Optimal Control of Dynamical Systems with Time-invariant Probabilistic Parametric Uncertainties

by

Dongying Erin Shen

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctorate in Chemical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2017 [February 2018]

© Massachusetts Institute of Technology 2017. All rights reserved.

Author ..... Signature redacted
Department of Chemical Engineering
September 23, 2017

Certified by ..... Signature redacted
Richard D. Braatz
Edwin R. Gilliland Professor of Chemical Engineering
Thesis Supervisor

Accepted by ..... Signature redacted
Patrick S. Doyle
Graduate Officer, Department of Chemical Engineering

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
NOV 29 2017
LIBRARIES
ARCHIVES

Optimal Control of Dynamical Systems with Time-invariant Probabilistic Parametric Uncertainties

by

Dongying Erin Shen

Submitted to the Department of Chemical Engineering
on September 23, 2017, in partial fulfillment of the
requirements for the degree of
Doctorate in Chemical Engineering

Abstract

The importance of taking model uncertainties into account during controller design is well established. Although this theory is well developed and quite mature, the worst-case uncertainty descriptions assumed in robust control formulations are incompatible with the uncertainty descriptions generated by commercial model identification software that produces time-invariant parameter uncertainties typically in the form of probability distribution functions. This doctoral thesis derives rigorous theory and algorithms for the optimal control of dynamical systems with time-invariant probabilistic uncertainties.

The main contribution of this thesis is new feedback control design algorithms for linear time-invariant systems with time-invariant probabilistic parametric uncertainties and stochastic noise. The originally stochastic system of equations is transformed into an equivalent deterministic system of equations using polynomial chaos (PC) theory. In addition, the $\mathcal{H}_2$- and $\mathcal{H}_\infty$-norms commonly used to describe the effect of stochastic noise on output are transformed such that the eventual closed-loop performance is insensitive to parametric uncertainties. A robustifying constant is used to enforce the closed-loop stability of the original system of equations. This approach results in the first PC-based feedback control algorithm with proven closed-loop stability, and the first PC-based feedback control formulation that is applicable to the design of fixed-order state and output feedback control designs. The numerical algorithm for the control design is formulated as optimization over bilinear matrix inequality (BMI) constraints, for which commercial software is available. The effectiveness of the approach is demonstrated in two case studies that include a continuous pharmaceutical manufacturing process.

In addition to model uncertainties, chemical processes must operate within constraints, such as upper and lower bounds on the magnitude and rate of change of manipulated and/or output variables. The thesis also demonstrates an optimal feedback control formulation that explicitly addresses both constraints and time-invariant probabilistic parameter uncertainties for linear time-invariant systems. The $\mathcal{H}_2$-optimal feedback controllers designed using the BMI formulations are incorporated

into a fast PC-based model predictive control (MPC) formulation. A numerical case study demonstrates the improved constraint satisfaction compared to past polynomial chaos-based formulations for model predictive control.

Thesis Supervisor: Richard D. Braatz
Title: Edwin R. Gilliland Professor of Chemical Engineering

Acknowledgments

Sergio Lucia, Yiming Wan, and Rolf Findeisen are acknowledged for their collaborations in some of the research. Financial support from Novartis® is acknowledged.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The importance of taking model uncertainties into account during controller design is well established, which has motivated the generation of a large literature on robust control theory. Although this theory is well developed and quite mature, the worst-case uncertainty descriptions assumed in robust control formulations are incompatible with the uncertainty descriptions generated by commercial model identification software (e.g., AdaptX, System Identification Toolbox), which produce time-invariant parameter uncertainties that belong to probability distribution functions.

The objective of this doctoral research is to derive rigorous theory and algorithms for the optimal control of linear time-invariant systems with time-invariant probabilistic uncertainties. Below is a summary of the main results obtained in the doctoral research, organized in a way consistent with the organization of the chapters within the thesis.

Chapter 1 motivates the objective of the thesis and summarizes its content.

Chapter 2 describes a new algorithm for the design of nonlinear dynamical systems with probabilistic parameter uncertainties. The dependence of the design objective and constraints on uncertainties is quantified by a polynomial chaos expansion (PCE), while the relationships between the design parameters and the design objective/constraints are parameterized by Legendre polynomials. In two case studies, the polynomial chaos-based algorithm reduces the number of system evaluations required by optimization by order of magnitude. In addition, quantifying the dependence on

parametric uncertainties via PCEs and including the quantification in the design optimization improve the distribution of the performance index and the probability of constraint fulfillment.

Chapter 3 describes a new feedback control design algorithm for linear time-invariant systems with time-invariant probabilistic parametric uncertainties and stochastic noise. The originally stochastic system of equations is transformed into an equivalent deterministic system of equations using a PCE, which an appropriate transformation of an $\mathcal{H}_2$-control objective so that the closed-loop performance is insensitive to the parametric uncertainties. A drawback of existing PCE-based control algorithms is that the truncation errors due to the use of PCE approximation can cause instability for the original system of equations. To address this issue, a robustifying constant derived from the small gain theorem is included in the existing PCE-based $\mathcal{H}_2$-norm minimization to enforce the closed-loop stability of the original system of equations. This approach results in the first PCE-based feedback control algorithm with proven closed-loop stability, and the first PCE-based feedback control formulation that is applicable to the design of fixed-order state and output feedback control designs. The numerical algorithm for the control design is formulated as optimization over bilinear matrix inequality constraints, for which commercial software is available.

The effectiveness of the approach is demonstrated for a numerical example and a continuous pharmaceutical manufacturing process. The inclusion of the robustifying constant is shown to stabilize the original system of equations over all possible values of uncertain parameters. The PCE-based $\mathcal{H}_2$-optimal controller leads to better distributions of the $\mathcal{H}_2$ norms compared to controllers optimized for the nominal system of equations and based on a worst-case strategy, and the PCE-based method significantly reduces the problem size compared to a Monte-Carlo-simulation-based method, in terms of the number of variables in the optimizer. Similar results are also derived for the $\mathcal{H}_\infty$ norm, which quantifies the amplification of the system output due to noise and disturbances when both inputs and outputs are quantified in terms of integral squared error over time. These latter results are the first PCE-based feedback control design algorithm derived for the $\mathcal{H}_\infty$ control objective, and the theory

satisfies similar rigorous stability proofs.

In addition to model uncertainties, chemical processes must operate within constraints, such as upper and lower bounds on the magnitude and rate of change of manipulated and/or output variables. The purpose of Chapter 4 is to demonstrate an optimal feedback control formulation that explicitly addresses both constraints and time-invariant probabilistic parameter uncertainties for linear time-invariant systems. This chapter incorporates the $\mathcal{H}_2$-optimal feedback controllers in Chapter 3 into a fast PCE-based model predictive control formulation. When the constraints are inactive, the manipulated variables are close to those computed by the PCE-based feedback control system from Chapter 3. When the constraints become limiting, the new model predictive control algorithm shifts the manipulated variables in an optimal manner to satisfy the constraints. A numerical case study demonstrates the improved closed-loop performance compared to past polynomial chaos-based formulations for model predictive control.

Chapter 5 summarizes the main results and points out some future directions.

Chapter 2

PCE-based Optimal Design

2.1 Introduction

Models for real systems have associated uncertainties, which can influence the system performance and/or constraint satisfaction.[37,52] It is well established that ignoring uncertainties during design optimization can produce designs that are highly sensitive to uncertainties. The potential consequences of ignoring uncertainties during design include large variability in product quality[15,32] and higher total costs.[36]

Such studies have motivated the development of numerical algorithms that include uncertainties in optimal design problems. A popular strategy is to optimize designs based on a worst-case objective, which ensures that each system within an uncertainty set has the same or better objective than in worst case.[39] A design optimized for the worst-case uncertainty can result in poorer product quality or higher costs for more representative uncertainties than designs that do not consider the effects of uncertainties or that weigh more equally the effects of all uncertainties.[12,31] Worst-case design can be very conservative in practice, especially for systems in which the worst-case uncertainty has a vanishingly low probability of occurrence.

This chapter considers the optimal design of nonlinear dynamical systems with parametric uncertainties described by probability distribution functions (aka "probabilistic uncertainties"). For this type of uncertainty, the dominant strategy is to optimize the distribution of the objective and to satisfy constraints within speci-

fied probabilities. This strategy often requires estimation of the expected values and/or the variances of the objective and the constraints.[15,28,30,41,49] The Monte-Carlo-simulation-based method, which samples the probabilistic distributions and propagates these samples through the system models, is a common approach to estimate the expected values and the variances[2] but has a slow convergence rate on the order of $1/\sqrt{n}$, for which n is the number of samples. As a result, the Monte-Carlo-simulation-based method, which requires a large number of system simulations to accurately estimate the expected values and variances, is computationally expensive. More efficient sampling techniques, such as the Latin hypercube sampling[21] and the Hammersley sequence sampling,[8] have convergence rate on the order of $1/n$.

Another way to take uncertainties into account during design optimization is to employ Gaussian quadrature to estimate the integrals for the expected values and the variances.[15,20] These integrals are estimated each time that the optimizer accesses a new set of design inputs (the term 'design inputs' is used instead of 'design parameters' in the remainder of this chapter to avoid potential confusion with model parameters). Consequently, the number of system evaluations required by the optimal design calculations depends on the details of the optimizer and the closeness of the initial guesses for the design inputs to the optimal solution.

Another method to account for uncertainties is via polynomial chaos (PC) theory that uses polynomial expansions to approximate the dependence of system outputs on probabilistic uncertainties.[50] The polynomial expansions that achieve the fastest convergence rate have been derived for a wide variety of distributions (see Table 2.1), meaning that these expansions have the highest accuracy among all polynomial expansions of the same order, and require the smallest number of terms to achieve a specified accuracy. With the optimal choices of polynomials in Table 2.1, the convergence rate is exponential, and it is straightforward to estimate means and variances. Due to its low computational cost, the application of PC to chemical engineering and system design problems has become of interest in recent years.[1,12,18,19,23,28,32]

This chapter proposes a new PC-based algorithm for optimizing design inputs for systems with probabilistic uncertainties. To make the number of required sys-

tem evaluations independent of the optimization algorithmic details and the initial guesses, this algorithm parameterizes the dependence of the optimization objective and constraints on design inputs with the Legendre polynomials. In addition, the impacts of uncertainties on the objective and constraints are quantified by PC expansions and included in the optimization. Computational efficiency of the design input parameterization and the robustness of polynomial chaos-based optimal designs are demonstrated in two case studies.

2.2 Problem Statement

Consider a nonlinear dynamical system described by differential-algebraic equations,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mathbf{k}, \boldsymbol{\eta}) \tag{2.1}$$

$$\mathbf{x}(t = 0) = \mathbf{x}_0 \tag{2.2}$$

$$\mathbf{0} = \mathbf{z}(t, \mathbf{x}, \mathbf{u}, \mathbf{k}, \boldsymbol{\eta}), \tag{2.3}$$

for which t is time, $\mathbf{x}$ is the vector of system states, $\mathbf{u} \in \mathbb{R}^{n\times 1}$ is the vector of design inputs, $\mathbf{k}$ is the vector of certain parameters (e.g., heat capacities), $\boldsymbol{\eta}$ is the vector of parameters with probabilistic uncertainties, $\mathbf{f}$ and $\mathbf{z}$ are algebraic functions, and $\mathbf{x}_0$ is the initial condition. The design inputs can include controller design parameters, initial conditions, and/or a parameterization of continuous-time trajectories such as temperature profiles. Although not explicitly treated here, the methodology described in this chapter can be directly extended to distributed parameter systems. To simplify the notation, the dependency on $\mathbf{k}$ is suppressed in all functions subsequently defined in this chapter, but can be explicitly included without loss in generality.

The design optimization objective g and constraint function $\mathbf{h} \in \mathbb{R}^{m\times 1}$ are functions of the design inputs and uncertain parameters, i.e.,

$$g = g(\mathbf{u}, \boldsymbol{\eta}) \tag{2.4}$$

$$\mathbf{h} = \mathbf{h}(\mathbf{u}, \boldsymbol{\eta}). \tag{2.5}$$

For example, a typical optimization objective for a batch design problem is

$$g(\mathbf{u}, \boldsymbol{\eta}) = \int_{t_0}^{t_f} \phi[\mathbf{x}(t, \mathbf{u}, \boldsymbol{\eta})] \, \mathrm{d}t + \xi(t_f, \mathbf{u}, \boldsymbol{\eta}),$$

for which t_0 is the initial time, t_f is the final time, and ϕ and ξ are algebraic functions. Typical constraints are defined on the states evaluated at specific points in time or on integrals of the states over time, with some examples given in the case studies.

With this nomenclature, the nominal optimal design problem is

$$\begin{aligned} &\min_{\mathbf{u} \in \mathcal{U}} \; g(\mathbf{u}, \boldsymbol{\eta}_{\text{nominal}}) \\ \text{s.t. } &\mathcal{U} = [u_{1,\text{lower}}, u_{1,\text{upper}}] \times \cdots \times [u_{n,\text{lower}}, u_{n,\text{upper}}] \\ &\mathbf{h}(\mathbf{u}, \boldsymbol{\eta}_{\text{nominal}}) \leq \mathbf{0}. \end{aligned} \tag{2.6}$$

The design inputs are assumed to have known finite bounds. Typically such bounds can be specified via knowledge about the phenomena (e.g., that the mixing speed in a bioreactor must be less than some value to avoid cell damage) and/or thermodynamic and/or kinetic arguments (e.g., that the heat transfer system limits the temperature to be within some range). Solutions to this optimization can result in a wide distribution of the objective and/or a high probability of constraint violation in the presence of probabilistic uncertainties in $\boldsymbol{\eta}$. A well-known formulation of the design problem that reduces the effects of probabilistic uncertainties is

$$\begin{aligned} &\min_{\mathbf{u} \in \mathcal{U}} \; \mathbb{E}_{\boldsymbol{\eta}}[g(\mathbf{u}, \boldsymbol{\eta})] + \alpha_0 \operatorname{Var}_{\boldsymbol{\eta}}[g(\mathbf{u}, \boldsymbol{\eta})] \\ \text{s.t. } &\mathcal{U} = [u_{1,\text{lower}}, u_{1,\text{upper}}] \times \cdots \times [u_{n,\text{lower}}, u_{n,\text{upper}}] \\ &\mathbb{E}_{\boldsymbol{\eta}}[h_1(\mathbf{u}, \boldsymbol{\eta})] + \alpha_1 \operatorname{Var}_{\boldsymbol{\eta}}[h_1(\mathbf{u}, \boldsymbol{\eta})] \leq 0 \\ &\mathbb{E}_{\boldsymbol{\eta}}[h_2(\mathbf{u}, \boldsymbol{\eta})] + \alpha_2 \operatorname{Var}_{\boldsymbol{\eta}}[h_2(\mathbf{u}, \boldsymbol{\eta})] \leq 0 \\ &\vdots \\ &\mathbb{E}_{\boldsymbol{\eta}}[h_m(\mathbf{u}, \boldsymbol{\eta})] + \alpha_m \operatorname{Var}_{\boldsymbol{\eta}}[h_m(\mathbf{u}, \boldsymbol{\eta})] \leq 0, \end{aligned} \tag{2.7}$$

for which $\mathbb{E}_{\boldsymbol{\eta}}$ and $\operatorname{Var}_{\boldsymbol{\eta}}$ are the expected value and the variance computed from in-

tegration with respect to $\boldsymbol{\eta}$ and the set of scales $\{\alpha_i\}$ controls the tradeoffs between the expected values and the variances.[‡] This optimization requires inexpensive and accurate estimates for

1. the functional dependence of $g(\mathbf{u}, \boldsymbol{\eta})$ and $h_i(\mathbf{u}, \boldsymbol{\eta})$ on $\boldsymbol{\eta}$ to quantify the effects of probabilistic uncertainties on the expected values and the variances;

2. the functional dependence of $\mathbb{E}_{\boldsymbol{\eta}}[g(\mathbf{u}, \boldsymbol{\eta})]$, $\mathrm{Var}_{\boldsymbol{\eta}}[g(\mathbf{u}, \boldsymbol{\eta})]$, $\mathbb{E}_{\boldsymbol{\eta}}[h_i(\mathbf{u}, \boldsymbol{\eta})]$, and $\mathrm{Var}_{\boldsymbol{\eta}}[h_i(\mathbf{u}, \boldsymbol{\eta})]$ on the vector of design inputs $\mathbf{u}$.

2.3 Proposed Approach

The first step of the approach is to approximate the dependence of the optimization objective and constraints on uncertainties with polynomial expansions,[50]

$$g(\mathbf{u}, \boldsymbol{\eta}) \approx \sum_{j=0}^{N_0} g_j(\mathbf{u}) \phi_j(\boldsymbol{\eta})$$
$$h_i(\mathbf{u}, \boldsymbol{\eta}) \approx \sum_{j=0}^{N_i} h_{i,j}(\mathbf{u}) \phi_j(\boldsymbol{\eta}),$$

for which the optimal polynomials $\phi_j(\boldsymbol{\eta})$ depend on the distributions of $\boldsymbol{\eta}$ and are given in Table 2.1, and N_i's are positive integers for all i's. The polynomials in Table 2.1 have been proven to be optimal, as they minimize the L_2 norm of the residual from using finite terms in the polynomial expansion and have exponential convergence in the corresponding Hilbert functional space.[50] This property of exponential convergence results in accurate approximations even with a relatively small number

[‡]For a design objective written as a maximization, the second term in (2.7) is multiplied by minus one.

of terms in the expansion. The polynomials are orthogonal and satisfy

$$\begin{aligned}\langle \phi_i(\boldsymbol{\eta}), \phi_j(\boldsymbol{\eta}) \rangle &= \int_{\mathbf{H}} \phi_i(\boldsymbol{\eta}) \phi_j(\boldsymbol{\eta}) \, \mathrm{d}\mu(\boldsymbol{\eta}) \\ &= \begin{cases} \langle \phi_i^2(\boldsymbol{\eta}) \rangle & \text{if } i = j; \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \tag{2.8}$$

for which $\mathbf{H}$ is the support for $\boldsymbol{\eta}$ and $\mu(\boldsymbol{\eta})$ is the weight function for $\phi_i(\boldsymbol{\eta})$. As a result of (2.8),

$$g_j(\mathbf{u}) = \frac{\langle g(\mathbf{u}, \boldsymbol{\eta}), \phi_j(\boldsymbol{\eta}) \rangle}{\langle \phi_j^2(\boldsymbol{\eta}) \rangle}. \tag{2.9}$$

Also, the expected value and the variance, which are computed from integration with respect to uncertainties $\boldsymbol{\eta}$, can be estimated once the expansion coefficients have been computed:

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\eta}}[g(\mathbf{u}, \boldsymbol{\eta})] &= \int_{\mathbf{H}} g(\mathbf{u}, \boldsymbol{\eta}) \mathrm{d}\mu(\boldsymbol{\eta}) \\ &\approx \int_{\mathbf{H}} \sum_{j=0}^{N_0} g_j(\mathbf{u}) \phi_j(\boldsymbol{\eta}) \, \mathrm{d}\mu(\boldsymbol{\eta}) \\ &= \sum_{j=0}^{N_0} g_j(\mathbf{u}) \int_{\mathbf{H}} \phi_j(\boldsymbol{\eta}) \, \mathrm{d}\mu(\boldsymbol{\eta}) \\ &= \sum_{j=0}^{N_0} g_j(\mathbf{u}) \, \langle \phi_j(\boldsymbol{\eta}), 1 \rangle \\ &= g_0(\mathbf{u}) \underbrace{\langle \phi_0(\boldsymbol{\eta}), 1 \rangle}_{1} \\ &= g_0(\mathbf{u}) \end{aligned} \tag{2.10}$$

$$
\begin{aligned}
\mathrm{Var}_{\boldsymbol{\eta}}[g(\mathbf{u},\boldsymbol{\eta})] &= \mathbb{E}_{\boldsymbol{\eta}}[g^2(\mathbf{u},\boldsymbol{\eta})] - \mathbb{E}^2_{\boldsymbol{\eta}}[g(\mathbf{u},\boldsymbol{\eta})] \\
&= \int_{\mathbf{u}} g^2(\mathbf{u},\boldsymbol{\eta})\,\mathrm{d}\mu(\boldsymbol{\eta}) - g_0^2(\mathbf{u}) \\
&\approx \sum_{j=0}^{N_0} g_j^2(\mathbf{u})\left\langle \phi_j^2(\boldsymbol{\eta})\right\rangle - g_0^2(\mathbf{u}) \\
&= \sum_{j=1}^{N_0} g_j^2(\mathbf{u})\left\langle \phi_j^2(\boldsymbol{\eta})\right\rangle . \qquad (2.11)
\end{aligned}
$$

The above approach is described in several papers on PC-based design.[28] For this approach, estimation of the expected values and the variances requires the computation of PCE coefficients. For simple systems, PCE coefficients can be computed via intrusive Galerkin projection, which takes the inner product of (2.1)–(2.3) with each basis function to obtain a system of equations for the expansion coefficients.[50] For more complex systems, to which Galerkin projection cannot be applied, non-intrusive methods are used, such as tensor-product quadrature, which estimates the integral for the numerator of (2.9), and linear regression, which solves for the complete set of expansion coefficients by evaluating the original system for selected values for uncertainties.[10]

A drawback of using (2.10) and (2.11) in (2.7) is that the expansion coefficients $g_j(\mathbf{u})$'s and $h_{i,j}(\mathbf{u})$'s have to be evaluated for every new $\mathbf{u}$ that the optimizer accesses. Since the evaluation of the expansion coefficients requires system evaluations, i.e., simulations of (2.1)–(2.3), the computational cost of the optimization is influenced by the choice of initial guesses and the effectiveness of the optimizer for the particular optimization. We propose to resolve this issue by approximating the dependence of the objective and constraints on the design inputs with an expansion of the Legendre polynomials:

$$
g(\mathbf{u},\boldsymbol{\eta}) \approx \sum_{j=0}^{n_0}\sum_{k=0}^{m_0} g_{jk}\phi_j(\boldsymbol{\eta})P_k(\mathbf{u}) \qquad (2.12)
$$

$$
h_i(\mathbf{u},\boldsymbol{\eta}) \approx \sum_{j=0}^{n_i}\sum_{k=0}^{m_i} h_{ijk}\phi_j(\boldsymbol{\eta})P_k(\mathbf{u}), \qquad (2.13)
$$

for which $P_k(\mathbf{u})$ is the Legendre polynomial in $\mathbf{u}$ of degree k, and n_i's and m_i's are positive integers for all i's. (2.12) and (2.13) are referred to as *design input parameterization* in this chapter.

In (2.12) and (2.13), the dependence of the expected values and the variances on the design inputs can be inexpensively computed, once the expansion coefficients g_{jk}'s and h_{ijk}'s have been found:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\eta}}[g(\mathbf{u},\boldsymbol{\eta})] &= \int_{\mathbf{H}} g(\mathbf{u},\boldsymbol{\eta})\,\mathrm{d}\mu(\boldsymbol{\eta}) \\
&\approx \sum_{j=0}^{n_0}\sum_{k=0}^{m_0} g_{jk}P_k(\mathbf{u}) \int_{\mathbf{H}} \phi_j(\boldsymbol{\eta})\,\mathrm{d}\mu(\boldsymbol{\eta}) \\
&\approx \sum_{j=0}^{n_0}\sum_{k=0}^{m_0} g_{jk}P_k(\mathbf{u})\,\langle\phi_j(\boldsymbol{\eta})\rangle \qquad (2.14)\\
\mathrm{Var}_{\boldsymbol{\eta}}[g(\mathbf{u},\boldsymbol{\eta})] &= \mathbb{E}_{\boldsymbol{\eta}}[g^2(\mathbf{u},\boldsymbol{\eta})] - \mathbb{E}^2_{\boldsymbol{\eta}}[g(\mathbf{u},\boldsymbol{\eta})] \\
&\approx \int_{\mathbf{H}} \left[\sum_{j=0}^{n_0}\sum_{k=0}^{m_0} g_{jk}\phi_j(\boldsymbol{\eta})P_k(\mathbf{u})\right]^2 \mathrm{d}\mu(\boldsymbol{\eta}) \\
&\quad - \left[\sum_{j=0}^{n_0}\sum_{k=0}^{m_0} g_{jk}P_k(\mathbf{u})\,\langle\phi_j(\boldsymbol{\eta})\rangle\right]^2 . \qquad (2.15)
\end{aligned}$$

The novelty of the proposed representation of the design inputs is the one-time evaluation of the expansion coefficients g_{jk}'s and h_{ijk}'s before (2.14) and (2.15) are sent to the optimizer. The other novelty of (2.14) and (2.15) is the polynomial dependence of the optimization objective and the constraints on the design inputs. Therefore, the computational cost of the optimization, which mainly depends on the number of system evaluations, is fixed. In summary, the proposed approach for designing systems with probabilistic uncertainties consists of three steps:

1. compute the expansion coefficients g_{jk}'s and h_{ijk}'s;

2. express the expected values and the variances of the optimization objective and constraints as polynomial functions of the design inputs using (2.14) and (2.15);

3. send these functions to the optimizer to find the optimal design inputs.

2.4 Case Studies

This section applies the proposed approach to two chemical reactor design problems. Tensor-product quadrature was used to determine the PCE coefficients.[10] All optimizations were solved with fmincon of MATLAB®.

2.4.1 Optimal Design of a Batch Chemical Reactor with a Series Reaction

Consider a series reaction in a batch reactor,

$$\mathrm{A} \xrightarrow{r_1} \mathrm{B} \xrightarrow{r_2} \mathrm{C},$$

for which

$$r_1 = k_{10} \exp\left(-\frac{E_1}{RT}\right) C_{\mathrm{A}},$$
$$r_2 = k_{20} \exp\left(-\frac{E_2}{RT}\right) C_{\mathrm{B}},$$

R is the gas constant, T is the reaction temperature, E_i are the activation energies, and k_{i0} are the prefactors. Table 2.2 lists the parameters used in the simulation. The design objective is to find the reaction temperature T that maximizes the concentration of the desired product B, which is produced from species A but consumed by the chemical reaction that converts species B to species C. The nominal optimization of the reaction temperature, which does not consider the dependence of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ on the uncertainties in the prefactors k_{10} and k_{20}, is

$$\max_{300\,\mathrm{K} \le T \le 400\,\mathrm{K}} C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10,\mathrm{nominal}}, k_{20,\mathrm{nominal}}, T). \tag{2.16}$$

The temperature T given by (2.16) is 324.69 K, which produces a maximum $C_{\mathrm{B}}(t_{\mathrm{f}}, T)$ of 199.06 M at the nominal values of k_{10} and k_{20}.

The uniformly distributed uncertainties in the prefactors k_{10} and k_{20} in Table 2.2 result in a wide distribution of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ at the nominal optimal reaction

temperature, as shown in Figure 2-3. The optimal design problem that takes the probabilistic uncertainties into account is

$$\max_{300\,\mathrm{K}\leq T\leq 400\,\mathrm{K}}\{\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)]-\alpha\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)]\},\qquad(2.17)$$

for which α quantifies the tradeoff between the maximization of the expected value and the reduction of the variance for $C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)$.

The dependence of $C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)$ on the uncertain prefactors k_{10} and k_{20} is quantified via PC expansions, and their dependence on T is parameterized by the Legendre polynomials:

$$C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)\approx\sum_{n=0}^{N}C_{\mathrm{B}n}(t_{\mathrm{f}})\phi_n(k_{10},k_{20},T)=\sum_{n=0}^{N}C_{\mathrm{B}n}(t_{\mathrm{f}})\phi_n(k'_{10},k'_{20},T'),$$

where

$$\begin{aligned}
k'_{10}&=\frac{k_{10}-k_{10,\mathrm{m}}}{k_{10,\mathrm{d}}}\sim\mathcal{U}(-1,1)\\
k_{10,\mathrm{m}}&=\frac{k_{10,\text{upper bound}}+k_{10,\text{lower bound}}}{2}\\
k_{10,\mathrm{d}}&=\frac{k_{10,\text{upper bound}}-k_{10,\text{lower bound}}}{2}\\
k'_{20}&=\frac{k_{20}-k_{20,\mathrm{m}}}{k_{20,\mathrm{d}}}\sim\mathcal{U}(-1,1)\\
k_{20,\mathrm{m}}&=\frac{k_{20,\text{upper bound}}+k_{20,\text{lower bound}}}{2}\\
k_{20,\mathrm{d}}&=\frac{k_{20,\text{upper bound}}-k_{20,\text{lower bound}}}{2}\\
T'&=\frac{T-T_{\mathrm{m}}}{T_{\mathrm{d}}}\sim\mathcal{U}(-1,1)\\
T_m&=\frac{T_{\text{upper bound}}+T_{\text{lower bound}}}{2}\\
T_d&=\frac{T_{\text{upper bound}}-T_{\text{lower bound}}}{2},
\end{aligned}$$

N is the total number of basis functions used in the expansion, and $\phi_n(k'_{10},k'_{20},T')$ is the n^{th} basis function that is the product of the Legendre polynomials in k'_{10}, k'_{20}, and T'.

The evaluation of the expansion coefficients $C_{\mathrm{B}n}(t_\mathrm{f})$'s uses (2.8):

$$\langle C_\mathrm{B}(t_\mathrm{f}, k_{10}, k_{20}, T), \phi_n(k'_{10}, k'_{20}, T')\rangle = C_{\mathrm{B}n}(t_\mathrm{f})\langle \phi_n^2(k'_{10}, k'_{20}, T')\rangle$$
$$C_{\mathrm{B}n}(t_\mathrm{f}) = \frac{\langle C_\mathrm{B}(t_\mathrm{f}, k_{10}, k_{20}, T), \phi_n(k'_{10}, k'_{20}, T')\rangle}{\langle \phi_n^2(k'_{10}, k'_{20}, T')\rangle},$$

where the numerator is

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} C_\mathrm{B}(t_\mathrm{f}, k_{10}, k_{20}, T)\phi_n(k'_{10}, k'_{20}, T')w(k'_{10}, k'_{20}, T')\,\mathrm{d}k'_{10}\,\mathrm{d}k'_{20}\,\mathrm{d}T', \tag{2.18}$$

and the denominator is

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \phi_n^2(k'_{10}, k'_{20}, T')w(k'_{10}, k'_{20}, T')\,\mathrm{d}k'_{10}\,\mathrm{d}k'_{20}\,\mathrm{d}T', \tag{2.19}$$

for which

$$w\,(k'_{10}, k'_{20}, T') = \left(\frac{1}{2}\right)^3 = \frac{1}{8}. \tag{2.20}$$

(2.19), which is the inner product of the n^{th} basis function with itself, depends only on the distributions of the uncertain parameters. On the other hand, the integral in (2.18) can be approximated by the Gaussian-Legendre quadrature, which evaluates $C_\mathrm{B}(t_\mathrm{f}, k_{10}, k_{20}, T)$ for selected values of k_{10}, k_{20}, and T:

$$(2.18) \approx \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} C_\mathrm{B}(t_\mathrm{f}, \bar{k}_{10,i}, \bar{k}_{20,j}, \overline{T}_k)\phi_n(\bar{k}_{10,i}, \bar{k}_{20,j}, \overline{T}_k)w(\bar{k}_{10,i})w(\bar{k}_{20,j})w(\overline{T}_k),$$

where

$$\bar{k}_{10,i} = \text{the } i^{\text{th}} \text{ root of } P_I, \text{ the Legendre polynomial of degree } I$$

$$w(\bar{k}_{10,i}) = \frac{1}{\left(1-\bar{k}_{10,i}^2\right)\left[P_I'\left(\bar{k}_{10,i}\right)\right]^2}$$

$$P_I'\left(\bar{k}_{10,i}\right) = \text{the derivative of } P_I \text{ evaluated at } \bar{k}_{10,i}$$

$$\bar{k}_{20,j} = \text{the } j^{\text{th}} \text{ root of } P_J, \text{ the Legendre polynomial of degree } J$$

$$w(\bar{k}_{20,j}) = \frac{1}{\left(1-\bar{k}_{20,j}^2\right)\left[P_J'\left(\bar{k}_{20,j}\right)\right]^2}$$

$$P_J'\left(\bar{k}_{20j}\right) = \text{the derivative of } P_J \text{ evaluated at } \bar{k}_{20,j}$$

$$\overline{T}_k = \text{the } k^{\text{th}} \text{ root of } P_K, \text{ the Legendre polynomial of degree } K$$

$$w(\overline{T}_k) = \frac{1}{\left(1-\overline{T}_k^2\right)\left[P_K'\left(\overline{T}_k\right)\right]^2}$$

$$P_K'\left(\overline{T}_k\right) = \text{the derivative of } P_K \text{ evaluated at } \overline{T}_k$$

$$C_{\mathrm{B}}(t_{\mathrm{f}}, \bar{k}_{10,i}, \bar{k}_{20,j}, \overline{T}_k) = C_{\mathrm{B}}(t_{\mathrm{f}}) \text{ evaluated at } \bar{k}_{10,i},\ \bar{k}_{20,j}, \text{ and } \overline{T}_k$$

$$\phi_n(\bar{k}_{10,i}, \bar{k}_{20,j}, \overline{T}_k) = \phi_n \text{ evaluated at } \bar{k}_{10,i},\ \bar{k}_{20,j}, \text{ and } \overline{T}_k$$

and I, J, K are positive integers. Table 2.3 lists the highest degrees of the Legendre polynomials in k_{10}', k_{20}', and T' and the values for I, J, K. If the highest order of the PC expansion is p, the minimum order of the Gaussian quadrature to accurately compute all the expansion coefficients is $p+1$.[10] As using more points to estimate the integral poses more computational cost, the minimum order of the Gaussian quadrature of $p+1$ was used in both case studies. This case study used the 2^{nd}-order Legendre polynomial in T', at which the optimal reaction temperature converges for $\alpha = 0.25$ (Figure 2-1). Once the expansion coefficients $C_{\mathrm{B}n}(t_{\mathrm{f}})$'s are evaluated, $\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$, $\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$, and therefore the objective in (2.17) can be approximated by polynomial functions in T'.

A pareto-optimality plot is a commonly used approach to select values for parameters that trade off multiple objectives. Figure 2-2 shows the pareto-optimality plot for the expected values and the variances of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ with respect to k_{10} and k_{20}

at optimal temperatures computed from (2.17) with different values of α. The optimal α, which should locate approximately at the knee of the pareto-optimality curve, is 0.25 and gives an optimal reaction temperature of 359.71 K (Figure 2-2). The 35.02 K difference between the optimal temperature for the nominal values of the prefactor k_{10} and k_{20} and that from the PC-based optimization results in drastically different distributions of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ (see Figure 2-3 and Table 2.4). Specifically, the distribution of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ for the PC-based optimal temperature has a higher average and a much smaller standard deviation than that for the optimal temperature for the nominal values of k_{10} and k_{20}. This difference demonstrates the importance of including the effects of parametric uncertainties in optimal design problems.

The effect of design-input parameterization on computational cost was also examined. When a PC expansion without design-input parameterization,

$$C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T) \approx \sum_{n=0}^{M} C_{\mathrm{B}n}(t_{\mathrm{f}}, T)\phi_n(k_{10}, k_{20}),$$

was used to approximate the dependence of $C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)$ on the uncertain parameters, the expansion coefficients $C_{\mathrm{B}n}(t_{\mathrm{f}}, T)$'s were evaluated for every new temperature the optimizer accessed. With the computed expansion coefficients, $\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$ and $\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$ were computed from

$$\begin{aligned}
\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)] &\approx \left\langle \sum_{n=0}^{M} C_{\mathrm{B}n}(t_{\mathrm{f}}, T)\phi_n(k_{10}, k_{20}), 1 \right\rangle = C_{\mathrm{B}0}(t_{\mathrm{f}}, T) \\
\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)] &= \mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}^2(t_{\mathrm{f}}, k_{10}, k_{20}, T)] - \left(\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]\right)^2 \\
&\approx \left\langle \sum_{n=0}^{M} C_{\mathrm{B}n}(t_{\mathrm{f}}, T)\phi_n(k_{10}, k_{20}), \sum_{n=0}^{M} C_{\mathrm{B}n}(t_{\mathrm{f}}, T)\phi_n(k_{10}, k_{20}) \right\rangle - C_{\mathrm{B}0}^2(t_{\mathrm{f}}, T) \\
&= \sum_{n=0}^{M} C_{\mathrm{B}n}^2(t_{\mathrm{f}}, T)\langle\phi_n^2(k_{10}, k_{20})\rangle - C_{\mathrm{B}0}^2(t_{\mathrm{f}}, T) \\
&= \sum_{n=1}^{M} C_{\mathrm{B}n}^2(t_{\mathrm{f}}, T)\langle\phi_n^2(k_{10}, k_{20})\rangle.
\end{aligned}$$

When fmincon of MATLAB® was used to find the reaction temperature that maxi-

mizes $\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)] - 0.25\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}},k_{10},k_{20},T)]$, the number of system evaluations without design-input parameterization was 576, whereas that with design-input parameterization was 108. For this case study, the design-input parameterization reduces computational cost by about a factor of 5.

2.4.2 Optimal Design of a Tubular Reactor with Five-Species Reaction Network

Molar balances for the five species in the reaction network in Figure 2-4, which carried out in a microscale automated continuous-flow tubular reactor, are[38]

$$\begin{aligned}
\frac{\mathrm{d}C_1}{\mathrm{d}t} &= -k_1C_1C_2 - k_2C_1C_2 \\
\frac{\mathrm{d}C_2}{\mathrm{d}t} &= -k_1C_1C_2 - k_2C_1C_2 - k_3C_2C_3 - k_4C_2C_4 \\
\frac{\mathrm{d}C_3}{\mathrm{d}t} &= k_1C_1C_2 - k_3C_2C_3 \\
\frac{\mathrm{d}C_4}{\mathrm{d}t} &= k_2C_1C_2 - k_4C_2C_4 \\
\frac{\mathrm{d}C_5}{\mathrm{d}t} &= k_3C_2C_3 + k_4C_2C_4,
\end{aligned}$$

where

$$k_i = A_i \exp\left(-\frac{E_{Ai}}{RT}\right)$$

and t is the distance down the reactor multiplied by its cross-sectional area and divided by the volumetric flowrate of the feed stream. Table 2.5 lists the reaction parameters. The nominal reactor design problem is to determine the residence time that maximizes the yield of the desired compound 4 with the remainng limiting reagent, compound 1, to be at most 1% of its starting amount:

$$\begin{aligned}
&\max_{0.5\,\mathrm{min}\le t_{\mathrm{res}}\le 20\,\mathrm{min}} \frac{C_4(\log_{10}A_{1,\mathrm{nominal}},\log_{10}A_{4,\mathrm{nominal}},t_{\mathrm{res}})}{C_{10}} \\
&\text{s.t.} \quad \frac{C_1(\log_{10}A_{1,\mathrm{nominal}},\log_{10}A_{4,\mathrm{nominal}},t_{\mathrm{res}})}{C_{10}} \le 1\%.
\end{aligned}$$

The optimal residence time for the tubular reactor for the nominal values of $\log_{10}A_1$ and $\log_{10}A_4$ is 0.921 min.

The uncertainties in $\log_{10}A_1$ and $\log_{10}A_4$ lead to distributions of $C_1(\log_{10}A_1, \log_{10}A_4, t_{\text{res}})$ and $C_4(\log_{10}A_1, \log_{10}A_4, t_{\text{res}})$. Therefore, both uncertainties are taken into account in the reactor design problem as

$$\max_{0.5\,\text{min}\le t_{\text{res}}\le 20\,\text{min}} \left\{ \text{E}_{\log_{10}A_1,\log_{10}A_4}\left[\frac{C_4(\log_{10}A_1,\log_{10}A_4,t_{\text{res}})}{C_{10}}\right] - \alpha \text{Var}_{\log_{10}A_1,\log_{10}A_4}\left[\frac{C_4(\log_{10}A_1,\log_{10}A_4,t_{\text{res}})}{C_{10}}\right]\right\}$$
$$\text{s.t. } \text{E}_{\log_{10}A_1,\log 10A_4}\left[\frac{C_1(\log_{10}A_1,\log_{10}A_4,t_{\text{res}})}{C_{10}}\right] \le 1\%.$$

Table 2.6 lists the highest degrees of the Legendre polynomials used for approximating the dependence of the concentrations C_1 and C_4 on $\log_{10}A_1$, $\log_{10}A_4$, and t_{res} and the number of the Gauss-Legendre quadrature points used to estimate the expansion coefficients.

The pareto-optimality curve for the expected values and the variances of $C_4(\log_{10}A_1, \log_{10}A_4, t_{\text{res}})$ at optimal residence times for different values of α is shown in Figure 2-5. A value of α of 115 was selected as a reasonable tradeoff between the expected value and the variance and gives an optimal residence time of 8.32 min. Figures 2-6 and 2-7 show that the PC-based optimal residence time has 100% constraint satisfaction for $\frac{C_1}{C_{10}}$ without significantly reducing $\frac{C_4}{C_{10}}$, whereas there is approximately 18% chance of constraint violation for the optimal residence time for the nominal values of $\log_{10}A_1$ and $\log_{10}A_4$.

The effect of design-input parametrization on computational cost was also examined. The number of system evaluations without parameterization of t_{res} was 2100, where that with parameterization of t_{res} was 125, representing a reduction of more than one order of magnitude in computational cost.

In general, the relative computational cost of using design-input parameterization versus not using design-input parameterization depends on the highest order of the Legendre-polynomial expansion in design inputs necessary to accurately approximate the dependence of the robust optimization objective and/or constraints on design inputs. If a very high order of the Legendre-polynomial expansion is required, design-

input parameterization often no longer has the advantage of reducing computational cost.

Both case studies have uniformly distributed uncertain parameters. Replacing the uniform distributions with other types of distributions involves replacing the Legendre polynomials for the uncertain parameters with other polynomials in Table 2.1. As in the case studies that had uniformly distributed uncertain parameters, the computational cost will be a function of the highest order of the PC expansion in uncertain parameters and the highest order of the Legendre-polynomial exapnsion in design inputs.

2.5 Conclusions

This chapter proposes a PC-based approach for the design of nonlinear dynamical systems with probabilistic uncertainties and bounds on design inputs. The two characteristics of this design approach are

1. the dependence of the optimization objective and constraints on design inputs is parameterized with the Legendre polynomials;

2. the effects of probabilistic uncertainties on the objective and constraints are quantified by the PC expansion.

The designs of batch and continuous-flow chemical reactors were optimized with this method in the presence of uniformly distributed uncertain parameters.

For a batch reactor with a series reaction, the reaction temperature was optimized to maximize the concentration of a desired species in the presence of two uncertain kinetic parameters. Compared to the optimal reaction temperature from the nominal optimization, the PC-based optimal temperature produced a distribution of the desired-species concentration with a higher average and a smaller standard deviation. In addition, parameterizing the dependence of the desired-species concentration on the reaction temperature with the Legendre polynomials reduced the number of system evaluations required by optimization by a factor of 5.

For a continuous-flow tubular chemical reactor with five species, the residence time was optimized to maximize the concentration of a desired species in the presence of two uncertain kinetic parameters. In addition, this reactor design problem is subject to a constraint that the remaining limiting reagent should be no greater than 1% of its starting amount. With similar distributions of the desired-species concentration at the nominal and the PC-based optimal residence times, the PC-based optimal residence time resulted in 100% constraint fulfillment, whereas the nominal optimal residence time resulted in 82% constraint fulfillment. Furthermore, design-input parameterization reduced the number of system evaluations required by optimization by a factor of 17.

The design-input parameterization significantly reduced the number of system evaluations required by optimization in both case studies. Also, including the effects of uncertainties in the optimal design problems via the PC expansions produced design inputs that improved distribution of the optimization objective for the batch reactor case study and increased the probability of constraint fulfillment for the tubular reactor case study.

PC expansions are most effective when the objective and constraints are smooth functions of the uncertain parameters, which are expected to occur in most chemical process design problems. When the objective and constraints are not smooth functions of the uncertain parameters, accurate approximation via the PC expansions will require a larger number of terms in the expansions and more system evaluations to compute the expansion coefficients, and efficient sampling methods can be less computationally expensive.[7] PC expansions are also most effective when each objective or constraint is sensitive to a relatively small number of uncertain parameters, e.g., less than ten. Properties of interest such as concentrations and yields for most chemical reaction networks and reaction-transport networks depend strongly on only a small number of key parameters, which are associated with rate-limiting steps.[47] A parameter sensitivity analysis can be conducted to determine which parameters to include in PC expansions for the design objective and each constraint. For process design problems that have larger numbers of sensitive uncertain parameters, the Smolyak

sparse grids can be used to reduce the number of function evaluations required to compute the PC expansion coefficients.[10]

Since the objective and constraints in the PC-based optimization are represented by polynomials, the resulting optimizations are polynomial programs. Although this chapter used local optimization, the polynomial dependencies mean that the optimization can be solved using any local or global optimization algorithms developed for the solution of polynomial programs.[17,40]

Uncertainty distributions η	**Optimal polynomials $\phi_i(\eta)$**
Gaussian	Hermite
Uniform	Legendre
Gamma	Laguerre
Beta	Jacobi

Table 2.1: Optimal polynomial expansions for some probabilistic distributions.[50]

k_{10} (s^{-1})	uniformly distributed between 100 and 1000; nominal value at 550
E_1/R (K)	2400
k_{20} (s^{-1})	uniformly distributed between 100 and 1000; nominal value at 550
E_2/R (K)	4800
Reaction time t_f (s)	20
Reaction temperature T (K)	between 300 and 400
Initial C_A (M)	200
Initial C_B (M)	0
Initial C_C (M)	0

Table 2.2: Parameters for the batch-reactor case study.

Highest degree of the Legendre polynomials in k'_{10} and k'_{20}	5
I and J	6
Highest degree of the Legendre polynomials in T'	2
K	3
Number of system evaluations for computing PCE coefficients for a new T without design input parameterization	$6^2 = 36$
Number of system evaluations for computing PCE coefficients with design input parameterization	$6^2 \times 3 = 108$

Table 2.3: The highest degrees of the Legendre polynomials in k'_{10}, k'_{20}, and T' and the number of the Gauss-Legendre quadrature points used to approximate the integral in (2.18) for the batch reactor case study.

Optimization	Temperature (K)	Average (M)	Standard deviation (M)
Nominal	324.69	194.12	11.03
PCE	359.96	196.17	2.51

Table 2.4: Averages and standard deviations of 10^4 samples for the nominal and the polynomial chaos-based optimal temperatures for the batch reactor case study.

C_{10} (M)	0.150
C_{20} (M)	0.375
T (K)	373.15
t_{res} (min)	between 0.5 and 20
R (J/mol·K)	8.314
$\log_{10}A_1$; A_i ($\text{M}^{-1}\text{s}^{-1}$)	uniformly distributed within $[3.0, 3.8]$; nominal value at 3.4
E_{A1} (kJ/mol)	27
$\log_{10}A_2$	3.5
E_{A2} (kJ/mol)	32.1
$\log_{10}A_3$	4.9
E_{A3} (kJ/mol)	60.0
$\log_{10}A_4$	uniformly distributed within $[2.6, 3.4]$; nominal value at 3.0
E_{A4} (kJ/mol)	45

Table 2.5: Parameters for the tubular reactor case study.[38]

Highest degree of the Legendre polynomials in $\log_{10}A_1$ and $\log_{10}A_4$	4
Number of quadrature points for each uncertain parameter	5
Highest degree of the Legendre polynomials in t_{res}	4
Number of quadrature points for t_{res}	5
Number of system evaluations for computing PCE coefficients for a new t_{res} without design input parameterization	$5^2 = 25$
Number of system evaluations for computing PCE coefficients with design input parameterization	$5^2 \times 5 = 125$

Table 2.6: The highest degrees of the Legendre polynomials in $\log_{10}A_1$, $\log_{10}A_4$, and t_{res} and the number of the Gauss-Legendre quadrature points used to approximate the expansion coefficients for the tubular reactor case study.

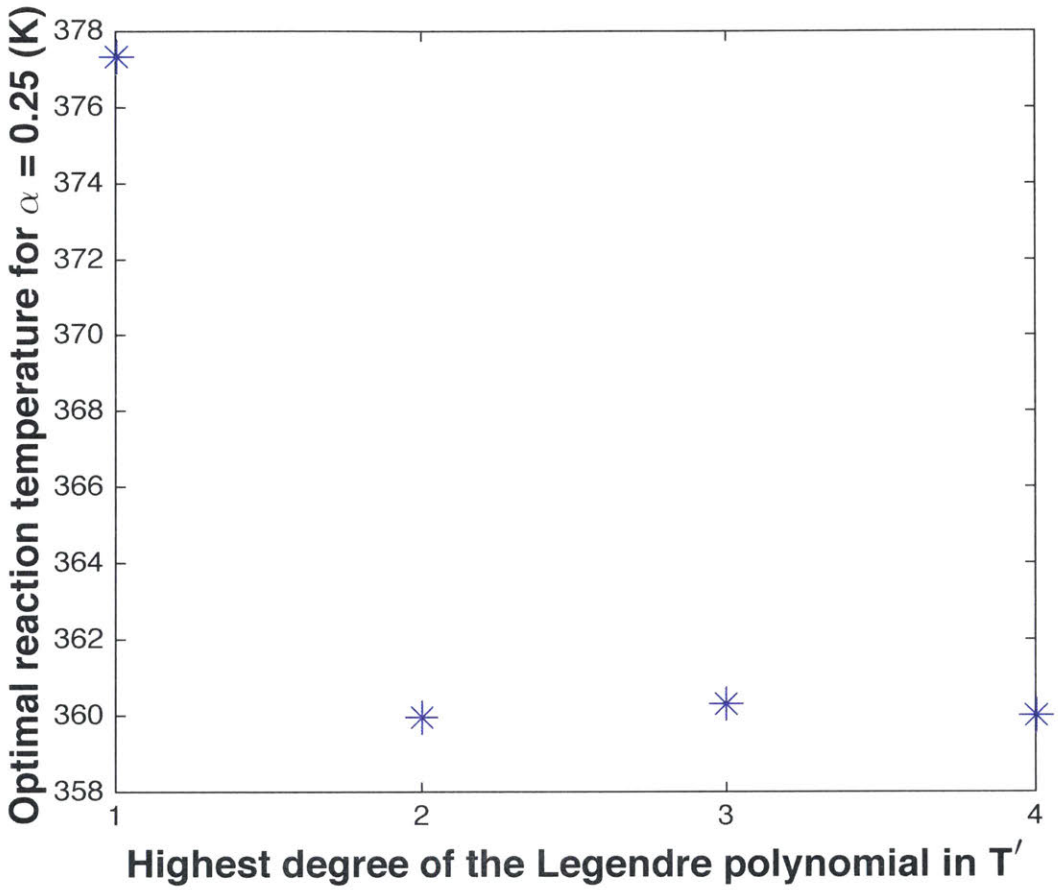

Figure 2-1: The convergence plot for the batch reactor case study: the highest degree of the Legendre polynomials in T' was based on the convergence of the optimal temperature for $\alpha = 0.25$.

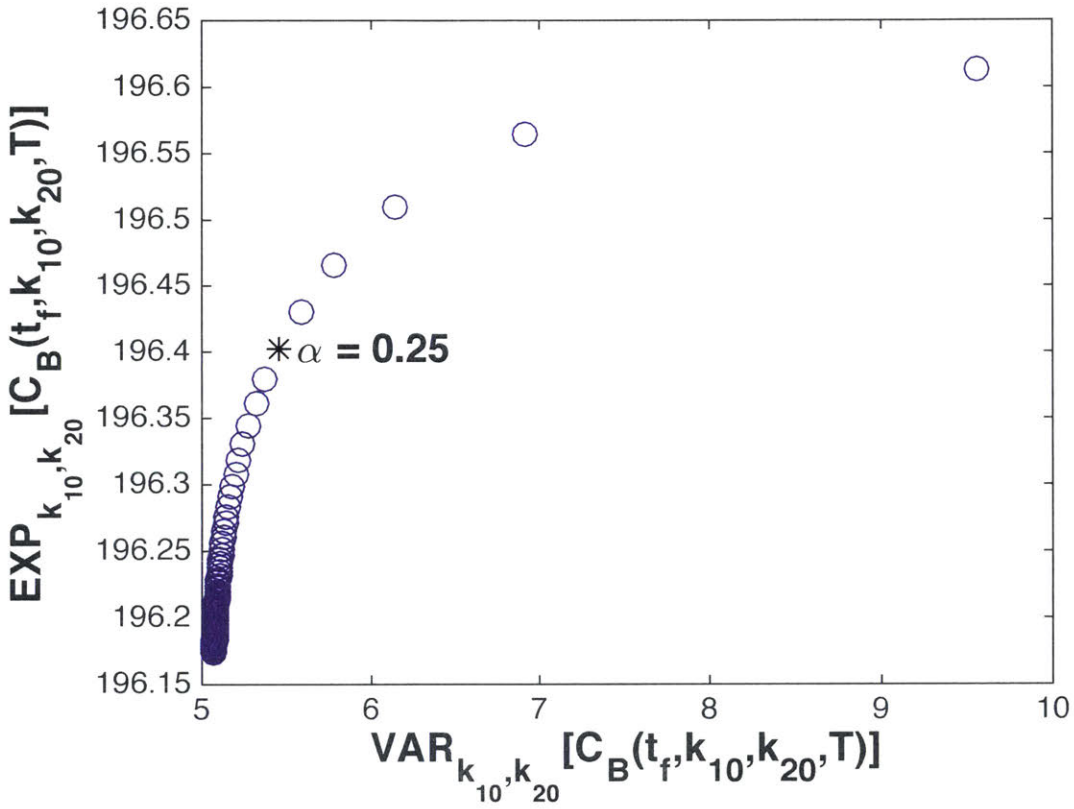

Figure 2-2: The pareto-optimality plot for the batch reactor case study that shows how $\mathbb{E}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$ changes with $\mathrm{Var}_{k_{10},k_{20}}[C_{\mathrm{B}}(t_{\mathrm{f}}, k_{10}, k_{20}, T)]$. A value of α of 0.25 is located at the knee of the curve.

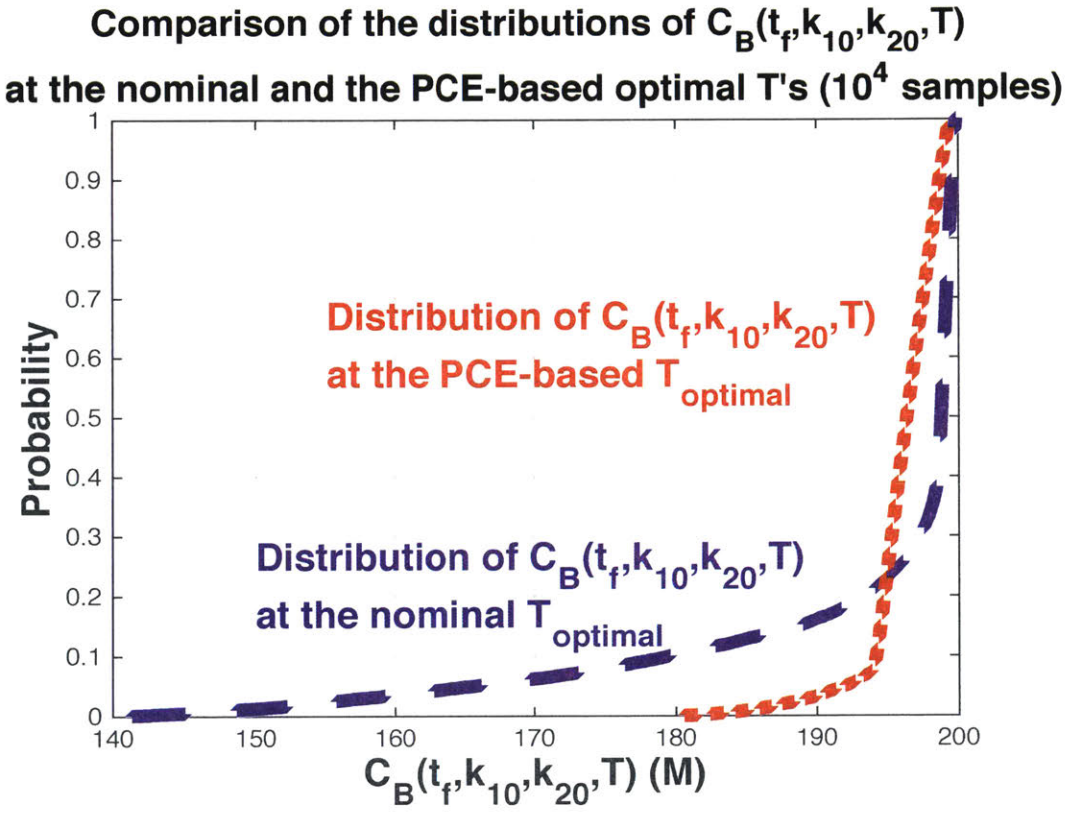

Figure 2-3: Comparison of the distributions of the intermediate concentration $C_B(t_f, k_{10}, k_{20}, T)$ at the nominal and polynomial chaos-based optimal temperatures for the batch reactor case study constructed from 10^4 Monte Carlo simulation samples.

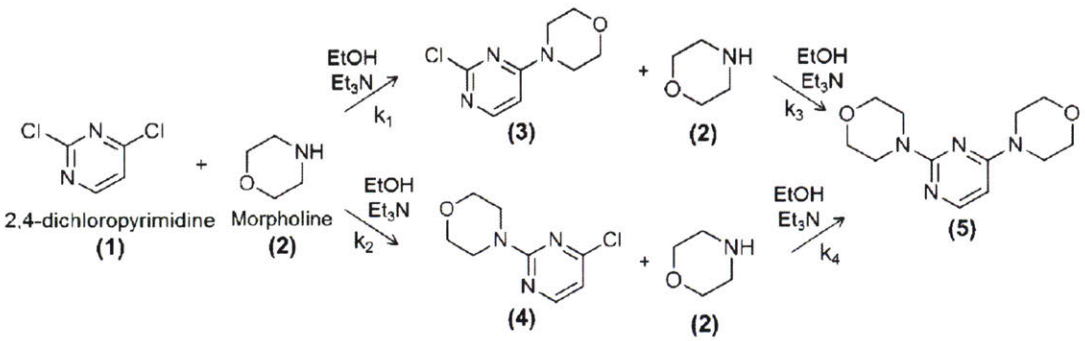

Figure 2-4: Chemical reaction network for tubular reactor.[38]

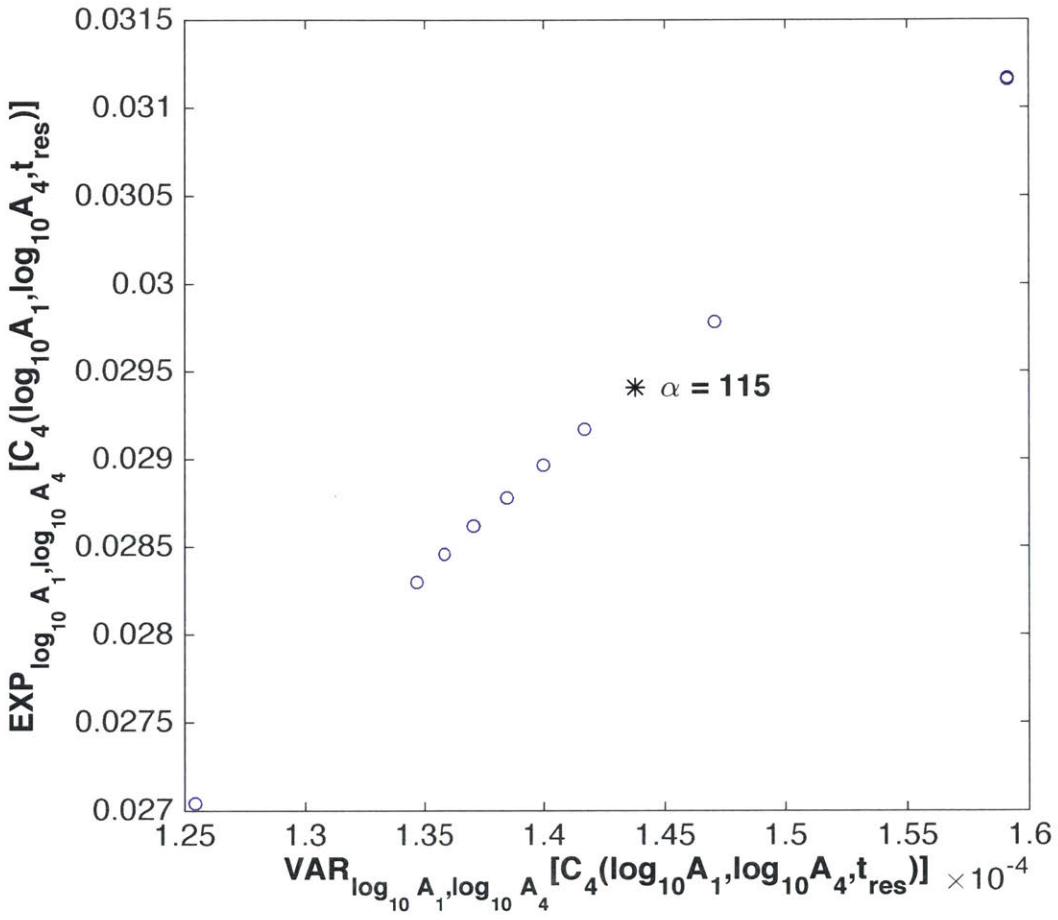

Figure 2-5: The pareto-optimality plot for the tubular reactor case study showing the tradeoff between $\mathrm{E}_{\log_{10}A_1,\log_{10}A_4}[C_4(\log_{10}A_1,\log_{10}A_4,t_{\mathrm{res}})]$ and $\mathrm{Var}_{\log_{10}A_1,\log_{10}A_4}[C_4(\log_{10}A_1,\log_{10}A_4,t_{\mathrm{res}})]$ for different values of α.

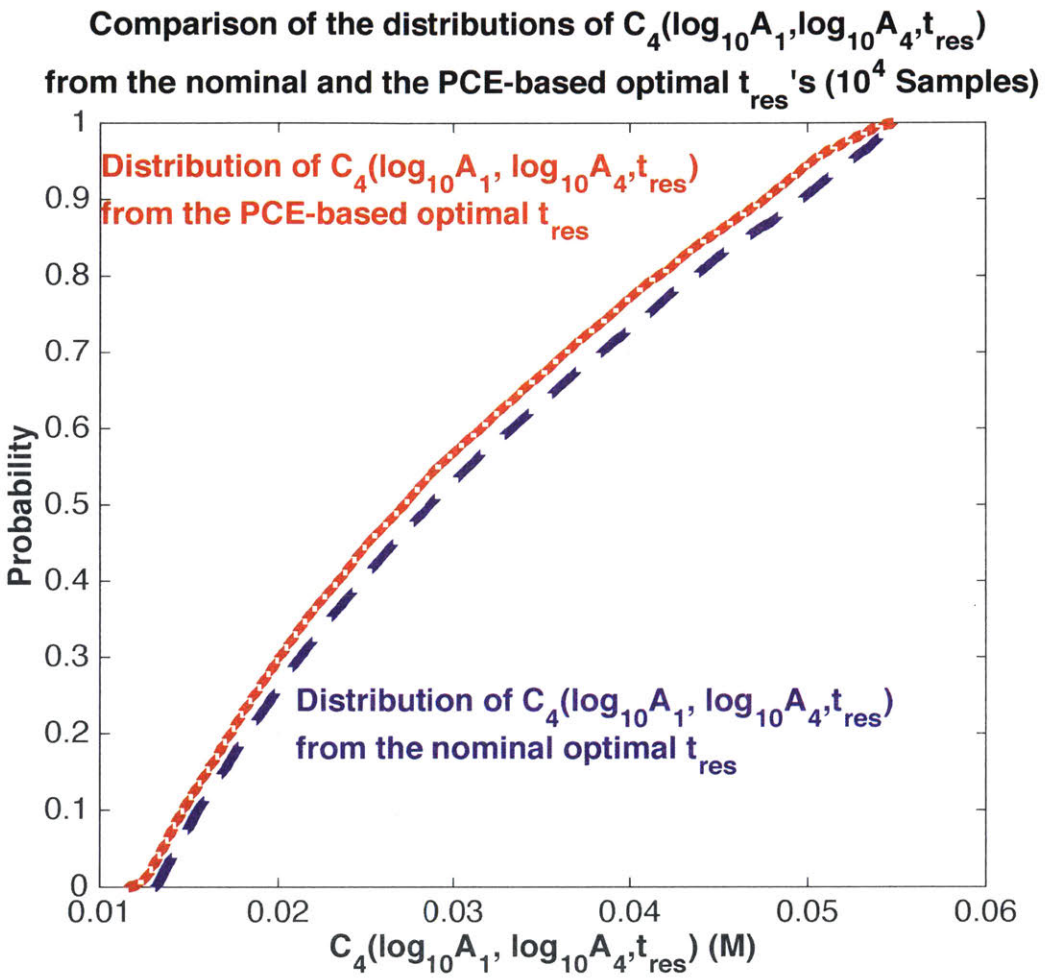

Figure 2-6: Comparison of the probability distribution functions of the intermediate concentration $C_4(\log_{10}A_1, \log_{10}A_4, t_{\text{res}})$ from the nominal and the polynomial chaos-based optimal residence times for the tubular reactor case study, constructed from 10^4 Monte Carlo simulation samples.

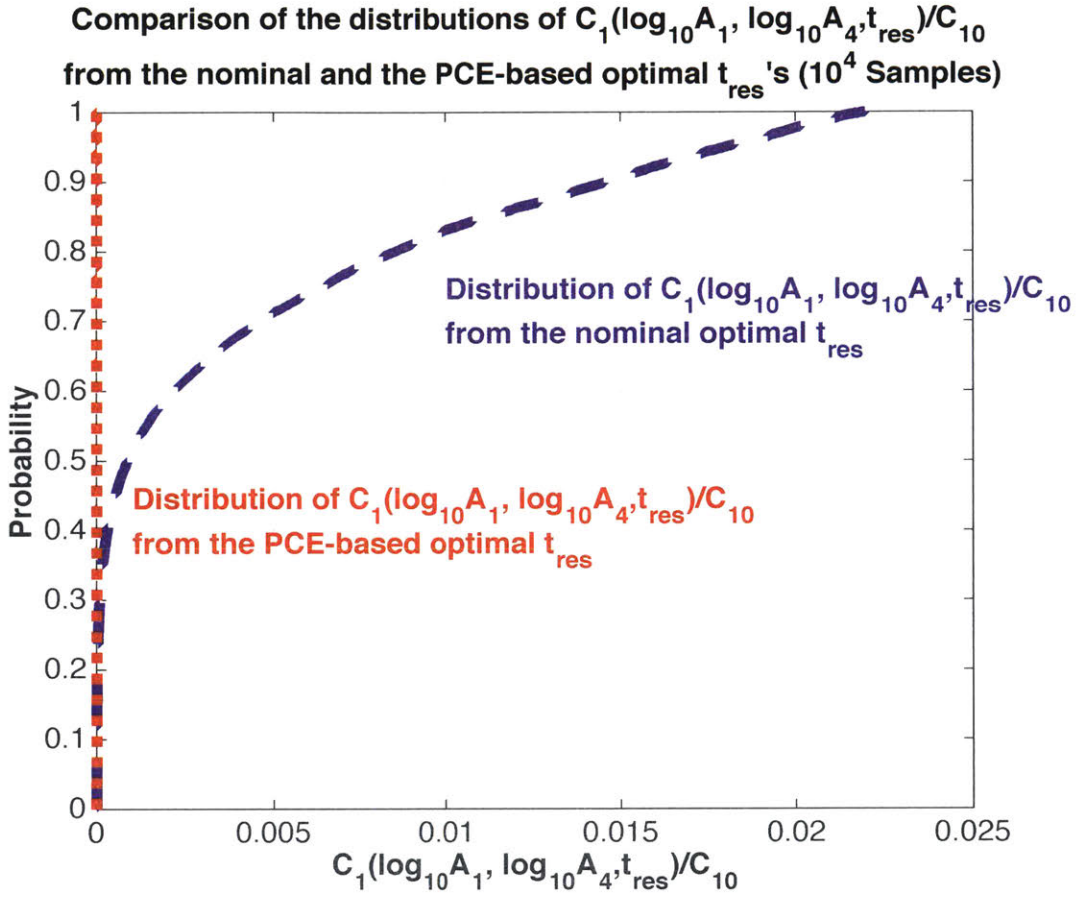

Figure 2-7: Comparison of the probability distribution functions of $C_1(\log_{10}A_1, \log_{10}A_4, t_{\text{res}})/C_{10}$ from the nominal and the polynomial chaos-based optimal residence times for the tubular reactor case study, constructed from 10^4 Monte Carlo simulation samples.

Chapter 3

Stochastic Optimal Control

3.1 Introduction

The closed-loop stability and performance obtained by state- and output-feedback control systems can be sensitive to model uncertainties, which has motivated numerous studies on the synthesis of robust control insensitive to uncertainties, e.g.,.[3,9,26,29,34] The static or reduced-order dynamic output-feedback control synthesis problem for both nominal and uncertain systems is NP-hard,[44] which implies that standard linear matrix inequalities (LMIs) and other convex optimization formulations do not exist.

The vast majority of the output-feedback and broader control literature adopts a worst-case design strategy to ensure stability and achieve a desired performance bound for all possible uncertainties. This worst-case approach tends to produce highly conservative performance because the worst-case scenario may have vanishingly low probability of occurrence. In addition, most worst-case robust control approaches are limited to a few uncertainty structures only, such as norm-bounded, affine, polytopic, and integral quadratic uncertainty.[34] A general nonlinear uncertainty structure cannot be effectively addressed without introducing overbounding.

In contrast to a worst-case performance bound, the practical interest in the performance variation or dispersion across the uncertainty region has motivated recent research on probabilistic robustness.[34] The design objective either relaxes the worst-case performance bound to a probability-guaranteed performance bound,[45,51] or optimizes

the averaged performance at the expense of a slightly worst performance bound.[5,14] Most of this research is limited to polytopic uncertainty,[5,51] or affine dependence on multiplicative white noises.[14] The randomized algorithm proposed in[45] can address general nonlinear dependence on uncertain parameters, but can be computationally demanding since a large number of samples is often needed.

The above observations have further led to robust control research that aims at addressing averaged performance in the presence of general nonlinear dependence on probabilistic time-invariant real parametric uncertainties. Such a uncertainty description is commonly generated by parameter identification techniques, but is poorly suited for any existing robust control design methods mentioned above. This robust optimal control design is non-trivial because uncertainty propagation in such an uncertain system is no longer a Markov process.[33] As a computationally efficient non-sampling approach for quantifying uncertainty propagation, polynomial chaos (PC) theory builds the foundation of a recent promising solution to this problem.[22] PC theory allows characterization of the evolution of probability distributions of the underlying stochastic system states by a high-dimensional expanded deterministic system describing the evolution dynamics of the polynomial chaos expansion (PCE) coefficients. Thus the control synthesis problem can be solved by using the expanded system. Up to now, the existing PCE-based control methods have been restricted to stability analysis[11,18,27] and state-feedback control.[4,12,33] The publications did not simultaneously address both time-invariant random parametric uncertainties and time-varying stochastic external disturbances, because the required number of PCE terms goes to infinity as time grows.[24] Moreover, due to trunction errors introduced by using finite-order PCEs, stability and performance derived for the expanded system may not be automatically achieved by the original system.[27] Although increasing the PCE order can alleviate the effect of PCE truncation errors, it may result in significant increase in computation complexity as the state dimension of the expanded system factorially grows with the PCE order.

In this chapter, PCE-based $\mathcal{H}_2$ static and dynamic *output-feedback* controls are investigated subject to general nonlinear dependence on probabilistic time-invariant

parametric uncertainties. The developed PCE-based expanded system includes the effect of probabilistic parametric uncertainties as well as white process and measurement noises. Moreover, the approximation errors introduced by the PCE truncations are captured by time-varying norm-bounded uncertainties whose bound is used as a robustifying tuning parameter. Based on the above expanded system, a nominal $\mathcal{H}_2$ synthesis approach is proposed when neglecting PCE truncation errors, while a $\mathcal{H}_2$ guaranteed cost control is adopted to cope with PCE truncation errors. The use of a robustifying parameter enforces closed-loop stability without resorting to a high-order PCE, thus avoiding high computational complexity due to a large number of PCE terms. In the synthesis and post-analysis iterations, a bisection algorithm is proposed to find the smallest robustifying parameter that ensures probabilistic closed-loop stability. In contrast, further analysis shows that the Monte-Carlo sampling based $\mathcal{H}_2$ synthesis is much less computationally efficient, and converges to imposing conservative worst-case stability constraints as the number of samples grows to infinity.

This chapter also extends the results to the $\mathcal{H}_\infty$ control objective, which is the first time that a rigorous theoretical framework has been developed for that objective. The same approach to ensuring closed-loop stability is applicable, and similar results are observed for a case study as was obtained for the $\mathcal{H}_2$ control objective.

This chapter is organized as follows. Section 3.2 states the probabilistic robust $\mathcal{H}_2$ control problem. Section 3.3 reviews preliminaries on PC theory and analyzes the effect of PCE truncation errors. Our proposed static and dynamic output-feedback control synthesis approaches are presented in Sections 3.4 and 3.5, respectively. Section 3.6 compares the approaches to Monte-Carlo sampling based synthesis. Section 3.8 extends the results to the $\mathcal{H}_\infty$ control objective. The simulation studies in Sections 3.7 and 3.9 demonstrate the advantages of using PC-based $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimization methods, respectively. Finally, some concluding remarks are presented in Section 3.10.

3.2 Problem Statement

Consider the linear time-invariant dynamical system described by

$$\dot{\mathbf{x}}(t,\boldsymbol{\eta}) = \mathbf{A}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) + \mathbf{B}(\boldsymbol{\eta})\mathbf{u}(t,\boldsymbol{\eta}) \tag{3.1a}$$

$$\mathbf{z}(t,\boldsymbol{\eta}) = \mathbf{C}_{\mathbf{z}}\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{D}_{\mathbf{zw}}\mathbf{w}(t) + \mathbf{D}_{\mathbf{z}}\mathbf{u}(t,\boldsymbol{\eta}) \tag{3.1b}$$

$$\mathbf{y}(t,\boldsymbol{\eta}) = \mathbf{C}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{D}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) \tag{3.1c}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the state, $\mathbf{u} \in \mathbb{R}^{n_u}$ is the control input, $\mathbf{w} \in \mathbb{R}^{n_w}$ is the stochastic disturbance or noise, $\mathbf{y} \in \mathbb{R}^{n_y}$ is the measured output, and $\mathbf{z} \in \mathbb{R}^{n_z}$ is the controlled output related to the performance of the control system. Since $\mathbf{A}$, $\mathbf{B_w}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D_w}$ in (3.1) are general nonlinear functions of a random parameter vector $\boldsymbol{\eta} \in \mathbb{R}^{n_\xi}$, the system state $\mathbf{x}$, control input $\mathbf{u}$, measured output $\mathbf{y}$, and controlled output $\mathbf{z}$ all depend on $\boldsymbol{\eta}$. Note that $\mathbf{C_z}$, $\mathbf{D_{zw}}$, and $\mathbf{D_z}$ in (3.1b) are independent of the uncertain parameter vector $\boldsymbol{\eta}$, because (3.1b) is determined by the control performance specifications. The objective of this chapter is to design

(i) a static output-feedback (SOF) controller

$$\mathbf{u}(t,\boldsymbol{\eta}) = \mathbf{K}\mathbf{y}(t,\boldsymbol{\eta}) \tag{3.2}$$

(ii) a dynamic output-feedback (DOF) controller

$$\begin{aligned} \dot{\mathbf{x}}_K(t,\boldsymbol{\eta}) &= \mathbf{A}_K\mathbf{x}_K(t,\boldsymbol{\eta}) + \mathbf{B}_K\mathbf{y}(t,\boldsymbol{\eta}) \\ \mathbf{u}(t,\boldsymbol{\eta}) &= \mathbf{C}_K\mathbf{x}_K(t,\boldsymbol{\eta}) + \mathbf{D}_K\mathbf{y}(t,\boldsymbol{\eta}) \end{aligned} \tag{3.3}$$

that minimizes the averaged $\mathcal{H}_2$ norm of the closed-loop system $\mathcal{T}_{\mathbf{zw}}(\boldsymbol{\eta})$ from the noisy input $\mathbf{w}$ to the controlled output $\mathbf{z}$, accounting for the time-invariant probabilistic parametric uncertainties $\boldsymbol{\eta}$, i.e.,

$$\min \mathbb{E}_{\boldsymbol{\eta}} \left\{ \|\mathcal{T}_{\mathbf{zw}}(\boldsymbol{\eta})\|_2^2 \right\}. \tag{3.4}$$

The finiteness of the $\mathcal{H}_2$ norm of $\mathcal{T}_{\mathbf{zw}}(\boldsymbol{\eta})$ requires $\mathbf{D_{zw}} + \mathbf{D_z K D_w} = \mathbf{0}$ for the SOF case and $\mathbf{D_{zw}} + \mathbf{D_z} \mathbf{D}_K \mathbf{D_w} = \mathbf{0}$ for the DOF case.

The above problem cannot be effectively addressed by most existing worst-case robust control methods due to the general nonlinear uncertainty structure. Inspired by[12] and citations therein, the PC theory is adopted to quantify the dependence of the squared $\mathcal{H}_2$ norm $\|\mathcal{T}_{\mathbf{zw}}(\boldsymbol{\eta})\|_2^2$ on $\boldsymbol{\eta}$. Specifically, the substitution of state $\mathbf{x}$, control input $\mathbf{u}$, controlled output $\mathbf{z}$, and measured output $\mathbf{y}$ with their PCE approximations transforms the original stochastic system (3.1) into a high-dimensional expanded system describing the dynamics of PCE coefficients. The $\mathcal{H}_2$ control synthesis is then solved by using the transformed expanded system.

The proposed approach aims at improving the existing PCE-based control design methodology by (i) explicitly taking into account stochastic disturbance $\mathbf{w}$; (ii) proposing systematic design procedures with a robustifying parameter to cope with PCE approximation errors which could destabilize the closed-loop system if neglected.

3.3 Polynomial chaos approximation to stochastic linear system

This section provides a brief introduction of polynomial chaos approximation to the stochastic linear system (3.1) using Galerkin projection, and then shows how the PCE truncation errors affect the PCE-approximated closed-loop dynamics.

3.3.1 Polynomial chaos expansion

For a random vector $\boldsymbol{\eta}$, a function $\psi(\boldsymbol{\eta}) : \mathbb{R}^{n_\xi} \to \mathbb{R}$ with a finite second-order moment admits a PCE[50]

$$\psi(\boldsymbol{\eta}) = \sum_{i=0}^{\infty} \psi_i \phi_i(\boldsymbol{\eta}), \tag{3.5}$$

where $\{\psi_i\}$ denotes the expansion coefficients, and $\{\phi_i(\boldsymbol{\eta})\}$ denotes the multivariate PC bases in terms of $\boldsymbol{\eta}$. By using the Askey scheme of orthogonal polynomial bases, the above expansion exponentially converges in the $\mathcal{L}_2$ sense, which results in accurate

approximations even with a relatively small number of terms.[50] These basis functions are orthogonal with respect to the probabilistic distribution $\mu(\boldsymbol{\eta})$ of the random vector $\boldsymbol{\eta}$, as shown in (2.8). Throughout this chapter, $\phi_i(\boldsymbol{\eta})$'s are normalized such that $\langle \phi_i^2(\boldsymbol{\eta}) \rangle = 1$.

In practical computations, a PCE with an infinite number of terms (3.5) needs to be truncated to a finite degree p,

$$\psi(\boldsymbol{\eta}) \approx \hat{\psi}(\boldsymbol{\eta}) = \sum_{i=0}^{N_p} \psi_i \phi_i(\boldsymbol{\eta}). \tag{3.6}$$

The total number of terms in (3.6) is $N_p + 1 = \frac{(n_\xi + p)!}{n_\xi! p!}$, depending on the dimension n_ξ of ξ and the highest order p of the retained polynomials $\{\phi_i(\boldsymbol{\eta})\}_{i=0}^{N_p}$.

3.3.2 Galerkin projection for stochastic linear system

Let s_i denote the ith component of a vector $\mathbf{s}$. The scalar $s_i(t, \boldsymbol{\eta})$ is expressed as

$$s_i(t, \boldsymbol{\eta}) = \hat{s}_i(t, \boldsymbol{\eta}) + \tilde{s}_i(t, \boldsymbol{\eta}),$$

where

$$\hat{s}_i(t, \boldsymbol{\eta}) = \sum_{j=0}^{N_p} s_{i,j}(t) \phi_j(\boldsymbol{\eta}) \tag{3.7}$$

is a truncated PCE with a degree p, and $\tilde{s}_i(t, \boldsymbol{\eta})$ represents the truncation error. Define

$$\begin{aligned}
\hat{\mathbf{s}}(t, \boldsymbol{\eta}) &= \begin{bmatrix} \hat{s}_1(t, \boldsymbol{\eta}) & \hat{s}_2(t, \boldsymbol{\eta}) & \cdots & \hat{s}_{n_s}(t, \boldsymbol{\eta}) \end{bmatrix}^\top, \\
\tilde{\mathbf{s}}(t, \boldsymbol{\eta}) &= \begin{bmatrix} \tilde{s}_1(t, \boldsymbol{\eta}) & \tilde{s}_2(t, \boldsymbol{\eta}) & \cdots & \tilde{s}_{n_s}(t, \boldsymbol{\eta}) \end{bmatrix}^\top, \\
\mathbf{s}_i^\top(t) &= \begin{bmatrix} s_{i,0}(t) & s_{i,1}(t) & \cdots & s_{i,N_p}(t) \end{bmatrix}, \\
\boldsymbol{\phi}(\boldsymbol{\eta}) &= \begin{bmatrix} \phi_0^\top(\boldsymbol{\eta}) & \phi_1^\top(\boldsymbol{\eta}) & \cdots & \phi_{N_p}^\top(\boldsymbol{\eta}) \end{bmatrix}^\top, \\
\mathbf{s}_{\mathrm{PCE}}(t) &= \begin{bmatrix} \mathbf{s}_1(t) & \cdots & \mathbf{s}_{n_s}(t) \end{bmatrix},
\end{aligned}$$

then the PCE approximation of the vector $\mathbf{s}(t,\boldsymbol{\eta})$ can be written as

$$\mathbf{s}(t,\boldsymbol{\eta}) = \hat{\mathbf{s}}(t,\boldsymbol{\eta}) + \tilde{\mathbf{s}}(t,\boldsymbol{\eta}) = \mathbf{s}_{\mathrm{PCE}}^{\top}(t)\boldsymbol{\phi}(\boldsymbol{\eta}) + \tilde{\mathbf{s}}(t,\boldsymbol{\eta}). \tag{3.8}$$

Let $\mathrm{vec}(\cdot)$ represent the vectorization of a matrix, then define the PCE coefficient vector

$$\mathbf{S}(t) = \mathrm{vec}\left(\mathbf{s}_{\mathrm{PCE}}(t)\right). \tag{3.9}$$

With $\mathbf{s}$ representing $\mathbf{x}$, $\mathbf{u}$, $\mathbf{y}$, or $\mathbf{z}$, the PCE coefficient vectors $\mathbf{X}$, $\mathbf{U}$, $\mathbf{Y}$, and $\mathbf{Z}$ are defined as in (3.9).

In the Galerkin projection, the PCEs of $\mathbf{x}(t,\boldsymbol{\eta})$ and $\mathbf{u}(t,\boldsymbol{\eta})$ in the form of (3.8) are inserted into (3.1a) to give

$$\begin{aligned}\dot{\mathbf{x}}_{\mathrm{PCE}}^{\top}(t)\boldsymbol{\phi}(\boldsymbol{\eta}) = {} & \mathbf{A}(\boldsymbol{\eta})\mathbf{x}_{\mathrm{PCE}}^{\top}(t)\boldsymbol{\phi}(\boldsymbol{\eta}) + \mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) \\ & + \mathbf{B}(\boldsymbol{\eta})\mathbf{u}_{\mathrm{PCE}}^{\top}(t)\boldsymbol{\phi}(\boldsymbol{\eta}) + \mathbf{r}_{\mathbf{x}}(t,\boldsymbol{\eta}), \qquad (3.10)\end{aligned}$$

$$\mathbf{r}_{\mathbf{x}}(t,\boldsymbol{\eta}) = -\dot{\tilde{\mathbf{x}}}(t,\boldsymbol{\eta}) + \mathbf{A}(\boldsymbol{\eta})\tilde{\mathbf{x}}(t,\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\tilde{\mathbf{u}}(t,\boldsymbol{\eta}). \tag{3.11}$$

Note that the high-order term $\mathbf{r}_{\mathbf{x}}(t,\boldsymbol{\eta})$ results from the PCE truncation errors $\tilde{\mathbf{x}}(t,\boldsymbol{\eta})$ and $\tilde{\mathbf{u}}(t,\boldsymbol{\eta})$. Then, transposing (3.10) and multiplying on the left by $\boldsymbol{\phi}$ gives

$$\begin{aligned}\boldsymbol{\phi}(\boldsymbol{\xi})\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\dot{\mathbf{x}}_{\mathrm{PCE}}(t) = {} & \boldsymbol{\phi}(\boldsymbol{\xi})\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\mathbf{x}_{\mathrm{PCE}}(t)\mathbf{A}^{\top}(\boldsymbol{\eta}) \\ & + \boldsymbol{\phi}(\boldsymbol{\eta})\mathbf{w}^{\top}(t)\mathbf{B}_{\mathbf{w}}^{\top}(\boldsymbol{\eta}) + \boldsymbol{\phi}(\boldsymbol{\xi})\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\mathbf{u}_{\mathrm{PCE}}(t)\mathbf{B}^{\top}(\boldsymbol{\eta}) \\ & + \boldsymbol{\phi}(\boldsymbol{\eta})\mathbf{r}_{\mathbf{x}}^{\top}(t,\boldsymbol{\eta}). \qquad (3.12)\end{aligned}$$

With $\otimes$ representing the Kronecker product, vectorizing (3.12) results in

$$\begin{aligned}[\mathbf{I}_{n_x} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^{\top})]\dot{\mathbf{X}} = {} & [\mathbf{A} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^{\top})]\mathbf{X} + (\mathbf{B}_{\mathbf{w}} \otimes \boldsymbol{\phi})\mathbf{w} \\ & + [\mathbf{B} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^{\top})]\mathbf{U} + \mathrm{vec}(\boldsymbol{\phi}\mathbf{r}_{\mathbf{x}}^{\top}), \qquad (3.13)\end{aligned}$$

according to the following property of Kronecker product:[6]

$$\text{vec}(ABC) = (C^\top \otimes A)\text{vec}(B). \tag{3.14}$$

The dependence on t and $\boldsymbol{\eta}$ in (3.13) is not explicitly expressed for the sake of brevity. By taking the expectation with respect to $\boldsymbol{\eta}$ on both sides of (3.13), it follows that the expanded system

$$\dot{\mathbf{X}} = \mathcal{A}\mathbf{X} + \mathcal{B}_{\mathbf{w}}\mathbf{w} + \mathcal{B}\mathbf{U} + \mathbf{R}_{\mathbf{x}}(t), \tag{3.15}$$

describes the dynamics of the PCE coefficient vector $\mathbf{X}$, with

$$\mathcal{A} = \mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{A} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^\top)\},\ \mathcal{B}_{\mathbf{w}} = \mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{B}_{\mathbf{w}} \otimes \boldsymbol{\phi}\}, \tag{3.16}$$

$$\mathcal{B} = \mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{B} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^\top)\},\ \mathbf{R}_{\mathbf{x}}(t) = \mathbb{E}_{\boldsymbol{\eta}}\{\text{vec}\left(\boldsymbol{\phi}(\boldsymbol{\eta})\mathbf{r}_{\mathbf{x}}^\top(t,\boldsymbol{\eta})\right)\}. \tag{3.17}$$

The above equation is obtained since $\mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{I}_{n_x} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^\top)\}$ is an identity matrix according to (2.8). Note that $\mathcal{A}$, $\mathcal{B}_{\mathbf{w}}$, and $\mathcal{B}$ are time-invariant matrices, while $\mathbf{R}_{\mathbf{x}}(t)$ is a time-varying error term since the high-order term $\mathbf{r}_{\mathbf{x}}(t,\boldsymbol{\eta})$ is not orthogonal to the low-order PC basis vector $\boldsymbol{\phi}(\boldsymbol{\eta})$.

Following similar procedures, the controlled output equation (3.1b), the measured output equation (3.1c), the SOF controller (3.2), and the DOF controller (3.3) can be transformed into

- expanded controlled output equation

$$\mathbf{Z} = \mathcal{C}_{\mathbf{Z}}\mathbf{X} + \mathcal{D}_{\mathbf{Zw}}\mathbf{w} + \mathcal{D}_{\mathbf{Z}}\mathbf{U}, \tag{3.18}$$

- expanded measured output equation

$$\mathbf{Y} = \mathcal{C}\mathbf{X} + \mathcal{D}_{\mathbf{w}}\mathbf{w} + \mathbf{R}_{\mathbf{y}}(t), \tag{3.19}$$

- expanded SOF controller

$$\mathbf{U} = \mathcal{K}\mathbf{Y}, \tag{3.20}$$

- expanded DOF controller

$$\begin{aligned} \dot{\mathbf{X}}_K &= \mathcal{A}_K \mathbf{X}_K + \mathcal{B}_K \mathbf{Y} \\ \mathbf{U} &= \mathcal{C}_K \mathbf{X}_K + \mathcal{D}_K \mathbf{Y} \end{aligned} \tag{3.21}$$

respectively, where

$$\mathcal{C}_{\mathbf{Z}} = \mathbf{C}_{\mathbf{z}} \otimes \mathbf{I}_{N_p+1},\ \mathcal{D}_{\mathbf{Zw}} = \mathbf{D}_{\mathbf{zw}} \otimes \mathbf{I}_{N_p+1}, \tag{3.22}$$

$$\mathcal{D}_{\mathbf{Z}} = \mathbf{D}_{\mathbf{z}} \otimes \mathbf{I}_{N_p+1},\ \mathcal{C} = \mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{C} \otimes (\boldsymbol{\phi}\boldsymbol{\phi}^\top)\}, \tag{3.23}$$

$$\mathcal{D}_{\mathbf{w}} = \mathbb{E}_{\boldsymbol{\eta}}\{\mathbf{D}_{\mathbf{w}} \otimes \boldsymbol{\phi}\},\ \mathcal{K} = \mathbf{K} \otimes \mathbf{I}_{N_p+1}, \tag{3.24}$$

$$\mathcal{A}_K = \mathbf{A}_K \otimes \mathbf{I}_{N_p+1},\ \mathcal{B}_K = \mathbf{B}_K \otimes \mathbf{I}_{N_p+1}, \tag{3.25}$$

$$\mathcal{C}_K = \mathbf{C}_K \otimes \mathbf{I}_{N_p+1},\ \mathcal{D}_K = \mathbf{D}_K \otimes \mathbf{I}_{N_p+1}, \tag{3.26}$$

$\mathbf{R}_{\mathbf{y}}(t)$ represents the error term similar to $\mathbf{R}_{\mathbf{x}}(t)$ in (3.15), as a result of the PCE truncation errors. Note that (3.18), (3.20), and (3.21) are exact equations, because the matrices in the controlled output equation (3.1b), the SOF controller (3.2), and the DOF controller (3.3) do not depend on the uncertain parameters $\boldsymbol{\eta}$, and then the high-order terms of these equations are orthogonal to the low-order PC basis $\boldsymbol{\phi}(\boldsymbol{\eta})$.

As the uncertain system (3.1) has general nonlinear uncertainty structure depending on $\boldsymbol{\eta}$, the matrices $\mathcal{A}$, $\mathcal{B}_{\mathbf{w}}$, $\mathcal{B}$, $\mathcal{C}_{\mathbf{Z}}$, $\mathcal{D}_{\mathbf{Zw}}$, $\mathcal{D}_{\mathbf{Z}}$, $\mathcal{C}$, and $\mathcal{D}_{\mathbf{w}}$ defined in (3.16), (3.17), (3.22), (3.23), and (3.24) are time-invariant, and can be obtained via numerical integration.[50]

3.3.3 Error analysis of PCE-approximated dynamics

Most existing PCE-based control design methods, e.g.,,[12] relied on the expanded system (3.15)-(3.21) but neglected the error terms $\mathbf{R}_{\mathbf{x}}(t)$ and $\mathbf{R}_{\mathbf{y}}(t)$ therein. In this case, even though the expanded system is stabilized, the closed-loop system might

be unstable due to perturbations from the neglected error terms $\mathbf{R}_{\mathbf{x}}(t)$ and $\mathbf{R}_{\mathbf{y}}(t)$, as will be analyzed in the following.

Combining the expanded open-loop dynamics (3.15) and the expanded SOF controller (3.20) gives the expanded closed-loop system

$$\begin{aligned}\dot{\mathbf{X}} &= (\mathcal{A}+\mathcal{B}\mathcal{K}\mathcal{C})\,\mathbf{X} + (\mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}})\,\mathbf{w} \\ &\quad + \mathbf{R}_{\mathbf{x}}(t) + \mathcal{B}\mathcal{K}\mathbf{R}_{\mathbf{y}}(t) \\ \mathbf{Z} &= (\mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C})\,\mathbf{X} + (\mathcal{D}_{\mathbf{Z}\mathbf{w}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}})\,\mathbf{w} \\ &\quad + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathbf{R}_{\mathbf{y}}(t).\end{aligned} \tag{3.27}$$

Since the error terms $\mathbf{R}_{\mathbf{x}}(t)$ and $\mathbf{R}_{\mathbf{y}}(t)$ are determined by the PCE truncation errors $\tilde{\mathbf{x}}(t,\boldsymbol{\eta})$, $\tilde{\mathbf{u}}(t,\boldsymbol{\eta})$, and $\tilde{\mathbf{y}}(t,\boldsymbol{\eta})$ which are dynamically coupled with the truncated PCE $\hat{\mathbf{x}}(t,\boldsymbol{\eta})$, it is reasonable to introduce time-varying matrices $\mathcal{F}_{\mathbf{x}}(t)$ and $\mathcal{F}_{\mathbf{y}}(t)$ such that

$$\mathbf{R}_{\mathbf{x}}(t) = \mathcal{F}_{\mathbf{x}}(t)\mathbf{X}(t),\ \mathbf{R}_{\mathbf{y}}(t) = \mathcal{F}_{\mathbf{y}}(t)\mathbf{X}(t).$$

Then the expanded closed-loop system (3.27) can be written as

$$\mathcal{T}_{\mathbf{Z}\mathbf{w}} = \left[\begin{array}{c|c} \mathcal{A}+\mathcal{B}\mathcal{K}\mathcal{C}+\mathcal{F}_{\mathbf{x}}(t)+\mathcal{B}\mathcal{K}\mathcal{F}_{\mathbf{y}}(t) & \mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}} \\ \hline \mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{F}_{\mathbf{y}}(t) & \mathcal{D}_{\mathbf{Z}\mathbf{w}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}} \end{array}\right]. \tag{3.28}$$

With similar procedures, the expanded closed-loop system under the expanded DOF controller (3.21) is

$$\bar{\mathcal{T}}_{\mathbf{Z}\mathbf{w}} = \left[\begin{array}{cc|c} \mathcal{A}+\mathcal{B}\mathcal{D}_K\mathcal{C}+\mathcal{F}_{\mathbf{x}}(t)+\mathcal{B}\mathcal{D}_K\mathcal{F}_{\mathbf{y}}(t) & \mathcal{B}\mathcal{C}_K & \mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{D}_K\mathcal{D}_{\mathbf{w}} \\ \mathcal{B}_K\mathcal{C}+\mathcal{B}_K\mathcal{F}_{\mathbf{y}}(t) & \mathcal{A}_K & \mathcal{B}_K\mathcal{D}_{\mathbf{w}} \\ \hline \mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{D}_K\mathcal{C}+\mathcal{D}_{\mathbf{Z}}\mathcal{D}_K\mathcal{F}_{\mathbf{y}}(t) & \mathcal{D}_{\mathbf{Z}}\mathcal{C}_K & \mathcal{D}_{\mathbf{Z}\mathbf{w}}+\mathcal{D}_{\mathbf{Z}}\mathcal{D}_K\mathcal{D}_{\mathbf{w}} \end{array}\right]. \tag{3.29}$$

In (3.28) and (3.29), the effect of the PCE truncation errors is described by the multiplicative uncertainties $\mathcal{F}_{\mathbf{x}}(t)\mathbf{X}$ and $\mathcal{F}_{\mathbf{y}}(t)\mathbf{X}$. They would destabilize the closed-loop system in certain cases if completely neglected. Note that $\mathcal{F}_{\mathbf{x}}(t)$ and $\mathcal{F}_{\mathbf{y}}(t)$ are impossible to be quantified before the control design since the PCE truncation errors

are determined by the closed-loop system dynamics. Still, the introduction of $\mathcal{F}_{\mathbf{x}}(t)$ and $\mathcal{F}_{\mathbf{y}}(t)$ in (3.28) and (3.29) allows a systematic robust control design in Sections 3.4.2 and 3.5.2 to explicitly account for the PCE truncation errors.

3.4 Static output-feedback synthesis using polynomial chaos

In this section, two $\mathcal{H}_2$ synthesis methods are proposed for the SOF problem formulated in Section 3.2, using the PCE-based expanded systems (3.15)–(3.20).

3.4.1 $\mathcal{H}_2$ static output-feedback synthesis

For the above purpose, the following time-domain characterization of the $\mathcal{H}_2$ norm is adopted:[43,53]

$$\|\mathcal{T}_{\mathbf{zw}}(\boldsymbol{\eta})\|_2^2 = \sum_{k=1}^{n_w} \left\{ \int_0^\infty \|\mathbf{z}_k(t,\boldsymbol{\eta})\|_2^2 \, \mathrm{d}t : \mathbf{w}(t) = \mathbf{e}_k \delta(t) \right\}, \tag{3.30}$$

where $\mathbf{z}_k(t,\boldsymbol{\eta})$ denotes the output response to the impulse disturbance $\mathbf{w}(t) = \mathbf{e}_k\delta(t)$, with $\delta(t)$ representing the unite impulse and $\mathbf{e}_k$ the kth column of an identity matrix $\mathbf{I}_{n_w}$. Such a time-domain interpretation is related to the impulse-to-energy system gain, and allows generalizing the $\mathcal{H}_2$ norm from linear time-invariant systems in the frequency domain to time-varying systems, see Section 4.7 of.[43] By substituting (3.30) into (3.4) and interchanging the order of expectation, summation and integration, the cost function (3.4) is rewritten as

$$\min_{\mathbf{K}} \sum_{k=1}^{n_w} \left\{ \int_0^\infty \mathbb{E}_{\boldsymbol{\eta}}\{\|\mathbf{z}_k(t,\boldsymbol{\eta})\|_2^2\} \, \mathrm{d}t : \mathbf{w}(t) = \mathbf{e}_k \delta(t) \right\}. \tag{3.31}$$

According to (3.8) and the property (3.14), the measured output $\mathbf{z}(t,\boldsymbol{\eta})$ can be

written as

$$\begin{aligned}\mathbf{z}(t,\boldsymbol{\eta}) &\approx \mathrm{vec}\left(\mathbf{z}_{\mathrm{PCE}}^{\top}(t)\boldsymbol{\phi}(\boldsymbol{\eta})\right)\\ &= \left(\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\otimes\mathbf{I}_{n_z}\right)\mathrm{vec}\left(\mathbf{z}_{\mathrm{PCE}}^{\top}(t)\right)\end{aligned}$$

by neglecting the truncation error $\tilde{\mathbf{z}}(t,\boldsymbol{\eta})$. This leads to

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\eta}}\{\|\mathbf{z}(t,\boldsymbol{\eta})\|_2^2\} &\approx \mathrm{vec}\left(\mathbf{z}_{\mathrm{PCE}}^{\top}(t)\right)^{\top}\mathbf{W}\mathrm{vec}\left(\mathbf{z}_{\mathrm{PCE}}^{\top}(t)\right)\\ &= \mathrm{vec}^{\top}\left(\mathbf{z}_{\mathrm{PCE}}(t)\right)\mathrm{vec}\left(\mathbf{z}_{\mathrm{PCE}}(t)\right)\\ &= \|\mathbf{Z}(t)\|_2^2\end{aligned} \tag{3.32}$$

since

$$\mathbf{W} = \mathbb{E}_{\boldsymbol{\eta}}\left\{\left(\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\otimes\mathbf{I}_{n_z}\right)^{\top}\left(\boldsymbol{\phi}^{\top}(\boldsymbol{\eta})\otimes\mathbf{I}_{n_z}\right)\right\} = \mathbf{I}_{n_z(N_p+1)}$$

as a result of (2.8) and the normalized PCE basis functions. From (3.32), the cost function (3.31) can be approximated by

$$\min_{\mathbf{K}}\sum_{k=1}^{n_w}\left\{\int_0^{\infty}\|\mathbf{Z}_k(t)\|_2^2\,\mathrm{d}t : \mathbf{w}(t) = \mathbf{e}_k\delta(t)\right\}, \tag{3.33}$$

where $\mathbf{Z}_k(t)$ is the PCE coefficient vector of the output response $\mathbf{z}_k(t,\boldsymbol{\eta})$ in (3.31). By doing so, the original SOF problem (3.31) of minimizing the averaged $\mathcal{H}_2$ norm with respect to probabilistic uncertainties is transformed into a standard nominal $\mathcal{H}_2$ SOF problem (3.33) for the linear time-invariant expanded system (3.15)–(3.20), when the error terms $\mathbf{R}_{\mathbf{x}}(t)$ and $\mathbf{R}_{\mathbf{y}}(t)$ are neglected.

The problem (3.33) aims at minimizing the $\mathcal{H}_2$ norm of the expanded closed-loop system (3.28) from the disturbance $\mathbf{w}$ to the measured output $\mathbf{Z}$. Note that $\mathcal{D}_{\mathbf{Zw}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}} = \mathbf{0}$ is required to obtain a finite $\mathcal{H}_2$ norm. Then it follows standard procedures to convert (3.33) into the following optimization problem

$$\begin{aligned}&\min_{\mathbf{P}_s,\mathbf{K}}\ \mathrm{trace}\left(\mathcal{B}_{\mathbf{w},\mathrm{cs}}^{\top}\mathbf{P}_s\mathcal{B}_{\mathbf{w},\mathrm{cs}}\right)\\ &\text{s.t. } \mathbf{P}_s > 0,\ \mathcal{A}_{\mathrm{cs}}^{\top}\mathbf{P}_s + \mathbf{P}_s\mathcal{A}_{\mathrm{cs}} + \mathcal{C}_{\mathbf{Z},\mathrm{cs}}^{\top}\mathcal{C}_{\mathbf{Z},\mathrm{cs}} < 0\end{aligned}$$

where

$$\begin{aligned} \mathcal{A}_{\text{cs}} &= \mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C}, \qquad \mathcal{B}_{\mathbf{w},\text{cs}} = \mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}, \\ \mathcal{C}_{\mathbf{Z},\text{cs}} &= \mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C}, \end{aligned} \tag{3.34}$$

and the subscript "cs" indicates that all three matrices are for the closed-loop system under SOF. This above problem can be equivalently transformed into

$$\begin{aligned} &\min_{\mathbf{\Lambda}_s, \mathbf{Q}_s, \mathbf{K}} \operatorname{trace}(\mathbf{Q}_s) \\ &\text{s.t.} \begin{bmatrix} \mathbf{Q}_s & \star \\ \mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}} & \mathbf{\Lambda}_s \end{bmatrix} > 0, \\ &\begin{bmatrix} \text{He}\{(\mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C})\mathbf{\Lambda}_s\} & \star \\ (\mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C})\mathbf{\Lambda}_s & -\mathbf{I} \end{bmatrix} < 0, \end{aligned} \tag{3.35}$$

using $\mathbf{\Lambda}_s = \mathbf{P}_s^{-1}$, and $\mathcal{K}$ defined in (3.24), according to the Schure complement lemma. $\text{He}\{\cdot\}$ denotes the sum of a square matrix and its transpose. As in any standard SOF problem, the second matrix inequality in (3.35) is a bilinear matrix inequality (BMI)[46] due to the multiplication between $\mathbf{\Lambda}_s$ and $\mathcal{K}$ as well as the special structure of $\mathcal{K} = \mathbf{K} \otimes \mathbf{I}_{N_p+1}$. The BMI problem (3.35) can be solved by numerical solvers such as PENBMI.[25]

The above $\mathcal{H}_2$ synthesis problem extends the PCE-based linear quadratic regulation method proposed in[12] by including the additive stochastic disturbance $\mathbf{w}$. It has the same limitation as[12] as a result of neglecting the PCE truncation errors, i.e., the above synthesis might fail to stabilize the original dynamics (3.1a). Specifically, the accuracy of the PCE approximation degrades over time, and the perturbation from the neglected error terms $\mathbf{R}_{\mathbf{x}}(t)$ and $\mathbf{R}_{\mathbf{y}}(t)$ in the closed-loop system (3.28) grows. When the control action does not provide sufficient compensation for such a model-plant mismatch, the system state would diverge. Few existing PCE-based control designs explicitly address this problem. The commonly adopted remedy in literature is the use of higher-order PCE approximations at the cost of larger computational burden when solving (3.35). However, as the PCE degree p increases, the number of PCE terms increases factorially, and then the involved computational burden rapidly

grows and easily become prohibited.

3.4.2 $\mathcal{H}_2$ guaranteed cost static output-feedback synthesis

In order to compensate for the error terms of the PCE-approximated system without significantly increasing the PCE degree, the expanded closed-loop system (3.28) with norm-bounded time-varying uncertainties $\|\mathcal{F}_{\mathbf{x}}(t)\| \leq \rho_{\mathbf{x}}$ and $\|\mathcal{F}_{\mathbf{y}}(t)\| \leq \rho_{\mathbf{y}}$ is considered, where $\|\cdot\|$ represents the spectral norm of a matrix. Tuning of these robustifying parameters will be discussed in Section 3.4.3.

First, the expanded closed-loop system (3.28) is rewritten as

$$\begin{aligned}
\dot{\mathbf{X}} &= \mathcal{A}_{\text{cs}}\mathbf{X} + \mathcal{G}_s\boldsymbol{\omega}_s + \mathcal{B}_{\mathbf{w},\text{cs}}\mathbf{w} \\
\mathbf{Z} &= \mathcal{C}_{\mathbf{Z},\text{cs}}\mathbf{X} + \mathcal{L}_s\boldsymbol{\omega}_s \\
\boldsymbol{\omega}_s &= \begin{bmatrix} \boldsymbol{\Delta}_{\mathbf{x}}(t) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Delta}_{\mathbf{y}}(t) \end{bmatrix} \boldsymbol{\psi}_s, \ \boldsymbol{\psi}_s = \begin{bmatrix} \rho_{\mathbf{x}}\mathbf{I} \\ \rho_{\mathbf{y}}\mathbf{I} \end{bmatrix} \mathbf{X}
\end{aligned} \tag{3.36}$$

with $\mathcal{A}_{\text{cs}}$, $\mathcal{B}_{\mathbf{w},\text{cs}}$, and $\mathcal{C}_{\mathbf{Z},\text{cs}}$ defined in (3.34), and

$$\mathcal{G}_s = \begin{bmatrix} \mathbf{I} & \mathcal{B}\mathcal{K} \end{bmatrix}, \ \mathcal{L}_s = \begin{bmatrix} \mathbf{0} & \mathcal{D}_{\mathbf{Z}}\mathcal{K} \end{bmatrix}, \tag{3.37}$$

$$\boldsymbol{\Delta}_{\star}(t) = \frac{\mathcal{F}_{\star}(t)}{\rho_{\star}}, \ \|\boldsymbol{\Delta}_{\star}(t)\| \leq 1, \ \star \text{ presents } \mathbf{x} \text{ or } \mathbf{y}. \tag{3.38}$$

With the same procedures in Section 3.4.1, the design objective (3.31) is transformed into (3.33).

Theorem 1. *The closed-loop system (3.36) is quadratically stable for all* $\|\Delta_{\mathbf{x}}(t)\| \leq 1$ *and* $\|\Delta_{\mathbf{y}}(t)\| \leq 1$ *if and only if there exist* $\mathbf{P}_s > 0$ *and a scalar* $\mu_s > 0$ *such that*

$$\begin{aligned}
&\begin{bmatrix} \mathcal{A}_{\text{cs}}^{\top}\mathbf{P}_s + \mathbf{P}_s\mathcal{A}_{\text{cs}} + \mu_s(\rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2)\mathbf{I} & \star \\ \mathcal{G}_s^{\top}\mathbf{P}_s & -\mu_s\mathbf{I} \end{bmatrix} \\
&\qquad + \begin{bmatrix} \mathcal{C}_{\mathbf{Z},cs}^{\top} \\ \mathcal{L}_s^{\top} \end{bmatrix} \begin{bmatrix} \mathcal{C}_{\mathbf{Z},cs} & \mathcal{L}_s \end{bmatrix} < 0.
\end{aligned} \tag{3.39}$$

Suppose the above statement holds, then the $\mathcal{H}_2$ cost function (3.33) is upper bounded by $\text{trace}\left\{\mathcal{B}_{\mathbf{w},\text{cs}}^{\top}\mathbf{P}_s\mathcal{B}_{\mathbf{w},\text{cs}}\right\}$.

Proof 1. *According to the small gain theorem (Theorem 4.7.1 and Definition 4.7.3 in[43]), the system (3.36) is quadratically stable if and only if there exist* $\bar{\mathbf{P}}_s > 0$ *such that*

$$\begin{bmatrix} \mathcal{A}_{cs}^{\top}\bar{\mathbf{P}}_s + \bar{\mathbf{P}}_s\mathcal{A}_{cs} + (\rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2)\mathbf{I} & \star \\ \mathcal{G}_s^{\top}\bar{\mathbf{P}}_s & -\mathbf{I} \end{bmatrix} < 0. \tag{3.40}$$

The above inequality is equivalent to

$$\begin{bmatrix} \mathcal{A}_{cs}^{\top}\bar{\mathbf{P}}_s + \bar{\mathbf{P}}_s\mathcal{A}_{cs} + (\rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2)\mathbf{I} & \star \\ \mathcal{G}_s^{\top}\bar{\mathbf{P}}_s & -\mathbf{I} \end{bmatrix} + \epsilon \begin{bmatrix} \mathcal{C}_{\mathbf{Z},cs}^{\top} \\ \mathcal{L}_s^{\top} \end{bmatrix} \begin{bmatrix} \mathcal{C}_{\mathbf{Z},cs} & \mathcal{L}_s \end{bmatrix} < 0,$$

for a sufficiently small $\epsilon > 0$. *Dividing the left-hand side by* ϵ, *and defining* $\mathbf{P}_s = \epsilon^{-1}\bar{\mathbf{P}}_s > 0$ *and* $\mu_s = \epsilon^{-1} > 0$, *(3.39) is derived.*

Define a quadratic Lyapunov function $V(\mathbf{X}) = \mathbf{X}^{\top}\mathbf{P}_s\mathbf{X}$. *By multiplying (3.39) with* $\begin{bmatrix} \mathbf{X}^{\top} & \boldsymbol{\omega}_s^{\top} \end{bmatrix}$ *to its left and with* $\begin{bmatrix} \mathbf{X}^{\top} & \boldsymbol{\omega}_s^{\top} \end{bmatrix}^{\top}$ *to its right, we have*

$$\dot{V}(\mathbf{X}) < -\mathbf{Z}^{\top}\mathbf{Z} + \mu_s\left(\boldsymbol{\omega}_s^{\top}\boldsymbol{\omega}_s - \boldsymbol{\psi}_s^{\top}\boldsymbol{\psi}_s\right).$$

Let $\mathbf{X}_k(t)$, $\mathbf{Z}_k(t)$, $\boldsymbol{\psi}_{s,k}(t)$, *and* $\boldsymbol{\omega}_{s,k}(t)$ *denote the impulse responses to the unit-impulse input* $\mathbf{w}(t) = \mathbf{e}_k\delta(t)$ *in the kth coordinate of* $\mathbf{w}$. *Integrating both sides from* $t = 0$ *to* ∞ *leads to*

$$\begin{aligned} V(\mathbf{X}_k(0)) &> \int_0^{\infty} \|\mathbf{Z}_k(t)\|_2^2\, dt \\ &\quad + \mu_s \int_0^{\infty} \|\boldsymbol{\psi}_{s,k}(t)\|_2^2 - \|\boldsymbol{\omega}_{s,k}(t)\|_2^2\, dt \\ &\geq \int_0^{\infty} \|\mathbf{Z}_k(t)\|_2^2\, dt \end{aligned}$$

by using $\|\boldsymbol{\psi}_{s,k}(t)\|_2^2 > \|\boldsymbol{\omega}_{s,k}(t)\|_2^2$ *according to* (3.38). *Since the impulse response to*

the unit-impulse $\mathbf{w}(t) = \mathbf{e}_k\delta(t)$ *is equivalent to the initial state response under the initial condition* $\mathbf{X}_k(0) = \mathcal{B}_{\mathbf{w},c}\mathbf{e}_k$*, the upper bound of the* $\mathcal{H}_2$ *cost function (3.33) is*

$$\begin{aligned}\sum_{k=0}^{n_w}\int_0^\infty \|\mathbf{Z}_k(t)\|^2\,dt &< \sum_{k=0}^{n_w} V(\mathbf{X}_k(0)) \\ &= \sum_{k=0}^{n_w}(\mathcal{B}_{\mathbf{w},cs}\mathbf{e}_k)^\top\mathbf{P}_s\mathcal{B}_{\mathbf{w},cs}\mathbf{e}_k \\ &= trace\{\mathcal{B}_{\mathbf{w},cs}^\top\mathbf{P}_s\mathcal{B}_{\mathbf{w},cs}\}.\end{aligned}$$

According to Theorem 1, the robust $\mathcal{H}_2$ control synthesis aims at minimizing the performance upper bound trace$\{\mathcal{B}_{\mathbf{w},\mathrm{cs}}^\top\mathbf{P}_s\mathcal{B}_{\mathbf{w},\mathrm{cs}}\}$ under the constraint (3.39), which can be written as

$$\begin{aligned}&\min_{\mathbf{P}_s,\mathbf{K},\mu_s}\ \mathrm{trace}\,(\mathbf{Q}_s) \\ &\text{s.t.}\ \begin{bmatrix}\mathbf{Q}_s & \star \\ \mathbf{P}_s(\mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}) & \mathbf{P}_s\end{bmatrix} > 0, \\ &\begin{bmatrix}\mathrm{He}\{\mathbf{P}_s(\mathcal{A}+\mathcal{B}\mathcal{K}\mathcal{C})\}+\mu_s\rho^2\mathbf{I} & \star & \star & \star \\ \mathbf{P}_s & -\mu_s\mathbf{I} & \star & \star \\ \mathcal{K}^\top\mathcal{B}^\top\mathbf{P}_s & \mathbf{0} & -\mu_s\mathbf{I} & \star \\ \mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} & \mathbf{0} & \mathcal{D}_{\mathbf{Z}}\mathcal{K} & -\mathbf{I}\end{bmatrix} < 0\end{aligned} \tag{3.41}$$

with $\rho^2 = \rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2$. Both inequalities in (3.41) are BMIs due to the multiplication between $\mathbf{P}_s$ and $\mathcal{K}$ as well as the structure of $\mathcal{K} = \mathbf{K}\otimes\mathbf{I}_{N_p+1}$.

3.4.3 Post-analysis of stability and parameter tuning

Since the PCE-based control is actually affected by the uncertainty matrices $\mathcal{F}_{\mathbf{x}}(t)$ and $\mathcal{F}_{\mathbf{y}}(t)$ in the closed-loop system, the norm bounds of these uncertainty matrices cannot be quantified before a control law is designed. Even with the closed-loop dynamics given, it is still extremely difficult, if not impossible, to verify these norm bounds over an infinite time horizon. Due to the above reasons, the robustifying parameter $\rho^2 = \rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2$ may not be verified, thus the controller generated by (3.41) dose not

necessarily imply stability of the original system. However, it is clear that tuning the robustifying parameter ρ involves trade-offs between stability and performance: by setting a larger ρ, the resulting controller enhances stability in the presence of larger PCE approximation errors, but it suffers from reduced performance due to the conservative bound ρ.

Considering the above facts, a nominal PCE-based synthesis (3.35) is first computed, and the resulting closed-loop system is analyzed for its stochastic stability. Here, stochastic stability is tested by computing the probability that $\mathbf{A}(\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{C}(\boldsymbol{\eta})$ is Hurwitz. Such a probabilistic stability analysis is used here, because stochastic stability in the presence of random parameters allows a set of unstable modes with measure zero, and it enables a probabilistic relaxation to the robust stability problem that is known to be NP-hard.[48] The probabilistic stability analysis proposed recently by[35] is adopted, which is a semi-definite program relaxation using moments of the uncertain parameters.

If the closed-loop system generated by (3.35) is stable with a probability higher than a predefined level γ, a stabilizing PCE-based control is found. Otherwise, the robust PCE-synthesis (3.41) is used, and a bisection algorithm is proposed to find the smallest parameter ρ such that the resulting closed-loop system is stable with a probability higher than γ, see Algorithm 1. Note that after m iterations, $\rho_U - \rho_L = 2^{-m}\rho_0$. Finite number of iterations are needed to achieve the predefined accuracy ϵ.

Algorithm 1 Bisection method for tuning ρ in (3.41)

Initialization: $\rho_L \leftarrow 0$, $\rho_U \leftarrow \rho_0$, where $\rho_0 > 0$ ensures that the closed-loop system resulted from (3.41) with $\rho = \rho_0$ is stable with a probability higher than γ
repeat
  $\rho \rightarrow \frac{1}{2}(\rho_L + \rho_U)$
  if the closed-loop system resulted from (3.41) with ρ is stable with a probability higher than γ **then**
    $\rho_U \leftarrow \rho$
  else
    $\rho_L \leftarrow \rho$
  end if
until $\rho_U - \rho_L \leq \epsilon$

Due to the use of the robustifying parameter ρ, certain conservativeness is intro-

duced. Such conservatism can be reduced by moderately increasing the PCE order p and accordingly reducing ρ. This is achieved at the cost of significantly higher computational load, since the size of the synthesis problem grows factorially with the PCE order.

3.5 Dynamic output-feedback synthesis using polynomial chaos

In this section, BMI synthesis conditions are derived for the PCE-based $\mathcal{H}_2$ DOF controller by reducing it to a SOF problem.

3.5.1 $\mathcal{H}_2$ dynamic output-feedback synthesis

When neglecting the PCE truncation errors, the cost function (3.33) is considered in the DOF synthesis for the expanded closed-loop system (3.29), i.e.,

$$\begin{aligned}\bar{\mathcal{T}}_{\mathbf{zw}} &= \left[\begin{array}{c|c} \bar{\mathcal{A}}+\bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{C}} & \bar{\mathcal{B}}_{\mathbf{w}}+\bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{D}}_{\mathbf{w}} \\ \hline \bar{\mathcal{C}}_{\mathbf{z}}+\bar{\mathcal{D}}_{\mathbf{z}}\bar{\mathcal{K}}\bar{\mathcal{C}} & \bar{\mathcal{D}}_{\mathbf{zw}}+\bar{\mathcal{D}}_{\mathbf{z}}\bar{\mathcal{K}}\bar{\mathcal{D}}_{\mathbf{w}} \end{array}\right] \\ &= \left[\begin{array}{c|c} \mathcal{A}_{\mathrm{cd}} & \mathcal{B}_{\mathbf{w},\mathrm{cd}} \\ \hline \mathcal{C}_{\mathbf{z},\mathrm{cd}} & \mathbf{0} \end{array}\right],\end{aligned} \tag{3.42}$$

where

$$\begin{aligned}&\bar{\mathcal{A}}=\begin{bmatrix}\mathcal{A} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix},\ \bar{\mathcal{B}}_{\mathbf{w}}=\begin{bmatrix}\mathcal{B}_w\\ \mathbf{0}\end{bmatrix},\ \bar{\mathcal{B}}=\begin{bmatrix}\mathcal{B} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix},\\ &\bar{\mathcal{C}}_{\mathbf{z}}=\begin{bmatrix}\mathcal{C}_{\mathbf{z}} & \mathbf{0}\end{bmatrix},\ \bar{\mathcal{D}}_{\mathbf{zw}}=\mathcal{D}_{\mathbf{zw}},\ \bar{\mathcal{D}}_{\mathbf{z}}=\begin{bmatrix}\mathcal{D}_{\mathbf{z}} & \mathbf{0}\end{bmatrix},\\ &\bar{\mathcal{C}}=\begin{bmatrix}\mathcal{C} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix},\ \bar{\mathcal{D}}_{\mathbf{w}}=\begin{bmatrix}\mathcal{D}_{\mathbf{w}}\\ \mathbf{0}\end{bmatrix},\ \bar{\mathcal{K}}=\begin{bmatrix}\mathcal{D}_K & \mathcal{C}_K\\ \mathcal{B}_K & \mathcal{A}_K\end{bmatrix},\end{aligned} \tag{3.43}$$

$\mathcal{A}_K$, $\mathcal{B}_K$, $\mathcal{C}_K$, $\mathcal{D}_K$ are defined in (3.25) and (3.26), the subscript "cd" indicates the closed-loop system matrices under DOF, and $\bar{\mathcal{D}}_{\mathbf{zw}}+\bar{\mathcal{D}}_{\mathbf{z}}\bar{\mathcal{K}}\bar{\mathcal{D}}_{\mathbf{w}}=\mathbf{0}$ is required to obtain a finite $\mathcal{H}_2$ norm. Comparing (3.42) with (3.28), it can be seen that the above DOF

synthesis problem is reducible to the SOF synthesis problem. Therefore, similarly to (3.35), the DOF synthesis problem is formulated as

$$\begin{aligned} &\min_{\mathbf{\Lambda}_d, \mathbf{Q}_d, \bar{\mathcal{K}}} \ \text{trace}\,(\mathbf{Q}_d) \\ &\text{s.t.} \begin{bmatrix} \mathbf{Q}_d & \star \\ \bar{\mathcal{B}}_{\mathbf{w}} + \bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{D}}_{\mathbf{Zw}} & \mathbf{\Lambda}_d \end{bmatrix} > 0, \\ &\begin{bmatrix} \text{He}\{(\bar{\mathcal{A}} + \bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{C}})\mathbf{\Lambda}_d\} & \star \\ (\bar{\mathcal{C}}_{\mathbf{Z}} + \bar{\mathcal{D}}_{\mathbf{Z}}\bar{\mathcal{K}}\bar{\mathcal{C}})\mathbf{\Lambda}_d & -\mathbf{I} \end{bmatrix} < 0, \end{aligned} \tag{3.44}$$

using notations in (3.42) and (3.43). The second inequality in (3.44) is a BMI with respect to the decision variables. In the conventioal output-feedback synthesis, the additional structure in a full-order DOF controller allows the use of congruence transformation and change of variables to obtain a LMI synthesis problem.[42] The same strategy, however, does not work for the above PCE-based DOF synthesis conditions using the expanded controller (3.21) because of the special structure of controller parameters as shown in (3.25) and (3.26). Therefore, a BMI solver is needed for (3.44).

3.5.2 $\mathcal{H}_2$ guaranteed cost dynamic output-feedback synthesis

By following Section 3.4.2, the effect of PCE truncation errors captured by $\mathcal{F}_{\mathbf{x}}(t)$ and $\mathcal{F}_{\mathbf{y}}(t)$ in (3.29) is regarded as time-varying norm-bounded uncertainties, i.e., $\|\mathcal{F}_{\mathbf{x}}(t)\| \leq \rho_{\mathbf{x}}$ and $\|\mathcal{F}_{\mathbf{y}}(t)\| \leq \rho_{\mathbf{y}}$. With the notations in (3.43), the expanded closed-loop dynamics (3.29) and the expanded output equations (3.18)–(3.19) can be rewritten as

$$\begin{aligned} \dot{\mathbf{X}}_d &= \mathcal{A}_{\text{cd}}\mathbf{X}_d + \mathcal{G}_d\boldsymbol{\omega}_d + \mathcal{B}_{\mathbf{w},\text{cd}}\mathbf{w} \\ \mathbf{Z} &= \mathcal{C}_{\mathbf{Z},\text{cd}}\mathbf{X}_d + \mathcal{L}_d\boldsymbol{\omega}_d \\ \boldsymbol{\omega}_d &= \begin{bmatrix} \mathbf{\Delta}_{\mathbf{x}}(t) & \mathbf{0} \\ \mathbf{0} & \mathbf{\Delta}_{\mathbf{y}}(t) \end{bmatrix} \boldsymbol{\psi}_d, \ \boldsymbol{\psi}_d = \begin{bmatrix} \rho_{\mathbf{x}}\mathbf{I} & \mathbf{0} \\ \rho_{\mathbf{y}}\mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{X}_d \end{aligned} \tag{3.45}$$

with $\mathbf{X}_d = \begin{bmatrix}\mathbf{X}^\top & \mathbf{X}_K^\top\end{bmatrix}^\top$, $\mathcal{A}_{\text{cd}}$, $\mathcal{B}_{\mathbf{w},\text{cd}}$, and $\mathcal{C}_{\mathbf{Z},\text{cd}}$ defined in (3.42), $\Delta_{\mathbf{x}}$ and $\Delta_{\mathbf{y}}$ defined in (3.38), and

$$\mathcal{G}_d = \begin{bmatrix}\mathbf{I} & \mathcal{B}\mathcal{D}_K \\ \mathbf{0} & \mathcal{B}_K\end{bmatrix}, \quad \mathcal{L}_d = \begin{bmatrix}\mathbf{0} & \mathcal{D}_{\mathbf{Z}}\mathcal{D}_K\end{bmatrix}. \tag{3.46}$$

By directly applying Theorem 1 to the system (3.45), the robust $\mathcal{H}_2$ DOF synthesis is formulated as as

$$\begin{aligned}
&\min_{\mathbf{P}_d,\mathbf{Q}_d,\mu_d,\bar{\mathcal{K}}} \ \text{trace}\,(\mathbf{Q}_d) \\
&\text{s.t.} \begin{bmatrix}\mathbf{Q}_d & \star \\ \mathbf{P}_d\left(\bar{\mathcal{B}}_{\mathbf{w}} + \bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{D}}_{\mathbf{Z}\mathbf{w}}\right) & \mathbf{P}_d\end{bmatrix} > 0, \\
&\begin{bmatrix}\text{He}\{\mathbf{P}_d(\bar{\mathcal{A}} + \bar{\mathcal{B}}\bar{\mathcal{K}}\bar{\mathcal{C}})\} + \mu_d\rho^2\mathbf{I} & \star & \star \\ \mathcal{G}_d^\top\mathbf{P}_d & -\mu_d\mathbf{I} & \star \\ \bar{\mathcal{C}}_{\mathbf{Z}} + \bar{\mathcal{D}}_{\mathbf{Z}}\bar{\mathcal{K}}\bar{\mathcal{C}} & \mathcal{L}_d & -\mathbf{I}\end{bmatrix} < 0
\end{aligned} \tag{3.47}$$

with $\mathcal{G}_d$ and $\mathcal{L}_d$ defined in (3.46), and $\rho^2 = \rho_{\mathbf{x}}^2 + \rho_{\mathbf{y}}^2$.

3.6 Comparison with Monte-Carlo sampling based $\mathcal{H}_2$ output-feedback synthesis

By following[4] and,[45] a Monte-Carlo based method is briefly reviewed here, to compare with the PCE-based synthesis proposed in the previous sections. For the sake of brevity, only the SOF case is discussed, and similar conclusions are applicable to the DOF case.

When applying the standard $\mathcal{H}_2$ synthesis conditions, the averaged $\mathcal{H}_2$ SOF problem stated in Section 3.2 can be formulated as

$$\begin{aligned}
&\min_{\mathbf{P}(\boldsymbol{\eta}),\mathbf{K}} \ \mathbb{E}_{\boldsymbol{\eta}}\left\{\text{trace}\left(\mathbf{B}_{\mathbf{w},\text{c}}^\top(\boldsymbol{\eta})\mathbf{P}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w},\text{c}}(\boldsymbol{\eta})\right)\right\} \\
&\quad \text{s.t. } \mathbf{P}(\boldsymbol{\eta}) > 0, \ \mathbb{E}_{\boldsymbol{\eta}}\left\{\mathbf{x}^\top(t,\boldsymbol{\eta})\boldsymbol{\Upsilon}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta})\right\} < 0,
\end{aligned} \tag{3.48}$$

where $\mathbf{P}(\boldsymbol{\eta}) \in \mathbb{R}^{n_x \times n_x}$ is a predefined function of $\boldsymbol{\eta}$, and

$$\begin{aligned}
\mathbf{A}_{\mathrm{c}}(\boldsymbol{\eta}) &= \mathbf{A}(\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{C}(\boldsymbol{\eta}), \\
\mathbf{B}_{\mathbf{w},\mathrm{c}}(\boldsymbol{\eta}) &= \mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{D}_{\mathbf{w}}(\boldsymbol{\eta}), \\
\mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta}) &= \mathbf{C}_{\mathbf{z}} + \mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{C}(\boldsymbol{\eta}), \\
\boldsymbol{\Upsilon}(\boldsymbol{\eta}) &= \mathbf{A}_{\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{P}(\boldsymbol{\eta}) + \mathbf{P}(\boldsymbol{\eta})\mathbf{A}_{\mathrm{c}}(\boldsymbol{\eta}) + \mathbf{C}_{\mathbf{z},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta}).
\end{aligned} \tag{3.49}$$

The Monte-Carlo-simulation-based approach uses a finite number of realizations of $\boldsymbol{\eta}$ to recast the above problem as

$$\min_{\substack{\boldsymbol{\Lambda}_1,\cdots,\boldsymbol{\Lambda}_N,\\ \mathbf{Q}_1,\cdots,\mathbf{Q}_N,\mathbf{K}}} \quad \frac{1}{N}\sum_{i=1}^{N} \mathrm{trace}(\mathbf{Q}_i) \tag{3.50a}$$

$$\text{s.t.} \quad \begin{bmatrix} \boldsymbol{\Lambda}_i\mathbf{A}_{\mathrm{c}}^{\top}(\boldsymbol{\eta}_i) + \mathbf{A}_{\mathrm{c}}(\boldsymbol{\eta}_i)\boldsymbol{\Lambda}_i & \star \\ \mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta}_i)\boldsymbol{\Lambda}_i & -\mathbf{I} \end{bmatrix} < 0, \tag{3.50b}$$

$$\begin{bmatrix} \mathbf{Q}_i & \star \\ \mathbf{B}_{\mathbf{w},\mathrm{c}}(\boldsymbol{\eta}_i) & \boldsymbol{\Lambda}_i \end{bmatrix} > 0,\ \boldsymbol{\Lambda}_i > 0,\ i = 1,\cdots,N,$$

where $\{\boldsymbol{\eta}_i\}$ are sampled from the probability distribution of $\boldsymbol{\eta}$, N is the number of samples, and each pair of $\boldsymbol{\Lambda}_i$ and $\mathbf{Q}_i$ is applied to a different sample. The inequalities in (3.50) are converted from $\mathbf{B}_{\mathbf{w},\mathrm{c}}^{\top}(\boldsymbol{\eta}_i)\mathbf{P}_i\mathbf{B}_{\mathbf{w},\mathrm{c}}(\boldsymbol{\eta}_i) < \mathbf{Q}_i$ and $\boldsymbol{\Upsilon}(\boldsymbol{\eta}_i) < 0$ by using $\boldsymbol{\Lambda}_i = \mathbf{P}_i^{-1}$, with $\boldsymbol{\Upsilon}(\boldsymbol{\eta}_i)$ defined in (3.49).

To achieve a satisfactory approximation to the original problem (3.48), a large number of samples are necessary, as analyzed in Sections 8.3 and 10.3 of.[45] Moreover, the solution of (3.50) is actually random due to its dependence on random samples. A large sample size is also necessary so that the variance of the solution is reduced to a satisfactory level. This leads to heavy computational load when solving the problem (3.50), as illustrated by a numerical example in.[4] In contrast, the PCE approximation exponentially converges with its order increasing, thus usually a relatively small order is needed. As a result, solving the PCE-based synthesis problems derived in Sections 3.4 and 3.5 can be much more efficient. Even when a small PCE order results in

PCE truncation errors to be accounted for, not only the PCE order p but also the robustifying parameter ρ introduced in Sections 3.4.2 and 3.5.2 are available to enhance our proposed PCE-based design without significantly increasing computational complexity.

Another limitation of the Monte-Carlo-simulation-based approach lies in replacing the stochastic stability condition $\mathbb{E}_{\boldsymbol{\eta}}\left\{\mathbf{x}^{\top}(t,\boldsymbol{\eta})\boldsymbol{\Upsilon}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta})\right\} < 0$ in (3.48) by (3.50b). This is conservative, because (3.50b) converges to a worst-case robust stability constraint as the sample size increases.

3.7 Case Study

In this section, both a numerical example and a continuous pharmaceutical manufacturing example are adopted to illustrate the efficacy of the proposed PCE-based $\mathcal{H}_2$ control compared to the standard nominal $\mathcal{H}_2$ control and the Monte-Carlo based approach in Section 3.6. Due to the page limit, only the results of SOF controls are presented.

3.7.1 A numerical example

Consider the system (3.1) whose parameters are as below:

$$\begin{aligned}
\mathbf{A}(\xi) &= \begin{bmatrix} 0.2+0.2\xi^3 & -0.4 \\ 0.1 & 0.5 \end{bmatrix}, \ \mathbf{B_w} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \ \mathbf{B} = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix}, \\
\mathbf{C_z} &= \mathbf{I}_2, \ \mathbf{D_{zw}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \mathbf{D_z} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \\
\mathbf{C} &= \begin{bmatrix} 0.1 & 0.1 \\ 0.3 & 0.4 \end{bmatrix}, \ \mathbf{D_w} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},
\end{aligned} \tag{3.51}$$

with the uncertain parameter ξ uniformly distributed over the interval $[-1, 1]$.

Four $\mathcal{H}_2$ SOF control synthesis methods are implemented for comparisons: (i) standard nominal control synthesis; (ii) worst-case robust control synthesis; (iii)

Monte-Carlo sampling based control synthesis; (iv) our proposed PCE-based control synthesis. The standard nominal controller is $\mathbf{K}_{\text{no}} = [270.4, -87.20]$ for the nominal system with $\xi = 0$. The worst-case guaranteed cost controller $\mathbf{K}_{\text{wc}} = [607.8, -188.5]$ accounts for the polytopic uncertainty $\xi^3 \in [-1, 1]$ in (3.51) by solving[13]

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{Q},\mathbf{K}} \ \text{trace}(\mathbf{Q}) \\ &\text{s.t.} \begin{bmatrix} \text{He}\{\mathbf{P}(\mathbf{A}(\xi_i) + \mathbf{BKC})\} & \star \\ \mathbf{C}_{\mathbf{z},\text{c}} & -\mathbf{I} \end{bmatrix} < 0, \ i = 1, 2, \\ &\begin{bmatrix} \mathbf{Q} & \star \\ \mathbf{P}(\mathbf{B}_{\mathbf{w}} + \mathbf{BKD}_{\mathbf{w}}) & \mathbf{P} \end{bmatrix} > 0, \end{aligned}$$

with $\xi_1 = -1$ and $\xi_2 = 1$, respectively. To cope with the probabilistic uncertain parameter ξ, the PCE-based control synthesis (3.35) is applied with different PCE orders. As illustrated in Figure 3-1, the two components of the controller converge to $\mathbf{K}_{\text{pc}} = [310.6, -98.50]$ as the PCE order increases to 10. The Monte-Carlo sampling-based formulation (3.50) converges to $\mathbf{K}_{\text{mc}} = [310.9, -98.58]$ for $N = 10^4$ samples, as shown in Figure 3-2. Figure 3-3 shows how the problem size in terms of the number of variables changes with the number of Monte-Carlo samples in (a) and the PCE order (b). The PCE-based control synthesis has a significantly reduced number of variables, approximately by two orders of magnitude.

In Figure 3-4(a), the standard nominal control fails to stabilize the system when the uncertain parameter ξ becomes close to 1. Both the worst-case robust control and the PCE-based control with a 10th-order PCE succeed in stabilizing the system for all possible parametric uncertainties. It can be also seen that although the PCE-based control gives a slightly larger worst-case performance bound for $\xi = 1$, it indeed achieves a smaller averaged performance than the worst-case robust control.

With the same PCE order as above, the robustness to PCE truncation errors is further investigated by introducing a robustifying parameter ρ. When setting $\rho = 0$, $\rho = 0.75 \times 10^{-4}$, and $\rho = 1.5 \times 10^{-3}$, the robust PCE-based control synthesis (3.41) gives almost the same controller as that from solving (3.35), $\mathbf{K}_{\text{pc}} = [353.0, -111.0]$,

and $\mathbf{K}_{\text{pc}} = [432.9, -134.7]$, respectively. As shown in Figure 3-4(b), the PCE-based control with $\rho = 0.75 \times 10^{-4}$ achieves both a smaller worst-case performance bound and a better averaged performance than that with $\rho = 0$. A larger ρ, i.e., $\rho = 1.5 \times 10^{-3}$ for this example, further reduces the worst-case performance bound while increases the $\mathcal{H}_2$ norm for ξ's far from 1.

3.7.2 A continuous pharmaceutical manufacturing example

As illustrated in Figure 3-5, a continuous stirred-tank reactor (CSTR) in a pharmaceutical manufacturing process includes four reactions among five species A, B, C, D, and E. The CSTR has two inlet flows of reagents A and B, whose flow rates are $q_1 = 1.1\,\text{L/sec}$ and $q_2 = 1\,\text{L/sec}$, respectively. The control objective is to produce the pharmaceutical D at a desired concentration $C_{D,\text{ref}}$ by manipulating the inlet concentration $C_{A,\text{in}}$ of reagent A as the control input. The inlet concentration $C_{B,\text{in}} = 3.5\,\text{M}$ of reagent B is much larger than $C_{A,\text{in}}$. Thus the reactor concentration C_B of reagent B remains approximately constant, i.e., $C_B \approx C_{B,\text{in}}$. According to the kinetics of reactions in,[38] the system dynamics relevant to the above setpoint tracking problem is

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} C_A \\ C_D \end{bmatrix} = &\begin{bmatrix} -(k_1+k_2)C_B - \frac{q_1+q_2}{V} & 0 \\ k_2 C_B & -k_4 C_B - \frac{q_1+q_2}{V} \end{bmatrix}\begin{bmatrix} C_A \\ C_D \end{bmatrix} \\ &+ \begin{bmatrix} \frac{q_1}{V} \\ 0 \end{bmatrix}(C_{A,\text{in}} + \Delta C_{A,in}),\end{aligned} \tag{3.52}$$

where $V = 5\,\text{L}$ is the reactor volume, $\Delta C_{A,\text{in}}$ denotes noises in the input channel, and the reaction rates k_3 and k_4 are 3.264×10^{-3} and $0.01591\ \text{sec}^{-1}\text{M}^{-1}$, respectively. The uncertain reaction rates k_1 and k_2 are uniformly distributed over $[0.1515, 1.001]$ and $[0.03023, 1.065]\ \text{sec}^{-1}\text{M}^{-1}$, respectively, and have their nominal values of 0.5763 and $0.5476\ \text{sec}^{-1}\text{M}^{-1}$, respectively. The available measured output includes only the concentration C_D of reagent D, i.e., $C_{D,\text{meas}} = C_D + \Delta C_D$, with ΔC_D representing the measurement noise.

In order to achieve zero set-point tracking error, a proportional integral control

law

$$C_{A,\text{in}} = K_\text{P}\left(C_{D,\text{meas}} - C_{D,\text{ref}}\right) + K_\text{I}\eta$$

is used, with $\dot{\eta} = C_{D,\text{meas}} - C_{D,\text{ref}}$. This leads to the closed-loop system

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} C_A \\ C_D \\ \eta \end{bmatrix} &= \begin{bmatrix} -(k_1+k_2)C_B - \frac{q_1+q_2}{V} & \frac{K_\text{P} q_1}{V} & \frac{K_\text{I} q_1}{V} \\ k_2 C_B & -k_4 C_B - \frac{q_1+q_2}{V} & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} C_A \\ C_D \\ \eta \end{bmatrix} \\ &\quad + \begin{bmatrix} \frac{q_1}{V} & \frac{K_\text{P} q_1}{V} \\ 0 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \Delta C_{A,\text{in}} \\ \Delta C_D \end{bmatrix} + \begin{bmatrix} -\frac{K_\text{P} q_1}{V} \\ 0 \\ -1 \end{bmatrix} C_{D,\text{ref}}.\end{aligned}$$

The control synthesis problem aims at minimizing the $\mathcal{H}_2$-norm of the closed-loop system from $[\Delta C_{A,\text{in}}\ \Delta C_D]^\top$ to

$$\begin{aligned}\mathbf{z} &= \begin{bmatrix} \omega_e \\ 0 \end{bmatrix}\left(C_D - C_{D,\text{ref}}\right) + \begin{bmatrix} 0 \\ \omega_u \end{bmatrix} C_{A,\text{in}} \\ &= \begin{bmatrix} 0 & \omega_e & 0 \\ 0 & \omega_u K_\text{P} & \omega_u K_\text{I} \end{bmatrix}\begin{bmatrix} C_A \\ C_D \\ \eta \end{bmatrix} - \begin{bmatrix} \omega_e \\ \omega_u K_\text{P} \end{bmatrix} C_{D,\text{ref}},\end{aligned}$$

where $\omega_e = 1$ and $\omega_u = 3$ are the weights for the setpoint tracking error and the penalty on the control effort.

The standard nominal $\mathcal{H}_2$ control synthesis gives $[K_\text{P}, K_\text{I}] = [-34.52, -9.265]$. Accounting for the probabilistic uncertain reaction rates k_1 and k_2 are considered, the PCE-based $\mathcal{H}_2$ control synthesis obtains $[K_\text{P}, K_\text{I}] = [-40.85, -12.87]$, when the PCE order and the robustifying parameter ρ are set to 2 and 0, respectively. As shown in Figure 3-6, compared to the nominal control, the PCE-based control results in better distributions of $\mathcal{H}_2$ norm for small values of k_2 without sacrificing the $\mathcal{H}_2$-norm performance for large values of k_2. Note that only $\rho = 0$ is used for this particular example, because it has already ensured sufficiently fast convergence and

negligible PCE truncation errors. In this case, a positive ρ would significantly reduce the $\mathcal{H}_2$ performance, although it increases the stability margin.

3.8 $\mathcal{H}_\infty$ static output-feedback control

This section extends $\mathcal{H}_2$ results to the $\mathcal{H}_\infty$ control objective, which is more challenging to handle theoretically than the $\mathcal{H}_2$ control objective. This mathematical framework is the first theory and algorithms that have been developed for the $\mathcal{H}_\infty$ control objective with rigorous closed-loop stability guarantees. The theoretical derivation of the optimization over bilinear matrix inequalities for $\mathcal{H}_\infty$-optimal control requires a potentially conservative step compared to the $\mathcal{H}_2$-optimal control derivation, but is observed to generate similarly good performance in a case study. The same approach to ensure closed-loop stability for the $\mathcal{H}_2$ control objective applies to the $\mathcal{H}_\infty$ control objective.

3.8.1 PCE expanded system

Recall that the ordinary differential-algebraic equations for the vectors of polynomial chaos (PC) coefficients for the state $\mathbf{x}$, controlled output $\mathbf{z}$, and measured output $\mathbf{y}$ including the effects of truncation errors $\mathbf{R}_\mathbf{x}(t)$ and $\mathbf{R}_\mathbf{y}(t)$ are

$$\begin{aligned}
\dot{\mathbf{X}} &= \mathcal{A}\mathbf{X} + \mathcal{B}_\mathbf{w}\mathbf{w} + \mathcal{B}\mathbf{U} + \mathbf{R}_\mathbf{x}(t), \\
\mathbf{Z} &= \mathcal{C}_\mathbf{Z}\mathbf{X} + \mathcal{D}_{\mathbf{Zw}}\mathbf{w} + \mathcal{D}_\mathbf{Z}\mathbf{U}, \\
\mathbf{Y} &= \mathcal{C}\mathbf{X} + \mathcal{D}_\mathbf{w}\mathbf{w} + \mathbf{R}_\mathbf{y}(t),
\end{aligned} \tag{3.53}$$

where the truncation errors are described by norm-bounded time-domain perturbations as

$$\mathbf{R}_\mathbf{x}(t) = \rho_\mathbf{x}\Delta_\mathbf{x}(t)\mathbf{X},\ \mathbf{R}_\mathbf{y}(t) = \rho_\mathbf{y}\Delta_\mathbf{y}(t)\mathbf{X},\ \|\Delta_\mathbf{x}(t)\| \leq 1,\ \|\Delta_\mathbf{y}(t)\| \leq 1. \tag{3.54}$$

Here this set of equations is called the *PCE expanded system.*

The dependency of the state equation for $\mathbf{x}$ and the output equation for $\mathbf{y}$ on the uncertain parameters leads to truncation errors in the associated PC coefficients. The vector $\mathbf{z}$ is the controlled output related to control specifications, so the matrices in $\mathbf{z} = \mathbf{C_z x} + \mathbf{D_{zw} w} + \mathbf{D_z u}$ do not depend on any uncertain parameters. As such, the equation for the PC coefficients for $\mathbf{z}$ in (3.53) does not have any truncation error.

3.8.2 Problem formulation

The optimal control problem for minimizing the expected value of the $\mathcal{H}_\infty$ norm can be written as

$$\begin{aligned} &\min_{\mathbf{K},\gamma^2} \gamma^2 \\ &\text{s.t. } \mathbf{E}_{\boldsymbol{\eta}}\{\|\mathbf{z}(t,\boldsymbol{\eta})\|^2_{L_2[0,\infty)}\} \le \gamma^2 \|\mathbf{w}(t)\|^2_{L_2[0,\infty)} \end{aligned} \tag{3.55}$$

Using $\|\mathbf{Z}(t)\|^2_{L_2[0,\infty)}$ to approximate $\mathbf{E}_{\boldsymbol{\eta}}\Big\{\|\mathbf{z}(t,\boldsymbol{\eta})\|^2_{L_2[0,\infty)}\Big\}$, this optimization can be approximated by

$$\begin{aligned} &\min_{\mathbf{K},\gamma^2} \gamma^2 \\ &\text{s.t. } \|\mathbf{Z}(t)\|^2_{L_2[0,\infty)} \le \gamma^2 \|\mathbf{w}(t)\|^2_{L_2[0,\infty)} \end{aligned} \tag{3.56}$$

so the PCE expanded system (3.53) can be used for control synthesis. Due to the PCE truncation errors, $\|\mathbf{Z}(t)\|^2_{L_2[0,\infty)}$ is actually smaller than $\mathbf{E}_{\boldsymbol{\eta}}\Big\{\|\mathbf{z}(t,\boldsymbol{\eta})\|^2_{L_2[0,\infty)}\Big\}$, thus the resulting L_2-induced norm can be larger than H_∞-norm of the expanded system, i.e., γ in (3.56).

3.8.3 PCE-based control with constant bounds on uncertainty

The values of $\rho_{\mathbf{x}}$ and $\rho_{\mathbf{y}}$ in (3.54) are specified as constants.

With the control law $\mathbf{u}(t,\xi) = \mathbf{K}\mathbf{y}(t,\xi)$ (i.e.,$\mathbf{U} = \mathcal{K}\mathbf{Y}$), the closed-loop PCE expanded system is

$$\begin{aligned} \dot{\mathbf{X}} &= (\mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C} + \rho_{\mathbf{x}}\Delta_{\mathbf{x}}(t) + \rho_{\mathbf{y}}\mathcal{B}\mathcal{K}\Delta_{\mathbf{y}})\mathbf{X} + (\mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}})\mathbf{w}, \\ \mathbf{Z} &= (\mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} + \rho_{\mathbf{y}}\mathcal{D}_{\mathbf{Z}}\mathcal{K}\Delta_{\mathbf{y}})\mathbf{X} + (\mathcal{D}_{\mathbf{Zw}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}})\mathbf{w}, \end{aligned} \tag{3.57}$$

Define the matrices

$$\begin{aligned}
\mathcal{A}_{\mathrm{c}} &= \mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C} + \rho_{\mathbf{x}}\Delta_{\mathbf{x}}(t) + \rho_{\mathbf{y}}\mathcal{B}\mathcal{K}\Delta_{\mathbf{y}}, \\
\mathcal{B}_{\mathbf{w},\mathrm{c}} &= \mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}, \\
\mathcal{C}_{\mathbf{Z},\mathrm{c}} &= \mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} + \rho_{\mathbf{y}}\mathcal{D}_{\mathbf{Z}}\mathcal{K}\Delta_{\mathbf{y}}, \\
\mathcal{D}_{\mathbf{Zw},\mathrm{c}} &= \mathcal{D}_{\mathbf{Zw}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}}.
\end{aligned} \tag{3.58}$$

According to Definition 2 of,[16] the system (3.57) is quadratically stable with an H_∞-norm bound γ if

$$\begin{aligned}
&\gamma^2 I - \mathcal{D}_{\mathbf{Zw},\mathrm{c}}^{\top}\mathcal{D}_{\mathbf{Zw},\mathrm{c}} > 0 \\
&\mathbf{P}\mathcal{A}_{\mathrm{c}} + \mathcal{A}_{\mathrm{c}}^{\top}\mathbf{P} + \mathcal{C}_{\mathbf{Z},\mathrm{c}}^{\top}\mathcal{C}_{\mathbf{Z},\mathrm{c}} + (\mathcal{C}_{\mathbf{Z},\mathrm{c}}^{\top}\mathcal{D}_{\mathbf{Zw},\mathrm{c}} + \mathbf{P}\mathcal{B}_{\mathbf{w},\mathrm{c}})(\gamma^2 I - \mathcal{D}_{\mathbf{Zw},\mathrm{c}}^{\top}\mathcal{D}_{\mathbf{Zw},\mathrm{c}})^{-1}(\mathcal{D}_{\mathbf{Zw},\mathrm{c}}^{\top}\mathcal{C}_{\mathbf{Z},\mathrm{c}} + \mathcal{B}_{\mathbf{w},\mathrm{c}}^{\top}\mathbf{P}) < 0.
\end{aligned} \tag{3.59}$$

This condition is equivalent to

$$\begin{bmatrix}
\mathbf{P}\mathcal{A}_{\mathrm{c}} + \mathcal{A}_{\mathrm{c}}^{\top}\mathbf{P} & \mathbf{P}\mathcal{B}_{\mathbf{w},\mathrm{c}} & \mathcal{C}_{\mathbf{Z},\mathrm{c}}^{\top} \\
\mathcal{B}_{\mathbf{w},\mathrm{c}}^{\top}\mathbf{P} & -\gamma^2\mathbf{I} & \mathcal{D}_{\mathbf{Zw},\mathrm{c}}^{\top} \\
\mathcal{C}_{\mathbf{Z},\mathrm{c}} & \mathcal{D}_{\mathbf{Zw},\mathrm{c}} & -\mathbf{I}
\end{bmatrix} < 0. \tag{3.60}$$

This linear matrix inequality forms the basis for a stability test for the closed-loop uncertain system in terms of the matrices of the PCE expanded system.

Theorem 2. *For the uncertainties in (3.54) with constant $\rho_{\mathbf{x}}$ and $\rho_{\mathbf{y}}$, the inequality* (3.60) *holds if*

$$\begin{bmatrix}
\mathrm{He}\{\mathbf{P}(\mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C})\} + (\tau_{\mathbf{x}} + \tau_{\mathbf{y}})\mathbf{I} & \mathbf{P}(\mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}) & (\mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C})^{\top} & \rho_{\mathbf{x}}\mathbf{P} & \rho_{\mathbf{y}}\mathbf{P}\mathcal{B}\mathcal{K} \\
(\mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}})^{\top}\mathbf{P} & -\gamma^2\mathbf{I} & (\mathcal{D}_{\mathbf{Zw}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}})^{\top} & \mathbf{0} & \mathbf{0} \\
\mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} & \mathcal{D}_{\mathbf{Zw}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}} & -\mathbf{I} & \mathbf{0} & \rho_{\mathbf{y}}\mathcal{D}_{\mathbf{Z}}\mathcal{K} \\
\rho_{\mathbf{x}}\mathbf{P} & \mathbf{0} & \mathbf{0} & -\tau_{\mathbf{x}}\mathbf{I} & \mathbf{0} \\
\rho_{\mathbf{y}}\mathcal{K}^{\top}\mathcal{B}^{\top}\mathbf{P} & \mathbf{0} & \rho_{\mathbf{y}}\mathcal{K}^{\top}\mathcal{D}_{\mathbf{Z}}^{\top} & \mathbf{0} & -\tau_{\mathbf{y}}\mathbf{I}
\end{bmatrix} < 0. \tag{3.61}$$

Proof. Define $\varphi_{\mathbf{x}} = \Delta_{\mathbf{x}}(t)\mathbf{X}$, $\varphi_{\mathbf{y}} = \Delta_{\mathbf{y}}(t)\mathbf{X}$, and

$$\mathcal{P} = \begin{bmatrix} \mathrm{He}\{\mathbf{P}(\mathcal{A}+\mathcal{B}\mathcal{K}\mathcal{C})\} & \mathbf{P}(\mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}) & (\mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C})^{\top} \\ (\mathcal{B}_{\mathbf{w}}+\mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}})^{\top}\mathbf{P} & -\gamma^2\mathbf{I} & (\mathcal{D}_{\mathbf{Z}\mathbf{w}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}})^{\top} \\ \mathcal{C}_{\mathbf{Z}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} & \mathcal{D}_{\mathbf{Z}\mathbf{w}}+\mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}} & -\mathbf{I} \end{bmatrix}. \tag{3.62}$$

With the norm-bounded uncertainties in (3.54) and the definitions in (3.58), for arbitrary vector $\chi = \begin{bmatrix} \mathbf{X} \\ \mathbf{w} \\ \psi \end{bmatrix}$, (3.60) implies that

$$\chi^{\top}\mathcal{P}\chi + 2\rho_{\mathbf{x}}\mathbf{X}^{\top}\mathbf{P}\varphi_{\mathbf{x}} + 2\rho_{\mathbf{y}}\mathbf{X}^{\top}\mathbf{P}\mathcal{B}\mathcal{K}\varphi_{\mathbf{y}} + 2\rho_{\mathbf{y}}\psi^{\top}\mathcal{D}_{\mathbf{Z}}\mathcal{K}\varphi_{\mathbf{y}} < 0 \tag{3.63}$$

for all $\varphi_{\mathbf{x}}$ and $\varphi_{\mathbf{y}}$ such that $\varphi_{\mathbf{x}}^{\top}\varphi_{\mathbf{x}} \leq \mathbf{X}^{\top}\mathbf{X}$ and $\varphi_{\mathbf{y}}^{\top}\varphi_{\mathbf{y}} \leq \mathbf{X}^{\top}\mathbf{X}$. According to the $\mathcal{S}$-procedure, this inequality holds if there exist $\tau_{\mathbf{x}} \geq 0$ and $\tau_{\mathbf{y}} \geq 0$ such that

$$\chi^{\top}\mathcal{P}\chi+2\rho_{\mathbf{x}}\mathbf{X}^{\top}\mathbf{P}\varphi_{\mathbf{x}}+2\rho_{\mathbf{y}}\mathbf{X}^{\top}\mathbf{P}\mathcal{B}\mathcal{K}\varphi_{\mathbf{y}}+2\rho_{\mathbf{y}}\psi^{\top}\mathcal{D}_{\mathbf{Z}}\mathcal{K}\varphi_{\mathbf{y}}-\tau_{\mathbf{x}}(\varphi_{\mathbf{x}}^{\top}\varphi_{\mathbf{x}}-\mathbf{X}^{\top}\mathbf{X})-\tau_{\mathbf{y}}(\varphi_{\mathbf{y}}^{\top}\varphi_{\mathbf{y}}-\mathbf{X}^{\top}\mathbf{X}) < 0 \tag{3.64}$$

holds for arbitrary vector $\left[\mathbf{X}^{\top}\ \mathbf{w}^{\top}\ \psi^{\top}\ \varphi_{\mathbf{x}}^{\top}\ \varphi_{\mathbf{y}}^{\top}\right]^{\top}$. This condition is equivalent to (3.61). □

A suboptimal solution to the optimal control problem (3.56) is given by

$$\min_{\mathbf{K},\mathbf{P},\gamma^2} \gamma^2 \text{ s.t. } (3.61) \tag{3.65}$$

3.8.4 PCE-based guaranteed cost control

Let $V(t,\boldsymbol{\eta}) = \mathbf{x}^{\top}(t,\boldsymbol{\eta})\mathbf{Q}\mathbf{x}(t,\boldsymbol{\eta})$ be a Lyapunov function and let $\mathbf{x}(t,\boldsymbol{\eta}) = \hat{\mathbf{x}}(t,\boldsymbol{\eta}) + \tilde{\mathbf{x}}(t,\boldsymbol{\eta})$, where $\hat{\mathbf{x}}(t,\boldsymbol{\eta})$ is the PCE approximation and $\tilde{\mathbf{x}}(t,\boldsymbol{\eta})$ is the PCE truncation error.

The inequality $\mathbf{E}_{\boldsymbol{\eta}}\left\{\|\mathbf{z}(t,\boldsymbol{\eta})\|^2_{L_2[0,\infty)}\right\} \leq \gamma^2\|\mathbf{w}(t)\|^2_{L_2[0,\infty)}$ holds if there exists $\mathbf{Q} > 0$ such that

$$\mathbf{E}_{\boldsymbol{\eta}}\left\{\frac{dV(t,\boldsymbol{\eta})}{dt}\right\} \leq -\mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{z}(t,\boldsymbol{\eta})^{\top}\mathbf{z}(t,\boldsymbol{\eta})\} + \gamma^2\mathbf{w}^{\top}(t)\mathbf{w}(t). \tag{3.66}$$

This equality forms the basis for deriving an upper bound on the closed-loop $\mathcal{H}_\infty$ norm (3.55) for the uncertain system.

Theorem 3. *The inequality* (3.66) *holds if there exists a positive-definite matrix* $\mathbf{Q}$ *such that*

$$\mathbf{E}_{\boldsymbol{\eta}}\{\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}_1(\rho_1,\rho_2,\boldsymbol{\eta})\hat{\mathbf{x}}(t,\boldsymbol{\eta}) + 2\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_1(\boldsymbol{\eta})\mathbf{w}(t) + \mathbf{w}^\top(t)\mathbf{S}_1(\boldsymbol{\eta})\mathbf{w}(t)\} \leq 0 \tag{3.67}$$

holds for positive scalars ρ_1 *and* ρ_2 *satisfying*

$$\mathbf{E}_{\boldsymbol{\eta}}\{\tilde{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\tilde{\mathbf{x}}(t,\boldsymbol{\eta})\} \leq \rho_1\mathbf{E}_{\boldsymbol{\eta}}\{\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\hat{\mathbf{x}}(t,\boldsymbol{\eta})\}, \tag{3.68}$$

$$\mathbf{E}_{\boldsymbol{\eta}}\{\tilde{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}(\boldsymbol{\eta})\tilde{\mathbf{x}}(t,\boldsymbol{\eta})\} \leq \rho_2\mathbf{E}_{\boldsymbol{\eta}}\{\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}(\boldsymbol{\eta})\hat{\mathbf{x}}(t,\boldsymbol{\eta})\}, \tag{3.69}$$

where

$$\begin{aligned}
\boldsymbol{\Pi}_1(\rho_1,\rho_2,\boldsymbol{\eta}) &= 2(1+\rho_1)\boldsymbol{\Pi}(\boldsymbol{\eta}) + \rho_2\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}(\boldsymbol{\eta}),\\
\boldsymbol{\Gamma}_1(\boldsymbol{\eta}) &= \boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta}),\\
\mathbf{S}_1(\boldsymbol{\eta}) &= \boldsymbol{\Gamma}_{\mathbf{w}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta}) + \mathbf{S}(\boldsymbol{\eta}),\\
\boldsymbol{\Pi}(\boldsymbol{\eta}) &= \mathbf{A}_c^\top(\boldsymbol{\eta})\mathbf{Q} + \mathbf{Q}\mathbf{A}_c(\boldsymbol{\eta}) + \mathbf{C}_{\mathbf{z},c}^\top(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},c}(\boldsymbol{\eta}),\\
\boldsymbol{\Gamma}_{\mathbf{x}}(\boldsymbol{\eta}) &= \begin{bmatrix}\mathbf{Q}\\ \mathbf{C}_{\mathbf{z},c}(\boldsymbol{\eta})\end{bmatrix},\ \boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta}) = \begin{bmatrix}\mathbf{B}_{\mathbf{w},c}(\boldsymbol{\eta})\\ \mathbf{D}_{\mathbf{zw},c}(\boldsymbol{\eta})\end{bmatrix},\\
\mathbf{S}(\boldsymbol{\eta}) &= \mathbf{D}_{\mathbf{zw},c}^\top(\boldsymbol{\eta})\mathbf{D}_{\mathbf{zw},c}(\boldsymbol{\eta}) - \gamma^2\mathbf{I}.
\end{aligned} \tag{3.70}$$

Proof. Substituting

$$\frac{dV(t,\boldsymbol{\eta})}{dt} = \left(\mathbf{A}_c(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{B}_{\mathbf{w},c}(\boldsymbol{\eta})\mathbf{w}(t)\right)^\top\mathbf{Q}\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{x}^\top(t,\boldsymbol{\eta})\mathbf{Q}\left(\mathbf{A}_c(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) + \mathbf{B}_{\mathbf{w},c}(\boldsymbol{\eta})\mathbf{w}(t)\right) \tag{3.71}$$

into (3.66) gives

$$\mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{x}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) + 2\mathbf{x}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) + \mathbf{w}^\top(t)\mathbf{S}(\boldsymbol{\eta})\mathbf{w}(t)\} \leq 0. \tag{3.72}$$

Using (3.68) and (3.69), and

$$\begin{aligned}
\mathbf{x}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta}) &= \left(\hat{\mathbf{x}}(t,\boldsymbol{\eta})+\tilde{\mathbf{x}}(t,\boldsymbol{\eta})\right)^\top \boldsymbol{\Pi}(\boldsymbol{\eta})\left(\hat{\mathbf{x}}(t,\boldsymbol{\eta})+\tilde{\mathbf{x}}(t,\boldsymbol{\eta})\right) \\
&\leq 2\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\hat{\mathbf{x}}(t,\boldsymbol{\eta}) + 2\tilde{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Pi}(\boldsymbol{\eta})\tilde{\mathbf{x}}(t,\boldsymbol{\eta}), \\
2\mathbf{x}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) &= 2\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) + 2\tilde{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) \\
&\leq 2\hat{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t) + \tilde{\mathbf{x}}^\top(t,\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}(\boldsymbol{\eta})\tilde{\mathbf{x}}(t,\boldsymbol{\eta}) \\
&\quad + \mathbf{w}^\top(t)\boldsymbol{\Gamma}_{\mathbf{w}}^\top(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\mathbf{w}(t),
\end{aligned} \tag{3.73}$$

the inequality (3.66), or equivalently (3.72), is satisfied if (3.67) holds. □

3.8.5 A synthesis condition

Now that stability and performance conditions have been derived for the closed-loop uncertain system, a synthesis condition can be derived. Let

$$\begin{aligned}
\hat{\mathbf{x}}(t,\boldsymbol{\eta}) &= \sum_{j=0}^{N_p} \hat{\mathbf{x}}_j(t)\Phi_j(\boldsymbol{\eta}) = \boldsymbol{\Lambda}_{\mathbf{x}}^\top(\boldsymbol{\eta})\mathbf{X}(t), \\
\boldsymbol{\Lambda}_{\mathbf{x}}^\top(\boldsymbol{\eta}) &= \begin{bmatrix}\Phi_0(\boldsymbol{\eta})\mathbf{I}_{n_x} & \Phi_1(\boldsymbol{\eta})\mathbf{I}_{n_x} & \cdots & \Phi_{N_p}(\boldsymbol{\eta})\mathbf{I}_{n_x}\end{bmatrix}, \\
\Psi_{N_p}^\top(\boldsymbol{\eta}) &= \begin{bmatrix}\Phi_0(\boldsymbol{\eta}) & \Phi_1(\boldsymbol{\eta}) & \cdots & \Phi_{N_p}(\boldsymbol{\eta})\end{bmatrix}.
\end{aligned} \tag{3.74}$$

The next result provides expressions for the expected value of various sets of matrices needed in further analysis.

Proposition 4. *Let K be a real matrix and $A(\boldsymbol{\eta})$, $B(\boldsymbol{\eta})$, and $C(\boldsymbol{\eta})$ be matrices with polynomial dependence on $\boldsymbol{\eta}$, which can be expressed as $S(\boldsymbol{\eta}) = \sum_{i=0}^{N_{ps}} S_i\Phi_i(\boldsymbol{\eta})$, where S represents A, B, or C, and p_s is the degree of $S(\boldsymbol{\eta})$. All matrices in the below equations have compatible dimensions. The expected value of two sets of matrices can be computed from PCE coefficients:*

(i) $\mathbf{E}_{\boldsymbol{\eta}}\{A(\boldsymbol{\eta})KB(\boldsymbol{\eta})\} = \bar{A}\bar{K}\bar{B}$, *where* $\bar{A} = \begin{bmatrix}A_1 & A_2 & \cdots A_{N_{pa}}\end{bmatrix}$, $\bar{B}^\top = \begin{bmatrix}B_1^\top & B_2^\top & \cdots B_{N_{p_b}}^\top\end{bmatrix}$, *and* $\bar{K} = \mathbf{1}_{N_{pa}\times N_{p_b}} \otimes K$, $\mathbf{1}_{m\times n}$ *represents a* $m \times n$*-matrix whose entries are all 1;*

(ii) $\mathbf{E}_{\boldsymbol{\eta}}\{A(\boldsymbol{\eta})KC(\boldsymbol{\eta})K^{\top}B(\boldsymbol{\eta})\} = \bar{A}\bar{K}_1\bar{C}\bar{K}_2^{\top}\bar{B}$, *where* $\bar{K}_1 = \mathbf{1}_{N_{p_a}\times N_{p_a}} \otimes K$, $\bar{K}_2 = \mathbf{1}_{N_{p_b}\times N_{p_b}} \otimes K$, $T_{i,j} = \mathbf{E}_{\boldsymbol{\eta}}\{C(\boldsymbol{\eta})\Phi_i(\boldsymbol{\eta})\Phi_j(\boldsymbol{\eta})\}$, *and*

$$\bar{C} = \begin{bmatrix} T_{0,0} & \cdots & T_{0,N_{p_b}} \\ \vdots & \ddots & \vdots \\ T_{N_{p_a},0} & \cdots & T_{N_{p_a},N_{p_b}} \end{bmatrix}.$$

Proof. (i)

$$\begin{aligned}\mathbf{E}_{\boldsymbol{\eta}}\{A(\boldsymbol{\eta})KB(\boldsymbol{\eta})\} &= \mathbf{E}_{\boldsymbol{\eta}}\left\{\left[\sum_{i=0}^{N_{p_a}} A_i\Phi_i(\boldsymbol{\eta})\right] K \left[\sum_{j=0}^{N_{p_b}} B_j\Phi_j(\boldsymbol{\eta})\right]\right\} \\ &= \mathbf{E}_{\boldsymbol{\eta}}\left\{\sum_{i=0}^{N_{p_a}}\sum_{j=0}^{N_{p_b}} A_iKB_j\Phi_i(\boldsymbol{\eta})\Phi_j(\boldsymbol{\eta})\right\} = \sum_{i=0}^{N_{p_a}}\sum_{j=0}^{N_{p_b}} A_iKB_j\end{aligned} \tag{3.75}$$

(ii)

$$\begin{aligned}\mathbf{E}_{\boldsymbol{\eta}}\{A(\boldsymbol{\eta})KC(\boldsymbol{\eta})K^{\top}B(\boldsymbol{\eta})\} &= \mathbf{E}_{\boldsymbol{\eta}}\left\{\left[\sum_{i=0}^{N_{p_a}} A_i\Phi_i(\boldsymbol{\eta})\right] KC(\boldsymbol{\eta})K^{\top} \left[\sum_{j=0}^{N_{p_b}} B_j\Phi_j(\boldsymbol{\eta})\right]\right\} \\ &= \mathbf{E}_{\boldsymbol{\eta}}\left\{\sum_{i=0}^{N_{p_a}}\sum_{j=0}^{N_{p_b}} A_iK\left[C(\boldsymbol{\eta})\Phi_i(\boldsymbol{\eta})\Phi_j(\boldsymbol{\eta})\right]KB_j\right\} \\ &= \sum_{i=0}^{N_{p_a}}\sum_{j=0}^{N_{p_b}} A_iK\mathbf{E}_{\boldsymbol{\eta}}\left\{C(\boldsymbol{\eta})\Phi_i(\boldsymbol{\eta})\Phi_j(\boldsymbol{\eta})\right\}KB_j \\ &= \sum_{i=0}^{N_{p_a}}\sum_{j=0}^{N_{p_b}} A_iKT_{i,j}KB_j\end{aligned} \tag{3.76}$$

□

The next set of theoretical results requires a series of definitions:

$$
\begin{aligned}
&\mathcal{A}_1 = \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{A}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\},\\
&\mathcal{B}_{\mathbf{w},1} = \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta})\},\\
&\mathcal{B}_{\mathbf{w},2}^{\top}\mathcal{B}_{\mathbf{w},2} = \mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{B}_{\mathbf{w}}^{\top}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta})\},\\
&\mathcal{S} = \mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{B}_{\mathbf{w}}^{\top}(\boldsymbol{\eta})\mathbf{B}_{(}\boldsymbol{\eta})\},\\
&\mathcal{C}_1 = \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\},\\
&\mathcal{C}_2 = \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}(\boldsymbol{\eta})\},\\
&\tilde{\mathcal{B}}_1^{\top}\tilde{\mathcal{B}}_1 = \mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{B}^{\top}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta})\},\\
&\mathcal{C}_{\mathbf{z},1} = \operatorname{diag}\left(\mathbf{C}_{\mathbf{z}},\cdots,\mathbf{C}_{\mathbf{z}}\right),\ \mathcal{D}_{\mathbf{z},1} = \operatorname{diag}\left(\mathbf{D}_{\mathbf{z}},\cdots,\mathbf{D}_{\mathbf{z}}\right),\\
&\mathcal{D}_{\mathbf{zw},1} = \operatorname{diag}\left(\mathbf{D}_{\mathbf{zw}},\cdots,\mathbf{D}_{\mathbf{zw}}\right),\\
&\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta}) = \sum_{i=0}^{N_{p_b}}\hat{\mathbf{B}}_i\Phi_i(\boldsymbol{\eta}),\ \mathbf{C}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta}) = \sum_{i=0}^{N_{pc}}\hat{\mathbf{C}}_i^{\top}\Phi_i(\boldsymbol{\eta}),\\
&\mathcal{B}_2 = \left[\hat{\mathbf{B}}_0\ \ \cdots\ \ \hat{\mathbf{B}}_{N_{p_b}}\right],\ \mathcal{C}_3 = \left[\hat{\mathbf{C}}_0\ \ \cdots\ \ \hat{\mathbf{C}}_{N_{pc}}\right],\\
&\mathcal{K}_1 = \mathbf{1}_{N_{p_b}\times N_{pc}}\otimes\mathbf{K},\ \mathcal{K}_2 = \mathbf{I}_{N_p}\otimes\mathbf{K},\ \mathcal{K}_3 = \mathbf{I}_{N_{pc}}\otimes\mathbf{K},\\
&\mathcal{M}_1^{\top}\mathcal{M}_1 = \mathbf{1}_{N_{pc}\times N_{pc}}\otimes(\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}),\ \mathcal{M}_2^{\top}\mathcal{M}_3 = \mathbf{1}_{N_{pc}\times N_{pw}}\otimes(\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}})
\end{aligned}
\tag{3.77}
$$

Proposition 4 and (3.70) imply that we have

$$
\begin{aligned}
&\mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\boldsymbol{\Pi}_1(\rho_1,\rho_2,\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\}\\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{2(1+\rho_1)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{A}_{\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{Q}+\mathbf{Q}\mathbf{A}_{\mathrm{c}}(\boldsymbol{\eta})+\mathbf{C}_{\mathbf{z},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta})\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right.\\
&\quad\left.+\rho_2\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{Q}^2+\mathbf{C}_{\mathbf{z},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta})\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right\}\\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{2(1+\rho_1)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{A}_{\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{Q}+\mathbf{Q}\mathbf{A}_{\mathrm{c}}(\boldsymbol{\eta})\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right.\\
&\quad\left.+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z},\mathrm{c}}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})+\rho_2\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{Q}^2\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right\}\\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{2(1+\rho_1)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{A}^{\top}(\boldsymbol{\eta})\mathbf{Q}+\mathbf{Q}\mathbf{A}(\boldsymbol{\eta})\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right.\\
&\quad+2(1+\rho_1)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{B}^{\top}(\boldsymbol{\eta})\mathbf{Q}+\mathbf{Q}\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{C}(\boldsymbol{\eta})\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\\
&\quad+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{C}_{\mathbf{z}}\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\\
&\quad+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{C}(\boldsymbol{\eta})+\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{C}_{\mathbf{z}}\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\\
&\quad\left.+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{C}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})+\rho_2\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{Q}^2\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right\}\\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{2(1+\rho_1)\left(\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{A}^{\top}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\mathcal{Q}+\mathcal{Q}\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{A}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right)\right.\\
&\quad+2(1+\rho_1)\left(\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{B}^{\top}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\mathcal{Q}+\mathcal{Q}\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{C}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right)\\
&\quad+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{C}_{\mathbf{z}}\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\\
&\quad+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{C}(\boldsymbol{\eta})+\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{C}_{\mathbf{z}}\right)\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\\
&\quad\left.+(2+2\rho_1+\rho_2)\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{C}(\boldsymbol{\eta})\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})+\rho_2\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{Q}^2\boldsymbol{\Lambda}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\right\}\\
&= 2(1+\rho_1)\left(\mathcal{A}_1^{\top}\mathcal{Q}+\mathcal{Q}\mathcal{A}_1+\mathcal{C}_3\mathcal{K}_1^{\top}\mathcal{B}_2^{\top}\mathcal{Q}+\mathcal{Q}\mathcal{B}_2\mathcal{K}_1\mathcal{C}_3^{\top}\right)+\rho_2\mathcal{Q}^2\\
&\quad+(2+2\rho_1+\rho_2)\left(\mathcal{C}_{\mathbf{z},1}^{\top}\mathcal{C}_{\mathbf{z},1}+\mathcal{C}_{\mathbf{z},1}^{\top}\mathcal{D}_{\mathbf{z},1}\mathcal{K}_2\mathcal{C}_1+\mathcal{C}_1^{\top}\mathcal{K}_2^{\top}\mathcal{D}_{\mathbf{z},1}^{\top}\mathcal{C}_{\mathbf{z},1}^{\top}+\mathcal{C}_3\mathcal{K}_3^{\top}\mathcal{M}_1^{\top}\mathcal{M}_1\mathcal{K}_3\mathcal{C}_3^{\top}\right)
\end{aligned}
\tag{3.78}
$$

$$
\begin{aligned}
&\mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{x}}^{\top}(\boldsymbol{\eta})\boldsymbol{\Gamma}_{\mathbf{w}}(\boldsymbol{\eta})\} \\
&= \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{Q}\mathbf{B}_{\mathbf{w},\mathrm{c}}(\boldsymbol{\eta}) + \mathbf{C}_{\mathbf{z},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{D}_{\mathbf{wz},\mathrm{c}}(\boldsymbol{\eta})\right)\} \\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\left(\mathbf{Q}\mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta}) + \mathbf{Q}\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{D}_{\mathbf{w}} + \mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}} + \mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}\right.\right. \\
&\qquad \left.\left. + \mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}} + \mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}\right)\right\} \\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{\mathcal{Q}\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta}) + \mathcal{Q}\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{D}_{\mathbf{w}} + \boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}}\right. \\
&\qquad + \boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}} + \boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}} \\
&\qquad \left. + \boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\mathbf{C}^{\top}(\boldsymbol{\eta})\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}\right\} \\
&= \mathcal{Q}\mathcal{B}_{\mathbf{w},1} + \mathcal{Q}\mathcal{B}_{\mathbf{w},1}\mathbf{K}\mathbf{D}_{\mathbf{w}}^{\top} + \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\}\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}} + \mathbf{E}_{\boldsymbol{\eta}}\{\boldsymbol{\Lambda}_{\mathbf{x}}(\boldsymbol{\eta})\}\mathbf{C}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}} \\
&\qquad + \mathcal{C}_2^{\top}\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{zw}} + \mathcal{C}_2^{\top}\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}
\end{aligned}
\tag{3.79}
$$

$$
\begin{aligned}
\mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{S}_1(\boldsymbol{\eta})\} &= \mathbf{E}_{\boldsymbol{\eta}}\{\mathbf{B}_{\mathbf{w},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w},\mathrm{c}}(\boldsymbol{\eta}) + 2\mathbf{D}_{\mathbf{zw},\mathrm{c}}^{\top}(\boldsymbol{\eta})\mathbf{D}_{\mathbf{zw},\mathrm{c}}(\boldsymbol{\eta}) - \gamma^2\mathbf{I}\} \\
&= \mathbf{E}_{\boldsymbol{\eta}}\left\{\mathbf{B}_{\mathbf{w}}^{\top}(\boldsymbol{\eta})\mathbf{B}_{\mathbf{w}}(\boldsymbol{\eta}) + \mathrm{He}\{\mathbf{B}_{\mathbf{w}}^{\top}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{D}_{\mathbf{w}}\} + \mathbf{D}_{\mathbf{w}}^{\top}\mathbf{K}^{\top}\mathbf{B}^{\top}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta})\mathbf{K}\mathbf{D}_{\mathbf{w}}\right. \\
&\qquad \left. + 2\mathbf{D}_{\mathbf{zw}}^{\top}\mathbf{D}_{\mathbf{zw}} + 2\mathrm{He}\{\mathbf{D}_{\mathbf{zw}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}\} + 2\mathbf{D}_{\mathbf{w}}^{\top}\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}} - \gamma^2\mathbf{I}\right\} \\
&= \mathcal{B}_{\mathbf{w},2}^{\top}\mathcal{B}_{\mathbf{w},2} + \mathrm{He}\{\mathcal{S}\mathbf{K}\mathbf{D}_{\mathbf{w}}\} + \mathbf{D}_{\mathbf{w}}^{\top}\mathbf{K}^{\top}\tilde{\mathcal{B}}_1^{\top}\tilde{\mathcal{B}}_1\mathbf{K}\mathbf{D}_{\mathbf{w}} \\
&\qquad + 2\mathbf{D}_{\mathbf{zw}}^{\top}\mathbf{D}_{\mathbf{zw}} + 2\mathrm{He}\{\mathbf{D}_{\mathbf{zw}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}}\} + 2\mathbf{D}_{\mathbf{w}}^{\top}\mathbf{K}^{\top}\mathbf{D}_{\mathbf{z}}^{\top}\mathbf{D}_{\mathbf{z}}\mathbf{K}\mathbf{D}_{\mathbf{w}} - \gamma^2\mathbf{I}
\end{aligned}
\tag{3.80}
$$

This derivation can be easily extended to the parameter-dependent Lyapunov function: $V(t,\boldsymbol{\eta}) = \mathbf{x}^{\top}(t,\boldsymbol{\eta})\mathbf{Q}(\boldsymbol{\eta})\mathbf{x}(t,\boldsymbol{\eta})$ with $\mathbf{Q}(\boldsymbol{\eta}) = \sum_{i=0}^{q}\mathbf{Q}_i\Phi_i^2(\boldsymbol{\eta})$, $\mathbf{Q}_i > 0$.

This guaranteed cost formulation avoids using Galerkin projection, which avoids the limitation of using Galerkin projection for long-term uncertainty propagation.

3.8.6 PCE-based control with decaying bounds on uncertainty

As the system states converge to zero, the truncated PCEs better approximate the original states, thus the truncation errors also converge to zero. Therefore, the above analysis that assumes that the truncation error is bounded by a constant can be conservative when the closed-loop system drives the states to a steady state. The decaying trunctation errors can be addressed by assuming that the robustifying parameters decay over time, i.e., $\dot{\rho}_{\mathbf{x}} = \alpha_{\mathbf{x}}\rho_{\mathbf{x}}$ and $\dot{\rho}_{\mathbf{y}} = \alpha_{\mathbf{y}}\rho_{\mathbf{y}}$ with $\alpha_{\mathbf{x}} < 0$ and $\alpha_{\mathbf{y}} < 0$.

Consider the control law $\mathbf{u} = (\mathbf{K} + \rho_{\mathbf{x}}\mathbf{K}_{\mathbf{x}} + \rho_{\mathbf{y}}\mathbf{K}_{\mathbf{y}})\mathbf{y}$ whose PCE approximation is $\mathbf{U} = (\mathcal{K} + \rho_{\mathbf{x}}\mathcal{K}_{\mathbf{x}} + \rho_{\mathbf{y}}\mathcal{K}_{\mathbf{y}})\mathbf{Y}$ where $\rho_{\mathbf{x}}(0)$, $\alpha_{\mathbf{x}}$, $\rho_{\mathbf{y}}(0)$, and $\alpha_{\mathbf{y}}$ are tuning parameters. Given these tuning parameters, $\rho_{\mathbf{x}}(t)$ and $\rho_{\mathbf{y}}(t)$ are available to the controller.

The closed-loop expanded system is

$$\begin{aligned}
\dot{\mathbf{X}} &= \mathcal{A}_{\mathrm{c}}\mathbf{X} + \mathcal{B}_{\mathbf{w},\mathrm{c}}\mathbf{w}, \\
\dot{\rho}_{\mathbf{x}} &= \alpha_{\mathbf{x}}\rho_{\mathbf{x}}, \\
\dot{\rho}_{\mathbf{y}} &= \alpha_{\mathbf{y}}\rho_{\mathbf{y}}, \\
\mathbf{Z} &= \mathcal{C}_{\mathbf{Z},\mathrm{c}}\mathbf{X} + \mathcal{D}_{\mathbf{Z}\mathbf{w},\mathrm{c}}\mathbf{w},
\end{aligned} \tag{3.81}$$

with definitions

$$\begin{aligned}
\mathcal{A}_{\mathrm{c}} &= \mathcal{A} + \mathcal{B}\mathcal{K}\mathcal{C} + \rho_{\mathbf{x}}\mathcal{B}\mathcal{K}_{\mathbf{x}}\mathcal{C} + \rho_{\mathbf{y}}\mathcal{B}\mathcal{K}_{\mathbf{y}}\mathcal{C} + \rho_{\mathbf{x}}\Delta_{\mathbf{x}}(t) + \rho_{\mathbf{y}}\mathcal{B}\mathcal{K}\Delta_{\mathbf{y}} + \rho_{\mathbf{x}}\rho_{\mathbf{y}}\mathcal{B}\mathcal{K}_{\mathbf{x}}\Delta_{\mathbf{y}} + \rho_{\mathbf{y}}^2\mathcal{B}\mathcal{K}_{\mathbf{y}}\Delta_{\mathbf{y}}, \\
\mathcal{B}_{\mathbf{w},\mathrm{c}} &= \mathcal{B}_{\mathbf{w}} + \mathcal{B}\mathcal{K}\mathcal{D}_{\mathbf{w}}, \\
\mathcal{C}_{\mathbf{Z},\mathrm{c}} &= \mathcal{C}_{\mathbf{Z}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{C} + \rho_{\mathbf{y}}\mathcal{D}_{\mathbf{Z}}\mathcal{K}\Delta_{\mathbf{y}}, \\
\mathcal{D}_{\mathbf{Z}\mathbf{w},\mathrm{c}} &= \mathcal{D}_{\mathbf{Z}\mathbf{w}} + \mathcal{D}_{\mathbf{Z}}\mathcal{K}\mathcal{D}_{\mathbf{w}}.
\end{aligned} \tag{3.82}$$

The theoretical analysis for constant robustifying parameters can be extended to this more complicated system.

3.9 Case Study

Consider a linear system, the system matrices of which are

$$\begin{gathered}
A = \begin{bmatrix} -5.01 + 5\xi & 0 \\ 0 & -1 \end{bmatrix}, \; B = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \; B_{\mathbf{w}} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\
D_{\mathbf{z}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \; D_{\mathbf{z}\mathbf{w}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix} \\
C_{\mathbf{z}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \; D_{\mathbf{w}} = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix},
\end{gathered} \tag{3.83}$$

for which ξ is uniformly distributed between -1 and 1. As shown in Fig. 3-7, the $\mathcal{H}_\infty$ norm peaks at $\xi = 0.611$, because one of two eigenvalues of the closed-loop system matrix A is 0.0043.

PCE-based $\mathcal{H}_\infty$-norm minimization discussed in the previous section addresses this sensitivity to the value of ξ. As shown in Fig. 3-8, the maximum value of $\mathcal{H}_\infty$-norm from SOF controller optimized by PCE-based method is much smaller than that by nominal optimization. Furthermore, as ρ_x increases, the peak value of $\mathcal{H}_\infty$ norm decreases near $\xi = 1$; however, for other values of ξ, the $\mathcal{H}_\infty$ norm does increase as ρ_x increases.

3.10 Conclusions

Polynomial chaos based $\mathcal{H}_2$ static and dynamic output-feedback control synthesis methods are presented for systems subject to time-invariant probabilistic parametric uncertainties and white noises. The effect of polynomial chaos expansion truncation errors is captured by a time-varying norm-bounded uncertainty, and explicitly taken into account by adopting a guaranteed cost control approach. This strategy enforces the closed-loop stability, which may not be achieved by existing PCE-based controls due to neglecting these truncation errors. In contrast to nominal control and worst-case robust control, the proposed PCE-based guaranteed cost synthesis allows a trade-off between the worst-case performance bound and the averaged performance by tuning the introduced robustifying parameter, as illustrated by the simulation results.

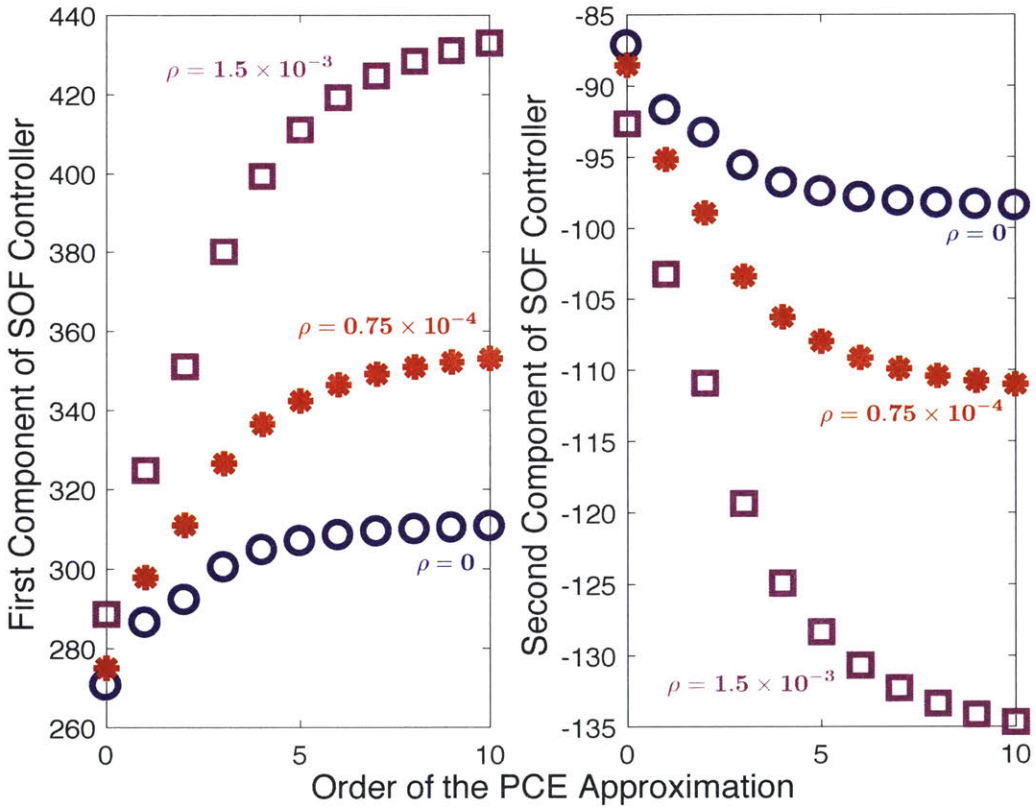

Figure 3-1: The numerical example in Section 3.7.1: the dependency of both components of the SOF controller on the order of the PCE approximation. The control gain converges for a 10th-order expansion.

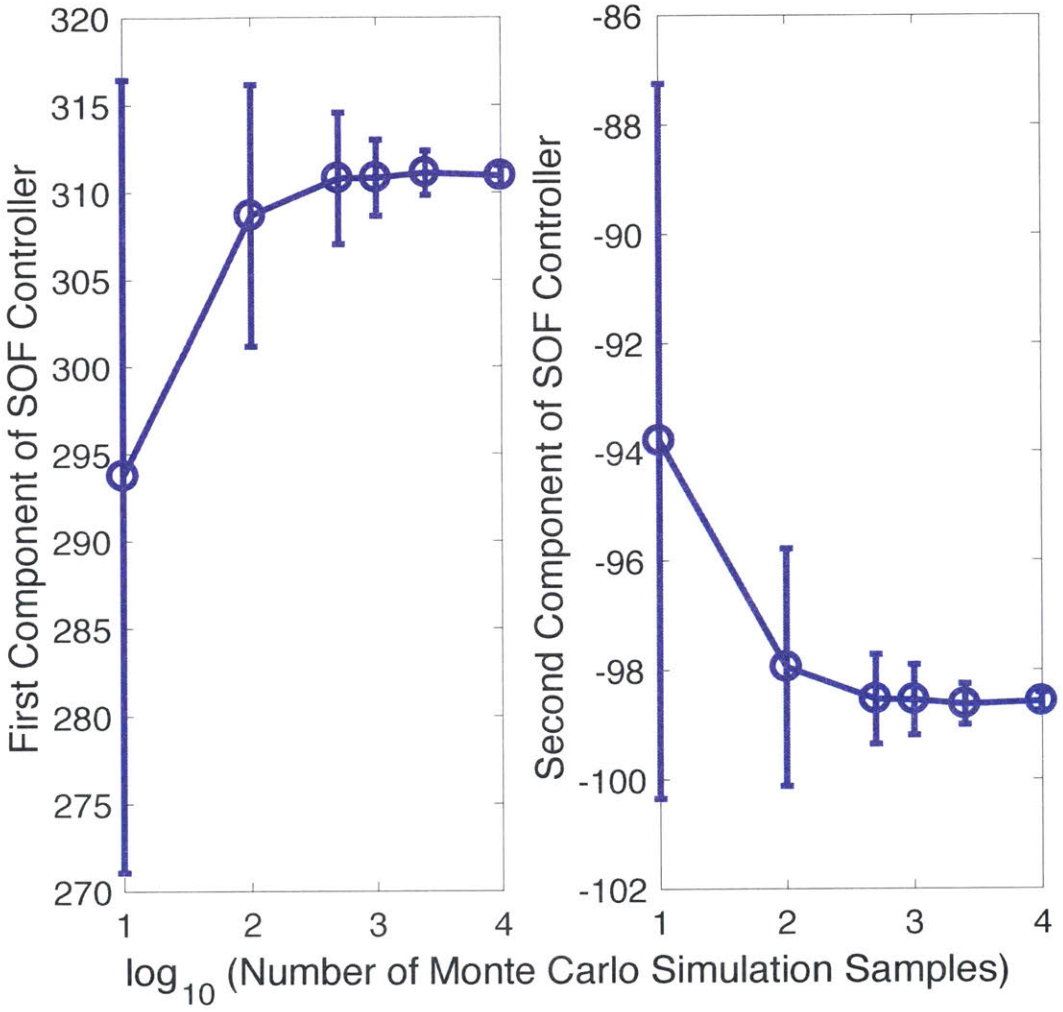

Figure 3-2: The numerical example in Section 3.7.1: the dependence of both components of the SOF controller on the number of Monte-Carlo samples (circles: the average of 100 trials with different samples of ξ; error bars: the standard deviation of 100 trials). The control gain converges when using 10^4 samples.

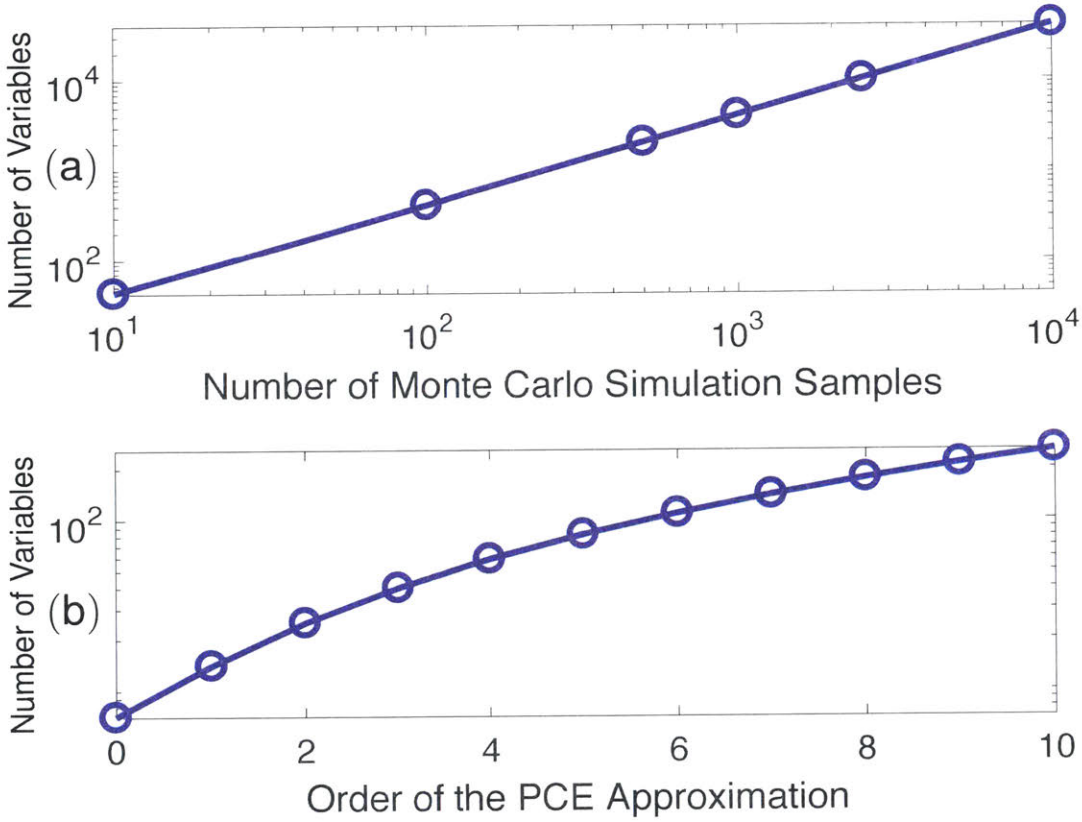

Figure 3-3: The numerical example in Section 3.7.1: comparison of the number of decision variables for the PCE-based and the Monte-Carlo-simulation-based control synthesis problems.

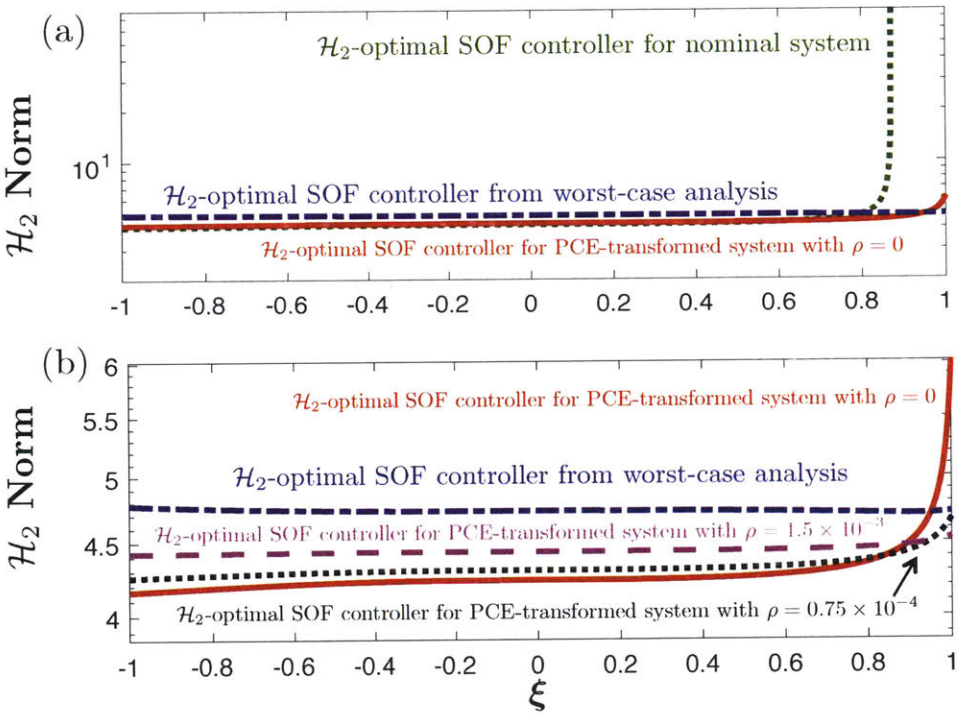

Figure 3-4: The numerical example in Section 3.7.1: comparison of distributions of $\mathcal{H}_2$ norms generated by different controls; (a) the standard nominal control, the worst-case robust control, and the PCE-based controls using a 10th-order PCE without accounting for PCE truncation errors; (b) the worst-case robust control and the PCE-based controls using a 10th-order PCE and different ρ's.

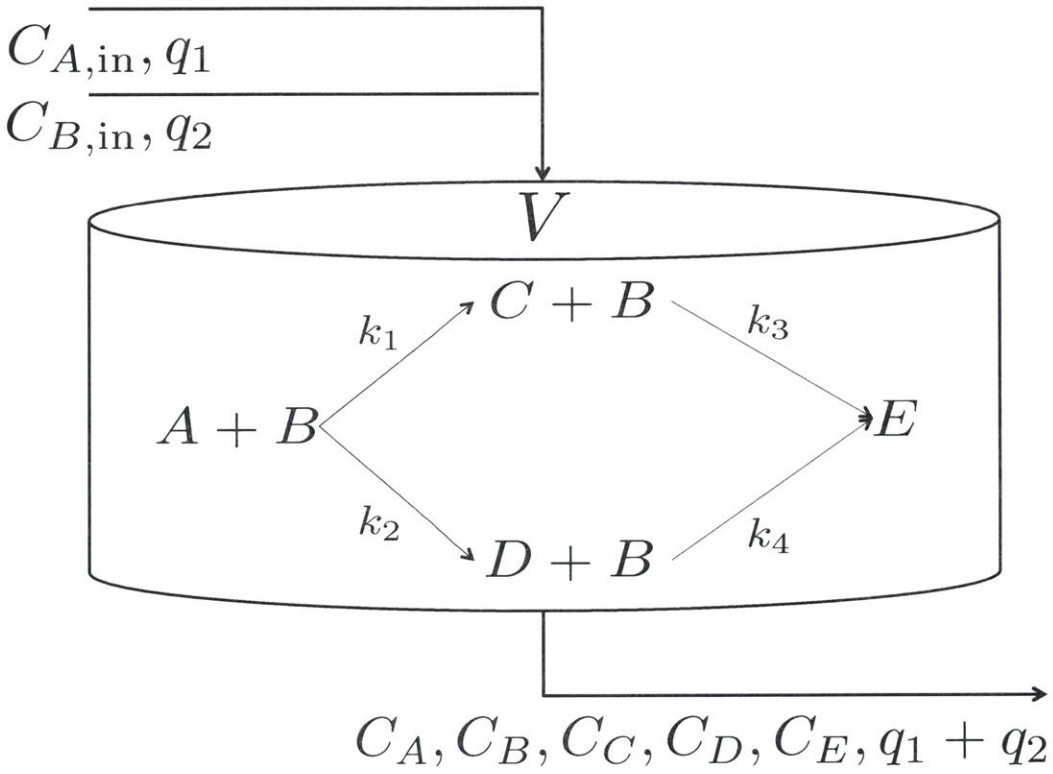

Figure 3-5: The CSTR studied in Section 3.7.2: the objective is to convert the reagent A to the desired pharmaceutical D.

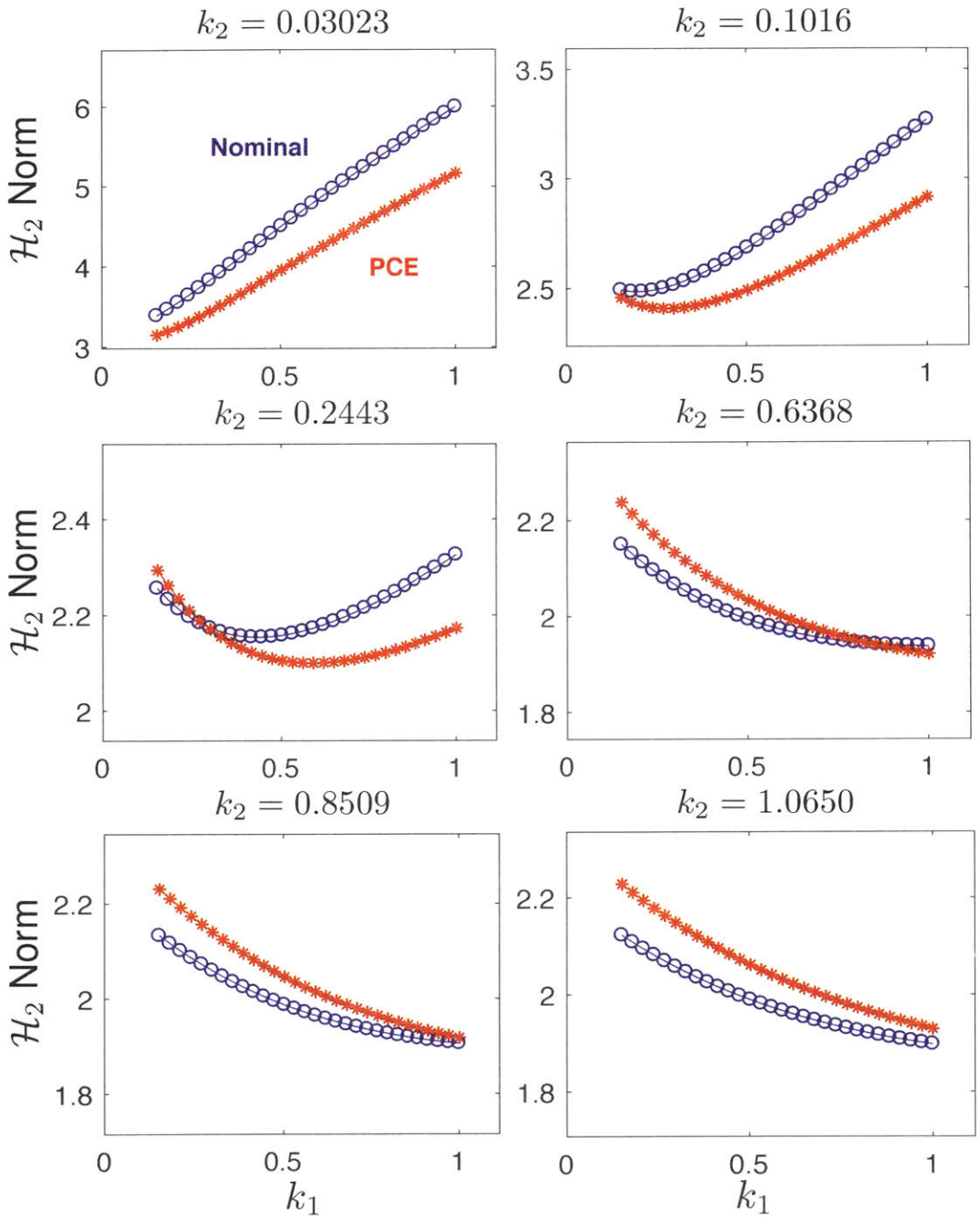

Figure 3-6: The CSTR studied in Section 3.7.2: comparison of distributions of $\mathcal{H}_2$ norm under the nominal control and the PCE-based control with a 2rd-order PCE.

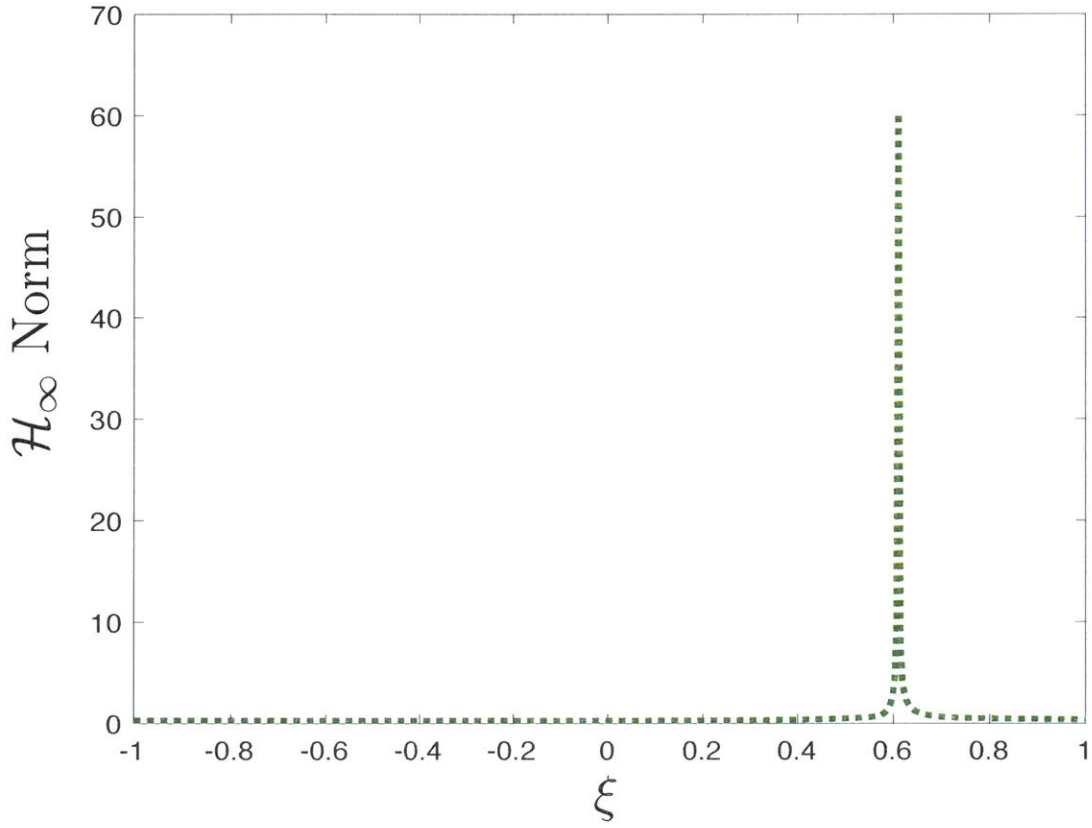

Figure 3-7: The $\mathcal{H}_\infty$-norm distribution of the linear system described by Eqn. (3.83) with SOF controller derived from nominal optimization for $\xi = 0$.

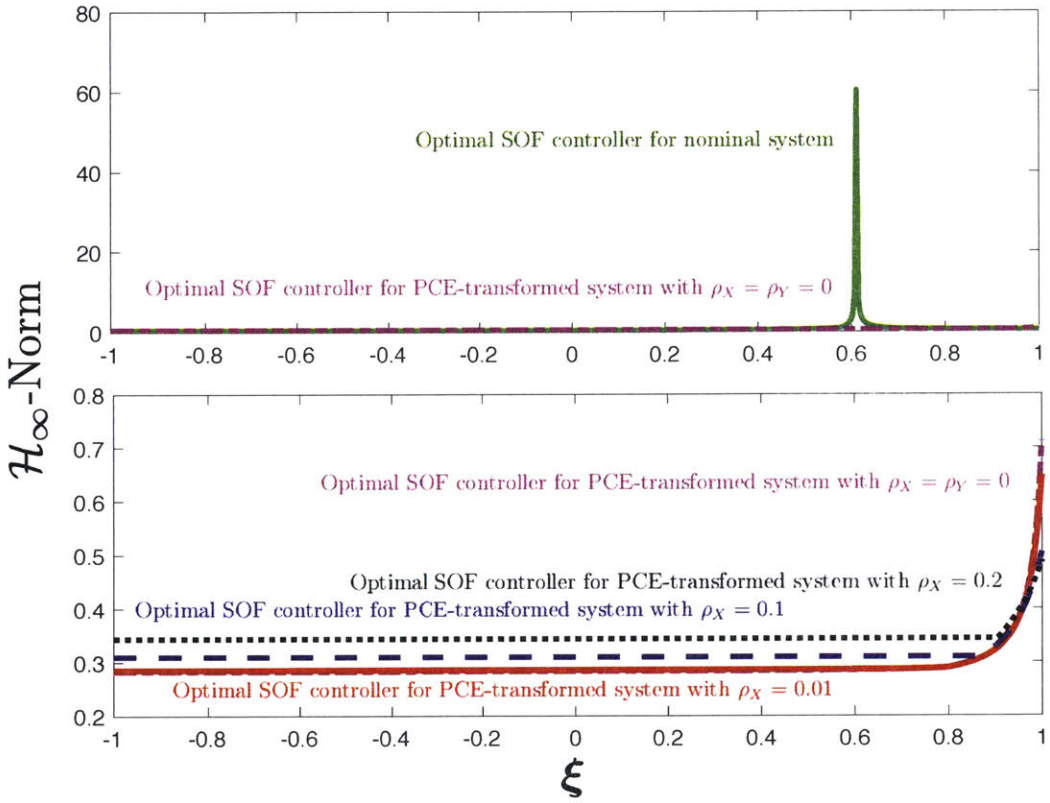

Figure 3-8: Comparison of different $\mathcal{H}_\infty$-norm distributions of the linear system described by Eqn. (3.83) with SOF controllers derived from nominal optimization for $\xi = 0$ and PCE-based $\mathcal{H}_\infty$-norm minimization with various ρ_X's.

Chapter 4

Stochastic Model Predictive Control

4.1 Introduction

Chapter 3 derived a rigorous theoretical framework for the optimal control of linear time-invariant systems with time-invariant parameters described by probability distribution functions. These optimal control formulations assume that the manipulated variables are not constrained in magnitude or rate. Chemical process systems typically operate in the presence of constraints on the manipulated variables, and the purpose of this chapter is to derive an optimal control formulation that explictly takes both time-invariant probabilistic parameters and actuator magnitude and rate constraints into account.

This chapter builds a PCE-based model predictive control (MPC) formulation that combines the fast MPC formulation of Joel A. Paulson with the optimal control formulation from Chapter 3. This integated algorithm combines the strengths of both optimal control formulations to have the same low online computational cost of Joel Paulson's algorithm while having the closed-loop optimality of the algorithms in Chapter 3 when the constraints are not active.

A statement of the online optimal control problem is followed by its derivation and application in a case study. To simplify the presentation, the process states are assumed to be measured.

4.2 Background

Consider the linear time-invariant system

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu. \tag{4.1}$$

The nominal optimization objective J for deriving the control inputs $\Delta\mathbf{u}$ for this system is

$$\min_{\Delta\mathbf{u}(k)} J = \min_{\Delta\mathbf{u}(k)} \sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \hat{\mathbf{x}}(k+i)||_2^2 + r \sum_{i=0}^{c-1} \|\Delta\mathbf{u}(k+i)\|^2, \tag{4.2}$$

$$\begin{aligned} \text{s.t. } \mathbf{x}_{\min} \leq \hat{\mathbf{x}} \leq \mathbf{x}_{\max} \\ \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \\ \Delta\mathbf{u}_{\min} \leq \Delta\mathbf{u} \leq \Delta\mathbf{u}_{\max}. \end{aligned} \tag{4.3}$$

for which

1. $\Delta\mathbf{u}(k) = [\Delta u(k), \Delta u(k+1), \Delta u(k+2), \ldots, \Delta u(k+c-1)]^\top$ is the vector of control inputs of length c that minimizes the objective function J in (4.2);

2. p is the prediction horizon, which is the number of predicted states taken into consideration when computing the objective function J;

3. $\mathbf{x}^{\text{s.p.}}(k+i) = [x_1^{\text{s.p.}}(k+i), x_2^{\text{s.p.}}(k+i), \ldots, x_{n_x}^{\text{s.p.}}(k+i)]^\top$ is the set point for the measured state vector $\mathbf{x}$, which is the desired value for the measured state vector to reach at $k+i$ instant;

4. k is the sampling instant $\in \mathbb{Z}$, is related to real time t by $t = kT_{\text{s}}$, for which T_{s} denotes the sampling time;

5. r is the relative importance of the cost caused by deviation from the state set point to that caused by changes in control input;

6. c is the control horizon, which is the number of changes in control input taken into consideration when computing the objective function J;

7. $\mathbf{x}_{\min}$ and $\mathbf{x}_{\max}$ are the lower and upper bounds, respectively, for the predicted state vector $\hat{\mathbf{x}}$;

8. $\mathbf{u}_{\min}$ and $\mathbf{u}_{\max}$ are the lower and upper bounds, respectively, for the control input;

9. $\Delta\mathbf{u}_{\min}$ and $\Delta\mathbf{u}_{\max}$ are the lower and upper bounds, respectively, for the change in control input.

As in Chapter 3, the optimal control problem that takes time-invariant probabilistic parameters into account is the same as in except that the optimal control objective is replaced by its expected value with respect to the probabilistic parameters.

A central issue that should be addressed to solve Problem 4.2 or its expected value extension is to predict the future state vector given some control input. This chapter uses the dynamic matrix control (DMC) formulation for making this prediction, which is described in the next section.

4.3 Background on Dynamic Matrix Control

DMC is a model predictive control formulation for linear time-invariant systems in which the model is formulated in terms of either a finite impulse response or finite step response. The steps in the derivation of DMC are reviewed here, to provide background for the formulation of a PCE-based extension to MPC presented later in this chapter.

4.3.1 Finite impulse response

Consider an impulse input,

$$\mathbf{u} = [1, 0, 0, \ldots, 0]^\top,$$

i.e. $u(0) = 1$ and $u(j) = 0$ for $j > 0$. The values of the process output for such an input for a finite number of future time instances is called the finite impulse response,

with each value represented by $h_i(j)$ where j is the time index and i is the index for the state variable. These values can be collected into a matrix that lists the current and future values of all states for all future time as

$$\begin{bmatrix} 0 & h_1(1) & \dots & h_1(n) & 0,0,\dots \\ 0 & h_2(1) & \dots & h_2(n) & 0,0,\dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & h_{n_x}(1) & \dots & h_{n_x}(n) & 0,0,\dots \end{bmatrix}^\top,$$

for which n_x denotes the number of system states. The finite impulse response matrix $\mathbf{h}$ is

$$\begin{bmatrix} h_1(1) & \dots & h_1(n) \\ h_2(1) & \dots & h_2(n) \\ \vdots & \vdots & \vdots \\ h_{n_x}(1) & \dots & h_{n_x}(n) \end{bmatrix}^\top,$$

for which two assumptions are made:

(1) the system does not immediately react to the impulse input, i.e. $\mathbf{h}(0) = \mathbf{0}$;

(2) the system decays to its original state after n instants, i.e. $\mathbf{h}(n+1) = \mathbf{h}(n+2) = \dots = 0$ (Figure 4-1).

To relate the finite impulse response of a system to the response of the system to any design input, the vector of manipulated variables at time t

$$\mathbf{u}(t) = [u(0), u(1), u(2), u(3), \dots]^\top$$

can be re-written as

$$\mathbf{u}(t) = [1,0,0,\dots]^\top u(0) + [0,1,0,\dots]^\top u(1) + \dots. \tag{4.4}$$

Therefore, the general expression for the system states

$$\mathbf{x} = \begin{bmatrix} x_1(0) & x_1(1) & x_1(2) & \dots & x_1(n) \\ x_2(0) & x_2(1) & x_2(2) & \dots & x_2(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n_x}(0) & x_{n_x}(1) & x_{n_x}(2) & \dots & x_{n_x}(n) \end{bmatrix}^\top$$

based on a given design input is

$$\begin{bmatrix} 0 & h_1(1) & \dots & h_1(n) & 0,0,\dots \\ 0 & h_2(1) & \dots & h_2(n) & 0,0,\dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & h_{n_x}(1) & \dots & h_{n_x}(n) & 0,0,\dots \end{bmatrix}^\top u(0)$$

$$+ \begin{bmatrix} 0 & 0 & h_1(1) & \dots & h_1(n) & 0,\dots \\ 0 & 0 & h_2(1) & \dots & h_2(n) & 0,\dots \\ \vdots & & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & h_{n_x}(1) & \dots & h_{n_x}(n) & 0,\dots \end{bmatrix}^\top u(1)$$

$$+ \begin{bmatrix} 0 & 0 & 0 & h_1(1) & \dots & h_1(n) & 0,0,\dots \\ 0 & 0 & 0 & h_2(1) & \dots & h_2(n) & 0,0,\dots \\ \vdots & & & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & h_{n_x}(1) & \dots & h_{n_x}(n) & 0,0,\dots \end{bmatrix}^\top u(2) + \dots$$

$$= \begin{bmatrix} 0 & h_1(1)u(0) & h_1(2)u(0) + h_1(1)u(1) & \dots \\ 0 & h_2(1)u(0) & h_2(2)u(0) + h_2(1)u(1) & \dots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & h_{n_x}(1)u(0) & h_{n_x}(2)u(0) + h_{n_x}(1)u(1) & \dots \end{bmatrix}^\top .$$

In other words,

$$x_j(k) = \sum_{i=1}^{n} h_j(i)u(k-i).$$

4.3.2 Finite step response

For a step design input,

$$\mathbf{u} = [1, 1, \ldots]^\top, \tag{4.5}$$

the current and future values of the step response cooefficients can be collected into the matrix

$$\begin{bmatrix} 0 & s_1(1) & \ldots & s_1(n) & s_1(n) & \ldots \\ 0 & s_2(1) & \ldots & s_2(n) & s_2(n) & \ldots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & s_{n_x}(1) & \ldots & s_{n_x}(n) & s_{n_x}(n) & \ldots \end{bmatrix}^\top,$$

for which two assumptions are made:

(1) the system does not immediately react to the step input, i.e. $\mathbf{s}(0) = \mathbf{0}$;

(2) the system reaches steady state after n instants, i.e. $\mathbf{s}(n+1) = \mathbf{s}(n+2) = \cdots = \mathbf{s}(\infty)$ (Figure 4-2).

Since (4.5) can be rewritten as (4.4) with

$$u(0) = u(1) = u(2) = \cdots = 1,$$

the matrix of finite step responses is related to the matrix of finite impulse responses by

$$\begin{bmatrix} 0 & h_1(1) & h_1(2)+h_1(1) & \dots \\ 0 & h_2(1) & h_2(2)+h_2(1) & \dots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & h_{n_x}(1) & h_{n_x}(2)+h_{n_x}(1) & \dots \end{bmatrix}^\top$$

$$\Rightarrow \boxed{\mathbf{s}(k) = \sum_{i=1}^{k} \mathbf{h}(i)} \tag{4.6}$$

$$\boxed{\mathbf{h}(k) = \mathbf{s}(k) - \mathbf{s}(k-1)} \tag{4.7}$$

On the other hand, any design input

$$\mathbf{u} = [u(0), u(1), u(2), \dots]^\top$$

can be re-written in terms of step inputs as

$$\begin{aligned} \mathbf{u} =& [1,1,1,1,\dots]^\top u(0) + [0,1,1,1,\dots]^\top [u(1)-u(0)] + [0,0,1,1,\dots]^\top [u(2)-u(1)] + \cdots \\ =& [1,1,1,1,\dots]^\top u(0) + [0,1,1,1,\dots]^\top \Delta u(1) + [0,0,1,1,\dots]^\top \Delta u(2) + \cdots \end{aligned}$$

for which

$$\Delta u(k) = \begin{cases} u(0), \text{ for } k = 0 \\ u(k) - u(k-1), \text{ for } k \geq 1 \end{cases}$$

Consequently, the resulting vectors of current and future values of the outputs can

be written in the form of a matrix as

$$\begin{bmatrix} 0 & s_1(1) & \cdots & s_1(n) & s_1(n) & \cdots \\ 0 & s_2(1) & \cdots & s_2(n) & s_2(n) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & s_{n_x}(1) & \cdots & s_{n_x}(n) & s_{n_x}(n) & \cdots \end{bmatrix}^\top u(0) +$$

$$\begin{bmatrix} 0 & 0 & s_1(1) & \cdots & s_1(n) & s_1(n) & \cdots \\ 0 & 0 & s_2(1) & \cdots & s_2(n) & s_2(n) & \cdots \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & s_{n_x}(1) & \cdots & s_{n_x}(n) & s_{n_x}(n) & \cdots \end{bmatrix}^\top \Delta u(1) +$$

$$\begin{bmatrix} 0 & 0 & 0 & s_1(1) & \cdots & s_1(n) & s_1(n) & \cdots \\ 0 & 0 & 0 & s_2(1) & \cdots & s_2(n) & s_2(n) & \cdots \\ \vdots & & & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & 0 & s_{n_x}(1) & \cdots & s_{n_x}(n) & s_{n_x}(n) & \cdots \end{bmatrix}^\top \Delta u(2) + \cdots$$

Therefore,

$$x_j(k) = \sum_{i=1}^{\infty} s_j(i)\Delta u(k-i) = s_j(n)u(k-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k-i).$$

4.3.3 State prediction with finite impulse and step responses

The predicted future states have two contributions: the free response with $\Delta u(k) = \Delta u(k+1) = \cdots = 0$ and the forced response, as seen in the expressions for the current

and future values for the states:

$$\hat{x}_j(k|k) = \underbrace{s_j(n)u(k-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k-i)}_{\text{free response}}$$

$$\begin{aligned}\hat{x}_j(k+1|k) &= s_j(n)u(k+1-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k+1-i) \\ &= \underbrace{s_j(n)u(k+1-n) + \sum_{i=2}^{n-1} s_j(i)\Delta u(k+1-i)}_{\text{free response}} + \underbrace{s_j(1)\Delta u(k)}_{\text{forced response}}\end{aligned}$$

$$\begin{aligned}\hat{x}_j(k+2|k) &= s_j(n)u(k+2-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k+2-i) \\ &= \underbrace{s_j(n)u(k+2-n) + \sum_{i=3}^{n-1} s_j(i)\Delta u(k+2-i)}_{\text{free response}} + \underbrace{s_j(1)\Delta u(k+1) + s_j(2)\Delta u(k)}_{\text{forced response}}\end{aligned}$$

$$\vdots$$

$$\begin{aligned}\hat{x}_j(k+n-1|k) &= s_j(n)u(k+n-1-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k+n-1-i) \\ &= \underbrace{s_j(n)u(k-1)}_{\text{free response}} + \underbrace{\sum_{i=1}^{n-1} s_j(i)\Delta u(k+n-1-i)}_{\text{forced response}}\end{aligned}$$

$$\begin{aligned}\hat{x}_j(k+n|k) &= s_j(n)u(k+n-n) + \sum_{i=1}^{n-1} s_j(i)\Delta u(k+n-i) \\ &= s_j(n)u(k+n-n) + \sum_{i=1}^{n} s_j(i)\Delta u(k+n-i) - s_j(n)\Delta u(k+n-n) \\ &= \underbrace{s_j(n)\Delta u(k-1)}_{\text{free response}} + \underbrace{\sum_{i=1}^{n} s_j(i)\Delta u(k+n-i)}_{\text{forced response}}.\end{aligned}$$

for which $\hat{x}_j(i|k)$ denotes the predicted j^{th} state at time instant i given the information up to and including time instant k. If the i^{th} free response of the j^{th} state given the

information up to and including instant k is denoted by $f_j(i|k)$, then

$$\begin{bmatrix} \hat{x}_j(k+1|k) \\ \vdots \\ \hat{x}_j(k+n-1|k) \\ \hat{x}_j(k+n|k) \end{bmatrix} = \begin{bmatrix} f_j(k+1|k) \\ \vdots \\ f_j(k+n-1|k) \\ f_j(k+n|k) \end{bmatrix} + \begin{bmatrix} s_j(1) & 0 & 0 & \dots & 0 \\ s_j(2) & s_j(1) & 0 & \dots & 0 \\ s_j(3) & s_j(2) & s_j(1) & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & 0 \\ s_j(n) & s_j(n-1) & s_j(n-2) & \dots & s_j(1) \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+n-1) \end{bmatrix}. \tag{4.8}$$

An expression for computing a vector $\mathbf{f}(k)$ can be obtained from

$$\begin{aligned} f_j(k+1|k+1) &= f_j(k+1|k) + s_j(1)\Delta u(k) \\ f_j(k+2|k+1) &= f_j(k+2|k) + s_j(2)\Delta u(k) \\ &\vdots \\ f_j(k+n-2|k+1) &= f_j(k+n-2|k) + s_j(n-2)\Delta u(k) \\ f_j(k+n-1|k+1) &= f_j(k+n-1|k) + s_j(n-1)\Delta u(k) \\ f_j(k+n|k+1) &= f_j(k+n-1|k) + s_j(n)\Delta u(k), \end{aligned}$$

for which the last equation for $f_j(k+n|k+1)$ repeats $f_j(k+n-1|k)$, since the transient response to a step input is assumed to end after n time instants. Therefore,

the recursion relation for $\mathbf{f}_j(k)$ is

$$\mathbf{f}_j(k+1) = \begin{bmatrix} f_j(k+1|k+1) \\ f_j(k+2|k+1) \\ f_j(k+3|k+1) \\ \vdots \\ f_j(k+n-1|k+1) \\ f_j(k+n|k+1) \end{bmatrix} = \begin{bmatrix} f_j(k+1|k) \\ f_j(k+2|k) \\ f_j(k+3|k) \\ \vdots \\ f_j(k+n-1|k) \\ f_j(k+n-1|k) \end{bmatrix} + \begin{bmatrix} s_j(1) \\ s_j(2) \\ s_j(3) \\ \vdots \\ s_j(n-1) \\ s_j(n) \end{bmatrix} \Delta u(k). \tag{4.9}$$

With

$$\mathbf{f}_j(k) \equiv \begin{bmatrix} f_j(k|k) \\ f_j(k+1|k) \\ \vdots \\ f_j(k+n-2|k) \\ f_j(k+n-1|k) \end{bmatrix},$$

$$\mathbf{s}_j = \begin{bmatrix} s_j(1) \\ s_j(2) \\ \vdots \\ s_j(n-1) \\ s_j(n) \end{bmatrix},$$

and

$$M \equiv \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & 0 & 1 \\ 0 & \dots & \dots & \dots & 0 & 1 \end{bmatrix}, \tag{4.10}$$

equation (4.9) can be rewritten as

$$\boxed{\mathbf{f}_j(k+1) = M\mathbf{f}_j(k) + \mathbf{s}_j\Delta u(k)}. \tag{4.11}$$

For the control horizon is c and prediction horizon p, (4.8) can be written as

$$\begin{aligned}
\begin{bmatrix} \hat{x}_j(k+1|k) \\ \vdots \\ \hat{x}_j(k+p-1|k) \\ \hat{x}_j(k+p|k) \end{bmatrix} &= \begin{bmatrix} f_j(k+1|k) \\ \vdots \\ f_j(k+p-1|k) \\ f_j(k+p|k) \end{bmatrix} \\
&+ \begin{bmatrix} s_j(1) & 0 & 0 & \dots & 0 \\ s_j(2) & s_j(1) & 0 & \dots & 0 \\ s_j(3) & s_j(2) & s_j(1) & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & s_j(p-c) \\ s_j(p) & s_j(p-1) & s_j(p-2) & \dots & s_j(p-c+1) \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+c-1) \end{bmatrix} \\
&= T\mathbf{f}_j(k) + G_{j,p,c}\Delta\mathbf{u}(k),
\end{aligned} \tag{4.12}$$

where $T \in \mathbb{R}^{p \times n}$ is

$$\begin{bmatrix} 0 & 1 & 0 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{if } p \geq n, \tag{4.13}$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & \dots & \dots & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & \dots & 0 & \dots & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 0 & \dots & \dots & 0 \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 & 0 & \dots & 0 \end{bmatrix} \quad \text{if } p < n, \tag{4.14}$$

and $G_{j,p,c} \in \mathbb{R}^{p \times c}$ is

$$\begin{bmatrix} s_j(1) & 0 & 0 & \dots & 0 \\ s_j(2) & s_j(1) & 0 & \dots & 0 \\ s_j(3) & s_j(2) & s_j(1) & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & s_j(p-c) \\ s_j(p) & s_j(p-1) & s_j(p-2) & \dots & s_j(p-c+1) \end{bmatrix}. \tag{4.15}$$

If the unmeasured disturbance is included into the prediction for future states, then (4.12) becomes

$$\begin{bmatrix} \hat{x}_j(k+1|k) \\ \vdots \\ \hat{x}_j(k+p-1|k) \\ \hat{x}_j(k+p|k) \end{bmatrix} = T\mathbf{f}_j(k) + G_{j,p,c}\Delta\mathbf{u}(k) + \begin{bmatrix} w_j(k+1|k) \\ \vdots \\ w_j(k+p-1|k) \\ w_j(k+p|k) \end{bmatrix}, \tag{4.16}$$

for which $w_j(i|k)$ denotes the unmeasured disturbance for the j^{th} component of state at the i^{th} time instant given the information of system up to time instant k. DMC assumes that the unmeasured disturbances are constant in the present and the future, i.e.

$$w_j(k|k) = w_j(k+1|k) = w_j(k+2|k) = \cdots = w_j(k+p|k),$$

and can be estimated from the difference between the measured and the predicted states, i.e.

$$w_j(k|k) \approx x_{j,\text{measured}}(k) - f_j(k|k),$$

where $x_{j,\text{measured}}(k)$ denotes the measured j^{th} state at instant k. With these assumptions,

$$\begin{aligned} \begin{bmatrix} \hat{x}_j(k+1|k) \\ \vdots \\ \hat{x}_j(k+p-1|k) \\ \hat{x}_j(k+p|k) \end{bmatrix} &\approx T\mathbf{f}_j(k) + G_{j,p,c}\Delta\mathbf{u}(k) + \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix} [x_{j,\text{measured}}(k) - f_j(k|k)] \\ &= \mathbf{x}_j^{\text{present}}(k) + G_{j,p,c}\Delta\mathbf{u}(k), \end{aligned} \tag{4.17}$$

where

$$\mathbf{x}_j^{\text{present}}(k) \equiv T\mathbf{f}_j(k) + \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix} \left[x_{j,\text{measured}}(k) - f_j(k|k) \right]. \tag{4.18}$$

Consequently, the summation $\sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \hat{\mathbf{x}}(k+i)||_2^2$ in the optimization objective can be rewritten as

$$\begin{bmatrix} \mathbf{x}_1^{\text{s.p.}} - \mathbf{x}_1^{\text{present}}(k) - G_{1,p,c}\Delta\mathbf{u}(k) \\ \mathbf{x}_2^{\text{s.p.}} - \mathbf{x}_2^{\text{present}}(k) - G_{2,p,c}\Delta\mathbf{u}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{s.p.}} - \mathbf{x}_{n_x}^{\text{present}}(k) - G_{n_x,p,c}\Delta\mathbf{u}(k) \end{bmatrix}^{\top} \begin{bmatrix} \mathbf{x}_1^{\text{s.p.}} - \mathbf{x}_1^{\text{present}}(k) - G_{1,p,c}\Delta\mathbf{u}(k) \\ \mathbf{x}_2^{\text{s.p.}} - \mathbf{x}_2^{\text{present}}(k) - G_{2,p,c}\Delta\mathbf{u}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{s.p.}} - \mathbf{x}_{n_x}^{\text{present}}(k) - G_{n_x,p,c}\Delta\mathbf{u}(k) \end{bmatrix}, \tag{4.19}$$

where

$$\mathbf{x}_j^{\text{s.p.}} \equiv \begin{bmatrix} x_j^{\text{s.p.}}(k+1) \\ x_j^{\text{s.p.}}(k+2) \\ \vdots \\ x_j^{\text{s.p.}}(k+p) \end{bmatrix}.$$

4.3.4 Rewriting the model-based control optimization

Defining the control error

$$\mathbf{e}_j(k) \equiv \mathbf{x}_j^{\text{s.p.}} - \mathbf{x}_j^{\text{present}}(k), \tag{4.20}$$

and rewriting (4.19) as

$$\begin{bmatrix} \mathbf{e}_1(k) - G_{1,p,c}\Delta\mathbf{u}(k) \\ \mathbf{e}_2(k) - G_{2,p,c}\Delta\mathbf{u}(k) \\ \vdots \\ \mathbf{e}_{n_x}(k) - G_{n_x,p,c}\Delta\mathbf{u}(k) \end{bmatrix}^\top \begin{bmatrix} \mathbf{e}_1(k) - G_{1,p,c}\Delta\mathbf{u}(k) \\ \mathbf{e}_2(k) - G_{2,p,c}\Delta\mathbf{u}(k) \\ \vdots \\ \mathbf{e}_{n_x}(k) - G_{n_x,p,c}\Delta\mathbf{u}(k) \end{bmatrix},$$

the $\Delta\mathbf{u}(k)$-dependent part can be written as

$$\boxed{\sum_{j=1}^{n_x} \Delta\mathbf{u}^\top(k) G_{j,p,c}^\top G_{j,p,c}\Delta\mathbf{u}(k) - 2\mathbf{e}_j(k) G_{j,p,c}\Delta\mathbf{u}(k)}. \tag{4.21}$$

The constraints (4.3) can be rewritten as

$$\begin{bmatrix} G_{1,p,c} \\ G_{2,p,c} \\ \vdots \\ G_{n_x,p,c} \end{bmatrix} \Delta\mathbf{u}(k) \leq \mathbf{x}_{\max} - \begin{bmatrix} \mathbf{x}_1^{\text{present}}(k) \\ \mathbf{x}_2^{\text{present}}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{present}}(k) \end{bmatrix}$$

$$-\begin{bmatrix} G_{1,p,c} \\ G_{2,p,c} \\ \vdots \\ G_{n_x,p,c} \end{bmatrix} \Delta\mathbf{u}(k) \leq -\mathbf{x}_{\min} + \begin{bmatrix} \mathbf{x}_1^{\text{present}}(k) \\ \mathbf{x}_2^{\text{present}}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{present}}(k) \end{bmatrix}$$

$$I_L \Delta\mathbf{u}(k) \leq \mathbf{u}_{\max} - \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} u(k-1) \tag{4.22}$$

$$-I_L \Delta\mathbf{u}(k) \leq -\mathbf{u}_{\min} + \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} u(k-1)$$

$$I^{c\times c}\Delta\mathbf{u}(k) \leq \Delta\mathbf{u}_{\max}$$

$$-I^{c\times c}\Delta\mathbf{u}(k) \leq \Delta\mathbf{u}_{\min},$$

for which

$$I_L \in \mathbb{R}^{c\times c} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & \cdots & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 1 & 1 \end{bmatrix}. \tag{4.23}$$

4.4 PCE-based MPC

The optimized $\Delta\mathbf{u}(k)$ depends on the system model, since $\mathbf{s}_j$'s and $G_{j,p,c}$'s are model-dependent. Let us now examine what happens to system-model-based control if the model parameters are uncertain.

A PCE-based formulation for MPC applies similar steps as above. This formulation builds on Joel A. Paulson's Ph.D. thesis that considered standard DMC in which the vector of manipulated variables $\mathbf{u}(j)$ is directly optimized online. His PCE-based DMC algorithm had low online computational cost while being less sensitive to parameter uncertainties than DMC, but had poorer closed-loop performance than DMC when the constraints were no longer active.

The main contribution of this chapter is to develop a PCE-based DMC algorithm has low online computational cost while having better closed-loop performance than Joel's algorithm. The main idea is borrowed from the robust MPC literature, which replaces the vector of manipulated variables at time j, $\mathbf{u}(j)$, with

$$\mathbf{K}\mathbf{x}(j) + \mathbf{v}(j) \tag{4.24}$$

where the feedback gain matrix $\mathbf{K}$ is computed offline using an optimal control algorithm that ignores constraints. In the online optimization, the vector of optimization variables $\mathbf{u}(j)$ up to the control horizon c is replaced by the vector of optimization variables $\mathbf{v}(j)$ up to the control horizon c. The robust MPC literature has demonstrated that such a reformulation results in improved worst-case performance for uncertain parameters that belong to deterministic sets.

In what follows, this idea of incorporating feedback into the MPC algorithm is applied to time-invariant parameters described by probability distribution functions. In this formulation, the feedback gain matrix $\mathbf{K}$ is computed offline using an offline control algorithm, such as linear-quadratic control of the $\mathcal{H}_2$-optimal PCE-based BMI formulation from Chapter 3. A case study demonstrates the improved closed-loop performance of this formulation.

4.4.1 Potential effects of parametric uncertainties

As a demonstrative example, consider a linear time-invariant system described by (4.1), with the parameters and values for a model-based control optimization listed in Table 4.1. Consider the control action u of the form:

$$u = -\mathbf{K}\mathbf{x} + v, \tag{4.25}$$

for which $\mathbf{K}$ is computed off-line by the linear-quadratic (LQ) control for the system, which has the optimization objective

$$\min_{\mathbf{K}} \int_0^\infty (\mathbf{x}^\top Q\mathbf{x} + u^\top Ru)\,\mathrm{d}t$$

with $Q = I^{n_x \times n_x}$ and $R = r$. With Inserting (4.25) into (4.1) gives

$$\dot{\mathbf{x}} = A\mathbf{x} + B(-\mathbf{K}\mathbf{x} + v) = (A - B\mathbf{K})\mathbf{x} + Bv. \tag{4.26}$$

As a result of this transformation,

$$\Delta u = -\Delta(\mathbf{K}\mathbf{x}) + \Delta v = -\mathbf{K}\Delta\mathbf{x} + \Delta v$$

the optimization problem (4.2) becomes

$$\begin{aligned}
\min_{\Delta\mathbf{v}(k)} J &= \min_{\Delta\mathbf{v}(k)} \sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \hat{\mathbf{x}}(k+i)||_2^2 + r \sum_{i=0}^{c-1} \left[-\mathbf{K}\Delta\mathbf{x}(k+i) + \Delta v(k+i)\right]^2 \\
&= \min_{\Delta\mathbf{v}(k)} \sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \hat{\mathbf{x}}(k+i)||_2^2 + r\left\{-\mathbf{K}\left[\mathbf{x}(k) - \mathbf{x}(k-1)\right] + \Delta v(k)\right\}^2 \\
&\quad + r\left\{-\mathbf{K}\left[\hat{\mathbf{x}}(k+1) - \mathbf{x}(k)\right] + \Delta v(k+1)\right\}^2 + r \sum_{i=2}^{c-1} \left[-\mathbf{K}\Delta\hat{\mathbf{x}}(k+i) + \Delta v(k+i)\right]^2.
\end{aligned} \tag{4.27}$$

Along with the change of the optimization objective is the change of constraints (4.3):

$$\begin{aligned}
\begin{bmatrix} G_{1,p,c} \\ G_{2,p,c} \\ \vdots \\ G_{n_x,p,c} \end{bmatrix} \Delta\mathbf{v}(k) &\leq \mathbf{x}_{\max} - \begin{bmatrix} \mathbf{x}_1^{\text{present}}(k) \\ \mathbf{x}_2^{\text{present}}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{present}}(k) \end{bmatrix} \\
-\begin{bmatrix} G_{1,p,c} \\ G_{2,p,c} \\ \vdots \\ G_{n_x,p,c} \end{bmatrix} \Delta\mathbf{v}(k) &\leq -\mathbf{x}_{\min} + \begin{bmatrix} \mathbf{x}_1^{\text{present}}(k) \\ \mathbf{x}_2^{\text{present}}(k) \\ \vdots \\ \mathbf{x}_{n_x}^{\text{present}}(k) \end{bmatrix} \\
I_L\left[-\Delta\mathbf{X}^\top(k)\mathbf{K}^\top + \Delta\mathbf{v}(k)\right] + \mathbf{u}(k-1) &\leq \mathbf{u}_{\max} \\
-I_L\left[-\Delta\mathbf{X}^\top(k)\mathbf{K}^\top + \Delta\mathbf{v}(k)\right] - \mathbf{u}(k-1) &\leq -\mathbf{u}_{\min} \\
-\Delta\mathbf{X}^\top(k)\mathbf{K}^\top + \Delta\mathbf{v}(k) &\leq \Delta\mathbf{u}_{\max} \\
\Delta\mathbf{X}^\top(k)\mathbf{K}^\top - \Delta\mathbf{v}(k) &\leq -\Delta\mathbf{u}_{\min}
\end{aligned} \tag{4.28}$$

for which the $\mathbf{s}_j$ and $G_{j,p,c}$ are computed based on the response of $\mathbf{x}$ to the step change in v and

$$\Delta\mathbf{X}(k) = \begin{bmatrix} | & | & | & | \\ \mathbf{x}(k) - \mathbf{x}(k-1) & \hat{\mathbf{x}}(k+1) - \mathbf{x}(k) & \dots & \hat{\mathbf{x}}(k+c-1) - \hat{\mathbf{x}}(k+c-2) \\ | & | & | & | \end{bmatrix}.$$

Figure 4-3 demonstrates the use of DMC to make the system state reach its set point without constraint violation.

Now consider what happens if the parameter values different from their true values are used to compute $\mathbf{s}_j$ and $G_{j,p,c}$. Instead of $A = -1$ and $B = 1$, consider the $\mathbf{s}_j$ and $G_{j,p,c}$ computed with $A = -3$ and $B = 0.5$. Figure 4-4 demonstrates how the DMC computed with $A = -3$ and $B = 0.5$ fails to satisfy the state constraint for the system with $A = -1$ and $B = 1$.

4.4.2 Background on Polynomial Chaos Expansions

Consider again the system described by (4.1) with parameters described in Table 4.2. With uncertainties in A and B, the same problem as described in Section 4.4.1 is encountered if the MPC is computed for the nominal values of A and B. DMC based on the nominal values of uncertain parameters does not take into consideration the variance in the state prediction introduced by the parametric uncertainties. Here the variance information is taken into account using polynomial chaos expansion (PCE), which approximates the dependence of states on uncertain parameters with appropriate choices of polynomial expansions. For uncertain parameters denoted by $\boldsymbol{\xi}$,

$$x_j(t,\boldsymbol{\xi}) \approx \sum_{i=1}^{N} x_{i,j}(t)\phi_i(\boldsymbol{\xi}) = \begin{bmatrix} x_{1,j}, & x_{2,j}, & \ldots, & x_{N,j} \end{bmatrix} \begin{bmatrix} \phi_1(\boldsymbol{\xi}) \\ \phi_2(\boldsymbol{\xi}) \\ \vdots \\ \phi_N(\boldsymbol{\xi}) \end{bmatrix} = \mathbf{x}_j^\top \boldsymbol{\phi}_N, \qquad (4.29)$$

for which the $x_j(t)$ denote the expansion coefficients; $\phi_j(\boldsymbol{\xi})$ denotes the expansion polynomials, the type of which depend on the type of $\boldsymbol{\xi}$; N denotes the number of polynomials used in the expansion, and

$$\mathbf{x}_j = \begin{bmatrix} x_{1,j} \\ x_{2,j} \\ \vdots \\ x_{N,j} \end{bmatrix}, \quad \boldsymbol{\phi}_N = \begin{bmatrix} \phi_1(\boldsymbol{\xi}) \\ \phi_2(\boldsymbol{\xi}) \\ \vdots \\ \phi_N(\boldsymbol{\xi}) \end{bmatrix}.$$

An advantage of using PCE is the convenience to compute the expected value and the variance of $x_j(t, \boldsymbol{\xi})$ with respect to $\boldsymbol{\xi}$:

$$\mathbb{E}_{\boldsymbol{\xi}}\left[x_j(t,\boldsymbol{\xi})\right] = \left\langle \sum_{i=1}^{\infty} x_{i,j}(t)\phi_i(\boldsymbol{\xi}), 1 \right\rangle = x_{1,j}(t)$$

$$\mathbf{Var}_{\boldsymbol{\xi}}\left[x_j(t,\boldsymbol{\xi})\right] = \mathbb{E}_{\boldsymbol{\xi}}\left[x_j^2(t,\boldsymbol{\xi})\right] - \mathbb{E}_{\boldsymbol{\xi}}^2\left[x_j(t,\boldsymbol{\xi})\right] \approx \sum_{i=2}^{N} x_{i,j}^2(t)\left\langle \phi_i^2(\boldsymbol{\xi})\right\rangle,$$

as a result of

$$\langle \phi_i, \phi_j \rangle = \delta_{ij}.$$

4.4.3 Transformation of the original system of equations using PCE

Using (4.29), the state vector can be approximated by

$$\mathbf{x}(t,\boldsymbol{\xi}) \approx \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_{n_x}^\top \end{bmatrix} \boldsymbol{\phi}_N. \tag{4.30}$$

Substituting (4.30) into (4.1) gives

$$\begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top \end{bmatrix} \boldsymbol{\phi}_N \approx (A - B\mathbf{K}) \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_{n_x}^\top \end{bmatrix} \boldsymbol{\phi}_N + Bv$$

$$\boldsymbol{\phi}_N^\top \begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top \end{bmatrix}^\top \approx \boldsymbol{\phi}_N^\top \begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top \end{bmatrix}^\top (A - B\mathbf{K})^\top + v^\top B^\top$$

$$\boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top \begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top \end{bmatrix}^\top \approx \boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top \begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top \end{bmatrix}^\top (A - B\mathbf{K})^\top + \boldsymbol{\phi}_N v^\top B^\top$$

$$I^{n_x \times n_x} \otimes (\boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top) \dot{\boldsymbol{\mathcal{X}}} \approx (A - B\mathbf{K}) \otimes (\boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top) \boldsymbol{\mathcal{X}} + B \otimes \boldsymbol{\phi}_N v$$

$$\mathbb{E}_{\boldsymbol{\xi}} \left[I^{n_x \times n_x} \otimes (\boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top) \right] \dot{\boldsymbol{\mathcal{X}}} \approx \mathbb{E}_{\boldsymbol{\xi}} \left[(A - B\mathbf{K}) \otimes (\boldsymbol{\phi}_N \boldsymbol{\phi}_N^\top) \right] \boldsymbol{\mathcal{X}} + \mathbb{E}_{\boldsymbol{\xi}} \left[B \otimes \boldsymbol{\phi}_N \right] v, \quad (4.31)$$

for which

$$\boldsymbol{\mathcal{X}} \equiv \mathbf{vec}\left\{\begin{bmatrix}\dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_{n_x}^\top\end{bmatrix}^\top\right\} = \begin{bmatrix} x_{1,1} \\ x_{2,1} \\ \vdots \\ x_{N,1} \\ x_{1,2} \\ x_{2,2} \\ \vdots \\ x_{N,2} \\ \vdots \\ x_{1,n_x} \\ x_{2,n_x} \\ \vdots \\ x_{N,n_x} \end{bmatrix} \in \mathbb{R}^{Nn_x \times 1}.$$

Note that

$$\mathbb{E}_{\boldsymbol{\xi}}[\mathbf{x}(t,\boldsymbol{\xi})] = \underbrace{\begin{bmatrix} \boldsymbol{\epsilon} & \mathbf{0} & \dots & \mathbf{0} & \dots \\ \mathbf{0} & \boldsymbol{\epsilon} & \mathbf{0} & \dots & \dots \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\epsilon} & \mathbf{0} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{\epsilon} \end{bmatrix}}_{\in \mathbb{R}^{n_x \times Nn_x}} \boldsymbol{\mathcal{X}}$$

$$\sum_{j=1}^{n_x} \mathbf{Var}_{\boldsymbol{\xi}}[x_j(t,\boldsymbol{\xi})] = \underbrace{\begin{bmatrix} \Phi_N & 0 & 0 & \dots & 0 \\ 0 & \Phi_N & 0 & \dots & 0 \\ 0 & 0 & \Phi_N & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & 0 & \Phi_N \end{bmatrix}}_{\in \mathbb{R}^{Nn_x \times Nn_x}} \boldsymbol{\mathcal{X}},$$

for which

$$\boldsymbol{\epsilon} = [1, 0, \ldots, 0] \in \mathbb{R}^{1\times N}$$
$$\Phi_N = \mathbf{diag}(0, \phi_2(\boldsymbol{\xi}), \ldots, \phi_N(\boldsymbol{\xi})).$$

4.4.4 Mathematical formulation of PCE-based MPC with embedded feedback

The MPC-based control optimization in the presence of parametric uncertainties can be reformulated as

$$\begin{aligned}
&\min_{\Delta\mathbf{v}(k)} \sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \mathbb{E}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}(k+i)\right]||_2^2 \\
&\qquad + r_0 \sum_{i=1}^{p}\sum_{j=1}^{n_x} \mathbf{Var}_{\boldsymbol{\xi}}\left[\hat{x}_j(k+i)\right] \qquad (4.32)\\
&+r_1 \sum_{i=0}^{c-1} \left[-\mathbf{K}\Delta\mathbf{x}(k+i) + \Delta v(k+i)\right]^2 \\
&= \min_{\Delta\mathbf{v}(k)} \sum_{i=1}^{p} ||\mathbf{x}^{\text{s.p.}}(k+i) - \mathbb{E}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}(k+i)\right]||_2^2 \\
&\qquad + r_0 \sum_{i=1}^{p}\sum_{j=1}^{n_x} \mathbf{Var}_{\boldsymbol{\xi}}\left[\hat{x}_j(k+i)\right] \\
&+r_1 \left\{-\mathbf{K}\left[\mathbf{x}(k) - \mathbf{x}(k-1)\right] + \Delta v(k)\right\}^2 \\
&+ r_1 \left\{-\mathbf{K}\left[\hat{\mathbf{x}}(k+1) - \mathbf{x}(k)\right] + \Delta v(k+1)\right\}^2 + r_1 \sum_{i=2}^{c-1} \left[-\mathbf{K}\Delta\hat{\mathbf{x}}(k+i) + \Delta v(k+i)\right]^2, \qquad (4.33)
\end{aligned}$$

where r_0 denotes the relative importance of the variance of the predicted states to the deviation of the predicted states from the set point.

For the constraints, $\mathbf{x}_{\min} \leq \hat{\mathbf{x}} \leq \mathbf{x}_{\max}$ is transformed to

$$\mathbb{E}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}\right] + \alpha_1 \mathbf{SD}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}\right] \leq \mathbf{x}_{\max}$$
$$\mathbb{E}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}\right] - \alpha_2 \mathbf{SD}_{\boldsymbol{\xi}}\left[\hat{\mathbf{x}}\right] \geq \mathbf{x}_{\min}.$$

where α_1 and α_2 are non-negative numbers that consider the variance in the predicted states due to parametric uncertainties.

4.4.5 Case Study

With the parameters listed in Table 4.2, the PCE-based MPC algorithm is able to satisfy all the constraints and reach the state set point (Figure 4-5).

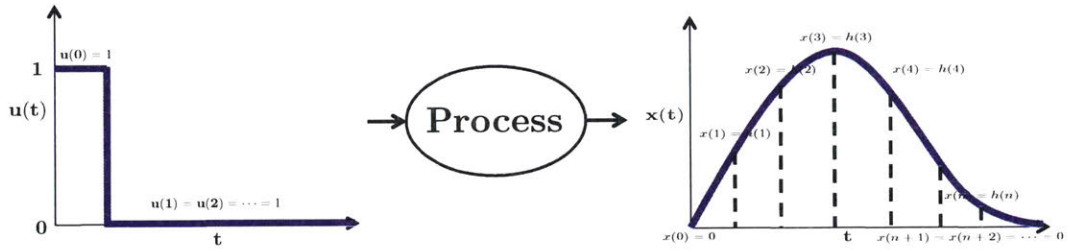

Figure 4-1: Finite impulse response.

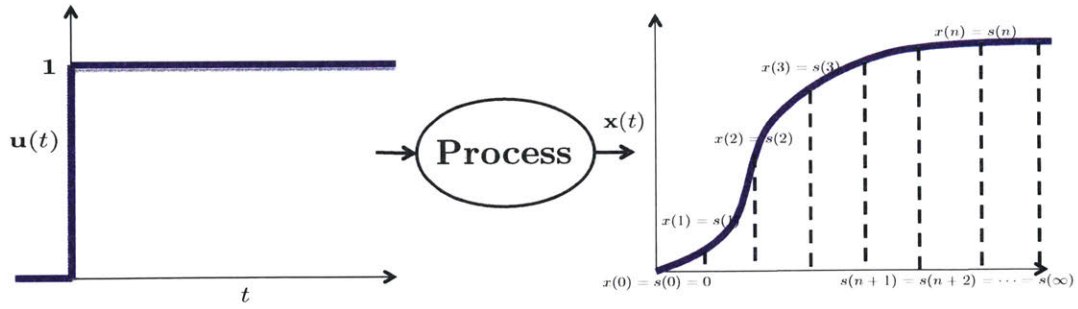

Figure 4-2: Finite step response.

A	-1
B	1
r	0.2
c	5
p	20
T_s	0.2 s
x_initial	0
$\mathbf{x}^\mathrm{s.p.}$	$[0.12, 0.12, \ldots, 0.12]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{x}_\mathrm{max}$	$[0.132, 0.132, \ldots, 0.132]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{x}_\mathrm{min}$	$[-0.15, 0.15, \ldots, 0.15]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{u}_\mathrm{max}$	$[0.25, 0.25, \ldots, 0.25]^\top \in \mathbb{R}^{c\times 1}$
$\mathbf{u}_\mathrm{min}$	$[-0.25, -0.25, \ldots, -0.25]^\top \in \mathbb{R}^{c\times 1}$
$\Delta\mathbf{u}_\mathrm{max}$	$[0.1, 0.1, \ldots, 0.1]^\top \in \mathbb{R}^{c\times 1}$
$\Delta\mathbf{u}_\mathrm{min}$	$[-0.1, -0.1, \ldots, -0.1]^\top \in \mathbb{R}^{c\times 1}$

Table 4.1: Parameters for the example system in Section 4.4.1.

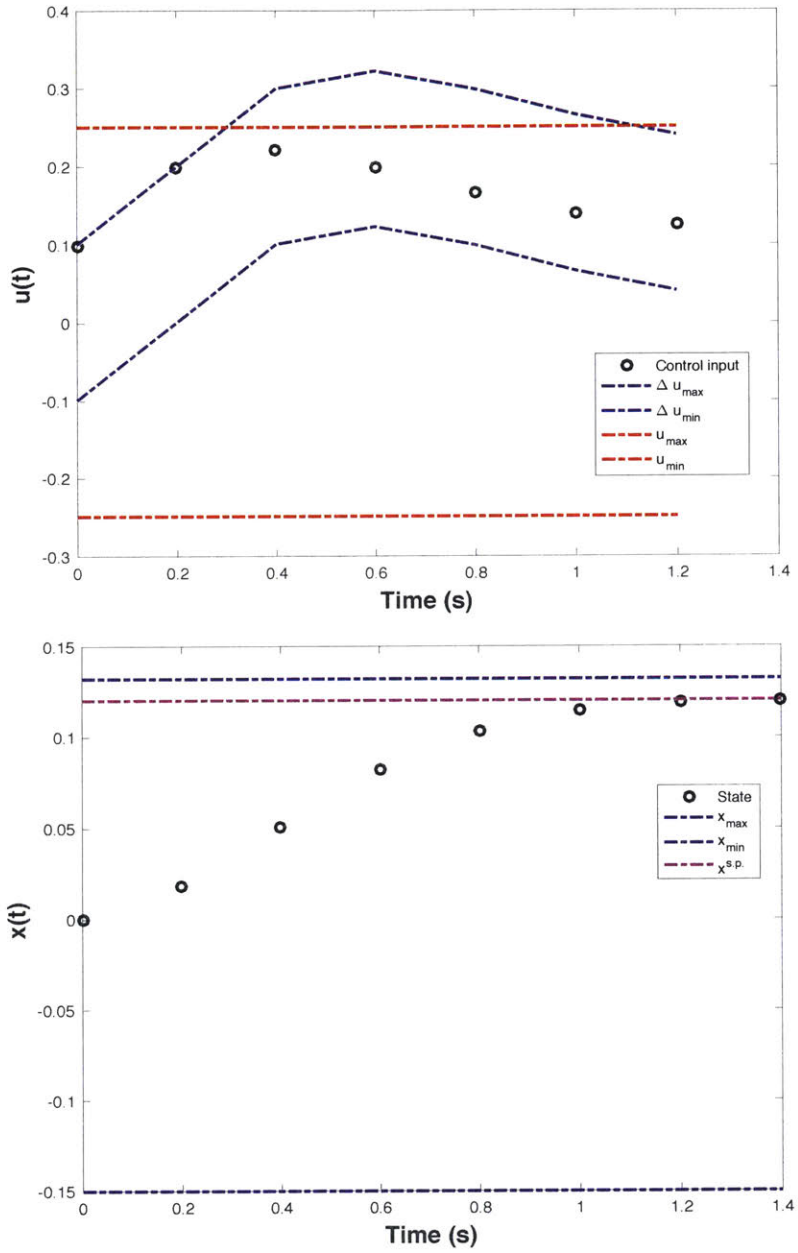

Figure 4-3: DMC for the nominal system results in the system state reaching its set point without constraint violation.

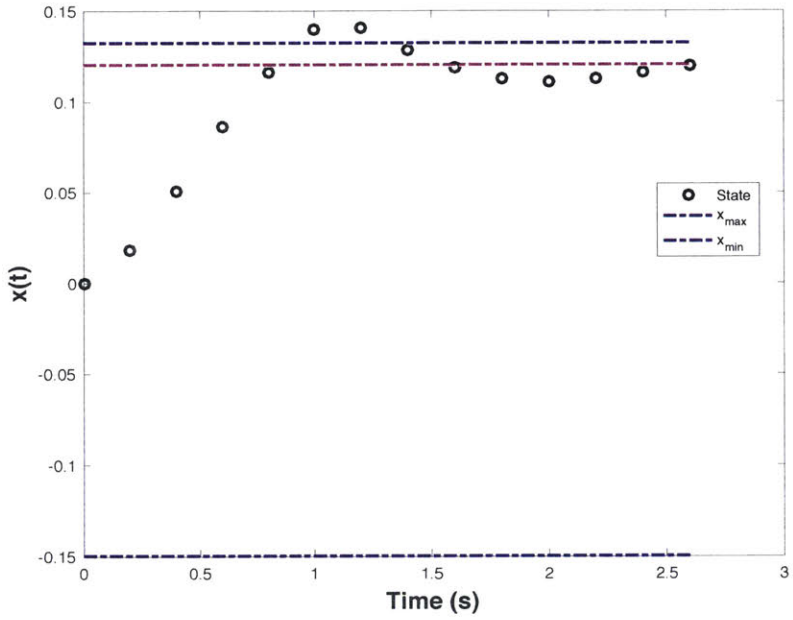

Figure 4-4: DMC based on the nominal system is unable to satisfy the state constraint when the control is applied to the actual system, illustrating the effect of model uncertainty for a control system that does not take that uncertainty into account.

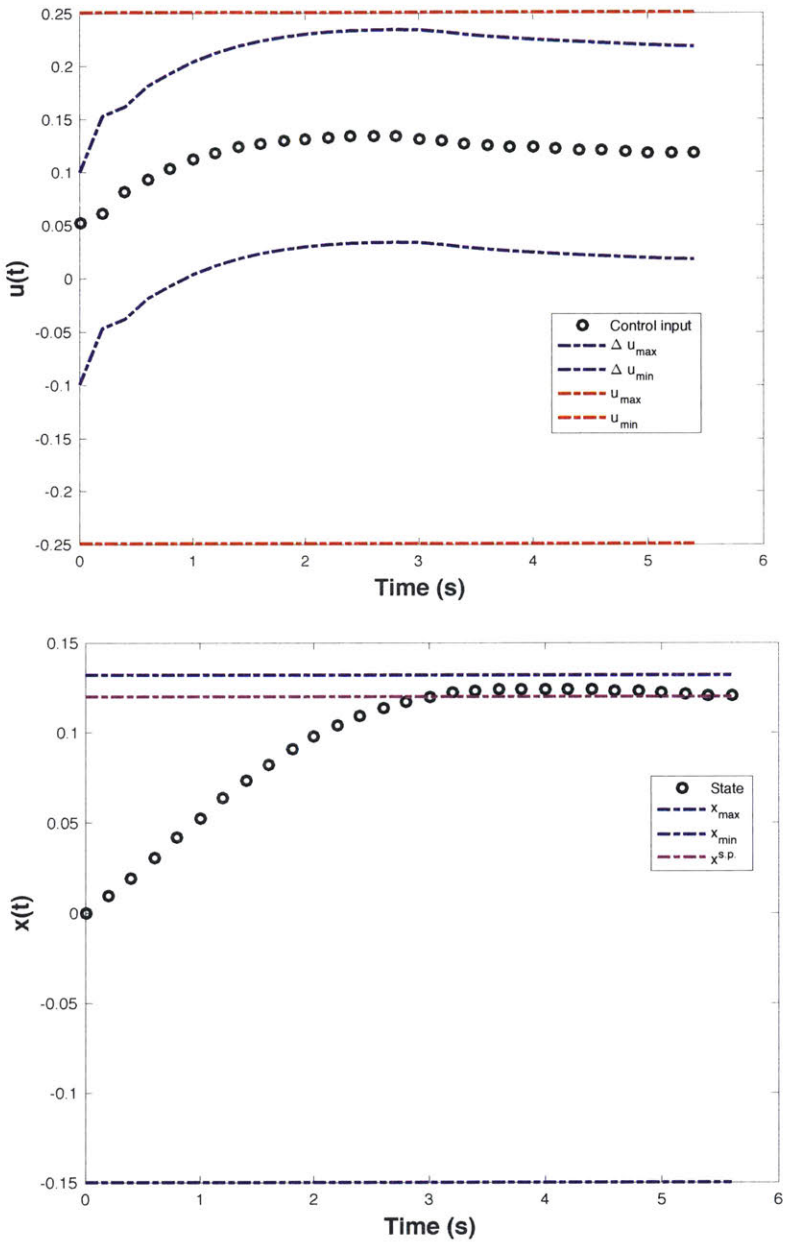

Figure 4-5: PCE-based MPC converges the state to its set point without constraint violation, even with significant model uncertainty.

θ_1	$\mathcal{U}(-1, 1)$
θ_2	$\mathcal{U}(-1, 1)$
A	$2\theta_1 - 3$; nominal value $= -3$
B	$\theta_2 + 0.5$; nominal value $= 0.5$
r_0	0
r_1	0.2
α_1	50
α_2	0
c	5
p	20
T_s	0.2 s
x_initial	0
$\mathbf{x}^\mathrm{s.p.}$	$[0.12, 0.12, \ldots, 0.12]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{x}_\mathrm{max}$	$[0.132, 0.132, \ldots, 0.132]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{x}_\mathrm{min}$	$[-0.15, 0.15, \ldots, 0.15]^\top \in \mathbb{R}^{p\times 1}$
$\mathbf{u}_\mathrm{max}$	$[0.25, 0.25, \ldots, 0.25]^\top \in \mathbb{R}^{c\times 1}$
$\mathbf{u}_\mathrm{min}$	$[-0.25, -0.25, \ldots, -0.25]^\top \in \mathbb{R}^{c\times 1}$
$\Delta\mathbf{u}_\mathrm{max}$	$[0.1, 0.1, \ldots, 0.1]^\top \in \mathbb{R}^{c\times 1}$
$\Delta\mathbf{u}_\mathrm{min}$	$[-0.1, -0.1, \ldots, -0.1]^\top \in \mathbb{R}^{c\times 1}$

Table 4.2: Parameters for the example system in Section 4.4.2.

Chapter 5

Conclusions and Future Directions

This thesis includes several firsts, with the main contribution being the first rigorous theoretical frameworks and numerical algorithms for the optimal control of linear time-invariant systems with time-invariant parameters described by probability distribution functions (Chapter 3). A combination of the Galerkin method and optimal control theory results in offline optimizations over bilinear matrix inequalities (BMI) that can be solved using commercial software. Formulations are derived for both $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control objectives. The formulations are the first to handle the static output feedback control problem for systems with time-invariant probabilistic parameters, which is general enough to include the state feedback control problem and the low- and full-order output feedback control problem as special cases.

Another first is the development of a fast model predictive control algorithm for linear time-invariant systems with time-invariant probabilistic parameters that incorporates a feedback control law to improve closed-loop performance (Chapter 4). The embedded feedback control law is computed offline using the BMI formulation in Chapter 3 whereas a quadratic program with a low number of optimization variables is solved online in which the Hessian and Jacobian are computed from PCE coefficients computed using Galerkin projection.

The main directions for future students would be to extend the theoretical framework and numerical algorithm in Chapter 4 to linear time-invariant systems in which the states are not measured, which would be relatively straightforward since Chapter

3 already addresses output feedback control design. A much more challenging direction would be to extend the methodologies of Chapters 3 and 4 to nonlinear dynamical systems. A more promising approach to tackling nonlinear dynamical systems is to first derive the methodologies for special cases, such as linear time-varying (LTV) systems, linear parameter-varying (LPV) systems (aka gain scheduled systems), and polynomial systems (that is, when the nonlinear dependence on the states are described by polynomials). Many startup, shutdown, and changeover operations can be accurately modeled as LTV and LPV systems, whereas many chemical and biological processes can be described as polynomial systems. A challenge in such formulations is that the Galerkin method for computing the PCE coefficients is not as computationally efficient when the system is not linear time-invariant, and the most commonly used alternative methods for computing the PCE coefficients are much more computationally expensive. It is likely that exploiting mathematical structure would be needed to develop efficient PCE computation and efficient stochastic optimal control algorithms for LTV, LPV, and polynomial systems.

Bibliography

[1] Masoud Babaei, Ali Alkhatib, and Indranil Pan. Robust optimization of subsurface flow using polynomial chaos and response surface surrogates. *Computational Geosciences*, 19(5):979–998, 2015.

[2] R. Becker, Stephen G. Hall, and Berc Rustem. Robust optimal decisions with stochastic nonlinear economic systems. *Journal of Economic Dynamics and Control*, 18(1):125–147, 1994.

[3] Dennis S Bernstein. Robust static and dynamic output-feedback stabilization: Deterministic and stochastic perspectives. *IEEE Transactions on Automatic Control*, 32(12):1076–1084, 1987.

[4] Raktim Bhattacharya. Robust state feedback control design with probabilistic system parameters. In *53rd IEEE Conference on Decision and Control*, pages 2828–2833. IEEE, 2014.

[5] Shmuel Boyarski and Uri Shaked. Robust $\mathcal{H}_\infty$ control design for best mean performance over an uncertain-parameters box. *Systems & control letters*, 54(6):585–595, 2005.

[6] J. Brewer. Kronecker products and matrix calculus in system theory. *IEEE Transactions on Automatic Control*, 25(9):772–781, 1978.

[7] Thierry Crestaux, Olivier Le Maitre, and Jean-Marc Martinez. Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering & System Safety*, 94(7):1161–1172, 2009.

[8] Urmila M. Diwekar and Jayant R. Kalagnanam. Efficient sampling technique for optimization under uncertainty. *AIChE Journal*, 43(2):440–447, 1997.

[9] Jiuxiang Dong and Guang-Hong Yang. Robust static output feedback control synthesis for linear continuous systems with polytopic uncertainties. *Automatica*, 49(6):1821–1829, 2013.

[10] Michael Scott Eldred. Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design. *AIAA Paper*, 2274(2009):1–37, 2009.

[11] James Fisher and Raktim Bhattacharya. Stability analysis of stochastic systems using polynomial chaos. In *American Control Conference, 2008*, pages 4250–4255. IEEE, 2008.

[12] James Fisher and Raktim Bhattacharya. Linear quadratic regulation of systems with stochastic parameter uncertainties. *Automatica*, 45(12):2831–2841, 2009.

[13] José Claudio Geromel, Jacques Bernussou, and Maurício Carvalho de Oliveira. $\mathcal{H}_2$-norm optimization with constrained dynamic output feedback controllers: decentralized and reliable control. *IEEE Transactions on Automatic Control*, 44(7):1449–1454, 1999.

[14] E. Gershon, U. Shaked, and I. Yaesh. $\mathcal{H}_\infty$ control and filtering of discrete-time stochastic systems with multiplicative noise. *Automatica*, 37:409–417, 2001.

[15] Ignacio E. Grossmann and Roger W. H. Sargent. Optimum design of chemical plants with uncertain parameters. *AIChE Journal*, 24(6):1021–1028, 1978.

[16] K. Gu. H_∞ control of systems under norm bounded uncertainties in all system matrices. *IEEE Transactions on Automatic Control*, 39(6):1320–1322, 1994.

[17] Didier Henrion and Jean-Bernard Lasserre. GloptiPoly: Global optimization over polynomials with Matlab and SeDuMi. *ACM Transactions on Mathematical Software (TOMS)*, 29(2):165–194, 2003.

[18] Franz S. Hover and Michael S. Triantafyllou. Application of polynomial chaos in stability and control. *Automatica*, 42(5):789–795, 2006.

[19] Elçin Içten, Zoltan K. Nagy, and Gintaras V. Reklaitis. Process control of a dropwise additive manufacturing system for pharmaceuticals using polynomial chaos expansion based surrogate model. *Computers & Chemical Engineering*, 83:221–231, 2015.

[20] Marianthi G. Ierapetritou, Joaquin Acevedo, and Efstratios N. Pistikopoulos. An optimization approach for process engineering problems under uncertainty. *Computers & Chemical Engineering*, 20(6–7):703–709, 1996.

[21] Ronald L. Iman and William J. Conover. Small sample sensitivity analysis techniques for computer models, with an application to risk assessment. *Communications in Statistics - Theory and Methods*, 9(17):1749–1842, 1980.

[22] K.-K. K. Kim, D. E. Shen, Z. K. Nagy, and R. D. Braatz. Wiener's polynomial chaos for the analysis and control of nonlinear dynamical systems with probabilistic uncertainties. *IEEE Control Systems*, 33(5):58–67, 2013.

[23] Kwang-Ki K. Kim, Dongying E. Shen, Zoltan K. Nagy, and Richard D. Braatz. Wiener's polynomial chaos for the analysis and control of nonlinear dynamical systems with probabilistic uncertainties [Historical Perspectives]. *Control Systems, IEEE*, 33(5):58–67, 2013.

[24] Umamaheswara Konda, Puneet Singla, Tarunraj Singh, and Peter D Scott. State uncertainty propagation in the presence of parametric uncertainty and additive white noise. *Journal of Dynamic Systems, Measurement, and Control*, 133(5):051009, 2011.

[25] Michal Kočvara and Michael Stingl. PENBMI userâĂŹs guide. *Available from http://www.penopt.com*, 2005.

[26] Christian Løvaas, María M Seron, and Graham C Goodwin. Robust output-feedback model predictive control for systems with unstructured uncertainty. *Automatica*, 44(8):1933–1943, 2008.

[27] Sergio Lucia, Joel A. Paulson, Rolf Findeisen, and Richard D. Braatz. On stability of stochastic linear systems via polynomial chaos expansions. In *American Control Conference*. AACC, 2017. accepted.

[28] Jasdeep Mandur and Hector Budman. Robust optimization of chemical processes using Bayesian description of parametric uncertainty. *Journal of Process Control*, 24(2):422–430, 2014.

[29] SO Reza Moheimani and Ian R Petersen. Optimal guaranteed cost control of uncertain systems via static and dynamic output feedback. *Automatica*, 32(4):575–579, 1996.

[30] Jeonghwa Moon, Seon Kim, and Andreas A. Linninger. Integrated design and control under uncertainty: embedded control optimization for plantwide processes. *Computers & Chemical Engineering*, 35(9):1718–1724, 2011.

[31] Zoltan K. Nagy and Richard D. Braatz. Open-loop and closed-loop robust optimal control of batch processes using distributional and worst-case analysis. *Journal of Process Control*, 14(4):411–422, 2004.

[32] Zoltan K. Nagy and Richard D. Braatz. Distributional uncertainty analysis using power series and polynomial chaos expansions. *Journal of Process Control*, 17(3):229–240, 2007.

[33] J. A. Paulson, E. Harinath, L. C. Foguth, and R. D. Braatz. Nonlinear model predictive control of systems with probabilistic time-invariant uncertainties. In *Proceedings of 5th IFAC Conference on Nonlinear Model Predictive Control*, pages 16–25, Seville, 2015.

[34] Ian R Petersen and Roberto Tempo. Robust control of uncertain systems: classical results and recent developments. *Automatica*, 50(5):1315–1335, 2014.

[35] Dario Piga and Alessio Benavoli. A unified framework for deterministic and probabilistic d-stability analysis of uncertain polynomial matrices. *IEEE Transactions on Automatic Control*, 2017. DOI:10.1109/TAC.2017.2699281.

[36] Mir S. Pishvaee, Masoud Rabbani, and Seyed A. Torabi. A robust optimization approach to closed-loop supply chain network design under uncertainty. *Applied Mathematical Modelling*, 35(2):637–649, 2011.

[37] Efstratios N. Pistikopoulos. Uncertainty in process design and operations. *Computers & Chemical Engineering*, 19:553–563, 1995.

[38] Brandon J. Reizman and Klavs F. Jensen. An automated continuous-flow platform for the estimation of multistep reaction kinetics. *Organic Process Research & Development*, 16(11):1770–1782, 2012.

[39] Berc Rustem and Melendres Howe. *Algorithms for worst-case design and applications to risk management.* Princeton University Press, 2009.

[40] Nikolaos V. Sahinidis. BARON: A general purpose global optimization software package. *Journal of Global Optimization*, 8(2):201–205, 1996.

[41] Nikolaos V. Sahinidis. Optimization under uncertainty: state-of-the-art and opportunities. *Computers & Chemical Engineering*, 28(6):971–983, 2004.

[42] Carsten Scherer, Pascal Gahinet, and Mahmoud Chilali. Multiobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, 42(7):896–911, 1997.

[43] Robert E. Skelton, Tetsuya Iwasaki, and Karolos M. Grigoriadis. *A unified algebraic approach to control design.* CRC Press, 1997.

[44] V. L. Syrmos, C. T. Abdallah, P. Dorato, and D. Grigoriadis. Static output feedback – A survey. *Automatica*, 33(2):125–137, 1997.

[45] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems.* Springer-Verlag, London, 2005.

[46] J. VanAntwerp and R. D. Braatz. A tutorial on linear and bilinear matrix inequalities. *Journal of Process Control*, 10(4):363–385, 2000.

[47] Arvind Varma, Massimo Morbidelli, and Hua Wu. *Parametric Sensitivity in Chemical Systems.* Cambridge University Press, 2005.

[48] M Vidyasagar and Vincent D Blondel. Probabilistic solutions to some np-hard matrix problems. *Automatica*, 37(9):1397–1405, 2001.

[49] Jing Wei and Matthew J. Realff. Sample average approximation methods for stochastic MINLPs. *Computers & Chemical Engineering*, 28(3):333–346, 2004.

[50] Dongbin Xiu and George E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002.

[51] Isaac Yaesh, Shmuel Boyarski, and Uri Shaked. Probability-guaranteed robust $\mathcal{H}_\infty$ performance analysis and state-feedback design. *Systems & control letters*, 48(5):351–364, 2003.

[52] Chaoping Zang, Michael I. Friswell, and John E. Mottershead. A review of robust optimal design and its application in dynamics. *Computers & Structures*, 83(4–5):315–326, 2005.

[53] Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control*. Prentice-Hall, New Jersey, 1996.