text{robot}} \to (\mu^{(b)}, \Sigma^{(b)})$ to bias the latent distribution while keeping the rest of the CVAE fixed. The risk-level σ is randomly sampled on $[0, 1]$ during training and chosen by the user at test time.

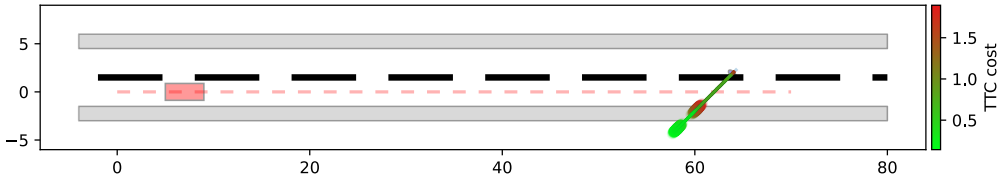
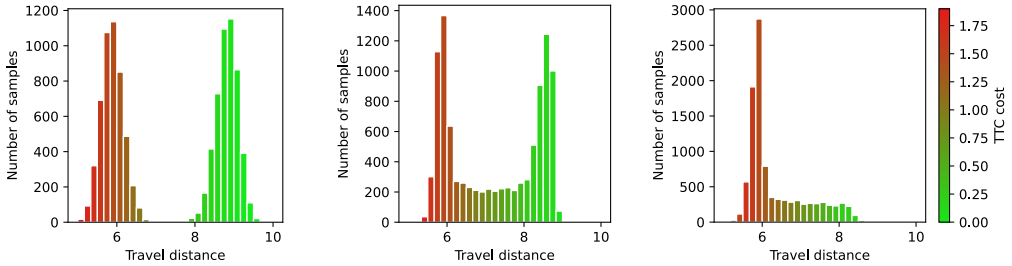


Figure 1: Top-down view of a simulated scene. The robot in red moves left to right down the road as a pedestrian in blue is crossing. The color of the depicted pedestrian trajectory samples indicates their corresponding TTC cost for the robot. The slow mode in red is more costly than the fast mode in green.

6 Experiments

6.1 Biasing forecasts in a didactic scenario

As an initial evaluation of our model, we created the didactic simulation environment in Fig. 1 where a red robot drives at constant speed along a straight road with a stochastic pedestrian. The pedestrian either walks slowly or quickly, yielding a bimodal distribution over their travel distance. We collected a dataset in this environment where the initial position and orientation of the pedestrian is set at random. We used it to train a risk-biased CVAE model according to the method presented in sections 4 and 5. Fig. 2b shows the risk-neutral prediction ($\sigma = 0$) of the pedestrian's travel distance in a specific scene. As can be seen, the model captures both of the equally-likely modes. In contrast, in the risk-biased case ($\sigma = 0.95$), the model predicts the slower mode with much greater frequency because, in this scene, if the pedestrian walks slowly it will collide with the robot. If, alternatively, the pedestrian walks quickly, the vehicle will pass behind it safely without collision. In other words, the risk-biased model pessimistically predicts collisions with a greater probability than does the risk-neutral model. With $\sigma = 0.95$, pessimism is so high that the safer mode falls in the tail of the distribution. In supplemental D we explore the latent representation of this model.



(a) Ground truth data distribution. (b) Risk-neutral prediction. (c) Risk-biased prediction $\sigma = 0.95$.

Figure 2: Histograms of the pedestrian travel distances at the end of the 5-second episode in the defined scene. Each bar is colored with the average TTC cost of the bin.

6.2 Planning with a biased prediction

The previous experiment demonstrates the ability of our approach to bias predictions towards dangerous outcomes. The experiment in this section evaluates whether this ability can benefit online planning using a model-based trajectory optimization algorithm. In this setting, predictions are used to evaluate the risk of various candidate robot trajectories, y_{robot}, in order to select the best.

We generated a new dataset in which the robot's initial speed and per-timestep accelerations are sampled randomly, as opposed to the constant velocity model used previously. This variation ensures that generated trajectories are within the training distribution of the predictor. The pedestrian behavior remained the same as the previous experiment. We trained a new risk-biased CVAE model on this diversified dataset. As the TTC cost is a function of both the robot trajectory and the pedestrian trajectory, we additionally pass y_{robot} to the biasing encoder in order to achieve pessimistic forecasting with respect to the particular y_{robot}.

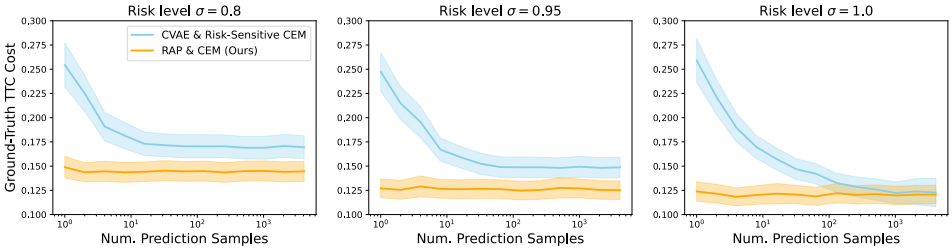


Figure 3: Ground-truth TTC cost of the optimized y^*_{robot}, averaged over 500 episodes (lower the better). Ribbons show 95% confidence intervals of the mean. At each value of σ, our risk-biased predictor (RAP) coupled with CEM consistently achieves low cost regardless of the number of prediction samples that CEM draws online from the predictor. On the other hand, the risk estimation quality of the baseline critically degrades as we decrease the number of samples, leading to overconfident robot behavior that is reflected in higher cost of y^*_{robot}.

The online planner controls the longitudinal acceleration of the robot, which is modeled as a double integrator system. We employed the cross entropy method (CEM) [68, 69] as the underlying optimization algorithm. CEM is a stochastic optimization method that locally optimizes y_{robot} based on a given initial $y^{\text{init}}_{\text{robot}}$. In each episode, CEM first draws risk-biased prediction samples given $y^{\text{init}}_{\text{robot}}$ and observed human motion x, and uses those samples to produce the locally optimal y^*_{robot}. We incorporated a quadratic trajectory tracking cost in the planner's objective so that the robot is encouraged to continue moving at a constant speed when the risk of collision is deemed negligible.

We evaluated the performance of the combined risk-biased predictor and CEM planner across 500 episodes. Fig. 3 shows that CEM using the risk-biased predictor consistently produces y^*_{robot} with low TTC cost values, even when sampling few trajectories. We compared with a baseline approach in which a risk-sensitive version of CEM performs planning using an unbiased CVAE predictor. We obtained this risk-sensitive planner by replacing the Monte Carlo expectation with the CVaR estimator [67]. This baseline is an instance of the conventional risk-sensitive planning with data-driven human motion forecasting [6, 7], which evaluates the risk within the planner rather than in the predictor. Fig. 3 shows significantly higher TTC cost of y^*_{robot} for the baseline when CEM uses fewer than 16 samples. This is because the collision risk is underestimated when using few samples, and thus the planner over-optimistically optimizes trajectory tracking to the detriment of safety.

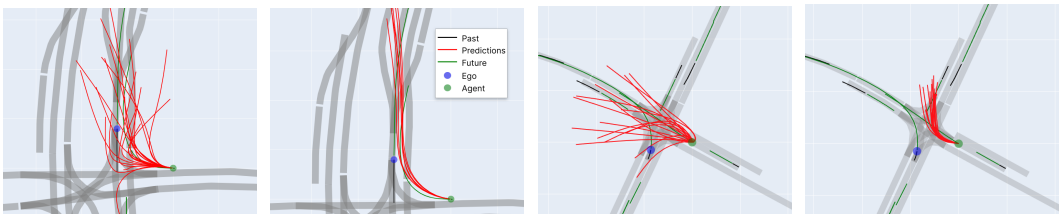
6.3 Application to real-world data

We next evaluated our approach using real-world data from the Waymo open motion dataset [70]. Following a similar approach to [29], we selected the annotated scenarios that cover interesting interactions between two agents. We randomly selected one of the two interacting agent as the ego and the other as the agent to predict, and trained a biased CVAE model as described in Section 5. In this experiment we only condition the biased-encoder on the ego past trajectory, not its future trajectory, in order to avoid ground truth information leakage. This means that the biased-encoder is making an implicit forecast of the ego future, which leads to the failure mode presented in Fig. 4 wherein the wrong implicit forecast leads to an incorrectly-biased distribution.

Table 1: Motion forecasting error and risk estimation error on the Waymo validation set. **minFDE (K)**: minimum final displacement error over K samples, **risk error (K)**: mean value of the signed difference between the average cost of the biased forecasts over K samples and the risk estimation using the unbiased forecasts, **risk |error| (K)**: mean value of the absolute values of the risk estimation error.

Predictive Model	σ	minFDE(16)	FDE (1)	Risk error (4)	Risk |error| (4)
Unbiased CVAE	NA	3.82 $\pm$0.04	13.06 $\pm$0.01	NA	NA
Biased CVAE (RAP)	0	3.81 $\pm$0.04	13.07 $\pm$0.02	0.00 $\pm$0.00	0.12 $\pm$0.00
Biased CVAE (RAP)	0.3	4.32 $\pm$0.05	11.89 $\pm$0.02	0.02 $\pm$0.00	0.13 $\pm$0.00
Biased CVAE (RAP)	0.5	5.32 $\pm$0.06	12.05 $\pm$0.02	0.02 $\pm$0.00	0.16 $\pm$0.00
Biased CVAE (RAP)	0.8	7.78 $\pm$0.07	13.53 $\pm$0.02	0.01 $\pm$0.00	0.26 $\pm$0.00
Biased CVAE (RAP)	0.95	10.13 $\pm$0.09	15.33 $\pm$0.02	0.03 $\pm$0.01	0.43 $\pm$0.00
Biased CVAE (RAP)	1	11.58 $\pm$0.09	16.29 $\pm$0.02	-0.22 $\pm$0.01	0.60 $\pm$0.01

Table 1 shows the results of this experiment. First note the large difference between the minFDE and FDE values of the unbiased model, which illustrates that the predictions are diverse. This is qualitatively supported by Fig. 4, which shows a wide diversity of predicted trajectories. As expected, our biased CVAE model (RAP) with a risk level $\sigma = 0$ shows results that match the ones from the unbiased model. As the risk-level increases, the predictions increasingly differ from those of the unbiased distribution as well as the ground truth trajectory. This is reflected by the larger minFDE and FDE values. At $\sigma = 1$, the biased prediction distribution collapses to the mode that the model estimates to be the most costly, yielding minFDE close to FDE. In Table 1, the risk estimation error is the average difference of the mean *cost* under the biased prediction and the *risk* estimated using a large number of samples from the unbiased prediction. Its mean value shows the bias on risk estimation of the proposed approach while its mean *absolute* value shown in the next column shows the average error that is made in either direction. Up to $\sigma = 0.95$, the risk estimation is nearly unbiased. At $\sigma = 1$, the risk estimation is slightly under-estimated. Section C of the supplement provides additional experimental details and results.



(a) Risk-neutral (b) **Risk-aware**, $\sigma = 0.95$ (c) Risk-neutral (d) **Risk-aware**, $\sigma = 0.95$

Figure 4: Visualization of Waymo scenes and forecasts. The lane centerlines are represented in gray, the past observations in black, the ground truth futures in green, and the 16 forecast samples in red. The ego is in blue and the agent to predict in green. In figure **(b)** the risk-aware forecasts turns towards the ego. In figure **(d)** the risk-aware forecasts would be costly if the ego went straight ahead. This is a failure case where the forecasts are biased towards an expected ego trajectory that did not occur.

7 Limitations

A first limitation of our approach is that the constrained optimization problem (6) is difficult to solve in practice, due to challenges presented in Section 4. The constraint relaxation and the neural network optimization method yield a sub-optimal risk-aware predictor that may still underestimate risk at a particularly high risk-levels ($\sigma > 0.95$). Therefore, our method might be inadequate when an extremely conservative behavior is desired. Second, in the real data application, our risk-aware prediction is not conditioned on a specific robot plan which may lead the forecast to collapse onto a mode that is not the most critical. We plan to address this limitation in future work.

8 Conclusion

This paper proposes a risk-aware trajectory forecasting method for robust planning in human-robot interaction problems. We present a novel framework to learn a pessimistic distribution offline that simplifies online risk evaluation to expected cost estimation. Our experimental results show that this method leads to safe robot plans with reduced sample complexity. We additionally demonstrate the effectiveness of our approach in real-world scenarios with low risk-estimation error and strong qualitative results. In future work, we intend to evaluate our approach in a realistic simulator, and also improve the accuracy of risk estimation on real-world data by conditioning biased prediction on potential ego plans.

References

[1] E. Schmerling, K. Leung, W. Vollprecht, and M. Pavone. Multimodal probabilistic model-based planning for human-robot interaction. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3399–3406. IEEE, 2018.

[2] H. Fan, F. Zhu, C. Liu, L. Zhang, L. Zhuang, D. Li, W. Zhu, J. Hu, H. Li, and Q. Kong. Baidu Apollo EM motion planner. *arXiv preprint arXiv:1807.08048*, 2018.

[3] W. Zeng, W. Luo, S. Suo, A. Sadat, B. Yang, S. Casas, and R. Urtasun. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8660–8669, 2019.

[4] B. Ivanovic, A. Elhafsi, G. Rosman, A. Gaidon, and M. Pavone. MATS: An interpretable trajectory forecasting representation for planning and control. In *Conference on Robot Learning*, pages 2243–2256. PMLR, 2020.

[5] A. Cui, S. Casas, A. Sadat, R. Liao, and R. Urtasun. Lookout: Diverse multi-future prediction and planning for self-driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16107–16116, 2021.

[6] H. Nishimura, B. Ivanovic, A. Gaidon, M. Pavone, and M. Schwager. Risk-sensitive sequential action control with multi-modal human trajectory forecasting for safe crowd-robot interaction. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11205–11212. IEEE, 2020.

[7] R. S. Novin, A. Yazdani, A. Merryweather, and T. Hermans. Risk-aware decision making for service robots to minimize risk of patient falls in hospitals. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3299–3305. IEEE, 2021.

[8] A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[9] A. Majumdar and M. Pavone. How should a robot assess risk? towards an axiomatic theory of risk in robotics. In *Robotics Research*, pages 75–84. Springer, 2020.

[10] R. Cheng, R. M. Murray, and J. W. Burdick. Limits of probabilistic safety guarantees when considering human uncertainty. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3182–3189. IEEE, 2021.

[11] Y. Yuan and K. Kitani. Diverse trajectory forecasting with determinantal point processes. *arXiv preprint arXiv:1907.04967*, 2019.

[12] X. Huang, S. G. McGill, J. A. DeCastro, L. Fletcher, J. J. Leonard, B. C. Williams, and G. Rosman. Diversitygan: Diversity-aware vehicle motion prediction via latent semantic sampling. *IEEE Robotics and Automation Letters*, 5(4):5089–5096, 2020.

[13] A. Shaiju and I. R. Petersen. Formulas for discrete time LQR, LQG, LEQG and minimax LQG optimal control problems. *IFAC Proceedings Volumes*, 41(2):8773–8778, 2008.

[14] N. Bäuerle and J. Ott. Markov decision processes with average-value-at-risk criteria. *Mathematical Methods of Operations Research*, 74(3):361–379, 2011.

[15] Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[16] S. Samuelson and I. Yang. Data-driven distributionally robust control of energy storage to manage wind power fluctuations. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 199–204. IEEE, 2017.

[17] A. Sinha, M. O'Kelly, H. Zheng, R. Mangharam, J. Duchi, and R. Tedrake. Formulazero: Distributionally robust online adaptation via offline population synthesis. In *International Conference on Machine Learning*, pages 8992–9004. PMLR, 2020.

[18] J. Mercat, N. E. Zoghby, G. Sandou, D. Beauvois, and G. P. Gil. Kinematic single vehicle trajectory prediction baselines and applications with the ngsim dataset. *arXiv preprint arXiv:1908.11472*, 2019.

[19] C. Schöller, V. Aravantinos, F. Lay, and A. Knoll. What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robotics and Automation Letters*, 5(2):1696–1703, 2020.

[20] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51 (5):4282, 1995.

[21] M. Treiber, A. Hennecke, and D. Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, 62(2):1805, 2000.

[22] A. Kesting, M. Treiber, and D. Helbing. General lane-changing model mobil for car-following models. *Transportation Research Record*, 1999(1):86–94, 2007.

[23] S. Lefèvre, D. Vasquez, and C. Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH journal*, 1(1):1–14, 2014.

[24] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2255–2264, 2018.

[25] N. Deo and M. M. Trivedi. Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1179–1184. IEEE, 2018.

[26] A. Bhattacharyya, B. Schiele, and M. Fritz. Accurate and diverse sampling of sequences based on a "best of many" sample objective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8485–8493, 2018.

[27] M. Khakzar, A. Rakotonirainy, A. Bond, and S. G. Dehkordi. A dual learning model for vehicle trajectory prediction. *IEEE Access*, 8:21897–21908, 2020.

[28] J. Ngiam, B. Caine, V. Vasudevan, Z. Zhang, H.-T. L. Chiang, J. Ling, R. Roelofs, A. Bewley, C. Liu, A. Venugopal, et al. Scene transformer: A unified multi-task model for behavior prediction and planning. *arXiv e-prints*, pages arXiv–2106, 2021.

[29] X. Huang, G. Rosman, A. Jasour, S. G. McGill, J. J. Leonard, and B. C. Williams. TIP: Task-informed motion prediction for intelligent systems. *arXiv preprint arXiv:2110.08750*, 2021.

[30] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *European Conference on Computer Vision*, pages 683–700. Springer, 2020.

[31] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde. Gohome: Graph-oriented heatmap output for future motion estimation. *arXiv preprint arXiv:2109.01827*, 2021.

[32] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde. Thomas: Trajectory heatmap output with learned multi-agent sampling. *arXiv preprint arXiv:2110.06607*, 2021.

[33] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese. Sophie: An attentive GAN for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019.

[34] J. Mercat, T. Gilles, N. El Zoghby, G. Sandou, D. Beauvois, and G. P. Gil. Multi-head attention for multi-modal joint vehicle motion forecasting. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9638–9644. IEEE, 2020.

[35] N. Rhinehart, K. M. Kitani, and P. Vernaza. R2P2: A reparameterized pushforward policy for diverse, precise generative path forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 772–788, 2018.

[36] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde. Home: Heatmap output for future motion estimation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 500–507. IEEE, 2021.

[37] J. Amirian, J.-B. Hayet, and J. Pettré. Social ways: Learning multi-modal distributions of pedestrian trajectories with gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[38] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, et al. TNT: Target-driven trajectory prediction. *arXiv preprint arXiv:2008.08294*, 2020.

[39] S. Narayanan, R. Moslemi, F. Pittaluga, B. Liu, and M. Chandraker. Divide-and-conquer for lane-aware diverse trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15799–15808, 2021.

[40] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. P. Lam, D. Anguelov, et al. MultiPath++: Efficient information fusion and trajectory aggregation for behavior prediction. *arXiv preprint arXiv:2111.14973*, 2021.

[41] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 336–345, 2017.

[42] S. Casas, C. Gulino, S. Suo, K. Luo, R. Liao, and R. Urtasun. Implicit latent variable model for scene-consistent motion forecasting. In *European Conference on Computer Vision*, pages 624–641. Springer, 2020.

[43] Y. Yuan and K. Kitani. DLow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, pages 346–364. Springer, 2020.

[44] R. McAllister, B. Wulfe, J. Mercat, L. Ellis, S. Levine, and A. Gaidon. Control-aware prediction objectives for autonomous driving. *arXiv preprint arXiv:2204.13319*, 2022.

[45] L. J. Savage. *The foundations of statistics*. Courier Corporation, 1972.

[46] Y. Shen, M. J. Tobia, T. Sommer, and K. Obermayer. Risk-sensitive reinforcement learning. *Neural computation*, 26(7):1298–1328, 2014.

[47] A. Tversky and D. Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, 5(4):297–323, 1992.

[48] N. C. Barberis. Thirty years of prospect theory in economics: A review and assessment. *Journal of Economic Perspectives*, 27(1):173–96, 2013.

[49] M. Kwon, E. Biyik, A. Talati, K. Bhasin, D. P. Losey, and D. Sadigh. When humans aren't optimal: Robots that collaborate with risk-aware humans. In *2020 15th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 43–52. IEEE, 2020.

[50] L. Sun, W. Zhan, Y. Hu, and M. Tomizuka. Interpretable modelling of driving behaviors in interactive driving scenarios based on cumulative prospect theory. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 4329–4335. IEEE, 2019.

[51] D. Jacobson. Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic differential games. *IEEE Transactions on Automatic control*, 18(2): 124–131, 1973.

[52] P. Whittle. Risk-sensitive linear/quadratic/Gaussian control. *Advances in Applied Probability*, 13(4):764–777, 1981.

[53] R. A. Howard and J. E. Matheson. Risk-sensitive Markov decision processes. *Management science*, 18(7):356–369, 1972.

[54] V. Roulet, M. Fazel, S. Srinivasa, and Z. Harchaoui. On the convergence of the iterative linear exponential quadratic gaussian algorithm to stationary points. In *2020 American Control Conference (ACC)*, pages 132–137. IEEE, 2020.

[55] V. S. Borkar. Q-learning for risk-sensitive control. *Mathematics of operations research*, 27(2): 294–311, 2002.

[56] Y. Chow and M. Ghavamzadeh. Algorithms for CVaR optimization in MDPs. *Advances in neural information processing systems*, 27, 2014.

[57] J. Choi, C. Dance, J.-e. Kim, S. Hwang, and K.-s. Park. Risk-conditioned distributional soft actor-critic for risk-sensitive navigation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8337–8344. IEEE, 2021.

[58] Y. Wang and M. P. Chapman. Risk-averse autonomous systems: A brief history and recent developments from the perspective of optimal control. *arXiv preprint arXiv:2109.08947*, 2021.

[59] K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

[60] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[61] G. C. Pflug. Some remarks on the value-at-risk and the conditional value-at-risk. In *Probabilistic constrained optimization*, pages 272–281. Springer, 2000.

[62] A. A. Trindade, S. Uryasev, A. Shapiro, and G. Zrazhevsky. Financial prediction with constrained tail risk. *Journal of Banking & Finance*, 31(11):3524–3538, 2007.

[63] R. T. Rockafellar. Coherent approaches to risk in optimization under uncertainty. In *OR Tools and Applications: Glimpses of Future Technologies*, pages 38–61. Informs, 2007.

[64] B. P. Van Parys, D. Kuhn, P. J. Goulart, and M. Morari. Distributionally robust control of constrained stochastic systems. *IEEE Transactions on Automatic Control*, 61(2):430–442, 2015.

[65] G. MacDonald, A. Godbout, B. Gillcash, and S. Cairns. Volume-preserving neural networks. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9. IEEE, 2021.

[66] M. J. Kochenderfer and T. A. Wheeler. *Algorithms for optimization*. Mit Press, 2019.

[67] L. J. Hong, Z. Hu, and G. Liu. Monte carlo methods for value-at-risk and conditional value-at-risk: a review. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 24(4): 1–37, 2014.

[68] K. Chua, R. Calandra, R. McAllister, and S. Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

[69] A. Nagabandi, K. Konolige, S. Levine, and V. Kumar. Deep dynamics models for learning dexterous manipulation. In *Conference on Robot Learning*, pages 1101–1112. PMLR, 2020.

[70] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The Waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021.