Constraint Violation Probability Minimization for Norm-Constrained Linear Model Predictive Control

Michael Fink, Tim Brüdigam, Dirk Wollherr and Marion Leibold

Abstract— **In autonomous driving, it is essential to be able to avoid any type of collision with the environment by appropriate control. Therefore, the distance between vehicle and obstacles needs to be sufficiently large, providing a norm constraint e.g. for optimal control of the vehicle. In general, future positions of dynamic obstacles are highly uncertain and thus predictions are e.g. made using a stochastic model of the obstacle dynamics. We propose an application-independent framework that extends Linear Model Predictive Control to minimize the probability of norm constraint violation in the prediction horizon. Thus, for the autonomous driving application, the probability of collision is minimized. In contrast to Robust Model Predictive Control approaches, the proposed approach can deal with unexpected behavior of the obstacle without loss of feasibility.The applicability of the method is demonstrated in simulation of a vehicle that is successfully avoiding a suddenly emerging pedestrian.**

I. INTRODUCTION

Control applications like autonomous driving [1] and human-robot collaboration [2] have gained in importance over the last decade. In both applications, a collision between a human and a machine is a threat to the human and must be avoided. Therefore, in this paper, a control approach is proposed that allows to avoid collisions of a controlled system with dynamic obstacles and determines a dynamically feasible maneuver around the obstacle, e.g., avoiding a collision of a car with a pedestrian.

Collision avoidance is commonly included as constraint to the control problem and an established control method to handle constraints is Model Predictive Control (MPC) [3], [4]. MPC repeatedly solves constrained optimal control problems on a short prediction horizon to find suitable control inputs for the overall horizon. For collision avoidance problems, it is essential that each of the constrained optimal control problems is feasible, otherwise, a collision due to loss of control is possible. Therefore, recursive feasibility is a necessary and essential property of MPC in particular for collision avoidance, see [5], [6].

To account for uncertainties that e.g. come from highly uncertain predictions of obstacle behavior, MPC is extended to Robust Model Predictive Control (RMPC). RMPC in particular enables constraint admissibility despite of disturbances. The tube-based RMPC method in [7] uses tightened constraints such that even a worst-case realization of a bounded disturbance does not result in constraint violation. A disadvantage is that constraint tightening is performed offline and thus it is difficult to include constraints that change over time. Furthermore, the necessary safety margin leads to a conservative controller. In contrast, Stochastic Model Predictive Control (SMPC) enables less conservative controls despite of disturbances accepting a small risk of constraint violation and assuming that stochastic properties of the disturbances are known [8], [9]. A drawback of SMPC is that recursive feasibility is difficult to address. For example, the SMPC algorithm in [10] is provably recursively feasible for bounded disturbances. Though also here, an offline computation of tightened constraints is necessary. In the SMPC method presented in [11], recursive feasibility is guaranteed by providing a backup strategy, but only one specific application, autonomous driving, is addressed. Another way is is to replace the hard constraints by soft constraints when the current state is infeasible [3].

All SMPC approaches allow for a non-zero constraint violation probability. Though in many applications, it would be more beneficial to choose a control with almost zero or zero constraint violation probability as long as it exists. For example in autonomous driving, a vehicle may overtake with small distance accepting a small probability of collision or it may also overtake with slightly more distance resulting in a negligibly small probability of collision. Therefore, in [12] a method is introduced that first determines a set of inputs for the controlled system that leads to minimal probability of collision, and then applies the MPC optimization with this input set as a constraint. This method is referred to as Constraint Violation Probability Minimization (CVPM) and it uses the assumptions that the support of the disturbance is bounded and collision avoidance is modeled as a norm constraint. An advantage when comparing to RMPC methods is that it is not required to precompute tightened constraints and thus this approach e.g. allows for varying support of the disturbances over time without affecting recursive feasibility. However, in [12], the probability of constraint violation is evaluated only on a very short horizon of length 1. As a consequence, systems with complex dynamics start to oscillate or are not manageable at all.

Therefore, in this paper, the CVPM approach of [12] is extended to consider the minimization of the probabilistic norm constraint for a larger sub-horizon within the MPC prediction horizon. The CVPM method is designed such that recursive feasibility is always fulfilled and we will show that the origin is stabilized. Similar to [12], we assume that the controlled system is linear, the cost is quadratic, and the only uncertainty is in the position of the obstacle.

The remainder of this paper is structured as follows: Section II provides the problem statement. In Section III

M. Fink, T. Brüdigam, D. Wollherr, M. Leibold are with the Chair of Automatic Control Engineering, Technical University of Munich, Munich, Germany {michael.fink; tim.bruedigam; dw; marion.leibold}@tum.de

the proposed method is presented. Recursive feasibility and stability are investigated in Section IV. Simulation results are given in Section V and a discussion in Section VI. Section VII concludes the paper.

II. PROBLEM SETUP

First, we define the dynamics of the controlled system and the obstacle. Then, a brief overview of the underlying MPC method is given. The section concludes with a formal problem statement.

A. System Dynamics

The proposed method utilizes two models. The first is a deterministic linear discrete-time model for the controlled system. The second is a model of the obstacle that includes uncertainty. The controlled system is represented by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{1a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{1b}$$

with time instance k, states $\boldsymbol{x}_k \in \mathbb{R}^n$, control input $\boldsymbol{u}_k \in \mathbb{R}^m$, output $\boldsymbol{y}_k \in \mathbb{R}^q$, and matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$, $\boldsymbol{C} \in \mathbb{R}^{q\times n}$. The inputs are constraint to be within the input set $\boldsymbol{u}_k \in \mathcal{U}$ and the states must be within the state set $\boldsymbol{x}_k \in \mathcal{X}$ for all k. Both sets are convex and contain at least the origin. The input set is assumed to be compact and the state set is closed. The obstacle is modeled by the uncertain discrete-time dynamics

$$\boldsymbol{y}_{\mathrm{r},k+1} = \boldsymbol{y}_{\mathrm{r},k} + \boldsymbol{u}_{\mathrm{r},k} + \boldsymbol{w}_k, \tag{2}$$

where the output $\boldsymbol{y}_{\mathrm{r},k+1}$ depends on the previous output $\boldsymbol{y}_{\mathrm{r},k} \in \mathbb{R}^q$, a deterministic, known input $\boldsymbol{u}_{\mathrm{r},k} \in \mathbb{R}^q$, and a stochastic disturbance $\boldsymbol{w}_k \in \mathbb{R}^q$. Each disturbance is an independent realization of a random variable $\boldsymbol{W}_k$. It is assumed to have zero-mean and a radially decreasing probability density function. Furthermore, the support of the random variable is assumed to be bounded:

$$\boldsymbol{w}_k \in \{\boldsymbol{w} \mid \|\boldsymbol{w}\|_2 \leq w_{\max}\} \tag{3}$$

Here $\|\cdot\|_2$ denotes the Euclidean norm. The dynamics in (2) is referred to as obstacle model since the approach is motivated by obstacle avoidance applications. However, the method is presented independent of any application and $\boldsymbol{y}_{\mathrm{r},k}$ can be used to represent a state-space region that has to be avoided.

B. Model Predictive Control

The proposed CVPM method builds up on a standard MPC method, that is introduced in the following. MPC finds a control input for (1). Based on the current state, the MPC method optimizes the next N inputs, resulting in an optimal state trajectory in terms of the objective function. The first input of the determined input sequence is applied to the system and the process is repeated with the resulting state of the system in the following time step. The initial state of each iteration is denoted as $\boldsymbol{x}_0$, which is possible because the system is time-invariant. The optimal input minimizes the cost function

$$V_N(\boldsymbol{x}_0, \boldsymbol{U}) = \sum_{j=0}^{N-1} l(\boldsymbol{x}_j, \boldsymbol{u}_j) + V_{\mathrm{t}}(\boldsymbol{x}_N), \tag{4}$$

where N is the prediction horizon. The stage cost is chosen as $l(\boldsymbol{x}_j, \boldsymbol{u}_j) = \boldsymbol{x}_j^\top \boldsymbol{Q}\boldsymbol{x}_j + \boldsymbol{u}_j^\top \boldsymbol{R}\boldsymbol{u}_j$ and the terminal cost as $V_{\mathrm{t}}(\boldsymbol{x}_N) = \boldsymbol{x}_N^\top \boldsymbol{Q}_{\mathrm{t}}\boldsymbol{x}_N$. The index k is used for the ongoing time instance in which the control method is applied and the index j is used to denote the prediction time step. Our goal is to solve a regulation problem, i.e., the stabilization of the origin. The input sequence is

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{u}_0{}^\top & \boldsymbol{u}_1{}^\top & \dots & \boldsymbol{u}_{N-1}{}^\top \end{bmatrix}^\top, \tag{5}$$

and contains all inputs within the prediction horizon. The terminal set $\mathcal{X}_{\mathrm{f}}$ is a constraint for the last predicted state $\boldsymbol{x}_N$ and the set of admissible input sequences

$$\mathcal{U}_{\boldsymbol{x}_0} = \{\boldsymbol{U} \mid \forall j \in \mathbb{Z}_{0:N-1} : \boldsymbol{x}_N \in \mathcal{X}_{\mathrm{f}}, \boldsymbol{x}_j \in \mathcal{X}, \boldsymbol{u}_j \in \mathcal{U}\} \tag{6}$$

contains all input sequences that are feasible for the optimization. The set $\mathbb{Z}_{a:b}$ denotes the closed interval of integers between a and b.

Note that the admissible input set $\mathcal{U}_{\boldsymbol{x}_0}$ is a convex set since all input and state sets are convex and all relations between the variables are linear. Therefore, the optimization problem that is solved at each time instance is

$$\boldsymbol{U}^* = \arg\min_{\boldsymbol{U}} V_N(\boldsymbol{x}_0, \boldsymbol{U}) \tag{7a}$$

$$\text{s.t.} \quad \boldsymbol{x}_{j+1} = \boldsymbol{A}\boldsymbol{x}_j + \boldsymbol{B}\boldsymbol{u}_j, \quad j \in \mathbb{Z}_{0:N-1} \tag{7b}$$

$$\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}. \tag{7c}$$

The first input $\boldsymbol{u}_0{}^*$ of the optimal input sequence $\boldsymbol{U}^*$ is then applied to the system and the process is repeated. The MPC method with the optimal control problem (7) is the underlying method of the CVPM approach, introduced in this work. The following assumptions are made:

Assumption 1 (Terminal Set): The terminal set $\mathcal{X}_{\mathrm{f}}$ is a control invariant set and it holds that $\mathcal{X}_{\mathrm{f}} \subseteq \mathcal{X}$.

Assumption 2 (Lyapunov Function): The optimal cost function $V_N(\boldsymbol{x}_0, \boldsymbol{U}^*)$ is a Lyapunov function for the set of feasible initial states

$$\mathcal{X}_0 = \{\boldsymbol{x} \mid \mathcal{U}_{\boldsymbol{x}_0}(\boldsymbol{x}) \neq \emptyset\}. \tag{8}$$

With Assumption 1, the MPC method is recursively feasible [4] and Assumption 2 guarantees in addition stability [5].

C. Problem Statement and Contribution

In the following, we will again make use of the interpretation that the uncertain system (2) models an obstacle to the controlled system modeled by (1) and that we have to avoid collisions. The output of both system represents their locations and the distance between both must exceed a safety margin. This results in an additional constraint for the MPC problem in (7), which is the norm constraint

$$\left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \geq c \tag{9}$$

where c is the safety margin. The norm constraint ensures that no collisions occur in the predictions $j \in \mathbb{Z}_{1:N}$. Since the obstacle is modeled as an uncertain system, the constraint depends on a random variable. Therefore, it is not possible to add the norm constraint (9) to the MPC problem (7). For this purpose, a common method is SMPC with chance constraints, i.e.,

$$\Pr\left(\left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \leq c\right) \leq \beta_j. \tag{10}$$

The probability of violation of the constraint in (9) is then bounded by a user-defined parameter β_j. This SMPC approach allows a certain probability of constraint violation. In (3), it is assumed that the disturbance is bounded. Therefore, a violation probability of zero is possible in principle. For this reason, we will minimize the probability of constraint violation instead of inserting the chance constraint (10). Thus, as long as an input exists that provides a zero probability of violating the norm constraint, a control approach results that is similar to robust control. If such an input does not exist, the proposed method finds an input resulting in the smallest possible probability of constraint violation. The crucial point is the inclusion of the probability minimization in the nominal MPC problem. The solution for this issue is to use a subset of the admissible inputs such that only inputs are used that minimize the probability of constraint violation. This subset is determined in Section III.

In order to be able to consider the constraint (9) in the prediction horizon, an additional horizon $\hat{N} \in \mathbb{Z}_{1:N}$ is introduced for the proposed method. This CVPM horizon is a design parameter for the controller.

III. METHOD

This section introduces the CVPM method for norm constraints in (9) applied for $\hat{N}$ prediction steps. In Section III-A, the constraint violation probability is introduced and in Section III-B the general CVPM method is presented. Though, the general CVPM method is computationally expensive due to a non-convex optimization. Therefore in Section III-C an approximation for the probability is introduced, which is used to obtain a computationally feasible solution in Section III-D.

A. Constraint Violation Probability

In MPC, the dynamics (1) is used to do predictions for the state trajectory in the prediction horizon N. These predictions are used to optimize the cost (4) and will now be used to derive the probability that the norm constraint is violated. Additionally, for the norm constraint (9), the prediction of the obstacle outputs $\boldsymbol{y}_{\mathrm{r},j}, j \in \mathbb{Z}_{1:\hat{N}}$ is utilized, which is

$$\boldsymbol{y}_{\mathrm{r},j} = \boldsymbol{y}_{\mathrm{r},0} + \sum_{i=0}^{j-1} (\boldsymbol{u}_{\mathrm{r},i} + \boldsymbol{w}_i) = \overline{\boldsymbol{y}}_{\mathrm{r},j} + \sum_{i=0}^{j-1} \boldsymbol{w}_i, \tag{11}$$

where $\boldsymbol{y}_{\mathrm{r},0}$ is the measured position at the beginning of the prediction. It is assumed that the inputs to the obstacle model $\boldsymbol{u}_{\mathrm{r},i}$ are known. In the following the inputs $\boldsymbol{u}_{\mathrm{r},i}$ are included in the deterministic part $\overline{\boldsymbol{y}}_{\mathrm{r},j}$ of the prediction. This allows any dynamics to be modeled, since only the predicted outputs are considered.

For the constraint violation probability at the j-th step of the prediction, given in (10), the abbreviation

$$\begin{aligned} &p_{\mathrm{cv},j}(\boldsymbol{x}_0, \boldsymbol{u}_0 ... \boldsymbol{u}_{j-1}) = \\ &\Pr\left(\left\|\boldsymbol{y}_j(\boldsymbol{x}_0, \boldsymbol{u}_0 ... \boldsymbol{u}_{j-1}) - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \leq c\right) \end{aligned} \tag{12}$$

is used in the remainder of this paper. Since the predicted output $\boldsymbol{y}_j$ of the system depends on all previous inputs $\boldsymbol{u}_0, ..., \boldsymbol{u}_{j-1}$ and the current state $\boldsymbol{x}_0$, so does the probability $p_{\mathrm{cv},j}$. For simplicity, the notation of dependency is omitted in the following.

If the inequality $\left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \leq c$ is true for at least one $j \in \mathbb{Z}_{1:\hat{N}}$, then the norm constraint is violated within the CVPM horizon $\hat{N}$. The overall probability that at least one constraint is violated is given as the probability of the disjunction of all $\hat{N}$ inequalities and it is upper bounded by Boole's inequality, i.e.,

$$\Pr\left(\bigvee_{j=1}^{\hat{N}} \left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \leq c\right) \leq \sum_{j=1}^{\hat{N}} p_{\mathrm{cv},j}. \tag{13}$$

where the and-operator means that the argument of the probability is true if at least one inequality is true. Thus, if the sum of all particular probabilities is zero, no constraint violation occurs in the CVPM horizon $\hat{N}$.

B. General CVPM Method

The aim of the proposed method is to determine the input set $\mathcal{U}_{\mathrm{cvpm}} \subseteq \mathcal{U}_{\boldsymbol{x}_0}$ in each iteration of the controller such that the constraint violation probability $p_{\mathrm{cv},j}$ is as small as possible for all $j \in \mathbb{Z}_{1:\hat{N}}$. The set of admissible inputs $\mathcal{U}_{\boldsymbol{x}_0}$ in the optimization (7) is then replaced by the CVPM set $\mathcal{U}_{\mathrm{cvpm}}$. For this purpose, three different cases are distinguished. These cases are a generalization of the case analysis of [12]. For case 1, all input sequences of the admissible input set $\mathcal{U}_{\boldsymbol{x}_0}$ allow for zero probability of constraint violation $p_{\mathrm{cv},j}$ for all prediction steps within the horizon $\hat{N}$. In collision avoidance application, CVPM applies this case if the distance between the obstacle and the vehicle is large enough. Case 2 is when there is at least one time step in the CVPM horizon where a constraint violation probability of zero is not achievable for the input sequences of the admissible input set $\mathcal{U}_{\boldsymbol{x}_0}$. All remaining situations belong to case 3, i.e., there exists an input sequence $\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}$ such that a constraint violation probability $p_{\mathrm{cv},j}$ of zero is feasible for all $j \in \mathbb{Z}_{1:\hat{N}}$. The input sequence set $\mathcal{U}_{\mathrm{cvpm}}$ is derived differently in each case, as described in the following.

Case 1 - Constraint Admissibility Guarantee: The probability of constraint violation is zero in each time step within the horizon $\hat{N}$, independent of the choice of the input sequence $\boldsymbol{U}$, i.e.,

$$\forall \boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}, j \in \mathbb{Z}_{1:\hat{N}} : p_{\mathrm{cv},j} = 0. \tag{14}$$

Therefore, every input sequence $\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}$ is a valid sequence, from which it follows that $\mathcal{U}_{\mathrm{cvpm}} = \mathcal{U}_{\boldsymbol{x}_0}$.

Case 2 - Constraint Admissibility Impossible: There is no input sequence in $\mathcal{U}_{\boldsymbol{x}_0}$ such that the norm constraint is satisfied in presence of uncertainty for all steps within the horizon $\hat{N}$, i.e.,

$$\forall \boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0} : \exists j \in \mathbb{Z}_{1:\hat{N}} : p_{\mathrm{cv},j} > 0. \tag{15}$$

Since it is impossible to find an input that guarantees $p_{\mathrm{cv},j} = 0$ for all $j \in \mathbb{Z}_{1:\hat{N}}$, we propose to minimize the upper bound (13) of the probability that the constraint is violated within the CVPM horizon $\hat{N}$. The CVPM set $\mathcal{U}_{\mathrm{cvpm}}$ obtains the input sequence $\boldsymbol{U}^*$ that minimizes (13), i.e.,

$$\mathcal{U}_{\mathrm{cvpm}} = \left\{ \boldsymbol{U}^* \;\middle|\; \boldsymbol{U}^* = \arg\min_{\boldsymbol{U}\in\mathcal{U}_{\boldsymbol{x}_0}} \textstyle\sum_{j=1}^{\hat{N}} p_{\mathrm{cv},j} \right\}. \tag{16}$$

The solution of (16) is not unique because, on one hand, the sum of probabilities is non-convex and, on the other hand, in the case $\hat{N} < N$ the inputs $\boldsymbol{u}_j$ with $j > \hat{N}$ have no influence on the objective function in (16) and are thus still free.

Case 3 - Constraint Admissibility Possible: In this case, there exist inputs such that constraint admissibility can be guaranteed under uncertainties, i.e., a constraint violation probability of zero is feasible in each predicted time step. This is the case if

$$\exists \boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0} : \forall j \in \mathbb{Z}_{1:\hat{N}} : p_{\mathrm{cv},j} = 0. \tag{17}$$

The CVPM set then contains all input sequences that allow for zero constraint violation probability, i.e.,

$$\mathcal{U}_{\mathrm{cvpm}} = \mathcal{U}_{\boldsymbol{x}_0} \cap \left\{ \boldsymbol{U} \;\middle|\; \forall j \in \mathbb{Z}_{1:\hat{N}} : p_{\mathrm{cv},j} = 0 \right\}. \tag{18}$$

Note that conditions (14), (15) and (17) are difficult to evaluate, since the probability $p_{\mathrm{cv},j}$ is non-convex with respect to the inputs. Therefore, in the next section, the probability (12) is replaced by a convex approximation and thus the case decision becomes computationally simpler.

C. Approximation of Constraint Violation Probability

The constraint violation probability $p_{\mathrm{cv},j}$ is the probability that the distance between the obstacle and controlled system is smaller than the safety distance c. It can be upper bounded by applying the reverse triangle inequality on (9) yielding the lower bound of the norm constraint

$$\left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \geq \left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2 - \left\|\sum_{i=0}^{j-1} \boldsymbol{w}_i\right\|_2. \tag{19}$$

Therefore, the probability (12) is upper bounded by

$$p_{\mathrm{cv},j} \leq \Pr\left(\left\|\sum_{i=0}^{j-1} \boldsymbol{w}_i\right\|_2 \geq \left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2 - c \right). \tag{20}$$

The bound (20) of the constraint violation probability depends on the deterministic norm $\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2$. If $\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2$ increases, then $p_{\mathrm{cv},j}$ decreases. Vividly, for the vehicle application, the larger the distance to the obstacle, the smaller the probability of collision. This allows to replace the minimization of the constraint violation probability $p_{\mathrm{cv},j}$ with a maximization of the auxiliary function

$$h : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}, \tag{21}$$

which is scalar, twice differentiable and strictly monotonically increasing. The deterministic distance is applied as the argument of the auxiliary function h, i.e.,

$$h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) \tag{22}$$

and it increases with an increase of the norm. Since $p_{\mathrm{cv},j}$ is decreasing with an increasing norm $\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2$, small values of the auxiliary function h refer to a high constraint violation probability and vice versa. Therefore, in the following, we use a maximization of (22) instead of the minimization of the violation probability $p_{\mathrm{cv},j}$. A convenient choice of the auxiliary function is $h(x) = x^2$ since it simplifies (22) to an inner product.

D. Approximating Method of CVPM

In this section, the auxiliary function h, instead of the constraint violation probability $p_{\mathrm{cv},j}$, is used to compute the set $\mathcal{U}_{\mathrm{cvpm}}$. The following Lemma allows to find the value of the auxiliary function h representing a zero probability.

Lemma 1: For all $\boldsymbol{w}_i, i \in \mathbb{Z}_{0:j-1}$ in (3) it holds that

$$\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2 \geq c + j w_{\max} \implies \left\|\boldsymbol{y}_j - \boldsymbol{y}_{\mathrm{r},j}\right\|_2 \geq c. \tag{23}$$

Therefore, if the deterministic norm $\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2$ is greater then $c + j w_{\max}$, the constraint violation probability is $p_{\mathrm{cv},j} = 0$.

Proof: The reverse triangle inequality in (19) yields a lower bound of the norm (9). A further lower bound is given by assuming the worst-case disturbance, i.e.,

$$\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2 - j w_{\max} \geq c. \tag{24}$$

It follows that the smallest value of $\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2$ where (24) is satisfied is $c + j w_{\max}$. ∎

Therefore, for the auxiliary function h, $p_{\mathrm{cv},j} = 0$ if $h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) \geq h_{\mathrm{safe},j}$ with

$$h_{\mathrm{safe},j} = h\left(c + j w_{\max}\right). \tag{25}$$

Furthermore, the distinction of the cases requires the minimal and maximal feasible values of (22) with respect to the admissible inputs, which are

$$h_{\min,j} = \min_{\boldsymbol{U}\in\mathcal{U}_{\boldsymbol{x}_0}} h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) \tag{26}$$

and

$$h_{\max,j} = \max_{\boldsymbol{U}\in\mathcal{U}_{\boldsymbol{x}_0}} h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right), \tag{27}$$

respectively. Due to the linearity of the model in (1) and the properties of the norm, (22) is a convex function. The minimum can be found with a straightforward approach, because the admissible input set is also convex. However, the maximization is not a convex optimization. Nevertheless, a solution of (27) exists due to Bauer's maximum principle [13]. From this, it follows that the maximum of the function must be located at the edges of the constraint set, i.e., on the boundary of $\mathcal{U}_{\boldsymbol{x}_0}$. Using the auxiliary variables $h_{\mathrm{safe},j}$, $h_{\min,j}$, and $h_{\max,j}$ the cases of Section III-B can be reformulated as follows:

Case 1 - Constraint Admissibility Guarantee: This case occurs if all admissible inputs result in $p_{\mathrm{cv},j} = 0$. This is equivalent to the fact that all feasible values of (22) exceed $h_{\mathrm{safe},j}$, i.e.,

$$\forall j \in \mathbb{Z}_{1:\hat{N}} : h_{\min,j} \geq h_{\mathrm{safe},j}. \tag{28}$$

Therefore, the set $\mathcal{U}_{\mathrm{cvpm}} = \mathcal{U}_{\boldsymbol{x}_0}$ is used as a constraint for the MPC optimization.

Case 2 - Constraint Admissibility Impossible: There is no input sequence $\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}$ that guarantees that $p_{\mathrm{cv},j} = 0$ in each predicted time step within the horizon $\hat{N}$. The case is applied if

$$\exists j \in \mathbb{Z}_{1:\hat{N}} : h_{\max,j} < h_{\mathrm{safe},j}. \tag{29}$$

Since large values of (22) refer to a low constraint violation probability, it is desirable that (22) is as large as possible. Therefore, the auxiliary function h is used in the optimization (16), resulting in

$$\boldsymbol{U}^* = \underset{\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}}{\arg\max} \sum_{j=1}^{\hat{N}} h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) \tag{30a}$$

$$\mathcal{U}_{\mathrm{cvpm}} = \left\{\boldsymbol{U}^*\right\}. \tag{30b}$$

The optimization in (30) is convex and it yields a approximation of the solution of the optimization in (16).

Case 3 - Constraint Admissibility Possible: In all other cases, i.e.,

$$\forall j \in \mathbb{Z}_{1:\hat{N}} : h_{\max,j} \geq h_{\mathrm{safe},j} \tag{31}$$

it is possible to find input sequences leading to zero constraint violation probability. In terms of the function h, all input sequences are valid that fulfill the condition

$$h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) \geq h_{\mathrm{safe},j} \tag{32}$$

for each j within the horizon $\hat{N}$. However, in general, the set of input sequences fulfilling (32) is non-convex. For an efficient MPC algorithm, a convex set is needed, therefore, a linear approximation of (32) is determined. This is achieved by using a linear subspace that is calculated by the linear part of a Taylor series of h. The evaluation point is $\boldsymbol{y}_j = \boldsymbol{\xi}_j$ and it is chosen such that

$$h\left(\left\|\boldsymbol{\xi}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right) = h_{\mathrm{safe},j}. \tag{33}$$

Let

$$h_{\boldsymbol{\xi}_j} = \nabla_{\boldsymbol{y}_j} h\left(\left\|\boldsymbol{y}_j - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2\right)\Big|_{\boldsymbol{y}_j = \boldsymbol{\xi}_j} \tag{34}$$

be the gradient of (22) at the point $\boldsymbol{\xi}_j$, then the linear constraint at the j-th prediction step is

$$h_{\boldsymbol{\xi}_j}^\top (\boldsymbol{y}_j - \boldsymbol{\xi}_j) \geq 0. \tag{35}$$

From (35) it follows that for each predicted time step within the horizon $\hat{N}$ a particular linear constraint exists. The input sequence must fulfill each constraint, therefore, the intersection of the linear constraints and the admissible input set is the CVPM set

$$\mathcal{U}_{\mathrm{cvpm}} = \mathcal{U}_{\boldsymbol{x}_0} \cap \left\{ \boldsymbol{U} \;\middle|\; \bigcap_{j \in \mathbb{Z}_{1:\hat{N}}} h_{\boldsymbol{\xi}_j}^\top (\boldsymbol{y}_j - \boldsymbol{\xi}_j) \geq 0 \right\}. \tag{36}$$

The choice of the evaluation point $\boldsymbol{\xi}_j$ is not unique; It is subject only to condition (33). Therefore, a suitable choice for the evaluation point is

$$\boldsymbol{\xi}_j = (c + j w_{\max}) \frac{\boldsymbol{y}_0 - \overline{\boldsymbol{y}}_{\mathrm{r},j}}{\left\|\boldsymbol{y}_0 - \overline{\boldsymbol{y}}_{\mathrm{r},j}\right\|_2} + \overline{\boldsymbol{y}}_{\mathrm{r},j}, \tag{37}$$

which is a point that is located between the current position of the controlled system $\boldsymbol{y}_0$ and the known position of the obstacle, such that (33) holds.

Remark 1: Since the set in (36) is an intersection of several sets, it is possible that $\mathcal{U}_{\mathrm{cvpm}} = \emptyset$. In this case, the set $\mathcal{U}_{\mathrm{cvpm}}$ must be determined with the procedure of case 2.

For the problems in Section III-B and Section III-D, we obtain the optimization

$$\boldsymbol{U}^* = \underset{\boldsymbol{U}}{\arg\min}\, V_N(\boldsymbol{x}_0, \boldsymbol{U}) \tag{38a}$$

$$\text{s.t.} \quad \boldsymbol{x}_{j+1} = \boldsymbol{A}\boldsymbol{x}_j + \boldsymbol{B}\boldsymbol{u}_j,\ j \in \mathbb{Z}_{0:N-1} \tag{38b}$$

$$\boldsymbol{U} \in \mathcal{U}_{\mathrm{cvpm}} \subseteq \mathcal{U}_{\boldsymbol{x}_0}, \tag{38c}$$

where V_N is defined as in (4). Therefore, only those input sequences are allowed that have a minimal constraint violation probability.

IV. PROPERTIES OF CVPM

In the following, we investigate recursive feasibility and stability.

A. Recursive Feasibility

Definition 1 (Recursive feasibility [14]): A control law $\boldsymbol{\mu}(\boldsymbol{x})$ is recursively feasible in $\mathcal{A} \subseteq \mathcal{X}$ if for all $\boldsymbol{x} \in \mathcal{A}$ admissible inputs exist, i.e., $\mathcal{U}_{\boldsymbol{x}_0} \neq \emptyset$, and $\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\mu}(\boldsymbol{x}) \in \mathcal{A}$. We can show, that the CVPM approach is recursively feasible if the underlying MPC is recursively feasible.

Lemma 2: The underlying MPC problem (7) is recursively feasible if Assumption 1 holds.

Proof: See [4, Theorem 13.1]. ■

Moreover, [4] points out that the recursive feasibility is given for all input sequences $\boldsymbol{U}$ as long as they are feasible, i.e. $\boldsymbol{U} \in \mathcal{U}_{\boldsymbol{x}_0}$. The CVPM method uses inputs from the CVPM set $\mathcal{U}_{\mathrm{cvpm}} \subseteq \mathcal{U}_{\boldsymbol{x}_0}$ leading to the following theorem for recursive feasibility.

Theorem 1: The approximation method from Section III-D is recursively feasible if $\mathcal{U}_{\boldsymbol{x}_0}$ is a non-empty set.

Proof: First, we will show that $\mathcal{U}_{\mathrm{cvpm}}$ is always non-empty and $\mathcal{U}_{\mathrm{cvpm}} \subseteq \mathcal{U}_{\boldsymbol{x}_0}$. In case 1, $\mathcal{U}_{\mathrm{cvpm}}$ is equal to $\mathcal{U}_{\boldsymbol{x}_0}$. In case 2, the only element of $\mathcal{U}_{\mathrm{cvpm}}$ is selected from $\mathcal{U}_{\boldsymbol{x}_0}$ by optimization. If case 3 leads to an empty set, based on Remark 1 case 2 is applied, otherwise $\mathcal{U}_{\mathrm{cvpm}}$ results from an intersection with $\mathcal{U}_{\boldsymbol{x}_0}$.

For that reason $\mathcal{U}_{\mathrm{cvpm}}$ contains at least one element if $\mathcal{U}_{\boldsymbol{x}_0}$ is non-empty. Since only input sequences from the admissible input set $\mathcal{U}_{\boldsymbol{x}_0}$ are used, i.e., $\mathcal{U}_{\mathrm{cvpm}} \subseteq \mathcal{U}_{\boldsymbol{x}_0}$, the method remains recursively feasible due to Lemma 2. ■

B. Stability

Unlike recursive feasibility, stability cannot be shown in every situation because stability is only meaningful if the path to reach the origin is feasible. Since the priority is to avoid collision with an obstacle, the state cannot converge to the origin if it is occupied by the obstacle. Thus, we will only provide a stability of the origin for the case where the obstacle does not restrict the mobility of the system, which arises when the obstacle is far enough away from the system and the origin.

Assumption 3: For a $k_0 < \infty$, there exists an input sequence $\boldsymbol{U}$ such that for all time instances $k \geq k_0$ it holds that $p_{\text{cv},j} = 0$ for all $j \in \mathbb{Z}_{1:\hat{N}}$ and for the position of the obstacle it holds that $\left\|\boldsymbol{y}_{\text{r},k}\right\|_2 > \left\|\boldsymbol{y}_k\right\|_2 + c + \hat{N}w_{\max}$.

This assumption holds if the obstacle is sufficiently far away from the origin. The stability of the origin in CVPM is mainly based on the stability properties of the underlying MPC. First, the stability of the general method from Section III-B is shown and afterward the reasoning is also used for the approximation method in Section III-D.

Theorem 2: If Assumptions 2 and 3 hold, the proposed CVPM method from Section III-B stabilizes the origin from time instance k_0 on.

Proof: Based on Assumption 3, case 2 is not used for $k \geq k_0$, since a constraint violation probability of zero is always feasible. Furthermore, the path to the origin is not occupied since the distance between obstacle and origin is larger than the distance between origin and the worst-case position of the controlled system at the last predicted step of the method. Therefore, the MPC optimization is performed without norm-constraints on the path to the origin and stability is achieved due to Assumption 2. ■

The proven stability of the general method allows extending the reasoning to the approximation method.

Corollary 1: With the approximation method in Section III-D, the origin of the controlled system is stable.

Proof: Theorem 2 is based on Assumption 3, which assumes that from time step k_0 on, there exists an input such that $\forall j \in \mathbb{Z}_{1:\hat{N}} : p_{\text{cv,j}} = 0$. This expression is equivalent to

$$\forall j \in \mathbb{Z}_{1:\hat{N}} : h_{\max,j} \geq h_{\text{safe},j} \tag{39}$$

due to the choice of $h_{\text{safe},j}$. Therefore, only case 1 and 3 are possible and Theorem 2 also holds for the approximation method in Section III-D. ■

V. SIMULATION

In this section, a numerical simulation is presented using CVPM as a control strategy for obstacle avoidance of an autonomously driving vehicle. All quantities are given in SI units and the simulation has been carried out with Matlab. The model in (1) is a double integrator that represents the vehicle dynamics. The system matrices are given as

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & T \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{40}$$

where the sample time is $T = 0.1$. The model is similar to the model in [15]. The inputs are the longitudinal and lateral accelerations of the system. The states x_1 and x_2 are referred to as the longitudinal and lateral positions, respectively. The lateral position is limited to $2 \leq x_2 \leq 8$ such that the full vehicle is within the lane, which goes from 0 to 10. The absolute values of the velocities and inputs are constrained, i.e., $|x_3| \leq 20$, $|x_4| \leq 20$, $|u_1| \leq 5$, and $|u_2| \leq 5$. The cost function V_N of the optimization (38) is used here for stabilizing the velocities and the state x_2 with $\boldsymbol{Q} = \text{diag}([0, 1, 1, 1])$ and $\boldsymbol{R} = \text{diag}([0.001, 0.001])$. The terminal cost weighting matrix $\boldsymbol{Q}_\text{t}$ is determined by solving the discrete-time algebraic Riccati equation [3] and the terminal set is chosen control invariant, calculated with the MPT3 toolbox [16]. Because of the structure of $\boldsymbol{B}$ and $\boldsymbol{C}$, the inputs have no direct influence on the outputs, i.e., $\frac{\partial \boldsymbol{y}_1}{\partial \boldsymbol{u}_0} = \boldsymbol{CB} = \boldsymbol{0}$. Therefore, with a short CVPM horizon $\hat{N} = 1$, the optimization in (30) is not able to find a maximum, because the objective does not change with the optimization variable. The effect of the input $\boldsymbol{u}_0$ only impacts subsequent outputs, i.e., from $\boldsymbol{y}_2$ on. For this reason, the CVPM horizon is chosen $\hat{N} \geq 2$, making the single-step approach from [12] not suitable. Here the MPC horizon and the CVPM horizon are chosen to be equal, i.e. $N = \hat{N} = 10$. The obstacle in this example is a pedestrian suddenly appearing on the side that wants to cross the road. Therefore, the input $\boldsymbol{u}_{\text{r},k}$ is assumed to be known and deviations are modeled with the disturbance $\boldsymbol{w}_k$. The reader is referred to [17] for the prediction of pedestrian movements. The support of the probability distribution of the disturbance is given with $w_{\max} = 0.2$. The vehicle and the pedestrian are modeled as circles with radii 2 and 1, respectively. Therefore a safety distance of $c = 3$ is chosen. [17]

The arrangement of vehicle and pedestrian is shown in Fig. 1. The vehicle and the pedestrian are over-approximated by circles. At the beginning of the simulation, no obstacle is present. Therefore, case 1 is used, where all inputs of $\mathcal{U}_{\boldsymbol{x}_0}$ are admissible and the standard MPC is applied. The vehicle is visualized as a blue circle. The direction of the velocity is visualized with a blue arrow located in the circle. At $t = 7.5\,\text{s}$ a pedestrian suddenly appears, represented as a red circle. The pedestrian is crossing the street, thus its trajectory will cross the track of the vehicle. Due to the momentum of the vehicle, an instantaneous change of the direction of driving is not possible. Therefore, a collision in the predicted steps is possible, i.e., the collision probability of zero is not feasible. For this reason, the CVPM method selects case 2, which applies the input resulting in the smallest collision probability for all predicted steps. In the figure, the applied acceleration is shown as a red arrow pointing in the opposite direction. The vehicle is slowing down and steering away from the pedestrian. Case 2 is also applied in the next time steps and the vehicle is avoiding the obstacle. Due to the momentum, the distance between vehicle and pedestrian decreases at $t = 8.2\,\text{s}$. However, the constraint violation probability of zero for the next ten prediction steps is again feasible, since

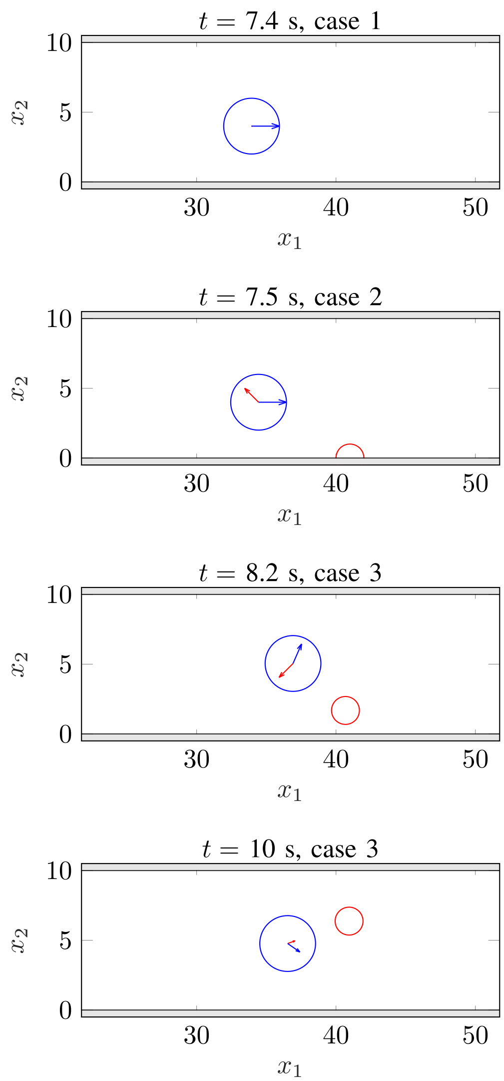

Fig. 1. Vehicle (blue circle) drives with constant speed, while suddenly a pedestrian (red circle) appears. The blue arrow shows the direction of the velocity and the red arrow is the system input, i.e., the acceleration.

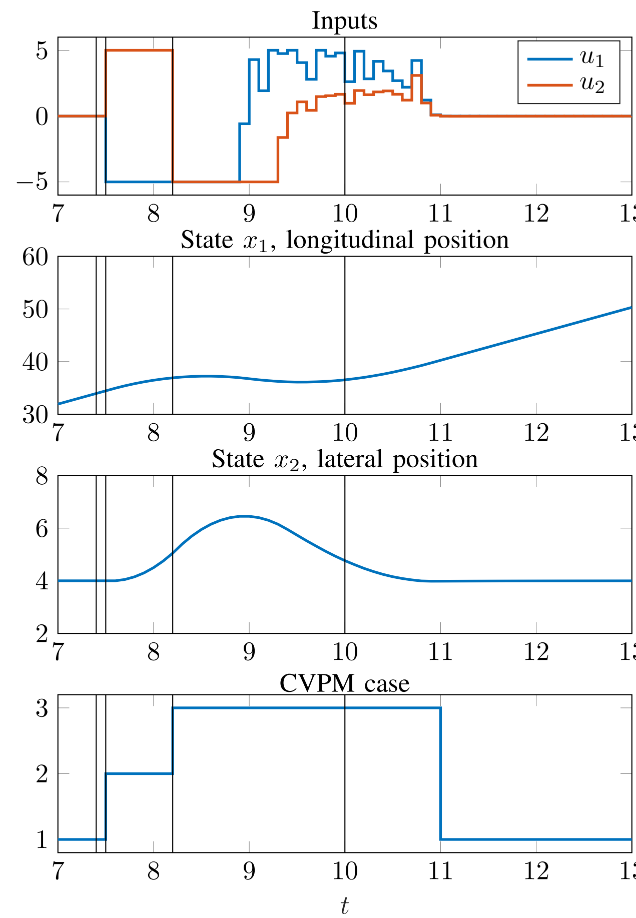

Fig. 2. Simulation results for the inputs and positions of the controlled vehicle. The black vertical lines indicate the time where the snapshots from Fig. 1 are taken. The fourth subplot shows the CVPM cases over time.

the direction of driving has changed. Consequently, case 3 is applied. The car cannot pass the pedestrian on the left side, since the pedestrian's trajectory restricts this position in the CVPM horizon $\hat{N}$. Therefore, the car moves to the right side to eventually overtake the obstacle after $t = 10\,\mathrm{s}$. The input is then selected to compensate for the lateral position of the vehicle. The snapshots from Fig. 1 are indicated as black vertical lines in the simulation results in Fig. 2. The second and third subplots show the progression of the position from the vehicle position. The CVPM case used in the respective simulation step is shown in the fourth subplot. It can be seen that the inputs saturate when case 2 is used. Note that obstacle avoidance may not be feasible if for this scenario only a standard MPC approach is used.

VI. Discussion

The proposed approach is a generalization of the single-step approach in [12]. Instead of taking into account only the constraint violation probability $p_{\mathrm{cv},1}$ for the first step in the prediction horizon, $p_{\mathrm{cv},j}$ for all $j \in \mathbb{Z}_{1:\hat{N}}$ is taken into account. Since the single-step method can only react within one step, it is applicable only for simple dynamics, e.g., a single integrator. In contrast, the proposed method is able to deal with more complex systems and a double integrator system is used in Section V for demonstration.

The CVPM horizon $\hat{N}$ is a parameter, which allows to adjust the conservativeness of the method. A large CVPM horizon allows to take more predictions of future obstacle locations into account but predictions far in the future are also more uncertain. Additionally, in case 3 the approximations of non-convex constraints (36) is conservative and a longer horizon adds more conservatism to the method.

In contrast, a small horizon $\hat{N}$ allows for riskier behavior. For example, for $\hat{N} < 4$ instead of $\hat{N} = 10$ in the scenario of Section V, a collision with the pedestrian occurs, because a longer horizon would be necessary to change the direction of the vehicle trajectory due to its momentum. Furthermore, when choosing $\hat{N} < 3$ in this scenario, the vehicle will overtake the obstacle on the left side and finally collide with the pedestrian. This is because only a few predictions of the pedestrian's behavior are taken into account and passing on the left appears possible.

Analyzing computational complexity, we find that the majority of the time is needed for the calculation of (26) and (27). In both optimizations, the optimization variable is the input sequence $\boldsymbol{U}$, thus the time needed for a single optimization increases as N increases. Therefore, an increasing of $\hat{N}$ yields a linear increasing in the computation time, since the optimizations are done $\hat{N}$ times. In the scenario of Section V, a single evaluation of the method takes $0.261\,\mathrm{s}$ in average.

Once constraint admissibility is possible (case 3), the norm constraint (9) is satisfied even under the worst case disturbances. However, the constraint is not handled robustly, as in RMPC, because CVPM only considers disturbances within the CVPM Horizon $\hat{N}$. In the subsequent execution of CVPM, it is possible that the last predicted step has an unavoidable probability of constraint violation due to the disturbance. Therefore the case where constraint admissibility is possible (case 3) cannot be applied. However, this does not lead to a loss of recursive feasibility, since the case where constraint admissibility is impossible (case 2) is still applicable in this situation.

However, CVPM is more flexible compared to tube-based RMPC [3], because tube-based RMPC requires an offline computation of the constraint tightening. Therefore, in contrast to CVPM, tube-based RMPC is not able to handle time-variant constraints.

VII. CONCLUSIONS

In this paper, we propose a multi-step approach for the CVPM method, extending previous work on MPC with CVPM. The extended method optimizes the probability of constraint violation within the next $\hat{N}$ predicted steps. Since the constraint violation probabilities from several predicted time steps are taken into account, a predictive behavior can be achieved. The advantage of the method is that it is able to handle unexpected situations, such as a pedestrian suddenly appearing, as shown in the simulation example. Therefore, unmodeled situations can be handled and the method is flexible with respect to changing conditions. A drawback, however, is that the method is limited to one type of constraint, namely norm constraints. Investigating a generalization to a wider class of constraints, e.g., polytopic constraints, seems promising.

ACKNOWLEDGMENTS

This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) with the project number 490649198.

REFERENCES

[1] G. Schildbach and F. Borrelli, "Scenario model predictive control for lane change assistance on highways," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 611–616, 2015.

[2] M. Vasic and A. Billard, "Safety issues in human-robot interactions," in *2013 IEEE international conference on robotics and automation*, pp. 197–204, 2013.

[3] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub., 2009.

[4] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[5] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[6] L. Grüne, "Nominal model predictive control," in *Samad, Tariq; Baillieul, John (Hrsg.): Encyclopedia of Systems and Control*, London: Springer, 2014.

[7] D. Q. Mayne, M. M. Seron, and S. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[8] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 30–44, 2016.

[9] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear model predictive control with chance constraints–a review," *Journal of Process Control*, vol. 44, pp. 53–67, 2016.

[10] M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer, "Constraint-tightening and stability in stochastic model predictive control," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3165–3177, 2016.

[11] 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*, 2021.

[12] 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.

[13] C. Niculescu and L.-E. Persson, *Convex functions and their applications*. Springer, 2006.

[14] L. Grüne, "NMPC without terminal constraints," *IFAC Proceedings Volumes*, vol. 45, no. 17, pp. 1–13, 2012.

[15] T. Bäthge, S. Lucia, and R. Findeisen, "Exploiting models of different granularity in robust predictive control," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 2763–2768, 2016.

[16] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0," in *Proc. of the European Control Conference*, pp. 502–510, 2013.

[17] F. Camara, N. Bellotto, S. Cosar, F. Weber, D. Nathanael, M. Althoff, J. Wu, J. Ruenz, A. Dietrich, G. Markkula, A. Schieben, F. Tango, N. Merat, and C. Fox, "Pedestrian models for autonomous driving part II: High-level models of human behavior," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5453–5472, 2021.