Stochastic Model Predictive Control of Autonomous Systems with Non-Gaussian Correlated Uncertainty

Huishan Chen and Zheng Zhang

Abstract— **Many systems such as autonomous vehicles and quadrotors are subject to parametric uncertainties and external disturbances. These uncertainties can lead to undesired performance degradation and safety issues. Therefore, it is important to design robust control strategies to safely regulate the dynamics of a system. This paper presents a novel framework for chance-constrained stochastic model predictive control of dynamic systems with non-Gaussian correlated probabilistic uncertainties. We develop a new stochastic Galerkin method to propagate the uncertainties using a new type of basis functions and an optimization-based quadrature rule. This formulation can easily handle non-Gaussian correlated uncertainties that are beyond the capability of generalized polynomial chaos expansions. The new stochastic Galerkin formulation enables us to convert a chance-constraint stochastic model predictive control problem into a deterministic one. We verify our approach by several stochastic control tasks, including obstacle avoidance, vehicle path following, and quadrotor reference tracking.**

I. INTRODUCTION

Model predictive control (MPC, also known as receding horizon control) has been widely used to predict and control the future events of a system in an uncertain environment. The representative applications of MPC include chemical process control [1], economics [2], path planning and obstacle avoidance of vehicles and robots [3]–[8], and air traffic management [9]. One popular strategy for MPC is the robust MPC formulation. With the assumption that uncertainty belongs to a bounded set, robust MPC analyzes the stability and performance of the system against worst-case perturbations [10], [11]. However, design based on worst-case uncertainties can be over-conservative in practice and may lead to in-feasibility in real applications.

In contrast to robust MPC, stochastic MPC with probabilistic constraints (also known as chance constraints) incorporates probabilistic descriptions of constraint violations, and allows for acceptable levels of risk during operations [12]–[14]. Chance-constraint stochastic MPC avoids over-conservative decision making by directly incorporating the trade-offs between closed-loop performance and control constraints. One challenge in stochastic MPC is the computational cost of propagating uncertainties. A common approach is to use random sampling methods [15], [16] such as Monte Carlo. The system model is simulated repeatedly based on the samples to predict the time evolution of the uncertainty. Even though sampling-based approaches are applicable to most problems, these techniques can be extremely expensive due to the large number of samples required to achieve accurate uncertainty propagation.

Polynomial chaos expansions provide an efficient alternative to propagate the uncertainties through the system dynamics [17]–[20]. Based on the generalized polynomial chaos theory [21], some popular distributions such as Gaussian distributions and independent uniform distributions have been handled successfully in stochastic MPC. These techniques have been applied in the trajectory optimization of quadrotors and oscillators [20], and chemical process control [19]. However, uncertainties are often non-Gaussian correlated in practice and do not follow the common probability distributions described in [21]. Probabilistic reachable sets have been used by [22] to overestimate the impact caused by correlated external disturbances. However, accurate uncertainty propagation for systems with non-Gaussian correlated parametric uncertainties has not been considered.

Paper Contributions. Inspired by a new stochastic collocation technique described in [23], [24], this paper presents an efficient stochastic MPC framework that can deal with more realistic non-Gaussian correlated uncertainties. Our specific contributions include:

- A novel stochastic Galerkin method to propagate the uncertainties of a dynamic system with non-Gaussian correlated uncertainties. Traditional stochastic Galerkin methods [25] assume that uncertain parameters are mutually independent or Gaussian correlated, thus cannot handle the challenging cases presented in this paper.
- We apply our stochastic Galerkin formulation to develop a chance-constraint stochastic model predictive control solver that can handle non-Gaussian correlated uncertainties efficiently and accurately.
- Numerical results. We verify our proposed framework by solving obstacle avoidance, vehicle path following, and quadrotor reference tracking problems that involve non-Gaussian correlated uncertainties.

II. CHANCE CONSTRAINED STOCHASTIC MPC

A. Problem Formulation

Consider a stochastic discrete-time linear system:

$$\mathbf{x}_{t+1} = \mathbf{A}(\boldsymbol{\xi})\mathbf{x}_t + \mathbf{B}(\boldsymbol{\xi})\mathbf{u}_t + \mathbf{D}(\boldsymbol{\xi})\boldsymbol{\omega}_t, \forall t = 0, \dots, T-1, \quad (1)$$

where $\mathbf{x}_t \in \mathbb{R}^{n_\mathbf{x}}$, $\mathbf{u}_t \in \mathbb{R}^{n_\mathbf{u}}$, and $\boldsymbol{\omega}_t \in \mathbb{R}^{n_\boldsymbol{\omega}}$ denote the stochastic system states, inputs, and disturbances at current time, respectively. The system has probabilistic uncertainty

This work was supported by NSF CCF No. 1763699, NSF CAREER Award No. 1846476, and UCSB Academic Senate Faculty Research Grant.

Huishan Chen and Zheng Zhang are with Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106, USA. huishan_chen@ucsb.edu, zhengzhang@ece.ucsb.edu.

in the system parameters, characterized by $\mathbf{A}(\boldsymbol{\xi})$, $\mathbf{B}(\boldsymbol{\xi})$, and $\mathbf{D}(\boldsymbol{\xi})$, which depend on the uncertain parameter $\boldsymbol{\xi} \in \mathbb{R}^d$. Assume that the joint probability density function (PDF) of random variable $\boldsymbol{\xi}$ is known as $\mathrm{PDF}(\boldsymbol{\xi})$. Assume that $\boldsymbol{\omega}_t$ depends on $\boldsymbol{\xi}$ and its PDF is also known at every time point t. Because of the stochasiticity of parameters and disturbances, the system trajectory will also be stochastic.

Chance constraint is an efficient technique for solving control or optimization problems with uncertainty [26]. Unlike a worst-case optimization where all constraints are satisfied with probability 1, a chance-constraint optimization ensures the probability of satisfying a control/optimization constraint is above a certain confidence level β:

$$\Pr\left[\mathbf{x}_t \notin \mathcal{F}_{\mathbf{x}}\right] \geq \beta, \tag{2}$$

where $\mathcal{F}_x$ denotes the forbidden (e.g., unsafe) region for the state vector $\mathbf{x}_t$, and β is the confidence level.

In finite-horizon stochastic MPC, the goal is to determine a control policy $\boldsymbol{\mu}_T \triangleq (\mathbf{u}_0, ..., \mathbf{u}_{T-1})$ that can drive the state vector $\mathbf{x}_t$ to have a desirable statistical performance. The optimal control can be solved from the following optimization:

Problem 1: Stochastic MPC with chance constraints

$$\begin{aligned} &\min_{\boldsymbol{\mu}_T} \mathbb{E}\left[\sum_{t=1}^{T} \mathbf{x}_t^T \mathbf{Q}_t \mathbf{x}_t + \mathbf{u}_{t-1}^T \mathbf{R}_t \mathbf{u}_{t-1}\right] \\ &\text{s.t. } \mathbf{x}_{t+1} = \mathbf{A}(\boldsymbol{\xi})\mathbf{x}_t + \mathbf{B}(\boldsymbol{\xi})\mathbf{u}_t + \mathbf{D}(\boldsymbol{\xi})\boldsymbol{\omega}_t, \\ &\quad \Pr\left[\mathbf{x}_t \notin \mathcal{F}_{\mathbf{x}}\right] \geq \beta, \\ &\quad \mathbf{u}_t \in \mathcal{U},\ \mathbf{x}_0 = \mathbf{x}_{\text{init}}\left(\boldsymbol{\xi}\right), \\ &\quad \boldsymbol{\xi} \sim f_{\boldsymbol{\xi}}, \quad \boldsymbol{\omega}_t \sim f_{\boldsymbol{\omega}_t}, \end{aligned} \tag{3}$$

where the initial condition $\mathbf{x}_{\text{init}}\left(\boldsymbol{\xi}\right)$ is given and can be uncertain, $\mathcal{U} \subset \mathbb{R}^{n_\mathbf{u}}$ is the compact set of input constraints, $\mathbf{Q}_t$ and $\mathbf{R}_t$ are positive definite weight matrices, $f_{\boldsymbol{\xi}}$ and $f_{\boldsymbol{\omega}_t}$ denote the PDFs of $\boldsymbol{\xi}$ and $\boldsymbol{\omega_t}$, respectively.

B. Polynomial Chaos-Based Stochastic MPC

In order to obtain $\mathbf{x}_t(\boldsymbol{\xi})$ in the MPC, one can employ a truncated generalized polynomial-chaos expansion [21]:

$$\mathbf{x}_t(\boldsymbol{\xi}) \approx \sum_{k=1}^{N_p} \mathbf{c}_{k,t} \Psi_k(\boldsymbol{\xi}), \tag{4}$$

where $\Psi_k(\boldsymbol{\xi})$ is the orthonormal multivariate basis function satisfying $\langle \Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \rangle = \mathbb{E}\left[\Psi_k(\boldsymbol{\xi})\Psi_j(\boldsymbol{\xi})\right] = \delta_{k,j}$. $\delta_{k,j} = 1$ if $k = j$ and otherwise $\delta_{k,j} = 0$. If we upper bound the total polynomial order by p, then the total number of basis functions is $N_p = \frac{(p+d)!}{(p!d!)}$. With the above truncated generalized polynomial-chaos expansion, the time-dependent deterministic coefficients $\mathbf{c}_{k,t}$ can be computed efficiently via numerical schemes such as stochastic collocation [27], stochastic Galerkin [25], or stochastic testing [28]. Finally, the formulation in (3) can be converted to a deterministic one and solved efficiently [5], [17], [19].

Limitations. Despite their high efficiency, existing polynomial chaos-based stochastic MPC [17]–[20] is limited by a strong assumption of generalized polynomial chaos [21]: the uncertain parameters $\boldsymbol{\xi}$ should be mutually independent or just Gaussian correlated (which can be easily de-correlated). In practice, $\boldsymbol{\xi}$ can be non-Gaussian correlated, thus the basis functions proposed in [21] cannot be employed. Furthermore, the numerical solvers such as stochastic collocation [27] and stochastic Galerkin [25] reply on fast numerical integration rules such as Gaussian quadrature [29] or spare grid [30], which fail for non-Gaussian correlated parameters as well.

III. Stochastic Galerkin with Non-Gaussian Correlated Uncertainties

In order to handle non-Gaussian correlated uncertainties in a stochastic MPC problem efficiently, this section proposes a new stochastic Galerkin formulation. Our formulation extends the work of [23], [24]. Note that the methods in [23], [24] are stochastic collocation, and they are less efficient than our method when solving time-evolving problems.

A. Deterministic Formulation via Stochastic Galerkin

Since the system parameters and disturbances are stochastic variables, we express them with truncated expansion of stochastic basis functions:

$$\mathbf{u}_t(\boldsymbol{\xi}) \approx \sum_{k=1}^{N_p} \mathbf{h}_{k,t} \Psi_k(\boldsymbol{\xi}), \;\; \boldsymbol{\omega}_t \approx \sum_{k=1}^{N_p} \mathbf{w}_{k,t} \Psi_k(\boldsymbol{\xi}). \tag{5}$$

If the control input $\mathbf{u}_t$ is deterministic, then $\mathbf{h}_{1,t} = \mathbf{u}_t$ and all other coefficients $\mathbf{h}_{k,t} (k \neq 1)$ are set to zero. Based on the above expansions, (1) can be rewritten as:

$$\begin{aligned} \sum_{k=1}^{N_p} \mathbf{c}_{k,t+1} \Psi_k(\boldsymbol{\xi}) &\approx \sum_{k=1}^{N_p} \mathbf{c}_{k,t} \mathbf{A}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}) \\ &+ \sum_{k=1}^{N_p} \mathbf{h}_{k,t} \mathbf{B}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}) + \sum_{k=1}^{N_p} \mathbf{w}_{k,t} \mathbf{D}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}). \end{aligned} \tag{6}$$

Stochastic Galerkin can reformulate the stochastic system into a deterministic one by enforcing the residual of (6) orthogonal to each stochastic basis function. By Galerkin projection, for each $j = 1, ..., N_p$, we have

$$\begin{aligned} \mathbf{c}_{j,t+1} \approx &\sum_{k=1}^{N_p} \mathbf{c}_{k,t} \left\langle \mathbf{A}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \right\rangle \\ &+ \sum_{k=1}^{N_p} \mathbf{h}_{k,t} \left\langle \mathbf{B}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \right\rangle \\ &+ \sum_{k=1}^{N_p} \mathbf{w}_{k,t} \left\langle \mathbf{D}(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \right\rangle, \end{aligned} \tag{7}$$

where the inner product of $f_1(\boldsymbol{\xi})$ and $f_2(\boldsymbol{\xi})$ equals:

$$\langle f_1(\boldsymbol{\xi}), f_2(\boldsymbol{\xi}) \rangle = \int_{\mathbb{R}^d} f_1(\boldsymbol{\xi}) f_2(\boldsymbol{\xi}) \mathrm{PDF}(\boldsymbol{\xi}) d\boldsymbol{\xi}. \tag{8}$$

Let us define

$$\begin{aligned} \hat{\mathbf{x}}_t &= \begin{bmatrix} \mathbf{c}_{1,t} & \cdots & \mathbf{c}_{N_p,t} \end{bmatrix}^T, \\ \hat{\mathbf{u}}_t &= \begin{bmatrix} \mathbf{h}_{1,t} & \cdots & \mathbf{h}_{N_p,t} \end{bmatrix}^T, \\ \hat{\boldsymbol{\omega}}_t &= \begin{bmatrix} \mathbf{w}_{1,t} & \cdots & \mathbf{w}_{N_p,t} \end{bmatrix}^T. \end{aligned} \tag{9}$$

The Galerkin projection in Equation (7) generates the following deterministic dynamic system:

$$\hat{\mathbf{x}}_{t+1} = \hat{\mathbf{A}}\hat{\mathbf{x}}_t + \hat{\mathbf{B}}\hat{\mathbf{u}}_t + \hat{\mathbf{D}}\hat{\boldsymbol{\omega}}_t, \tag{10}$$

where the vectors $\hat{\mathbf{x}}_t \in \mathbb{R}^{n_\mathbf{x} N_p}$ and $\hat{\boldsymbol{\omega}}_t \in \mathbb{R}^{n_\boldsymbol{\omega} N_p}$ contain their truncated expansion coefficients at time t. The matrices $\hat{\mathbf{A}} \in \mathbb{R}^{n_\mathbf{x} N_p \times n_\mathbf{x} N_p}$, $\hat{\mathbf{B}} \in \mathbb{R}^{n_\mathbf{x} N_p \times n_\mathbf{u} N_p}$, $\hat{\mathbf{D}} \in \mathbb{R}^{n_\mathbf{x} N_p \times n_\boldsymbol{\omega} N_p}$ are matrices with $N_p \times N_p$ blocks. Their (k,j)-th blocks can be computed off-line by the following formula:

$$\begin{aligned}
\hat{\mathbf{A}}_{k,j} &= \Big\langle \mathbf{A}(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \Big\rangle \in \mathbb{R}^{n_\mathbf{x} \times n_\mathbf{x}},\\
\hat{\mathbf{B}}_{k,j} &= \Big\langle \mathbf{B}(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \Big\rangle \in \mathbb{R}^{n_\mathbf{x} \times n_\mathbf{u}},\\
\hat{\mathbf{D}}_{k,j} &= \Big\langle \mathbf{D}(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \Big\rangle \in \mathbb{R}^{n_\mathbf{x} \times n_\boldsymbol{\omega}}.
\end{aligned} \tag{11}$$

As shown in Eq. (11), we need to evaluate a set of inner products in order to set up the deterministic dynamic system model (10). The inner product calculation requires an accurate numerical integration method. For a general smooth function $g(\boldsymbol{\xi})$, its integration can be evaluated with a set of quadrature points and weights $\{\boldsymbol{\xi}_l, w_l\}_{l=1}^M$ as follows:

$$\int_{\mathbb{R}^d} g(\boldsymbol{\xi})\text{PDF}(\boldsymbol{\xi})d\boldsymbol{\xi} \approx \sum_{l=1}^{M} g(\boldsymbol{\xi}_l)w_l. \tag{12}$$

There exist two challenges in the stochastic Galerkin formulation. Firstly, how shall we choose the stochastic basis functions $\{\Psi_k(\boldsymbol{\xi})\}$ such that they can capture the impacts caused by non-Gaussian correlated uncertainties? Secondly, how shall we choose the quadrature points and weights in the correlated parameter space? We will employ the methods of [23], [24] in our stochastic Galerkin formulation, as elaborated below.

B. Selection of Basis Functions

The generalized polynomial chaos [21] cannot be used when the parameters $\boldsymbol{\xi}$ are non-Gaussian correlated. In this case, orthonormal basis functions $\{\Psi_k(\boldsymbol{\xi})\}$ can be obtained based on a multi-variate Gram-Schmidt process [23], [24]:

1) Reorder the monomials $p_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \xi_1^{\alpha_1} \ldots\ \xi_d^{\alpha_d}$ in the graded lexicographic order and denote them as $\{p_k(\boldsymbol{\xi})\}$.
2) Set $\Psi_1(\boldsymbol{\xi}) = 1$, calculate the orthonormal polynomials $\{\Psi_k(\boldsymbol{\xi})\}_{k=1}^{N_p}$ recursively:

$$\begin{aligned}
\hat{\Psi}_k(\boldsymbol{\xi}) &= p_k(\boldsymbol{\xi}) - \sum_{i=1}^{k-1} \mathbb{E}\left[p_k(\boldsymbol{\xi})\Psi_i(\boldsymbol{\xi})\right]\Psi_i(\boldsymbol{\xi}),\\
\Psi_k(\boldsymbol{\xi}) &= \frac{\hat{\Psi}_k(\boldsymbol{\xi})}{\sqrt{\mathbb{E}\left[\hat{\Psi}_k^2(\boldsymbol{\xi})\right]}}, \quad k = 2, ..., N_p.
\end{aligned} \tag{13}$$

Once the dynamic system (10) is set up and solved, the statistics of the stochastic variable $\mathbf{x}_t(\boldsymbol{\xi})$ can be efficiently computed. For example, the mean and variance of $\mathbf{x}_t(\boldsymbol{\xi})$ are

$$\mathbb{E}\left[\mathbf{x}_t(\boldsymbol{\xi})\right] = \mathbf{c}_{1,t},\ \text{Var}\left[\mathbf{x}_t(\boldsymbol{\xi})\right] = \sum_{k=2}^{N_p} \mathbf{c}_{k,t}^2. \tag{14}$$

C. Optimization Based Quadrature Rule

To evaluate the matrices in (11), we need some quadrature points and samples such that the integration in (12) can be calculated with a high accuracy for a general smooth function $g(\boldsymbol{\xi})$. We employ the optimization-based quadrature [23], [24] for non-Guassian correlated parameters $\boldsymbol{\xi}$.

Assume that we want to get an (almost) exact integration for a polynomial function $g(\boldsymbol{\xi})$ bounded by degree $2p$. In this case, we can express $g(\boldsymbol{\xi})$ with $N_{2p} = \frac{(2p+d)!}{(2p!d!)}$ basis functions bounded by order-$2p$. The quadrature nodes and weights $\{\boldsymbol{\xi}_l, w_l\}_{l=1}^M$ are determined by enforcing exact integration over these basis functions:

$$\mathbb{E}\left[\Psi_k(\boldsymbol{\xi})\right] = \sum_{l=1}^{M} \Psi_k(\boldsymbol{\xi}_l)w_l, \quad \forall k = 1, ..., N_{2p}. \tag{15}$$

Based on the orthonormal condition for $\Psi_k(\boldsymbol{\xi})$, it is easy to show that $\mathbb{E}\left[\Psi_k(\boldsymbol{\xi})\right] = \mathbb{E}\left[\Psi_k(\boldsymbol{\xi})\Psi_1(\boldsymbol{\xi})\right] = \delta_{1k}$, where $\delta_{1k} = 1$ if $k = 1$ and otherwise $\delta_{1k} = 0$. As a result, (15) is reformulated into a nonlinear least square optimization:

$$\min_{\bar{\boldsymbol{\xi}}, \mathbf{w}} \ \left\| \boldsymbol{\Phi}\left(\bar{\boldsymbol{\xi}}\right)\mathbf{w} - \mathbf{e}_1 \right\|_2^2, \tag{16}$$

where $\bar{\boldsymbol{\xi}} = \left[\boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_M\right]^T \in \mathbb{R}^{M \times d}$, $\mathbf{w} = \left[w_1, ..., w_M\right]^T \in \mathbb{R}^M$, $\mathbf{e}_1 = [1, 0, ..., 0] \in \mathbb{R}^{N_{2p}}$, $\left[\boldsymbol{\Phi}(\bar{\boldsymbol{\xi}})\right]_{kl} = \Psi_k(\boldsymbol{\xi}_l)$ and $\boldsymbol{\Phi}(\bar{\boldsymbol{\xi}}) \in \mathbb{R}^{N_{2p} \times M}$. A block coordinate-descent method was used in [23], [24] to solve this optimization problem.

In order to make the algorithm more efficient, a weighted clustering approach is used in order to automatically reduce the number of quadrature points and achieve desired level of accuracy. Specifically, once (16) is solved, M is reduced by one by a weighted clustering method [24], then (16) is solved again using the reduced samples and weights. This process is repeated until a further reduction leads to failures when solving (16) [23], [24].

IV. APPLICATION IN STOCHASTIC MPC

In this section, we describe how to perform stochastic MPC using the deterministic system (10) obtained by the above stochastic Galerkin formulation.

A. Minimum Expectation Control

We consider the minimum expectation control problem:

$$\min_{\boldsymbol{\mu}_T} \ \mathbb{E}\left[\sum_{t=1}^{T} \mathbf{x}_t^T\mathbf{Q}_t\mathbf{x}_t + \mathbf{u}_{t-1}^T\mathbf{R}_t\mathbf{u}_{t-1}\right], \tag{17}$$

where $\mathbf{Q}_t$ and $\mathbf{R}_t$ are user defined state and input weight matrices satisfying $\mathbf{Q}_t = \mathbf{Q}_t^T > 0$ and $\mathbf{R}_t = \mathbf{R}_t^T > 0$. With a truncated expansion using the stochastic basis functions described in Section III-B, the term $\mathbb{E}\left[\mathbf{x}_t^T\mathbf{x}_t\right]$ is obtained as

$$\mathbb{E}\left[\mathbf{x}_t^T\mathbf{x}_t\right] = \hat{\mathbf{x}}_t^T\left(\mathbf{I}_{n_\mathbf{x}} \otimes \mathbf{V}\right)\hat{\mathbf{x}}_t, \tag{18}$$

Algorithm 1: Chance-Constrained Stochastic MPC solver with Non-Gaussian Correlated Uncertainties

```
Input: PDF of the non-Gaussian correlated random
       parameters $\boldsymbol{\xi}$ and $\boldsymbol{\omega}_t$, system parameters
       $\mathbf{A}(\boldsymbol{\xi})$, $\mathbf{B}(\boldsymbol{\xi})$, and $\mathbf{D}(\boldsymbol{\xi})$, polynomial order $p$,
       final time $T$, initial condition $\mathbf{x}_{\text{init}}(\boldsymbol{\xi})$, weight
       matrices $\mathbf{Q}_t$ and $\mathbf{R}_t$, input constraints set $\mathcal{U}$,
       forbidden regions $\mathcal{F}_{\mathbf{x}}$, confidence level $\beta$.
  1. Construct the basis functions $\Psi_k(\boldsymbol{\xi})$ by (13).
  2. Construct the quadrature points $\boldsymbol{\xi}_l$ and weights $w_l$ by
     (16) and weighted clustering.
  3. Set up the deterministic system (10) via (11).
  4. Convert the objective function and chance constraints
     into deterministic ones by (19) and (21), respectively.
  5. Solve the optimization problem (22).
Output: Optimized input signals $\mathbf{u}_t^*$, and the
        time-dependent stochastic state $\mathbf{x}_t^*$.
```

where $\mathbf{I}_{n_\mathbf{x}} \in \mathbb{R}^{n_\mathbf{x} \times n_\mathbf{x}}$ is an identity matrix, $\otimes$ is a Kronecker product, and $\mathbf{V} \in \mathbb{R}^{N_p \times N_p} = \{v_{ij}\}$ with $v_{ij} = \sum_{l=1}^{M} \Psi_i(\boldsymbol{\xi}_l)\Psi_j(\boldsymbol{\xi}_l)w_l$. Based on (18) the cost function in (17) can now be written in terms of deterministic variables:

$$\min_{\boldsymbol{\mu}_T} \sum_{t=1}^{T} \hat{\mathbf{x}}_t \hat{\mathbf{Q}}_t \hat{\mathbf{x}}_t + \hat{\mathbf{u}}_{t-1} \hat{\mathbf{R}}_t \hat{\mathbf{u}}_{t-1}, \tag{19}$$

where $\hat{\mathbf{Q}}_t = \mathbf{Q}_t \otimes \mathbf{V}$ and $\hat{\mathbf{R}}_t = \mathbf{R}_t \otimes \mathbf{V}$.

B. Deterministic Optimization for Stochastic MPC

Assume that the forbidden region in (3) is described by some inequalities, and the chance constraints can be written in the following form:

$$\Pr\left[g(\mathbf{x}_t(\boldsymbol{\xi})) \leq 0\right] \geq \beta. \tag{20}$$

This probabilistic constraint can be reformulated as some deterministic constraints [31] by using the mean and variance of $g(\mathbf{x}_t(\boldsymbol{\xi}))$:

$$\mathbb{E}\left[g(\mathbf{x}_t(\boldsymbol{\xi}))\right] + \kappa_{1-\beta}\sqrt{\text{Var}\left[g(\mathbf{x}_t(\boldsymbol{\xi}))\right]} \geq 0, \tag{21}$$

where $\mathbb{E}[\cdot]$ and $\text{Var}[\cdot]$ are obtained via (14). The constant $\kappa_{1-\beta}$ is chosen as $\kappa_{1-\beta} = \sqrt{\beta/(1-\beta)}$.

With (10), (19) and (21), we are ready to re-write Problem 1 as a deterministic optimization problem:

Problem 2: Deterministic Optimization for Stochastic MPC:

$$\begin{aligned}
\min_{\boldsymbol{\mu}_T} \quad & \sum_{t=1}^{T} \hat{\mathbf{x}}_t \hat{\mathbf{Q}}_t \hat{\mathbf{x}}_t + \hat{\mathbf{u}}_{t-1} \hat{\mathbf{R}}_t \hat{\mathbf{u}}_{t-1} \\
\text{s.t.} \quad & \hat{\mathbf{x}}_{t+1} = \hat{\mathbf{A}}\hat{\mathbf{x}}_t + \hat{\mathbf{B}}\hat{\mathbf{u}}_t + \hat{\mathbf{D}}\hat{\boldsymbol{\omega}}_t, \\
& \mathbb{E}\left[g(\mathbf{x}_t(\boldsymbol{\xi}))\right] + \kappa_{1-\beta}\sqrt{\text{Var}\left[g(\mathbf{x}_t(\boldsymbol{\xi}))\right]} \geq 0, \\
& \mathbf{u} \in \mathcal{U}, \quad \hat{\mathbf{x}}_0 = \hat{\mathbf{x}}_{\text{init}},
\end{aligned} \tag{22}$$

where $\hat{\mathbf{x}}_{\text{init}}$ is the expanded initial condition containing the expansion coefficients.

The whole framework is summarized in Alg. 1.

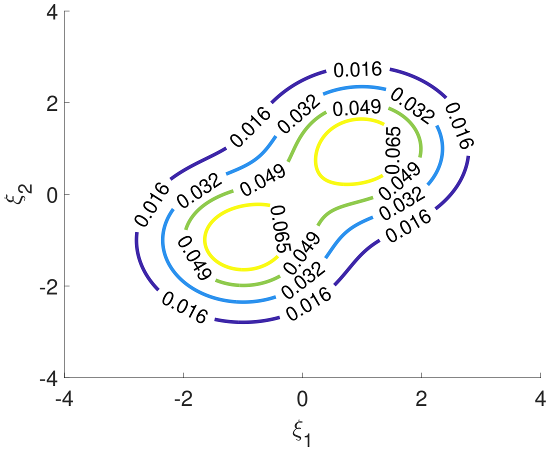

Fig. 1: Joint PDF of random variables ξ_1 and ξ_2 for the obstable avoidance example.

V. NUMERICAL RESULTS

In order to verify our proposed chance constrained stochastic MPC method, we implement the algorithm in MATLAB on a Windows Desktop Workstation with 8-GB RAM and a 3.4-GHz CPU.

A. Obstacles Avoidance

Consider a stochastic linear time-invariant system

$$\mathbf{x}_{t+1} = \begin{bmatrix} 0.9 + \rho_1\xi_1 & 0.1 \\ 0.1 & 0.85 \end{bmatrix} \mathbf{x}_t + \begin{bmatrix} 0.25 - \rho_1\xi_1 \\ 0.75 + \rho_2\xi_2 \end{bmatrix} \mathbf{u}_t, \tag{23}$$

with initial condition $\mathbf{x}_0 = [20, 10]^T$. The non-Gaussian correlated random variables ξ_1 and ξ_2 obey a Gaussian-mixture distribution, as shown in Fig. 1. Here ρ_1 and ρ_2 are the weights on the random variables. The prediction horizon is $T = 4$ sec, and input constraint $\mathbf{u}_t \in [-0.5, 0.5]$. The constraints describe an infeasible region shown in Fig. 2(a).

Consider $\mathbf{Q}_t = \text{diag}(100, 100)$ and $\mathbf{R}_t = 1$ for all t, we use Alg. 1 to solve the problem with a 99% level of confidence for avoiding the infeasible region. The resultant controlled trajectory generated without parameter uncertainty, i.e., $\rho_1 = \rho_2 = 0$, is shown in Fig. 2(a), which stays in the feasible region as desired. Consider the system with parameter uncertainty, i.e., $\rho_1 = 0.001$ and $\rho_2 = 0.05$. Fig. 2(b) shows the controlled system trajectories generated with our proposed framework, where the red line shows the mean trajectory obtained via (14).

B. Vehicle Path-following Problem

Path-following control aims to enforce the system states converge to a given path, without temporal specifications [32]–[35]. We consider the vehicle path-following model in [36], where the state variables include position error, orientation error, and their first order derivative with respect to time. The dynamic model for vehicle path following is generally nonlinear [36], [37]. With the assumption that the vehicle is moving at a constant speed of 20 m/s and the influence of road bank angle is neglected [36], the

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t),$$
$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_f+2C_r}{mV_x} & \frac{2C_f+2C_r}{m} & -\frac{2aC_f+2bC_r}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2aC_f+2bC_r}{I_{zz}V_x} & \frac{2aC_f-2bC_r}{I_{zz}} & -\frac{2a^2C_f+2b^2C_r}{I_{zz}V_x} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ \frac{2C_f}{m} & \frac{2aC_f-2bC_r}{mV_x} - V_x \\ 0 & 0 \\ \frac{2aC_f}{I_{zz}} & \frac{2a^2C_f+2b^2C_r}{I_{zz}V_x} \end{bmatrix}, \tag{24}$$
$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{x}(t) = \begin{bmatrix} e_1(t) & \dot{e}_1(t) & e_2(t) & \dot{e}_2(t) \end{bmatrix}^T, \quad \mathbf{u}(t) = \begin{bmatrix} \delta(t) & \dot{\psi}_{\text{des}}(t) \end{bmatrix}.$$

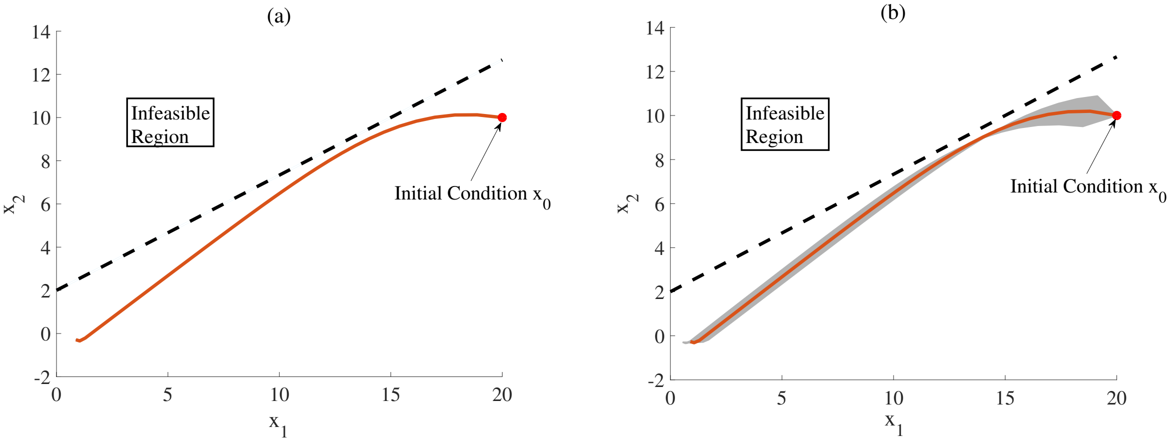

Fig. 2: Controlled trajectories for the obstacles avoidance experiment. (a): without parameter uncertainty; (b): with non-Gaussian correlated uncertainty. Red line shows the controlled trajectory or its mean value, black dotted line separates the feasible and infeasible regions, and grey lines show the trajectory uncertainties.

TABLE I: Parameters of the vehicle path-following model.

Symbol	Description	Value
V_x	vehicle speed	20 m/s
m	vehicle mass	1270 kg
a	distance from center to front axis	1.015 m
b	distance from center to rear axis	1.895 m
I_{zz}	vehicle yaw inertia	1536.7 kg·m^2
g	gravitational acceleration	9.81 m/s^2

vehicle can be described by the linear state-space model shown in Eq. (24). Here $e_1(t)$, $e_2(t)$, and $\delta(t)$ represent the lateral deviation of the center of mass of the vehicle from the desired path, deviation of yaw angle from the desired yaw angle $\psi_{\text{des}}(t)$, and front steering wheel angle, respectively. The desired yaw rate is defined by $\dot{\psi}_{\text{des}} = \frac{V_x}{R(t)}$, where $R(t)$ is the radius of a desired path. C_f and C_r are the front and the rear tire cornering stiffness values, respectively. The values of C_f and C_r are uncertain since they are influenced by the tire slip angle. Therefore, instead of assigning them with deterministic values, we treat them as random variables and assume that they satisfy a Gaussian-mixture joint distribution. In this experiment, we assume that they have the same nominal value 967 N/deg. Other parameters of the vehicle model are taken from CarSim (C-Class Hatchback 2017) [36].

Our goal is to enforce the output $\mathbf{y}(t) = e_1(t)$ (i.e., lateral position error) to zero by controlling the steering wheel angle $\delta(t)$. The constraints imposed on the state vector are based on a 99% level of confidence and are specified below:

$$e_1(t) \in [-1, 1], \;\; \dot{e}_1(t) \in [-10, 10],$$
$$e_2(t) \in [-28.65, 28.65], \;\; \dot{e}_2(t) \in [-572.96, 572.96].$$

Assuming that the vehicle starts with 1m of a lateral position error, we intend to calculate the optimal steering angle by minimizing the following objective function:

$$\sum_{t=1}^{T} \mathbf{x}_t^T \mathbf{Q} \mathbf{x}_t + \mathbf{u}_{t-1}^T \mathbf{R} \mathbf{u}_{t-1} + (\mathbf{y}_t - \mathbf{y}_{\text{ref},k})^T \mathbf{S} (\mathbf{y}_t - \mathbf{y}_{\text{ref},k}).$$

The weighting matrices $\mathbf{Q}$, $\mathbf{R}$, and $\mathbf{S}$ are selected as follows:

$$\mathbf{Q} = \begin{bmatrix} 7000 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 20000 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{R} = \mathbf{I}, \quad \mathbf{S} = 100.$$

We employ Alg. 1 to solve this path-following problem, and the resulting lateral error and control input are plotted in Fig. 3(a) and Fig. 3(b), respectively. The lateral position error goes to zeros, which indicates that the vehicle is travelling along the desired path.

C. Reference Tracking of Quadrotor

Finally we consider the control of a quadrotor mini-helicopter described in [38], which consists of four propulsion rotors in a cross configuration. The dynamics of this quadrotor can be described by a nonlinear model with 12 state variables:

$$\dot{\mathbf{q}} = f(\mathbf{q}, \mathbf{u}), \tag{25}$$

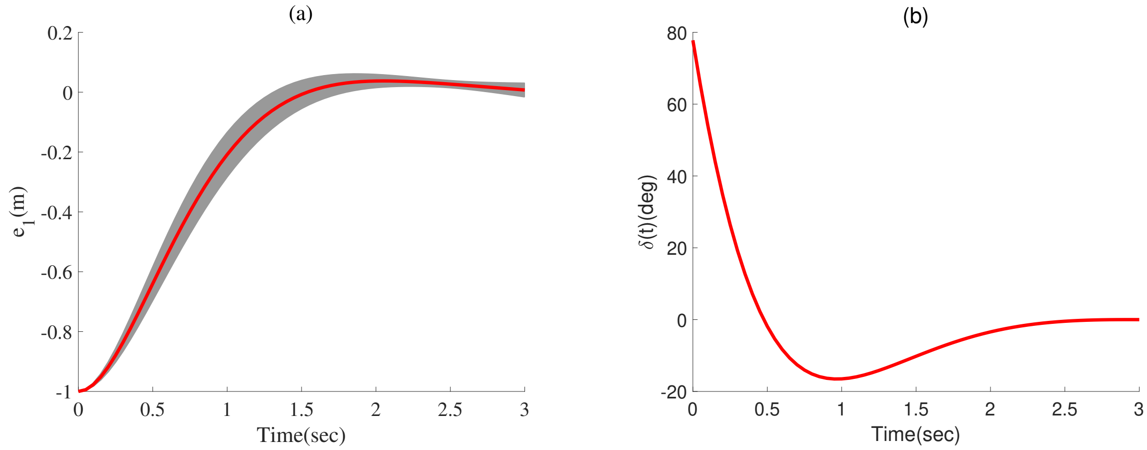

Fig. 3: Numerical results for vehicle path following experiment. (a): resulting lateral position error, where the red line shows the mean value, and the grey lines show the trajectory uncertainties; (b): Optimal control input $\delta(t)$.

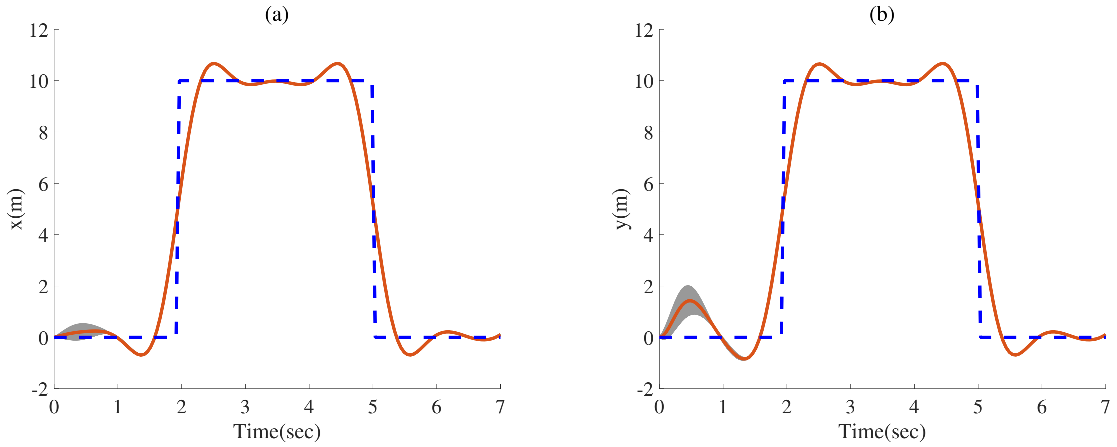

Fig. 4: The Quadrotor tracks step reference. (a): in x direction; (b): in y direction. Red line shows the mean controlled trajectory, the grey line show the effects caused by the uncertainties, and blue dotted line is the reference trajectory.

where $\mathbf{q} = \left[x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}\right]^T$ is the state vector consisting of Cartesian positions x, y, and z (in meters), the attitude angles ϕ (pitch), θ (roll), ψ (yaw) in radians, and their respective rates $\left(\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}\right)$. Vector $\mathbf{u} = \left[\Omega_1, \Omega_2, \Omega_3, \Omega_4\right]$ is the control input which includes the four propellers' rotational speeds (in radians per second) and Ω_i is the rotation imposed on the i-th motor.

In this experiment, we approximate the quadrotor's dynamics by a linear model around an equilibrium point. The equilibrium point $\mathbf{q}_{\text{eq}} = \left[0, 0, 0, 0, 10, 0, 0, 0, 0, 0, 0, 0\right]^T$ and $\mathbf{u}_{\text{eq}} = \left[192.8, 192.8, 192.8, 192.8\right]^T$ are used to linearize the system, and this is corresponding to a hover flight condition at a 10m height [38]. With this equilibrium point, the resulting linear state-space model is:

$$\mathbf{q}_{t+1} = \mathbf{A}\mathbf{q}_t + \mathbf{B}\mathbf{u}_t, \ \mathbf{y}_t = \mathbf{C}\mathbf{q}_t, \tag{26}$$

where the model matrices are specified in [38].

Since there are disturbances acting on the quadrotor, we consider the linear quadrotor system with additive disturbances impose on the three Cartesian positions:

$$\mathbf{q}_{t+1} = \mathbf{A}\mathbf{q}_t + \mathbf{B}\mathbf{u}_t + \mathbf{D}\boldsymbol{\omega}_t(\boldsymbol{\xi}), \tag{27}$$

where $\boldsymbol{\xi}$ obeys a Gaussian-mixture distribution and matrix $\mathbf{D} \in \mathbb{R}^{12\times 3}$.

The constraints were imposed on ϕ and θ such that $\phi_{\max} = 5°$, $\phi_{\min} = -5°$, $\theta_{\max} = 5°$, and $\theta_{\min} = -5°$. We first let the quadrotor to track 10-m step references in the x and y directions by keeping the elevation at a fixed operating level ($z = 10$ m). By using Algorithm 1, we obtain the controlled x and y in Fig. 4(a) and 4(b), respectively.

Next, we illustrate that the stability of the control loop by letting the quadrotor to track the following reference trajectory:

$$x_r(t) = 2\cos(0.2t), \quad y_r(t) = 2\sin(0.2t), \quad z_r(t) = 0.2t, \tag{28}$$

where x_r, y_r, and z_r denote the reference trajectories in x, y, ans z directions, respectively. The quadrotor's behavior following the defined trajectory in xyz plane is shown in

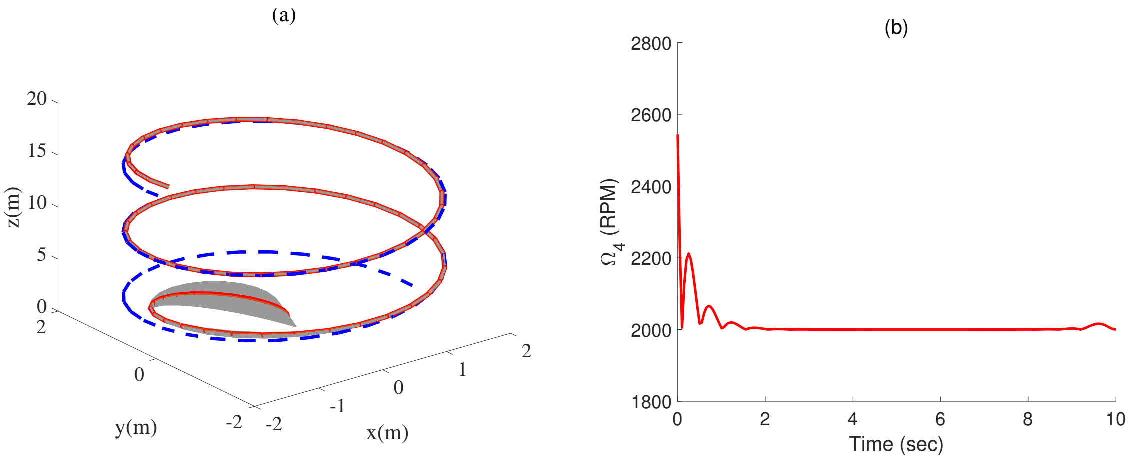

Fig. 5: The quadrotor tracks the reference trajectory in (28). (a): quadrotor trajectories. Red line: mean controlled trajectory. Grey: uncertainties of the controlled trajectory. Blue dotted line: reference trajectory. (b): optimal control input Ω_4.

TABLE II: CPU Time comparison of stochastic MPC solvers.

Examples	Proposed	Monte Carlo
Obstacle Avoidance	250s	2.9×10^3s
Vehicle Path-following	620s	8.5×10^3s
Quadrotor Tracking A Step Reference	1300s	10^4s
Quadrotor Tracking Reference (28)	1200s	7000 s

Fig. 5. The simulation result shows that the controller is able to make the quadrotor follow the desired path successfully.

D. Comparison with Monte Carlo-based MPC

We compare our proposed algorithm 1 with Monte Carlo-based MPC approach in terms of CPU time. We use 5000 sample points for Monte Carlo-based MPC. The CPU time required by two methods are summarized in Table II. Due to the choice of specialized stochastic Galerkin formulation, our method can accurately capture the uncertainties caused by non-Gaussian correlated uncertainties. For instance, Fig. 6 shows the PDF of x_1 at $t = 2$ s for the obstacle avoidance example, where the result from our method is indistinguishable from that of Monte Carlo.

Our method is not compared with existing polynomial chaos-based MPC because the latter cannot handle non-Gaussian correlated uncertainties.

VI. CONCLUSION

This paper has presented a method for solving chance constrained stochastic MPC problem under non-Gaussian correlated uncertainties. With the proposed stochastic Galerkin formulation, the propagation of uncertainties through the system model can be efficiently and accurately represented. Our framework has reformulated the stochastic system model, objective function and constraints into deterministic ones, resulting in a deterministic optimization problem. This method has been verified on three benchmarks. On these benchmarks, our technique have efficiently handled the non-Gaussian correlated uncertainties and led to a significant reduction of CPU time compared with the Monte Carlo-based model predictive control.

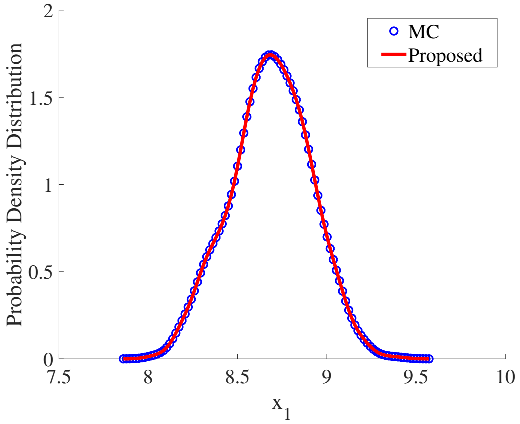

Fig. 6: Probability density functions of x_1 at $t = 2$ s for the obstacle avoidance example.

ACKNOWLEDGEMENT

The authors would like to thank Dr. Chunfeng Cui and Mr. Zichang He for their technical discussions and for their help in proofreading this manuscript.

REFERENCES

[1] P. Li, M. Wendt, and G. Wozny, "Robust model predictive control under chance constraints," *Computers & Chemical Engineering*, vol. 24, no. 2-7, pp. 829–834, 2000.

[2] F. Herzog, G. Dondi, and H. P. Geering, "Stochastic model predictive control and portfolio optimization," *International Journal of Theoretical and Applied Finance*, vol. 10, no. 02, pp. 203–233, 2007.

[3] L. Blackmore, M. Ono, and B. C. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1080–1094, 2011.

[4] E.-H. Guechi, S. Bouzoualegh, L. Messikh, and S. Blažic, "Model predictive control of a two-link robot arm," in *2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*. IEEE, 2018, pp. 409–414.

[5] K.-K. K. Kim and R. D. Braatz, "Generalised polynomial chaos expansion approaches to approximate stochastic model predictive control," *International journal of control*, vol. 86, no. 8, pp. 1324–1337, 2013.

[6] M. A. Abbas, R. Milman, and J. M. Eklund, "Obstacle avoidance in real time with nonlinear model predictive control of autonomous vehicles," *Canadian journal of electrical and computer engineering*, vol. 40, no. 1, pp. 12–22, 2017.

[7] C. Liu, A. Carvalho, G. Schildbach, and J. K. Hedrick, "Stochastic predictive control for lane keeping assistance systems using a linear time-varying model," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 3355–3360.

[8] C. Huang, F. Naghdy, and H. Du, "Model predictive control-based lane change control system for an autonomous vehicle," in *2016 IEEE Region 10 Conference (TENCON)*. IEEE, 2016, pp. 3349–3354.

[9] N. Kantas, J. Maciejowski, and A. Lecchini-Visintini, "Sequential monte carlo for model predictive control," in *Nonlinear model predictive control*. Springer, 2009, pp. 263–273.

[10] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in identification and control*. Springer, 1999, pp. 207–226.

[11] S. Garatti and M. C. Campi, "Modulating robustness in control design: Principles and algorithms," *IEEE Control Systems Magazine*, vol. 33, no. 2, pp. 36–51, 2013.

[12] L. Blackmore, M. Ono, A. Bektassov, and B. C. Williams, "A probabilistic particle-control approximation of chance-constrained stochastic predictive control," *IEEE transactions on Robotics*, vol. 26, no. 3, pp. 502–517, 2010.

[13] J. A. Primbs and C. H. Sung, "Stochastic receding horizon control of constrained linear systems with state and control multiplicative noise," *IEEE transactions on Automatic Control*, vol. 54, no. 2, pp. 221–230, 2009.

[14] A. T. Schwarm and M. Nikolaou, "Chance-constrained model predictive control," *AIChE Journal*, vol. 45, no. 8, pp. 1743–1752, 1999.

[15] M. Vidyasagar, "Randomized algorithms for robust controller synthesis using statistical learning theory," *Automatica*, vol. 37, no. 10, pp. 1515–1528, 2001.

[16] A. Shapiro, "Stochastic programming approach to optimization under uncertainty," *Mathematical Programming*, vol. 112, no. 1, pp. 183–220, 2008.

[17] J. Fisher and R. Bhattacharya, "Optimal trajectory generation with probabilistic system uncertainty using polynomial chaos," *Journal of dynamic systems, measurement, and control*, vol. 133, no. 1, p. 014501, 2011.

[18] J. R. Fisher, *Stability analysis and control of stochastic dynamic systems using polynomial chaos*. Texas A&M University, 2008.

[19] A. Mesbah, S. Streif, R. Findeisen, and R. D. Braatz, "Stochastic nonlinear model predictive control with probabilistic constraints," in *American Control Conference*, 2014, pp. 2413–2419.

[20] G. I. Boutselis, Y. Pan, and E. A. Theodorou, "Numerical trajectory optimization for stochastic mechanical systems," *SIAM Journal on Scientific Computing*, vol. 41, no. 4, pp. A2065–A2087, 2019.

[21] D. Xiu and G. E. Karniadakis, "The wiener–askey polynomial chaos for stochastic differential equations," *SIAM journal on scientific computing*, vol. 24, no. 2, pp. 619–644, 2002.

[22] L. Hewing, K. P. Wabersich, and M. N. Zeilinger, "Recursively feasible stochastic model predictive control using indirect feedback," *arXiv preprint arXiv:1812.06860*, 2018.

[23] C. Cui, M. Gershman, and Z. Zhang, "Stochastic collocation with non-Gaussian correlated parameters via a new quadrature rule," in *Electrical Performance of Electronic Packaging and Systems (EPEPS)*, 2018, pp. 57–59.

[24] C. Cui and Z. Zhang, "Stochastic collocation with non-Gaussian correlated process variations: Theory, algorithms and applications," *IEEE Trans. Components, Packaging and Manufacturing Technology*, vol. 9, no. 7, pp. 1362–1375, July 2019.

[25] R. G. Ghanem and P. D. Spanos, *Stochastic finite elements: a spectral approach*. Courier Corporation, 2003.

[26] P. Li, H. Arellano-Garcia, and G. Wozny, "Chance constrained programming approach to process optimization under uncertainty," *Computers & chemical engineering*, vol. 32, no. 1-2, pp. 25–45, 2008.

[27] D. Xiu and J. S. Hesthaven, "High-order collocation methods for differential equations with random inputs," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 1118–1139, 2005.

[28] Z. Zhang, T. A. El-Moselhy, I. M. Elfadel, and L. Daniel, "Stochastic testing method for transistor-level uncertainty quantification based on generalized polynomial chaos," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 10, pp. 1533–1545, 2013.

[29] G. H. Golub and J. H. Welsch, "Calculation of gauss quadrature rules," *Math. Comp.*, vol. 23, no. 106, pp. 221–230, 1969.

[30] F. Nobile, R. Tempone, and C. G. Webster, "A sparse grid stochastic collocation method for partial differential equations with random input data," *SIAM J. Numer. Anal.*, vol. 46, no. 5, pp. 2309–2345, 2008.

[31] G. Calafiore, L. El Ghaoui, *et al.*, "Distributionally robust chance-constrained linear programs with applications," *Techical Report, DAUIN, Politecnico di Torino, Torino, Italy*, 2005.

[32] L. Lapierre, D. Soetanto, and A. Pascoal, "Nonsingular path following control of a unicycle in the presence of parametric modelling uncertainties," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 16, no. 10, pp. 485–503, 2006.

[33] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE transactions on automatic control*, vol. 52, no. 8, pp. 1362–1379, 2007.

[34] P. Walters, R. Kamalapurkar, L. Andrews, and W. E. Dixon, "Online approximate optimal path-following for a kinematic unicycle," *arXiv preprint arXiv:1310.0064*, 2013.

[35] L. Lapierre and B. Jouvencel, "Robust nonlinear path-following control of an auv," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 2, pp. 89–102, 2008.

[36] J. Lee and H.-J. Chang, "Analysis of explicit model predictive control for path-following control," *PloS one*, vol. 13, no. 3, p. e0194110, 2018.

[37] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[38] R. V. Lopes, P. Santana, G. Borges, and J. Ishihara, "Model predictive control applied to tracking and attitude stabilization of a vtol quadrotor aircraft," in *21st International Congress of Mechanical Engineering*, 2011, pp. 176–185.