Safe and Non-Conservative Trajectory Planning for Autonomous Driving Handling Unanticipated Behaviors of Traffic Participants

Tommaso Benciolini*, Michael Fink*, Nehir Güzelkaya, Dirk Wollherr, Marion Leibold

Abstract—**Trajectory planning for autonomous driving is challenging because the unknown future motion of traffic participants must be accounted for, yielding large uncertainty. Stochastic Model Predictive Control (SMPC)-based planners provide non-conservative planning, but do not rule out a (small) probability of collision. We propose a control scheme that yields an efficient trajectory based on SMPC when the traffic scenario allows, still avoiding that the vehicle causes collisions with traffic participants if the latter move according to the prediction assumptions. If some traffic participant does not behave as anticipated, no safety guarantee can be given. Then, our approach yields a trajectory which minimizes the probability of collision, using Constraint Violation Probability Minimization techniques. Our algorithm can also be adapted to minimize the anticipated harm caused by a collision. We provide a thorough discussion of the benefits of our novel control scheme and compare it to a previous approach through numerical simulations from the CommonRoad database.**

I. INTRODUCTION

An essential component enabling autonomous driving are motion planning algorithms that handle the interplay with other vehicles, pedestrians, and cyclist, referred to as Dynamic Obstacles (DO) in this work. Trajectory planning algorithms must account for the unknown future behavior of DOs in a safe yet non-conservative, human-like manner.

Machine learning approaches are very popular for their ability to replicate human-like motion using past data [1]–[3]. However, giving formal guarantees of collision avoidance with machine learning methods is challenging.

Model Predictive Control (MPC) is a popular algorithm for trajectory planning [4], which realizes a human-like control strategy [5], [6], while still allowing to provide formal guarantees. In MPC the trajectory of the automated vehicle, here Ego Vehicle (EV), is computed by iteratively solving an Optimal Control Problem (OCP) over fixed-length receding horizons. Desired control objectives are given as a cost function and safety conditions and traffic rules, together with physical limitations of the vehicle, result in a set of constraints. At each sampling time the action optimal sequence of the EV for the full prediction horizon is obtained. The first input of the optimal sequence is applied and the process is repeated when new measurements of the environment are collected, adapting the future trajectory to the changing traffic configuration.

The unknown future trajectory of DOs is a source of uncertainty that must be accounted for. The uncertain future

motion of DOs can be described in terms of worst-case movements underlying a given intention and considering the traffic rules. Then, the EV trajectory is *safe* if it is collision free for all realizations of the uncertain DO future motion within the worst-case assumptions. Furthermore, a probability distribution for movements within the worst-case outcomes can be assumed. The key challenge from a control perspective lies in balancing the need for safety and avoiding overly conservative behaviors typical of robust approaches based on a *robust* worst-case analysis [7]–[11]. Furthermore, robust approaches are only applicable if the assumed uncertainty is bounded and small and come at the price of a significant conservatism in the trajectory planning of the EV, leading to considerable inefficiency in environments with large uncertainty about the future motion of DOs. To reduce conservatism, Stochastic Model Predictive Control (SMPC) [12], [13] has been proposed, reformulating collision avoidance constraints in chance (probabilistic) constraints, that must only be satisfied up to a user-specified probability level. SMPC yields a non-conservative EV trajectory [5], but produces open-loop predictions with non-zero probability of collision that may consequently result in collisions in closed loop. Therefore, SMPC does not provide safety guarantees typical of robust approaches. A scheme to take advantage of the optimistic planning of SMPC while still giving collision avoidance guarantees has been proposed in our previous work [14]. However, the scheme requires the cascaded solution of two optimal control problems at each iterations, which is impractical in real-world applications with small sampling times. Furthermore, the approach relies on assumptions for the worst-case behavior of DOs, without addressing the case in which such assumptions are violated. Differently, contingency MPC [15] has been proposed, in which non-conservative yet safe planning is achieved by computing both a nominal and a safe input sequence that produce the same first-step prediction. Although the cascaded solution of two cascaded optimization problems, computing the two input sequences at once does significantly increase the dimension of the optimization problem. Furthermore, the algorithm does not allow to use the full SMPC trajectory even when it is safe to do so, since the first element of the input sequence is always optimized jointly with the robust trajectory.

In this work we propose a novel control scheme for efficient and safe trajectory planning for automated driving that leverages on the optimistic SMPC planning still yielding a collision-free motion. At first, the SMPC OCP is solved, yielding an optimistic non-conservative prediction for the EV trajectory. Then, the first predicted state of the SMPC

trajectory is used as initial condition for a further planning problem, *robust* with respect to the uncertain DO motion. The solution to this problem is not computed, the problem only serves as certification of safety. If such further OCP admits a solution, the first SMPC input is guaranteed not to lead to unavoidable collisions. Therefore, in this case, the SMPC input is considered safe and is applied. By contrast, if a solution to the further robust control problem does not exist, i.e., the robust control problem is infeasible, the SMPC solution is disregarded, since it could lead to collisions that can not be prevented in the next time steps. If the SMPC solution fails the safety certification, we plan a robust trajectory from the current initial condition, which exists because of the safety certification of the previous iteration. Thus, the EV does not cause collisions with DOs.

In practice, the actual motion of DOs might violate the worst-case assumptions. We refer to these DO motions as *unanticipated* behaviors. Also illegal behaviors, where a DO violates the traffic rules, are considered unanticipated behaviors. In presence of unanticipated behaviors of DOs the safety guarantee from combining SMPC with a robust backup does not hold. In fact, there might be time instants in which the SMPC solution is not certified to be safe, that is, the robust problem from the first predicted step is infeasible, and furthermore also the robust planner for the current initial condition could fail, that is, it could not be possible to plan a trajectory that is collision-free. In such situations, we plan the EV trajectory by minimizing the probability of collision over the prediction horizon. In our approach, we implement the robust planner and the probability-minimization planner using the Constraint Violation Probability Minimization (CVPM) approach [16], [17]: when feasible, a control action yielding zero probability of collision is applied (*robust case*); otherwise, the probability of collision is minimized (*probabilistic case*).

Our novel scheme, named SMPC+CVPM, is computationally efficient since it is not necessary to compute the robust backup after the solution of the SMPC problem. Instead, the solution of the cascaded OCP is replaced by a fast set computation to check the existence of the robust solution, and the logical structure can benefit from parallelization. Furthermore, addressing those situations in which a DO does not behave as anticipated greatly enhance the applicability of the algorithm. In fact, most other planners do not address these situations and just apply previously stored backup plans [14], [15]; however, in presence of unanticipated behaviors the stored backup is not supported by safety guarantees, nor it is designed to react to the unforeseen situation according to a reasonable criterion. We validate the algorithm and discuss the properties through simulations on benchmark scenarios from the CommonRoad dataset [18]. The simulation code is available at https://github.com/tbenciolini/Safe-and-Non-Conservative-Trajectory-Planning.

The remainder of the paper is organized as follows. Sec. II recalls the dynamical models used for the EV and the dynamic obstacles and presents the MPC-based trajectory planners that are considered in the control scheme. Sec. III introduces our novel SMPC+CVPM scheme, discussing its properties and giving an overview of the implementation details. An efficient way of solving the probability minimization is given in Sec. IV. Sec. V presents the simulations from the CommonRoad database. Sec. VI and VII, respectively, offer a discussion and conclusive remarks.

II. PRELIMINARIES

In the following we outline the dynamical models used for the EV and for DOs, Sec. II-A. Then, we outline the SMPC-based trajectory planners (Sec. II-B) and CVPM (Sec. II-C).

A. Models of Ego Vehicle and Dynamic Obstacles

In this section we introduce the dynamical models used to make predictions of the EV trajectories and of DOs. For the EV, we use a curvature-aware kinematic bicycle model referred to the road-aligned Frenet coordinate system [19]. The state of the EV $\boldsymbol{\xi} = [s, d, \phi, v]^\top$ consists of the longitudinal and lateral positions of the center of mass of the EV in the road-aligned reference frame, s and d, respectively, of the yaw angle ϕ, and of the linear velocity v. The input of the EV $\boldsymbol{u} = [a, \delta]^\top$ consists of the acceleration a and the front steering angle δ. In the following, an approximated discrete-time model is used

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{A}_\mathrm{d}(\boldsymbol{\xi}_k - \boldsymbol{\xi}_0) + \boldsymbol{B}_\mathrm{d}\boldsymbol{u}_k + \boldsymbol{c}_\mathrm{d}, \qquad (1)$$

that is obtained linearizing the kinematic bicycle model around the current state $\boldsymbol{\xi}_0$ and zero input $\boldsymbol{u}^* = [0, 0]^\top$, and discretizing with zero-order hold of sampling time T. All details about the linearization and discretization procedure, together with the explicit analytical expression of $\boldsymbol{A}_\mathrm{d} \in \mathbb{R}^{4 \times 4}$, $\boldsymbol{B}_\mathrm{d} \in \mathbb{R}^{4 \times 2}$, and $\boldsymbol{c}_\mathrm{d} \in \mathbb{R}^4$ from (1), are given in our previous work [20]. Such model couples the longitudinal and lateral dynamics of the vehicle and allows efficient computation. However, it does not accurately represent some of the dynamics features, such as the nonlinear tire characteristics, and the linearization introduces further approximations, that we neglect because within this work, we primarily focus on the uncertainty introduced by the unknown future trajectory of DOs. Different, more accurate models can also be used without compromising the validity of the presented approach. However, the considerations on the computational complexity given in the following hold true only for linear models. Similarly, uncertainty about the measurements of the EV state might be considered and accounted for in the derivation of the safety collision-avoidance constraints without significant changes in the algorithm.

For the assumed dynamics of the DOs, we adopt a double integrator model with decoupled dynamics for the longitudinal and lateral components. For ease of notation, in the following we refer to one DO only. However, all considerations hold similarly for multiple DOs close enough to the EV. The state of each DO is $\boldsymbol{\xi}^\mathrm{DO} = [s^\mathrm{DO}, v_s^\mathrm{DO}, d^\mathrm{DO}, v_d^\mathrm{DO}]^\top$, consisting of longitudinal and lateral positions and velocities, and the input $\boldsymbol{u}^\mathrm{DO} = [a_s^\mathrm{DO}, a_d^\mathrm{DO}]^\top$ comprises the longitudinal and lateral acceleration, a_s^DO and a_d^DO, respectively. We model the action of DO drivers as an LQR-based controller tracking

a constant reference state that expresses the intention of the driver, for example maintaining a given cruise speed or a lateral target position. The DO prediction model is thus denoted as

$$\boldsymbol{\xi}_{k+1}^{\text{DO}} = \boldsymbol{A}^{\text{DO}}\boldsymbol{\xi}_k^{\text{DO}} + \boldsymbol{B}^{\text{DO}}\boldsymbol{u}_k^{\text{DO}} \tag{2a}$$

$$\boldsymbol{u}_k^{\text{DO}} = \boldsymbol{K}^{\text{DO}}\left(\boldsymbol{\xi}_k^{\text{DO}} - \boldsymbol{\xi}_{\text{ref},k}^{\text{DO}}\right) + \boldsymbol{w}_k^{\text{DO}}, \tag{2b}$$

where $\boldsymbol{A}^{\text{DO}}$ and $\boldsymbol{B}^{\text{DO}}$ are the discrete-time point-mass model matrices as in [14], [20]. The input, which is bounded

$$\boldsymbol{u}_{\min}^{\text{DO}} \le \boldsymbol{u}^{\text{DO}} \le \boldsymbol{u}_{\max}^{\text{DO}}, \tag{3}$$

consists of a deterministic feedback loop and of a stochastic term. On the one hand, the deterministic term is designed to steer the state towards the reference state $\boldsymbol{\xi}_{\text{ref},k}^{\text{DO}}$, with a rate depending on the LQR-based controller gain $\boldsymbol{K}^{\text{DO}}$. This term represents the likely nominal trajectory being pursued. On the other hand, we account for uncertainty about the input for such nominal trajectory. Within the scope of this work, the uncertainty models the unpredictability of the human decision and the fact that the human intention is not known. Therefore, we consider the disturbance $\boldsymbol{w}^{\text{DO}}$ as part of the input. Such uncertainty is modeled as truncated Gaussian, $\boldsymbol{w}^{\text{DO}} \sim \mathcal{N}\left(0, \boldsymbol{\Sigma}^{w,\text{DO}}, \mathcal{W}^{\text{DO}}\right)$. Moreover, we assume that measurements of the DO states available to the EV are corrupted by the measurement noise $\boldsymbol{v}^{\text{DO}}$, which is a zero-mean truncated Gaussian distribution $\boldsymbol{v}^{\text{DO}} \sim \mathcal{N}\left(0, \boldsymbol{\Sigma}^{v,\text{DO}}, \mathcal{V}^{\text{DO}}\right)$. In this work, stochastic and robust control approaches are combined, therefore the assumption of bounded support for the distributions is later relaxed and unbounded support is assumed for the stochastic control approaches.

Definition 1: An unanticipated behavior of DOs is any trajectory that is not in the reachable space of model (2) subject to constraints (3), that is, trajectories that cannot be obtained from dynamics (2) for any input satisfying (3).

Remark 1: In practice, the actual DO control policy and intended behavior and are not known to the EV and therefore the assumptions could be violated. Thus, the bounds on the DO input (3) and the Gaussian probability distribution for the stochastic component of the DO input are to be intended as the assumption of the EV on the DO model, on what is considered an anticipated behavior.

Such unanticipated behaviors represent both, future trajectories that are not feasible given the assumed physical limitations of the DOs and future trajectories that violate traffic rules. These assumed models are used to derive safe areas for the EV in the MPC schemes presented in the following sections. More sophisticated prediction models that more precisely characterize unanticipated behaviors, possibly taking traffic rules into account, could be used without major modification of the proposed approach. For our control architecture, which is the main contribution of this work, it is sufficient that prediction models result in a bounded set of possible future positions of DOs.

B. Stochastic Model Predictive Control

In trajectory planning for automated driving, constraints account for the physical limitations of the EV, for the traffic rules, and for collision avoidance safety constraints.

However, the future positions of DOs are not known deterministically, rather, nominal predictions are affected by uncertainty. The introduced schemes consider the same prediction horizon length N and share the same cost function

$$J(\boldsymbol{\xi}_0, \boldsymbol{U}_N) = \|\Delta\boldsymbol{\xi}_N\|_{\boldsymbol{P}}^2 + \sum_{k=0}^{N-1}\left(\|\Delta\boldsymbol{\xi}_k\|_{\boldsymbol{Q}}^2 + \|\boldsymbol{u}_k\|_{\boldsymbol{R}}^2\right). \tag{4}$$

The cost J is designed to penalize large inputs, where $\boldsymbol{R} \succ 0$. Large deviations $\Delta\boldsymbol{\xi} = \boldsymbol{\xi} - \boldsymbol{\xi}^*$ of the EV from the desired reference $\boldsymbol{\xi}^*$ are penalized, with $\boldsymbol{Q}, \boldsymbol{P} \succeq 0$.

The schemes presented in the following differ in the formulation of safety constraints. In this work, for simplicity we refer to $\boldsymbol{x}$ as the current state of the environment, considering the current states of the EV and DOs, i.e. $\boldsymbol{x} = \left[\boldsymbol{\xi}^\top, \boldsymbol{\xi}^{\text{DO}\top}\right]^\top$. Therefore, we refer with SMPC($\boldsymbol{x}$) and CVPM($\boldsymbol{x}$) to problems evaluated for the same current traffic configurations, i.e., only differing in the formulation of the safety constraints.

In the SMPC approach, hard constraints for collision avoidance are relaxed and must be satisfied only up to a user-specified level of probability, $\beta < 1$. The OCP considered at each sampling time is

$$\text{SMPC}(\boldsymbol{x}): \min_{\boldsymbol{U}_N} J(\boldsymbol{\xi}_0, \boldsymbol{U}_N) \tag{5a}$$

$$\text{s.t. } \boldsymbol{\xi}_{k+1} = \boldsymbol{f}(\boldsymbol{\xi}_k, \boldsymbol{u}_k), \quad \forall k = 0,\dots,N-1 \tag{5b}$$

$$\boldsymbol{\xi}_k \in \mathcal{X}_k, \quad \forall k = 1,\dots,N \tag{5c}$$

$$\boldsymbol{u}_k \in \mathcal{U}_k, \quad \forall k = 0,\dots,N-1 \tag{5d}$$

$$\Pr[\boldsymbol{\xi}_k \in \mathcal{S}(\boldsymbol{\xi}_k^{\text{DO}})] \ge \beta, \forall k = 1,\dots,N, \tag{5e}$$

where N is the prediction horizon, $\boldsymbol{U}_N = \left[\boldsymbol{u}_0^\top,..,\boldsymbol{u}_{N-1}^\top\right]^\top$, $\boldsymbol{f}$ in (5b) is a compact representation of (1), and $\mathcal{X}_k$ and $\mathcal{U}_k$ in (5c) and (5d), respectively, are state and input hard constraints due to the physical limitations of the EV.

Constraints (5e) are probabilistic chance constraints: the EV must meet the safety requirements $\mathcal{S}(\boldsymbol{\xi}_k^{\text{DO}})$ with respect to the future positions of the DOs. The set of safe positions $\mathcal{S}(\boldsymbol{\xi}_k^{\text{DO}})$ is obtained considering the probability distribution of the uncertainties $\boldsymbol{w}^{\text{DO}}$, i.e., the unknown future behavior of DOs, and $\boldsymbol{v}^{\text{DO}}$, i.e., the measurement noise. To handle chance constraints numerically, we utilize unbounded disturbances and reformulate (5e) analytically using the assumptions on the probability distribution of the uncertainty. Observe that the SMPC approach allows a $1 - \beta$ probability of violation in open loop for each time step k in the prediction horizon, that is, in the predictions a collision is possible.

C. Constraint Violation Probability Minimization

In the following, we recall the concept of CVPM which was introduced in [16] and [17]. CVPM switches between two optimization problems to enable a robust solution as long as possible by solving the problem CVPM Robust($\boldsymbol{x}$), denoted as *robust case*. If CVPM Robust($\boldsymbol{x}$) is not feasible, the *probabilistic case* is applied, where CVPM Prob($\boldsymbol{x}$) is solved and minimizes the probability of constraint violation.

CVPM considers the set of safe EV state sequences as a Cartesian product of safe position sets, i.e., $\hat{\mathcal{S}}\left(\boldsymbol{\Xi}_N^{\text{DO}}\right) = \prod_{k=1}^N \mathcal{S}(\boldsymbol{\xi}_k^{\text{DO}})$, for a given DO state sequence

$\boldsymbol{\Xi}_N^{\mathrm{DO}} = [\boldsymbol{\xi}_1^{\mathrm{DO}\top}, \ldots, \boldsymbol{\xi}_N^{\mathrm{DO}\top}]^\top$. The set $\hat{\mathcal{S}}(\boldsymbol{\Xi}_N^{\mathrm{DO}})$ is implemented as in [14, Sec. V-B]. The robust case OCP is

$$\text{CVPM Robust}(\boldsymbol{x}) : \min_{\boldsymbol{U}_N} \; J(\boldsymbol{\xi}_0, \boldsymbol{U}_N) \tag{6a}$$

$$\text{s.t. } \boldsymbol{\xi}_{k+1} = \boldsymbol{f}(\boldsymbol{\xi}_k, \boldsymbol{u}_k), \qquad \forall k = 0, \ldots, N-1 \tag{6b}$$

$$\boldsymbol{\xi}_k \in \mathcal{X}_k, \qquad \forall k = 1, \ldots, N \tag{6c}$$

$$\boldsymbol{u}_k \in \mathcal{U}_k, \qquad \forall k = 0, \ldots, N-1 \tag{6d}$$

$$\Pr\left[\boldsymbol{\Xi}_k \notin \hat{\mathcal{S}}(\hat{\boldsymbol{\Xi}}_k^{\mathrm{DO}})\right] = 0. \tag{6e}$$

Constraint (6e) imposes that the EV state sequence satisfies the collision avoidance constraint robustly.

In the *probabilistic case* a solution with zero probability of constraint violation does not exist. Therefore, as in [16], we minimize the probability of constraint violation over the whole prediction horizon by solving the following OCP

$$\text{CVPM Prob}(\boldsymbol{x}) : \min_{\boldsymbol{U}_N} \; \Pr\left[\boldsymbol{\Xi}_k \notin \hat{\mathcal{S}}(\hat{\boldsymbol{\Xi}}_k^{\mathrm{DO}})\right] \tag{7a}$$

$$\text{s.t. } \boldsymbol{\xi}_{k+1} = f(\boldsymbol{\xi}_k, \boldsymbol{u}_k), \qquad \forall k = 0, \ldots, N-1 \tag{7b}$$

$$\boldsymbol{\xi}_k \in \mathcal{X}_k, \qquad \forall k = 1, \ldots, N \tag{7c}$$

$$\boldsymbol{u}_k \in \mathcal{U}_k, \qquad \forall k = 0, \ldots, N-1. \tag{7d}$$

More details on the minimization of the probability of constraint violation in (7a) are given in Sec. IV.

The distinction into two different cases, introduced in [17], is outlined in the following. The *robust case* is applied if

$$\exists \boldsymbol{U}_N : \Pr\left[\boldsymbol{\Xi}_N \notin \hat{\mathcal{S}}(\boldsymbol{\Xi}_N^{\mathrm{DO}})\right] = 0. \tag{8}$$

To evaluate this condition in practice, the set of feasible input trajectories for the robust case

$$\mathcal{U}_{\mathrm{robust}} = \left\{ \boldsymbol{U}_N \,\middle|\, \boldsymbol{\Xi}_N \in \hat{\mathcal{S}}(\boldsymbol{\Xi}_N^{\mathrm{DO}}), \forall \boldsymbol{w}_k^{\mathrm{DO}} \in \mathcal{W}^{\mathrm{DO}}, \forall \boldsymbol{v}_k^{\mathrm{DO}} \in \mathcal{V}^{\mathrm{DO}} \right\} \tag{9}$$

is determined. A fast linear program with $\mathcal{U}_{\mathrm{robust}}$ as a constraint is used to determine whether it is empty [21]. If $\mathcal{U}_{\mathrm{robust}}$ is nonempty, the linear program is feasible, and the robust case is applied. If $\mathcal{U}_{\mathrm{robust}} = \emptyset$, the probabilistic case is used. The solutions to the problems CVPM Robust$(\boldsymbol{x})$ and CVPM Prob$(\boldsymbol{x})$ yield the optimal input sequence $\boldsymbol{U}_N^*$, where the first element is applied to the system.

Remark 2: In SMPC, chance constraints (5e) are defined separately for each time step, to allow constraint tightening. In CVPM, the joint probability distribution is used for minimization, because reducing too much the probability of constraint violation in the first steps might increase the probability of constraint violation for further steps.

III. NOVEL SMPC+CVPM CONTROL SCHEME

In this section we outline the proposed novel SMPC+CVPM scheme and discuss its properties. SMPC plans a non-conservative EV trajectory. However, it accepts a non-zero risk of collision. We use CVPM to evaluate the safety of the SMPC trajectory. Furthermore, CVPM is also used to plan an alternative trajectory, that is safe for any realization of the uncertainty. If the real behavior of DOs violates the assumptions of the EV, then CVPM returns a trajectory which minimizes the probability of a collision.

The logic to decide whether SMPC or CVPM must be applied is illustrated in Figure 1. At each time step, at first, the SMPC$(\boldsymbol{x})$ OCP is solved, that is, an optimistic
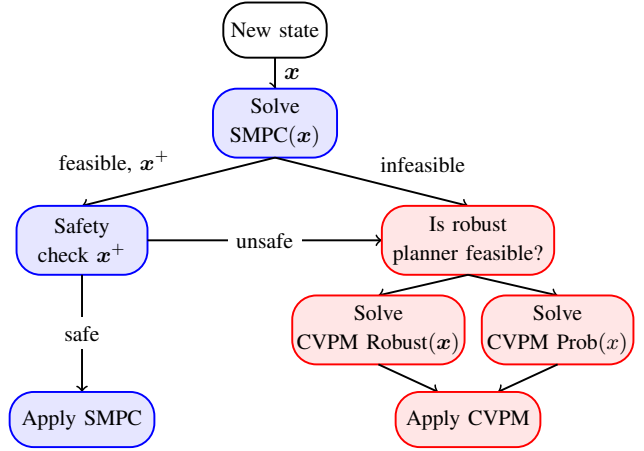

Fig. 1. Diagram of the logic of the novel SMPC+CVPM scheme.

trajectory based on the current traffic configuration $\boldsymbol{x}$ is computed. If a solution to SMPC$(\boldsymbol{x})$ exists, the predicted next step is considered, that is, the traffic configuration at the next sampling time, $\boldsymbol{x}^+$ resulting from the first input of the SMPC$(\boldsymbol{x})$ solution. From $\boldsymbol{x}^+$, a safety check is performed considering the prediction horizon from prediction step $k = 1$ to $k = N + 1$. The goal of the safety check is to verify whether after applying the solution of SMPC$(\boldsymbol{x})$, at the next time step it is still possible to determine a safe trajectory accounting for the worst case movement of the DOs, following a robust approach. Precisely, the existence of a feasible solution to OCP CVPM Robust$(\boldsymbol{x}^+)$ (6) is evaluated. Yet, a solution to CVPM Robust$(\boldsymbol{x}^+)$ must not be determined at present, in fact OCP (6) is not solved; rather, only the existence of a solution is discussed by performing a set computation and therefore the procedure is computationally undemanding. If the safety check succeeds, SMPC will lead to a safe state from which a feasible robust solution can be obtained at the next sampling time. In such case, the first input of the solution to SMPC$(\boldsymbol{x})$ is applied.

Definition 2: The solution to the SMPC$(\boldsymbol{x})$ OCP is safe if the safety check for $\boldsymbol{x}^+$ succeeds, that is, if it is certified that applying the SMPC solution leads to a state $\boldsymbol{x}^+$ from which a collision free trajectory can be planned, if DOs behave consistently with the assumptions in Sec. II-A.

Remark 3: The safety check in particular verifies the safety of the initial condition of the following OCP. Therefore, if the first predicted state of the SMPC trajectory does not satisfy collision avoidance constraints in a robust sense, the safety check fails as well. Rigorously speaking, the safety check succeeds if and only if $\mathcal{U}_{\mathrm{robust}} \neq \emptyset$ for $\boldsymbol{x}^+$, with $\mathcal{U}_{\mathrm{robust}}$ from (9), but also including initial condition in the state sequence $\boldsymbol{\Xi}_N$.

Remark 4: The predicted traffic configuration $\boldsymbol{x}^+$ used for the safety check is not the prediction used in the SMPC scheme, based on the approximated EV model (1). Rather, a nonlinear and continuous-time model is used to compute a more accurate prediction of the future state of the EV if the SMPC input is applied. Since the safety check is not part of the SMPC OCP, the predicted traffic configuration $\boldsymbol{x}^+$ is computed only once per iteration, after that the SMPC OCP

has been solved, and a continuous-time nonlinear model does not severely affect the computation time of the scheme. Yet, the prediction models of DOs must be consisted with the assumptions in Sec. II-A.

Since the SMPC($\boldsymbol{x}$) is not recursively feasible, sometimes it does not yield a solution. Furthermore, even when a solution exists, this might not be safe, that is, applying the first input prevents the existence of a solution to the robust planner for the first predicted step. In either of these situations, the OCP CVPM($\boldsymbol{x}$) is solved, as shown in the right branch in Figure 1. First, we check which case of the CVPM algorithm must be used. If possible, the robust case is solved, producing a trajectory that is collision-free for any realization of the uncertainty within the assumptions from Sec. II-A. If this is not possible, the probabilistic case is used to obtain a trajectory which minimizes the probability of collision.

Remark 5: If SMPC($\boldsymbol{x}$) does not yield a solution, it is unlikely that the robust case of CVPM($\boldsymbol{x}$) is applicable, since the latter is a robust planner with tighter constraints. Still, because of minor differences in the derivation of the safety constraints, this cannot be completely ruled out for rare cases. Nevertheless, the robust case of CVPM($\boldsymbol{x}$) is mainly used to plan a safe trajectory when the solution to SMPC($\boldsymbol{x}$) exists but is not safe, i.e., if it fails the safety check.

The SMPC+CVPM scheme allows to benefit from the efficient and optimistic SMPC planning when it is safe to do so, i.e., if a safe trajectory can be obtained for the predicted traffic configuration $\boldsymbol{x}^+$. Yet, the CVPM robust case computes a safe trajectory if the SMPC solution is not safe. Thus, if DOs move as expected according to the assumed models (Sec. II-A), our scheme yields always a collision-free trajectory.

Moreover, the proposed scheme addresses situations where the EV must deal with unforeseen behaviors of DOs, whose motion violates the hypotheses on which the robust motion planner is built. This might happen, for example, due to unanticipated behaviors of traffic participants (Definition 1). If the actual motion of the DOs violates the worst-case hypotheses on the disturbance distributions considered in the robust approach, the previously established safety guarantees do not longer hold. In such cases, the EV re-plans a new safe trajectory if possible; if not possible, the EV aims at minimizing the probability of a collision due to the possibly unlawful behavior of other traffic participants. Observe that if DOs behave as expected, the EV returns a collision-free trajectory, therefore possible collisions are due to the DO unanticipated behaviors, not caused by the EV.

The logical structure of the novel SMPC+CVPM scheme is computationally efficient and can be parallelized. In fact, the CVPM branch (in red in Figure 1) is independent of the solution of the SMPC branch, therefore can be computed in parallel and only applied if necessary. Furthermore, the safety check for $\boldsymbol{x}^+$, which follows the SMPC OCP, only consists of a set computation, since no safety backup must be computed or stored, only the existence of a solution must be evaluated. As a result, the solution of two cascaded optimization problems is never required to obtain the solution.

IV. PROBABILITY MINIMIZATION

In the *probabilistic case* of CVPM, the goal is to minimize the constraint violation probability as denoted in (7). In this section, we reformulate the minimization to solve a slightly different but efficient problem. Optimizing the probability in (7) requires numerical integration over a multivariate Gaussian, which is tedious and limits online usability. Hence, the problem is approximated with a quadratic program.

We first specify the set of feasible state sequence $\hat{\mathcal{S}}\left(\boldsymbol{\Xi}_N^{\mathrm{DO}}\right)$, derived by the procedure from [14], in a half space representation with n_{q} inequalities, i.e.,

$$\hat{\mathcal{S}}\left(\boldsymbol{\Xi}_N^{\mathrm{DO}}\right) = \{\boldsymbol{\Xi}_N \mid \boldsymbol{Q}_N \boldsymbol{\Xi}_N + \boldsymbol{q}_N \leq 0\}, \qquad (10)$$

where $\boldsymbol{Q}_N \in \mathbb{R}^{n_{\mathrm{q}} \times 4N}$, $\boldsymbol{q}_N \in \mathbb{R}^{n_{\mathrm{q}}}$ are the inequality matrix and the offset vector, respectively, and both are functions of the predicted DO state sequence. Since the DO state sequence $\boldsymbol{\Xi}_N^{\mathrm{DO}}$ is affected by the disturbance $\boldsymbol{w}_k^{\mathrm{DO}}$ and noise $\boldsymbol{v}_k^{\mathrm{DO}}$, the set $\hat{\mathcal{S}}\left(\boldsymbol{\Xi}_N^{\mathrm{DO}}\right)$ is uncertain, which is modeled with a random offset vector $\boldsymbol{q}_N$ following the normal distribution $\boldsymbol{q}_N \sim \mathcal{N}(\bar{\boldsymbol{q}}_N, \boldsymbol{\Sigma}^{\boldsymbol{q}_N})$ with the mean $\bar{\boldsymbol{q}}_N$ and covariance $\boldsymbol{\Sigma}^{\boldsymbol{q}_N}$. The optimization of (7a) is efficiently solved with

$$\text{CVPM Prob}(\boldsymbol{x}): \min_{\boldsymbol{U}_N, \boldsymbol{S}_N} \|\boldsymbol{S}_N - \boldsymbol{\mu}_N\|_{(\boldsymbol{\Sigma}^{\boldsymbol{q}_N})^{-1}}^2 \qquad (11\text{a})$$

$$\text{s.t. } \boldsymbol{\xi}_{k+1} = f(\boldsymbol{\xi}_k, \boldsymbol{u}_k), \qquad \forall k = 0, \dots, N-1 \quad (11\text{b})$$

$$\boldsymbol{\xi}_k \in \mathcal{X}_k, \qquad \forall k = 1, \dots, N \qquad (11\text{c})$$

$$\boldsymbol{u}_k \in \mathcal{U}_k, \qquad \forall k = 0, \dots, N-1 \quad (11\text{d})$$

$$\boldsymbol{\mu}_N = \boldsymbol{Q}\boldsymbol{\Xi}_N + \bar{\boldsymbol{q}}_N, \qquad (11\text{e})$$

$$\boldsymbol{S}_N \leq 0. \qquad (11\text{f})$$

The decision variable $\boldsymbol{S}_N \in \mathbb{R}^{4N}$ in (11a) is a non-positive slack variable. Problem (11) is a quadratic problem that is solved efficiently, yielding a solution of the optimal input sequence $\boldsymbol{U}_N^*$ to the original problem (7).

In the following, we derive the approximated quadratic program (11), by deriving the probabilistic distribution of the offset $\boldsymbol{q}_N$. We start by decomposing the state sequence of the DO $\boldsymbol{\Xi}_N^{\mathrm{DO}}$ in a nominal component $\bar{\boldsymbol{\Xi}}_k^{\mathrm{DO}}$ and the stochastic component based on the disturbances $\boldsymbol{w}_k^{\mathrm{DO}}$. For the minimization of the constraint violation probability, the untruncated Gaussian distribution of the disturbance is used, along the lines of [16], [22]. The nominal component $\bar{\boldsymbol{\Xi}}_N^{\mathrm{DO}}$ is determined with a prediction based on (2) resulting in the random prediction

$$\boldsymbol{\Xi}_N^{\mathrm{DO}} = \bar{\boldsymbol{\Xi}}_N^{\mathrm{DO}} + \boldsymbol{B}^{\mathrm{DO}} \boldsymbol{W}_N^{\mathrm{DO}}, \qquad (12)$$

where $\boldsymbol{B}^{\mathrm{DO}}$ is defined as in [21, Eq. (8.5)]. The covariance matrix of the disturbance sequence given as $\boldsymbol{\Sigma}^{\boldsymbol{W},\mathrm{DO}} = \mathrm{diag}\left(\boldsymbol{\Sigma}^{\boldsymbol{w},\mathrm{DO}}, \boldsymbol{\Sigma}^{\boldsymbol{w},\mathrm{DO}}, \dots\right)$, yielding the distribution of the DO state sequence as

$$\boldsymbol{\Xi}_N^{\mathrm{DO}} \sim \mathcal{N}\left(\bar{\boldsymbol{\Xi}}_N^{\mathrm{DO}}, \boldsymbol{B}^{\mathrm{DO}} \boldsymbol{\Sigma}^{\boldsymbol{W},\mathrm{DO}} \boldsymbol{B}^{\mathrm{DO}\top}\right). \qquad (13)$$

The offset $\boldsymbol{q}_N$ of (10) depends on the DO state sequence and therefore the covariance matrix for the offset $\boldsymbol{\Sigma}^{\boldsymbol{q}_N}$ is

$$\boldsymbol{\Sigma}^{\boldsymbol{q}_N} = \boldsymbol{Q}_N \boldsymbol{B}^{\mathrm{DO}} \boldsymbol{\Sigma}^{\boldsymbol{W},\mathrm{DO}} \boldsymbol{B}^{\mathrm{DO}\top} \boldsymbol{Q}_N^\top, \qquad (14)$$

and the mean of the offset vector $\bar{\boldsymbol{q}}_N$ is based on [14].

With the stochastic properties of the offset vector, the constraint violation probability is

$$\Pr\left[\boldsymbol{\Xi}_k \notin \hat{\mathcal{S}}(\hat{\boldsymbol{\Xi}}_k^{\mathrm{DO}})\right] = 1 - c \int_{\boldsymbol{S}_N \leq 0} e^{-\frac{1}{2}\|\boldsymbol{S}_N - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Sigma}^{\boldsymbol{q}_N - 1}}} d\boldsymbol{S}_N, \quad (15)$$

where $c = \frac{1}{\sqrt{(2\pi)^{4N}|\boldsymbol{\Sigma}^{\boldsymbol{q}_N}|}}$ and $\boldsymbol{\mu}_k = \boldsymbol{Q}_N \boldsymbol{\Xi}_N + \bar{\boldsymbol{q}}_N$. The constraint violation probability is determined by the integration of the constraint satisfaction probability (15). The integration variable $\boldsymbol{S}_N \in \mathbb{R}^{4N}$ is non-positive to integrate over the space where the left side of the constraint $\boldsymbol{Q}_N \boldsymbol{\Xi}_N + \boldsymbol{q}_N$ is negative, i.e., integration over the space where the constraint is fulfilled. We are not interested in the minimum value of (7a), rather in the minimizer $\boldsymbol{U}_N^*$. Therefore, the numerical integration in (15) is approximated by a quadratic program in (11a) that yields an approximated minimizer [22]. The approximation leads to a minimization of the probability by finding a mean value that is close to the boundary of the constraints based on a distance weighted by the covariance matrix. This method ensures that the solution is centered around the most probable scenarios for satisfying the constraints, rather than setting a strict upper or lower bound. The solver uses the integration variable $\boldsymbol{S}_N$ as decision variable, finding the smallest weighted distance, which is the dominant part in the integral (15).

V. SIMULATION RESULTS

We validate the proposed SMPC+CVPM scheme through numerical simulations from the CommonRoad database [18]. We compare the performance of our novel method to that of methods considering a pre-stored fail-safe trajectory as emergency backup plan, i.e., fail-safe trajectory planning (FTP) [14], [23]. Specifically, we tested the SMPC+FTP scheme from [14], where, to facilitate the comparison, FTP is implemented using CVPM robust case.

The code is implemented in Python and both the set computations and the solution to all OCPs are computed using *CVXPY* [24]. The EV trajectory is replanned every $T = 0.1$ s, since this is the sampling time of the trajectories in CommonRoad. Simulations were run on a computer with an AMD Ryzen 5 3500U eight-core processor.

We analyze the simulations of two challenging scenarios, selected because they consider trajectories of the DOs that trigger a reaction of the EV suitable to analyze the properties of our algorithm. We compare the performance of the control schemes by means of two metrics:

- the computation time, which is of primary concern for the real-time applicability of the algorithms, since the input must be computed within short sampling times to iteratively re-plan the optimal trajectory of the EV;
- the optimality of the closed-loop trajectory, assessed by computing the average stage cost:

$$J_{\mathrm{sim}} = \sum_{t=1}^{N_{\mathrm{sim}}} \frac{\left(\|\Delta\boldsymbol{\xi}(t)\|^2_{\boldsymbol{Q}} + \|\boldsymbol{u}(t)\|^2_{\boldsymbol{R}}\right)}{N_{\mathrm{sim}}}, \quad (16)$$

where N_{sim} is the number of steps in the scenario.

The values of the weighing matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, and the reference target state $\boldsymbol{\xi}^*$ in $\Delta\boldsymbol{\xi} = \boldsymbol{\xi} - \boldsymbol{\xi}^*$, are as in the SMPC
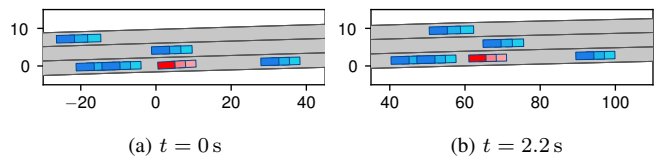

Fig. 2. Traffic configuration of CommonRoad scenario *DEU_A99-1_2_T-1*. The EV is represented in red. The driving direction is from left to right. Faded colors show the positions of vehicles in the following two steps.

OCP (4). Thus, J_{sim} evaluates the optimality of the algorithms in closed loop as for the efficiency goals of the SMPC.

A. Overtaking Maneuver

In the CommonRoad commonroad scenario *DEU_A99-1_2_T-1*, the EV is located on the right-most lane of the highway, surrounded by five other vehicles. The initial condition is represented in Figure 2a. The vehicle located in the middle lane, to the left of the EV, performs an overtaking maneuver.

At the beginning, the proposed SMPC+CVPM control scheme applies the SMPC control input, since the SMPC OCP is feasible and the safety check is successful, i.e., the SMPC solution is safe. Thus, the SMPC+CVPM control scheme takes advantage of the optimistic planning of SMPC. At simulation time $t = 2.2$ s, see Figure 2b, the vehicle performing the overtaking maneuver passes the EV and the optimal input obtained by SMPC fails in the safety check, that is, applying the SMPC solution might lead to a collision and the CVPM module is used. In this case, there exists a solution with zero probability of constraint violation (robust case). In fact, the behavior of the overtaking vehicle did not violate the assumptions on the model. Hence, CVPM is used only to prevent the risk of collision that SMPC would lead to, and a solution with zero probability of collision is obtained.

In turn, at the beginning the SMPC+FTP scheme [14] also always finds the SMPC solution to be safe and applies it. Yet, this is only possible after that at each sampling time a safety backup is computed and stored. At the end, the SMPC solution is not guaranteed to be safe, thus the algorithm applies the backup stored during the previous iteration.

In this scenario in which no unanticipated situation happens our novel SMPC+CVPM scheme yields an average stage cost (16) $J_{\mathrm{sim}} = 0.283$, which is marginally lower than the cost $J_{\mathrm{sim}} = 0.302$ of the SMPC+FTP scheme [14]. Further, we analyzed the average computation time for each module used, considering 100 simulations of the scenario. The average computation time for the SMPC branch for the SMPC+FTP scheme [14] is 12.287 ms, with 3.158 ms to solve the SMPC($\boldsymbol{x}$) problem and 10.144 ms to solve the FTP($\boldsymbol{x}^+$) problem. By contrast, in the SMPC+CVPM scheme 3.873 ms are required for the SMPC branch, namely 3.401 ms for the SMPC($\boldsymbol{x}$) problem and just 0.525 ms for the Safety check($\boldsymbol{x}^+$), and 0.554 ms for the CVPM branch. Since the problems are solved in parallel, our scheme requires only 3.873 ms, with a reduction of 70% with respect to the SMPC+FTP scheme [14]. The increased efficiency lies in the fact that the SMPC+FTP scheme [14], similar to [23], requires the solution of two cascaded OCPs (SMPC($\boldsymbol{x}$) and

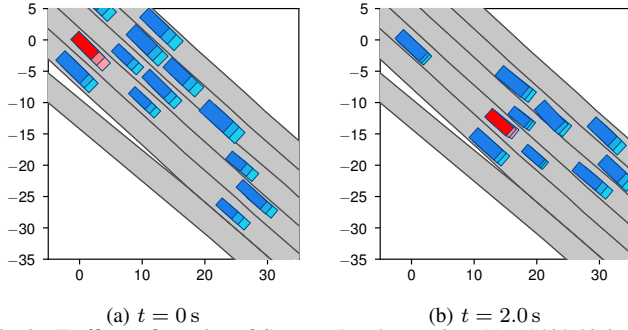(a) $t = 0\,\mathrm{s}$ (b) $t = 2.0\,\mathrm{s}$

Fig. 3. Traffic configuration of CommonRoad scenario *USA_US101-13_2_T-1*. The EV is represented in red. The driving direction is from top-left to bottom-right. Faded colors show the positions of vehicles in the following two steps.

FTP($\boldsymbol{x}^+$)), whereas in our SMPC+CVPM scheme SMPC($\boldsymbol{x}$) is followed by a quick set computation (Safety check($\boldsymbol{x}^+$)). Although it requires less computation time, our scheme still only applies the SMPC solution after ensuring it is safe.

B. Critical Emergency Braking on a Multi-Lane Highway

In the CommonRoad simulation *USA_US101-13_2_T-1*, the EV is located on the second lane from the right of the highway, surrounded by thirteen other vehicle, see Figure 3a. In the scenario, the traffic comes to a critical stop, forcing the EV to perform an emergency break in order to minimize the probability of collision with surrounding vehicles.

During the initial part of the braking maneuver, our novel SMPC+CVPM scheme applies the SMPC solution, as the safety check succeeds. Thus, at first the algorithm benefits from the efficient planning of SMPC. Yet, from simulation time $t = 2.0\,\mathrm{s}$, represented in Figure 3b, the SMPC OCP continues to provide a solution but the safety check fails, i.e., the SMPC solution might result in collisions. Hence, the solution provided by the CVPM branch is applied, obtained in parallel to the SMPC branch. Here, no solution with zero probability of constraint violation exists, because the breaking maneuver of the DOs is very sudden and the actual trajectory violates the assumptions on the DO behavior from Sec. II-A. Thus, the probabilistic case is used, i.e., the SMPC+CVPM scheme determines a new sequence of input explicitly aiming at minimizing the overall probability of collision. In this scenario, the trajectory consists of a braking and steering maneuver to the left, similar to the decision of several other human drivers in the recorded scenario. The position and heading of the EV at the end of the simulation is shown in Figure 4a.

At first, the SMPC+FTP scheme [14] also applies the SMPC solution, after computing a safety backup. When the SMPC solution is not guaranteed to be safe, the SMPC+FTP scheme [14] relies on the safety backup stored during the previous iterations, consists of a rapid deceleration while remaining at the center of the lane. Figure 4b shows the position and heading of the EV at the end of the simulation. Differently from the solution of our SMPC+CVPM scheme (Figure 4a), here the EV does not turn to the left to minimize the probability of collisions with DOs.
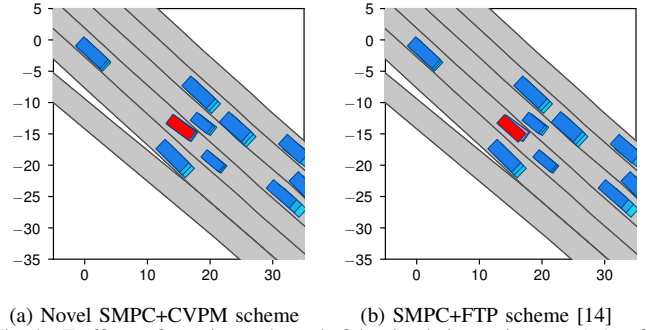

(a) Novel SMPC+CVPM scheme   (b) SMPC+FTP scheme [14]

Fig. 4. Traffic configuration at the end of the simulation at time $t = 2.4\,\mathrm{s}$ of CommonRoad scenario *USA_US101-13_2_T-1* used in the second simulation.

In this scenario the SMPC+FTP scheme [14] results in an average stage cost (16) of $J_{\mathrm{sim}} = 102.9$, whereas the novel SMPC+CVPM scheme yields a marginally lower cost, $J_{\mathrm{sim}} = 100.7$. Moreover, we analyzed again the average computation time for each module used, considering 100 simulations of the scenario. For the SMPC+FTP scheme [14], SMPC branch is solved in $23.115\,\mathrm{ms}$ on the average, with $10.326\,\mathrm{ms}$ to solve the SMPC($\boldsymbol{x}$) problem and $12.789\,\mathrm{ms}$ to solve the FTP($\boldsymbol{x}^+$) problem. In the SMPC+CVPM scheme, just $10.181\,\mathrm{ms}$ are required for the SMPC branch, namely $9.434\,\mathrm{ms}$ for the SMPC($\boldsymbol{x}$) problem and $0.746\,\mathrm{ms}$ for the Safety check($\boldsymbol{x}^+$). The CVPM branch is solved in $2.068\,\mathrm{ms}$ on the average, with $0.753\,\mathrm{ms}$ in Check case CVPM($\boldsymbol{x}$) and $1.316\,\mathrm{ms}$ in CVPM Prob($\boldsymbol{x}$). Since the problems are solved in parallel, our scheme requires only $10.181\,\mathrm{ms}$, with a reduction of 55% with respect to the SMPC+FTP scheme [14]. As computation time scales significantly with scenario complexity, such as a larger number of surrounding vehicles, this reduction can be crucial for practical application.

VI. DISCUSSION

One major advantage of the novel SMPC+CVPM scheme is that unanticipated situations, for which a feasible and safe solution cannot be determined, are explicitly addressed with a view at minimizing the probability of collision. Though intuitively reasonable, minimizing the probability of collision should not be the only objective of the EV. Between a first trajectory yielding a 40% probability of collision with anticipated 10% severity of the impact and a second trajectory in which the probability of collision is 30% and the expected severity of the impact is 20%, it is not obvious that the latter trajectory is preferable. The anticipated harm caused by a collision is to be taken into account. Sometimes choosing the trajectory yielding a higher probability of collision proves beneficial, if this in turns significantly reduces the severity caused by the collision. To balance among different types of conflicting objectives, i.e., minimizing the probability of collision and the anticipated severity of the impact, respectively, an ad-hoc cost function could be designed and used in the CVPM probabilistic case. The balance between these objectives is worth a special discussion and is not a purely technical matter, rather ethical, related to the well-known Trolley Problem [25].

Compared to schemes relying on a pre-stored fail-safe

trajectory [14], [23], our novel SMPC+CVPM scheme offers several advantages. Concerning the computation complexity, the SMPC+FTP scheme [14] intrinsically requires the cascaded solution of two OCPs, since to guarantee safety a backup for the next iteration must be stored. For short sampling times, the solution of two cascaded optimization problems is challenging. The scheme from [15] computes jointly the nominal and the backup trajectory, significantly increasing the size of the OCP and, thus, the computation time. In turn, our novel scheme does not require the computation of a backup and requires a rather smaller computation time. In practice, it is not necessary to pre-compute the backup, rather it suffices to verify that from the first predicted state of the SMPC trajectory a safe solution can be obtained. If necessary, such safe solution will be quickly computed at the next iteration by the robust case of CVPM, in parallel to the SMPC branch. Hence, the increased efficiency of our scheme does not compromise safety.

Besides, in presence of unanticipated behaviors of DOs that prevents the existence of a robust safe trajectory, applying the backup derived at the previous iteration [14] is not safe, as the safety backup is a valid and safe trajectory only under the assumption that the DOs behave as expected at the previous iteration. Hence, the scheme does not react to the present traffic configuration, which is hazardous and could lead to danger in practice if DOs exhibit unanticipated behaviors. At best, this leads to suboptimal behaviors in terms of minimization of the probability of collisions. In turn, our SMPC+CVPM scheme uses the probabilistic case of CVPM to plan the trajectory minimizing the probability of constraint violation.

VII. CONCLUSION

A novel scheme for safe trajectory planning is introduced, which permits to benefit from the efficient planning of SMPC. However, the sequence of inputs obtained by the SMPC planner is applied only if safe, i.e., if it will lead to a state from which it is possible to compute a feasible trajectory robust against the uncertainty. Therefore, the EV does not cause a collision for anticipated behaviors of traffic participants. In the event of unanticipated behaviors of traffic participants, the scheme yields a trajectory obtained by minimizing the probability of collision. The scheme is highly efficient, as the two OCPs can be solved in parallel and the approach can be modified to account for the harm of the collision. Numerical simulations based on scenarios drawn from the benchmark dataset CommonRoad [18] and comparing with the performance of the previous scheme from [14] demonstrated the benefit of our approach. The presented SMPC+CVPM scheme is suitable for application beyond autonomous driving, especially in frameworks characterized by large uncertainty because of the interaction with humans, e.g., human-robot-collaboration.

REFERENCES

[1] P. Kumar, M. Perrollaz, S. Lefèvre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *IEEE Intelligent Vehicles Symposium IV*, 2013.

[2] D. Phillips, T. Wheeler, and M. Kochenderfer, "Generalizable intention prediction of human drivers at intersections," in *IEEE IV*, 2017.

[3] U. Rosolia, A. Carvalho, and F. Borrelli, "Autonomous racing using learning Model Predictive Control," in *ACC*, 2017.

[4] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *IEEE IV*, 2011.

[5] A. Carvalho, Y. Gao, S. Lefevre, and F. Borrelli, "Stochastic predictive control of autonomous vehicles in uncertain environments," in *International Symposium on Advanced Vehicle Control*, 2014.

[6] B. Gutjahr, L. Gröll, and M. Werling, "Lateral Vehicle Trajectory Optimization Using Constrained Linear Time-Varying MPC," *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[7] S. Dixit, U. Montanaro, M. Dianati, D. Oxtoby, T. Mizutani, A. Mouzakitis, and S. Fallah, "Trajectory Planning for Autonomous High-Speed Overtaking in Structured Environments Using Robust MPC," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[8] R. Soloperto, J. Köhler, F. Allgöwer, and M. Müller, "Collision avoidance for uncertain nonlinear systems with moving obstacles using robust Model Predictive Control," in *ECC*, 2019.

[9] S. Söntges and M. Althoff, "Computing the Drivable Area of Autonomous Road Vehicles in Dynamic Road Scenes," *IEEE T-ITS*, 2018.

[10] J. H. Gillula, C. J. Tomlin, R. P. Fedkiw, and J.-C. Latombe, "Guaranteeing safe online machine learning via reachability analysis," Ph.D. dissertation, Stanford University, 2013.

[11] B. Schürmann, N. Kochdumper, and M. Althoff, "Reachset Model Predictive Control for Disturbed Nonlinear Systems," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018.

[12] A. Mesbah, "Stochastic Model Predictive Control: An Overview and Perspectives for Future Research," *IEEE Control Systems Magazine*, 2016.

[13] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear Model Predictive Control with chance constraints – A review," *Journal of Process Control*, 2016.

[14] T. Brüdigam, M. Olbrich, D. Wollherr, and M. Leibold, "Stochastic Model Predictive Control with a Safety Guarantee for Automated Driving," *IEEE Transactions on Intelligent Vehicles*, 2023.

[15] J. P. Alsterda, M. Brown, and J. C. Gerdes, "Contingency Model Predictive Control for Automated Vehicles," in *ACC*, 2019.

[16] T. Brüdigam, V. Gaßmann, D. Wollherr, and M. Leibold, "Minimization of constraint violation probability in model predictive control," *International Journal of Robust and Nonlinear Control*, 2021.

[17] M. Fink, T. Brüdigam, D. Wollherr, and M. Leibold, "Constraint Violation Probability Minimization for Norm-Constrained Linear Model Predictive Control," in *2022 European Control Conference (ECC)*, 2022.

[18] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: Composable benchmarks for motion planning on roads," in *IEEE IV*, 2017.

[19] M. Werling and L. Groll, "Low-level controllers realizing high-level decisions in an autonomous vehicle," in *IEEE IV*, 2008.

[20] T. Benciolini, D. Wollherr, and M. Leibold, "Non-Conservative Trajectory Planning for Automated Vehicles by Estimating Intentions of Dynamic Obstacles," *IEEE Transactions on Intelligent Vehicles*, 2023.

[21] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[22] M. Fink, T. Brüdigam, D. Wollherr, and M. Leibold, "Minimal Constraint Violation Probability in Model Predictive Control for Linear Systems," 2024, accepted at IEEE-TAC. [Online]. Available: http://arxiv.org/abs/2402.10538

[23] C. Pek and M. Althoff, "Computationally Efficient Fail-safe Trajectory Planning for Self-driving Vehicles Using Convex Optimization," in *International Conference on Intelligent Transportation Systems*, 2018.

[24] S. Diamond and S. Boyd, "CVXPY: A Python-Embedded Modeling Language for Convex Optimization," *Journal of Machine Learning Research*, 2016.

[25] M. Geisslinger, F. Poszler, J. Betz, C. Lütge, and M. Lienkamp, "Autonomous Driving Ethics: From Trolley Problem to Ethics of Risk," *Philosophy & Technology*, 2021.