


Minimization of constraint violation probability in model predictive control

Tim Brüdigam¹  | Victor Gaßmann² | Dirk Wollherr¹ | Marion Leibold¹

¹Department of Electrical and Computer Engineering, Chair of Automatic Control Engineering, Technical University of Munich, Munich, Germany

²Department of Informatics, Chair of Robotics, Artificial Intelligence and Real-time Systems, Technical University of Munich, Munich, Germany

Correspondence

Tim Brüdigam, Chair of Automatic Control Engineering, Technical University of Munich, Arcisstr. 21, 80333 Munich, Germany.

Email: tim.bruedigam@tum.de

Abstract

For Model Predictive Control in safety-critical systems it is not only important to bound the probability of constraint violation but to reduce this constraint violation probability as much as possible. Therefore, an approach is necessary that minimizes the constraint violation probability while ensuring that the Model Predictive Control optimization problem remains feasible even under changing uncertainty. We propose a novel two-step Model Predictive Control scheme that yields a solution with minimal constraint violation probability for a norm constraint in an environment with uncertainty. After minimal constraint violation is guaranteed, the solution is then also optimized with respect to other control objectives. Recursive feasibility and convergence of the method are proved. A simulation demonstrates the effectiveness of the proposed method for a collision avoidance example.

KEYWORDS

chance constraint, recursive feasibility, robust model predictive control, safety-critical system, stochastic model predictive control

1 | INTRODUCTION

Autonomous systems in safety-critical applications, such as autonomous driving or human–robot interaction, depend on controllers that are able to safely and efficiently cope with uncertainties. In these applications, autonomous vehicles and robots must avoid collisions to ensure safety, while also optimizing other objectives, for example, energy consumption. For this problem setup, Model Predictive Control (MPC) is a promising method due to its ability to cope with hard constraints while optimizing an objective function.

Classic MPC methods deal well with deterministic systems and provide guarantees for stability as well as recursive feasibility.^{1–3} Recursive feasibility ensures that the MPC optimization problem remains feasible at future time steps if the optimization problem is initially feasible.

More advanced MPC algorithms are necessary in the presence of uncertainty in the system. Robust Model Predictive Control (RMPC) methods⁴ provide control laws that satisfy the control objectives and constraints by accounting for the worst-case realization of the uncertainty, assuming that the bound of the probability distribution for the uncertainty is known a priori.^{5,6} A major drawback of RMPC is that it accounts for worst-case uncertainty realizations, resulting in conservative control laws. This can be problematic in applications with high levels of uncertainty, for example, autonomous driving in dense traffic.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2021 The Authors. *International Journal of Robust and Nonlinear Control* published by John Wiley & Sons Ltd.

This issue is resolved by Stochastic Model Predictive Control (SMPC) methods, which exploit knowledge of the uncertainty by introducing probabilistic constraints. Instead of considering the worst possible uncertainty realization as in RMPC, SMPC methods introduce a risk parameter that specifies how likely a constraint violation may be. This probabilistic constraint is referred to as a chance constraint. In many applications, it is acceptable to allow a small probability of constraint violation. This results in a positive effect on performance, as the control law is no longer required to account for unlikely uncertainty realizations. Extensive summaries of diverse SMPC methods are given by Mesbah⁷ and specifically for linear SMPC by Farina et al.⁸ Stability of SMPC without a terminal constraint is addressed by Lorenzen et al.,⁹ while a performance analysis comparing MPC and SMPC is presented by Seron et al.¹⁰ Oldewurtel et al.¹¹ proposed an SMPC approach where the constraint is adapted online, based on the frequency of constraint violations.

In general, it is difficult to handle probabilistic constraints directly, requiring a transformation into a tractable deterministic representation of the chance constraint. For Gaussian uncertainties, an analytic expression can be determined,¹² while other uncertainties require different methods such as the particle MPC and the scenario MPC approach.^{13,14}

However, the benefits of using chance constraints come with the disadvantage of constraint violations if unlikely uncertainty realizations occur. This also leads to the problem of ensuring recursive feasibility, that is, guaranteeing that the MPC optimization problem remains solvable at every step. If an uncertainty realization with low probability occurs, it is possible that no admissible control input exists that satisfies the chance constraints. Recursive feasibility also becomes an issue if the maximal uncertainty value is not constant, that is, the bound of the uncertainty probability density function changes over time.

Recursive feasibility of SMPC for bounded disturbances is addressed by Korda et al.¹⁵ A further approach addressing recursive feasibility in SMPC are stochastic tube methods^{16,17} using constraint tightening. Lorenzen et al.¹⁸ suggested an approach that combines the works of Korda et al.¹⁵ and Kouvaritakis et al.¹⁶ where a tuning parameter is introduced that allows for shifting priority between performance and an increased feasible region. Covariance steering-based SMPC^{19,20} is a further SMPC approach that ensures recursive feasibility for linear systems with unbounded noise. Recursive feasibility in SMPC for probabilistically constrained Markovian jump linear systems is addressed by Lu et al.²¹ Recursively feasible SMPC with closed-loop chance constraint satisfaction for potentially unbounded disturbance distributions is presented in Hewing et al.²²

Due to its ability to efficiently cope with environments subject to uncertainty, SMPC has become increasingly popular in applications such as process control,^{12,23} energy control²⁴ and power systems,^{25,26} finance,²⁷ general automotive applications,²⁸ as well as more specifically safety-critical applications, for example, path planning¹³ and autonomous driving.²⁹⁻³⁴ However, the possible constraint violation and the resulting infeasibility of the optimization problem are limiting factors when designing an efficient SMPC algorithm in practice, especially for safety-critical applications.

A further drawback of chance constraints in SMPC appears if the optimal solution is “on the chance constraint” even though other solutions are possible with no or minimal effects on the cost function. A solution of the SMPC optimization problem minimizes the cost function and satisfies the required probability for the chance constraint. There may be other solutions with low cost that have a chance constraint violation probability less than required by the risk parameter or even zero. However, the SMPC optimization problem is considered to be solved once a solution is found with minimal cost that satisfies the chance constraint, given the risk parameter. This means that the solution with a lower constraint violation probability is not obtained. Additionally, choosing a suitable risk parameter is challenging, as high values increase risk, while low values reduce efficiency.

These issues are especially relevant in safety-critical systems. One example is an autonomous vehicle that plans to avoid collisions in an uncertain environment, for example, a car avoiding a bicycle with uncertain behavior. If the bound of the uncertainty is not known a priori, RMPC algorithms are either not applicable or require that the vehicle does not move until all surrounding vehicles are distanced enough. This, however, is not practical. Therefore, the collision constraint, realized with a norm constraint, could be transformed into a chance constraint in an SMPC approach, allowing a small collision probability. While this yields a more efficient solution than RMPC, applying SMPC may potentially result in a collision. Further, if the chance constraint in SMPC cannot be satisfied anymore because an unlikely scenario occurred or the uncertainty bound changed, the optimization problem becomes infeasible. Alternative control laws and recovery strategies can then be used to regain a feasible controller. However, in such scenarios where the chance constraint cannot be satisfied, the controller should ideally yield the safest solution possible, which is not guaranteed with standard recovery strategies. In the example of the autonomous vehicle, the safest solution is the one with the lowest collision probability.

In this paper we propose a novel MPC strategy for linear, discrete-time systems that not only satisfies general hard constraints over the entire prediction horizon, but additionally minimizes the probability of violating a norm constraint in the next predicted step while also optimizing for other control objectives. This is achieved by first calculating a set that

constrains the system inputs such that only those inputs are allowed that minimize the constraint violation probability. This is then followed by an optimization problem, which optimizes further required objectives, such as fast reference tracking or energy consumption. In this subsequent optimization problem, only those inputs are admissible that minimize the norm constraint violation probability. The proposed method can handle time-varying bounds of the system uncertainty and considers hard constraints on the states and inputs for the entire prediction horizon, for example, due to actuator limitations. Choosing a suitable risk parameter is not required.

We will first present the general method to minimize constraint violation probability. For the general method, it can be difficult to determine a tightened set of admissible inputs that guarantee minimal constraint violation probability. Therefore, we suggest an approach that allows the computation of this tightened input set for uncertainties with symmetric, unimodal probability density functions, that is, the relative likelihood of uncertainty realizations decreases with increased distance to the mean. This tractable approach yields a convex set of inputs, minimizing the constraint violation probability. Guarantees are provided for recursive feasibility, even for an increasing uncertainty bound, and for convergence of the proposed MPC algorithm. In the following, we will refer to the proposed method as CVPM-MPC, that is, MPC with constraint violation probability minimization (CVPM). A simulation for a vehicle collision avoidance scenario shows the effectiveness of the proposed method and highlights its advantages compared to SMPC and RMPC.

In summary, the main contributions are as follows:

- Proposition of a general CVPM-MPC method to minimize the constraint violation probability for the next predicted step while satisfying state and input constraints and optimizing further objectives
- Derivation of a specific CVPM-MPC approach for uncertainties with symmetric, unimodal probability density function
- Guarantee of recursive feasibility and convergence for the CVPM-MPC method

The proposed CVPM-MPC method can be beneficial to multiple applications, especially to safety-critical applications such as autonomous driving or human–robot interaction where the risk measure regarding collision is norm-based.^{29,35,36} In these safety-critical applications, there is a clear priority on maximizing safety, that is, the constraint violation probability of safety constraints needs to be minimal, before optimizing other objectives, for example, energy consumption. The proposed CVPM-MPC is particularly suited for potentially changing uncertainty bounds.

While in general it is possible to minimize the constraint violation probability not only for the first step but for multiple steps, this significantly increases conservatism, resulting in solutions that are more similar to RMPC solutions. Minimizing the constraint violation probability of the first step iteratively yields the advantage of safer solutions than SMPC and less conservatism compared to RMPC. We therefore focus on iteratively minimizing the constraint violation probability for the next step, that is, the first predicted MPC step, however, a solution approach for a multistep CVPM-MPC method is also provided.

The remainder of the paper is structured as follows. Section 2 introduces the system to be considered, the uncertainty, and the control objective. The proposed method is introduced in Section 3, first focusing on minimizing the constraint violation probability, then introducing the resulting MPC algorithm. Section 4 analyzes the properties of the suggested method, guarantees on recursive feasibility and convergence, while the CVPM-MPC method is discussed in Section 5. An example of the applied method is given in Section 6, simulating a vehicle collision avoidance scenario. Section 7 provides conclusive remarks.

Notation: Regular letters indicate scalars, bold lowercase letters denote vectors, and bold uppercase letters are used for matrices, for example, a , \mathbf{a} , \mathbf{A} , respectively. Random variables are represented by bold uppercase letters. The Euclidean norm is denoted by $\|\cdot\|_2$. The probability of an event is given by $\mathbb{P}(\cdot)$ and a probability density function is described by f if a probability density function exists. The support of a probability density function is denoted by $\text{supp}(f)$. Step k of a state or parameter is represented by a subscript k , for example, \mathbf{x}_k for state \mathbf{x} . The integers in the interval between a and b , including the boundaries, are denoted by $\mathbb{I}_{a:b}$.

2 | PROBLEM SETUP

In this section, we define the system class and the general MPC algorithm. Additionally, a probabilistic norm constraint is introduced. Based on these preliminaries, the problem statement is given subsequently.

2.1 | System dynamics and control objective

Consider the controlled linear, time-invariant, discrete-time system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \quad (1a)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k, \quad (1b)$$

with time step k , states $\mathbf{x}_k \in \mathbb{R}^n$, control input $\mathbf{u}_k \in \mathbb{R}^m$, output $\mathbf{y}_k \in \mathbb{R}^q$, and matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{q \times n}$.

Furthermore, we consider an uncertain system, which, for instance, describes the trajectory of an object. The uncertain system dynamics are given by

$$\mathbf{y}_{r,k+1} = \mathbf{y}_{r,k} + \mathbf{u}_{r,k} + \mathbf{w}_k \quad (2a)$$

$$= \bar{\mathbf{y}}_{r,k+1} + \mathbf{w}_k, \quad (2b)$$

depending on the output $\mathbf{y}_{r,k} \in \mathbb{R}^q$ at step k , a deterministic, known input $\mathbf{u}_{r,k} \in \mathbb{R}^q$, and a stochastic part $\mathbf{w}_k \in \mathbb{R}^q$, which is the realization of a random variable \mathbf{W}_k . The nominal prediction of $\mathbf{y}_{r,k+1}$ is indicated by $\bar{\mathbf{y}}_{r,k+1} = \mathbf{y}_{r,k} + \mathbf{u}_{r,k}$, consisting of the previous output $\mathbf{y}_{r,k}$ and the deterministic, known input $\mathbf{u}_{r,k}$.

Assumption 1 (Uncertainty properties). The random variables $\mathbf{W}_k(\mathbf{w}_k) \sim f_{\mathbf{w}_k}$ with the probability distribution $p_{\mathbf{w}_k}$ and density function $f_{\mathbf{w}_k}$ have zero mean and are truncated with the initially known, convex and bounded support $\text{supp}(f_{\mathbf{w}_k})$.

The support of $f_{\mathbf{w}_k}$ is given by

$$\text{supp}(f_{\mathbf{w}_k}) = \{\mathbf{w}_k \mid \|\mathbf{w}_k\|_2 \leq w_{\max,k}\} \quad (3)$$

where $w_{\max,k} \in \mathbb{R}_{\geq 0}$.

The controller for (1) is designed to optimize a finite horizon objective function while accounting for input and state constraints. In the following, the index k represents regular time steps, whereas the index j indicates prediction steps within an MPC optimization problem.

2.2 | Model predictive control

We consider an MPC algorithm to solve the control problem with a finite horizon objective function V . MPC repeatedly solves an optimization problem on a finite horizon. After the optimization, only the first optimized control input is applied. Then, the horizon is shifted by one step and a new optimization is performed. Without loss of generality it is assumed that an MPC optimization starts with \mathbf{x}_0 . The finite horizon cost $V(\mathbf{x}_0, \mathbf{U}_0)$, with the MPC horizon N , continuous stage cost $l(\mathbf{x}_j, \mathbf{u}_j) = \mathbf{x}_j^\top \mathbf{Q}\mathbf{x}_j + \mathbf{u}_j^\top \mathbf{R}\mathbf{u}_j$ with $l(\mathbf{0}, \mathbf{0}) = 0$, $\mathbf{Q} \in \mathbb{R}^{n \times n}$, $\mathbf{R} \in \mathbb{R}^{m \times m}$, $\mathbf{Q} \succcurlyeq 0$, $\mathbf{R} \succ 0$, and continuous terminal cost $V_f(\mathbf{x}_N)$ with $V_f(\mathbf{0}) = 0$, is then given by

$$V(\mathbf{x}_0, \mathbf{U}_0) = \sum_{j=0}^{N-1} l(\mathbf{x}_j, \mathbf{u}_j) + V_f(\mathbf{x}_N), \quad (4)$$

with the input sequence $\mathbf{U}_0 = (\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1})$.

We first formulate the MPC optimization problem with input constraints \mathcal{U} and state constraints \mathcal{X} , which are independent of the uncertain system (2), resulting in

$$V^* = \min_{\mathbf{U}_0} V(\mathbf{x}_0, \mathbf{U}_0), \quad (5a)$$

$$\text{s.t. } \mathbf{x}_{j+1} = \mathbf{A}\mathbf{x}_j + \mathbf{B}\mathbf{u}_j, \quad j \in \mathbb{I}_{0:N-1} \quad (5b)$$

$$\mathbf{u}_j \in \mathcal{U}, \quad j \in \mathbb{I}_{0:N-1} \quad (5c)$$

$$\mathbf{x}_{j+1} \in \mathcal{X}, \quad j \in \mathbb{I}_{0:N-1} \quad (5d)$$

$$\mathbf{x}_N \in \mathcal{X}_f. \quad (5e)$$

The input \mathbf{u}_j is bounded by the nonempty input value space $\mathcal{U} \subseteq \mathbb{R}^m$, that is, the input constraint (5c). The convex state constraint and terminal constraint sets are denoted by \mathcal{X} and \mathcal{X}_f , respectively.

Assumption 2 (Control invariant terminal set). For all $\mathbf{x}_j \in \mathcal{X}_f$, there exists an admissible \mathbf{u}_j such that $\mathbf{x}_{j+1} \in \mathcal{X}_f$.

Assumption 3 (Lyapunov functions). The cost V is a Lyapunov function in \mathcal{X} and the terminal cost V_f is a control Lyapunov functions in \mathcal{X}_f .

We denote with \mathbb{U}_j the set of admissible inputs \mathbf{u}_j such that all constraints of (5) are satisfied for $j \in \mathbb{I}_{0:N-1}$, that is,

$$\mathbb{U}_j = \left\{ \mathbf{u}_j \mid (5c), (5d), (5e) \right\} \quad \forall j \in \mathbb{I}_{0:N-1}. \quad (6)$$

2.3 | MPC with a probabilistic norm constraint

In the following, the uncertain system (2) is considered.

Assumption 4 (Initially known uncertainty). The initial state $\mathbf{y}_{r,0}$ and deterministic input $\mathbf{u}_{r,0}$ are known at the beginning of each MPC optimization.

Here, we consider an additional constraint for the MPC problem (5), which is the norm constraint

$$\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 \geq c_k, \quad (7)$$

representing a constraint on the 2-norm $\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2$, for example, the distance between two points must not be smaller than a minimal value c_k . While (7) is a hard constraint, we will first transform (7) into a chance constraint and later, in Section 3, we will minimize the probability that this norm constraint is violated.

Remark 1. It is also possible to consider a p -norm constraint with $\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_p$ instead of the 2-norm. Similar to the 2-norm, all p -norms are convex. Without loss of generality we will consider the 2-norm, as most applications require a 2-norm to represent the Euclidean distance.

As $\mathbf{y}_{r,k}$ is subject to uncertainty, the norm constraint (7) is difficult, potentially impossible, to fulfill, or it might lead to overly conservative control inputs. The hard norm constraint (7) can be relaxed if substituted by the chance constraint

$$\mathbb{P} \left(\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 < c_k \right) \leq \beta_k, \quad (8)$$

with

$$p_k := \mathbb{P} \left(\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 < c_k \right), \quad (9)$$

where β_k is a risk parameter and p_k denotes the constraint violation probability for the norm constraint (7). The constraint violation probability p_k for step k is evaluated at step $k - 1$, that is, at the previous step. Therefore, the probability p_k depends on the input \mathbf{u}_{k-1} , yielding \mathbf{y}_k according to (1).

The following example will illustrate the idea of the chance constraint. We consider a controlled object with position \mathbf{y}_k and a dynamic obstacle with position $\mathbf{y}_{r,k}$ where $\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2$ is the distance between both objects. The objects collide if $\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 < c_k$. An interpretation for (8) is that p_k represents the probability of a collision and this constraint violation probability is bounded by a predefined risk parameter β_k . A similar example is analyzed in a simulation in Section 6.

The bounded support of p_k is given by

$$\text{supp}(p_k) = \{\mathbf{u}_{k-1} | p_k > 0\}, \quad (10)$$

resulting in $p_k = 0$ if the maximal uncertainty value $w_{\max, k-1}$ cannot cause $\|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 < c_k$.

While it is possible to consider the norm constraint (7) over multiple steps, we will only consider the norm constraint for the next predicted step $j = 1$ with a horizon $N \geq 1$. Applying (7) over the entire horizon N results in a conservative control law similar to RMPC. The one-step chance constraint is given by

$$\mathbb{P}(\|\mathbf{y}_1 - \mathbf{y}_{r,1}\|_2 < c_1) \leq \beta_1, \quad (11)$$

where we define

$$p_1(\mathbf{u}_0) := \mathbb{P}(\|\mathbf{y}_1 - \mathbf{y}_{r,1}\|_2 < c_1). \quad (12)$$

Only the general MPC problem (5) is addressed in Assumptions 2 and 3. The norm constraint (7) is not considered, as it is specifically addressed in the method presented in Sections 3 and 4.

Remark 2. The norm constraint (7) is only considered in the first step, that is, at step $j = 1$, as we later minimize the probability of constraint violation for the first step. However, if this norm constraint is required to be considered at future steps $j \in \mathbb{I}_{2:N}$, this can be achieved by treating (7) as a chance constraint, similar to (11), resulting in

$$\mathbb{P}(\|\mathbf{y}_j - \mathbf{y}_{r,j}\|_2 < c_j) \leq \beta_j, \quad j \in \mathbb{I}_{2:N}. \quad (13)$$

This chance constraint (13) is then added to (5) and subsequently needs to be considered in (6). Assumptions 2 and 3 still need to be fulfilled if chance constraints are included for $j \in \mathbb{I}_{2:N}$ within the optimization problem.

2.4 | Problem statement

Instead of only bounding the chance constraint (11) by the risk parameter β_1 , we aim at minimizing the constraint violation probability p_1 within the MPC optimization problem. The challenge is to solve the MPC problem

$$V^* = \min_{\mathbf{U}_0} V(\mathbf{x}_0, \mathbf{U}_0), \quad (14a)$$

$$\text{s.t. } \mathbf{x}_{j+1} = \mathbf{A}\mathbf{x}_j + \mathbf{B}\mathbf{u}_j, \quad j \in \mathbb{I}_{0:N-1} \quad (14b)$$

$$\mathbf{y}_j = \mathbf{C}\mathbf{x}_j, \quad j \in \mathbb{I}_{0:N} \quad (14c)$$

$$\mathbf{u}_j \in \mathbb{U}_j, \quad j \in \mathbb{I}_{0:N-1} \quad (14d)$$

while it needs to be guaranteed that

$$\mathbf{u}_0 = \arg \min_{\mathbf{u}_0 \in \mathbb{U}_{x_0}} \mathbb{P}(\|\mathbf{y}_1 - \mathbf{y}_{r,1}\|_2 < c_1), \quad (15)$$

and that the MPC problem remains recursively feasible. Here, (14d) summarizes the constraints of the initial MPC problem (5), according to the definition of \mathbb{U}_j in (6), and (15) ensures that p_1 is minimized.

Besides minimizing constraint violation probability, this problem formulation allows to derive a method that also handles a further important challenge in safety-critical applications: maintaining recursive feasibility of the MPC optimal control problem for an unexpectedly increasing uncertainty support, which is problematic to ensure when using SMPC and RMPC.

In this paper, we propose a novel MPC approach, CVPM-MPC, that first ensures the minimal constraint violation probability p_1 , but then still optimizes the cost function $V(\mathbf{x}_0, \mathbf{U}_0)$. This approach yields a control input resulting in the lowest

possible constraint violation probability, given input and state constraints, while still optimizing further objectives. The CVPM-MPC method guarantees recursive feasibility, also for a changing uncertainty support, and ensures convergence of the MPC algorithm.

3 | METHOD

In this section, we derive the CVPM-MPC method to minimize the constraint violation probability p_j for the first predicted step $j = 1$ in an MPC problem. First, a general approach is presented to find a tightened admissible input set that minimizes the first step constraint violation probability. In the following part, it is shown how this approach can be incorporated into MPC. A visualization of the method is displayed in Figure 1. As determining the tightened input set within the CVPM-MPC method is difficult in general, we then provide an alternative, computable approach, assuming an uncertainty with a symmetric, unimodal probability density function (PDF). A solution approach for a multistep CVPM-MPC is described in Appendix B.

3.1 | General method to minimize constraint violation probability for the one-step problem

When minimizing p_1 over \mathbf{u}_0 within the MPC algorithm, three different cases need to be considered. In each case a set $\mathbb{U}_{\text{cvpm},0}$ is determined, which consists of inputs \mathbf{u}_0 that minimize the constraint violation probability. Ideally, even considering the bounded uncertainty, satisfaction of the constraint may be guaranteed in the next step, for all choices of $\mathbf{u}_0 \in \mathbb{U}_0$, which will be referred to as case 1. However, for stochastic systems we potentially have the situation that case 1 cannot be guaranteed. Here, two cases need to be distinguished. First, given the uncertainty, there is no choice for \mathbf{u}_0 that guarantees constraint satisfaction (case 2). Second, some choices for \mathbf{u}_0 guarantee constraint satisfaction, whereas other choices do not lead to such a guarantee (case 3). Depending on the case, $\mathbb{U}_{\text{cvpm},0}$ is determined differently as described in the following.

Case 1 (Guaranteed constraint satisfaction)

The probability of violating the norm constraint is zero, independent of the choice for \mathbf{u}_0 , that is,

$$p_1(\mathbf{u}_0) = 0 \quad \forall \mathbf{u}_0 \in \mathbb{U}_0. \quad (16)$$

Therefore, every $\mathbf{u}_0 \in \mathbb{U}_0$ is a valid input, resulting in

$$\mathbb{U}_{\text{cvpm},0} = \mathbb{U}_0. \quad (17)$$

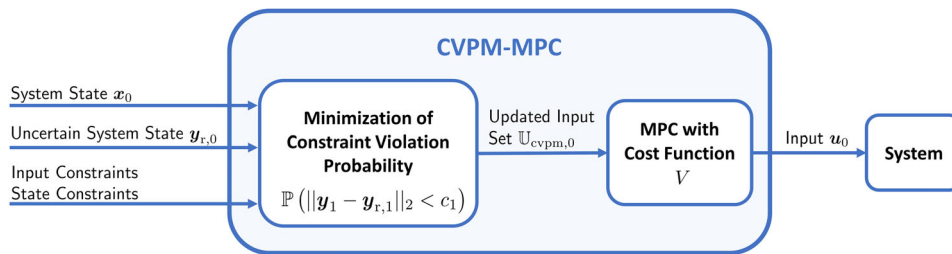


FIGURE 1 Visualization of a CVPM-MPC optimization: Given an input set and state constraints, as well as the current system state and uncertain system state, an updated input set is determined. This updated input set ensures a minimal norm constraint violation probability for the next step. Then, an MPC optimization problem is solved, optimizing for further objectives [Colour figure can be viewed at wileyonlinelibrary.com]

Case 2 (Impossible constraint satisfaction guarantee)

There is no choice for \mathbf{u}_0 such that constraint satisfaction can be guaranteed in the presence of uncertainty, that is,

$$p_1(\mathbf{u}_0) > 0 \quad \forall \mathbf{u}_0 \in \mathbb{U}_0. \quad (18)$$

As it is impossible to guarantee $p_1 = 0$, the aim is to minimize p_1 . Selecting

$$\mathbb{U}_{\text{cvpm},0} = \left\{ \mathbf{u}_0 \mid \mathbf{u}_0 = \arg \min_{\mathbf{u}_0 \in \mathbb{U}_0} p_1(\mathbf{u}_0) \right\} \quad (19)$$

yields the set $\mathbb{U}_{\text{cvpm},0}$, which only consists of inputs \mathbf{u}_0 that minimize p_1 .

Case 3 (Possible Constraint Satisfaction Guarantee)

If only some inputs \mathbf{u}_0 guarantee satisfaction of the norm constraint (7), that is,

$$\exists \mathbf{u}_0 \in \mathbb{U}_0 \text{ s.t. } p_1(\mathbf{u}_0) = 0, \quad (20)$$

then the set

$$\mathbb{U}_{\text{cvpm},0} = \{ \mathbf{u}_0 \mid (p_1(\mathbf{u}_0) = 0) \wedge (\mathbf{u}_0 \in \mathbb{U}_0) \}, \quad (21)$$

consists of those inputs that yield constraint satisfaction.

In all three cases $\mathbb{U}_{\text{cvpm},0}$ needs to be found, leading to the following strong assumption.

Assumption 5. The set $\mathbb{U}_{\text{cvpm},0}$ can be determined for all cases 1–3.

While it is possible to approximate $\mathbb{U}_{\text{cvpm},0}$ by sampling, finding an analytic solution for $\mathbb{U}_{\text{cvpm},0}$ highly depends on the probability distribution. However, if $\mathbb{U}_{\text{cvpm},0}$ can be determined, the CVPM-MPC method guarantees minimal constraint violation probability for p_1 .

Theorem 1. *If Assumption 5 holds, minimization of the constraint violation probability p_1 is guaranteed by selecting $\mathbb{U}_{\text{cvpm},0}$ according to cases 1–3.*

Proof. The proof is derived in Appendix A1. ■

In dynamic environments, the worst-case uncertainty $w_{\max,k}$ may change over time, which influences the probability of constraint violations. If the support changes, the CVPM-MPC approach still minimizes this constraint violation probability.

Corollary 1. *If the uncertainty support $\text{supp}(f_{w_k})$ changes from step k to $k+1$, the CVPM-MPC problem solved at step $k+1$ guarantees that the constraint violation probability p_{k+2} is minimized.*

Proof. The proof is derived in Appendix A2. ■

The MPC problem (14) is now adapted given the set $\mathbb{U}_{\text{cvpm},0}$ to guarantee minimal constraint violation probability of the norm constraint while still optimizing for further objectives.

3.2 | MPC with minimal first step constraint violation probability

Applying the previously determined $\mathbb{U}_{\text{cvpm},0}$ yields the CVPM-MPC problem

$$V^* = \min_{\mathbf{U}_0} V(\mathbf{x}_0, \mathbf{U}_0), \quad (22a)$$

$$\text{s.t. } \mathbf{x}_{j+1} = \mathbf{A}\mathbf{x}_j + \mathbf{B}\mathbf{u}_j, \quad j \in \mathbb{I}_{0:N-1} \quad (22b)$$

$$\mathbf{y}_j = \mathbf{C}\mathbf{x}_j, \quad j \in \mathbb{I}_{0:N} \quad (22c)$$

$$\mathbf{U}_0 \in \mathbb{U}_0^*. \quad (22d)$$

The set \mathbb{U}_0^* defines the admissible inputs, which yield minimal constraint violation probability combined with keeping the inputs and states within the input and state constraint sets. The set \mathbb{U}_0^* is given by

$$\mathbb{U}_0^* = \left\{ \mathbf{U}_0 \mid (\mathbf{u}_0 \in \mathbb{U}_{\text{cvpm},0}) \wedge (\mathbf{u}_j \in \mathbb{U}_{\mathbf{x},j}, j \in \mathbb{I}_{1:N-1}) \right\}, \quad (23)$$

where $\mathbb{U}_{\mathbf{x},j}$ is defined in (6) and $\mathbb{U}_{\text{cvpm},0}$ is obtained according to Section 3.1.

The complete CVPM-MPC problem (22) allows to optimize a cost function and satisfy state and input constraints, while minimization of the constraint violation probability p_1 is ensured.

3.3 | Minimal constraint violation probability for the one-step problem with symmetric unimodal PDF

The proposed CVMP-MPC method in Section 3.1 only guarantees minimal constraint violation probability if Assumption 5 is fulfilled. Therefore, it must be possible to always determine $\mathbb{U}_{\text{cvpm},0}$, which is a strong assumption. In the following, we provide an adapted approach of the CVMP-MPC method that guarantees minimal constraint violation probability if the PDF of the uncertainty is symmetric and unimodal.

In the following, we first give a definition for symmetric, unimodal probability density functions PDFs. Further, we introduce a substitute for the constraint violation probability p_k . Then, the three cases from Section 3.1 are adapted in order to minimize p_1 for the PDF addressed in the following. For each case a convex set of admissible inputs $\mathbb{U}_{\text{cvpm},0}$ is determined.

3.3.1 | Symmetric unimodal PDF

We first define the class of symmetric, unimodal probability distributions.

Definition 1 (Symmetric unimodal distribution). A probability distribution is symmetric and unimodal if its PDF has a single mode, that is, a single global maximum, which coincides with its mean $\boldsymbol{\mu}$ and

$$f(\boldsymbol{\mu} + \boldsymbol{\delta}_1) = f(\boldsymbol{\mu} + \boldsymbol{\delta}_2) \quad \forall \|\boldsymbol{\delta}_1\|_2 = \|\boldsymbol{\delta}_2\|_2. \quad (24)$$

With Definition 1 it is ensured that the PDF has its peak at mean $\boldsymbol{\mu}$ and that the PDF is strictly radially decreasing. As the probability distribution is symmetric, all realizations with similar distance to $\boldsymbol{\mu}$ have the same relative likelihood. Since there is only one global maximum of the PDF at $\boldsymbol{\mu}$, realizations with increasing distance to $\boldsymbol{\mu}$ have a lower relative likelihood.

The constraint violation probability p_k is a probabilistic expression and cannot directly be used in the optimization problem. The following assumption will allow to find a deterministic substitute for p_k .

Assumption 6 (Uncertainty with symmetric unimodal PDF). The PDF $f_{\mathbf{W}_k}$ for \mathbf{W}_k in (2) is symmetric and unimodal with mean $\boldsymbol{\mu} = \mathbf{0}$.

An example for an admissible probability distribution $p_{\mathbf{W}_k}$ with symmetric, unimodal PDF is a truncated isotropic bivariate normal distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with covariance matrix

$$\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \sigma_2^2) = \sigma^2 \mathbf{I}, \quad \sigma = \sigma_1 = \sigma_2, \quad (25)$$

with variance σ^2 and identity matrix \mathbf{I} . The support in each direction is required to be equal, which can be achieved by over-approximating. Distributions with $\sigma_1 \neq \sigma_2$ can be over-approximated by choosing

$$\Sigma = \sigma_{\max} \mathbf{I}, \quad \sigma_{\max} = \max(\sigma_1, \sigma_2). \quad (26)$$

We now address the relation between p_k and $f_{\mathbf{w}_k}$ considering Assumption 6. The following lemma shows that the constraint violation probability p_k can be decreased by choosing \mathbf{u}_{k-1} such that the distance is increased between the next system output \mathbf{y}_k and the next known, nominal random system output $\bar{\mathbf{y}}_{r,k}$.

Lemma 1. *If Assumption 6 holds, the probability p_k is decreasing for an increasing norm $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2$.*

Proof. The proof is derived in Appendix A3. ■

This lemma shows that the probability of violating the norm constraint (7) decreases if the difference between \mathbf{y}_k and $\bar{\mathbf{y}}_{r,k}$ increases. Lemma 1 now allows to find a substitute function for p_k .

3.3.2 | Substitute probability function

The probability p_j cannot be used directly to obtain the set $\mathbb{U}_{\text{cvpm},0}$. Therefore, a substitution is required for p_j . Based on Lemma 1, the probability p_j decreases for an increasing norm $\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2$. This property is used to choose a substitution for the constraint violation probability p_j . Here, the substitute function is selected to be

$$h(\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2) = \|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2^2. \quad (27)$$

While p_j is decreasing with the norm $\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2$, the function $h(\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2)$ is increasing with $\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2$. Therefore, increasing the value of function $h(\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2)$ yields a reduced probability p_j , which is exploited in the following to minimize constraint violation probability.

Remark 3. While $h(\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2) = \|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2^2$ is adequate for most safety-critical applications, other scalar functions $h'(\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2)$ are possible, as long as they are twice differentiable and strictly monotonically increasing.

Considering the constraint violation probability for the first predicted step $j = 1$, this probability p_1 is minimized for a maximal $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$. However, since $f_{\mathbf{w}_k}$ is truncated and $\text{supp}(p_k)$ is bounded, there potentially are multiple admissible inputs that result in an equal constraint violation probability. The aim is now to find the convex set $\mathbb{U}_{\text{cvpm},0}$ including all inputs $\mathbf{u}_{\text{cvpm},0} \in \mathbb{U}_{\text{cvpm},0}$ that result in a minimal p_1 . As $\mathbf{u}_{r,0}$ is deterministic and known according to Assumption 4, $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$ is a deterministic expression that can be evaluated.

The set $\mathbb{U}_{\text{cvpm},0}$ can then be found by comparing the worst-case uncertainty $w_{\max,0}$ with the minimum and maximum possible values of $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$, that is, $h_{\min,1}$ and $h_{\max,1}$, respectively. The maximal value $h_{\max,1}$ is determined by

$$h_{\max,1} := \max_{\mathbf{u}_0 \in \mathbb{U}_0} h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) = h\left(\max_{\mathbf{u}_0 \in \mathbb{U}_0} (\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)\right) \quad (28)$$

corresponding to the largest distance between \mathbf{y}_1 and $\bar{\mathbf{y}}_{r,1}$. Analogously, $h_{\min,1}$ can be found by

$$h_{\min,1} := \min_{\mathbf{u}_0 \in \mathbb{U}_0} h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) = h\left(\min_{\mathbf{u}_0 \in \mathbb{U}_0} (\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)\right). \quad (29)$$

The result for $h_{\min,1}$ can be obtained by determining the minimum value of $\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2$, as the objective function $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$ and \mathbb{U}_0 are convex. The following lemma provides a strategy to find $h_{\max,1}$.

Lemma 2. *Let the nonempty convex polytope $\mathcal{V} \subset \mathbb{R}^s$, $g \in \mathbb{N}$, be bounded by a finite set of hyperplanes, such that \mathcal{V} has a finite number of edge vertices with a convex function $z : \mathcal{V} \rightarrow \mathbb{R}$. Then, a global maximum*

$$z_{\max} = \max_{\mathbf{v} \in \mathcal{V}} z(\mathbf{v}) \quad (30)$$

is obtained by searching for the maximum value of z on the boundary $\partial\mathcal{V}$ of its domain \mathcal{V} .

Proof. The proof is derived in Appendix A4. ■

3.3.3 | Determination of the updated admissible input set

Similar to Section 3.1 we regard three cases. The resulting set $\mathbb{U}_{\text{cvpm},0}$, depending on the three cases, is then used in the CVPM-MPC problem (22) to guarantee minimal constraint violation probability of the norm constraint. In order to distinguish between the cases, we will consider the relation

$$\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2 \geq c_1 + w_{\max,0} \Rightarrow \|\mathbf{y}_1 - \mathbf{y}_{r,1}\|_2 \geq c_1, \quad (31)$$

which follows using a reverse triangle inequality, given the dynamics of the uncertain system (2). Here, $c_1 + w_{\max,0}$ represents the necessary distance between \mathbf{y}_1 and $\bar{\mathbf{y}}_{r,1}$, consisting of the required minimal distance c_1 at step $j = 1$ and the maximal random system step $w_{\max,0}$ at $j = 0$, such that $\|\mathbf{y}_1 - \mathbf{y}_{r,1}\|_2 \geq c_1$ for all $\|\mathbf{w}_0\|_2 \leq w_{\max,0}$.

Case 1 (Guaranteed constraint satisfaction)

For any $\mathbf{u}_0 \in \mathbb{U}_0$ constraint satisfaction is guaranteed, that is, $p_1 = 0$ for

$$h_{\min,1} \geq h(c_1 + w_{\max,0}). \quad (32)$$

The initial state configuration of the controlled and stochastic system is such that the minimum value possible for $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$, that is, $h_{\min,1}$, still yields a larger value than inserting c_1 combined with the worst-case $w_{\max,0}$ into h , which moves $\mathbf{y}_{r,1}$ closest to \mathbf{y}_1 . This results in a guaranteed constraint satisfaction $p_1 = 0$. Therefore, every $\mathbf{u}_0 \in \mathbb{U}_0$ is an admissible input, that is,

$$\mathbb{U}_{\text{cvpm},0} = \mathbb{U}_0. \quad (33)$$

Case 2 (Impossible constraint satisfaction guarantee)

There is no input $\mathbf{u}_0 \in \mathbb{U}_0$ that can guarantee $p_1 = 0$, that is,

$$h_{\max,1} < h(c_1 + w_{\max,0}). \quad (34)$$

The largest value for $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$ that can be achieved with $\mathbf{u}_0 \in \mathbb{U}_0$ is $h_{\max,1}$, corresponding to the lowest possible p_1 . However, to guarantee constraint satisfaction of (7), $h_{\max,1}$ is required to be larger or at least equal to $h(c_1 + w_{\max,0})$, with the worst-case absolute value $w_{\max,0}$ for the realization of \mathbf{w}_0 . Constraint satisfaction cannot be guaranteed here.

The solution corresponding to $h_{\max,1}$ is denoted by $\mathbf{u}_{\text{cvpm},0}$. Minimal p_1 is achieved with

$$\mathbf{u}_{\text{cvpm},0} = \arg \max_{\mathbf{u}_0 \in \mathbb{U}_0} h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2), \quad (35)$$

as $h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2)$ increases and p_1 decreases with an increasing norm.

Therefore,

$$\mathbb{U}_{\text{cvpm},0} = \{\mathbf{u}_{\text{cvpm},0}\}, \quad (36)$$

is selected since the input choice $\mathbf{u}_{\text{cvpm},0}$ guarantees the lowest constraint violation probability when $p_1 > 0$.

Remark 4. If (35) yields more than one solution, $\mathbb{U}_{\text{cvpm},0}$ in (36) may also consist of more than one element, that is, all solutions of (35). However, there can be restrictions if convexity of $\mathbb{U}_{\text{cvpm},0}$ is required.

Case 3 (Possible constraint satisfaction guarantee)

The final case yields $p_1 = 0$ for some \mathbf{u}_0 and applies if

$$h_{\max,1} \geq h(c_1 + w_{\max,0}) > h_{\min,1}. \quad (37)$$

While some $\mathbf{u}_0 \in \mathbb{U}_0$ cannot guarantee zero constraint violation probability, it is possible to find \mathbf{u}_0 such that

$$h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) \geq h(c_1 + w_{\max,0}). \quad (38)$$

Therefore, for some \mathbf{u}_0 constraint satisfaction can be guaranteed in the presence of uncertainty. Hence, the task is to find a set

$$\mathbb{U}_{\text{cvpm},0} = \left\{ \mathbf{u}_0 \mid (h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) \geq h(c_1 + w_{\max,0})) \wedge (\mathbf{u}_0 \in \mathbb{U}_0) \right\}, \quad (39)$$

which consists of all inputs $\mathbf{u}_0 \in \mathbb{U}_0$ that yield constraint satisfaction and therefore $p_1 = 0$.

The first part of the set in (39),

$$\mathbb{U}_{\text{mode3},0} = \left\{ \mathbf{u}_0 \mid h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) \geq h(c_1 + w_{\max,0}) \right\}, \quad (40)$$

describes a super-level set, including only inputs \mathbf{u}_0 that lead to $p_1 = 0$. This super-level set is generally non-convex. In order to receive a convex set $\mathbb{U}_{\text{cvpm},0}$ for the optimization problem, an approximation is performed, based on the boundary

$$\partial\mathbb{U}_{\text{mode3},0} = \left\{ \mathbf{u}_0 \mid h(\|\mathbf{y}_1 - \bar{\mathbf{y}}_{r,1}\|_2) = h(c_1 + w_{\max,0}) \right\}. \quad (41)$$

Proposition 1. *An approximated, convex solution of (39) in case 3 is obtained by*

$$\mathbb{U}_{\text{cvpm},0} = \hat{\mathbb{U}}_{\text{cvpm},0} = \left\{ \mathbf{u}_0 \mid \hat{\mathbb{U}}_0(\mathbf{u}_0^*) \cap \mathbb{U}_0 \right\}, \quad (42)$$

with

$$\hat{\mathbb{U}}_0(\mathbf{u}_0^*) = \left\{ \mathbf{u}_0 \mid \left(\nabla_{\mathbf{u}_0^*} h(\|\mathbf{y}_1(\mathbf{u}_0^*) - \bar{\mathbf{y}}_{r,1}\|_2) \right)^\top (\mathbf{u}_0 - \mathbf{u}_0^*) \geq 0 \right\}, \quad (43)$$

the gradient operator $\nabla_{\mathbf{u}_0^*}$, and a point $\mathbf{u}_0^* \in \partial\mathbb{U}_{\text{mode3},0} \cap \mathbb{U}_0$, which is an admissible input.

Remark 5. While it was previously not explicitly stated that \mathbf{y}_1 depends on \mathbf{u}_0 , the dependence of \mathbf{y}_1 on \mathbf{u}_0^* is stated for clarity in Proposition 1.

Proof. The proof is derived in Appendix A5. ■

An approach to finding \mathbf{u}_0^* is solving the system

$$h(\|\mathbf{y}_1(\mathbf{u}_0^*) - \bar{\mathbf{y}}_{r,1}\|_2) = h(c_1 + w_{\max,0}) \quad (44)$$

with $\mathbf{u}_0^* \in \mathbb{U}_0$. The choice of \mathbf{u}_0^* is not unique. It is possible that $\hat{\mathbb{U}}_{\text{cvpm},0}$ is empty due to approximating even though case 3 applies.

Remark 6. If $\hat{\mathbb{U}}_{\text{cvpm},0} = \emptyset$ in case 3, then \mathbf{u}_0 can be determined by following the procedure of case 2.

Following the approach in Remark 6 still provides a solution that minimizes p_1 . However, in case 2 only a single option $\mathbb{U}_{\text{cvpm},0} = \mathbf{u}_{\text{cvpm},0}$ is given, while case 3 has the advantage of providing a set $\mathbb{U}_{\text{cvpm},0}$ with multiple possible inputs \mathbf{u}_0 . Case 3 therefore offers the possibility to then optimize to account for further objectives, given the set of admissible inputs $\mathbb{U}_{\text{cvpm},0}$.

4 | PROPERTIES

In the following, two important properties are analyzed. First, recursive feasibility of the proposed method is shown. This is followed by a proof of convergence.

4.1 | Recursive feasibility

Recursive feasibility guarantees that if the MPC optimization problem is solvable at step k , it is also solvable at step $k + 1$. This needs to hold as MPC requires the solution of an optimal control problem at every time step.

Definition 2. (Recursive feasibility) Recursive feasibility of an MPC algorithm is guaranteed if

$$\mathbb{U}_k^N \neq \emptyset \Rightarrow \mathbb{U}_{k+1}^N \neq \emptyset \quad (45)$$

where \mathbb{U}_k^N is the set of admissible inputs \mathbf{U}_k to fulfill the constraint (22d) from step k to step $k + N$.

In the following, recursive feasibility will be established for the proposed method.

Theorem 2. *The CVPM-MPC algorithm in (22) is recursively feasible with the general CVPM approach of Section 3.1.*

The proof is divided into two parts. First, it is shown that $\mathbb{U}_{\text{cvpm},0} \neq \emptyset$ at any step, which is different to standard MPC methods. Then, recursive feasibility of the optimization problem (22) is shown.

Proof. The proof is derived in Appendix A6. ■

The proof for the general CVPM-MPC method can be extended for the CVPM-MPC approach for uncertainties with symmetric, unimodal PDFs in Section 3.3.

Corollary 2. *If Assumption 6 holds, the CVPM-MPC algorithm in (22) is recursively feasible with the CVPM approach of Section 3.3.*

Proof. The proof is derived in Appendix A7. ■

Theorem 2 and Corollary 2 show that if the MPC problem (5) is designed to be recursively feasible, the CVPM-MPC algorithm (22), based on (5), remains recursively feasible. According to Corollary 1, minimizing p_k is independent of the uncertainty support, therefore, recursive feasibility is guaranteed if the uncertainty support changes.

4.2 | Convergence

In the following, convergence of the proposed method is shown. In this section, the MPC optimization starts at \mathbf{x}_k . While it is possible to track a reference varying from the origin, here, without loss of generality, we will only consider the regulation of the origin.

The uncertain output $\mathbf{y}_{r,k}$ can potentially lie close to the origin or even directly in the origin. In order to minimize p_k , an area around $\mathbf{y}_{r,k}$ is then inadmissible for the system output \mathbf{y}_k . This can lead to the case where the origin is inadmissible for the controlled system, that is, $\mathbf{0} \in \mathcal{X}_{\text{cv},k}$, where

$$\mathcal{X}_{\text{cv},k} = \left\{ \mathbf{x}_k \mid p_k(\mathbf{u}_{k-1}) > 0, \mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \right\}, \quad (46)$$

denotes the bounded and open set of states \mathbf{x}_k with $p_k > 0$, that is, constraint violation is possible for all $\mathbf{x}_k \in \mathcal{X}_{cv,k}$. An inadmissible origin, of course, is an issue when investigating the stability of the proposed algorithm. However, we will provide a convergence guarantee under the following two assumptions concerning the stochastic nature of $\mathbf{y}_{r,k}$.

Assumption 7 (Admissible origin). (a) There exists a $k_0 < \infty$ such that for all $k \geq k_0$ it holds that

$$\mathbf{0} \notin \mathcal{X}_{cv,k} \quad \forall k \geq k_0. \quad (47)$$

(b) There exists a $k_{y0} < \infty$ and a finite sequence of inputs \mathbf{u}_k such that $\mathbf{y}_k = \mathbf{0}$ for all $k \geq k_{y0} \geq k_0$.

(c) There exists a $k_{\text{case1,3}} < \infty$ and for all $k \geq k_{\text{case1,3}} \geq k_0$

$$\exists \mathbf{u}_{k-1} \text{ s.t. } p_k(\mathbf{u}_{k-1}) = 0, \quad (48)$$

and $\cup_{\text{cvpm},k} \neq \emptyset$.

Assumption 7 (a) ensures that even if $\mathbf{y}_{r,k}$ is occupying the space around the origin for some time, eventually $\mathbf{y}_{r,k}$ will be distanced enough that the origin becomes admissible for the controlled system, as the boundedness of the stochastic system state yields a closed admissible space for the controlled system. Assumption 7 (b) ensures that there is a possibility for the controlled system to reach the origin.

With Assumption 7 (c) it is guaranteed that either case 1 or case 3 is applicable if Assumption 7 (a) holds. This ensures that $p_k = 0$ at some time after the origin becomes admissible for the controlled system.

Lemma 3. *If Assumption 7 holds, there exists a closed, control invariant set $\tilde{\mathcal{X}}_k = \mathcal{X} \setminus \mathcal{X}_{cv,k}$ for $k \geq k_{\text{case1,3}}$, which contains the origin.*

Proof. The proof is derived in Appendix A8. ■

The set $\tilde{\mathcal{X}}$ consists of the states that ensure constraint satisfaction of \mathcal{X} and yield $p_k = 0$ for $k \geq k_{\text{case1,3}}$.

Assumption 8 (Terminal constraint set). The terminal constraint set \mathcal{X}_f is a subset of $\tilde{\mathcal{X}}_k$, that is, $\mathcal{X}_f \subset \tilde{\mathcal{X}}_k$.

In the following, convergence of the proposed method is addressed.

Theorem 3. *If Assumptions 3 and 7 hold, the proposed CVPM-MPC method in Section 3.1 satisfies that \mathbf{x}_k converges to $\mathbf{0}$ for $k \rightarrow \infty$.*

Proof. The proof is derived in Appendix A9. ■

In Theorem 3 it is only shown that the system converges to the origin once the random system fulfills Assumption 7. However, every time the stochastic output allows the system to reach the origin, the system will move toward the origin. The system state \mathbf{x}_k remains at $\mathbf{0}$ until $\mathbf{y}_{r,k}$ moves in such a way that the origin has nonzero constraint violation probability. As the main goal is to ensure minimum constraint violation probability of (8), \mathbf{y}_k will move away from the origin to minimize p_k if $\mathbf{y}_{r,k}$ behaves in such a way that it causes $p_k > 0$ in the origin.

Corollary 3. *If Assumptions 7 holds, the proposed CVPM-MPC method in Section 3.3 for uncertainties with symmetric, unimodal PDFs satisfies that $\mathbf{x}_k \in \mathcal{X}$ for all k and that \mathbf{x}_k converges to $\mathbf{0}$ for $k \rightarrow \infty$.*

Proof. The proof is derived in Appendix A10. ■

Therefore, if the origin is admissible, the controlled system will converge. However, satisfying the norm constraint has priority over converging to the origin. Compared to standard MPC methods, the origin is not necessarily within the constraint set \mathcal{X} . Therefore, the standard MPC stability approach cannot be applied, but convergence is proved as in Theorem 3.

5 | DISCUSSION OF THE PROPOSED CVPM-MPC METHOD

One could argue now that the proposed algorithm is a combination of RMPC in the first step and, potentially, SMPC in the following steps. While there are some similarities to this combination, we solve a different problem. The most important difference is that the constraint violation probability is minimized in the first predicted step and the initial uncertainty probability is not required to be zero. RMPC approaches require constraint satisfaction initially and ensure that constraints are satisfied throughout the prediction horizon.

Our proposed CVPM-MPC method is more closely related to SMPC than RMPC, as constraint violations are possible. Nevertheless, the suggested method can be interpreted as lying between SMPC and RMPC. The results are more conservative than SMPC, as a zero percent constraint violation probability is found if possible, that is, $p_k = 0$ in (8), but less conservative than RMPC. An advantage over both, SMPC and RMPC, is the ability to minimize the constraint violation probability and to successfully cope with sudden uncertainty support changes, as recursive feasibility can still be guaranteed. The uncertainty support can change due to unexpected events or modeling inaccuracies. Presuming and considering this potential change in uncertainty support is reasonable if the system is subject to epistemic uncertainty. Epistemic uncertainty refers to uncertainty that could theoretically be known but, in practice, is not known precisely, for example due to approximations in modelling, insufficient data, or challenging behavior prediction. Accounting for epistemic uncertainty is an investigated topic in machine learning literature.^{37,38}

In SMPC with chance constraints, recursive feasibility is a major issue. For example, an unexpected realization of the uncertainty at step k , where the uncertainty realization likelihood lies below the chance constraint risk parameter at step k , leads to a state at step $k + 1$ with no solution to the optimization problem if the required risk parameter of the chance constraint cannot be met. An option to regain feasibility is to solve an alternative optimization problem or apply an input that was previously defined. However, these alternatives do not necessarily lead to a solution that yields the lowest constraint violation probability. Furthermore, it is possible to soften chance constraints by using slack variables in the cost function. However, this approach is not acceptable in applications where the chance constraint represents a safety constraint. If a slack variable is introduced, it competes with other objectives within the cost function and does not ensure constraint satisfaction. The proposed CVPM-MPC method always finds the optimal input that results in the lowest constraint violation probability while remaining recursively feasible.

RMPC guarantees recursive feasibility but at the cost of reduced efficiency, as worst-case scenarios need to be taken into account. Additionally, if the support of the uncertainty can suddenly change over time, for example, the future motion of an object becomes more uncertain due to a changing environment, RMPC can become too conservative to be applicable. A robust solution can only be obtained by always considering the largest possible uncertainty support. The proposed method deals with this by adjusting to changing uncertainty supports at every step, as will be illustrated in Section 6. A suddenly or unexpectedly increasing uncertainty support, for example, due to an inaccurate prediction model, may lead to increased constraint violation probability for a limited time after the support changes. Before the support changes, the optimized inputs of the proposed algorithm lead to a less conservative result than RMPC while ensuring that the constraint violation probability is kept at a minimal level immediately after the change.

In the proposed method, we only consider minimizing the constraint violation for the first predicted step. It is possible to consider multiple steps by increasing the uncertainty support for each considered step as described in Appendix B, however, this leads to a more conservative solution. For every additional predicted step in which the constraint violation probability is minimized, the maximal possible uncertainty value must be considered. This yields a highly restrictive set of admissible inputs, which minimize the constraint violation probability over multiple predicted steps. As it is assumed that the support of the uncertainty PDF can change over time, considering multiple steps with the initially known support does not guarantee lower constraint violation probability for multiple steps. If the support increases, the previously obtained multi-step CVPM-MPC solution becomes invalid. Therefore, given an updated uncertainty support at each step, it is a reasonable approach to only minimize the constraint violation probability for the first predicted step, resulting in the safest solution at the current step. It is possible to consider the norm constraint for collision avoidance in multiple predicted steps by either formulating a chance constraint, as mentioned in Remark 2, or a robust constraint. However, this can result in infeasibility of the optimization problem, particularly if the uncertainty support varies over time. Despite only considering the norm constraint for the next predicted step, it is still beneficial to use an MPC horizon $N > 1$. Other objectives are optimized over the entire horizon, given that the first input is included in the set $\mathcal{U}_{\text{cvpm},0}$, which potentially consists of multiple admissible inputs that all minimize the constraint violation for the next step.

Applying the CVPM-MPC approach possibly results in oscillating behavior. As long as case 1 is valid, the proposed method does not affect the optimization, as $\mathbb{U}_{\text{cvpm},0} = \mathbb{U}_0$. Once case 2 is active, a solution is found that minimizes the probability of constraint violation, ignoring the reference and potentially moving from the reference, as only one input is admissible. When case 1 is valid again, the optimized the reference is tracked again until, possibly, case 2 becomes active again. This can be improved by considering the norm constraint as a chance constraint for multiple predicted steps, however, recursive feasibility is not guaranteed, as mentioned before.

The main focus of the suggested method is to minimize the constraint violation probability. It is clear that stability cannot always be guaranteed, as the origin can be excluded from the admissible state set. We consider a narrow road where a bicycle is located between the controlled vehicle and the vehicle reference point. If the road is too narrow for the vehicle to pass, it will remain behind the bicycle and never reach the reference point, that is, Assumption 7 (b) is violated. However, Assumption 7 implies that the origin is not inadmissible at all times, and once the origin is admissible, the controlled system converges.

It is also important to note that minimizing the constraint violation probability has priority over other optimization objectives. Especially in safety-critical applications, this can be of major interest, for example, an autonomous car must ensure that the collision probability is always minimal, prior to reducing energy or increasing passenger comfort. If SMPC were to be applied in such scenarios, the question would arise of how to choose the SMPC risk parameter β_k . A large β_k yields efficient behavior but might be unacceptable due to an insufficient safety level. Finding a reduced value for β_k in SMPC is challenging, as even very small risk parameters allow for constraint violations, while $\beta_k = 0$ does not yield a chance constraint and the advantages of SMPC are lost. In the proposed CVPM-MPC method, the task of appropriately choosing a risk parameter is not required.

Additionally, in safety-critical systems a further aspect reduces the usability of chance constraints in SMPC. A solution is valid as long as the probability of violating the safety constraint satisfies the risk parameter. Assuming there exists a solution with lower, or even zero percent, constraint violation probability, the optimization solution will still be “on the chance constraint” if this results in lower objective costs. The allowed constraint violation then depends on the risk parameter. We consider again the example in the introduction of a car overtaking a bicycle. Using a chance constraint with $\beta_k > 0$, the car will pass the bicycle but will choose a trajectory around the bicycle that allows a collision with a low probability due to $\beta_k > 0$. Given a finite bicycle uncertainty support, passing the bicycle with slightly more distance yields zero collision probability with only a small increase of cost. However, in practice, this slightly increased cost is acceptable if thereby safety is guaranteed.

For the approach in Section 3.3 the PDF $f_{\mathbf{w}_k}$ does not need to be known exactly as long as it fulfills Assumption 6. If $f_{\mathbf{w}_k}$ is symmetric and unimodal, it is ensured that increasing $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2$ results in a lower constraint violation probability p_k .

The proposed method is especially useful in collision avoidance applications, which are either in two- or three-dimensional space. While applying the proposed method in 2D is straightforward, 3D applications can be more challenging to solve, especially finding \mathbf{u}_0^* in (42). For collision avoidance scenarios, possible uncertainty in (1) can be considered by increased uncertainty in (2).

The structure of CVPM-MPC considers two main aims: minimizing constraint violation probability and optimizing additional objectives, such as energy consumption. Multi-objective MPC^{39,40} is an MPC scheme to trade-off multiple opposing objectives within one optimal control problem, where each individual objective is assigned a weighting factor. In our case, we do not aim at a trade-off, but one of the two objectives is regarded first. Only if optimizing the first objective, that is, minimizing constraint violation probability, allows multiple possible inputs, further objectives are taken into account. In multi-objective MPC this translates to assigning a weight of zero to all additional objectives. While this could be an approach to minimize constraint violation probability, other objectives would not be optimized. Therefore, compared to multi-objective MPC, the proposed CVPM-MPC method provides a solution for safety-critical systems where constraint violation probability needs to be minimized and further objectives should be optimized if possible.

Applying CVPM-MPC to safety-critical applications, specifically collision avoidance for automated vehicles or robots, only requires defining a simple norm constraint for the obstacle to be avoided. For example, in the case of vehicles, the norm constraint represents the distance to other surrounding vehicles, whereas for robots the norm constraint represents the distance between the end effector and an object or human. Especially for robots it needs to be considered that, depending on the robot, other robot links must also avoid collisions. The simulation described in the following section may serve as an example of how to use CVPM-MPC in automated driving.

6 | SIMULATION RESULTS

In the following, a simulation is presented and discussed to further explain the general idea and its application. This collision avoidance scenario with two vehicles illustrates an application where the proposed method is beneficial. The simulations were run in MATLAB on a standard desktop computer using MPT3⁴¹ and YALMIP.⁴² Solving a single optimization of the MPC algorithm takes 54 ms on average. All quantities are given in SI units.

6.1 | Collision avoidance simulation

A collision occurs if the distance between two objects becomes too small. This distance can be represented by a norm constraint. The priority is then enforcing the norm constraint, or if not feasible, minimizing the probability of violating the norm constraint.

We consider the example mentioned in Sections 1 and 5 where a controlled vehicle avoids collision with a bicycle, referred to as obstacle in the following. The controlled vehicle is approximated by a circle with radius $r_c = 2.0$ and the obstacle is approximated by a circle with radius $r_r = 0.8$ and is subject to stochastic motion in a bounded area, for example, a road. The circles are chosen to fully cover the individual shapes of the controlled vehicle and obstacle. The scenario setup is shown in Figure 2. The continuous system dynamics of the controlled vehicle in x and y direction are given by

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (49)$$

where $\mathbf{x} = [x, y]^T$ and $[v_x, v_y]^T$ are the position and velocity in a 2D environment, respectively. The inputs are given by $[u_1, u_2]^T$. Using zero-order hold with sampling time $\Delta t = 0.1$ yields the discretized system

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} e^{\Delta t} - 1 & 0 \\ 0 & e^{\Delta t} - 1 \end{bmatrix} \mathbf{u}_k, \quad (50a)$$

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}_k \quad (50b)$$

which is similar to (1). We will consider the input constraints

$$\mathcal{U} = \left\{ \mathbf{u} = \begin{bmatrix} v_x \\ v_y \end{bmatrix} \mid 1 \leq v_x \leq 9, |v_y| \leq 3.5 \right\}. \quad (51)$$

In x -direction there exists a minimum velocity $v_{x,\min} = 1$ to ensure that the controlled vehicle is always moving forward, which also limits the potential oscillating behavior due to the CVPM-MPC approach. We also consider the state constraints

$$\mathcal{X} = \left\{ \mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} \mid y_{\text{lb}} \leq y \leq y_{\text{ub}} \right\}, \quad (52)$$

where $y_{\text{lb}} = 2.0$ and $y_{\text{ub}} = 8.0$ are the boundaries of the road minus the radius r_c .

The assumed behavior of the obstacle with random behavior is given by

$$\mathbf{y}_{r,k} = \mathbf{y}_{r,0} + \sum_{i=0}^{k-1} (\mathbf{u}_{r,i} + \mathbf{w}_i) \quad (53)$$

depending on the initial output $\mathbf{y}_{r,0}$, the input $\mathbf{u}_{r,k}$, and the realization \mathbf{w}_k of the random variable $\mathbf{W}_k \sim f_{\mathbf{W}_k}$ and $\mathbf{y}_r = [x_r, y_r]^T$. We assume $f_{\mathbf{W}_k}$ to be symmetric, unimodal, and truncated, resulting in the support of $f_{\mathbf{W}_k}$

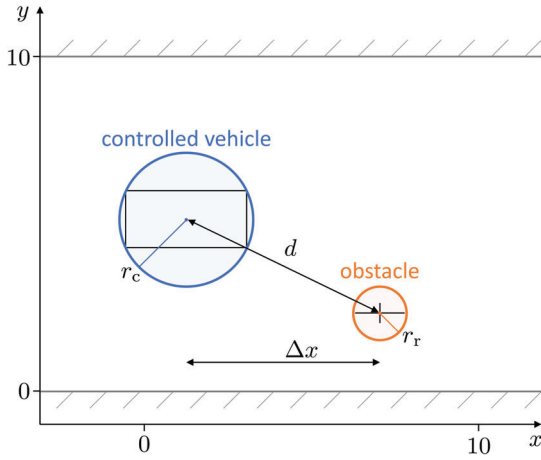


FIGURE 2 Vehicle avoidance scenario. Approximated shapes of the controlled vehicle (car) and obstacle (bicycle) are indicated by black lines within the objects [Colour figure can be viewed at wileyonlinelibrary.com]

$$\text{supp} (f_{\mathbf{w}_k}) = \{ \mathbf{w}_k \mid \|\mathbf{w}_k\|_2 \leq w_{\max,k} \}, \quad (54)$$

where $w_{\max,k}$ is the radius of the support boundary of \mathbf{W}_k . The physical interpretation of $w_{\max,k}$ is that it is the maximum uncertain distance the obstacle can move in one step, additionally to the deterministic distance $\mathbf{u}_{r,k}$. At step k , the controlled vehicle knows the obstacle position $\mathbf{y}_{r,k}$ and deterministic input $\mathbf{u}_{r,k}$, but \mathbf{w}_k is unknown. The deterministic input represents the forward motion of the dynamic obstacle, the random variable denotes the uncertainty within the forward motion. Without the deterministic input, the dynamic obstacle would only move around its initial position $\mathbf{y}_{r,0}$ due to the zero-mean random variable. The combination of a deterministic input with the random variable with zero mean is similar to a random variable with nonzero mean, where the controlled vehicle knows the mean. In general, the exact PDF is not required to be known. The proposed CVPM-MPC approach is applicable as long as the actual PDF adheres to Assumption 6 and the uncertainty bounds are known.

As the main aim of this simulation is to minimize the collision probability, an expression for this probability is necessary in order to analyze the simulation results. The collision probability at step k between the two vehicles will be denoted by $p_{\text{col},k}$ and it has finite support as $f_{\mathbf{w}_k}$ is truncated. In this example, a norm constraint is used to avoid a collision, that is, the norm constraint violation probability is minimized. Therefore, the probability of a collision $p_{\text{col},k}$ is defined analogous to p_k in Section 2. The derivation and expression for the collision probability $p_{\text{col},k}$ is omitted here due to readability. Details can be found in Appendix C.

The collision probability $p_{\text{col},k}$ depends on the Euclidean distance

$$d_k = \|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2, \quad (55)$$

between the controlled vehicle and obstacle. Similar to (7), a norm constraint can be formulated where $c_k = d_{\text{safe},k}$ may be interpreted as the minimal distance between the controlled vehicle and the obstacle such that a collision is avoided. The support of $p_{\text{col},k}$ results from adding the radius of the controlled vehicle and the obstacle to $\text{supp} (f_{\mathbf{w}_k})$ that is,

$$\text{supp} (p_{\text{col},k}) = \{ \mathbf{y}_k \mid d_k \leq d_{\text{safe},k} \}, \quad (56)$$

where $d_{\text{safe},k} = w_{\max,k-1} + r_r + r_c$ is the safety distance required to avoid a collision between the controlled vehicle and the obstacle, taking into account the radius of both vehicles, r_r and r_c , and the maximal obstacle step $w_{\max,k-1}$. Similar to Lemma 1 for p_k , $p_{\text{col},k}$ is decreasing for increasing d_k .

As proposed in Section 3.3.2, we choose $h(\xi) = \xi^2$, which is strictly monotonically increasing with ξ . This yields

$$h(\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2) = \|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2^2, \quad (57)$$

which can be considered a substitution for the probability function $p_{\text{col},k}$.

The controlled vehicle uses the CVPM-MPC algorithm (22) with $N = 10$ and

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}. \quad (58)$$

The x -position references for the controlled vehicle are obtained by $x_{\text{ref},k} = x_0 + v_{x,\text{ref}}k\Delta t$, where $v_{x,\text{ref}}$ is the reference velocity in x -direction.

In the following, two scenarios will be analyzed. In the first scenario, the controlled vehicle is located close to its state boundary, that is, the road boundary, showing that the norm constraint is minimized in the presence of state constraints. In the second scenario, the obstacle uncertainty support will suddenly increase. The orientation ϕ of the controlled vehicle in Figures 3 and 5 is approximated by

$$\phi = \arctan \frac{u_2}{u_1}. \quad (59)$$

6.1.1 | Active state constraint

In the first simulation, it is shown that the proposed method is applicable if state constraints are active. The reference velocity and y -position for the controlled vehicle are set to $v_{x,\text{ref}} = 5.0$ and $y_{\text{ref}} = 8.0$, respectively, with initial position $\mathbf{y}_0 = [0, 8]^\top$. The obstacle motion consists of a deterministic part $\mathbf{u}_{r,k} = [0.5, 0]^\top$ combined with random behavior subject to a Gaussian uncertainty with $w_{\text{max},k} = 0.15$, with mean y -position $y_r = 4.0$ and a mean x -velocity $v_{r,x} = 5.0$. Therefore, the x -position reference of the controlled vehicle is the same as the mean x -position of the obstacle in every step. Here, a sine motion is applied to the y -position of the obstacle with constant x -velocity, which is one possible outcome given the Gaussian uncertainty. The sine motion ensures that the maximal uncertainty values appear in the simulation, while the constant obstacle x -velocity keeps the controlled vehicle and the obstacle close together.

The vehicle configurations at different time steps are shown in Figure 3 and the results of the simulation are displayed in Figure 4. Initially, the controlled vehicle and obstacle have the same x -position. Starting at $t = 3.9$ s, the controlled vehicle needs to slow down to maintain a safe distance to the obstacle. As the maximal obstacle uncertainty is known by the controlled vehicle, the collision probability is kept at zero. After $t = 4.5$ s, the obstacle moves away from the controlled vehicle, resulting in increased input u_1 in order to get closer to the x -position reference. At $t = 5.0$ s, the controlled vehicle catches up with its x -position reference, which is then followed by constant inputs. Between $t = 9.0$ s and $t = 11.0$ s, similar behavior can be observed. It can be seen that the CVPM-MPC ensures $p_k = 0$ with active state constraints. As mentioned in Section 4, the motion of the obstacle can result in an inadmissible origin, i.e., Assumption 7 (c) is violated and the controlled vehicle cannot keep its reference velocity. However, as shown in Theorem 3, once the obstacle moves away, the velocity of the controlled vehicle again reaches the reference velocity.

This first simulation scenario is also used for a Monte Carlo simulation with 2000 simulation runs to evaluate the effectiveness of constraint violation probability minimization. Instead of applying the deterministic sine motion to the obstacle y -position and keeping the x -velocity constant, a random step is applied to the obstacle in addition to the deterministic part $\mathbf{u}_{r,k} = [0.5, 0]^\top$. This random step is based on a truncated bivariate normal distribution with covariance matrix $\Sigma = \text{diag}(\sigma^2, \sigma^2)$, $\sigma = 0.05$, mean $\mathbf{0}$, and $w_{\text{max},k} = 0.15$. The covariance matrix was chosen in such a way that truncating the normal distribution does not have a large effect, that is, for the nontruncated normal distribution the probability of $\|\mathbf{w}_k\|_2 > w_{\text{max},k}$ is less than 1%. The results underline the effectiveness of the CVPM-MPC method. In 98.8% of the simulations the constraint violation probability remained at $p_{\text{col},k} = 0$. The maximal constraint violation observed was $p_{\text{col},k} = 0.21\%$. No collisions occurred in any of the 2000 simulation runs.

6.1.2 | Change of uncertainty support

In the second simulation, we show that the proposed method is capable of dealing with varying uncertainty support of the obstacle. The controlled vehicle aims to obtain the reference velocity $v_{x,\text{ref}} = 4.0$ while maintaining $y_{\text{ref}} = 4.0$ with the initial position $\mathbf{y}_0 = [0, 4]^\top$. The obstacle moves with a constant input $\mathbf{u}_{r,k} = [0.25, 0]^\top$ at $y_r = 4.0$. We consider here that the obstacle uncertainty support suddenly changes, for example due to a changing environment. At first, the expected uncertainty support is $w_{\text{max},k} = 0.15$ and at $t = 2.0$ s it changes to $w_{\text{max},k} = 0.9$, while returning to $w_{\text{max},k} = 0.15$ at $t = 4.0$ s.

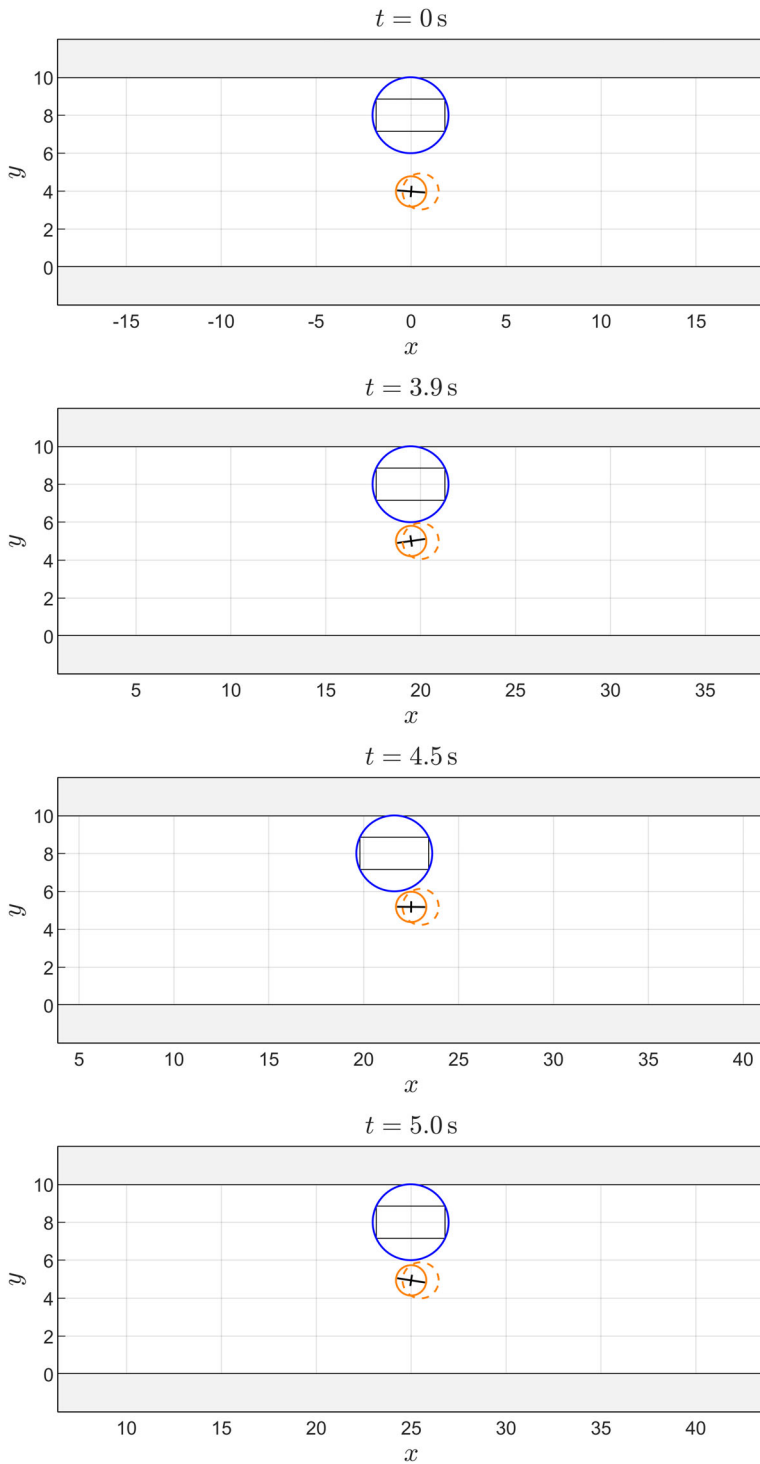
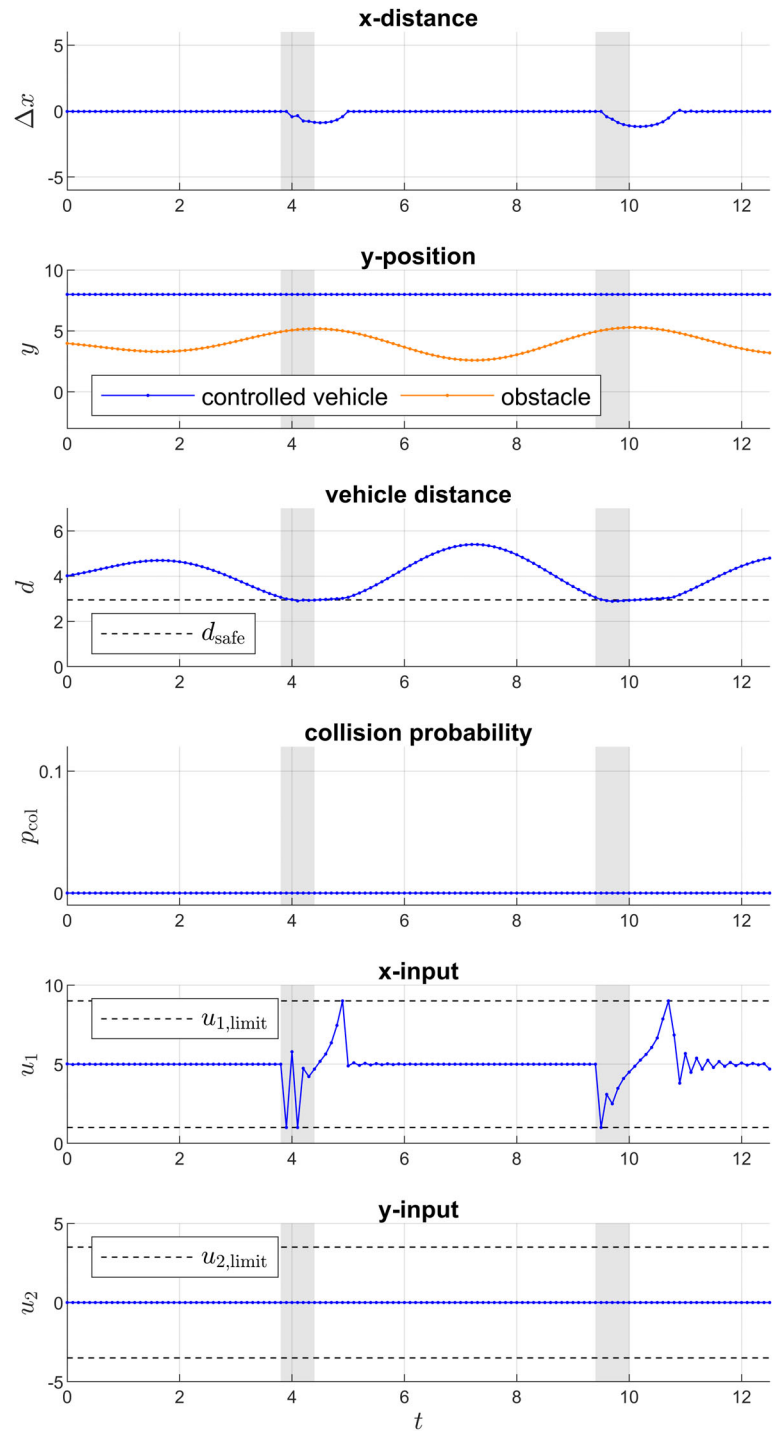


FIGURE 3 Vehicle configurations for the simulation with active state constraints. The controlled vehicle boundary is shown as a solid blue line and the obstacle boundary is a solid orange line. The dashed orange circle represents the possible obstacle location at the next time step [Colour figure can be viewed at wileyonlinelibrary.com]

In the simulation, the obstacle does not move randomly, which helps to better understand the action of the controlled vehicle once the uncertainty support changes. At each time step, the controlled vehicle knows the current uncertainty support of the obstacle.

The vehicle configurations at different time steps are shown in Figure 5 and the results of the simulation are displayed in Figure 6. As the controlled vehicle has a higher velocity it will eventually pass the obstacle, therefore, the distance $\Delta x = x - x_r$ turns positive. At $t = 0.8$ s, the controlled vehicle gets close enough to the obstacle that the controlled vehicle moves away from y_{ref} to maintain $v_{x,\text{ref}}$ and ensures that the distance $d_k = \|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2 \geq d_{\text{safe},k}$. As $w_{\text{max},k}$ increases at $t = 2.0$ s, so does the required distance between the controlled vehicle and obstacle, causing the controlled vehicle to move further away from y_{ref} . Due to input limitations, the controlled vehicle cannot move fast enough. This results

FIGURE 4 Simulation results for the simulation with active state constraints. The controlled vehicle is close to the state constraint. The gray area denotes actions by the controlled vehicle to avoid a collision. The collision probability remains 0 [Colour figure can be viewed at wileyonlinelibrary.com]



in $d_k < d_{\text{safe},k}$, that is, $p_{\text{col},k} > 0$ at $t = 2.0$ s, that is, there is a probability of collision for the next time step. However, d_k is increased to a maximal level, given $\mathbf{u}_k \in \mathbb{U}_k$, resulting in a minimal constraint violation probability $p_{\text{col},k}$. Once the distance satisfies $d_k \geq d_{\text{safe},k}$ at $t = 2.2$ s, $p_{\text{col},k}$ becomes zero, and the controlled vehicle moves along the obstacle boundary for the next step, as seen for $t = 2.3$ s. At $t = 4.0$ s, $w_{\text{max},k}$ decreases, and the controlled vehicle converges to y_{ref} at $t = 4.8$ s.

In order to validate the probability of constraint violation, the simulation was run 2000 times with an arbitrary random obstacle step at $t = 2.0$ s, which is the first step with the increased uncertainty bound $w_{\text{max},k} = 0.9$. The vehicles collided in 144 simulations, yielding a collision probability of 0.072 compared to the calculated collision probability 0.0723, as described in Appendix C.

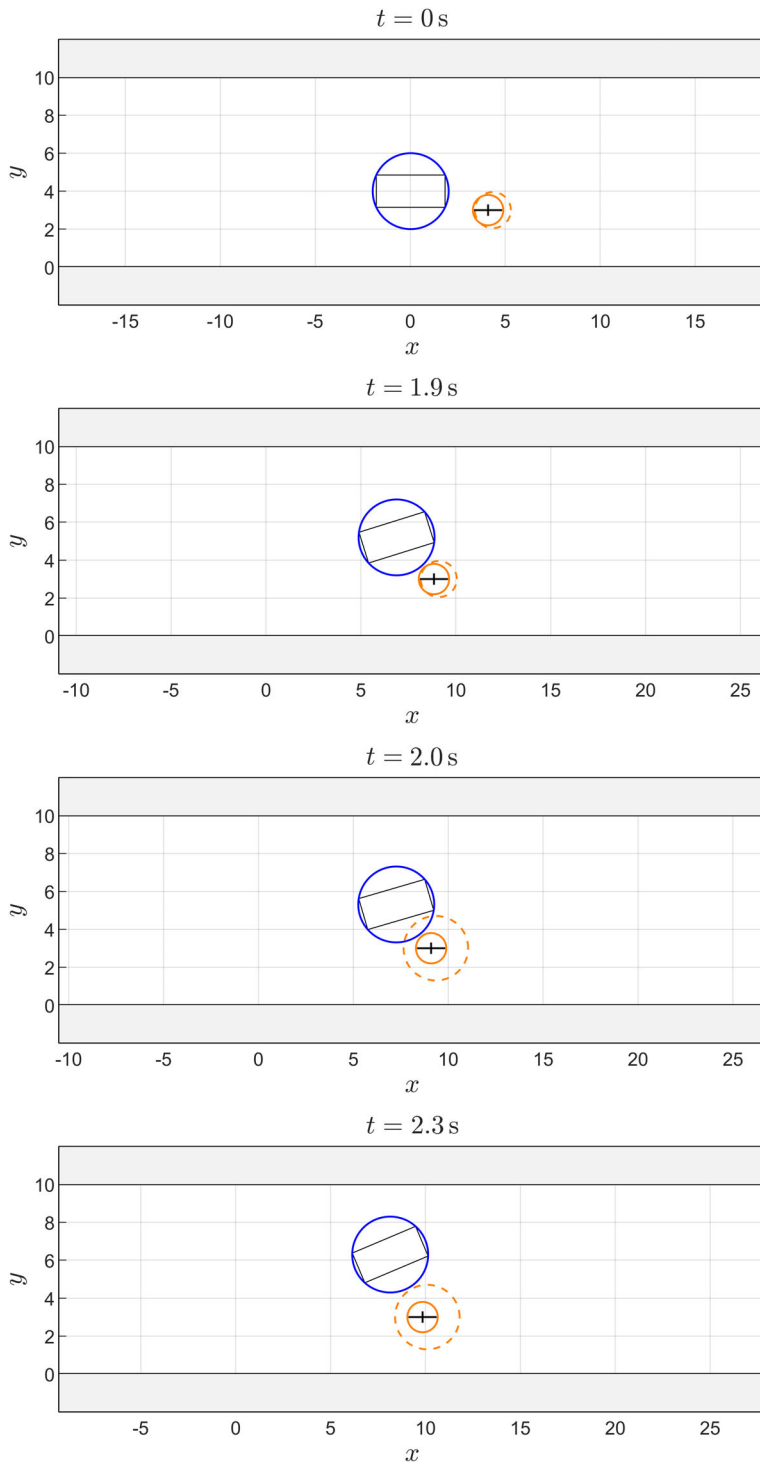


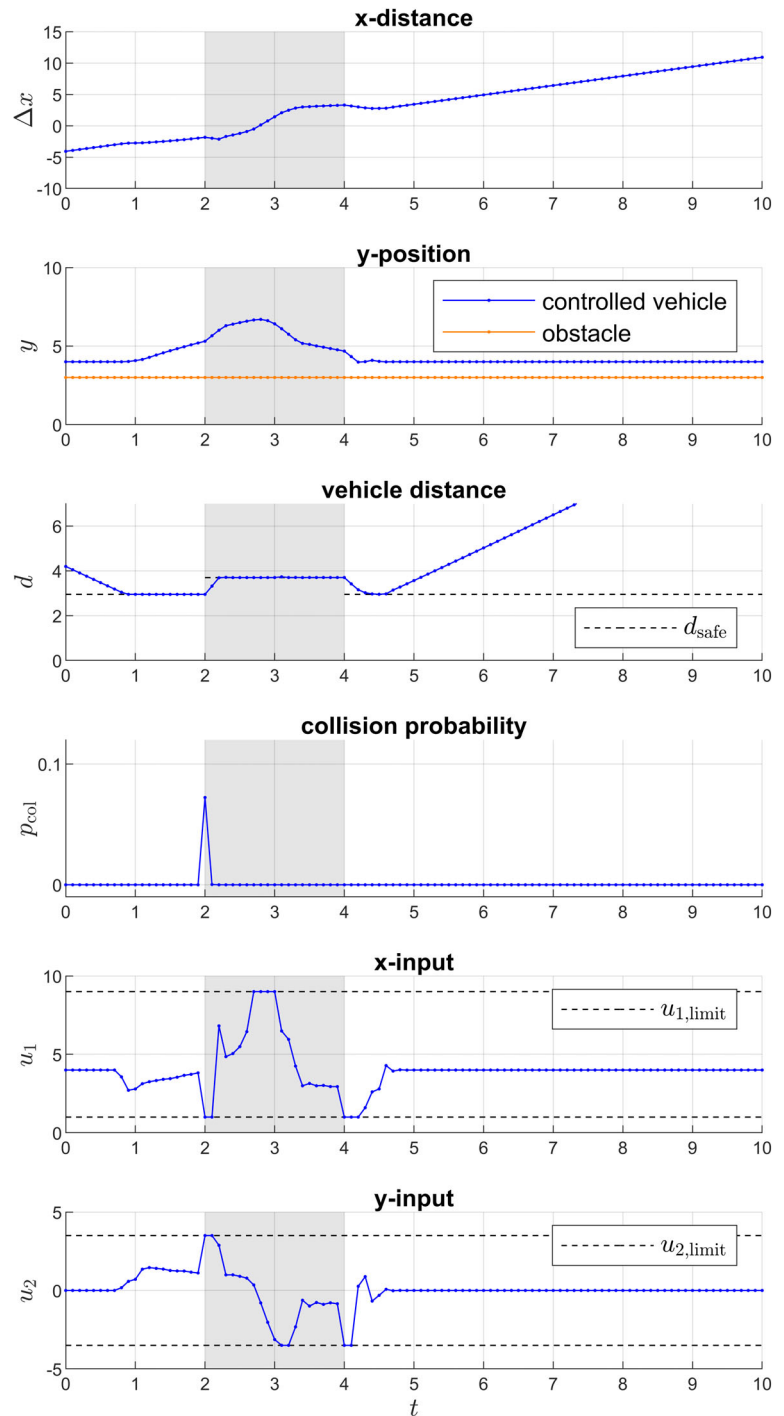
FIGURE 5 Vehicle configurations for the simulation with changing uncertainty support. The controlled vehicle and obstacle boundaries are shown as solid blue and orange lines, respectively. The dashed orange circle represents the possible obstacle location at the next time step [Colour figure can be viewed at wileyonlinelibrary.com]

6.2 | Comparison to RMPC and SMPC

If RMPC and SMPC are applied in the previous simulations, certain problems arise, mainly due to infeasibility of the optimization problem. This could be solved by providing rigorous alternative optimization problems, predefined alternative inputs, or highly conservative worst-case considerations. However, there is no ideal RMPC or SMPC approach to deal with the scenario in the simulation. In the following, we will compare the simulation results of the proposed method to RMPC and SMPC qualitatively and quantitatively.

We will first consider the behavior with RMPC applied to the controlled vehicle. In the first simulation, RMPC delivers safe results similar to the CVPM-MPC method, while remaining behind the obstacle in order to account

FIGURE 6 Simulation results for the simulation with changing uncertainty support. The gray area represents a higher uncertainty support. Once the uncertainty support changes, the collision probability temporarily increases to the minimal level possible [Colour figure can be viewed at wileyonlinelibrary.com]



for the worst-case obstacle behavior. In the second simulation, two cases can be distinguished. If the initially considered uncertainty support is $w_{\max,k} = 0.15$, the behavior is similar to the proposed method until the uncertainty support changes. As it is impossible to find a state with zero collision probability after the uncertainty support is altered, the RMPC optimization problem becomes infeasible. If the considered uncertainty support is initially chosen such that the larger support after $t = 2.0$ s is covered, RMPC yields a safe solution, however, it is passing the obstacle at a larger distance than initially required. In many applications it is also difficult to choose the worst-case uncertainty support a priori, as higher supports might occur later, resulting in even more conservative RMPC solutions.

It is now assumed that the controlled vehicle is controlled using SMPC with a chance constraint with risk parameter $\beta_k > 0$ for collision avoidance. In the second simulation, before the uncertainty support changes, the controlled

TABLE 1 Performance comparison

Method	SMPC risk parameter β_k	Changing support	Small support $w_{\max,k} = 0.15$	Large support $w_{\max,k} = 0.90$
CVPM-MPC	—	1.43e2	—	—
RMPC	—	5.93e2	1.63e2	1.01e4
SMPC	0.8	1.03e2	0.87e2	1.12e2
SMPC	0.9	1.16e2	0.89e2	1.32e2
SMPC	0.95	1.31e2	0.91e2	1.51e2
SMPC	0.99	1.61e2	0.95e2	1.97e2

Abbreviations: MPC, Model Predictive Control; RMPC, Robust Model Predictive Control; SMPC, Stochastic Model Predictive Control.

vehicle passes the obstacle a little closer than with the proposed CVPM-MPC method, as the chance constraint allows for small constraint violations. However, whereas the proposed CVPM-MPC method ensures safety while only passing the vehicle with little more distance, the SMPC approach would pass the obstacle “on the chance constraint,” that is, as close as β_k allows, sacrificing guaranteed safety for small cost improvements. In other words, leaving slightly more space between the controlled vehicle and the obstacle would result in $p_k = 0$ with only little higher cost.

When the uncertainty support changes, the SMPC solution is as close to the obstacle as β_k allowed in the previous step. The chance constraint cannot be met anymore because the uncertainty support increased, resulting in a constraint violation probability larger than allowed by β_k . The SMPC optimization problem then becomes infeasible, requiring an alternative optimization problem to be defined beforehand. In the first simulation, a similar situation occurs. If the chance constraint allows the controlled vehicle to be in a position that yields $p_{\text{col},k} > \beta_k$ due to the unconsidered worst-case obstacle motion, this leads to infeasibility of the optimization problem.

We evaluate the performance of CVPM-MPC by comparing the overall cost to implementations of RMPC⁴³ and SMPC^{29,44} for the simulation scenario of Section 6.1.2. Performance is compared based on the cost function of the optimal control problem, where the applied inputs and resulting states of the entire simulation are evaluated according to

$$J_{\text{sim}} = \sum_{k=0}^{N_{\text{sim}}-1} \mathbf{x}_{k+1}^{\top} \mathbf{Q} \mathbf{x}_{k+1} + \mathbf{u}_k^{\top} \mathbf{R} \mathbf{u}_k, \quad (60)$$

with N_{sim} simulation steps. Four different risk parameters are analyzed for SMPC. The results are shown in Table 1.

We first consider the changing uncertainty support as in Section 6.1.2. The CVPM-MPC method performs significantly better than RMPC, while SMPC has slightly lower cost, except for a conservative choice $\beta_k = 0.99$, where the SMPC cost is larger.

However, both the RMPC and SMPC optimal control problem become infeasible for some steps when the uncertainty support changes at $t = 2.0$ s. Therefore, we analyze how RMPC and SMPC perform if only a small support of $w_{\max,k} = 0.15$ or only a large support of $w_{\max,k} = 0.9$ is assumed and applied. For SMPC, a nontruncated Gaussian distribution is necessary to compute an analytic solution. The distribution is selected to have zero mean and covariance matrix $\Sigma = \text{diag}(\sigma^2, \sigma^2)$ with $\sigma_{0.15} = 0.05$ and $\sigma_{0.9} = 0.3$. The covariance values are chosen such that $\sigma_{0.15}$ and $\sigma_{0.9}$ approximate a distribution with support $w_{\max,k} = 0.15$ or $w_{\max,k} = 0.90$, respectively. For both support cases, the CVPM-MPC performance is always better compared to the RMPC performance, with a significant advantage if $w_{\max,k} = 0.90$ is assumed for RMPC. If a small support is assumed, SMPC is less conservative. If SMPC considers a large support, SMPC is only less conservative for lower risk parameters. Nevertheless, assuming $w_{\max,k} = 0.90$ for RMPC and SMPC is only a partly satisfactory solution. In addition to increased cost, feasibility becomes an issue in the case that the uncertainty support increases again.

The comparison shows that the proposed method offers certain advantages over RMPC and SMPC, especially guaranteeing recursive feasibility of the optimization problem in the presence of a changing uncertainty support.

7 | CONCLUSION

The proposed CVPM-MPC algorithm yields a minimal violation probability for a norm constraint for the next step while also optimizing further objectives and satisfying state and input constraints. Recursive feasibility and, under certain assumptions, convergence to the origin is guaranteed. While the suggested method is inspired by RMPC and SMPC, it provides feasible and efficient solutions in scenarios where RMPC and SMPC encounter difficulties or are not applicable.

As norm constraints are especially useful in collision avoidance applications, the advantages of the presented CVPM-MPC method can be exploited in applications such as autonomous vehicles or robots, especially in shared environments with humans. A brief example is introduced where a controlled vehicle is overtaking a bicycle while minimizing the collision probability. Here, we focus on minimizing the next step constraint violation for a norm constraint. However, depending on the application, a multistep CVPM-MPC could be beneficial. Especially for collision avoidance, it is also of interest not only to focus on the collision probability but to consider the severity of collision if a collision is inevitable.

CVPM-MPC is especially suited for collision avoidance applications. Besides the previously mentioned applications, CVPM-MPC may also be useful in other areas where norm constraints are considered, such as portfolio selection in finance.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial and scientific support by the BMW Group within the CAR@TUM project. We thank Johannes Teutsch for assisting setting up the comparison simulations.

CONFLICT OF INTEREST

The authors declare no potential conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Tim Brüdigam  <https://orcid.org/0000-0001-8913-8003>

REFERENCES

1. Mayne DQ, Rawlings JB, Rao CV, Sckaert POM. Constrained model predictive control: stability and optimality. *Automatica*. 2000;36(6):789-814. [https://doi.org/10.1016/S0005-1098\(99\)00214-9](https://doi.org/10.1016/S0005-1098(99)00214-9)
2. Primbs JA, Nevistić V. Feasibility and stability of constrained finite receding horizon control. *Automatica*. 2000;36(7):965-971. [https://doi.org/10.1016/S0005-1098\(00\)00004-2](https://doi.org/10.1016/S0005-1098(00)00004-2)
3. Grüne L. NMPC without terminal constraints. *IFAC Proc Vol*. 2012;45(17):1-13. <https://doi.org/10.3182/20120823-5-NL-3013.00030>
4. Mayne DQ. Model predictive control: recent developments and future promise. *Automatica*. 2014;50(12):2967-2986. <https://doi.org/10.1016/j.automatica.2014.10.128>
5. Mayne DQ, Seron MM, Raković SV. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*. 2005;41(2):219-224. <https://doi.org/10.1016/j.automatica.2004.08.019>
6. Magni L, De Nicolao G, Scattolini R, Allgöwer F. Robust model predictive control for nonlinear discrete-time systems. *Int J Robust Nonlinear Control*. 2003;13(3-4):229-246. <https://doi.org/10.1002/rnc.815>
7. Mesbah A. Stochastic model predictive control: an overview and perspectives for future research. *IEEE Control Syst*. 2016;36(6):30-44. <https://doi.org/10.1109/MCS.2016.2602087>
8. Farina M, Giullioni L, Scattolini R. Stochastic linear model predictive control with chance constraints - a review. *J Process Control*. 2016;44(Suppl C):53-67. <https://doi.org/10.1016/j.jprocont.2016.03.005>
9. Matthias L, Müller MA, Frank A. Stochastic model predictive control without terminal constraints. *Int J Robust Nonlinear Control*. 2019;29(15):4987-5001. <https://doi.org/10.1002/rnc.3912>
10. Seron MM, Goodwin GC, Carrasco DS. Stochastic model predictive control: insights and performance comparisons for linear systems. *Int J Robust Nonlinear Control*. 2019;29:5038-5057. <https://doi.org/10.1002/rnc.4106>
11. Oldewurtel F, Sturzenegger D, Esfahani PM, Andersson G, Morari M, Lygeros J. Adaptively constrained stochastic model predictive control for closed-loop constraint satisfaction. Paper presented at: Proceedings of the 2013 American Control Conference; 2013; Washington, DC.
12. Schwarm AT, Nikolaou M. Chance-constrained model predictive control. *AIChE J*. 1999;45(8):1743-1752. <https://doi.org/10.1002/aic.690450811>
13. Blackmore L, Ono M, Bektassov A, Williams BC. A probabilistic particle-control approximation of chance-constrained stochastic predictive control. *IEEE Trans Robot*. 2010;26(3):502-517. <https://doi.org/10.1109/TRO.2010.2044948>
14. Schildbach G, Fagiano L, Frei C, Morari M. The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations. *Automatica*. 2014;50(12):3009-3018. <https://doi.org/10.1016/j.automatica.2014.10.035>

15. Korda M, Gondhalekar R, Cigler J, Oldewurtel F. Strongly feasible stochastic model predictive control. Paper presented at: Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference; 2011; Orlando, FL.
16. Kouvaritakis B, Cannon M, Rakovic SV, Cheng Q. Explicit use of probabilistic distributions in linear predictive control. *Automatica*. 2010;46(10):1719-1724. <https://doi.org/10.1016/j.automatica.2010.06.034>
17. Cannon M, Kouvaritakis B, Rakovic SV, Cheng Q. Stochastic tubes in model predictive control with probabilistic constraints. *IEEE Trans Autom Control*. 2011;56(1):194-200. <https://doi.org/10.1109/TAC.2010.2086553>
18. Lorenzen M, Dabbene F, Tempo R, Allgoewer F. Constraint-tightening and stability in stochastic model predictive control. *IEEE Trans Autom Control*. 2017;62(7):3165-3177. <https://doi.org/10.1109/TAC.2016.2625048>
19. Okamoto K, Goldshtein M, Tsiotras P. Optimal covariance control for stochastic systems under chance constraints. *IEEE Control Syst Lett*. 2018;2(2):266-271. <https://doi.org/10.1109/LCSYS.2018.2826038>
20. Okamoto K., Tsiotras P. Stochastic model predictive control for constrained linear systems using optimal covariance steering; 2019. arXiv:1905.13296.
21. Lu J, Xi Y, Li D. Stochastic model predictive control for probabilistically constrained Markovian jump linear systems with additive disturbance. *Int J Robust Nonlinear Control*. 2019;29(15):5002-5016. <https://doi.org/10.1002/rnc.3971>
22. Hewing L, Wabersich KP, Zeilinger MN. Recursively feasible stochastic model predictive control using indirect feedback. *Automatica*. 2020;119:109095. <https://doi.org/10.1016/j.automatica.2020.109095>
23. Jurado I, Millán P, Quevedo D, Rubio FR. Stochastic MPC with applications to process control. *Int J Control*. 2015;88(4):792-800. <https://doi.org/10.1080/00207179.2014.975845>
24. Oldewurtel F, Jones CN, Parisio A, Morari M. Stochastic model predictive control for building climate control. *IEEE Trans Control Syst Technol*. 2014;22(3):1198-1205. <https://doi.org/10.1109/TCST.2013.2272178>
25. Kumar R, Wenzel MJ, Ellis MJ, ElBsat MN, Drees KH, Zavala VM. A stochastic model predictive control framework for stationary battery systems. *IEEE Trans Power Syst*. 2018;33(4):4397-4406. <https://doi.org/10.1109/TPWRS.2017.2789118>
26. Jiang Y, Wan C, Wang J, Song Y, Dong ZY. Stochastic receding horizon control of active distribution networks with distributed renewables. *IEEE Trans Power Syst*. 2019;34(2):1325-1341. <https://doi.org/10.1109/TPWRS.2018.2879451>
27. Graf PM, Puglia L, Gabbriellini T, Bemporad A. Dynamic option hedging with transaction costs: a stochastic model predictive control approach. *Int J Robust Nonlinear Control*. 2019;29(15):5058-5077. <https://doi.org/10.1002/rnc.3915>
28. Ripaccioli G, Bernardini D, Di Cairano S, Bemporad A, Kolmanovsky IV. A stochastic model predictive control approach for series hybrid electric vehicle power management. Paper presented at: Proceedings of the 2010 American Control Conference; 2010; Baltimore.
29. Carvalho A, Gao Y, Lefevre S, Borrelli F. Stochastic predictive control of autonomous vehicles in uncertain environments. Paper presented at: Proceedings of the 12th International Symposium on Advanced Vehicle Control; 2014; Tokyo, Japan.
30. Schildbach G, Borrelli F. Scenario model predictive control for lane change assistance on highways. Paper presented at: Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV); 2015; Seoul, South Korea.
31. Lenz D, Kessler T, Knoll A. Stochastic model predictive controller with chance constraints for comfortable and safe driving behavior of autonomous vehicles. Paper presented at: Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV); 2015; Seoul, South Korea.
32. Cesari G, Schildbach G, Carvalho A, Borrelli F. Scenario model predictive control for lane change assistance and autonomous driving on highways. *IEEE Intell Transp Syst Mag*. 2017;9(3):23-35. <https://doi.org/10.1109/MITS.2017.2709782>
33. Brüdigam T, Olbrich M, Leibold M, Wollherr D. Combining stochastic and scenario model predictive control to handle target vehicle uncertainty in autonomous driving. Paper presented at: Proceedings of the 21st IEEE International Conference on Intelligent Transportation Systems; 2018; Maui, HI.
34. Suh J, Chae H, Yi K. Stochastic model-predictive control for lane change decision of automated driving vehicles. *IEEE Trans Veh Technol*. 2018;67(6):4771-4782. <https://doi.org/10.1109/TVT.2018.2804891>
35. Schulman J, Duan Y, Ho J, et al. Motion planning with sequential convex optimization and convex collision checking. *Int J Robot Res*. 2014;33(9):1251-1270. <https://doi.org/10.1177/0278364914528132>
36. Zanchettin AM, Ceriani NM, Rocco P, Ding H, Matthias B. Safety in human-robot collaborative manufacturing environments: metrics and control. *IEEE Trans Automat Sci Eng*. 2016;13(2):882-893. <https://doi.org/10.1109/TASE.2015.2412256>
37. Varshney KR, Alemzadeh H. On the safety of machine learning: cyber-physical systems, decision sciences, and data products. *Big Data*. 2017;5(3):246-255. <https://doi.org/10.1089/big.2016.0051>
38. Depeweg S, Hernandez-Lobato JM, Doshi-Velez F, Udfluft S. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. Paper presented at: Proceedings of the 35th International Conference on Machine Learning; 2018; Stockholm, Sweden.
39. Wojsznis W, Mehta A, Wojsznis P, Thiele D, Blevins T. Multi-objective optimization for model predictive control. *ISA Trans*. 2007;46(3):351-361. <https://doi.org/10.1016/j.isatra.2006.10.002>
40. Bemporad A, de la Peña DM. Multiobjective model predictive control. *Automatica*. 2009;45(12):2823-2830. <https://doi.org/10.1016/j.automatica.2009.09.032>
41. Herceg M, Kvasnica M, Jones CN, Morari M. Multi-parametric toolbox 3.0. Paper presented at: Proceedings of the European Control Conference; 2013; Zürich, Switzerland.
42. Löfberg J. YALMIP: a toolbox for modeling and optimization in MATLAB. Paper presented at: Proceedings of the CACSD Conference; 2004; Taipei, Taiwan.
43. Rawlings JB, Mayne DQ, Diehl M. *Model Predictive Control: Theory, Computation, and Design*. 2nd ed. Madison, Wisconsin: Nob Hill Publishing; 2017.

44. Brüdigam T, Teutsch J, Wollherr D, Leibold M. Combined Robust and Stochastic Model Predictive Control for Models of Different Granularity. *IFAC-PapersOnLine*. 2020;53(2):7123-7129. <http://dx.doi.org/10.1016/j.ifacol.2020.12.515>.
45. Bauer H. Minimalstellen von Funktionen und Extrempunkte. *Archiv der Mathematik*. 1958;9(4):389-393.
46. Boyd S, Vandenberghe L. *Convex Optimization*. 1st ed. Cambridge, MA: Cambridge University Press; 2004.

How to cite this article: Brüdigam T, Gaßmann V, Wollherr D, Leibold M. Minimization of constraint violation probability in model predictive control. *Int J Robust Nonlinear Control*. 2021;1–33. <https://doi.org/10.1002/rnc.5636>

APPENDIX A. PROOFS

The following appendix contains the proofs for this work.

A.1 Proof of Theorem 1

Proof. The proof follows straightforward from the definition of the three cases. All possibilities are covered regarding the guarantee of constraint satisfaction, that is, guaranteed constraint satisfaction (case 1), impossible constraint satisfaction guarantee (case 2), and the case where constraint satisfaction is only guaranteed for some but not all $\mathbf{u}_0 \in \mathbb{U}_0$ (case 3). If $p_1 = 0$ is possible, that is, case 1 or 3, (17) and (21) guarantee that $\mathbb{U}_{\text{cvpm},0}$ consists only of inputs $\mathbf{u}_0 \in \mathbb{U}_0$ that yield $p_1 = 0$. If no $\mathbf{u}_0 \in \mathbb{U}_0$ guarantees $p_1 = 0$, minimal constraint violation is guaranteed by only allowing inputs $\mathbf{u}_0 \in \mathbb{U}_0$ that minimize p_1 according to (19). ■

A.2 Proof of Corollary 1

Proof. The proof follows directly from the problem definition. First, the CVPM-MPC approach ensures that the constraint violation probability is minimized for each step, which allows $p_{k+2} > p_{k+1}$ if the uncertainty support increases. Second, minimizing p_{k+2} is independent of minimizing p_{k+1} . ■

A.3 Proof of Lemma 1

Proof. According to Assumption 6, $f_{\mathbf{w}_k}$ is symmetric and unimodal, and therefore $f_{\mathbf{w}_k}$ is decreasing with increasing $\|\mathbf{w}_k\|_2$, i.e., the larger $\|\mathbf{w}_k\|_2$, the lower its PDF value. The uncertainty realization with the highest relative likelihood is the mode of $f_{\mathbf{w}_k}$ with $\mathbf{w}_k = \mathbf{0}$, yielding the most likely random output $\mathbf{y}_{r,k+1} = \bar{\mathbf{y}}_{r,k+1}$. It follows that

$$f_{\mathbf{w}_k}(\tilde{\mathbf{w}}_k) < f_{\mathbf{w}_k}(\mathbf{w}_k) \quad \text{for} \quad \|\tilde{\mathbf{w}}_k\|_2 > \|\mathbf{w}_k\|_2, \quad (\text{A1})$$

where $\tilde{\mathbf{y}}_{r,k+1} = \bar{\mathbf{y}}_{r,k+1} + \tilde{\mathbf{w}}_k$ is less likely than $\mathbf{y}_{r,k+1} = \bar{\mathbf{y}}_{r,k+1} + \mathbf{w}_k$ and

$$\|\bar{\mathbf{y}}_{r,k+1} - \tilde{\mathbf{y}}_{r,k+1}\|_2 > \|\bar{\mathbf{y}}_{r,k+1} - \mathbf{y}_{r,k+1}\|_2, \quad (\text{A2})$$

due to $\|\tilde{\mathbf{w}}_k\|_2 > \|\mathbf{w}_k\|_2$.

It follows that the larger $\|\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{r,k+1}\|_2$, the higher the PDF value of a large $\|\mathbf{y}_{k+1} - \mathbf{y}_{r,k+1}\|_2$ due to (A1). Therefore, the larger $\|\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{r,k+1}\|_2$, the higher the PDF value of $\|\mathbf{y}_{k+1} - \mathbf{y}_{r,k+1}\|_2 \geq c_k$. This results in

$$\|\tilde{\mathbf{y}}_{k+1} - \bar{\mathbf{y}}_{r,k+1}\|_2 > \|\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{r,k+1}\|_2 \Leftrightarrow p_{k+1}(\tilde{\mathbf{u}}_k, \mathbf{y}_{r,k}) \leq p_{k+1}(\mathbf{u}_k, \mathbf{y}_{r,k}) \quad (\text{A3})$$

with $\tilde{\mathbf{y}}_{k+1} \neq \mathbf{y}_{k+1}$ and $\tilde{\mathbf{y}}_{k+1} = \mathbf{C}(\mathbf{A}\mathbf{x}_k + \mathbf{B}\tilde{\mathbf{u}}_k)$ according to (1).

The same holds for $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2$ instead of $\|\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{r,k+1}\|_2$, showing that p_k is decreasing with increasing $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2$. ■

A.4 Proof of Lemma 2

Proof. This proof is based on Bauer's maximum principle.⁴⁵ We consider any two points $\mathbf{v}_1, \mathbf{v}_2 \in \partial\mathcal{V}$ on the boundary of \mathcal{V} . Any point on the line between $\mathbf{v}_1, \mathbf{v}_2$ can be described by $\mathbf{b} = \lambda\mathbf{v}_1 + (1 - \lambda)\mathbf{v}_2$, using the definition of convexity. Due

to the convexity of z , it holds that $z(\mathbf{b}) \leq \max\{z(\mathbf{v}_1), z(\mathbf{v}_2)\}$. Any point on the line between $\mathbf{v}_1, \mathbf{v}_2$ can be reached by a convex combination. Since $\mathbf{v}_1, \mathbf{v}_2$ can be chosen arbitrarily, every point \mathbf{b} in the interior of \mathcal{V} can be reached. Therefore, a global maximum z_{\max} is found on the boundary $\partial\mathcal{V}$. ■

A.5 Proof of Proposition 1

Proof. The set $\mathbb{U}_{\text{mode}3,0}$ is nonempty and nonconvex with the boundary point $\mathbf{u}_0^* \in \partial\mathbb{U}_{\text{mode}3,0}$ of $\mathbb{U}_{\text{mode}3,0}$. There exists a supporting hyperplane to $\mathbb{U}_{\text{mode}3,0}$ at \mathbf{u}_0^* .⁴⁶ This supporting hyperplane is used to approximate the nonconvex set $\mathbb{U}_{\text{mode}3,0}$. The gradient $\nabla_{\mathbf{u}_0^*} \left(h \left(\left\| \mathbf{y}_1(\mathbf{u}_0^*) - \bar{\mathbf{y}}_{r,1} \right\|_2 \right) \right)$ is a vector orthogonal to the hyperplane on the boundary $\partial\mathbb{U}_{\text{mode}3,0}$ at \mathbf{u}_0^* , pointing away from the convex set $\mathbb{U}_{\text{mode}3,0}$. The scalar product of $\nabla_{\mathbf{u}_0^*} \left(h \left(\left\| \mathbf{y}_1(\mathbf{u}_0^*) - \bar{\mathbf{y}}_{r,1} \right\|_2 \right) \right)$ and any point \mathbf{u}_0 on this hyperplane is zero, while the scalar product of $\nabla_{\mathbf{u}_0^*} \left(h \left(\left\| \mathbf{y}_1(\mathbf{u}_0^*) - \bar{\mathbf{y}}_{r,1} \right\|_2 \right) \right)$ and any point in the half plane not containing $\mathbb{U}_{\text{mode}3,0}$ is positive. Therefore, (43) approximates $\mathbb{U}_{\text{mode}3,0}$. As the intersection of two convex sets yields a convex set,⁴⁶ the resulting approximated set $\hat{\mathbb{U}}_{\text{cvpm},0}$ is convex as well. ■

A.6 Proof of Theorem 2

Proof. As shown in the proof of Theorem 1, the three cases (16), (18), and (20) cover all possibilities with individual, nonempty sets $\mathbb{U}_{\text{cvpm},0}$. This yields that there always exists a $\mathbf{u}_0 \in \mathbb{U}_{\text{cvpm},0}$.

As $\mathbf{u}_0 \in \mathbb{U}_0$, it holds that conditions (5c), (5d), (5e) can be fulfilled with $\mathbf{u}_j \in \mathbb{U}_j$ for $j \in \mathbb{I}_{1:N-1}$ according to (6) and due to Assumption 2. No input \mathbf{u}_0 is possible which would cause $\mathbb{U}_j = \emptyset$ for $j \in \mathbb{I}_{1:N-1}$. Therefore, feasible solutions \mathbf{u}_j exist and \mathbb{U}_{x_j} is a nonempty set for $j \in \mathbb{I}_{1:N-1}$.

The first condition in (23) considers the first input \mathbf{u}_0 , while the second condition covers the following inputs \mathbf{u}_j with $j \in \mathbb{I}_{1:N-1}$. Therefore, the two conditions are independent and $\mathbb{U}_0^* \neq \emptyset$ for any MPC optimization. The MPC algorithm (22) is guaranteed recursively feasible. ■

A.7 Proof of Corollary 2

Proof. The proof follows straightforward from Theorem 2, showing that $\mathbb{U}_{\text{cvpm},k} \neq \emptyset$ for all three cases (32), (34), and (37). According to Lemma 2, $h_{\min,1}$ and $h_{\max,1}$ can always be found. Given any value for $h(c_1 + w_{\max,0})$ exactly one of the three cases is applicable, yielding $\mathbb{U}_{\text{cvpm},0} \neq \emptyset$. For cases 1 and 2 no approximation is necessary. If $\hat{\mathbb{U}}_{\text{cvpm},0} = \emptyset$ for case 3, the approach of case 2 is used according to Remark 6, i.e., $\mathbb{U}_{\text{cvpm},0} = \{\mathbf{u}_{\text{cvpm},0}\}$. Therefore, $\mathbb{U}_{\text{cvpm},k} \neq \emptyset$ for all three cases. ■

A.8 Proof of Lemma 3

Proof. As cases 1 or 3 are applied, the space blocked by $\mathcal{X}_{\text{cv},k}$ around $\mathbf{y}_{r,k}$ with nonzero constraint violation probability can be regarded as a hard constraint. This yields $\mathbf{x}_k \notin \mathcal{X}_{\text{cv},k}$ for all $k \geq k_{\text{case}1,3}$. As \mathcal{X} is closed and $\mathcal{X}_{\text{cv},k}$ is open, the resulting set $\tilde{\mathcal{X}}_k$ is closed. As $\mathbf{x}_k \in \tilde{\mathcal{X}}_k \subseteq \mathcal{X}$, there exists a \mathbf{u}_k such that $\mathbf{x}_{k+1} \in \mathcal{X}$ according to Theorem 2. Assumption 7 (c) ensures that $\mathbb{U}_{\text{cvpm},k}$ is not empty, therefore $\mathbf{x}_{k+1} \in \tilde{\mathcal{X}}_k$ and $\tilde{\mathcal{X}}_k$ is control invariant. ■

A.9 Proof of Theorem 3

Proof. First, the MPC algorithm in (5) will be considered without the norm constraint (7). As $V(\mathbf{x}_0, \mathbf{U}_0)$ is a Lyapunov function in \mathcal{X} , given Assumption 3, the MPC algorithm of (22) without (7) is asymptotically stable, following the MPC stability proof of Rawlings et al.⁴³

Now, the CVPM-MPC method is considered. According to Theorem 2, for all $k, \mathbf{x}_k \in \mathcal{X}$ there exists a feasible \mathbf{U}_k such that \mathbf{x}_{k+1} remains in \mathcal{X} . Lemma 3 ensures that $\mathbf{x}_{k^*} \in \tilde{\mathcal{X}}_k$ for $k^* \geq k_{\text{case}1,3}$, where $\tilde{\mathcal{X}}_k$ replaces \mathcal{X} to ensure constraint satisfaction of the norm constraint. The set $\tilde{\mathcal{X}}_k$ is closed, control invariant, contains the origin according to Assumption 7, and $\mathcal{X}_i \subseteq \tilde{\mathcal{X}}_k$, given Assumption 8. Therefore, the system (1), controlled by the CVPM-MPC algorithm in (22), is asymptotically stable and converges to $\mathbf{0}$ for $k > k^*$ and $k \rightarrow \infty$, similar to the MPC algorithm in (5). ■

A.10 Proof of Corollary 3

Proof. The proof is similar to the proof of Theorem 3. The set $\mathcal{X}_{\text{cv},k}$ in (46) can be expressed as

$$\mathcal{X}_{cv,k} = \left\{ \mathbf{x}_k \mid h(\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2) < h(c_k + w_{\max,k-1}), \mathbf{y}_k = \mathbf{C}\mathbf{x}_k \right\}. \quad (\text{A4})$$

Equation (47) is satisfied by

$$h(\|\mathbf{0} - \bar{\mathbf{y}}_{r,k}\|) \geq h(c_k + w_{\max,k-1}) \quad \forall k \geq k_0, \quad (\text{A5})$$

while (48) transforms into

$$\exists \mathbf{u}_{k-1} \text{ s.t. } h(\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2) \geq h(c_k + w_{\max,k-1}), \quad (\text{A6})$$

for the CVPM-MPC method in Section 3.3.

Similar to Lemma 3, given the open and constant set $\mathcal{X}_{cv,k}$ and Assumption 7, $\tilde{\mathcal{X}}_k$ is closed, constant, control invariant, and contains the origin. With the MPC algorithm (5) and $k > k^*$, $k \rightarrow \infty$ the system (1) is asymptotically stable and therefore converges to $\mathbf{0}$.

APPENDIX B. MINIMAL CONSTRAINT VIOLATION PROBABILITY FOR THE MULTI-STEP PROBLEM

The method presented in Section 3 minimizes the constraint violation probability for the next step. In the following, a possible extension of the one-step CVPM-MPC method is shown. Considering multiple steps $l > 1$ yields a method closer related to RMPC, as it provides advantages with respect to robustness but conservatism is increased.

Considering the stochastic process $(\mathbf{W}_k)_{k \in \mathbb{I}_{0:j-1}}$, its realization, a sequence $(\mathbf{w}_k)_{k \in \mathbb{I}_{0:j-1}}$ with $j \in \mathbb{N}_{\geq 0}$, and the initially known output $\mathbf{y}_{r,0}$ yields

$$\mathbf{y}_{r,k} = \mathbf{y}_{r,0} + \sum_{i=0}^{k-1} \mathbf{w}_i. \quad (\text{B1})$$

While in the one-step method p_j only needs to be minimized for the next step $j = 1$, for the l -step approach p_j needs to be minimized for $1 \leq j \leq l$.

Similar to Section 3, we first address the general method and then provide a solution for $f_{\mathbf{w}_k}$ satisfying Assumption 6.

B.1 General method to minimize constraint violation probability for the multistep problem

It is necessary to find the set $\mathbb{U}_{cvpm,0:l-1}$, which represents the set of admissible input sequences $\mathbf{U}_{0:l-1} = [\mathbf{u}_0, \dots, \mathbf{u}_{l-1}]^\top$ that minimize p_j for $1 \leq j \leq l$. In the following, three cases are again considered. The constraint violation p_{j+1} for step $j + 1$ depends on the previous output \mathbf{y}_j , the input \mathbf{u}_j , and the uncertain output $\mathbf{y}_{r,j}$.

Case 1 (Guaranteed constraint satisfaction)

Constraint satisfaction is guaranteed for all steps $j \in \mathbb{I}_{0:l-1}$, that is,

$$p_{j+1}(\mathbf{u}_j) = 0 \quad \forall \mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1}, \quad (\text{B2})$$

resulting in

$$\mathbb{U}_{cvpm,0:l-1} = \left\{ \mathbf{u}_j \mid \mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1} \right\}. \quad (\text{B3})$$

Case 2 (Impossible constraint satisfaction guarantee)

For a j with $0 \leq j \leq l - 1$, potentially at multiple steps j , constraint satisfaction cannot be guaranteed by any input $\mathbf{u}_j \in \mathbb{U}_j$, that is, $p_{j+1} = 0$. This can be expressed by

$$\exists j \in \mathbb{I}_{0:l-1} \text{ s.t. } p_{j+1}(\mathbf{u}_j) > 0 \quad \forall \mathbf{u}_j \in \mathbb{U}_j. \quad (\text{B4})$$

The set of admissible inputs that minimize the constraint violation probability is then given by

$$\mathbb{U}_{\text{cvpm},0:l-1} = \left\{ \mathbf{u}_j \mid \mathbf{u}_j = \arg \min_{\mathbf{u}_j \in \mathbb{U}_j} p_{j+1}(\mathbf{u}_j), j \in \mathbb{I}_{0:l-1} \right\}. \quad (\text{B5})$$

Case 3 (Possible constraint satisfaction guarantee)

At each step $0 \leq j \leq l-1$ it is possible, but not guaranteed, that the norm constraint is satisfied for $j+1$, that is,

$$\exists \mathbf{u}_j \in \mathbb{U}_j \text{ s.t. } p_{j+1}(\mathbf{u}_j) = 0 \quad \forall j \in \mathbb{I}_{0:l-1}. \quad (\text{B6})$$

This yields

$$\mathbb{U}_{\text{cvpm},0:l-1} = \left\{ \mathbf{u}_j \mid (p_{j+1}(\mathbf{u}_j) = 0) \wedge (\mathbf{u}_j \in \mathbb{U}_j), j \in \mathbb{I}_{0:l-1} \right\}.$$

B.2 Minimal constraint violation probability for the multistep problem with symmetric unimodal PDF

After defining the general case, we now address the multi-step CVPM-MPC method for a symmetric, unimodal PDF. We make the following assumptions.

Assumption 9 (Constant minimal norm value). The minimal norm value $c_j = c$ is constant.

Assumption 10 (Known deterministic input). The deterministic input $\mathbf{u}_{r,j}$ is known for $j \in \mathbb{I}_{0:l-1}$.

A simple approach to find $\mathbb{U}_{\text{cvpm},0:l-1}$ is to maximize $\|\mathbf{y}_l - \bar{\mathbf{y}}_{r,l}\|_2$ with

$$\bar{\mathbf{y}}_{r,l} = \mathbf{y}_{r,0} + \sum_{i=0}^{l-1} \mathbf{u}_{r,i}, \quad (\text{B7})$$

as this automatically results in a maximization of $\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2$ for $j \in \mathbb{I}_{0:l-1}$ because p_j is decreasing with increasing $\|\mathbf{y}_j - \bar{\mathbf{y}}_{r,j}\|_2$. Therefore, if p_l is minimized, p_j is also minimized for $j \in \mathbb{I}_{0:l-1}$. Similar to (28) and (29), we define

$$h_{\max,l} = h \left(\max_{\mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1}} (\|\mathbf{y}_l - \bar{\mathbf{y}}_{r,l}\|_2) \right) \quad (\text{B8})$$

$$h_{\min,l} = h \left(\min_{\mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1}} (\|\mathbf{y}_l - \bar{\mathbf{y}}_{r,l}\|_2) \right). \quad (\text{B9})$$

We now regard the three possible cases and determine $\mathbb{U}_{\text{cvpm},0:l-1}$.

Case 1 (Constraint satisfaction guarantee)

For any $\mathbf{U}_{0:l-1}$ it follows that $p_{j+1} = 0$ for $j \in \mathbb{I}_{0:l-1}$, that is,

$$h_{\min,l} \geq h \left(c + \sum_{i=0}^{l-1} w_{\max,i} \right). \quad (\text{B10})$$

In comparison to (32), \mathbf{w}_i needs to be considered for $i \in \mathbb{I}_{0:l-1}$. This yields

$$\mathbb{U}_{\text{cvpm},0:l-1} = \left\{ \mathbf{U}_{0:l-1} \mid \mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1} \right\}. \quad (\text{B11})$$

Case 2 (Impossible constraint satisfaction guarantee)

It is not possible to guarantee $p_{j+1} = 0$ for $j \in \mathbb{I}_{0:l-1}$ as

$$h_{\max,l} < h \left(c + \sum_{i=0}^{l-1} w_{\max,i} \right) \quad (\text{B12})$$

Therefore, $\mathbb{U}_{\text{cvpm},0:l-1}$ consists of the input sequence $\mathbf{U}_{0:l-1}$ that minimizes p_l , resulting in

$$\mathbb{U}_{\text{cvpm},0:l-1} = \left\{ \mathbf{U}_{0:l-1} \mid \mathbf{U}_{0:l-1} = \arg \max_{\mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1}} h(\|\mathbf{y}_l - \bar{\mathbf{y}}_{r,l}\|_2) \right\}. \quad (\text{B13})$$

Case 3 (Possible constraint satisfaction guarantee)

There are possible input sequences $\mathbf{U}_{0:l-1}$ such that $p_l = 0$. Similar to case 3 for the one-step CVPM-MPC, we again need to find a set $\mathbb{U}_{\text{cvpm},0:l-1}$ that only allows input sequences $\mathbf{U}_{0:l-1}$ resulting in constraint satisfaction of (7) for $j \in \mathbb{I}_{1:l}$. This is achieved by choosing

$$\mathbb{U}_{\text{cvpm},0:l-1} = \left\{ \mathbf{U}_{0:l-1} \mid h(\|\mathbf{y}_l - \bar{\mathbf{y}}_{r,l}\|_2) \geq h \left(c + \sum_{i=0}^{l-1} w_{\max,i} \right) \cap \mathbf{u}_j \in \mathbb{U}_j, j \in \mathbb{I}_{0:l-1} \right\}, \quad (\text{B14})$$

where the approximation $\hat{\mathbb{U}}_{\text{cvpm},0:l-1}$ can be found analogously to Proposition 1.

The l -step CVPM-MPC algorithm can then be formulated as in (22) with

$$\mathbb{U}_0^* = \left\{ \mathbf{U}_0 \mid \mathbf{U}_{0:l-1} \in \mathbb{U}_{\text{cvpm},0:l-1} \wedge \mathbf{u}_j \in \mathbb{U}_{x_j}, j \in \mathbb{I}_{l:N-1} \right\}. \quad (\text{B15})$$

APPENDIX C. COLLISION PROBABILITY FUNCTION

Here, the collision probability $p_{\text{col},k}$ is described in detail, which is only needed for the evaluation of the simulation but not the proposed method. The PDF $f_{\mathbf{w}_k}$ is chosen to be

$$f_{\mathbf{w}_k}(r_k) = \begin{cases} \frac{1}{\sigma z \sqrt{2\pi}} e^{-\frac{r_k^2}{2\sigma^2}} & \text{if } 0 \leq r_k \leq w_{\max,k}, \\ 0 & \text{otherwise} \end{cases} \quad (\text{C1})$$

where r_k is used instead of \mathbf{w}_k and

$$\text{supp}(f_{\mathbf{w}_k}) = \left\{ r_k \mid 0 \leq r_k \leq w_{\max,k} \right\}, \quad (\text{C2})$$

with variance $\sigma = 1$ and

$$z = \Phi(w_{\max,k}) - \Phi(0) \quad (\text{C3})$$

$$\Phi(r) = 0.5 \left(1 + \text{erf} \left(\frac{r}{\sqrt{2}} \right) \right) \quad (\text{C4})$$

such that

$$\int_{\text{supp}(f_{\mathbf{w}_k})} f_{\mathbf{w}_k}(r_k) dr_k = 1. \quad (\text{C5})$$

As the main aim of this simulation is to minimize the constraint violation probability, that is, the collision probability, an expression for this probability is necessary in order to analyze the simulation results. The controlled vehicle and the obstacle collide if their bodies overlap, that is, $r_{\text{comb}} > \|\mathbf{y}_k - \mathbf{y}_{r,k}\|_2$ with the combined radius $r_{\text{comb}} = r_c + r_r$. A collision at step k is inevitable if $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2 + w_{\text{max},k-1} < r_{\text{comb}}$, i.e., even for the best-case $w_{\text{max},k-1}$ the objects will collide at step k . For $\|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2 - w_{\text{max},k-1} \geq r_{\text{comb}}$ it follows that $p_{\text{col},k} = 0$.

The collision probability is calculated according to the following procedure. We consider a circle where the radius is the required distance r_{comb} and a circle with radius r_k . The intersection of both circles can be interpreted as the collision probability, by integrating the intersection area of both circles, weighted with $f_{w_k}(r_k)$. This is illustrated in Figure C1. In case that there is no intersection area, then $p_{\text{col},k} = 0$. If an intersection exists, there are two intersection points. The intersection area is therefore bounded on one side by the arc with radius r_{comb} and on the other side by the arc of the boundary of the uncertainty. As the intersection area is symmetric, it is sufficient to derive the calculation for one half, that is, the area between the line connecting $\bar{\mathbf{y}}_{r,k}$ and \mathbf{y}_k and the intersection point $\mathbf{p}_{\text{int},1}$ as depicted by the striped area in Figure C1. This yields an angle $\theta_{\text{int},1} \in [0; 0.5\pi]$ between the two lines connecting $\bar{\mathbf{y}}_{r,k}$ and \mathbf{y}_k as well as $\bar{\mathbf{y}}_{r,k}$ and $\mathbf{p}_{\text{int},1}$. The distance $r_{\text{int}}(\theta)$ between $\bar{\mathbf{y}}_{r,k}$ and the controlled vehicle boundary between the two intersection points follows from the law of cosines

$$r_{\text{comb}}^2 = r_{\text{int}}(\theta)^2 + d_k^2 - 2d_k r_{\text{int}}(\theta) \cos(\theta), \quad (\text{C6})$$

where $d_k = \|\mathbf{y}_k - \bar{\mathbf{y}}_{r,k}\|_2$ and $\theta \in [0; \theta_{\text{int},1}]$ with

$$\theta_{\text{int},1} = \sin^{-1} \left(\frac{a}{2w_{\text{max},k-1}} \right), \quad (\text{C7})$$

$$a = \frac{\sqrt{4d_k^2 w_{\text{max},k-1}^2 - (d_k^2 - r_{\text{comb}}^2 + w_{\text{max},k-1}^2)^2}}{d_k}. \quad (\text{C8})$$

This yields

$$r_{\text{int}}(\theta) = 0.5 \left(2d_k \cos(\theta) - \sqrt{(2d_k \cos(\theta))^2 - 4(d_k^2 - r_{\text{comb}}^2)} \right). \quad (\text{C9})$$

The intersection area on both sides of the line between \mathbf{y}_k and $\bar{\mathbf{y}}_{r,k}$, weighted with the PDF $f_{w_{k,\text{pol}}}$, yields the collision probability

$$p_{\text{col},k} = 2 \int_0^{\theta_{\text{int},1}} \frac{1}{2\pi} \int_{r_{\text{int}}(\theta)}^{w_{\text{max},k-1}} f_{w_k}(r) dr d\theta, \quad (\text{C10})$$

for $d_k + w_{\text{max},k-1} \geq r_{\text{comb}}$, depending on the angle $\theta_{\text{int},1}$.

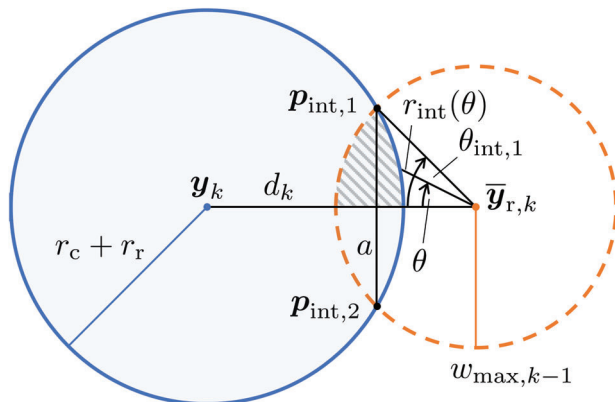


FIGURE C1 Collision probability calculation. The blue circle combines the radius of the controlled vehicle and the obstacle, the dashed orange circle represents the area potentially covered by the uncertainty. The striped area represents one half of the intersection between the two circles [Colour figure can be viewed at wileyonlinelibrary.com]

This yields the overall collision probability

$$p_{\text{col},k} = \begin{cases} 1 & \text{if } d_k + w_{\text{max},k-1} < r_{\text{comb}}, \\ 0 & \text{if } d_k - w_{\text{max},k-1} \geq r_{\text{comb}}, \\ (C31) & \text{otherwise.} \end{cases} \quad (C11)$$

For reasons of readability, the dependence on k for \mathbf{p}_{int} , $\theta_{\text{int},1}$, r_{int} , and a is omitted.