Shaping the Closed-loop Behavior of Nonlinear Systems under Probabilistic Uncertainty using Arbitrary Polynomial Chaos

Joel A. Paulson and Ali Mesbah

Abstract—This paper presents a computationally efficient method for performance verification and tuning of model-based controllers in constrained nonlinear systems subject to probabilistic uncertainties with arbitrary distributions. The proposed method is based on an extension of generalized polynomial chaos, an asymptotically convergent spectral method for uncertainty propagation, to directly handle arbitrary uncertainty distributions. The proposed arbitrary polynomial chaos (aPC) method only requires knowledge of the statistical moments of the distribution, and is capable of estimating the aPC expansion coefficients using a minimal number of closed-loop simulations. Advantages of the proposed aPC method are demonstrated on a benchmark continuous reactor problem controlled by a scenario-based nonlinear model predictive controller.

I. INTRODUCTION

Modern control design methods, such as model predictive control (MPC), commonly use a control-relevant model of a system that is amenable to online computations [1]. As a result, there may be mismatch between the model used for control and some high-fidelity model (e.g., physics-based model) of the system used to verify the performance of the controller in simulations. The mismatch can arise from a number of sources including the complexity of the model equations and the dimension and interpretation of system states and/or uncertainties. In particular, this mismatch poses a significant challenge in robust or stochastic MPC (e.g., [2], [3], [4]) of uncertain nonlinear systems, where offline controller tuning and verification is generally considered indispensable [5]. However, closed-loop simulations can be computationally expensive in these situations because the control law is implicitly represented as the solution to a potentially large-scale nonlinear optimization problem. The prospect of controller tuning is even further complicated by the fact that important control objectives for stochastic systems are usually quantified in terms of the full distribution of relevant closed-loop quantities of interest, which can be challenging to estimate accurately in practice.

This paper addresses the problem of controller tuning and closed-loop performance verification for continuous-time nonlinear systems subject to parameter uncertainty, where the uncertainty is modeled with some arbitrary probability distribution. Monte Carlo (MC) sampling is the method of choice for propagation of high-dimensional uncertainties due to the fact that its convergence rate depends only on the number of samples used to estimate statistical properties [6]. However, the convergence rate of MC methods is often slow in practice, especially when dealing with a limited number of uncertainations. Generalized polynomial chaos (gPC) expansions [7] are a widely-used alternative to MC since they are capable of achieving much faster convergence rates for reasonably smooth functions. The basic concept of gPC is to expand any \( L_2 \) function in terms of a set of orthogonal polynomials. However, gPC can only be applied after the uncertainty has been transformed to a set of independent random variables. These transformations have been shown to lead to Gibbs phenomena as well as to the deterioration of gPC’s convergence rate [8].

This work presents an arbitrary polynomial chaos (aPC) method, which is an extension of gPC to directly handle arbitrary distributions. The proposed aPC method consists of two main steps: constructing polynomials that are orthogonal with respect to any distribution of uncertainty and computing the coefficients of the expansion. The first important feature of the proposed aPC method is that both of these problems can be solved using only \( \textit{moments} \) of the uncertainty distribution. As a result, the full probability distribution does not need to be known in closed-form, which is particularly important when these distributions are characterized from limited experimental data via Bayesian inference. The second key feature of the method is that the expansion coefficients can be estimated using a minimal number of closed-loop simulations. To this end, an optimization-based, \( \textit{moment-matching} \) method is presented for deriving efficient quadrature rules that accurately approximate the multivariate integrals defining the true values of the coefficients.

The proposed aPC method is demonstrated on a benchmark continuous reactor problem that is controlled using a scenario-based nonlinear MPC strategy [9]. It is shown that aPC is capable of achieving the same accuracy as gPC with a much lower order expansion and an order-of-magnitude fewer closed-loop simulations. The aPC method is then used to systematically tune parameters of the controller in order to achieve tight satisfaction of state chance constraints.

II. PROBLEM STATEMENT

Consider a closed-loop system described by the following set of nonlinear ordinary differential equations (ODEs)

\[
\dot{x}(t, \theta) = f(t, x(t, \theta), u(t, \theta), \theta), \quad \forall t \in (0, t_f], \quad (1a)
\]

\[
u(t, \theta) = \kappa \lambda(D_t), \quad (1b)
\]

where \( f : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n} \rightarrow \mathbb{R}^{n_x} \) is some known function that describes the system dynamics; \( x(t, \theta) \in \mathbb{R}^{n_x} \) is the closed-loop state with time derivatives \( \dot{x} \) and initial conditions \( x_0(\theta) \); \( u(t, \theta) \in \mathbb{R}^{n_u} \) is the control input; \( D_t \) is the collection of system measurements in the time period \([0, t_f]\);
\( \kappa_{\lambda} \) is the control law that maps the measurement space to the input space and is defined in terms of some user-specified tuning parameters \( \lambda \); and \( \theta \in \Gamma \subseteq \mathbb{R}^n \) denotes a set of time-invariant uncertain parameters.

Since the measurements \( D_t \) depend on the system parameters, \( \kappa_{\lambda} \) is implicitly a function of \( \theta \). We do not make any assumptions on the smoothness of \( \kappa_{\lambda} \), or need to know \( \kappa_{\lambda} \) in closed-form. The parameters \( \theta \) are modeled with a \( n \)-variate random vector that can have an arbitrary probability distribution denoted by \( p_\theta : \Gamma \rightarrow \mathbb{R}_{\geq 0} \), where \( \Gamma \subseteq \mathbb{R}^n \) is its support. Hence, \( p_\theta \) represents probabilistic uncertainties associated with unknown model parameters or initial conditions of the closed-loop model (1). These parameters may be estimated from data using, e.g., Bayesian inference, such that \( p_\theta \) may be multi-modal or have statistically dependent elements. That is, \( p_\theta \) can be arbitrarily complex.

The performance of the closed-loop system is assumed to be quantified in terms of the cost function

\[
J(\theta) = \int_0^T \ell(t, x(t, \theta), u(t, \theta))dt,
\]

where \( \ell : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \) is some stage cost function. We assume that \( \ell(\cdot) \) is continuous and smooth; however, it does not have to satisfy any positive definite requirements such that it accommodates fairly general “economic” objectives. Without loss of generality, we assume that the control objective is to minimize the expected cost \( \mathbb{E}\{J(\theta)\} \). Additionally, we assume that the closed-loop system is subject to chance constraints on the state and hard constraints on the input

\[
\mathbb{P}\{x(t, \theta) \in \mathbb{X}\} \geq 1 - \beta, \quad u(t, \theta) \in \mathbb{U},
\]

where \( \beta \in [0, 1] \) is the allowed probability of violation of the state constraints; \( \mathbb{X} \subseteq \mathbb{R}^{n_x}; \) and \( \mathbb{U} \subseteq \mathbb{R}^{n_u} \).

In theory, the optimal controller \( \kappa_{\lambda}^* \) that minimizes \( \mathbb{E}\{J(\theta)\} \) subject to constraints (3) is the solution to a dynamic programming problem, which is generally intractable (see, e.g., [10] for details). Thus, we focus on “tractable” (but suboptimal) MPC methods for specifying \( \kappa_{\lambda} \) in this work. Depending on the complexity of the system model (1a), the MPC problem might be too expensive to solve online, e.g., when (1a) is a high-fidelity model based on a set of physics-based equations. In these cases, a control-relevant model (e.g., a lumped-parameter or data-driven model) that is more amenable to real-time computations is typically used in the MPC problem. Denote the control-relevant model by

\[ q_{k+1} = \hat{f}(q_k, u_k, d_k), \]

where \( k \) denotes discrete time; \( q \in \mathbb{R}^{n_q} \) is the control-relevant state; and \( d \in \mathbb{D} \subseteq \mathbb{R}^{n_d} \) is an “effective” disturbance. The model (4) may substantially differ from the assumed structurally accurate high-fidelity model (1a). For example, these may differ in terms of the model equations \( f \) and \( \hat{f} \), dimension and physical interpretation of states \( x \) and \( q \), or the dimension, description, and physical interpretation of the uncertainties \( \theta \) and \( d \). Note that inclusion of \( d \) in (4) allows for consideration of robust/stochastic MPC methods that are capable of systematically accounting for uncertainty in the cost and constraints to improve upon the inherent robustness of nominal MPC that treats \( \mathbb{D} = \{0\} \).

Due to mismatch between the high-fidelity and control-relevant models, any designed \( \kappa_{\lambda} \) will likely not yield the target closed-loop objectives in terms of the minimum cost \( \mathbb{E}\{J(\theta)\} \) or the desired probability of state constraint satisfaction (3). Hence, the controller parameters \( \lambda \) must be tuned in practice to ensure that the closed-loop system behaves adequately. This is especially important in large-scale nonlinear systems with complicated uncertainty structures since the relationships between \( \lambda \) and \( p_{J(\theta)} \) or \( \mathbb{P}\{x(t, \theta) \in \mathbb{X}\} \) can be highly nontrivial and difficult to estimate from intuition alone. Therefore, the main goal of this paper is to present an efficient method, based on an extension of generalized polynomial chaos (gPC), for systematically extracting the relationships between the controller parameters and the actual closed-loop properties of the system while accounting for uncertainties \( \theta \). The gPC method is reviewed next, and the proposed extension is presented in Section IV.

### III. Generalized Polynomial Chaos Theory

This section describes the general concept of polynomial chaos and its application to (1). For clarity, we denote the quantity of interest (QoI) of the closed-loop system by

\[ z(t, \theta) = g(x(t, \theta)), \]

where \( g : \mathbb{R}^{n_x} \rightarrow \mathbb{R} \). We assume \( z \) is scalar; however, the developed procedure is merely repeated for each component of \( z \) in the case of multiple QoIs. The key idea is to (numerically) establish the function \( z = z(t, \theta) \), which is implicitly defined by (1) and (5). That is, we do not have an explicit representation for how \( \theta \) affects \( z \), and our goal is to develop an asymptotically convergent surrogate model that can represent the distribution \( \{p_{z(t, \theta)}\}_{t \in [0,T]} \). For notational convenience, we often refer to \( z(t, \theta) \) as \( z(\theta) \), and drop the QoI’s dependence on \( t \) and controller parameters \( \lambda \).

The notion of polynomial chaos is to expand the random variable \( z \) in terms of a user-specified germ \( \xi \) [7]

\[ z(\theta) = \sum_{i=0}^{\infty} a_i \Psi_i(\xi(\theta)) \approx \sum_{i=0}^{L} a_i \Psi_i(\xi(\theta)), \]

where \( a_i \) are the expansion coefficients, \( L \) is the truncation level so that a total of \( L + 1 \) terms are retained in the expansion, and \( \Psi_i \) are polynomial functions of the germ \( \xi \). The germ is a random vector that is fully specified by its distribution \( p_\xi \). The polynomial basis functions can be uniquely determined to be orthogonal with respect to \( p_\xi \)

\[ \langle \Psi_i, \Psi_j \rangle = \int \Psi_i(\xi)\Psi_j(\xi)p_\xi(\xi)d\xi = (\Psi_i^2)\delta_{ij}, \]

where \( \delta_{ij} \) is the Kronecker delta. Since the parameters \( \theta \sim p_\theta \) may have a different distribution than the germ \( \xi \sim p_\xi \), an invertible transformation \( T : \Xi \rightarrow \Gamma \) must be used such that \( \theta = T(\xi) \) for all \( \xi \in \Xi \) and \( \xi = T^{-1}(\theta) \) for all \( \theta \in \Gamma \). This transformation need not be unique. As long as the dimension of these two random spaces is the same, the Rosenblatt transformation can be used to define the transformation \( T \).
Definition 1 (Rosenblatt transformation [11]): Let $X$ be a random vector of dimension $n$. The Rosenblatt transformation for $X$, denoted by $R_X : \mathbb{R}^n \rightarrow \mathbb{R}^n$, is defined by the conditional cumulative distribution functions (CDFs) of $X$

$$U_1 = F_{X_1}(X_1),$$

$$\vdots$$

$$U_n = F_{X_n|X_{n-1},\ldots,X_1}(X_n|X_{n-1},\ldots,X_1).$$

The vector $U = R_X(X)$ is composed of i.i.d. standard uniform random variables, i.e., $U_i \sim \mathcal{U}(0,1)$. Hence, $X = R_X^{-1}(U)$ denotes the inverse Rosenblatt transformation. \qed

A transformation (or diffeomorphism) between the two probabilistic uncertainties $\theta$ and $\xi$ can now be defined as

$$\theta = T(\xi) = R_\theta^{-1}(R_\xi(\xi)).$$

The function $z(\theta)$ in (6) is in fact the composition $z \circ T$. The convergence rate of the expansion (6) is strongly dependent on $z \circ T$ and, when $z \circ T$ is highly nonlinear, (6) can only be well-approximated using high-order polynomials in $\xi$. The generalized polynomial chaos (gPC) method extends the choice of germs from the normal distribution originally proposed by [12] to those with distributions corresponding to orthogonal polynomials from the Askey scheme [7]. It is important to note, however, that gPC suffers from three limitations: (i) the elements of the germ $\xi = (\xi_1,\ldots,\xi_n)$ must be independent; (ii) every element $\xi_i \sim \mathcal{N}(0,1)$ should be chosen from the Askey scheme; and (iii) the entire function $p_0$ must be known to define the transformation $T$. If $z \circ T$ is an $L_2$ function and $\xi_1,\ldots,\xi_n$ is a complete basis for the $L_2$ space, then (6) converges asymptotically as $L \rightarrow \infty$ [13]. In practice, however, the rate of convergence of (6), which depends entirely on the nonlinearity and smoothness of $z \circ T$, is the key computational bottleneck as it sets the truncation level $L$ needed to achieve a desired level of accuracy.

The functional form of the QoI $z(\cdot)$ is dictated by the closed-loop dynamics (1) while, on the other hand, the functional form of the transformation $T(\cdot)$ is governed by the user’s choice of the germ $\xi$. As such, we can avoid introducing nonlinearity in $z(T(\xi))$, which will in turn lead to a larger $L$ in (6), by selecting $T$ to have a simple form. In fact, when $T$ equals the identity map, the polynomials specified by (7) can be determined using $p_0$ directly, which alleviates the need for explicitly constructing $T$. To this end, we propose the so-called arbitrary polynomial chaos (aPC) method that can be interpreted as a generalization of gPC to allow the germ distribution $p_\xi$ to be arbitrary. There are two main steps needed to specify an aPC expansion: construction of orthogonal polynomials with respect to any germ distribution and computation of the expansion coefficients $a_i$. As shown in the next section, both of these steps can be performed using only moments of the parameter distribution $p_\theta$, so that a closed-form expression for $p_\theta$ need not be known. This is a key advantage of the aPC method compared to gPC.

IV. ARBITRARY POLYNOMIAL CHAOS EXPANSIONS

In this section, we summarize the proposed aPC method, its key properties, and how it can be used to estimate the distribution of the QoIs of the closed-loop system (1).

A. Orthogonal polynomial basis functions

For standard probability distributions, their respective orthogonal polynomials can be selected from the Askey scheme of polynomials [7]. For arbitrary germ distributions, however, the polynomials in (6) must be constructed numerically. Let the multivariate monic polynomials be denoted by

$$\varphi_j(\xi) = \prod_{k=1}^{n} \xi_k^{\alpha_k(j)}, \quad j = 0, \ldots, L, \quad \sum_{k=1}^{n} \alpha_k(j) \leq d, \quad (8)$$

where $d$ is the highest order of polynomials retained in the expansion (6) and $\alpha_k(j)$ is a multi-index that describes the degree of parameter $k$ in the expansion term $j$. The total number of terms in the truncated expansion (6) is given by

$$L + 1 = \frac{(n + d)!}{n!d!}.$$  

The enumeration within the same degree is arbitrary, while $i < j$ must hold if $\deg(\varphi_i) < \deg(\varphi_j)$. By taking the expected value of the product of these monomials, we can construct the Gram-moment matrix $M$, which is a positive definite matrix whose $(i,j)$-th elements are $[M]_{ij} = \mathbb{E}[\varphi_i(\theta)\varphi_j(\theta)] = \langle \varphi_i, \varphi_j \rangle$ (i.e., the raw moments of $\xi$) [14].

Theorem 1 ([14]): For all $j = 0, \ldots, L$, the set of polynomials

$$\Psi_j(\xi) = s_{1j}\varphi_1(\xi) + s_{2j}\varphi_2(\xi) + \cdots + s_{jj}\varphi_j(\xi) \quad (9)$$

form an orthonormal system such that $\mathbb{E}[\Psi_i(\xi)\Psi_j(\xi)] = \langle \Psi_i, \Psi_j \rangle = \delta_{ij}$, where coefficients $\{s_{ij}\}_{i,j=1}^{L}$ are defined by

$$R^{-1} = \begin{bmatrix}
1 & 1 & \cdots & 1 \\
1 & s_{11} & \cdots & s_{1L} \\
\vdots & \vdots & \ddots & \vdots \\
1 & s_{L1} & \cdots & s_{LL}
\end{bmatrix}, \quad (10)$$

and $R$ is the Cholesky decomposition of $M = R^\top R$. \qed

For $\theta = T(\xi) = \xi$, Theorem 1 states that the polynomials $\{\Psi_j\}_{j=0}^{L}$ in (6) can be readily constructed from the moments of $\theta$. Thus, the distribution $p_\theta$ is not required for constructing the polynomials. Furthermore, the convergence rate of (6) is improved by avoiding nonlinearities in $T$. Note that the Gram-Schmidt procedure is an alternative approach for constructing orthogonal polynomials with respect to arbitrary distributions, as discussed in [15].

B. Optimization-based estimation of expansion coefficients

For general nonlinear systems, the expansion coefficients $\{a_i\}_{i=0}^{L}$ in (6) must be estimated numerically. There are two main approaches for approximating the coefficients: intrusive and non-intrusive [16], [17]. Here, we adopt the non-intrusive approach since this allows the function $z(\cdot)$ to be treated as a “black box”. Non-intrusive methods are based on taking the inner product of (5) with $\Psi_i$ and applying the orthogonality
property of the basis to obtain an analytic expression for the coefficients [18]

\[ a_i = \frac{\langle z, \Psi_i \rangle}{\langle \Psi_i \rangle} = \frac{1}{\langle \Psi_i \rangle} \int z(T(\xi)) \Psi_i(\xi) p_{\xi}(\xi) d\xi. \]  

(11)

Since the multivariate integrals in (11) cannot be evaluated exactly, we look to approximate them using a finite number of forward model evaluations based on some chosen quadrature (or integration) rule

\[ a_i \approx \tilde{a}_i = \frac{1}{\langle \Psi_i \rangle} \sum_{j=1}^{Q} w_j z(T(\xi_j)) \Psi_i(\xi_j), \]  

(12)

where \( \{\xi^1, \ldots, \xi^Q\} \in \Xi \subseteq \mathbb{R}^n \) is a set of nodes and \( \{w^1, \ldots, w^Q\} \in \mathbb{R} \) are their corresponding weights. This approximation is particularly useful when \( z(\theta) \) is not known explicitly, e.g., when \( \kappa_\lambda \) is implicitly defined as the solution to an optimization problem as in MPC. The most important step in any non-intrusive method is the selection of the integration points and weights, which should be chosen such that \( Q \) is as small as possible to obtain a desired level of accuracy. Various rules have been proposed in the context of PC expansion such as grid-based, randomized, monomial cubature, and optimization-based rules (e.g., see [19] and the references therein).

Optimization-based rules minimize some metric relating the quadrature operator \( U_Q : f \mapsto \sum_{j=1}^{Q} w_j f(\xi_j) \) to the exact integral operator \( I : f \mapsto \int_{\Xi} f(\xi) p_{\xi}(\xi) d\xi \). One such approach, known as moment matching, looks to determine a non-negative measure on \( \Xi \) that minimizes a sensitivity function subject to constraints that the measure matches moments of \( p_{\xi} \) up to a certain finite order [20]. We consider moment matching in this work as it is the natural multivariate extension of univariate Gaussian quadrature rules, which are optimal in one dimension. The moment-matching optimization problem is formulated as

\[ \min_{\nu \geq 0} \int r(\xi) d\nu(\xi), \]  

(13)

s.t. \[ \varphi_j(\xi) d\nu(\xi) = \mathbb{E}\{\varphi_j(\xi)\}, \quad j = 1, \ldots, n_M, \]

where the decision variable \( \nu \in \mathcal{M} \) is a non-negative measure that belongs to the space \( \mathcal{M} \) of finite Borel measures on \( \Xi \); \( \varphi_0, \ldots, \varphi_{n_M} \) are test functions; \( r \) is a user-specified continuous sensitivity function that is linearly independent of the test functions; and \( n_M = \frac{n + d_M}{n + d_M} \) is the total number of moments matched for the highest order \( d_M \). Note that the test functions can be chosen as any continuous function (including non-polynomial functions if desired), but typically are chosen to be monomials (8). The optimization problem (13) is a linear program (LP) with \( n_M \) equality constraints and an infinite-dimensional decision variable \( \nu \).

Although (13) is convex, the optimization is NP-hard when \( n > 1 \) [20]. Hence, we propose a tractable algorithm for solving (13) for multi-dimensional uncertainty. The proposed algorithm consists of three steps. The first step involves solving the finite-dimensional LP

\[ \min_{\{w^k\}_{k=1}^{S}} \sum_{k=0}^{S} w^k r(\xi^k), \]  

(14)

s.t. \[ \sum_{k=0}^{S} w^k \varphi_j(\xi^k) = \mathbb{E}\{\varphi_j(\xi)\}, \quad j = 1, \ldots, n_M, \]

where \( \{\xi^1, \ldots, \xi^S\} \subseteq \Xi \) is a finite set of sample points chosen to form a grid in \( \Xi \) with small mesh size such that \( S \gg n_M \), and \( \{w^1, \ldots, w^S\} \) is the corresponding set of weights that are decision variables in (14). The LP (14) is equal to (13) when the measure \( \nu \) is represented in terms of a finite number of points \( \nu(\xi) = \sum_{k=1}^{S} w^k \delta(\xi - \xi^k) \), where \( \delta(\cdot) \) is the Dirac delta function. In general, there can be many feasible solutions to (14), but all solutions have at most \( n_M \) non-zero coefficients \( w^k \). Thus, the LP problem (14) yields an integration rule with at most \( S = n_M \) nodes.

The solution to (14) typically contains \( Q < n_M \) “clusters” of sample points, which lie near each point in the support of the optimal measure \( \nu^* \). In the second step, the clusters of sample points are identified, for example, using a greedy algorithm [21]. After identifying the clusters, the points within each cluster are combined through a weighted convex combination

\[ \hat{w}^i = \sum_{i=1}^{n_M} w^i \quad \text{and} \quad \hat{\xi}^i = \sum_{i=1}^{n_M} \frac{w^i}{\hat{w}^i} \xi^i, \]  

(15)

for all clusters \( i = 1, \ldots, Q \). However, the resulting approximate quadrature rule with nodes \( \{\hat{\xi}^1, \ldots, \hat{\xi}^Q\} \) and weights \( \{\hat{w}^1, \ldots, \hat{w}^Q\} \) may not exactly match the moments of \( p_{\xi} \). Thus, the third and final step involves refining the rule (15) by solving a nonlinear least-squares optimization problem

\[ \min_{\xi^k \in \Xi, w^k \geq 0} \sum_{j=1}^{n_M} \left( \mathbb{E}\{\varphi_j(\xi)\} - \sum_{k=0}^{Q} w^k \varphi_j(\xi^k) \right)^2, \]  

(16)

using \( \{\hat{x}^k, \hat{w}^k\}_{k=1}^{Q} \) as the initial guess. The optimization (16) is a nonlinear program, which can be solved with standard algorithms such as interior point or active set methods. Although the quadrature rule obtained from (16) can be substantially different from that of (15), the solution to (16) typically converges quickly to an objective value of zero. This procedure is illustrated in the example (Section V).

C. Summary of aPC algorithm and properties

We now summarize the algorithm for constructing the aPC expansion for the closed-loop QoL \( z(\theta) \):

1. Specify the number of uncertain parameters \( n \), polynomial order \( d \), number of moments matched \( d_M \), germ distribution \( p_{\xi} \), transformation \( \theta = T(\xi) \), and controller \( \kappa_\lambda(\cdot) \).
2. Find \( \{\Psi_i\}_{i=0}^{L} \) using the Cholesky decomposition (9).
3. Determine a moment-matching quadrature rule \( U_Q \).
4. Solve (1) with fixed \( \theta^j = T(\xi^j) \) to get \( \{x^N(t, \theta^j)\}_{t \in [0, t_f]} \) and then evaluate the QoL \( z^N(\theta^j) = g(x^N(t, \theta^j)). \)
5. Estimate the aPC expansion coefficients
\[ \tilde{a}_i^N = \frac{1}{\langle \Psi_i \rangle} \sum_{j=1}^{Q} w_j z^N(\theta^j) \Psi_i(\xi^j), \quad i = 0, \ldots, L. \] (17)

6. Construct the \( d \)-th order approximation to (6)
\[ \tilde{z}_L(\theta) = \sum_{i=0}^{L} \tilde{a}_i^N \Psi_i(\xi(\theta)). \] (18)

Here, the superscript “\( N \)” denotes quantities derived from numerical integration of the ODEs (1). Since the coefficients \( \{\tilde{a}_i^N\}_{i=0}^{L} \) are approximations of the true coefficients \( \{a_i\}_{i=0}^{L} \), a proper choice of the moment-matching quadrature rule is critical for obtaining high-accuracy estimates.

Let \( z_L(\theta) = \sum_{i=0}^{L} a_i \Psi_i(\xi(\theta)) \) denote the truncated expansion with exact coefficients and \( \tilde{z}_L(\theta) = \sum_{i=0}^{L} \tilde{a}_i \Psi_i(\xi(\theta)) \) denote the truncated expansion with approximated coefficients (without numerical integration error). Then, we can bound the error between the true random variable \( z(\theta) \) and its approximation \( \tilde{z}_L^N(\theta) \) using Theorem 2.

**Theorem 2 (aPC approximation error):** The mean-square error between (6) and (18) satisfies
\[ \varepsilon = \mathbb{E}((z(\theta) - \tilde{z}_L^N(\theta))^2)^{1/2} \leq \left[ \varepsilon_G^2 + \varepsilon_Q^2 + \varepsilon_N^2 \right]^{1/2}, \]
where \( \varepsilon_G = \mathbb{E}\{(z(\theta) - z_L(\theta))^2\}^{1/2} \) denotes the truncation error due to the finite-order of the expansion, \( \varepsilon_Q = \mathbb{E}\{(z_L(\theta) - \tilde{z}_L(\theta))^2\}^{1/2} \) denotes the aliasing error due to the integral approximation, and \( \varepsilon_N = \mathbb{E}\{(\tilde{z}_L(\theta) - \tilde{z}_L^N(\theta))^2\}^{1/2} \) denotes the integration error when solving the ODEs (1).

**Proof:** The proof follows directly from the triangle inequality. See Proposition 4.1 of [22] for a similar result. \( \Box \)

Theorem 2 indicates that the error associated with the approximated aPC expansion (18) can be divided into the three sources \( \varepsilon_G, \varepsilon_Q, \) and \( \varepsilon_N \), which can all be controlled independently. Under the assumption that the function \( z \circ T(\cdot) \) is a sufficiently smooth function of the germ, \( \varepsilon_G \) can be decreased by increasing \( d \), \( \varepsilon_Q \) can be decreased by increasing \( d_M \), and \( \varepsilon_N \) can be decreased by increasing the order of the (possibly adaptive) numerical integration scheme.

**Remark 1:** We note that only \( z(\theta) \) should satisfy some form of “smoothness” condition. The closed-loop state or control trajectories can exhibit jumps or discontinuities as long as these do not directly translate onto the QoI.

V. CASE STUDY

The aPC method is demonstrated on an uncertain nonlinear continuously-stirred-tank reactor (CSTR) [23], described by
\[ \dot{c}_A = F(c_A - c_A^0) - k_1 c_A - k_2 c_A^2, \] (19a)
\[ \dot{c}_B = -F c_B + k_1 c_A - k_2 c_B, \] (19b)
\[ \dot{T}_R = F(T_m - T_R) + \frac{k_{FW} A}{\rho c_p} \] \[ - k_1 c_A \Delta H_{AB} + k_2 c_B \Delta H_{BC} + k_3 c_A^2 \Delta H_{AD}, \] (19c)
\[ \dot{T}_K = \frac{1}{m_{K} c_{ph}} (\dot{Q}_K + k_{w} A(T_R - T_K)), \] (19d)

where the reaction rates \( k_i \) follow the Arrhenius law
\[ k_i = k_{0,i} \exp \left( \frac{E_{A,i}}{R(T_R + 273.15)} \right), \quad i = 1, 2, 3. \] (19)

In (19), \( c_A \) and \( c_B \) denote concentration of A and B, \( T_R \) denotes the reactor temperature, and \( T_K \) denotes the coolant temperature. The inlet flow \( F = \dot{V}_{in}/V_R \) and heat removed \( Q_K \) are the control inputs. The parameters \( (E_{A,3}, k_{0,1}) \in [7704, 9416] \) [kJ/mol \times [1.16 \times 10^{12}, 1.42 \times 10^{12}] \) [hr\(^{-1}\)] are uncertain. For simplicity, these parameters are scaled, i.e., \( \theta = (\theta_1, \theta_2) \in [-1, 1] \times [-1, 1] \) using a simple affine transformation.

The joint distribution \( p_\theta(\theta) = p_{\theta_1}(\theta_1)p_{\theta_2|\theta_1}(\theta_2|\theta_1) \) is specified by two beta random variables \( \theta_1 \sim \mathcal{B}(2, 2) \) and \( \theta_2|\theta_1 \sim \mathcal{B}(\theta_1 + 3, -\theta_1 + 2) \). A contour plot of \( p_\theta \) is shown in Fig. 1. Note that there is a high degree of correlation between the two parameters due to the fact that \( \theta_2|\theta_1 \) has a strong dependence on \( \theta_1 \). The initial conditions, state constraints, and input constraints of the CSTR are given in Table I, while the remaining constant parameters are given in [23].

The control inputs \( F \) and \( Q_K \) are computed using a full-state feedback multi-stage NMPC controller, which uses a scenario tree to account for a finite number of realizations of \( d_k \in \{d_k^1, \ldots, d_k^6\} \) over the prediction horizon [9]. To mitigate the exponential growth in the number of uncertainty realizations over time, a robust horizon is introduced that treats the uncertainty as constant after a certain stage. As such, there are a number of tuning (or changeable) parameters in multi-stage NMPC including the stage cost, constraint backoffs, robust and prediction horizons, and the number and values of the uncertainties.

To design the multi-stage NMPC, a control-relevant model of the form (4) was obtained by applying a fourth-order Runge-Kutta scheme to (19) with piece-wise constant inputs on a sampling interval of \( \Delta t = 0.005 \) hr. The control-relevant uncertainty description \( d_k \) was chosen based on the extreme values and center of the bounding box for \( (\theta_1, \theta_2) \), resulting in 5 scenarios. The closed-loop control objective (2) is to maximize the expected production of B at final time \( t_f = 0.2 \) hr. A similar cost function is chosen in multi-stage NMPC, but control moves are also penalized, i.e., stage cost is \( -\dot{V}_{in} c_B + r_1 \Delta F^2 + r_2 \Delta Q_K^2 \) where \( r_1 = 10^{-5} \), \( r_2 = 10^{-7} \) and \( \Delta \) denotes the difference between the current and previous times. Each scenario is assumed to have equal probability, the prediction horizon was set to 40, and the robust horizon was set to 1. The resulting nonlinear program was solved using CasADi [24] and IPOPT [25].

In this case study, the key tuning parameter is the maximum reactor temperature \( T_{max} \). The goal is to select a value for \( T_{max} \), enforced as a hard state constraint in the multi-stage NMPC problem, such that the closed-loop system satisfies the chance constraint
\[ \mathbb{P}\{50^\circ \text{C} \leq T_R(t) \leq 140^\circ \text{C} \} \geq 1 - \beta, \quad \forall t \in [0, t_f], \]
for any \( \beta \in [0, 1] \). Thus, we aim to quantitatively relate \( T_{max} \) and \( \beta \) using the proposed aPC method in Section IV. First, we must construct an accurate moment-matching quadrature rule. By selecting the germ to be equal to the parameters such
that $\theta = \xi$, we obtain the quadrature rule shown in Fig. 1 for $d_M = 5$. This rule matches intuition as nodes are placed in regions of the parameter space with high probability and the weights appear to be proportional to the probability of the parameters lying in the surrounding area.

Convergence properties for both gPC and aPC expansions are reported in Fig. 2, which shows the mean-square error (MSE) of the expansions for the closed-loop reactor temperature (averaged over time) versus order. Multiple values of $d_M$ were used to derive the moment-matching quadrature rules for aPC. For $d_M$ values of 5, 7, 9, and 11, the total number of nodes $Q$ was found to be 9, 16, 25, and 36, respectively. As seen in Fig. 2, errors only converge as order increases when a sufficient number of nodes is used. When too few nodes are utilized, the aliasing error $\epsilon_Q$ dominates. In fact, this error can even lead to a jump in MSE as order increases, as seen for $d_M = 5$. This highlights the fact that $U_Q$ must be carefully chosen and validated. We also compare the rate of convergence of aPC and gPC with $\xi \sim \mathcal{N}(0, I)$. Fig. 2 shows that gPC requires an expansion of double the order to achieve the same level of accuracy as aPC. Additionally, the quadrature rule used to compute the gPC coefficients requires more than 100 nodes due to the extra nonlinearity in the gPC expansion. This is particularly important here since each closed-loop simulation takes around 30–60 seconds, indicating that the aPC method leads to vast computational savings compared to gPC. We again emphasize that the aPC method does not require exact knowledge of $p$, as it can be implemented using only moments of this distribution.

Now that an accurate aPC “surrogate” for the QoI (i.e., the reactor temperature $T_R(t)$) has been established, it can be used to verify and tune the closed-loop control performance. The probability distribution of the reactor temperature over time with $T_{\text{max}} = 140^\circ C$ in the multi-stage NMPC controller is shown in Fig. 3. These distributions were estimated with kernel density estimation using $5 \times 10^5$ i.i.d samples, which can be generated efficiently by substituting the i.i.d. samples of $\xi$ into the aPC expansion (18). Note that the full distribution is within the constrained region $[50, 140]^\circ C$, meaning that the multi-stage NMPC leads to robust state constraint satisfaction. In addition, Fig. 3 shows that $p_{T_R}(t)$ has a non-standard form that can be captured by a low-order aPC expansion.

Lastly, we explore the relationship between $T_{\text{max}}$ and $\beta$. In particular, we address the question of, given a $\beta$, what is the best choice of $T_{\text{max}}$ in order to guarantee satisfaction of chance constraints? Fig. 4 shows $\mathbb{P}\{T_R(t) \leq 140^\circ C\}$ versus $T_{\text{max}} = 140 + \text{backoff} ^\circ C$; note the lower bound of $50^\circ C$ is irrelevant since the support lies well-above $100^\circ C$. For example, the backoff should be set to $3^\circ C$ when $\beta = 0.25$ because $\beta = 1 - \mathbb{P}\{T_R(t) \leq 140^\circ C\}$ makes the chance constraint active, meaning minimal performance giveaway.

Fig. 4 implies that the relationship between $T_{\text{max}}$ and the true state constraint violation probability is generally nonlinear and not necessarily intuitive, such that it is difficult to imagine a way of properly selecting $T_{\text{max}}$ without simulating the closed-loop system. Since the proposed aPC method is able to extract closed-loop information using a limited number of these computationally expensive simulations, it enables efficient tuning of the controller parameters.

### VI. Conclusions

This paper presents an arbitrary polynomial chaos method for approximating the distribution of closed-loop system properties of constrained, continuous-time nonlinear systems subject to multivariate probabilistic uncertainties with arbitrary distributions. The aPC method only requires knowledge of statistical moments of the distribution of uncertainties. The proposed method applies an optimization-based, moment-matching technique to generate quadrature rules with a minimal number of nodes, so that a (convergent) surrogate model of closed-loop dynamics can be estimated from a limited number of closed-loop simulations. It is shown that efficient uncertainty propagation through nonlinear closed-loop dynamics is crucial for systematically selecting controller parameters that ensure desired closed-loop properties are achieved in practice. Future work will include investigation of alternative methods such as optimized stochastic collocation (OSC) [19] for generating minimal quadrature rules. Recent work by the authors investigated the use of OSC for explicit constraint backoff calculations in the context of nonlinear MPC of stochastic systems [26].

### REFERENCES

Fig. 2. Mean-square error (normalized over time) versus order of the polynomial chaos expansion of the reactor temperature. The aPC errors correspond to different moment-matching cubature rules obtained by increasing order $d_M$ of moments matched, with $d_M = 11$ requiring only $Q = 36$.

Fig. 3. Reactor temperature distribution versus time under multi-stage NMPC with the temperature constraint set to 140°C.

Fig. 4. The actual probability of violating the joint chance constraint on the reactor temperature versus the backoff value of the temperature constraint used in the multi-stage NMPC problem, i.e., $T_{\text{max}} = 140 + \text{backoff}$ °C.


