

# **Global optimisation, model predictive control and uncertainty quantification methodologies for distributed parameter systems**

A thesis submitted to the University of Manchester for the degree of  
Doctor of Philosophy  
in the Faculty of Science and Engineering

2021

Min Tao  
Department of Chemical Engineering  
School of Engineering

# Contents

<b>Contents</b>	<b>2</b>
<b>Abstract</b>	<b>4</b>
<b>Declaration of originality</b>	<b>5</b>
<b>Copyright statement</b>	<b>6</b>
<b>Acknowledgements</b>	<b>7</b>
<b>1 Introduction</b>	<b>12</b>
1.1 Large-scale distributed parameter systems . . . . .	12
1.1.1 Background . . . . .	12
1.1.2 Air combustion of NO <sub>x</sub> emissions . . . . .	14
1.1.3 Oxidation of CO on a catalytic surface . . . . .	15
1.2 Challenges for optimisation, control and uncertainty quantification of large-scale distributed parameter systems . . . . .	17
1.3 Research contributions and thesis structure . . . . .	20
1.3.1 Contributions . . . . .	20
1.3.2 Conference presentations . . . . .	22
1.3.3 Conference Proceedings . . . . .	22
<b>2 Literature review</b>	<b>24</b>
2.1 Optimisation and control . . . . .	26
2.2 Uncertainty quantification . . . . .	34
2.3 Optimisation and control under uncertainty . . . . .	37
2.4 Model reduction . . . . .	39
2.5 Conclusions and summary . . . . .	44

<b>3</b>	<b>Model reduction based global optimisation for large-scale steady state nonlinear systems</b>	<b>46</b>
3.1	Introduction . . . . .	46
3.2	Author's contribution . . . . .	47
<b>4</b>	<b>Robust model predictive control of large-scale distributed systems under uncertainty</b>	<b>80</b>
4.1	Introduction . . . . .	80
4.2	Author's contribution . . . . .	81
<b>5</b>	<b>Uncertainty quantification for distributed parameter systems: deterministic and stochastic cases</b>	<b>96</b>
5.1	Introduction . . . . .	96
5.2	Author's contribution . . . . .	97
<b>6</b>	<b>Robust steady-state optimisation of large-scale distributed systems under uncertainty</b>	<b>121</b>
6.1	Introduction . . . . .	121
6.2	Author's contribution . . . . .	122
<b>7</b>	<b>Conclusions and future work</b>	<b>157</b>
7.1	Conclusions . . . . .	157
7.2	Future work . . . . .	159

## Abstract

Large-scale distributed parameter systems cover a wide range of practical applications in industrial engineering, such as chemical tubular reactors, bio-production reactors, combustion processes and microscopic reactions on the surface. Intelligent operations including optimisation and control could improve process performance, satisfying process constraints. However, intelligent computations are challenging for large-scale systems due to the global optima issues, process uncertainty and expensive system evaluations. The research in this thesis presents model reduction based global optimisation, model predictive control and uncertainty quantification methodologies for large-scale distributed parameter systems. Firstly, a double reduction (principal component analysis and artificial neural networks) based global optimisation framework was built, which was then improved through piecewise affine and deep rectifier neural network reformulations. Then a combination of nonlinear model predictive control and polynomial chaos expansion was employed to robustly control distributed parameter systems under parametric uncertainty, where the offline reformulations (proper orthogonal decomposition and recurrent neural networks) based global optimisation method was utilised within recursive control steps. Next, an “equation-free” Monte Carlo uncertainty quantification methodology was proposed for large-scale distributed parameter systems, where the recursive projection method and lifting-restriction operations were performed to accelerate the computations of distributional steady states through a black-box dynamic simulator. Finally, a Bayesian optimisation approach was adopted to globally search optimal solutions for large-scale system under uncertainty, where parametric uncertainty was addressed using our presented uncertainty propagation algorithm. The proposed computational frameworks were validated through several practical chemical and biochemical production cases.

# **Declaration of originality**

I hereby confirm that no portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

# Copyright statement

- i The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.
- ii Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made *only* in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.
- iii The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.
- iv Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see <http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=24420>), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.library.manchester.ac.uk/about/regulations/>) and in The University’s policy on Presentation of Theses.

# Acknowledgements

First of all, I want to acknowledge the continued support and guidance of my supervisor, Professor Kostas Theodoropoulos, throughout this PhD. In particular, I would thank for his time, conversations and suggestions over my methodology development, computational works, writing and presentations.

I would also thank the financial support of University of Manchester and China Scholarship Council joint scholarship.

I would also thank my co-supervisor Dr. Jie Li, and members in Kostas' group, especially Dr. Panagiotis Petsagkourakis, for their help during my early stage of this PhD.

I would also thank my examiners Professor Vivek Dua, Dr. Nan Zhang and Dr. Dongda Zhang for their help to give comments to improve my research work.

Much thanks to my friends from the office B14 of Centre of Process Integration, and other friends in UK and China. I also would like to thanks my roommates for providing the quiet environment during the serious covid period.

Finally, I want to express my gratitude to my family, especially my parents. Specially, they would unconditionally support and respect my decisions for my personal life, even when they have different views with me.

# Nomenclature

## List of Abbreviations

*DPS* Distributed Parameter System

*ANN* Artificial Neural Network

*EF* Equation-free

*LP* Linear Programming

*MC* Monte Carlo

*MILP* Mixed Integer Linear Programming

*MIP* Mixed Integer Programming

*MOR* Model Order Reduction

*MPC* Model Predictive Control

*NLP* Nonlinear Programming

*NMPC* Nonlinear Model Predictive Control

*OCP* Optimal Control Problem

*PCA* Principal Component Analysis

*PCE* Polynomial Chaos Expansion

*PDE* Partial Differential Equation

*POD* Proper Orthogonal Decomposition

*PSE* Power Series Expansion

*RNN* Recurrent Neural Network

*RPM* Recursive Projection Method

*SQP* Sequential Quadratic Programming

*UQ* Uncertainty Quantification

### **List of Greek symbols**

$\beta$  Parameter for lower and upper probabilistic bounds

$\alpha^{lo}$  Lower bounds of the discrete-time variables

$\alpha^{up}$  Upper bounds of the discrete-time variables

$\lambda'$  Discrete manipulated variable(s)

$\lambda$  Manipulated variable(s)

$\mu$  Mean value of the discrete-time variables

$\Omega'$  Probabilistic domain of uncertain parameter(s)  $p$

$\Omega$  Boundaries of partial differential equations

$\rho'(\cdot)$  Simplified likelihood estimation function

$\rho(\cdot)$  Likelihood estimation function

$\tau$  The discretised time step

### **List of Latin symbols**

$\mathcal{G}(\cdot)$  Joint multivariate Gaussian distribution

$\mathcal{Y}$  Spatially distributed microscopic states

$D$  Input data across the design space

$d$  Design variable(s)

$l$  Characteristic length-scale parameters of Gaussian kernel

$p$  Uncertain parameter(s)

$Y^{ss}$  Steady-state values of discrete variables  $Y$

$y^{ss}$  Steady-state values of state variable(s)  $y$

$y_0$  The initial values of state variables  $y$

$\mathbf{Y}$	Spatially discrete state variables
$\mathbf{y}$	State variable(s)
$\mathbf{Z}$	Orthonormal basis for the slow subspace $\mathbb{P}$
$\mathbf{z}$	Measurement variables
$\mathbb{E}$	Expected value
$\mathbb{P}$	The slow domain subspace of the system
$\mathbb{Q}$	The complement subspace of $\mathbb{P}$
$\mathbb{V}$	Variance value
$\sigma_f^2$	Covariance magnitude of Gaussian kernel
$A$	The operator of boundary condition equations
$C_y$	Covariance matrix
$D$	Dissipative spatial differential operator
$D_1$	The diagonal matrix of the covariance matrix $C_y$
$EI(.)$	Expectation Improvement function
$F$	A time-stepper for the dynamic system
$G(.)$	Objective function
$g_{cons}(.)$	The general constraints
$H$	The reduced Jacobian
$h_{bds}(.)$	The right hand sides of the boundary conditions
$I(.)$	Improvement function
$K(.)$	Covariance of Gaussian distribution
$k$	Number of dominant modes
$M$	Number of realisations for uncertain parameter(s) $\mathbf{p}$
$m$	Number of spatial discrete interval points

$N$	Number of samples across the design space
$N_d$	Number of design variable(s)
$N_p$	Number of uncertain parameter(s)
$N_x$	Number of spatial dimension(s): 1, 2, 3
$N_y$	Number of state variable(s)
$N_z$	Number of measurement variable(s)
$P$	The orthogonal projection onto the slow subspace $\mathbb{P}$
$Q$	The orthogonal projection onto the subspace $\mathbb{Q}$
$R(\cdot)$	The nonlinear terms of partial differential equations
$t$	Time dimension
$tt$	Time interval for the time-stepper $F$
$x$	Spatial dimension(s)

# Chapter 1

## Introduction

### 1.1 Large-scale distributed parameter systems

#### 1.1.1 Background

Complex chemical processes usually comprise multiple phenomena involving spatial dynamics, known as distributed parameter systems (DPS) (Ahmed, 1987). In fact, most physical processes span multiple space and time scales, from the microscopic (Dittrich & Reuter, 1994; Leimkuhler & Matthews, 2016), to the mesoscopic (Reguera et al., 2005) to the macroscopic level (J. D. Anderson & Wendt, 1995; Bird et al., 2006). Such multi-scale spatiotemporal phenomena are often one of the great challenges for understanding complex multi-scale (bio)chemical and industrial processes, which are attracting numerous efforts for production-oriented investigation (Charpentier & McKenna, 2004). Furthermore, the high complexity of multi-scale DPS contributes to more inter-disciplinary research and collaborations within different disciplines on multiple scales/levels (Wolkenhauer et al., 2014).

Distributed parameter systems are widely found in the engineering and life sciences. For example, single phase or multiphase flow systems (Brennen & Brennen, 2005) in traditional mechanical and chemical engineering, would exhibit velocity variations in different times and space positions. Similarly, spatiotemporal multi-scale fluid flow patterns can be observed in aerodynamics. Moreover, growth of materials (such as thin films) through a range of reaction systems involve both macroscopic fluid and microscopic growth dynamics, typically described by multi-scale models (Kleijn et al., 2007). Furthermore, both emerging nanotechnology (Song et al., 2017) and micro-reactor technology systems (Roberge et al., 2005) can

be described as distributed parameter systems. Almost all systems dominated by physical phenomena belong to distributed parameter systems. Even in social science, the population level in ecological systems (Evans, 2012) may be modelled as a DPS, exhibiting dynamics in different areas.

There are two fundamental descriptions for distributed parameter systems at all scales. One is deterministic description, assuming the continuously evolving behaviours on space domain. Deterministic description could be quantified using deterministic (partial) differential equations. For example, widely existing processes in engineering applications, such as fluid flow (Constantin & Foias, 1988), heat and mass transport processes (Rosner, 2012), are often described using Partial Differential Equations (PDEs) (Mattheij et al., 2005). The other is stochastic description with the discrete behaviours on space scales. Stochastic description could be quantified by either stochastic differential equations (Protter, 2005) or stochastic events based simulators (Gillespie, 1977). Stochastic differential equations use so-called "white noise" to describe random influences on processes including closed form equations while stochastic simulators employ stochastic events based evolutionary processes without closed form equations. Typical examples of stochastic description could be found for distributed parameter systems including surface catalytic reactions (Boudart & Djéga-Mariadassou, 2014) and thin film decomposition (Armaou & Varshney, 2004). Moreover, complexity of distributed parameter systems could be derived from coupled processes. For instance, chemical reactor systems with catalyst pellets (Y.-N. Wang et al., 2001) involve coupled processes in two physical scales, the fluid flow and transport process in bulk gas scale and the diffusion-reaction process in pellet scale. Multiple coupled processes could exploit more details of systems but would increase the complexity of distributed parameter systems, leading to more challenges to cope with them. Specifically in chemical and biochemical engineering, early research focused on unit operations such as distillation, absorption and crystallisation (McCabe et al., 1993) using lumped models (Olufsen & Nadim, 2003) through the assumption of uniform spatial behaviours. Then, the transport phenomena and reaction engineering (Levenspiel, 1999) were developed toward investigating more behaviours of coupled heat, mass transfer and reaction processes. Meanwhile, the gap between increasing requirements of computational accuracy and low-fidelity lumped models drove researchers to study the spatial distribution of transport-reaction processes. Thus, distributed parameter models,

in the form of highly nonlinear PDEs, unclosed-form simulators and hybrids were introduced to describe the complex transport-reaction phenomena (Hasmady et al., 2009).

Nowadays, numerous experimental and computational research studies have produced plenty of high-fidelity PDEs models and multi-scale simulators, which motivates further optimisation, control and uncertainty quantification for practical transport and reaction processes. For example, the performances of cyclic adsorption processes were improved through the reduced sequential quadratic programming (SQP) algorithm for PDE-based models (Ko et al., 2005). Moreover, a model predictive control strategy was recently employed to stabilise the coupled parabolic PDEs based tubular reactor (Khatibi et al., 2021). In addition, polynomial chaos expansion (PCE) (Xiu & Karniadakis, 2002) was used to propagate the parametric uncertainty for the turbulent flow in a nozzle (Mathelin et al., 2004).

To provide more insights into distributed parameter systems, two illustrative examples are introduced with details in the next sections (1.12-1.14): a typical large-scale combustor as a deterministic example and oxidation of CO on a catalytic surface as a stochastic case.

### **1.1.2 Air combustion of NO<sub>x</sub> emissions**

This subsection introduces a common large-scale CFD example (Lang et al., 2009; Wei et al., 2012) in form of deterministic description for distributed parameter systems. The combustion process of NO<sub>x</sub> pollutants (Wei et al., 2012) is an important technological process with environmental repercussions since these pollutants damage the atmospheric layer. Complex combustion processes involve the general fluid flow, heat-mass transport and chemical reactions across the whole combustor (Chan, 1996), which can be modelled as a deterministic distributed parameter system. Figure 1.1 illustrates a 2D axisymmetric combustor with the inlet air and fuel, wall, and outlet boundary conditions. First, recycled hot combustion products are used to dilute and heat preheated air. The resulting low-concentration preheated air (a mixture of O<sub>2</sub>, N<sub>2</sub>, H<sub>2</sub>O, CO<sub>2</sub> and a few NO) would be mixed with the fuel (a mixture of CH<sub>4</sub>, C<sub>2</sub>H<sub>4</sub>, C<sub>3</sub>H<sub>8</sub>, C<sub>4</sub>H<sub>10</sub> and N<sub>2</sub>) from a nozzle to enhance the species interaction. High speed turbulent flow is generated to guarantee enough mixing efficiency. The operation temperature in the combustor need to be high enough to control the complex reaction processes, ensuring high-level thermal efficiency of reactions and less NO<sub>x</sub> emission at the outlet.

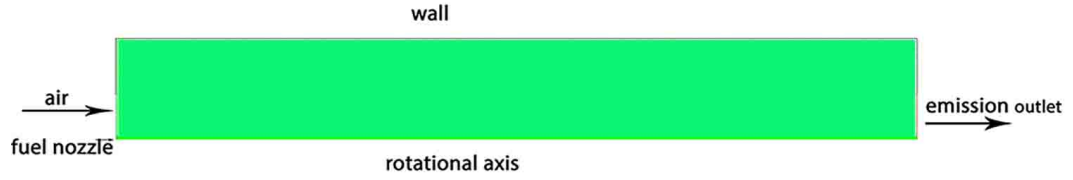
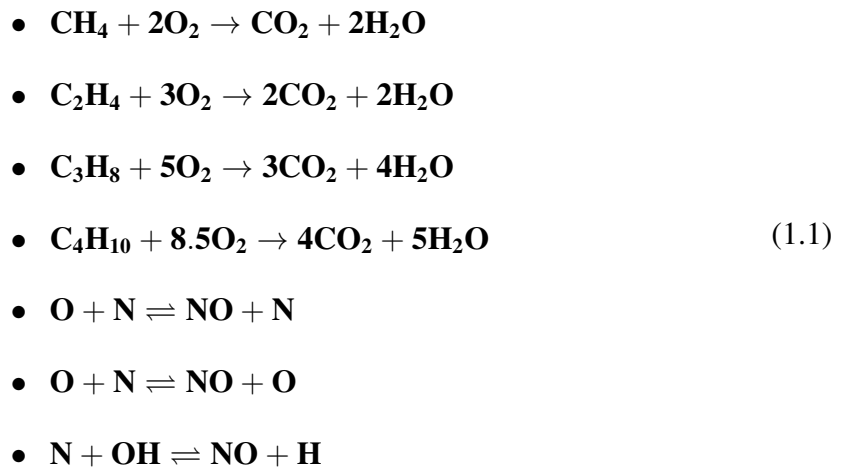


Figure 1.1: 2D axisymmetric combustor geometry

NO<sub>x</sub> emission involves thermal NO, prompt NO and N<sub>2</sub>O intermediate mechanisms. The fuel NO mechanism can be ignored if no nitrogen exists. The other reactions include oxidation of hydrocarbon. The hydrocarbon oxidation and oxynitride reactions (Srivastava et al., 2005) are as follows (Wei et al., 2012):



To capture these complex fluid dynamic, species transport and reaction phenomena, deterministic PDE-based distributed parameter models (in the form of black-box CFD code for this example (Wei et al., 2012)) can be employed, which could be further exploited to analyse the process and guide the choice of operating conditions.

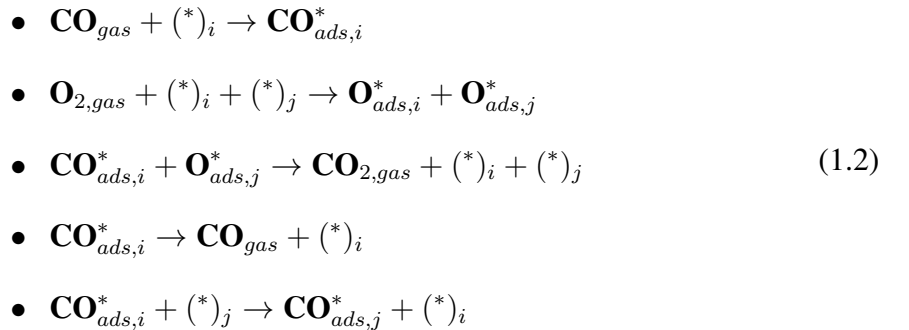
This is an illustrative deterministic distributed parameter system, where complex velocity, heat and mass behaviours across the space domain are described in a deterministic way. Many general commercial modelling software such as ANSYS Fluent (Fluent, 2015) and COMSOL (Zimmerman, 2006) describe the complex processes in the similar deterministic way, which is accurate enough to solve many engineering applications.

### 1.1.3 Oxidation of CO on a catalytic surface

In the previous example a distributed parameter system was described in a deterministic manner. Hence, the inherent process stochastics were neglected. Nevertheless, the stochastic

noises might have a significant impact in many complex phenomena, where the deterministic distributed parameter models would generate considerable errors. In these cases, distributed parameter systems need to be described in a stochastic way, putting the stochastic noises into the discrete behaviours on space domain. The stochastic distributed parameter models are extremely important when the number of participants of evolving process is relatively small. Under these circumstances, the process noises would be so obvious that only stochastic models could capture the system behaviours.

Here the heterogeneous oxidation of carbon monoxide (CO) in catalytic reaction engineering (Zissis, 2002) is taken as an illustrative example for the stochastic description, which can be modelled by multi-step coupled reactions on a rectangular lattice that represents a metal (Pt) catalytic surface. A set of dynamic behaviours occur including absorption onto the lattice active sites, reaction with nearest neighbors and desorption to the bulk gas phase. Many kinetic processes could be described using a deterministic mean field model, which can predict accurate system behaviours for high reactant concentration cases. However, there are some other applications with lowly covered reaction surfaces such as the heterogeneous oxidation on the catalytic surface. Under these conditions, lateral interactions and/or diffusion significantly affect system events, leading to strong process randomness. If deterministic models were utilised, inaccurate results would be obtained. While stochastic kinetic models can accommodate the randomness and predict the oscillation behaviours in the heterogeneous catalytic reactions across space scale. As Figure 1.2 shows, a set of coupled elementary reaction steps are given as below (Sales et al., 1982).



where  $\text{CO}_{gas}$ ,  $\text{O}_{2,gas}$ ,  $\text{CO}_{2,gas}$  are gas molecules in the bulk gas phase around the surface,  $(*)_i$  and  $(*)_j$  are the empty adsorbed sites with the index  $i$  and  $j$  while  $\text{CO}_{ads,i}^*$  and  $\text{O}_{ads,j}^*$  are adsorbed species on the surface sites  $i$  and  $j$ . The reaction mechanism involves the adsorption of gas molecules  $\text{CO}_{gas}$  and  $\text{O}_{2,gas}$  into vacant sites  $(*)_i$  and  $(*)_j$ , reactive event between

two adsorbed molecules  $\text{CO}_{ads,i}^*$  and  $\text{O}_{ads,j}^*$ , desorption of the adsorbed molecule  $\text{CO}_{ads,i}^*$  into the gas molecule  $\text{CO}_{gas}$ , and diffusion of the adsorbed molecule  $\text{CO}_{ads,i}^*$  to neighboring empty site  $(^*)_j$ . These surface events include the effects of the lateral adsorbate interactions and diffusion across the reaction surface, which could be simulated using kinetic Monte Carlo method (Zhang et al., 2019). Thus, stochastics of the adsorbate concentrations on the reaction surface could be efficiently predicted in the case of low-level gas reactants. The stochastic kinetic modelling for the oxidation reaction of CO on a catalytic surface is a typical stochastic description for distributed parameter systems, which has been popular in modelling the surface chemistry, chemical reactions and parts of multi-scale processes (Zhang et al., 2019). Similar stochastic distributed parameter models could be exploited to describe other distributed parameter systems with obvious stochastics.

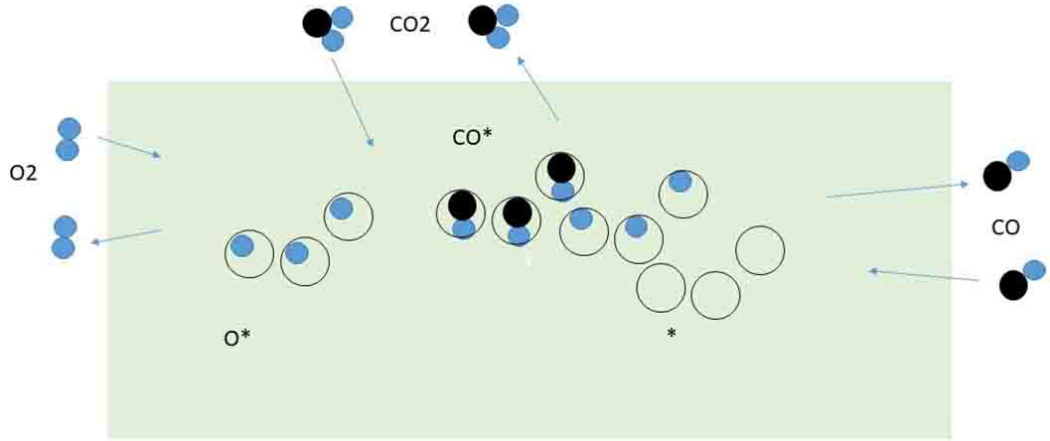


Figure 1.2: Stochastic kinetic reactions on the catalytic surface

## 1.2 Challenges for optimisation, control and uncertainty quantification of large-scale distributed parameter systems

Nowadays, modelling techniques are more and more advanced for complex process systems, since a wide range of experimental and computational studies have contributed to numerous high-fidelity, large-scale and multi-scale distributed parameter models. Moreover, advanced computational strategies have led to the development of commercial modelling software, such as ANSYS Fluent (Fluent, 2015), COMSOL (Zimmerman, 2006) and even open-source software (eg. (Jasak et al., 2007)). The combination of such efficient simulators with high performance computational facilities accelerates the development of large-scale process

digitalisation. Nevertheless, further optimisation, control and uncertainty quantification of such processes are still challenging.

Firstly, both high dimensionality from discretisation and inherent non-linearity require dramatic computational cost as a compromise. Derived from complex spatiotemporal events, high dimensionality contributes to a huge number of dependent variables from discretisation, resulting in expensive system evaluations and scalably increasing computation costs on upper-level tasks. Moreover, complex thermodynamic and reaction mechanisms, coupled multi-physical and multi-scale phenomena exhibit highly nonlinear behaviours, modelled by large-scale non-convex equations. Non-convexity could generate multiple local optima, leading to great difficulties for existing optimisation solvers to identify the best (global) optima. Popular stochastic search approaches require a large number of system evaluations to skip local optima and find better optimisation solutions, which would cause intensive computations for evaluating large-scale systems. Although stochastic approaches can often find good solutions, they could not guarantee the global optimality conditions, even the local optimal conditions. Deterministic branch and bound (BB) algorithms can provide a rigorous guarantee on global optimality conditions. However, almost all rigorous BB algorithms require intractable number of system evaluations to compute the low-upper bounds, which would result in huge computational burdens for large-scale problems. Secondly, the mature commercial modelling software are often black-box to users, providing no optimisation and control options. Black-box characteristics imply inability to use the highly efficient equation-based algorithms, putting more barriers on optimal designs. For cases where legacy codes or open-source codes are available, but revisiting the large volume of code and utilising them for upper-level optimisation and control tasks are still time-consuming works.

Moreover, complex process systems often exhibit spatiotemporal behaviours, along with inherent stochastics. Together with incomplete system knowledge, process uncertainty can result in the mismatch between models and true processes. Uncertainty quantification approaches are helpful to make more robust decisions. However, traditional sample-based uncertainty quantification techniques require intensive computational costs.

Computational challenges are the motivations behind this work to develop more efficient optimisation, control and uncertainty quantification algorithms for large-scale and multi-scale distributed parameter systems. Model reduction techniques are promising tools to satisfy the

requirements of intensive computations. Here, this thesis aims to build high-performance model reduction based computational algorithms for the purpose of optimisation, control and uncertainty quantification.

As for the black-box characteristics and expensive system evaluation issues to global optimisation algorithms, sampling based projective order reduction techniques combining neural network surrogate models could construct simple but accurate explicit models to deterministic global optimisation algorithms. Thus, reduced surrogate model based near global solutions may be computed efficiently. Uncertainty is one of the most challenging difficulties to intelligently control in practical large-scale systems. Model predictive control could deal with process constraints and introduce feedback to update the state(s) of dynamic models and eliminate some degree of uncertainties such as disturbance noise and state estimation errors. Significant parametric uncertainty of process models would be addressed using popular polynomial chaos expansion. However, model unavailability and expensive system evaluation issues would still prevent the (global) optimisation within the recursive control procedures. The previous reduced surrogate model based approximate global optimisation approach could be employed to overcome these problems. Thus a robust model predictive control strategy could be built for large-scale systems under uncertainty.

Although polynomial chaos expansion could efficiently deal with parametric uncertainty, it requires a prior knowledge about the type of uncertainty distributions and may be not accurate enough for complex systems. While traditional sample-based uncertainty quantification algorithms such as Monte Carlo algorithm and its extensions could cope with the general uncertainty distributions of complex systems, but they are extremely expensive for large-scale systems. The "Equation-free" approach, a projective model reduction technique, could significantly accelerate system computations through black-box dynamic systems. A combination of Monte Carlo methods and the "Equation-free" technique may efficiently perform uncertainty quantification tasks in terms of computational cost and accuracy.

Even with efficient uncertainty quantification algorithms, large-scale system under uncertainty is still expensive. Globally optimising these expensive systems is challenging. Accurate, reduced surrogate model based global optimisation needs a large number of samples to construct simple but accurate models, leading to huge computations for the cases with more independent variables. The common local optimisation algorithms with multiple start

points may be a possible choice. However, derivatives require expensive system evaluations. Furthermore, large-scale systems under uncertainty may exploit strong systems noises, resulting in unreliable derivatives and tortuous optimisation processes. Bayesian optimisation approach may globally find a good optimisation solution for large-scale systems under uncertainty with the limited number of system evaluations, avoiding the derivatives. The Bayesian robust optimisation approach, together with the previous "Equation-free" Monte Carlo algorithm, may efficiently deal with large-scale system under uncertainty.

### **1.3 Research contributions and thesis structure**

Motivated by the high demand on efficient optimisation, control and uncertainty quantification algorithms for distributed parameter systems, a number of scientific contributions are presented in the upcoming sections of this thesis. These contributions are presented in a 'journal format' as a series of academic papers published or submitted for publication in peer reviewed scientific journals.

#### **1.3.1 Contributions**

1. This work provides a double model reduction based global optimisation framework for steady state distributed parameter systems. Initially, a combination of principle component analysis (PCA) and artificial neural networks (ANNs) is employed to construct accurate reduction models for large-scale steady-state systems. In order to further improve the computational efficiency, two reformulation approaches are used for the reduction models. One way is to use piecewise linear affine (PWA) to approximate the high nonlinear hyperbolic tangent functions in artificial neural networks. Another way is to reformulate the ReLU based deep ANNs into MILP problems. Both of the reformulations could significantly improve the computational speed.

M. Tao, P. Petsagkourakis, J. Li, C. Theodoropoulos,

Model reduction based global optimisation for large-scale steady state nonlinear systems for Distributed Parameter Systems,

Submitted to AICHE J, Under Review

2. Next, this contribution still utilises model reduction techniques but considers more complex dynamic control with parametric uncertainty. Firstly, polynomial chaos expansions (PCE) are introduced to propagate the parametric uncertainty onto quantities of interest. Then model construction techniques, proper orthogonal decomposition (POD) and recurrent neural networks (RNNs), are employed to capture the high-dimensional dynamics of statistics of the computed quantities of interest, using a systematic data collection from the original simulator. Finally, the resulting dynamic models for the statistic moments/probabilistic bounds would be adopted within the model predictive control (MPC) framework for robust control of distributed parameter systems.

M. Tao, I. Zacharopoulos, C. Theodoropoulos,

Robust model predictive control for large-scale distributed parameter systems under uncertainty,

To be submitted.

3. This work provides a novel equation-free Monte Carlo uncertainty propagation methodology for large-scale steady-state distributed parameter systems. The key idea is to employ equation-free techniques to accelerate the computations of a dynamic simulator to steady states. Compared with popular polynomial chaos expansion, the computational accuracy could be improved while the computational speed is faster than the traditional Monte Carlo method. The performance of the proposed equation-free uncertainty propagation algorithms are demonstrated through an illustrative tubular reactor, the typical FitzHugh-Naguma model and a stochastic catalytic surface reaction case.

M. Tao, C. Theodoropoulos,

Uncertainty quantification for distributed parameter systems: deterministic and stochastic cases,

To be submitted.

4. Next, this contribution presents a robust optimisation framework using our previous uncertainty propagation algorithms. The equation-free uncertainty quantification method would be employed to efficiently address the parametric uncertainty. However, large-scale systems under uncertainty are likely to exploit unpredictable noise, which prevent the usage

of traditional gradient based local search methods. Here, a Kriging model based Bayesian optimisation strategy is adopted to globally find the optima of large-scale distributed parameter systems under uncertainty.

M. Tao, C. Theodoropoulos,

Robust steady-state optimisation for large-scale distributed parameter systems with uncertainty,

To be submitted.

The journal format is adopted in this thesis, where Chapter 2 gives the literature review and Chapters 3 - 6 include: 1) a brief introduction to the contributions of authors, 2) the paper written in a journal format attached with any possible supplementary information. Finally, both the conclusions of this finished project and incoming future work are summarised in the last chapter.

### **1.3.2 Conference presentations**

1. 'Reduced model-based global optimisation of large-scale steady state nonlinear systems', Oral Presentation, ESCAPE29,2019, Eindhoven.

2. 'Model Reduction-Based Global Optimisation for Large-Scale Steady State Nonlinear Systems', Oral Presentation, 2019 AIChE Annual Meeting2019, 2019, Orlando.

3. 'Reduced model based Uncertainty Quantification method of Distributed Large-scale Systems', Oral Presentation, ESCAPE30,2020, Milan. (online)

4. 'Robust model predictive control of large-scale distributed parameter systems under uncertainty', Oral Presentation, 2021 AIChE Annual Meeting2021,2021, Boston. (online)

### **1.3.3 Conference Proceedings**

1. Tao, M., Li, J. and Theodoropoulos, C., 2019. Reduced model-based global optimisation of large-scale steady state nonlinear systems. In Computer Aided Chemical Engineering (Vol. 46, pp. 1039-1044). Elsevier.

2. Tao, M. and Theodoropoulos, C., 2020. Uncertainty Analysis and Model Reduction Based Global Optimisation of Distributed Large-scale Systems. In *Computer Aided Chemical Engineering* (Vol. 48, pp. 1975-1980). Elsevier.

# Chapter 2

## Literature review

Many practical engineering and science problems can be formulated into optimisation problems, made of design variable (s), objective(s) and constraint(s) (Biegler et al., 1997; R. Smith, 2005). Optimisation aims to search the extreme points of an objective function under a number of constraints in a domain of design variables. Specifically in chemical and biochemical engineering, the objective functions are related to obtaining optimal conditions for process economic and operation performance under the constraints of physical laws and safety requirements. The general optimisation problem can be formulated as follows (Edger & Himmelblau, 2001; Floudas, 1995):

$$\begin{aligned} \min_{\mathbf{d}} \quad & G(\mathbf{d}) \\ \text{s.t.} \quad & \\ & f(\mathbf{d}) = 0 \\ & g(\mathbf{d}) \leq 0 \\ & \mathbf{d}^L \leq \mathbf{d} \leq \mathbf{d}^U \end{aligned} \tag{2.1}$$

where  $G : \mathbb{R}^{N_d} \rightarrow \mathbb{R}$  is the objective function,  $f : \mathbb{R}^{N_d} \rightarrow \mathbb{R}$  are the equality constraints while  $g : \mathbb{R}^{N_d} \rightarrow \mathbb{R}$  are the inequality constraints.  $\mathbf{d} \in \mathbb{R}^{N_d}$  are the design variables within the low and upper bounds of  $[\mathbf{d}^L, \mathbf{d}^U]$ . The properties of the design variables  $\mathbf{d}$  decide the type of mathematical programming to be used. If design variables  $\mathbf{d}$  contain integer variables, the above formulation belongs to mixed integer programming (MIP) problems. General MIP problems are NP-hard problems for global optimisation, needing efficient mathematical techniques such as branch and bound algorithms. If design variables  $\mathbf{d}$  include continuous variables within nonlinear terms, the above problems are classified as nonlinear

programming (NLP) ones. Furthermore, the related nonlinear parts are often non-convex for large-scale practical problems, which would also cause the difficulty of identifying the global optimal solution for most existing local optimisation algorithms. If no integer variable or nonlinear term exists in the optimisation formulation, the above problems are simply linear programming (LP). Conversely, the problems are the complex mixed integer nonlinear programming (MINLP) if the above formulation involves both integer variables and nonlinear terms. Other important classifications include steady state and dynamic optimisation problems, where steady state optimisation problems are formulated using steady state system models while dynamic optimisation problems would involve optimisation on the transition behaviours of dynamic systems. In addition, more details about dynamic optimisation could be seen in the subsequent optimal control parts.

This work focuses on NLP-based global optimisation of steady state behaviours, optimisation-based model predictive control strategies of dynamic behaviours and uncertainty quantification of steady state behaviours, especially for partial differential equations (PDEs) based distributed parameter systems. Finite approximation transforms the original steady-state PDEs into large-scale algebraic equations while large-scale ODEs would be generated from dynamic PDEs. Both of large-scale algebraic equations and ODEs could result in expensive system evaluations, scalably increasing the computations on upper-level optimisation, control and uncertainty quantification tasks.

In this Chapter, a comprehensive review and discussion is given about optimisation, control and uncertainty quantification technologies of large-scale distributed parameter systems. Section 2.1 presents an overview of general optimisation and control methods, then Section 2.2 discusses uncertainty quantification methods. Next, Section 2.3 describes the robust/stochastic optimization and control approaches under uncertainty. Then general model reduction techniques are provided that can deal with large-scale computations. Finally, a summary is presented for optimisation and control of large-scale distributed parameter systems under uncertainty.

## 2.1 Optimisation and control

The above large-scale nonlinear optimisation problem Eq.(2.1) can be solved efficiently by advanced model-based computational methodologies both for local and global optimisation.

Popular local optimisation approaches (Nocedal & Wright, 2006) for large-scale constrained optimization are generally gradient-based search ways, containing sequential quadratic programming (SQP) (Boggs & Tolle, 1989), penalty methods (Di Pillo & Grippo, 1989) and interior point (Wright, 1997) methodologies. The idea of SQP is to solve a set of quadratic programmings (QP) to approximate the Karush-Kuhn-Tucker (KKT) first necessary optimal conditions (Gordon & Tibshirani, 2012). Exploiting Newton-like methods, SQP optimization methods can achieve fast convergence rates (Boggs & Tolle, 1989). One of the successful SQP variants is reduced space SQP (rSQP) or reduced Hessian method (Byrd & Nocedal, 1990). rSQP method was initially designed for large-scale equality-constrained optimization problems with a small number of degrees of freedom. The computational cost could be significantly reduced through projecting the iteration space onto a small subspace of the degrees of freedom.

Penalty methods aim to transform the original constrained optimisation problems into unconstrained ones. A penalty multiplier is introduced to construct auxiliary functions (Bertsekas, 1976). In this way, complex constrained optimisation issues could be dealt with by advanced unconstrained optimisation algorithms (McKeown et al., 1990). However, new issues such as ill-conditioned models can arise (Dussault, 1995). Sequential penalty algorithms were developed to overcome this issue (Gould, 1989). Interior point methods perform similar reformulation tasks, dropping the large-scale constraints and adding more terms to the objective functions (Wright, 1997). Sequential decreasing barrier parameters are adopted to force the computations toward the optimal points. Highly efficient interior point methods contribute to the open-source software IPOPT (Wächter et al., 2002).

Local search optimisation methods have made great progress in both theoretical research and engineering practices, however they can be easily stuck in local optima, especially for high nonlinear and non-convex problems. Global optimization strategies have been developed to overcome this barrier, including stochastic search methods (Spall, 2005) and deterministic global optimisation techniques (Floudas, 2013).

Stochastic search methods such as simulated annealing (Van Laarhoven & Aarts, 1987), genetic algorithms (Holland, 1992) and particle swarm optimization (Kennedy & Eberhart, 1995), can globally explore the feasible solution space avoiding local optima. Simulated annealing (SA) approach borrows heating and cooling rules from material annealing processes. Motivated by the metropolis algorithm for simulating the annealing of solids, SA algorithm was developed for combinatorial optimization problems (Xu et al., 2007). SA-based algorithms have been applied widely in engineering and science areas including process network design (Dolan et al., 1989), electronic energy (Tsoo et al., 1990) and job scheduling (Leite & Topping, 1999). The stochastic approach is more likely to skip the neighborhood of local optima, and globally search the feasible space to reach good solutions (Chibante, 2010). Inspired by evolutionary rules, such as inheritance, selection, crossover and mutation from the biological gene evolution (Vose, 1999), genetic algorithms (GA) perform heuristic searches for the optimisation problems such as assembly sequences (Marian et al., 2006), rolling system design (Oduguwa & Roy, 2002) and parameter estimation (Q. Wang, 1997). Their main advantages are that they are inherently stochastic, easily parallelizable, and suitable for multi-modal problems (Mühlenbein et al., 1991). Another popular stochastic search optimization algorithm is particle swarm optimisation (PSO) (Mirjalili, 2019), which is motivated by the cooperative social behaviours of animals such as birds and bees (Krink et al., 2002). Through introducing the concepts of momentum, particle' position, velocity and current best solution, PSO search can accelerate the global search toward the minima (Lazinica, 2009). In (Li et al., 2015), an improved nonlinear dynamic adaptive PSO algorithm has been developed for producing energy-aware clusters with selection of optimal cluster heads, where a new cluster head competition mechanism was introduced. The computational results showed that the proposed novel algorithm could significantly increase the network lifespan and reduce the energy consumption. More successful engineering and science application cases can be seen in the literature including distillation column (Al-Dunainawi & Abbod, 2016), predictive surface roughness (Noor et al., 2011) and heat exchange network (Silva et al., 2009). Such stochastic search algorithms have been used to solve numerous optimisation problems. However, these algorithms need a lot of computational evaluations, which lead to really slow computational speed for large-scale systems. Moreover, these stochastic approaches can often find good optimisation solutions, but offer no theoretical guarantees on the global optimality of the computed solutions.

Deterministic global optimization methods are capable of computing global optima utilising branch-and-bound techniques (Lawler & Wood, 1966; Mitten, 1970). Rigorous deterministic branch-and-bound global optimisation algorithms have been studied for small and/or medium scale problems. In the paper (Adjiman et al., 1998), the  $\alpha$ -based branch and bound ( $\alpha$ BB) global optimisation algorithm was first proposed based on the rigorous convex underestimator (Adjiman & Floudas, 1996). The key idea is to construct the concave/convex relaxations for the general twice-differentiable non-convex terms in the original problems to iteratively compute the convergent upper-lower bounds for the global optimum. The  $\alpha$ BB algorithm offers a rigorous theoretical guarantee on the global optimum for the large class of twice-differentiable NLPs. Later,  $\alpha$ BB algorithms were successfully applied in commercial global optimisation software such as ANTIGONE (Misener & Floudas, 2014). Extensions of  $\alpha$ BB algorithms were also studied. For example, an error-in-variables approach was developed and global optimisation algorithms were proposed for parameter estimation and data reconciliation of differential-algebraic models using the principles of the  $\alpha$ BB relaxation (Esposito & Floudas, 1998).

In the literature (Tawarmalani & Sahinidis, 2005), a polyhedral branch-and-cut approach was presented to construct tight nonlinear convex relaxations in a global optimisation framework. Advanced automatic identification and polyhedral cutting plane methods improved the computational efficiency by several orders of magnitude. The branch-and-cut algorithm contributes to the state-of-art global optimisation solver BARON (Sahinidis, 1996). Its strong capability of computing global optima contributes a lot to engineering and science applications such as heat exchange networks (S. Y. Kim & Bagajewicz, 2017) and gas network operation (Puranik et al., 2016), and even mathematical computational methods. For instance, global optimisation using BARON was employed to construct new computational formulations of classical Runge-Kutta methods (Ketcheson et al., 2009; Ruuth, 2006). In (Mitsos et al., 2009), a McCormick-based relaxation algorithm was proposed to compute the lower and upper bounds of implicit functions, which was then put in a spatial branch and bound (SBB) framework to solve a simple ODE-based dynamic optimisation problem. In (Najman & Mitsos, 2019), convex/concave relaxations of implicit functions were constructed using tighter McCormick-based relaxations, employing sub-gradient propagation techniques to compute the simple affine underestimators and overestimators of factorable functions. The tighter re-

laxation could significantly improve the tightness of lower-upper bound gaps and decrease the computational time in a reduced space branch and bound framework. In the paper (Hasan, 2018), an edge-concave underestimator or the linear facets of a convex envelope was used to compute lower bounds in the SBB framework through the preconditioning of bounds of the Hessian matrix. Then this novel computational approach for deterministic global optima was applied to a challenging optimisation problem with an embedded system of ordinary differential equations (Bajaj & Hasan, 2020). The time dependent bounds on the state variables and diagonal elements of the Hessian matrix were computed through solving an auxiliary set of ordinary differential equations using the formulation of differential inequalities, which provides an efficient computational approach for low bounds in the branch and bound framework. A more comprehensive review of deterministic global optimisation algorithms and their applications can be seen in (Floudas & Gounaris, 2009). However, most of these research works are limited to small-scale application problems. The above algorithms are computationally intensive for large-scale systems due to the requirements for multiple evaluations of the lower bounds of the optimisation problems (Houska & Chachuat, 2019). Moreover, most large-scale optimisation problems, especially PDE-based optimisation problems are black or grey box, which are characterized by a total or partial lack of explicit mathematical equations describing the constraints and/or the objective of the problem (Boukouvala & Floudas, 2017). Black/grey box problems cover various applications which involve expensive simulations, input-output data, legacy codes or phenomena without physics-based equations (Boukouvala & Floudas, 2017). Closed-form mathematical equations are unavailable for these black/grey box problems, implying that direct model-based techniques such as branch and bound algorithms are not applicable.

Surrogate models provide an alternative tool to account for the unknown parts of black or grey box systems. In (Meyer et al., 2002), smart sampling was employed to construct an explicit interpolation blending function as a surrogate model in a deterministic global optimization algorithm. The designed interpolants provide explicit function formulations and allow valid convex/concave estimators, which ensures the  $\epsilon$ -global optimality for the surrogate problems. In (Davis & Ierapetritou, 2009), a Kriging-based branch and bound framework was proposed for mixed-integer nonlinear programming (MINLP), where a branch and bound computation route was utilised to deal with integers while surrogate Kriging models

were employed for sub-NLP problems with black box parts and process noise. The proposed algorithm could efficiently solve small process synthesis examples in terms of number of function evaluations and computational time to find the global optimum. In (S. H. Kim & Boukouvala, 2020), the one-hot encoding technique was introduced to deal with mixed-variable problems, where neural networks and Gaussian process were used as surrogate models. The computational results showed that one-hot encoding could lead to more accurate and robust mixed-variable surrogate models for surrogate-based optimisation compared with the traditional relaxation based surrogate approach. In (Boukouvala et al., 2017), a novel global optimization framework was presented for general grey-box constrained problems, especially for PDE-based problems. A large number of faster low-fidelity models were employed to select representative samples. Interpolating and non-interpolating functions, including quadratic, signomial and Kriging models, were used to formulate the whole constrained problems, which would be globally optimised through deterministic optimization solvers.

Artificial neural networks (ANN) provide one of the most efficient ways to construct surrogate models, due to both successful practices and proven theoretical supports that a feed-forward neural network with one single layer is sufficient to represent any smooth function (Hornik et al., 1989). However, global optimisation of ANN models is challenging due to their complex structures. In (J. D. Smith et al., 2013), small-scale ANN models (1 hidden layer, 3 neurons) were constructed and optimized globally by the advanced global solver BARON (Tawarmalani & Sahinidis, 2005). In (Schweidtmann & Mitsos, 2019), concave/convex envelopes were first constructed for the highly non-convex tangent activation function, which were then embedded in a reduced-space spatial branch and bound framework. The reduced space computational framework projects the iteration space of non-convex variables onto the subspace of dependent variables, resulting in small-size sub-problems and, consequently, in significant computational savings. Then the computational framework was successfully applied to globally optimise ANN-based grey box models (Schweidtmann et al., 2019). In (Grimstad & Andersson, 2019), advances in the machine learning community, including deep neural network structures and low-complex ReLU activation functions, were adopted to formulate the deep rectifier neural network into a mixed integer linear programming (MILP) problem, which was solved by a fast CPLEX optimiser. In (Keßler et al., 2019), a small-size Kriging model was globally optimised by BARON. In (Schweidtmann et al., 2021),

concave/convex envelopes of common covariance functions and tight relaxations of an acquisition functions were presented, which were then used in a reduced space optimisation framework. The reduced space computational framework achieved significant computation reduction for global optimisation.

Recent developments in data science, have made more progress on the research about global optimisation of hybrid data-driven models or grey-box models. In (Boukouvala & Floudas, 2017), a general framework for the global optimisation of non-linear constrained grey-box problems was proposed. Variable selection, bounds tightening and constrained sampling techniques were integrated to reconstruct accurate surrogate models for unknown parts of systems, which were then globally optimised using advanced global solvers. The computational results show the competitive performance of the methodology compared to existing derivative-free optimization algorithms. In (Demirhan et al., 2020), an integrated data-driven modeling and global optimization-based multi-period nonlinear framework was presented for real-life plants, which can significantly improve their performance.

Optimisation techniques have successfully been combined with control. Optimal control seeks the best control policy for some performance criterion (minimize cost or maximize benefits) with given constraints such as physical process laws. For example, proper control policy is needed to adjust the dynamic paths of batch or semi-batch processes related to bio-production to reach high conversions and satisfy safety and process constraints. The general optimal control problem can be defined as follows (Vinter & Vinter, 2010):

$$\begin{aligned}
& \min_{\lambda(t)} \int_{t_0}^{t_H} \sigma(\mathbf{y}(t_H)) + F_1(\mathbf{z}(t), \mathbf{y}(t), \boldsymbol{\lambda}(t), t) dt \\
& s.t. \\
& \dot{\mathbf{y}}(t) = f_1(\mathbf{y}, \boldsymbol{\lambda}, t) \\
& \mathbf{z} = g_1(\mathbf{y}, \boldsymbol{\lambda}, t) \\
& g_2(\mathbf{z}, \mathbf{y}, \boldsymbol{\lambda}, t) \leq 0 \\
& \mathbf{y}(t_0) = \mathbf{y}_0 \\
& t \in [t_0, t_H]
\end{aligned} \tag{2.2}$$

where  $\boldsymbol{\lambda}(t)$  denotes the control inputs,  $\mathbf{y}(t)$  the state variables and  $\mathbf{z}(t)$  the measurement variables.  $\sigma$  is the terminate part of the objective cost and  $F_1$  is the other cost. The constraints

include the system equations  $f_1$ , the state estimation  $g_1$  and others inequality constraints  $g_2$  such as bound constraints for state, manipulated and measurement variables. The initial state  $y_0$  and the operation time period  $[t_0, t_H]$  are also given.

One of the most successful practices to solve the above optimal control problems is the maximum principle, developed by Pontryagin and his coworkers (Pontryagin, 1987). This principle could be easily seen as an extension of the calculus of variations (Kamien & Schwartz, 2012) for the above problems Eq.(2.2), which is achieved through the additional necessary conditions for optimality of a Hamiltonian equation. A wide range of methodologies have been exploited to solve the optimal control problem (OCPs) as dynamic optimisation formulations. The general approaches include the single shooting method (Vassiliadis et al., 1994), orthogonal collocation (Biegler, 1984) and multiple shooting approaches (Morrison et al., 1962). All the three methods aim to transform the original dynamic optimisation in infinite space into NLP problems. These methods have been widely applied in commercial software such as gPROMs (Furlonge et al., 1999). More details can be found in the book (Biegler, 2010).

In practice, uncertainty is present including model-plant mismatch and process disturbance, which can make the system operate sub-optimally or worse. Feedback control algorithms provide efficient tools to deal with it, such as model predictive control (Garcia et al., 1989) and re-optimisation strategies (Xiong & Zhang, 2005). Model predictive control (MPC) employs the feedback as a compensate of process uncertainty and solves a set of dynamic optimisation problems (Rawlings et al., 2017). MPC becomes more and more popular in industry since the availability of highly accurate process models.

Most existing optimal control approaches only focus on satisfying the local optimal conditions. However, global optima issues could arise due to the highly non-convex process models within nonlinear model predictive control (NMPC). Additionally, local solutions may result in unexpected system responses or failure of the control policy for the purpose of NMPC. Stochastic approaches aim to obtain global optimal control of high complex systems but cannot provide guarantees on the rigorous global optimality of the computational results. Fortunately, deterministic global optimal control methods can provide a promising solution.

In (Long et al., (2004,2006)), a globally optimal nonlinear model predictive control algo-

rithm was presented for small scale nonlinear dynamic systems. A combination of convex relaxation and branch and bound framework was employed to compute the global solutions of real-time optimisation. The deterministic global algorithm was shown to significantly improve the poor performance caused by the suboptimal control policy computed by a local solver based controller.

In (Čižniar et al., 2008), a constrained global nonlinear predictive control algorithm was designed. The differential-algebraic equation was discretised into NLP problems, which was then utilised by a traditional spatial branch and bound computational framework. The computational algorithm could find the global optimum for the closed-loop control of small dynamic systems and improve the control performance.

In (Wang et al., 2017), a global nonlinear model predictive control algorithm was exploited for highly nonlinear processes with multiple operating conditions. The normalized multi-parametric disaggregation technique (Teles et al., 2013) was utilised to compute the upper and lower bounds in a spatial branch and bound global optimisation framework. In addition, an optimisation-based bound tightening technique was adopted to speed up the expensive computations and improve the computational robustness. Controller performance was improved in several small dynamic systems through the proposed global nonlinear control scheme, compared with a local optimisation solver-based control policy.

In (Caspari et al., 2019), a deterministic global optimisation scheme was put into an economic nonlinear model predictive controller of a flexible air separation process. The performance of the closed-loop controller using the global optimisation solver BARON was similar to the one using a local optimisation solver although BARON was much more computationally demanding.

Meanwhile, surrogate model-based global nonlinear predictive control strategies provide alternatives to improve the performance of systems' behaviour. In (Degachi et al., 2015), the original system model was replaced by surrogate polynomial functions, where the polynomial coefficients were computed through neural networks. Then the global optimisation method for geometric programming was employed in recursive control loops. The proposed global NMPC method efficiently achieved point tracking and disturbance rejection with acceptable computation burden.

In (Lu et al., 2020), Bayesian optimization techniques were employed for an NMPC control scheme. A Kriging surrogate model was utilised to build the input/output black-box relationship between the closed-loop controller performance and the tuning parameters. The exploring and exploiting strategies could compute the optimum tuning parameter values with acceptable computational costs.

In (Doncevic et al., 2020), recurrent neural networks (RNNs) were adopted to construct highly accurate surrogates for the complex dynamic systems. Then a reduced-space formulation was built for the RNN based NMPC controller. The resulting optimisation problems were solved by the global optimisation solver. The global nonlinear model predictive control could efficiently avoid multiple local solutions. To further reduce the computational costs, two alternatives were adopted: One was to exploit additional state variables as decision variables to the global optimizer while the other one was to replace the traditional RNN surrogate model with a neural multi-models. The improvements could achieve computation reduction by one order of magnitude.

## **2.2 Uncertainty quantification**

In complex process systems, uncertainty can arise due to a wide range of reasons (Sullivan, 2015). One such factor may be the inherent uncertainty of the system, which may be irreducible aleatoric uncertainty, including measurement noise and varying quantum effects. Aleatoric uncertainty cannot be avoided, which means that computational designs should take account of it. Another classification can be epistemic uncertainty, which is caused by insufficient knowledge about the processes, such as unknown model parameters. This uncertainty may be reduced through more computational and/or experimental research (Kramer, 1999). Both aleatoric and epistemic uncertainties can result in mismatch between model-based simulation and experiment-based observations. In particular, uncertainty can greatly impact computational results and may lead to worse decision-making for industrial production (Chambers & Quiggin, 2000). In order to make smart decisions for industrial processes, efficient uncertainty quantification methods should be used to account for the impact of uncertainty, which can speed up expensive robust optimization strategies for large-scale systems with uncertainty.

Uncertainty quantification aims to measure the impact of uncertainty on quantities of interest (Ghanem et al., 2017). Considering a simple situation where models of processes are available, explicit analytical solutions may be obtained. However, process models are mostly black-box for large-scale practical cases. Even if they are open to users, high complexity of models makes analytical solutions almost impossible to obtain. This way, uncertainty propagation techniques are more practical by using black-box system models for a large number of realisations, approximating the distribution of uncertainty. The large volume of propagated results can be then used to calculate distributions of quantities of interest. Typically, numerous samples from standard Monte Carlo methods can accurately represent distributions of uncertainty, leading to highly accurate propagated results. However, so many sample based evaluations of large-scale systems would be computationally intensive (Nagy & Braatz, 2007), especially for online optimisation and control (Bansal et al., 2000). These computational challenges drive the development of fast-computing methods.

One alternative approach is to exploit efficient sampling ideas such as the Latin hypercube sampling method (Florian, 1992), reducing the number of sample points while preserving computational accuracy. In (Dunn et al., 2011), turbulence model coefficients and parameters in turbulent flows were obtained through experimental investigation subject to measurement uncertainty, which was then quantified using a fast Latin hypercube sampling method. Computational evaluations of the coefficient uncertainty significantly improve the robustness of fluid flow.

To reduce the variance from the standard Monte Carlo estimator, Multilevel Monte Carlo (MLMC) method (Giles, 2008), an efficient sampling approach, utilizes discretisation with different step values, leading to much fewer samples compared to a fine level of discretisation. In (Icardi et al., 2016), macroscopic effective parameters such as permeability, effective diffusivity and hydrodynamic dispersion suffer significant errors and uncertainties in pore-scale and Digital Rock Physics (DRP) problems. The MLMC method was utilised to reduce the computational cost needed for computing accurate statistics of effective parameters and other quantities of interest, which was the first practice of uncertainty quantification in pore-scale physics and simulation. Moreover, MLMC method has been used as an efficient uncertainty quantification tool across surfactant/polymer enhanced-oil-recovery (Alkhatib & Babaei, 2016), wastewater treatment and distillation (Kimaev & Ricardez-Sandoval, 2018),

and biochemical engineering (D. F. Anderson & Higham, 2012) applications.

An alternative approach is to construct efficient low-order closed-form models to replace the expensive system models for the Monte Carlo method, including power series expansion (PSE) and polynomial chaos expansion (PCE). Power series expansion, as a perturbation method, utilises sensitivity information to build the relationship between output observations and uncertain parameters. The low-order sensitivity information only needs a small number of system simulations. PSE has been widely employed as a popular uncertainty quantification method. In (Ma & Braatz, 2001), the coefficients of growth rate in a multidimensional crystallisation process suffered irreducible uncertainty, which was accounted for by the PSE method to compute the worst-case scenario. The PSE-based worst-case computation significantly enhanced the robust analysis of the batch multidimensional crystallization process. In (Chaffart et al., 2016), uncertainty in kinetic parameters of spatially heterogeneous multi-scale catalytic reaction system were considered, where PSE method was applied to efficiently propagate parametric uncertainty throughout the large-scale systems. PSE-based uncertainty analysis unveiled the substantial effect of uncertainty in reaction rates on the reactor performance, which contributed to the robust operations of the catalytic reaction systems. However, the sensitivities of PSE-based uncertainty quantification are prone to the size of the perturbation step and usually need costly high-order sensitivities to preserve computational accuracy for strongly nonlinear processes (Nagy & Braatz, 2007).

Polynomial chaos expansion, firstly introduced by (Wiener, 1938), represents the observable variables with a series of polynomials in uncertain parameters, and then approximates the exact representation using an expansion of finite order (Eldred, 2009). Low-order PCE models, with the coefficients of polynomials calculated through the collocation method and Galerkin projection, can efficiently be utilised as an uncertainty quantification tool in process industry (Chaffart & Ricardez-Sandoval, 2018). In (Chaffart & Ricardez-Sandoval, 2017), distributional uncertainty was propagated through the multi-scale reactor model using the PCE method. Efficient PCE-based uncertainty propagation techniques provide advantageous preconditions for further robust dynamic optimisation. In (Bradford & Imsland, 2019), PCE method was employed to propagate parametric uncertainty considering the additive system disturbances. The PCE-based control strategies successfully improved the economic benefits of a semi-batch polymerization process. However, PCE requires prior knowledge of the

distributions of parametric uncertainty, because “suitable” orthogonal polynomials need to be chosen corresponding to the uncertainty distribution, which significantly affects computational accuracy (Xiu, 2010).

### **2.3 Optimisation and control under uncertainty**

The previous subsection introduced uncertainties in complex systems and uncertainty quantification techniques, which provide essential tools for further optimisation and control under uncertainty. The aim of optimisation and control of uncertain systems is to take account of substantial effects due to uncertainty and make smart decisions for process systems.

There are two approaches to deal with optimisation and control under uncertainty. One is robust optimisation and control. Here the word “robust” means the optimisation and control procedure would be made for the performance of worst-case system uncertainties. Although the robust approach can rigorously satisfy system constraint requirements, the resulting optimisation decisions and control laws are conservative for many practical applications, as the worst-case uncertainties usually have a really low probability of occurrence (Bertsimas et al., 2011; Bhattacharyya, 2017). An alternative approach is stochastic optimisation and control under uncertainty (Lewis et al., 2017). The stochastic approach does not need to take account of all uncertain scenarios, where only some randomised finite sampling based cases are considered, hence exploiting the statistic description of uncertainties. In stochastic schemes, less conservative chance requirements are utilized to replace the worst-case constraints, which may significantly improve the performance of systems and reduce the computational costs. Moreover, efficient uncertainty quantification strategies such as power series and polynomial chaos expansion would be employed to speed up the computational requirements for large-scale complex systems.

Lots of theoretical and practical studies have been performed for optimisation under uncertainty. In (Rasoulilian & Ricardez-Sandoval, 2014), a robust optimisation framework was firstly proposed for multi-scale process systems, where partial differential equations describe the macro-scale phenomena while a high-order lattice-based kinetic Monte Carlo simulator captures the thin film micro-structure dynamics. Power series expansion was employed to perform model uncertainty propagation due to parametric uncertainty of the bulk mole fractions

(boundary conditions) of the multi-scale model, which could result in irreducible effects of the roughness and thickness of the final thin film productions. Second order PSE method could accurately compute the lower and upper probabilistic bounds of output thin film thickness and roughness, which made further optimisation procedure possible. During the dynamic optimisation of the multi-scale processes, the objective was to maximise the lowest expected thin film thickness at the end of the batch deposition process subject to the parametric uncertainty, while expected surface roughness and minimum growth rate constraints were considered at the end of the batch process. Through adaptively adjusting the substrate temperature of the multi-scale deposition process, the thin film thickness of the end batch point could be minimised, while the resulting growth rate and surface roughness requirements were robustly guaranteed compared with optimisation under the nominal parameter values.

In (Li & Floudas, 2014), an optimal scenario reduction stochastic optimisation method was presented for production planning problems with various uncertain parameters such as selling price, production cost, storage cost and production capacity and demand. To reduce the challenging computations due to the large number of scenarios derived from the various unknowns, a small number of representative realisations were selected to capture the characteristics of the scenarios as much as possible, which was formulated into a mixed integer linear optimization problem, minimizing the probabilistic distance, the best, worst and expected performances of original scenario distributions and selected distributions. The first production planning examples show that the proposed stochastic approaches could consistently lead to high similarity between the original distribution and the reduced distribution. Meanwhile, the proposed algorithm also exhibits smaller errors in terms of the best, expected, and worst performance. The second planning examples show that the proposed approach produces better output performance than heuristic methods but compromises more computation time because the objective function includes the output performances.

In (Zheng et al., 2020), a robust economic control framework was presented for dynamic metabolic flux systems, which assumes that biological organisms could efficiently allocate resources to satisfy a specific biological objective such as growth rate. As the core part of the computational framework, a tree-based uncertainty propagation approach was proposed to account for the worst case of each active set where each branch of the tree corresponds to an active set of constraints with a relative probability according to the relative hypervolume

occupied by the active set solutions in the parameter space. A linear dynamic flux balance model was built to describe the behavior of the dynamic response of the bioprocess at each time interval with uncertain parameters such as media composition, inoculum composition and mass transfer variations. Then the robust control problem was reformulated into bi-level optimization problem, with the outer level EMPC optimisation and the inner level robust linear programming at each time interval for uncertain model parameters. The computational case studies showed that the proposed robust controller was much faster than the traditional Monte Carlo approach. Furthermore, the terminal biomass productivity could be improved compared with the nominal control strategy.

In (Schwarm & Nikolaou, 1999), a stochastic model predictive control framework was proposed using the process output constraints, formulating them as chance constraints employing the uncertain system models. The resulting successive online convex optimisation problems could be efficiently solved with the standard model predictive control procedure. The proposed algorithm was tested in a continuous-time high-purity distillation column case study, where the two output stream compositions were controlled by adjusting the reflux ratio and boilup rate. Different levels of chance constraints were incorporated into the model predictive control framework. The computational results showed that the chance constraints could enhance the robustness of performance of control systems compared with the standard model predictive control strategy.

## **2.4 Model reduction**

As previous sections discussed, optimisation, control and uncertainty quantification operations are really time consuming for large-scale systems. Current computational case studies still stay at the level of small-size and/or medium-size problems. To deal with the computational challenges, efficient computational strategies need to be developed.

A promising way to deal with large-scale systems is to use projective model order reduction methods, which reduce the complexity of detailed models but preserve their main input-output features (Schilders et al., 2008). Originally developed in the area of systems and control theory, model order reduction approaches aim to quickly capture the dominant structures of dynamic systems, with sufficient accuracy to represent the original detailed models.

Then model reduction techniques were adopted by scholars in other fields. There is no unified classification for model reduction techniques, which are sometimes divided into linear model reduction approaches and nonlinear model reduction techniques. Here, several popular model reduction techniques are reviewed, such as inertial manifold methods (Jones et al., 1995), proper orthogonal decomposition (Willcox & Peraire, 2002) and the equation-free methods (Kevrekidis et al., 2004). In most situations, the model reduction techniques would be combined into the optimisation, control and uncertainty quantification frameworks for large-scale systems.

Inertial manifold method is one of popular model reduction techniques based on the invariant finite-dimensional Lipschitz manifold, which attracts every trajectory exponentially (Christofides & Daoutidis, 1997). Explicit form of inertial manifold can only be obtained for a specific class of PDEs. For more general and complex systems, approximated inertial manifolds are employed instead of the standard inertial manifold methods. In (Christofides & Daoutidis, 1997), a computational methodology for output feedback controllers was proposed for large-scale distributed systems, employing both approximate inertial manifolds and Galerkin methods. The computational framework for constructing low-dimensional inertial manifolds could capture the characteristics of the original PDEs with desirable accuracy. In (Armaou & Christofides, 2002), a model reduction based dynamic optimisation framework was presented, where the empirical eigenfunctions were applied as global basis functions, and a combination of the method of weighted residuals with approximate inertial manifolds was employed. The diffusion-reaction application was used to validate the computational framework. Only three low-dimensional inertial manifolds could accurately represent the large-scale dynamics. The Kuramoto–Sivashinsky equation case study shows that the low-dimensional approximate inertial manifolds could capture the original high-dimensional process responses except for the special initial conditions where the high order modes are excited and exceed the capability of low-order approximations.

Probably, the most popular model reduction technique is Karhunen-Loeve decomposition, also termed principal component analysis (PCA) or proper orthogonal decomposition (POD). As an efficient dimensionality reduction technique in data science (Hinton & Salakhutdinov, 2006), PCA has been widely applied in the unsupervised machine learning area. The idea of PCA is to use the linear low-dimensional subspace to represent the high dimensional data

sets. Generally, POD would be combined together with projection and/or surrogate model approaches to construct reduced models. In (Theodoropoulou et al., 1998), a combination of POD and Galerkin methods was employed to generate a low-order model for a three-zone rapid thermal chemical vapor deposition system, where spatial wafer temperature patterns leads to nonuniform polysilicon depositions. The reduced surrogate model could accurately predict temperature profiles and the corresponding polysilicon deposition thickness under different input power conditions, which significantly accelerated computational processes, and was then utilised in a dynamic optimisation framework. The proposed model reduction based algorithm was also exploitable for control applications in a similar manner.

In (Xie et al., 2015), a data-driven reduction based nonlinear model predictive control framework was presented for general large-scale distributed parameter systems, where the collected input-output data from black-box simulator and/or experimental information is first employed to project the high system dimensionality onto only one time dimension through the POD method, and then the time coefficients were computed by the ANN surrogate models. Offline ANN model sets were generated, providing alternatives for online nonlinear MPC framework. The proposed computational strategy was validated for a tubular reactor with recycle, where different control objectives including stabilizing and destabilizing the reaction systems with eight actuators were tested. The computations show that the POD-ANN reduced models could predict the dynamic behaviour of the tubular reactor with desirable accuracy, and could be successfully used to compute the closed-loop system dynamics following a reference trajectory. The case study also illustrates that the presented nonlinear control framework could be used to track different kinds of reference profiles.

In (Malik et al., 2018), the PCA method coupled with a Kriging surrogate model was proposed for chemistry combustion systems, where complex thermo-chemical phenomena and heavy species transport equations were employed to describe a wide range of spatiotemporal events. PCA was firstly used to identify the low-dimensional manifolds and then Gaussian process was exploited to fit the hidden nonlinear relationships of the complex systems. The proposed model reduction technique was utilised for a syngas process with a complex fuel and significantly large number of species and reactions in a perfectly stirred reactor. The reduction models could produce very accurate representation of temperature, major and minor species and source terms using only a small number of principal components. Furthermore,

*equation-free* methodologies offer another effective model reduction approach for large-scale black-box systems, for optimisation and control purposes. Equation-free methods aim to exploit the dominant eigendirections of the outputs of complex black-box system models, or direct historical system data, to compute low-dimensional reduced Jacobian and Hessian matrices. In (Bonis & Theodoropoulos, 2012), an equation-free based reduced sequential quadratic programming method was proposed for the computation of low-dimensional Jacobians and Hessians, to accelerate the optimisation procedure for large-scale steady state nonlinear systems. A double projection strategy was utilised adaptively to first project the system onto the computed low-order dominant subspaces and secondly onto the subspace of the independent variables to speed up the computations of optimising large-scale PDE based system models. This reduced optimisation algorithm was performed for a typical chemical reaction system where an exothermic, first-order irreversible reaction takes place. Case studies showed that the proposed scheme is very efficient in terms of memory usage and computational speed compared with conventional deterministic optimisation methods. The computation procedure also successfully computed the optimal Damköhler number value and temperatures of cooling zones to maximize the output product concentration at the exit of the reactor. Then an aggregation function was subsequently applied to address general nonlinear inequality constraints, extending the scope and capability of the equation-free reduced SQP method (Petsagkourakis et al., 2018). The Kreisselmeier–Stainhauser (KS) aggregation function was employed to reduce the large number of inequality constraints to just one inequality. Compared with the typical slack-variable technique transforming the inequality constraints into equality ones, the number of the equality constraints would not be changed during the iterative optimisation procedure. The case study shows that the KS aggregation function-based optimisation method requires the least computational time and iterations compared with the typical reduced SQP and deterministic SQP methods.

Moreover, equation-free based optimisation and control methods have also been constructed using dynamic simulators. In (Bonis et al., 2013), an equation-free model reduction based model predictive control framework was presented, where equation-free model reduction techniques were employed to compute the low-dimensional subspaces and trajectory piecewise linearization strategy was used to construct multiple reduced linear models. The predictive controller synthesis framework could efficiently achieve stabilisation and destabil-

isation objectives with significant computation savings. In (Luna-Ortiz & Theodoropoulos, 2005) , a static optimisation algorithm was developed using available dynamic simulators. The approach employs the recursive projection method (Shroff & Keller, 1993) to identify the slow dominant low-dimensional subspace for the large-scale distributed parameter systems and then employ double projection schemes to accelerate the sequential quadratic programming optimisation procedures. In (Theodoropoulos & Luna-Ortiz, 2006) the equation-free scheme was incorporated into a multiple shooting computation approach, which successfully found the optimal solutions for dynamic process with lower computational resources. In (Armaou et al., 2004), equation-free based linear control methodology was developed for complex/multi-scale systems, where the coarse time-stepper was utilised to identify the slow steady states and the corresponding low-dimensional subspace for design efficient linear quadratic regulator controllers. An extensive discussion about model reduction based optimisation methodologies can be found in (Theodoropoulos, 2011).

Model reduction based uncertainty propagation approaches are important parts of uncertainty quantification techniques, such as PSE and PCE methods. To some extent, PSE and PCE based uncertainty propagation methods belong to surrogate model based uncertainty quantification techniques. In (Zou & Kevrekidis, 2008), an equation-free based uncertainty quantification method was developed for stochastic simulators, combining coarse-grained computations and generalised polynomial chaos expansions. Coarse projective integration and coarse fixed point computations were adopted to speed up the uncertainty computation to converge on multiple steady states. The case study, a heterogeneous catalytic reaction mechanism with an uncertain kinetic parameter, shows that the proposed algorithm could achieve uncertainty quantification with cheaper computational resources compared with traditional techniques.

Moreover, the complexities of practical large-scale nonlinear systems are usually very high. Often a single model reduction technique cannot easily deal with them. For example, although optimal principal component regressions (PCRs)(Pires et al., 2008) are popular to deal with high-dimensional input-output data, the linear or too low-complex models are not accurate enough to replace high nonlinear complex system models. POD on the other hand, is a very powerful method, but projecting the original system onto the global POD modes is not always easy and requires full knowledge of the full-scale system model. Meanwhile,

ANN models can capture highly nonlinear behaviours but usually require large-scale ANN structures (increasing number of neurons and layers) due to the high dimensionality of the original systems. Combining model reduction techniques, e.g. principal component analysis (PCA) with artificial neural networks (ANNs)(Lang et al., 2009), can produce accurate reduced surrogate models. These reduced ANN models could be explicitly utilised by the general-purpose optimisation, control and uncertainty quantification algorithms.

## 2.5 Conclusions and summary

This chapter has reviewed and discussed some of the related literature regarding the methodologies and applications for the optimisation, control and uncertainty quantification for large-scale systems (under uncertainty). At present, existing local optimisation methods with available model equations and derivative information could efficiently deal with large-scale systems (Biegler & Zavala, 2009). However, there is still a large number of practical problems where the availability of system models may not be possible, such as some black-box commercial software (Fluent, 2015; Schefflan, 2016) and complex process systems (Curtarolo & Ceder, 2002). Even if the large volume of legacy codes were explicitly available to users, it is still hard to employ fast model-based techniques. Moreover, local optimisation methods can only guarantee local optima, which may only exploit sub-optimal performance for system optimisation. Rigorous global optimisation algorithms (Horst & Pardalos, 2013) require a huge number of evaluations to ensure the global optimal conditions, which currently are applicable to small to medium size problems. The computational costs for rigorous deterministic global optimisation techniques on large-scale problems is too expensive, needing suitable strategies to overcome these issues. As for more complex optimal control topics, real-time optimisation makes model predictive control approach computationally prohibitive for large-scale systems. Additional computations need to compensate for the robustness of online operations. Nowadays, more robust decisions need be made for large-scale systems under practical uncertainty. However, uncertainty quantification for large-scale systems would be expensive. Polynomial chaos and power series expansions provide efficient tools to deal with this challenge. Furthermore, optimisation and control under uncertainty are still challenging for complex large-scale systems. Model reduction techniques can perform efficiently for large-scale systems, even if

the system models are black-box, which has been used in local optimisation and control technique developments in previous literature (Bonis & Theodoropoulos, 2012; Theodoropoulos & Luna-Ortiz, 2006).

In (Chapters 3-6), new methodologies for model reduction based global optimisation, model predictive control and uncertainty quantification methodologies for large-scale distributed parameter systems are presented.

# Chapter 3

## Model reduction based global optimisation for large-scale steady state nonlinear systems

### 3.1 Introduction

Optimisation techniques are important tools for performing decision-making . One important drawback of the most existing optimisation approaches is that only the local optima can be identified, which is usually worse than the global optimal solution. Fortunately, deterministic global optimisation strategies such as the branch and bound way, can be utilised to seek the global optima with  $\epsilon$  -convergence guarantee. Another challenge is the huge computational costs for large-scale highly nonlinear systems, of which the system evaluations are expensive. Model reduction approaches provide a promising solution to deal with complex large-scale nonlinear systems.

This Chapter focuses on the development of a model reduction based global optimisation computational framework for large-scale distributed parameter systems. Since the explicit formulations of the common large-scale systems in the black-box or open-source commercial modelling software are mostly unavailable, only inputs/outputs simulators can be employed to construct the global optimisation framework.

Specifically, large-scale distributed parameter systems always present complex phenomena, which could be described by PDEs. The direct use of steady-state PDEs is not practical

since the infinite dimensional systems would lead to a large number of algebraic equations through the discretisation in space, further causing huge computational burdens on the deterministic global optimisation procedure. In this work, double model reductions were employed to reduce the computational costs, where the principal component analysis (PCA) was first used to construct the reduced low-dimensional models using the collected high-dimensional simulation data, and then the surrogate artificial neural network (ANN) models were built in the reduced space. The double model reductions could provide accurate but less expensive models than the original large-scale models, accelerating the computations of the subsequent global optimisation procedure.

Nevertheless, the complexity of ANN models includes highly non-convex terms of the traditional hyperbolic tangent functions, which would lead to more expensive computations of low and upper bounds in the global optimisation framework. Two alternative strategies were adopted in this work. One is to approximate the highly hyperbolic tangent functions using piecewise linear affine (PWA) models. Therefore the original nonlinear programming problems could be converted into mixed integer linear programming (MILP) problems, which could employ the advanced MILP solvers such as CPLEX to improve the computational efficiency. The other is to replace the highly hyperbolic tangent activation function with the continuous piecewise linear activation function ReLU. Similarly, the original NLP problems could be directly transformed into MILP problems. Fast MILP solvers could find the near global optima within acceptable computational time.

The computational efficiency of the proposed model reduction based global optimisation framework and improvements were tested through a highly non-convex peak function, a typical tubular reactor and a large-scale combustion process.

## **3.2 Author's contribution**

**Min Tao**

Methodology-Equal, Validation-Lead, Writing-original draft-Lead

**Panagiotis Petsagkourakis**

Investigation-Supporting, Methodology-Supporting

**Jie Li**

Methodology-Supporting, Supervision-Supporting

**Constantinos Theodoropoulos**

Conceptualization-Lead, Funding acquisition-Lead, Methodology-Equal, Project administration-Lead, Supervision-Lead, Writing-review and editing-Lead

**ORIGINAL ARTICLE**

# Model reduction based global optimisation for large-scale steady state nonlinear systems

Min Tao | Panagiotis Petsagkourakis | Jie Li |  
Constantinos Theodoropoulos

Department of Chemical Engineering and Analytical Science, The University of Manchester, M13 9PL, UK

<sup>1</sup>Department of Chemical Engineering and Analytical Science, The University of Manchester, M13 9PL, UK

**Correspondence**

Constantinos Theodoropoulos, Department of Chemical Engineering and Analytical Science, The University of Manchester, M13 9PL, UK  
Email: k.theodoropoulos@manchester.ac.uk

**Funding information**

The University of Manchester and China Scholarship Council joint scholarship (file no. 201706250031)

Many engineering processes can be accurately modelled using partial differential equations (PDEs), but high dimensionality and non-convexity of the resulting systems pose limitations on their efficient optimisation. In this work, a model reduction methodology combining principal component analysis (PCA) and artificial neural networks (ANNs) is employed to construct a reduced surrogate model, which is then utilised by advanced deterministic global optimisation algorithms to compute global optimal solutions with theoretical guarantees. However, the optimisation framework is still time-consuming due to the high non-convexity of the activation functions inside the reduced ANN structure. To further enhance the capability of our optimisation framework, two alternative strategies have been proposed. The first one is a piecewise-affine reformulation while the second one is based on deep rectifier neural networks with ReLU activation function. The performance of the two improved frameworks is demonstrated through two illustrative case studies.

**KEYWORDS**

Model reduction, Distributed parameter systems, Piecewise affine reformulation, Deep rectifier neural networks, Data-driven methodology, Global optimisation

# 1 | INTRODUCTION

Partial differential equation (PDE)-based process models, also termed distributed-parameter systems, have wide applicability in industrial engineering areas [1], such as chemical [2], biochemical [3], and mechanical engineering [4] and aerodynamics [5]. However, complex PDEs are inherently high-dimensional and non-convex, including multiple local optima, hence resulting in intensive computational costs when the computation of global optima is sought. Moreover, most of the generic commercial PDE simulators [6, 7] are essentially black-box and offer no optimisation options. Even if complex model codes are accessible in open-source software (e.g. [8]), the cost of direct optimisation is often unacceptable. To date, performing optimisation tasks efficiently for large-scale complex systems, is still a challenge in engineering design.

A promising way to deal with high dimensionality is to use projective model order reduction methods, which reduce the complexity of detailed models but preserve their main input-output features [9]. The popular principal component analysis (PCA) strategy, an efficient dimensionality reduction technique in data science [10], also termed as Karhunen-Loeve decomposition or proper orthogonal decomposition (POD), is usually combined with projection and/or surrogate model approaches to construct reduced models. POD together with Galerkin projection is capable of producing high-fidelity low-dimensional models for optimisation tasks [11]. Similarly, the combination of POD and ANN can construct reduced surrogate models for black-box large-scale dynamic systems, resulting in efficient optimisation and control strategies [12]. In addition, PCA and Kriging models have been utilised to efficiently replace complex process models [13].

Furthermore, *equation-free* methodologies offer another effective model reduction approach for large-scale black-box systems, for optimisation and control purposes. Exploiting the dominant eigendirections of the outputs of complex black-box system models, or direct historical system data, low-dimensional reduced Jacobian and Hessian matrixes can be computed. An equation-free based reduced SQP method was proposed exploiting the computation of low-dimensional Jacobians and Hessians, to accelerate the optimisation procedure for large-scale steady state nonlinear systems [14]. An aggregation function was subsequently applied to address general nonlinear inequality constraints, extending the scope and capability of equation-free reduced SQP methods [15]. Furthermore, equation-free based dynamic optimisation and control methods have also been constructed [16, 17]. An extensive discussion about model reduction based optimisation methodologies can be found in [18].

To address non-convexity in complex nonlinear optimisation problems, both stochastic and deterministic algorithms can be utilised. Stochastic search methods, such as simulated annealing [19] and genetic algorithms [20], can globally explore the feasible solution space avoiding local optima. However, such stochastic search algorithms are slow for large-scale problems and offer no theoretical guarantees on the global optimality of the computed solutions. Deterministic global optimisation methods are capable of computing global optima utilising branch-and-bound techniques [21], but they are often computationally intensive for large-scale systems due to the need for multiple evaluations of the lower bounds of the optimisation problems.

The aim of this work is to construct an efficient model reduction-based deterministic global optimisation framework for large-scale steady-state input/output (black-box) systems. Often a single model reduction technique cannot easily deal with the complexities of large-scale nonlinear systems. For example, although optimal principal component regressions (PCRs) [22] are popular to deal with high dimensional input-output data, the linear or low-complex models are not accurate enough to replace high nonlinear complex system models. POD on the other hand, is a very powerful method, but projecting the original system onto the global POD modes is not always easy and requires full knowledge of the full-scale system model. Meanwhile, ANN models can capture highly nonlinear behaviours but usually require large-scale ANN structures (increasing number of neurons and layers) due to the high dimensionality of

the original systems. Combining model reduction techniques, e.g. principal component analysis (PCA) with artificial neural networks (ANNs) [23], can produce accurate reduced surrogate models. Then such reduced ANN models can be explicitly utilised by global general-purpose optimisation solvers.

Nevertheless, performing global optimisation tasks with general ANN models is still time consuming (even for reduced ANNs), hence most existing research focuses on local optimisation and/or small-scale problems. Surrogate ANN models have been used to replace superstructure process models and have been optimised locally [24, 25]. Small-scale ANN models (1 hidden layer, 3 neurons) were constructed and optimised globally by the advanced global solver BARON [26]. Larger ANN models are more expensive to optimise as high non-convexity often requires the repeated use of branch-and-bound algorithms. A reduced space-based global optimisation method, recently proposed by Schweidtmann and Mitsos [27], projected the iteration space of non-convex variables onto the subspace of dependent variables, resulting in small-size sub-problems and, consequently, in significant computational savings.

In this work, two strategies are adopted to construct efficient reduced models in the PCA-ANN global optimisation framework. The first is a piecewise affine (PWA) reformulation technique while the second is the use of a deep rectifier neural network. It should be noted that this work extends previous preliminary findings of the authors [28].

The rest of the paper is organised as follows. In Section 2, the basic PCA-ANN global optimisation framework is proposed and the detailed theoretical basis and implementation are provided. In Section 3, the PWA-based reformulation is outlined and illustrated with an example. In Section 4, the deep rectifier ANN-based improvement is employed in the optimisation framework and validated using a large-scale combustion case study. In Section 5, conclusions and further applications are discussed.

## 2 | PROBLEM FORMULATION

In this work, a model reduction-based optimisation framework is presented to deal with large-scale nonlinear steady-state systems focusing on the optimisation of spatially distributed processes, described by sets of steady-state dissipative PDEs:

$$\frac{\partial \mathbf{y}}{\partial t} = D\left\{\frac{\partial \mathbf{y}}{\partial x}, \frac{\partial^2 \mathbf{y}}{\partial x^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial x^n}, \mathbf{d}\right\} + R(\mathbf{d}, \mathbf{y}) \quad (1)$$

Here  $t \in \mathbb{R}$  denotes time,  $x \in \mathbb{R}^{N_x}$ ,  $N_x$  the spatial dimensions,  $N_x=1,2$ , or 3.  $D \in \mathbb{R}$  is the dissipative spatial differential operator,  $\mathbf{d} \in \mathbb{R}^{N_d}$  the parameter variables and  $\mathbf{y} \in \mathbb{R}^{N_y}$  a set of state variables,  $R(\mathbf{d}, \mathbf{y}) : \mathbb{R}^{N_d} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}^{N_y}$  are the nonlinear terms. Considering steady state analysis and assuming that  $\mathbf{y}(t, x) \rightarrow \mathbf{y}(x)$ , and  $\partial \mathbf{y} / \partial t = 0$ , the above equations become:

$$0 = D\left\{\frac{\partial \mathbf{y}}{\partial x}, \frac{\partial^2 \mathbf{y}}{\partial x^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial x^n}, \mathbf{d}\right\} + R(\mathbf{d}, \mathbf{y}) \quad (2)$$

Therefore, the general optimisation problems for steady state PDE-based systems can be formulated as the fol-

lowing problem **P1** :

$$\begin{aligned}
 (\mathbf{P1}) \quad & \min_{\mathbf{d}} \quad G(\mathbf{d}, \mathbf{y}) \\
 s.t. \quad & 0 = D\left\{\frac{\partial \mathbf{y}}{\partial x}, \frac{\partial^2 \mathbf{y}}{\partial x^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial x^n}, \mathbf{d}\right\} + R(\mathbf{d}, \mathbf{y}) \\
 & A\left\{\frac{\partial \mathbf{y}}{\partial x}, \frac{\partial^2 \mathbf{y}}{\partial x^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial x^n}\right\}\bigg|_{x=\Omega} = h_{bds}(\mathbf{d}, \mathbf{y}) \\
 & g_{cons}(\mathbf{d}, \mathbf{y}) \leq 0
 \end{aligned} \tag{3}$$

where  $G(\mathbf{d}, \mathbf{y}) : \mathbb{R}^{N_d} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}$  is the objective function. The equality constraints are the black-box system PDEs with corresponding boundary conditions.  $h_{bds}(\mathbf{d}, \mathbf{y}) : \mathbb{R}^{N_d} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}^{N_y}$  are the right hand sides of the boundary conditions,  $A$  is the operator of the boundary condition equations,  $\Omega$  are the boundaries and  $g_{cons}(\mathbf{d}, \mathbf{y}) : \mathbb{R}^{N_d} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}^{N_y}$  denote other general constraints, e.g. bounds and other constraints, for state variables  $\mathbf{y}$  and design parameter variables  $\mathbf{d}$ .

In general, the unavailability of system equations inside commercial software prohibits the use of direct model-based optimisation techniques. Even in the case that large-scale system equations are available, the optimisation problem **P1** can not be efficiently handled by global optimisation algorithms [29]. In this work, this barrier is overcome by employing accurate surrogate models to formulate a highly accurate approximate problem **P2**, which is then utilised by a general purpose global optimisation solver.

If we use explicit surrogate inputs-outputs to replace the black-box system equations in the above formulation, then problem **P1** can be transformed into the following problem **P2**:

$$\begin{aligned}
 (\mathbf{P2}) \quad & \min_{\mathbf{d}} \quad G(\mathbf{d}, \mathbf{y}') \\
 s.t. \quad & \mathbf{y}' = F(\mathbf{d}) \\
 & g_{cons}(\mathbf{d}, \mathbf{y}') \leq 0
 \end{aligned} \tag{4}$$

where  $\mathbf{y}'$  are the outputs of surrogate models and  $F$  the black-box nonlinear operator.

The gap between problems **P1** and **P2** can be measured by the output errors  $(\mathbf{y} - \mathbf{y}')$ . If these errors are small enough, the global optima of the reduced surrogate model-based problems **P2** will be close to the global solutions of the original problem **P1**. Thus, accurate surrogate models are key to guarantee small errors across the design domain. The accuracy of surrogate models depends on sampling quality and quantity and model building techniques. This work assumes that enough representative/informative samples are available for building accurate models. Details about efficient sampling approaches are discussed in Section 2.1.

Then model building procedures directly affect the accuracy and the complexity of constructed surrogate models, which in turn have a significant effect on the computational accuracy and speed of the subsequent deterministic global optimisation.

This work employs a double model reduction process through a combination of PCA and ANN and reformulation techniques to generate an accurate reduced model, which is subsequently used to compute near global solutions for the original problem **P1**. In the following sections, we are discussing the basic components of our PCA-ANN-global optimisation methodology.

## 2.1 | Sampling and data collection

To build accurate surrogate models, suitable sampling methods are needed to collect highly representative samples for a range of design variables. Inefficient sampling strategies, including too few samples and/or unrepresentative sampling, would result in inaccurate reduced models, in turn producing inaccurate optimal solutions. While provably representative sampling is still an open problem, there are several popular sampling techniques such as Hammersley sequences [30], D-optimal designs [31] and Latin Hypercube (LHC)[32] that can produce good quality results. Hammersley sequences employ quasi-deterministic sequences and generate less samples for convergence, while the D-optimal design approach aims to reduce the number of experiment runs and maximise sample variances. The LHC method can produce samples covering the whole design space and maximize the difference among the generated samples. Specifically, the sample domain is divided into many sub-intervals where sample points are generated randomly in order to represent the whole sub-domain. In this work, we choose LHC because it has been shown to be able to fill the design space to capture input/output relationships given an *adequate* number of samples. The number of LHC samples is decided by testing the model accuracy. In general, more LHC samples are more likely to contain the information needed to capture the complex input/output relationships. LHC sampling for complex systems often requires a relatively large number of samples, which is also the pre-condition to perform successful PCA reduction and ANN-based surrogate models construction.

In the presence of constraints (such as  $g_{cons}$  here), it is hard for the LHC algorithm to directly capture the complex design space. Previous work [33] employed constraints to filter the LHC samples in order to reduce function evaluations for expensive systems. In addition, a complex strategy was used to first decompose the design space into many subdomains, where system features were represented through multiple low-fidelity models. Nevertheless the adaptive optimisation performed within the constrained sampling strategy can be computationally intensive for high-dimensional outputs  $y$  and/or inequality constraints for  $y$ .

In this work, we utilised universal ANN surrogate models to capture the nonlinear behaviour of the black-box PDE equality in its entirety. The constrained sampling strategy may possibly lead to a discontinuous design space, requiring much larger ANN structures to capture it [34]. Here we separated the expensive black-box PDE-based equality constraints from the "known" inequality constraints. We aim to construct accurate but simple ANN models to replace the PDE-based equality constraints, which together with the known inequality constraints  $g_{cons}$ , provide a highly accurate explicit model formulation  $P2$  to the general-purpose global optimisation solvers.

Building accurate surrogate models requires enough representative/informative samples. However, too many samples would lead to intensive computations for high-dimensional systems. Improving sampling efficiency can significantly reduce computations. Exploiting process knowledge or advanced adaptive sampling approaches may help to achieve this goal. Prior knowledge about the processes can provide useful information to collect representative samples with higher probability. However, this requires case-by-case detailed experience about the black-box systems. Adaptive sampling uses a few prior input/output samples to subsequently generate representative samples through solving a set of optimisation problems. Most relevant previous studies in literature [35, 36] deal with low-dimensional inputs/outputs, leading to small-size optimisation problems. This work, however, deals with high-dimensional outputs  $y$ , possibly together with large numbers of inequality constraints  $g_{cons}$ , hence the computation costs for performing adaptive sampling could be high. Nevertheless adaptive sampling procedures are fully compatible with the algorithms developed here provided that the relevant computations can be appropriately reduced.

In this work, we consider a more general approach without exploiting process knowledge and adaptive sampling techniques. The assumption is that the sampling process takes place offline and does not directly affect the computational efficiency of the online optimisation computations. Nevertheless, the proposed model reduction based

global optimisation framework can be easily combined with prior knowledge and adaptive sampling approaches, as mentioned above to speed-up the offline parts of the computations.

We collect samples across the space of design parameters  $\mathbf{d}$  and corresponding input-output data sets ( $\mathbf{D} \in \mathbb{R}^{N_d \times N}$ ,  $\mathbf{Y} \in \mathbb{R}^{m \times N}$ ), where  $m \in \mathbb{N}$  is the number of discrete interval points, which for distributed parameter systems tends to be a large number, and  $N \in \mathbb{N}$  is the number of samples. The obtained data sets ( $\mathbf{D}$ ,  $\mathbf{Y}$ ) are then used to construct accurate reduced surrogate models through the combination of PCA and ANN.

## 2.2 | Principal Component Analysis (PCA)

Due to the high dimensionality of spatially discrete output data  $\mathbf{Y}$ , directly constructing surrogate ANN models would result in large ANN structures. Here, the popular PCA method is first employed to build a reduced model from output data  $\mathbf{Y}$ .

A sampling method (here LHC as discussed in the previous section) is firstly employed to construct a data ensemble  $\mathbf{Y}$  over a finite spatial interval  $\Omega' \in \mathbb{R}$ . PCA then calculates a "small" set of principal components (PCs)  $\mathbf{P} = (p_1, p_2, \dots, p_k)$ ,  $k \in \mathbb{N}$  being the number of PCs, by projecting the data sample  $\mathbf{Y}$  onto the subspace of the,  $k$ , principal components  $\mathbb{P}$ .

$$\mathbf{U} = \mathbf{P}\mathbf{Y} \quad (5)$$

Here  $\mathbf{U} \in \mathbb{R}^{k \times N}$  is the projection of the original data  $\mathbf{Y}$  onto  $\mathbb{P}$  and  $\mathbf{P} \in \mathbb{R}^{k \times m}$  is the orthogonal projector. In the PCA method the matrix  $\mathbb{P}$  is constructed through the covariance matrix,  $\mathbf{C}_y \in \mathbb{R}^{m \times m}$  of the output data  $\mathbf{Y}$ :

$$\mathbf{C}_y = \frac{1}{m-1} \mathbf{Y}\mathbf{Y}^T \quad (6)$$

Here we seek to minimise covariance between data and maximise variance i.e. minimise the off-diagonal elements of  $\mathbf{C}_y$ , while maximising its diagonal elements. This is equivalent to performing singular value decomposition (SVD) on  $\mathbf{C}_y$ :

$$\mathbf{C}_y = \mathbf{Z}^T \mathbf{Z} = \left( \frac{1}{\sqrt{m-1}} \mathbf{Y}^T \right)^T \left( \frac{1}{\sqrt{m-1}} \mathbf{Y}^T \right) = \mathbf{V} \mathbf{D}_1 \mathbf{V}^T \quad (7)$$

where  $\mathbf{D}_1 \in \mathbb{R}^{m \times m}$  is a diagonal matrix whose diagonal elements are the eigenvalues of  $\mathbf{Z}^T \mathbf{Z}$  and  $\mathbf{V}$  is the orthogonal matrix whose columns are the eigenvectors of  $\mathbf{Z}^T \mathbf{Z}$ , which as can be easily shown are equivalent to the principal components of  $\mathbf{Y}$ . In fact we can keep the first  $k$  PCs corresponding to the  $k$  dominant eigenvalues of  $\mathbf{C}_y$ , where usually  $k \ll m$ , hence  $\mathbf{V} \in \mathbb{R}^{m \times k}$  and  $\mathbf{D}_1$  now contains only the  $k$  most dominant eigenvalues of the system,  $\mathbf{D}_1 \in \mathbb{R}^{k \times k}$ . We can then set  $\mathbf{P} = \mathbf{V}^T$  and perform data reduction through the projection in Eq. 5. The original data sample,  $\mathbf{Y}$  can be reconstructed from the projected data:

$$\mathbf{Y} = \mathbf{P}^T \mathbf{U} \quad (8)$$

More details about the theory and application of PCA can be found in [37, 38, 39, 40, 41].

The PCA step aims to project the high dimensional output states arising from the discretisation of the PDE system, onto a small number of dominant variables. Then the resulting low-dimensional relationship between inputs and

projected outputs can be captured through small-size ANN structures, hence producing efficient doubly reduced models that will significantly reduce deterministic global optimisation computations. Implementing the PCA step is not always easy, as it is very sensitive to the quantity and quality of samples. Only with enough representative/informative samples can PCA be efficiently performed and globally capture the accurate dominant PCs. Here we took a tubular reactor case as an illustrative example to show how quantity and quality of samples affect the projection errors.

This is a typical chemical tubular reactor, where an exothermic reaction takes place [42]. The model of the reactor consists of 2 differential equations in dimensionless form as follows :

$$\begin{aligned}
 0 &= \frac{1}{Pe_1} \frac{\partial^2 C}{\partial y^2} - \frac{\partial C}{\partial y} + Da(1 - C) \exp(T/(1 + T/\gamma)) \\
 0 &= \frac{1}{LePe_2} \frac{\partial^2 T}{\partial y^2} - \frac{1}{Le} \frac{\partial T}{\partial y} - \frac{\beta}{Le} T + BDa(1 - C) \exp(T/(1 + T/\gamma)) + \frac{\beta}{Le} T_w
 \end{aligned}$$

(9)

b.c.

$$\begin{aligned}
 \frac{\partial C}{\partial y} - Pe_1 C &= 0, \quad \frac{\partial T}{\partial y} - Pe_2 T = 0, \quad \text{at } y = 0 \\
 \frac{\partial C}{\partial y} &= 0, \quad \frac{\partial T}{\partial y} = 0, \quad \text{at } y = 1
 \end{aligned}$$

Here  $C$  and  $T$  are the dimensionless concentration and temperature respectively, while  $C_{exit}$  is dimensionless output concentration.  $Da$  is the Damköhler number,  $Le$  is the Lewis number,  $Pe_1$  is the Peclet number for mass transport and  $Pe_2$  for heat transport,  $\beta$  a dimensionless heat transfer coefficient,  $C$  is the dimensionless adiabatic temperature rise,  $\gamma$  the dimensionless activation energy,  $T_w$  is the adiabatic wall temperature and  $y$  the dimensionless longitudinal coordinate. The system parameters are  $Pe_1 = 5$ ,  $Pe_2 = 5$ ,  $Le = 1$ ,  $\beta = 1.5$ ,  $\gamma = 10$ ,  $B = 12$ ,  $T_w = 0$ . The resulting discretized 500 algebraic equations comprise our in-house black-box simulator. We want to represent the nonlinear behaviour of 500 outputs (distributed temperature and production concentration) with respect to single design variable  $Da$ , varying in the range  $[0.121, 0.400]$ . For comparison purposes, 6 different sample groups were adopted to generate 10, 20, 30, 40, 50, 60 LHC samples, respectively. Then PCA projection was employed to compute the dominant PCs, where the maximum energy/variance ratio was set to be 99.8 %. Finally, we utilised 500 uniform design points ( 250000 outputs) to test the projection accuracy as the Tb.1 below.

**TABLE 1** Comparative results of projectors from different sample groups

Number of Samples	Number of PCs	Total errors (250000 points)	Maximum absolute error
10	2	5220	3.16
20	2	5350	3.23
30	2	3680	0.86
40	2	4060	3.03
50	3	1170	0.16
60	3	1210	0.50

Tb.1 shows that the smaller sample groups (10, 20, 30, 40, respectively) require only 2 PCs to capture the variance of data set while more sample groups (50, 60, respectively) need 3 PCs, indicating that the less (representa-

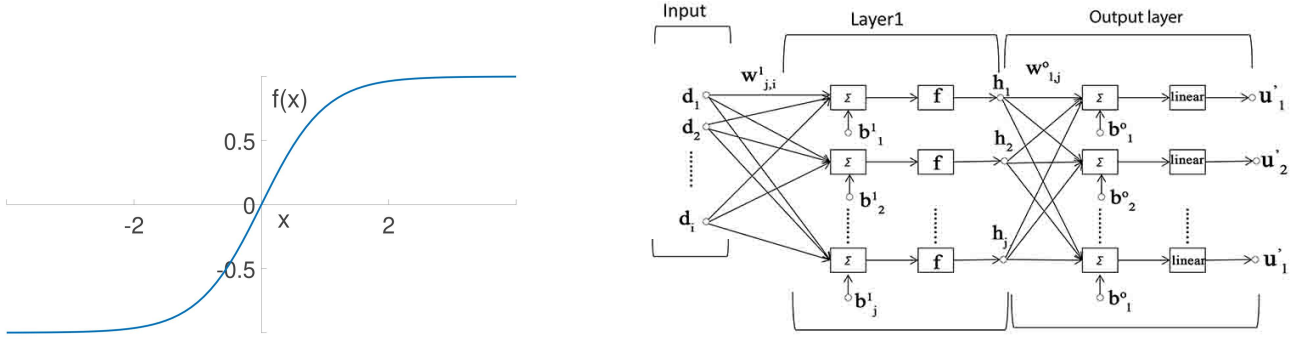
tive/informative) samples would miss some global information (PCs) of the whole design space. This is also the possible reason that the resulting projectors from the smaller sample groups generate higher errors in the validation process. One "abnormal" case is that 40-sample projector is less accurate than 30-sample one despite the larger number of samples. The possible explanation is that the generated 40 samples are less representative than the generated 30 LHC samples. Similar condition occurs between 50 and 60 sample groups. Both the total and the maximum errors of 50-sample group are small enough, implying these 50 samples include enough informative samples to accurately compute the three global PCs.

Moreover, the computed projections reduce the original 500 distributed state variables into only 3 projected variables, which significantly decreases the computations for training ANN models and deterministic global optimisation of the trained models. We employed the previous 50 LHC samples for training Neural Networks. To avoid over- and under-fitting, the defined domain is randomly divided into a training, a validation and a test set with respective size ratios of 0.7 : 0.15 : 0.15. The MATLAB Neural Network Toolbox was utilised to fit the weights and biases by minimizing the mean squared error (MSE) between the ANN model and the training set using Levenberg-Marquardt algorithm and the early stopping procedure. To obtain a suitable number of neurons in the hidden layer, the training process is repeated using an increasing number of neurons until the MSE for all three sets becomes less than a pre-defined tolerance, here  $1 \times 10^{-4}$ . We tested the training process for the single input and projected three outputs, taking less than 1s for the whole 849 iterations to become convergent with a 5-neuron shallow ANN to capture the nonlinear behaviours in reduced space. While each iteration requires 12.41s for the single input and the original 500 outputs to slowly reduce the training errors with the same hidden layer ANN structure. 20 runs for repeatedly training the ANN all failed due to the large validation and/or test errors, implying that a larger ANN is needed to accommodate the inputs/outputs information, which would lead to much more computations for deterministic global optimisation of the trained ANN models. All runs were performed on a Desktop (Intel® Core(TM) CPU 3.3 GHz, 8 GB memory, 64-bit operating system) running Windows 7.

In this work, we choose to gradually add more samples or regenerate (a similar number of) new LHC samples to obtain the global projections. Then a set of fresh samples is employed to validate the accuracy of the computed projections. Both adding more samples and regenerating new samples aim to produce as many representative/informative samples as possible. When the test errors are small enough, i.e. smaller than the pre-defined tolerances, we assume that accurate projections have been produced.

## 2.3 | Artificial Neural Networks (ANNs)

We employ ANNs on the reduced models (Eq.5) from the PCA step. ANN-based models are chosen due to both successful practices and proven theoretical support that a *shallow* feed-forward neural network with one single layer is sufficient to represent any smooth function [43]. Furthermore, advanced optimisation algorithms have been developed to handle the manipulated variables for ANN structures, such as Levenberg-Marquardt backpropagation [44] and Bayesian regularization backpropagation [45]. Fig.1 shows a conventional feed forward neural network with a hyperbolic tangent activation function  $\tanh(\cdot)$ .



**FIGURE 1** Feed-forward neural network with hyperbolic tangent activation function

Shallow ANNs, as the one displayed in Fig.1, are implemented in our basic PCA-ANN global optimisation framework. The feed-forward ANN contains three main components: The input layer, the hidden layer (only one in a shallow ANN) and the output layer, which sequentially perform transformations on the input variables. The input variables,  $\mathbf{d} = (d_1, d_2, \dots, d_{N_d})$ , are first linearly transformed and then non-linearly activated through the hidden layer, and further forced by linear transformation and sequential activation in the output layer, to finally formulate the output variables  $\mathbf{u}' = (u'_1, u'_2, \dots, u'_k), \in \mathbb{R}^k$ . The mathematical description is given in eq. (10) :

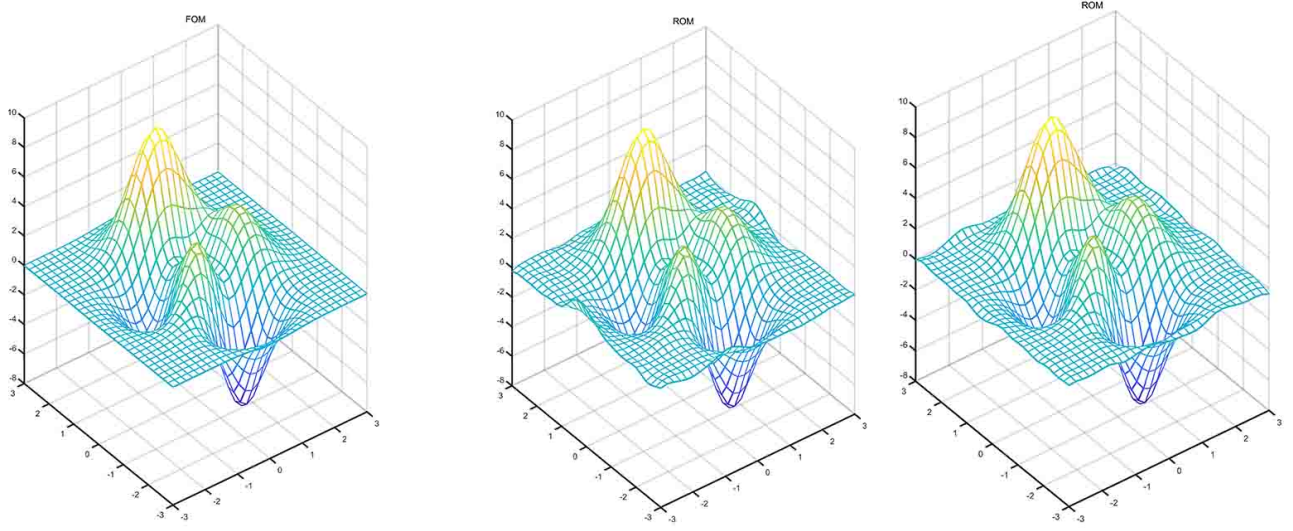
$$\begin{aligned} h_j &= f\left(\sum_{i=1}^{N_d} w_{j,i}^1 d_i + b_j^1\right), \quad \forall j \in \{1, 2, \dots, n\} \\ u'_l &= \sum_{j=1}^n w_{l,j}^o h_j + b_l^o, \quad \forall l \in \{1, 2, \dots, k\} \end{aligned} \quad (10)$$

Here  $h_j \in \mathbb{R}$  is the output value from the hidden layer with  $n \in \mathbb{N}$  neurons,  $j = 1, \dots, n$  and  $f \in \mathbb{R}$  is the activation function. Each neuron  $j$  contains two parameters: weights  $w_{j,i}^1 \in \mathbb{R}$  and biases  $b_j^1 \in \mathbb{R}$  which perform linear transformations. Similarly,  $u'_l \in \mathbb{R}$  is the final value from the output layer with  $k$  neurons,  $l = 1, \dots, k$ , including weights  $w_{l,j}^o \in \mathbb{R}$  and biases  $b_l^o \in \mathbb{R}$ . Three activation functions, the sigmoidal, the hyperbolic tangent and the linear function, are widely used in neural networks. In this work, the hyperbolic tangent function  $f$  was utilised to convert the output value into the range  $[0,1]$  in the hidden layer while the linear function was applied in the output layer. The configured feed-forward neural network was subsequently trained through the back-propagation algorithm using the reduced low-dimensional data sets  $(\mathbf{D}, \mathbf{U})$  from the PCA step. The detailed sampling and training methods used are the same as in previous sections. Similar to the PCA technique, the performance of surrogate ANN models is significantly affected by quantity and quality of collected samples. Here we chose a two-dimensional multi-modal peaks function with the following mathematical formulation as an illustrative example:

$$\begin{aligned} g_{peaks} &= 3(1-x)^2 \exp(-x^2 - (y+1)^2) - 10\left(\frac{x}{5} - x^3 - y^5\right) \exp(-x^2 - y^2) \\ &\quad - \frac{1}{3} \exp(-(x+1)^2 - y^2) \quad x, y \in [-3, 3] \end{aligned} \quad (11)$$

We treated this peaks function as a black-box input/output system and we employed the LHC sampling method to collect snapshots. 300, 500, 1000, 1600 four groups of LHC samples were collected, respectively, to construct the surrogate ANN models. We compared the different surrogate ANN models (ROM) with the original peaks function

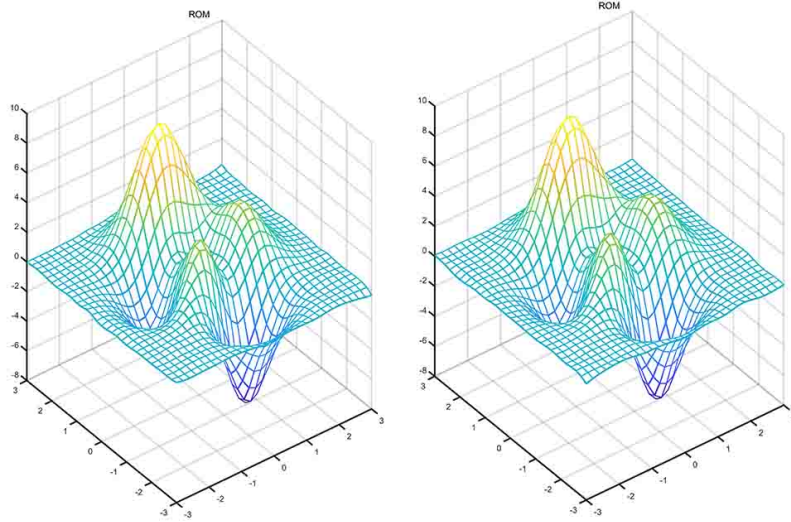
(FOM) using 961 (31 grid points in each direction) uniform grid points as below.



(a) The original peaks function with the global optima being -6.551 at (0.228, -1.626)

(b) 300 LHC samples, 50-neuron shallow neural network with the global optima being -6.461 at (0.237, -1.625)

(c) 500 LHC samples, 55-neuron shallow neural network with the global optima being -6.536 at (0.222, -1.632)



(d) 1000 LHC samples, 45-neuron shallow neural network with the global optima being -6.540 at (0.220, -1.630)

(e) 1600 LHC samples, 52-neuron shallow neural network with the global optima being -6.555 at (0.228, -1.625)

**FIGURE 2** Comparison of the original peaks function (FOM) and different surrogate models (ROMs)

Fig.2 shows that all the four sample groups could locate the general nonlinear behaviours of peaks function and the near global solutions of the peaks function. However, the prediction accuracy of the generated four ANN models are different. The ANN models from relatively more samples can provide more accurate surrogates for the original peaks function across the whole design space. the model across the whole design space would be distinct, smaller as increment of samples. That is really important for constrained optimisation problems where possible nonlinear inequality constraints may significantly change the non-linearity of the optimisation problems, leading to a totally dif-

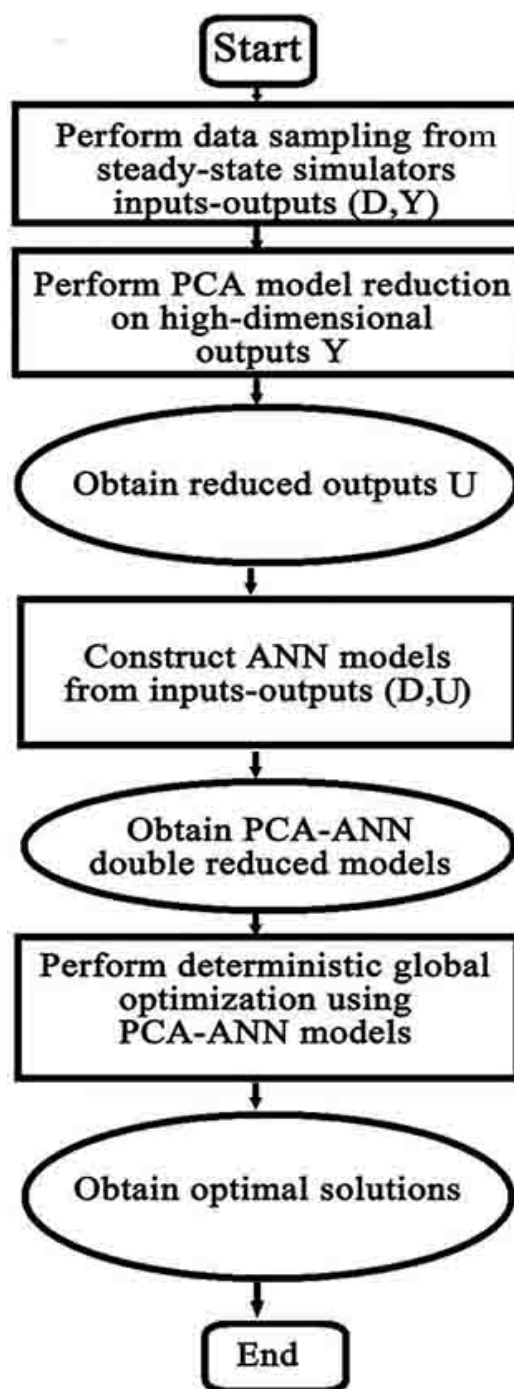
ferent global solution from the unconstrained optimisation problems. Thus more samples should be utilised to reduce the model errors across the whole design space for complex problems, especially with highly nonlinear inequality constraints.

## 2.4 | PCA-ANN global optimisation framework

To cope with the non-convexity of highly non-linear systems, deterministic optimisation methods are considered for the reduced surrogate model from the PCA-ANN reduction. The black-box or grey-box global optimisation problem can be transformed into the general explicit NLP optimisation problems as follows combining Eqs.(4,5,10):

$$\begin{aligned}
 \min_{\mathbf{d}=[d_1, d_2, \dots, d_{N_d}]} \quad & G(\mathbf{d}, \mathbf{u}') \\
 \text{s.t.} \quad & h_j = f\left(\sum_{i=1}^{N_d} w_{j,i}^1 d_i + b_j^1\right), \quad \forall j \in \{1, 2, \dots, n\} \\
 & u'_l = \sum_{j=1}^n w_{l,j}^o h_j + b_l^o, \quad \forall l \in \{1, 2, \dots, k\} \\
 & \mathbf{u}' = (u'_1, u'_2, \dots, u'_k), \\
 & g_{cons}(\mathbf{d}, \mathbf{P}^T \mathbf{u}') \leq 0
 \end{aligned} \tag{12}$$

In this work,  $g_{cons}(\mathbf{d}, \mathbf{P}^T \mathbf{u}')$  includes the box bound constraints and possible inequality constraints for design variables  $\mathbf{d}$  and discretised state variables  $\mathbf{y}'$ . The ANN-based nonlinear objection function  $G$ , can be reformulated into constraints. The main non-convexity of the optimisation problems lies on the constraints  $h_j = f(\cdot)$  due to the highly non-convex activation function, i.e. the hyperbolic tangent function  $\tanh(\cdot)$  in the feed-forward ANN structure. General-purpose global optimisation commercial software, including ANTIGONE [46], BARON [26] and SCIP [47], are efficient tools for the above problems due to the advanced bound tightening and branching techniques. Nevertheless, these general global solvers can not identify  $\tanh(\cdot)$  formulation directly, as high performance algorithms need the explicit model equations. Therefore the explicit algebraic form  $\tanh(z) = (e^z - 1)/(e^z + 1)$  is required [48]. The basic formulation is further transformed into  $\tanh(z) = -2/(e^z + 1) + 1$  in order to produce a tighter under-estimator for the global solver [27]. The flow chart of the basic PCA-ANN global optimisation framework is shown in Fig.3.



**FIGURE 3** Flow chart of the basic PCA-ANN global optimisation framework

Nevertheless, the general ANN constrained global optimisation is expensive. It will be first tested through the previous small peaks function as an illustrative example for equality constrained problems, formulated as below, and

then applied to large-scale nonlinear systems.

$$\begin{aligned}
 \min_{x_1, x_2} \quad & f \\
 \text{s.t.} \quad & f = g_{peaks}(x_1, x_2) \\
 & x_1, x_2 \in [-3, 3]
 \end{aligned} \tag{13}$$

The previously computed global optima was located at the point (0.228, -1.626) with the objective value being -6.551. Then the trained 52-neuron shallow neural network from the 1600 collected samples as shown in Fig.2(e) were employed to replace peaks function. 1600 samples may be a little excessive to compute the global optima but could accurately capture more global features of peaks function, which is more important for complex inequality constrained problems. Meanwhile, we focus on improving optimisation process assuming enough informative samples, which could be guaranteed through excessive samples. For practical large-scale problems, sampling reduction approaches could be combined to improve computational efficiency. We utilised BARON 17.4.1 as the global solver with both relative and absolute tolerances being 0.002 and a limit of 36000 seconds (10 hours). After 30294.26 CPU time, a near global solution was computed at the point (0.228, -1.625) with an objective of -6.555, which is really close to the known global solution of the original problems.

Furthermore, the surrogate model based global optimisation framework, with nonlinear inequality constraints, may have advantages over existing sample based procedures in terms of the computed solution quality. Although reconstructing simple but accurate surrogate models may require relatively more samples compared with sample based procedures, the quality of computed global solutions from sample based procedures would highly depend on the specific problems. Meanwhile, the feasibility and global optima issues caused by inequality constraints  $g_{cons}$ , would be hard for black-box optimisation algorithms to cope with. While surrogate models based global optimisation approaches could efficiently deal with the hardness. If simple but accurate models were built, highly approximate global solutions could be obtained. Here the peaks function as the black-box equality constraints, together with inequality constraints  $g_{cons}$  formulate the following optimisation problem:

$$\begin{aligned}
 \min_{x_1, x_2} \quad & f \\
 \text{s.t.} \quad & f = g_{peaks}(x_1, x_2) \\
 & x_1 f \geq 1.1 \\
 & x_1 x_2 \geq 0 \\
 & x_1, x_2 \in [-3, 3]
 \end{aligned} \tag{14}$$

The global optima could be computed at the optimal point (-0.226, -1.710) with an objective of -4.867 through BARON 17.4.1 with both relative and absolute tolerances being 0.002. Then 1600 LHC samples (0.044 seconds) based 52-neuron shallow neural network was utilised to represent the "black-box" peaks function. Only 9.23 seconds computational times could produce a near global optimal (-0.226, -1.709) with the optimal objective of -4.862, which is extremely close to the true global solution of the original problems. The possible reason for the fast computations is that the inequality constraints perform excellent cuts for design space, enhancing the branch and reduce procedures within BARON. Next, we test the same problem Eq.( 14) using the sample based search solver COBYLA [49] with the convergence tolerance being 0.002 and maximum function evaluations of 2000. Since computational results highly depend on initial points due to global optima issues, 50 runs were implemented with different LHC samples as

initialisation points. Only 20 feasible solutions were found for the 50 LHC samples. The 50 runs took 0.081 seconds to compute the best solution at the point  $(-1.387, 0)$  with the objective function being  $-2.861$ , which was far away from the true global solutions. The computational results imply that sample-based optimisation procedure may suffer serious feasibility and global optima issues. The feasibility issues would become strong if the inequality constraints involve high-dimensional outputs. Moreover, multiple initialisation requires more sample times to avoid local optima but could not guarantee the close global solutions.

Although the above two illustrative examples show that the surrogate ANNs based global optimisation framework with/without inequality constraints could produce near global solutions, the first one without nonlinear inequality constraints comprises intensive computations for deterministic global optimisation due to the complex ANN structures. This drawback drive our further improvements for the basic PCA-ANN global optimisation framework. We expect to reduce the complexity of built ANN models, leading to the faster computations for the general ANN constrained problems. It should be noted that the PCA step is not necessary for the small peaks function example Eq.(14) with the low-dimensional input-output variables, but is utilised for the two large-scale case studies, the tubular reactor and the combustion process, where also the two improvements of the basic framework constructed are demonstrated.

### 3 | PIECE-WISE AFFINE BASED FORMULATION

In this section, a piecewise affine (PWA) reformulation is introduced to deal with the non-convex hyperbolic tangent activation function in the reduced surrogate ANN model. Previous research has suggested the PWA technique for the ANN model [34], which has been verified to be efficient [50]. Although these studies provided some computational results, further detailed implement schemes and analysis have not been reported. In this work, the PWA reformulation was utilised to approximate the highly non-convex NLP problem with a MILP problem. The global optimisation algorithms for both NLP and MILP problems are based on the branch and bound framework. However, the branching step is performed on continuous variables for the NLP problems and on auxiliary binary variables for the MILP problems through the use of CPLEX 12.7.1. An adaptive procedure to construct PWA models is presented below.

#### 3.1 | Adaptive procedure

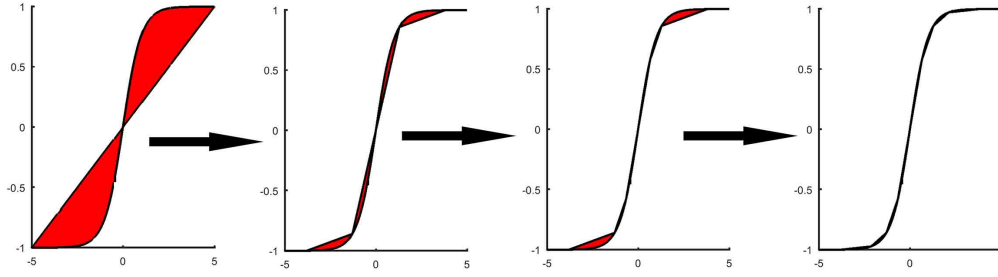
The hyperbolic tangent activation function  $f(z) = \tanh(z)$  is an odd function with central symmetry, which is concave on  $(0, +\infty]$  and convex on  $[-\infty, 0)$ . Therefore the PWA approximation on  $[-\infty, 0)$  can be directly computed from the PWA formulation on  $(0, +\infty]$ . Within the range of  $(0, +\infty]$ ,  $\tanh(z)$  function first increases and then tends to level off with a slight increase towards the limit value of 1. The adaptive PWA procedure starts from the interval  $(0, +\infty]$  and two points, the point of symmetry and one point close to the maximum value (equal to 1). Then a new point is chosen between the two original points so that the error  $E_{ero}$  between  $f(z)$  and its PWA approximation  $f_{PWA}(z)$  (currently consisting of two intervals) is minimised.

$$E_{ero} = \int \text{abs}(f(z) - f_{PWA}(z)) dz, \quad (15)$$

Then the segment with the largest error is chosen and a new point is added within to minimise  $E_{ero}$  in this segment. This procedure continues iteratively until the error in eq. 15 becomes less than a pre-defined tolerance. Finally the points chosen for the  $(0, +\infty]$  interval are mirrored to the  $[-\infty, 0)$  interval.

The iteration procedure efficiently produces a tight PWA representation of the  $\tanh(z)$  function. Fig.4 shows the

adaptive process, narrowing the interval sizes and reducing the error (red shade) between multiple linear models and  $\tanh(z)$ .



**FIGURE 4** Adaptive PWA procedure for the hyperbolic activation function

There are different approaches to formulate the PWA models, such as classic method, linear segmentation method, convex hull method and special structure methods. Classic method is the basic step of the other three methods and may work less efficiently [51]. Here we employed the special order sets since the advanced MILP solvers including CPLEX could smartly exploit the structures of special order sets and speed up the computations [52]. For  $N' + 1$  generated grid points  $z_1, z_2, \dots, z_{N'+1} \in \mathbb{R}$  and correspondingly  $N'$  linear models, the general PWA formulation introducing the special sets variables  $h'_i$  and  $\lambda'_i$ , is as follows [53]:

$$\begin{aligned}
 f(z) &\approx f_{PWA}(z) = \sum_{i=1}^{N'+1} \lambda'_i f(z_i), \\
 z &= \sum_{i=1}^{N'+1} \lambda'_i z_i, \\
 \sum_{i=1}^{N'+1} \lambda'_i &= 1, \\
 \lambda'_1 &\leq h'_1, \\
 \lambda'_i &\leq h'_i + h'_{i-1}, \forall i \in \{2, 3, \dots, N'\} \\
 \lambda'_{N'+1} &\leq h'_{N'}, \\
 \lambda'_i &\geq 0, \forall i \in \{1, 2, \dots, N' + 1\} \\
 \sum_{i=1}^{N'} h'_i &= 1, \\
 h'_i &\in \{0, 1\}^{N'}
 \end{aligned} \tag{16}$$

It should be noted here that the above formulation allows only two adjacent  $\lambda'_i$ 's to be non-zero.

Substituting the highly non-convex  $f(z)$  in the PCA-ANN optimisation formulation (Eq.12) with the above PWA

reformulation (Eq.16), the general PCA-ANN-PWA based MILP optimisation problem can be obtained:

$$\begin{aligned}
& \min_{\mathbf{d}} \quad G(\mathbf{d}, \mathbf{u}') \\
& s.t. \quad z^j = \sum_{i=1}^{N_d} w_{j,i}^1 d_i + b_j^1, \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad h_j = \sum_{i=1}^{N'+1} \lambda_i^j f(z_i), \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad z^j = \sum_{i=1}^{N'+1} \lambda_i^j z_i, \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad \sum_{i=1}^{N'+1} \lambda_i^j = 1, \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad \lambda_1^j \leq h_1^j, \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad \lambda_i^j \leq h_i^j + h_{i-1}^j, \quad \forall i \in \{2, 3, \dots, N'\}, \forall j \in \{1, 2, \dots, n\} \\
& \quad \lambda_{N'+1}^j \leq h_{N'}^j, \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad \lambda_i^j \geq 0, \quad \forall i \in \{1, 2, \dots, N'+1\}, \forall j \in \{1, 2, \dots, n\} \\
& \quad \sum_{i=1}^{N'} h_i^j = 1 \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad h_i^j \in \{0, 1\}^{N'} \quad \forall j \in \{1, 2, \dots, n\} \\
& \quad u'_l = \sum_{j=1}^n w_{lj}^o h_j + b_l^o, \quad \forall l \in \{1, 2, \dots, k\} \\
& \quad \mathbf{u}' = (u'_1, u'_2, \dots, u'_k), \\
& \quad g'_{cons}(\mathbf{d}, \mathbf{P}^T \mathbf{u}') \leq 0
\end{aligned} \tag{17}$$

where  $g'_{cons}$  is a PWA formulation of the possible nonlinear inequality constraints. The following case studies consider the optimisation problems with linear inequality constraints, which is enough to validate the efficiency of PWA formulation of nonlinear ANN models on computational cost and accuracy.

### 3.2 | Illustrative example

To verify the efficiency of the above PWA formulation, global optimisation is performed for the surrogate ANN model and the ANN-PWA model of the peaks function (Eq.(11)). Here, the ANN model is the reduced model (No PCA reduction was necessary). Two different PWA models (with 30 and 58 linear segments, respectively) following the above adaptive procedure. Tab.2 shows the comparison of optimal results of three reduced models. Almost the same optimal solutions are computed, which are close to the global optimum value of the FOM. The proposed ANN-PWA model with 30 linear segments could requires significantly less computational time compared to the other two. In fact we can observe a 4-fold reduction compared to the ANN-PWA model with 58 linear segments and a major 30-fold reduction compared to the ANN formulation. Furthermore, the typical uniform scheme based PWA formulation produces worse solutions than the results from the propose non-uniform scheme, indicating the proposed scheme could more efficiently reduce the errors generated by the PWA models than the common uniform manner. Meanwhile, solving the same-size MILP problem resulting from uniform partitioning scheme is more time-consuming, implying

stronger relaxation tightness of the uniform scheme based MILP than the one from the proposed non-uniform scheme.

**TABLE 2** Comparative results of ANN model and ANN-PWA models

Model	Solver	Optimal value	CPU time (s)	Rel.tolerance
ANN(1 layer, 52 neurons, tanh)	BARON	-6.555	30294.26	0.002
ANN-PWA(30 linear segments)	CPLEX	-6.542	1004.71	0.002
ANN-PWA(30 linear segments, uniform)	CPLEX	-6.406	418.66	0.002
ANN-PWA(58 linear segments)	CPLEX	-6.540	4190.16	0.002
ANN-PWA(58 linear segments, uniform)	CPLEX	-6.489	2840.36	0.002

### 3.3 | Case study

To further investigate the computational efficiency of the PWA method, the PCA-ANN-PWA optimisation framework is illustrated using previous chemical tubular reactor. The mathematical formulation of the optimisation problem is as follows :

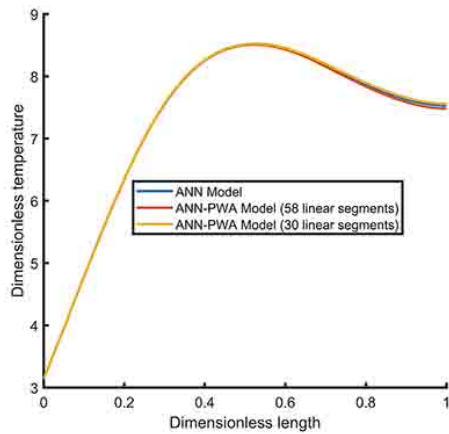
$$\begin{aligned}
 & \max_{T_{wi}} C_{exit} \\
 & s.t. \quad Eq.(9) \\
 & T_w(y) = \sum_{i=1}^3 (H(y - y_{i-1}) - H(y - y_i)) T_{wi}
 \end{aligned} \tag{18}$$

Here Eq.(9) denotes the two PDEs based system equations.  $C_{exit}$  is dimensionless output concentration. The system parameters are  $Pe_1 = 5$ ,  $Pe_2 = 5$ ,  $Le = 1$ ,  $\beta = 1.5$ ,  $\gamma = 10$ ,  $B = 12$ ,  $D_a = 0.1$ .  $T_w$  is the adiabatic wall temperature and  $T_{wi}$  are the corresponding wall temperatures at the three cooling zones.  $H$  is the Heaviside step function.

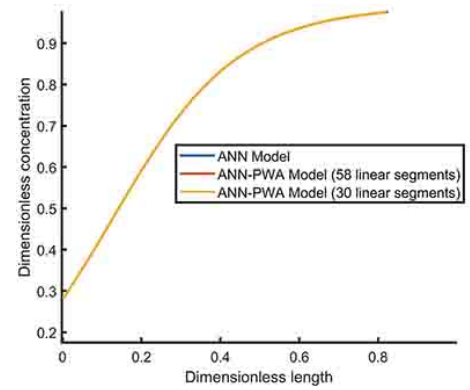
Similar as before, the resulting discretized 500 algebraic equations comprise our in-house FOM simulator. PCA reduction is performed first to reduce the 500 state variables down to 12. Subsequently ANNs are used to obtain a reduced PCA-ANN model comprising 3 inputs, 12 state variables, and 20 ANN neurons. The optimisation results, as displayed in Tab.3 and Fig.5, are computed to compare the optimisation performance using the PCA-ANN model and two PCA-ANN-PWA models with 30 and 58 linear segments, respectively. All three computational cases converge to almost the same solutions, with objective function values close to 0.99998, which is the values computed by performing optimisation with the FOM. The maximum error is 1.52%, and the optimal solution profiles for concentration and temperature distributions are very close to each other for all models (Fig.5). Fig.6 compares the computational time required to perform optimisation using the PCA- ANN and the PCA-ANN-PWA models with different number of ANN neurons. The limit time (max time for computations to stop) was set to 36000 seconds. The computational time increases rapidly with more neurons for all three surrogate models. The computational cost reaches the limit time for the PCA-ANN model with 40 neurons while the CPU time required for the two PCA-ANN-PWA models is less than 1000 seconds. It can be also seen that the computational time required is significantly less for both PCA-ANN-PWA models, irrespective of the number of ANN neurons, implying the high computational efficiency of the proposed PWA methods.

**TABLE 3** Optimal result comparisons for surrogate models of tubular reactor

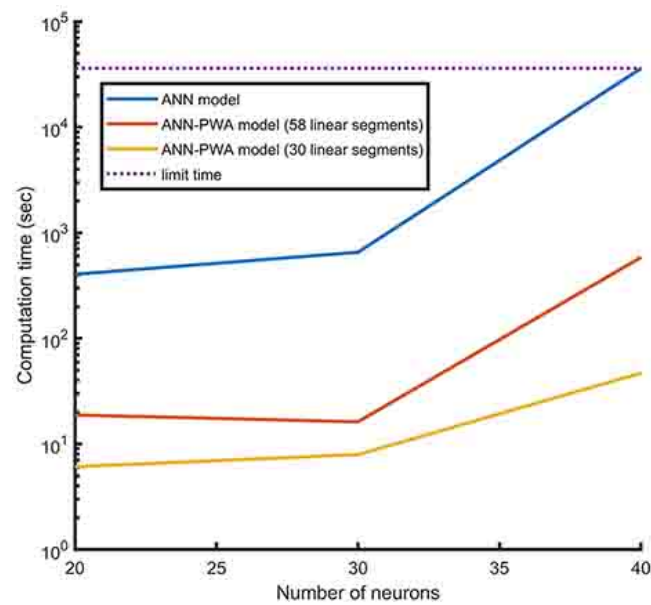
Model	Optimal value	True value (full model)	Error	Computational time(s)
PCA-ANN(1 layer, 20 neurons, tanh)	0.98682	0.99998	1.32%	455.09
PCA-ANN-PWA(30 linear segments)	0.98483	0.99998	1.52%	6.12
PCA-ANN-PWA(58 linear segments)	0.98623	0.99998	1.38%	14.32



(a) Solution profiles for temperature



(b) Solution profiles for concentration

**FIGURE 5** Solution profiles for dimensionless temperature and concentration**FIGURE 6** Computational time (seconds) for different numbers of neurons

## 4 | DEEP RECTIFIER NEURAL NETWORK BASED FORMULATION

PCA-ANN-PWA global optimisation framework worked efficiently for the peaks function and the tubular reactor cases. However, the ANN-PWA models will in general lead to additional approximation error especially for large-scale problems. To preserve the computational accuracy and still use the advanced MILP solver, the continuous piece-wise linear activation function is introduced and directly embedded in the ANN structures. Past efforts in computer science have developed efficient activation functions, such as the sigmoid and the  $\tanh(\cdot)$  function. The traditional S-shaped sigmoid function can transfer any input signal into the range  $[0,1]$  while the zero centered  $\tanh(\cdot)$  function can map the output values in the interval  $[-1,1]$ . Both of them can learn features of high nonlinear functions efficiently. Nevertheless, the high non-convexity of these functions makes ANN training hard to reach a satisfying result. The continuous piece-wise linear functions, including the  $ReLU$  function and its variants, have been adopted to deal with this problem. In this work, the widely applied standard  $ReLU$  function is utilised. Shallow neural networks require an exponentially larger number of nodes in one hidden layer to successfully represent a complex function, while deep neural networks result in more complex and non-convex training error [54] due to the multi-layer structure. Low-complexity two- or three-hidden layer NNs are, however, enough to capture the low-dimensional nonlinear behaviour of PCA-reduced systems.

Although deep rectifier NN-based MILP problems have been formulated in previous studies [55], the combination of PCA and deep rectifier NN has not been reported. The mathematical equations of deep rectifier NNs are similar to fig. 1 with more hidden layers and activation function  $f(z) = \max(0, z)$ , which can be reformulated into piece-wise

linear function through the big-M method [56]:

$$\begin{aligned}
z_1^{j_1} &= \sum_{i=1}^{N_d} w_{j_1,i}^1 d_i + b_{j_1}^1, & \forall j_1 \in \{1, 2, \dots, n_1\} \\
z_1^{j_1} &= z_1'^{j_1} - z_1''^{j_1}, & \forall j_1 \in \{1, 2, \dots, n_1\} \\
z_1'^{j_1} &\leq M_1(1 - bz_1^{j_1}), & \forall j_1 \in \{1, 2, \dots, n_1\} \\
z_1''^{j_1} &\leq M_1 bz_1^{j_1}, & \forall j_1 \in \{1, 2, \dots, n_1\} \\
z_2^{j_2} &= \sum_{j_1=1}^{n_1} w_{j_2,j_1}^2 z_1^{j_1} + b_{j_2}^2, & \forall j_2 \in \{1, 2, \dots, n_2\} \\
z_2^{j_2} &= z_2'^{j_2} - z_2''^{j_2}, & \forall j_2 \in \{1, 2, \dots, n_2\} \\
z_2'^{j_2} &\leq M_2(1 - bz_2^{j_2}), & \forall j_2 \in \{1, 2, \dots, n_2\} \\
z_2''^{j_2} &\leq M_2 bz_2^{j_2}, & \forall j_2 \in \{1, 2, \dots, n_2\} \\
&\cdot & \cdot \\
&\cdot & \cdot \\
&\cdot & \cdot \\
z_\theta^{j_\theta} &= \sum_{j_{\theta-1}=1}^{n_{\theta-1}} w_{j_\theta,j_{\theta-1}}^\theta h_{j_{\theta-1}}^{\theta-1} + b_{j_\theta}^\theta, & \forall j_\theta \in \{1, 2, \dots, n_\theta\} \\
z_\theta^{j_\theta} &= z_\theta'^{j_\theta} - z_\theta''^{j_\theta}, & \forall j_\theta \in \{1, 2, \dots, n_\theta\} \\
z_\theta'^{j_\theta} &\leq M_\theta(1 - bz_\theta^{j_\theta}), & \forall j_\theta \in \{1, 2, \dots, n_\theta\} \\
z_\theta''^{j_\theta} &\leq M_\theta bz_\theta^{j_\theta}, & \forall j_\theta \in \{1, 2, \dots, n_\theta\} \\
z_i'^{j_i} &\geq 0, & \forall i \in \{1, 2, \dots, \theta\}, \forall j_i \in \{1, 2, \dots, n_i\} \\
z_i''^{j_i} &\geq 0 & \forall i \in \{1, 2, \dots, \theta\}, \forall j_i \in \{1, 2, \dots, n_i\} \\
bz_i^{j_i} &\in \{0, 1\} & \forall i \in \{1, 2, \dots, \theta\}, \forall j_i \in \{1, 2, \dots, n_i\} \\
u_l' &= \sum_{j_\theta=1}^{n_\theta} w_{l,j_\theta}^o z_\theta^{j_\theta} + b_l^o, & \forall l \in \{1, 2, \dots, k\}
\end{aligned} \tag{19}$$

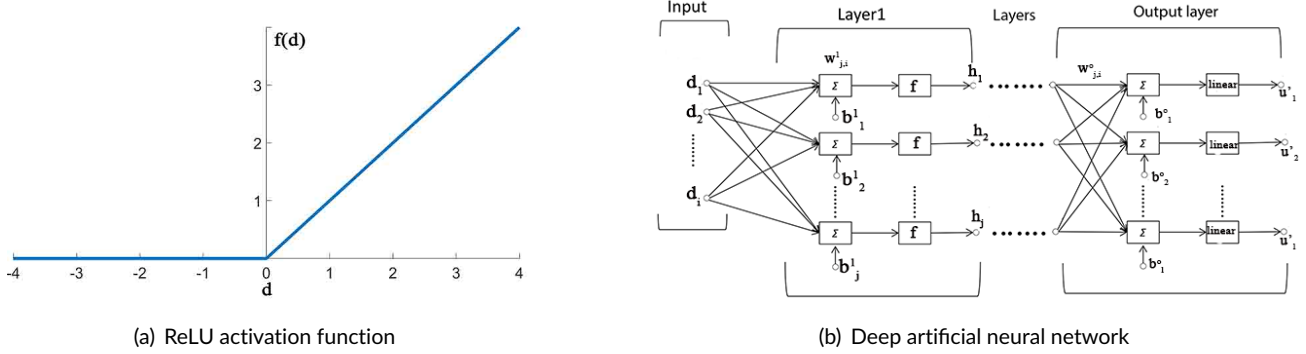
where  $M_i$  is the big-M constant. Here we used a uniform Big-M value being 10000, which has been showed enough to capture the near global solutions of small peaks function example without numerical issues.  $z_i'^{j_i}$  and  $z_i''^{j_i}$  are the auxiliary non-negative variables,  $bz_i^{j_i}$  is the auxiliary binary variable and  $h_{j_i}^i$  is the output value from the  $j_i$ th ReLU based neuron of the  $i$ th hidden layer.  $\theta$  is the number of hidden layers and  $n_i$  is number of neurons at  $i$ th hidden layer.

Substituting the ANN model equations in the PCA-ANN optimisation formulation (Eq.12) with the above PWA reformulation (Eq.20), the general PCA-DNN(ReLU) based MILP optimisation formulation can be obtained:

$$\begin{aligned}
&\min_{\mathbf{d}=[d_1, d_2, \dots, d_{N_d}]} G(\mathbf{d}, \mathbf{u}') \\
&s.t. \quad Eq.(19); \\
&\quad \mathbf{u}' = (u'_1, u'_2, \dots, u'_k), \\
&\quad g'_{cons}(\mathbf{d}, \mathbf{P}^T \mathbf{u}') \leq 0
\end{aligned} \tag{20}$$

where  $Eq.(19)$  denotes the equality constraints of deep neural networks through the big-M method.

This way, an improved framework is formulated using a deep neural network (DNN) with rectified linear units (ReLU) as illustrated in Fig.7. This improved framework is first tested with the small peaks function, and then extended to a large-scale combustion process.



**FIGURE 7** Deep neural network with rectified linear units

#### 4.1 | Illustrative example

To verify the superiority of the reLu-DNN models, in terms of computational efficiency global optimisation is first constructed for four surrogate models (ANN, ANN-PWA, tanh-DNN and relu-DNN model, respectively) for the peaks function previously presented. Tab.4 shows the optimisation results. The small-scale tanh-DNN could replace the larger shallow ANN model, resulting in a significant computational saving, more than one order of magnitude. The relu-DNN model requires much more neurons than the tanh-DNN model, due to the lower non-linearity of the *relu* activation function. Despite the fact that the relu-DNN model is larger, its optimisation cost is much lower, two orders of magnitude less than the cost of the tanh-NN models. The rapid global optimisation computations using the relu-NN model is attributed to the advanced MILP solver algorithm utilised. Furthermore, the computation cost using the relu-DNN model is much less than that using the ANN-PWA model with 30 linear segments because of the large(r) number of linear models involved in the PWA formulation. More linear models lead to more binary variables, requiring more branching steps reducing the computational efficiency.

**TABLE 4** Comparative optimisation results of different ANN models

Model	Solver	Optimal value	CPU time (s)	Rel.tolerance
ANN(1 layer, 52 neurons, <i>tanh</i> )	BARON	-6.555	30294.26	0.002
ANN-PWA (30 linear segments)	CPLEX	-6.542	1004.71	0.002
DNN(2 layers, 8-8 neurons, <i>tanh</i> )	BARON	-6.558	2579.68	0.002
DNN(2 layers, 40-40 neurons, <i>relu</i> )	CPLEX	-6.543	25.93	0.002

#### 4.2 | Case study

To verify the significant advantages of the deep rectifier neural network in our global optimisation formulation observed in the previous section, a more challenging combustion process [57, 23] is considered here.

### 4.2.1 | Process description

A combustion process taking place in a horizontal cylindrical combustor, 1.8m in length and 0.45m in diameter with a fuel nozzle with diameter 0.0045m is considered here. The overall reactions in the combustor are as follows:

- $\text{CH}_4 + 2 \text{O}_2 \rightarrow \text{CO}_2 + 2 \text{H}_2\text{O}$
- $\text{C}_2\text{H}_4 + 3 \text{O}_2 \rightarrow 2 \text{CO}_2 + 2 \text{H}_2\text{O}$
- $\text{C}_3\text{H}_8 + 5 \text{O}_2 \rightarrow 3 \text{CO}_2 + 4 \text{H}_2\text{O}$
- $\text{C}_4\text{H}_{10} + 8.5 \text{O}_2 \rightarrow 4 \text{CO}_2 + 5 \text{H}_2\text{O}$

In addition a complex NO mechanism, comprising thermal NO, prompt NO and  $\text{N}_2\text{O}$  intermediate mechanism is also taken into account. Fuel NO mechanism was ignored due to the small amount of nitrogen in the feed. Thermal efficiency can be improved by increasing combustion temperature, which however, inevitably leads to more pollutant emissions, such as NOx. The NOx production is dominated by the thermal NO mechanism, given below, which is very sensitive to temperature.

- $\text{O} + \text{N}_2 \rightleftharpoons \text{NO} + \text{N}$
- $\text{O}_2 + \text{N} \rightleftharpoons \text{NO} + \text{O}$
- $\text{N} + \text{OH} \rightleftharpoons \text{NO} + \text{H}$

This work focuses on the optimisation of inlet operational conditions (shown in Tb. 5) in order to minimise NOx emissions. In addition to chemical reactions, multiple physical phenomena are involved, including complex turbulent flows, heat and mass transfer. Commercial CFD software was used, namely ANSYS/FLUENT, to construct high-fidelity CFD models to calculate velocity, temperature and component fraction fields.

### 4.2.2 | CFD Model Description

The computation domain for the CFD model consisted of a 2-dimensional axisymmetric depiction of the combustor. To ensure that computations are grid independent, numerical experiments using 5481, 6381, 9081 and 14832 computational cells were performed for the maximum temperature. Finally, 9081 computational cells (9332 nodes) were chosen as solutions did not change with more computational cells/nodes. The renormalisation group (RNG)  $k - \epsilon$  turbulence model for fluid flow is employed. The eddy-dissipation model was employed for the species transport equations because the overall reaction rate is controlled by turbulent mixing. To take into account the effects thermal radiation, including absorption and scattering coefficients, a discrete ordinates (DO) radiation model was used.

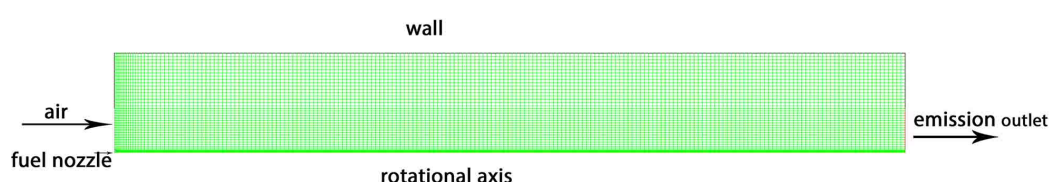
The second-order upwind scheme was applied for the space derivatives of the advection terms in all transport equations. The SIMPLE algorithm was employed to handle the velocity-pressure coupling in the flow field equations. Convergence criteria required the residual for the energy equation to be below  $1 \times 10^{-6}$  and the residuals for the other model equations below  $1 \times 10^{-3}$ . The mass-weighted-averages of temperature at the exit and the maximum temperature of the entire fluid were also monitored as other convergence criteria.

The base case inlet conditions are given below. For fuel gas, the base value for inlet velocity was 100 m/s and that for inlet temperature was 298K. The inlet composition was as follows:  $\text{CH}_4$ : 87.8%,  $\text{C}_2\text{H}_4$ : 4.6%,  $\text{C}_3\text{H}_8$ : 1.6%,  $\text{C}_4\text{H}_{10}$ : 0.5%,  $\text{N}_2$ : 5.5%. For preheated air gas, the inlet velocity was 85 m/s and the inlet temperature 1473 K while the inlet air composition was:  $\text{O}_2$ : 19.5%,  $\text{N}_2$ : 59.1%,  $\text{H}_2\text{O}$ : 15%,  $\text{CO}_2$ : 6.4%, NO: 110 ppm. Five independent variables were

used to optimise the whole process. The independent variables along with their allowable ranges are listed in Tab.5 below.

**TABLE 5** Range of independent variables

Variables	Range	Units
inlet air velocity	[85,125]	m/s
inlet fuel velocity	[80,120]	m/s
oxygen mass fraction (inlet air)	[18.5, 19.5]	%
inlet air temperature	[1450, 1600]	K
inlet fuel temperature	[298, 398]	K



**FIGURE 8** Two-dimensional geometry of a single axisymmetric combustor can and its mesh

### 4.2.3 | Model reduction

Although the high-fidelity CFD model can provide accurate simulation results, its black-box characteristics and overall complexity make further optimisation and control tasks computationally tedious. Reduced surrogate models need to be developed to deal with the challenges arising. The LHC sampling method was utilised to collect 1024 CFD samples, which took around 6 days through 4-CPU parallel computing. The input variables are the ones listed in Tab.5, while the output results are the physical field data along with the average NO<sub>x</sub> emission at the outlet surface. Due to the high dimensionality of the FOM, direct mapping of the input-output relationship would result in very large-scale ANN surrogate models, which often exceed the capability of current optimisation algorithms. Therefore, the PCA step was first employed and then surrogate ANN models were constructed based on the PCA-reduced models.

ANN models were built for the field data, to construct the reduced PCA-ANN constraints and for the average output NO<sub>x</sub> emission to formulate the ANN-reduced objective function. The field data include axial and radial velocity, Static Temperature, N<sub>2</sub>, H<sub>2</sub>O, O<sub>2</sub>, CO<sub>2</sub>, C<sub>4</sub>H<sub>10</sub>, CH<sub>4</sub>, C<sub>2</sub>H<sub>4</sub>, C<sub>3</sub>H<sub>8</sub>, and NO fraction concentrations (12 state variables). It should be noted that the average output NO<sub>x</sub> emission is only one variable so does not require a PCA reduction step. In this work, PCA was performed separately for each state variable. While some PCA methods compute principal components for all state variables together, we found that working on each state variable we could generate more accurate principal components. The standard criterion, of capturing 99.99% of the total energy, was set. This way, the reduced surrogate models, were built, as displayed in Tab.6.

**TABLE 6** Number of PCs and corresponding ANN models

Variables	Number of PCs	DNN (2 layers, tanh)	DNN (2 layers, relu)
		No of neurons	No of neurons
Axial velocity	4	14, 14	14, 14
Radial velocity	9	15, 15	22, 22
Temperature	6	16, 16	16, 16
N <sub>2</sub> concentration fraction	7	19, 19	24, 24
H <sub>2</sub> O concentration fraction	8	15, 15	18, 18
O <sub>2</sub> concentration fraction	6	17, 17	20, 20
CO <sub>2</sub> concentration fraction	7	12, 12	14, 14
C <sub>4</sub> H <sub>10</sub> concentration fraction	6	15, 15	17, 17
CH <sub>4</sub> concentration fraction	7	26, 26	28, 28
C <sub>2</sub> H <sub>4</sub> concentration fraction	6	10, 10	18, 18
C <sub>3</sub> H <sub>8</sub> concentration fraction	6	18, 18	24, 24
NO concentration fraction	4	12, 12	14, 14
Objective: output NO <sub>x</sub> emission	-	ANN (1 layer, tanh)	ANN (1 layer, relu)
	-	14	30

#### 4.2.4 | Model validation

Model validation was implemented on the reduced models before the subsequent optimisation step, taking into account two aspects, representation ability and prediction ability. The representation ability of the reduced models was tested through the comparison between the FOM and ROMs on the base case inlet conditions. Computational results show only very small differences, especially for N<sub>2</sub>, C<sub>4</sub>H<sub>10</sub>, CH<sub>4</sub>, C<sub>2</sub>H<sub>4</sub>, C<sub>3</sub>H<sub>8</sub>, and NO fraction fields. The above species fraction fields are close to uniform distribution across the combustor, except for the small area near the fuel nozzle. Fig.9(a), 9(b), 9(c), 9(d), 9(e) give the velocity field, temperature field, O<sub>2</sub>, CO<sub>2</sub> and H<sub>2</sub>O concentration fraction field of FOM, tanh-ROM and relu-ROM under the inlet being base values. The five contour diagrams illustrate that flow, temperature and mass fraction fields of FOM and ROMs are very close, indicating the strong representation ability of the ROMs. Moreover, the tanh-DNN reduced models show smaller difference than the relu-DNN reduced models, especially for the temperature field, implying the better accuracy of the tanh-DNN models due to the non-linearity of tanh function. Tab.7 shows the comparison of maximum field values between FOM and ROMs and the corresponding errors. The largest error is only 0.56%. To test the ROMs prediction ability, 40 random inlet condition points different than the base case ones were chosen and compared with FOM results. The largest error was less than 5% indicating that the ROMs can be reliably used for further optimisation studies. Furthermore, the ROMs exhibit significant computational savings compared to the full-order CFD models as expected. The average CPU time for the CFD model (run in ANSYS/FLUENT) is approximately 1560 CPU seconds, while each ROM requires less than 0.1 CPU seconds and can be efficiently used to perform global optimisation studies.

**TABLE 7** Average value comparison of FOM and ROMs

Variables	FOM	relu-ROMs	errors	tanh-ROMs	errors
Velocity(m/s)	29.82089	29.6652	0.40%	29.70166	0.56%
Temperature(K)	1625.259	1621.948	0.03%	1625.702	0.20%
H <sub>2</sub> O mass fraction	0.151743	0.1518191	0.02%	0.1517722	0.05%
O <sub>2</sub> mass fraction	0.1906906	0.1908453	0.01%	0.1907176	0.08%
CO <sub>2</sub> mass fraction	0.0663304	0.06627689	0.00%	0.06633245	0.08%
Computational time for each sample	1560	<0.1	-	<0.1	-

#### 4.2.5 | Global optimisation

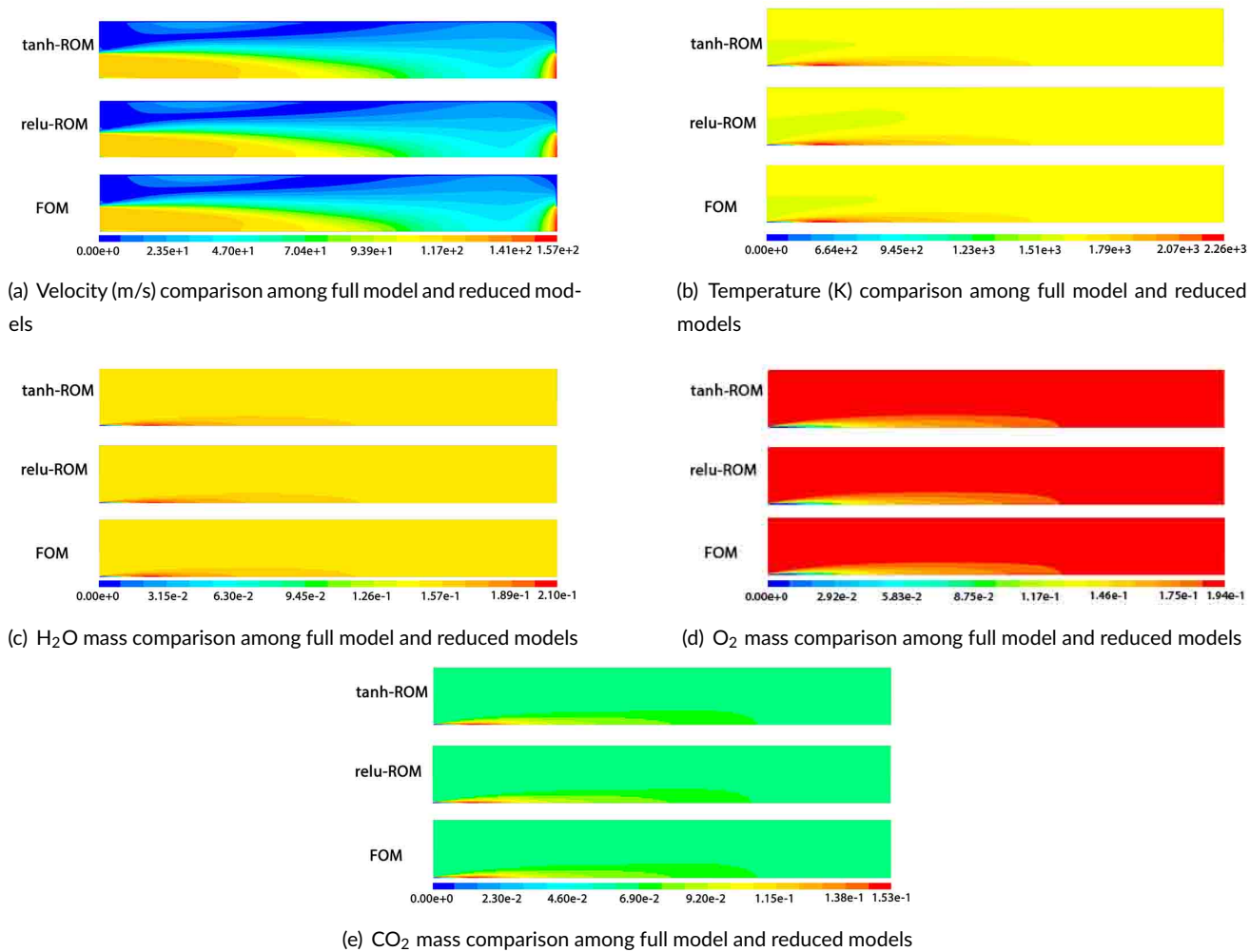
In this section, global optimisation is implemented using the validated reduced models. The general mathematical formulation is given in Eq.(20). In the combustion optimisation problem,  $\mathbf{d}$  are the 5 inlet operation parameters,  $\mathbf{u}'$  are the 76 reduced state variables. The objective function  $G(\mathbf{d}, \mathbf{u}')$  represents the average outlet NOx emission. The allowable ranges for the input variables are given in Tab.5, while the bounds for the state variables are given in Tab.8. It should be noted that the state variable bounds are implemented through the inverse projection

$$lb \leq P^T \mathbf{u}' \leq ub \quad (21)$$

where  $lb$  and  $ub$  denote lower and upper bounds, respectively.

Finally, an MILP problem with 29,903 linear constraints, corresponding to the equality constraints and 488 binary variables corresponding to the total number of ANN neurons is formulated for the relu-based ROM, while an NLP problem with 28247 linear constraints, and 392 nonlinear terms is constructed for the tanh-based ROM. The limit value for the computational time was set to be 100 hours. Both of the relative and absolute tolerances were set to be 0.002.

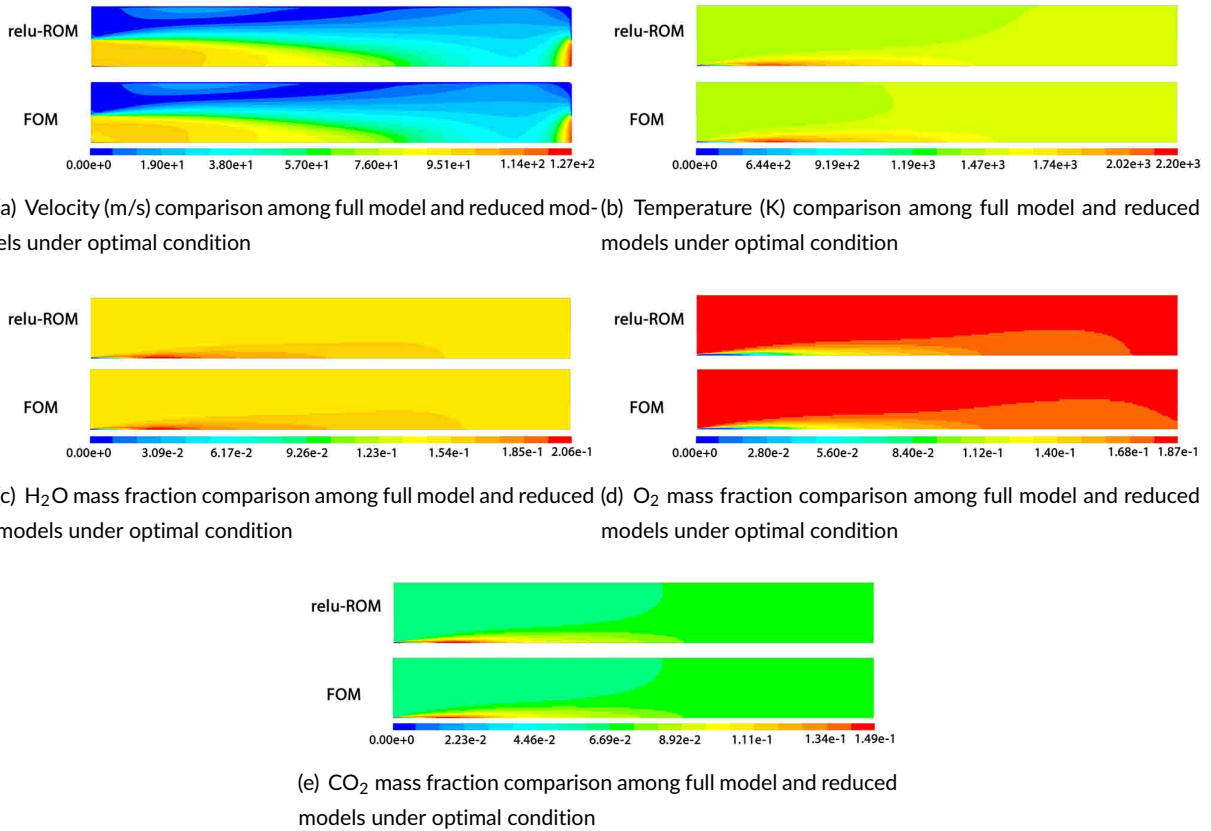
The NLP problem did not converge to a feasible solution in BARON within the allowable time, probably due to the high non-convex activation function  $\tanh$  and large number of variables than inhibited the branch-and-bound algorithm. The *relu*-based MILP problem converged in 1001.94s in CPLEX. The computed optimal solution was: NOx emission: 110.17 ppm, air velocity: 95.07 m/s, fuel velocity:119.08m/s, oxygen fraction concentration (air): 18.50 %, air temperature: 1450 K and fuel temperature: 369.83K. To validate the computed optimal solutions, we performed a full CFD simulation in ANSYS/FLUENT using the calculated optimal inlet conditions. The outlet NOx emission was computed to be 113.26 ppm, which was very close to the calculated optimum with an error of approximately 2.73 %, which is small enough for most industrial cases. Fig.10(a), 10(b), 10(c), 10(d), 10(e), depicts a comparison of the main field state variables at the optimal; conditions computed by the reduced and the full models, respectively. As it can be observed, the optimal solution computed through the ROM is very close to FOM simulation using the optimal inlet conditions. Tab.9 gives a comparison of the corresponding max values across the whole domain. The performance of the reduced model is very close to the full model with the biggest error being less than 3%. The computational cost for the relu-based MILP problem is significantly reduced compared to the NLP problem.



**FIGURE 9** Comparison of velocity, temperature and concentration fraction field between FOM and ROMs

**TABLE 8** Range of state variables

State variables	Range	Units
Axial velocity	[-150,150]	m/s
Radial velocity	[-150,150]	m/s
Temperature	[0, 2200]	K
N <sub>2</sub> concentration fraction	[0,1]	-
H <sub>2</sub> O concentration fraction	[0,1]	-
O <sub>2</sub> concentration fraction	[0,1]	-
CO <sub>2</sub> concentration fraction	[0,1]	-
C <sub>4</sub> H <sub>10</sub> concentration fraction	[0,1]	-
CH <sub>4</sub> concentration fraction	[0,1]	-
C <sub>2</sub> H <sub>4</sub> concentration fraction	[0,1]	-
C <sub>3</sub> H <sub>8</sub> concentration fraction	[0,1]	-
NO concentration fraction	[0,1]	-

**FIGURE 10** Comparison of optimal velocity, temperature and fraction concentration field between FOM and ROM

**TABLE 9** Average value comparison of FOM and ROMs

Variables	FOM	relu-ROMs	errors
Velocity(m/s)	23.11901	23.73662	2.67%
Temperature(K)	1487.854	1456.745	2.09%
H <sub>2</sub> O mass fraction	0.1522371	0.1520701	0.11%
O <sub>2</sub> mass fraction	0.1803872	0.1806236	0.13%
CO <sub>2</sub> mass fraction	0.06699268	0.06689615	0.14%
Output NOx emission (ppm)	113.26	110.17	2.73%

## 5 | CONCLUSIONS

This paper presents a model-reduction based global optimisation framework for large-scale nonlinear steady-state systems. A double model reduction, comprising principal component analysis and artificial neural networks, were first employed to construct the reduced model, which was then utilised by deterministic global optimisation methods. The high non-convexity of the activation function in reduced ANN models affects the computational speed branch-and-bound algorithms. To overcome this barrier, two improvements are proposed. Firstly, a piece-wise affine reformulation to transform the nonlinear branching into binary variables resulting in an MILP problem with higher computational efficiency. Secondly, the implementation of a continuous piece-wise linear activation function-based deep ANN structure to improve computational accuracy. Applications including peaks function, a tubular reactor and a complex large-scale combustion process were employed to illustrate the favorable performance of the improved framework. Nevertheless, it is still a challenge to efficiently compute the global optimum for large-scale optimisation problems. Firstly, this work assumes enough representative samples as a basis to construct the reduced order models. Smart sampling methods to achieve optimal trade-off between quality and quantity are important for improving both efficiency and accuracy, as well as verification methods to guarantee the accuracy of the computed solutions [58]. Secondly, global optimisation even using reduced surrogate models is still computationally expensive. Advanced data techniques and MILP algorithms [59] may further improve computational efficiency of this optimisation framework.

## acknowledgements

The financial support of the University of Manchester and China Scholarship Council joint scholarship (file no. 201706250031) for MT's PhD studies is gratefully acknowledged

## conflict of interest

The authors declare there is no conflict of interest.

## references

- [1] Boukouvala F, Hasan MF, Floudas CA. Global optimization of general constrained grey-box models: new method and its application to constrained PDEs for pressure swing adsorption. *Journal of Global Optimization* 2017;67(1-2):3–42.

- [2] Tao M, Guo K, Huang Z, Liu H, Liu C. A hybrid optimization method to design shapes of three-dimensional flow channels. *Chemical Engineering Research and Design* 2016;114:190–201.
- [3] Park S, Li Y. Integration of biological kinetics and computational fluid dynamics to model the growth of *Nannochloropsis salina* in an open channel raceway. *Biotechnology and bioengineering* 2015;112(5):923–933.
- [4] Yang GL, Zhou WH, Liu F. Simulation of flow field of high-pressure water-jet from nozzle with FLUENT [J]. *Journal of Lanzhou University of Technology* 2008;34(2):49–52.
- [5] Kleber A. Simulation of air flow around an Opel Astra vehicle with FLUENT. *Journal Article, International Technical Development Center Adam Opel AG* 2001;.
- [6] Multiphysics C. Introduction to COMSOL Multiphysics®. COMSOL Multiphysics, Burlington, MA, accessed Feb 1998;9:2018.
- [7] Fluent A. Ansys fluent. *Academic Research Release* 2015;14.
- [8] Jasak H, Jemcov A, Tukovic Z, et al. OpenFOAM: A C++ library for complex physics simulations. In: *International workshop on coupled methods in numerical dynamics*, vol. 1000 IUC Dubrovnik Croatia; 2007. p. 1–20.
- [9] Schilders WH, Van der Vorst HA, Rommes J. *Model order reduction: theory, research aspects and applications*, vol. 13. Springer; 2008.
- [10] Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *science* 2006;313(5786):504–507.
- [11] Theodoropoulou A, Adomaitis RA, Zafiriou E. Model reduction for optimization of rapid thermal chemical vapor deposition systems. *IEEE Transactions on Semiconductor Manufacturing* 1998;11(1):85–98.
- [12] Xie W, Bonis I, Theodoropoulos C. Data-driven model reduction-based nonlinear MPC for large-scale distributed parameter systems. *Journal of Process Control* 2015;35:50–58.
- [13] Malik MR, Isaac BJ, Coussement A, Smith PJ, Parente A. Principal component analysis coupled with nonlinear regression for chemistry reduction. *Combustion and flame* 2018;187:30–41.
- [14] Bonis I, Theodoropoulos C. Model reduction-based optimization using large-scale steady-state simulators. *Chemical engineering science* 2012;69(1):69–80.
- [15] Petsagkourakis P, Bonis I, Theodoropoulos C. Reduced Order Optimization of Large-Scale Nonlinear Systems with Nonlinear Inequality Constraints Using Steady State Simulators. *Industrial & Engineering Chemistry Research* 2018;57(30):9952–9963.
- [16] Bonis I, Xie W, Theodoropoulos C. Multiple model predictive control of dissipative PDE systems. *IEEE Transactions on Control Systems Technology* 2013;22(3):1206–1214.
- [17] Luna-Ortiz E, Theodoropoulos C. An input/output model reduction-based optimization scheme for large-scale systems. *Multiscale Modeling & Simulation* 2005;4(2):691–708.
- [18] Theodoropoulos C. Optimisation and linear control of large scale nonlinear systems: a review and a suite of model reduction-based techniques. In: *Coping with Complexity: Model Reduction and Data Analysis* Springer; 2011.p. 37–61.
- [19] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. *science* 1983;220(4598):671–680.
- [20] Chambers LD. *Practical handbook of genetic algorithms: complex coding systems*, vol. 3. CRC press; 2019.
- [21] Floudas CA, Akrotirianakis IG, Caratzoulas S, Meyer CA, Kallrath J. Global optimization in the 21st century: Advances and challenges. *Computers & Chemical Engineering* 2005;29(6):1185–1202.

- [22] Pires JCM, Martins FG, Sousa S, Alvim-Ferraz M, Pereira M. Selection and validation of parameters in multiple linear and principal component regressions. *Environmental Modelling & Software* 2008;23(1):50–55.
- [23] Lang Yd, Malacina A, Biegler LT, Munteanu S, Madsen JI, Zitney SE. Reduced order model based on principal component analysis for process simulation and optimization. *Energy & Fuels* 2009;23(3):1695–1706.
- [24] Henao CA, Maravelias CT. Surrogate-based superstructure optimization framework. *AIChE Journal* 2011;57(5):1216–1232.
- [25] Fahmi I, Cremaschi S. Process synthesis of biodiesel production plant using artificial neural networks as the surrogate models. *Computers & Chemical Engineering* 2012;46:105–123.
- [26] Tawarmalani M, Sahinidis NV. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming* 2005;103(2):225–249.
- [27] Schweidtmann AM, Mitsos A. Deterministic global optimization with artificial neural networks embedded. *Journal of Optimization Theory and Applications* 2019;180(3):925–948.
- [28] Tao M, Li J, Theodoropoulos C. Reduced model-based global optimisation of large-scale steady state nonlinear systems. In: *Computer Aided Chemical Engineering*, vol. 46 Elsevier; 2019.p. 1039–1044.
- [29] Houska B, Chachuat B. Global optimization in Hilbert space. *Mathematical programming* 2019;p. 1–29.
- [30] Faure H. On the star-discrepancy of generalized Hammersley sequences in two dimensions. *Monatshefte für Mathematik* 1986;101(4):291–300.
- [31] de Aguiar PF, Bourguignon B, Khots M, Massart D, Phan-Than-Luu R. D-optimal designs. *Chemometrics and intelligent laboratory systems* 1995;30(2):199–210.
- [32] Loh WL, et al. On Latin hypercube sampling. *Annals of statistics* 1996;24(5):2058–2080.
- [33] Boukouvala F, Floudas CA. ARGONAUT: AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems. *Optimization Letters* 2017;11(5):895–913.
- [34] Henao CA. A superstructure modeling framework for process synthesis using surrogate models. PhD thesis, The University of Wisconsin-Madison; 2012.
- [35] Eason J, Cremaschi S. Adaptive sequential sampling for surrogate model generation with artificial neural networks. *Computers & Chemical Engineering* 2014;68:220–232.
- [36] Liu H, Ong YS, Cai J. A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design. *Structural and Multidisciplinary Optimization* 2018;57(1):393–416.
- [37] Richardson M. Principal component analysis. URL: <http://people.maths.ox.ac.uk/richardsonm/SignalProcPCA.pdf> (last access: 35 2013) Aleš Hladnik Dr, Ass Prof, Chair of Information and Graphic Arts Technology, Faculty of Natural Sciences and Engineering, University of Ljubljana, Slovenia ales.hladnik@ntf.uni-lj.si 2009;6:16.
- [38] Hotelling H. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology* 1933;24(6):417.
- [39] Berkooz G, Holmes P, Lumley JL. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics* 1993;25(1):539–575.
- [40] Jirsa VK, Friedrich R, Haken H, Kelso JS. A theoretical model of phase transitions in the human brain. *Biological cybernetics* 1994;71(1):27–35.

- [41] Park S, Lee JJ, Yun CB, Inman DJ. Electro-mechanical impedance-based wireless structural health monitoring using PCA-data compression and k-means clustering algorithms. *Journal of intelligent material systems and structures* 2008;19(4):509–520.
- [42] Jensen KF, Ray WH. The bifurcation behavior of tubular reactors. *Chemical Engineering Science* 1982;37(2):199–222.
- [43] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural networks* 1989;2(5):359–366.
- [44] Yu H, Wilamowski BM. Levenberg-marquardt training. *Industrial electronics handbook* 2011;5(12):1.
- [45] MacKay DJ. A practical Bayesian framework for backpropagation networks. *Neural computation* 1992;4(3):448–472.
- [46] Misener R, Floudas CA. ANTIGONE: algorithms for continuous/integer global optimization of nonlinear equations. *Journal of Global Optimization* 2014;59(2-3):503–526.
- [47] Rehfeldt D, Koch T. SCIP-Jack—a solver for STP and variants with parallelization extensions: An update. In: *Operations Research Proceedings 2017 Springer*; 2018.p. 191–196.
- [48] Smith JD, Neto AA, Cremaschi S, Crunkleton DW. CFD-based optimization of a flooded bed algae bioreactor. *Industrial & Engineering Chemistry Research* 2012;52(22):7181–7188.
- [49] Johnson SG, The NLOpt nonlinear-optimization package, ab-initio. [mit.edu/nlopt](http://mit.edu/nlopt); 2015.
- [50] Keßler T, Mertens N, Kunde C, Nentwich C, Michaels D, Engell S, et al. Efficient global optimization of a novel hydroformylation process. In: *Computer Aided Chemical Engineering*, vol. 40 Elsevier; 2017.p. 2113–2118.
- [51] Misener R, Gounaris CE, Floudas CA. Global optimization of gas lifting operations: A comparative study of piecewise linear formulations. *Industrial & Engineering Chemistry Research* 2009;48(13):6098–6104.
- [52] Misener R, Floudas C. Piecewise-linear approximations of multidimensional functions. *Journal of optimization theory and applications* 2010;145(1):120–147.
- [53] Floudas CA. *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press; 1995.
- [54] Lee JH, Shin J, Realff MJ. Machine learning: Overview of the recent progresses and implications for the process systems engineering field. *Computers & Chemical Engineering* 2018;114:111–121.
- [55] Grimstad B, Andersson H. ReLU networks as surrogate models in mixed-integer linear programs. *Computers & Chemical Engineering* 2019;131:106580.
- [56] Belotti P, Liberti L, Lodi A, Nannicini G, Tramontani A. *Disjunctive inequalities: applications and extensions*. Wiley Encyclopedia of Operations Research and Management Science 2010;.
- [57] Wei Z, Li X, Xu L, Tan C. Optimization of operating parameters for low NO<sub>x</sub> emission in high-temperature air combustion. *Energy & Fuels* 2012;26(5):2821–2829.
- [58] Botoeva E, Kouvaros P, Kronqvist J, Lomuscio A, Misener R. Efficient verification of relu-based neural networks via dependency analysis. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34; 2020. p. 3291–3299.
- [59] Anderson R, Huchette J, Tjandraatmadja C, Vielma JP. Strong convex relaxations and mixed-integer programming formulations for trained neural networks. *arXiv preprint arXiv:181101988* 2018;.

# Chapter 4

## Robust model predictive control of large-scale distributed systems under uncertainty

### 4.1 Introduction

A model reduction based global optimisation framework for large-scale distributed parameter systems has been presented in Chapter 3. Several practical computational cases were employed to validate the computational advantages of the proposed framework. Successful computational practices of the previous near global optimisation methodology motivates the further applications in optimisation based control field. Control of distributed parameter systems is a meaningful but challenging online task in practical engineering and science areas. Efficient intelligent control strategies can significantly improve process performance and reduce the possibility of process safety problems. However, there are two key issues, preventing practical control policy for large-scale distributed parameters. One is process uncertainties such as significant parametric uncertainty and slight system disturbances, leading to plant-model mismatch. While the other one is the intensive online computations due to expensive evaluations of high dimensional systems from discretisation. Originated from these issues, a general computational methodology for robust nonlinear model predictive control is proposed in this work for large-scale distributed systems under parametric uncertainty.

Parametric uncertainty of large-scale complex systems needs to account for quantities of

interest, to robustly control systems and improve process performance. However, explicit solution formulation of the quantities of interest with respect to uncertain parameters is generally unavailable for most practical large-scale systems. Furthermore, direct sample-based uncertainty propagation, such as the traditional Monte Carlo methods, would lead to impractical computational costs for expensive distributed parameter systems. The popular polynomial chaos expansion method could be a promising solution for uncertainty propagation tasks. Through systematic data collection and choice of polynomial chaos corresponding to the type of uncertainty distribution, the low order polynomial chaos expansion approach could compute the probabilistic bounds and/or lower-upper bounds for the original highly complex systems with desirable precision, which could significantly reduce the computational costs of the uncertainty propagation process.

Nevertheless, model-based control of large-scale distributed parameter systems is still challenging due to the model-plant mismatch from system noises. A model predictive control strategy could be adopted to eliminate this error. Furthermore, Chapter 3 presents a systematic computational framework to calculate the near global optima of large-scale systems. Here, the similar framework will be applied to perform a real-time optimisation procedure within model predictive control, combining efficient data collection and model reduction. A combination of proper orthogonal decomposition and recurrent neural networks will be employed to construct accurate surrogate models that can predict the robust bounds of quantities of interest, computed by polynomial chaos based uncertainty propagation procedure.

The proposed robust model predictive control framework was tested through a typical chemical tubular reactor and a novel continuous bioreactor with cell recycle. The computational cases show that the presented computational framework can efficiently control the operating conditions to satisfy the safety and production requirements across time and space scales and to maintain high product quality for the running reaction systems.

## **4.2 Author's contribution**

**Min Tao**

Conceptualization-Equal, Methodology-Equal, Validation-Lead, Writing-original draft-

Lead

**Ioannis Zacharopoulos**

Investigation-Equal, Editing-Supporting

**Constantinos Theodoropoulos**

Conceptualization-Equal, Funding acquisition-Lead, Methodology-Equal, Project administration-Lead, Supervision-Lead, Writing-review and editing-Lead

# Robust model predictive control for large-scale distributed parameter systems under uncertainty

Min Tao, Ioannis Zacharopoulos and Constantinos Theodoropoulos

**Abstract**—Control of nonlinear distributed parameter systems (DPS) under uncertainty is a meaningful task for many industrial processes. However, both intrinsic uncertainty and high dimensionality of DPS require intensive computations, while non-convexity of nonlinear systems can inhibit the computation of global optima during the control procedure. In this work, polynomial chaos expansion (PCE) was used to account for the uncertainties in quantities of interest. Then the proper orthogonal decomposition (POD) method was adopted to project the high-dimensional nonlinear system dynamics onto a low-dimensional subspace, where recurrent neural networks (RNNs) were subsequently built to capture the reduced dynamics through a systematic data collection from the high-fidelity simulator. Finally, the reduced RNNs based model predictive control (MPC) would generate a set of sequential optimisation problems, of which near global optima could be computed through MILP reformulation techniques and advanced MILP solver. The effectiveness of the proposed framework is demonstrated through two case studies: a chemical tubular reactor and a cell-immobilisation packed-bed bioreactor for the bioproduction of succinic acid.

**Index Terms**—Nonlinear model predictive control, distributed parameter systems, uncertainty quantification, artificial neural network, model reduction, data-driven methodology

## I. INTRODUCTION

Spatial-temporal distributed parameter systems (DPS) exist widely in engineering practice [1], e.g in chemical [2], biochemical [3] and mechanical engineering [4]. Complex DPS usually exhibit uncertainty due to inherent stochastic and/or incomplete knowledge of processes [5], which lead to substantial model-plant mismatch. Efficient control strategies for large-scale complex DPS under uncertainty could speed up process production and ensure process safety [6]. Model predictive control (MPC), a popular advanced control method for multivariate plants with process constraints, reformulates the original optimal control problem (OCP) into a finite sequence of dynamic optimisation problems at each sampling time to obtain the corresponding control actions [7]. Feedback is introduced to this procedure through updating the state(s) of nonlinear dynamic models. In spite of both improved optimisation algorithms and fast hardware, the computational demand on MPC is still high for large-scale DPS problems. The high dimensionality of the discretised DPS results in slow computational speeds, which is impractical for real-time control, while

global optimisation of the non-convex MPC sub-problems is usually computationally intractable. In addition, black-box characteristics of high-fidelity commercial simulators [8], [9] prevent the direct utilisation of model-based computational techniques. Furthermore, dynamic model predictions are often significantly affected by parametric uncertainties [10], which may lead to wrong decision-making computed from MPC controllers. Therefore, efficient uncertainty quantification methods are of great importance for developing MPC strategies for large-scale DPS under parametric uncertainty. Uncertainty quantification procedures for DPS however, often require a large number of repeated computationally expensive evaluations. Thus, control of DPS under uncertainty is an important practical engineering challenge.

Uncertainty quantification (UQ) aims to measure the impact of uncertainty on quantities of interest [5]. The direct Monte Carlo (MC) sampling method is typically utilised to complete UQ tasks for generalised complex problems, generating a large number of realisations to accurately approximate the uncertainty distributions. Compared with the expensive MC method, both efficient sampling methods [11] and lower order model-based methods [10] are more powerful in terms of computational costs for complex large-scale systems. Efficient sampling methods, such as Latin hypercube sampling [11] and sparse grid methods [12], [13], [14], only require a few representative samples to be propagated utilising the original systems, which can greatly reduce computational demand. However, less samples would lead to lower computational accuracy for quantitative tasks. Lower order models on the other hand, including popular power series expansions (PSE) [15], [16] and polynomial chaos expansions (PCE) [17], can lighten the computational load by replacing the original expensive computational models [10], [18], which has been commonly employed to estimate the statistics for large-scale complex systems [19], [20]. Computational case studies have shown that the PCE method is faster and more accurate than the PSE method for large-scale thin film formation and heterogeneous catalytic flow problems [21], [19]. In this work, fast PCE-based uncertainty propagation method was employed to address parametric uncertainty for large-scale complex systems. Nevertheless, both high dimensionality and non-convexity issues are still computational barriers for applying MPC to large-scale distributed systems.

Model order reduction techniques are the most efficient methodologies to address the high dimensional issues for spatial-temporal distributed systems [22]. Together with Galerkin method [23], [24] or ANN surrogate models [25], the proper orthogonal decomposition (POD) approach, also named

The University of Manchester and China Scholarship Council joint scholarship (file no. 201706250031)

M. Tao, I. Zacharopoulos and C. Theodoropoulos are with the Department of Chemical Engineering and Analytical Science, University of Manchester, Manchester M13 9PL, U.K. (e-mail: min.tao@postgrad.manchester.ac.uk; ioannis.zacharopoulos@manchester.ac.uk; k.theodoropoulos@manchester.ac.uk)

as Karhunen-Loeve decomposition, can generate accurate low-dimensional models that can be efficiently used to perform control and optimisation tasks for large-scale distributed systems. Additionally, inertial manifolds or approximate inertial manifolds have been employed to construct stable controllers for dissipative partial differential equation (PDE) systems [26], [27]. Moreover, *equation-free* methodologies offer another effective model reduction approach for optimisation and control purposes [28], [29], [30], [31]. Equation-free methods utilise input/output data to compute dominant system eigendirections that *drive* the system dynamics as well as low-dimensional gradients that can accelerate computational evaluations for large-scale dissipative systems. A detailed discussion about model reduction based optimisation and control methodologies can be found in [32].

To overcome the non-convexity issues for large-scale nonlinear optimisation problems, both stochastic and deterministic methods can be used. Stochastic optimisation methods [33], [34] can perform global searches across the design space to avoid multiple local optima, but they can not guarantee optimal conditions. Deterministic global optimisation methods typically implement branch and bound procedures or their extensions to narrow the gap between low and upper bounds [35], and are often extremely expensive for large-scale problems but they can guarantee global optimality conditions.

The aim of this work is to construct a robust model predictive control framework for large-scale distributed systems under uncertainty. Firstly, polynomial chaos expansion approach is used to compute statistics for the quantities of interest of DPS. Then a combination of POD and recurrent neural networks (RNNs) is employed to capture the nonlinear dynamics of the calculated statistic moments and/or bounds. The resulting reduced ANN surrogate models can accurately represent the dynamics of the (black-box) large-scale systems and are efficiently implemented for MPC. In addition, the resulting optimisation sub-problems are globally optimised using advanced global optimisation solvers [36], [37], [38].

Since the high complexity of ANN structures leads to intractable computational problems for advanced global solvers, most previous studies of optimising surrogate ANN-based models focus on local optimisation [39] and/or small problems [40]. A reduced-space global optimisation algorithm [41] for ANN-based models was proposed to reduce computational costs and to allow an efficient nonlinear MPC formulation. [42]. The reduced-space global optimisation algorithm focuses more on the online reformulation techniques within global optimisation procedures for general ANN models. Offline reformulation strategies for constructing ANN models and choosing ANN structures with activation functions have not been involved to improve the computational performances. In our previous work [43], offline reformulation strategies including principal component analysis and deep rectifier neural networks were employed to build accurate but relatively simple ANN models for distributed parameter systems, which could be then formulated into mixed integer linear programming (MILP) problems and globally solved by advanced MILP solver CPLEX [44]. Here, offline reformulations techniques, POD and ReLU based RNN models were employed to de-

scribe the original high-dimensional nonlinear dynamics of statistic moments and/or bounds for the quantities of interest. Then the dynamic optimisation of distributed parameter systems could be reformulated into MILP problems solved by CPLEX, which could significantly reduce computational costs and capture the approximate global optima. Global NMPC studies have been performed in previous literature [45], [46], employing advanced global solvers and mixed integer programming strategies. The normalised multi-parametric disaggregation technique was utilised to compute upper and lower bounds in a spatial branch and bound formulation for global NMPC with multiple operating conditions [47]. Also, rigorous nonlinear MPC was reformulated into a sequence of mixed integer nonlinear programming problems, which were then solved globally [48]. However, all these works are limited to small and or medium size problems. To the best of our knowledge, global NMPC techniques have not been utilised for controlling large-scale DPS under uncertainty. In addition, no other work has been reported using a combination of double model reduction involving POD and RNN for high dimensional nonlinear systems, combined with PCE to address parametric uncertainty. The novelty of this work is to provide a PCE-POD-RNN based robust NMPC strategy for large-scale distributed systems under uncertainty, where the resulting surrogate model based optimisation sub-problems are globally solved. The performance of the proposed computational framework is validated via a receding horizon NMPC formulation for a chemical tubular reactor and a cell-immobilisation packed-bed biochemical reactor.

The rest of the paper is organized as follows. In Section 2, the robust NMPC strategy framework is proposed. Furthermore, the detailed theoretic basis and implementation are provided. In Section 3, the model based control framework is verified by practical chemical and biochemical reactors. In Section 4, conclusions and further applications are discussed.

## II. PCE-POD-RNN BASED ROBUST NMPC METHODOLOGY

In this part, the PCE-POD-RNN based robust nonlinear MPC strategy would be introduced. Firstly, the general optimal control problem is formulated. Then detailed polynomial chaos expansion, proper orthogonal decomposition and recurrent neural network parts are discussed, respectively. Finally, the general robust nonlinear model predictive control methodology is illustrated.

### A. Problem formulation

The general optimal control problem for PDE-based distributed parameter systems with parametric uncertainty:

$$\begin{aligned}
 & \min_{\lambda} \quad G(y, p, \lambda) \\
 & s.t. \quad \frac{\partial y}{\partial t} = D \left\{ \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2}, \dots, \frac{\partial^n y}{\partial x^n}, p, \lambda \right\} + R(y, p, \lambda) \\
 & \quad A \left\{ \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2}, \dots, \frac{\partial^n y}{\partial x^n}, p, \lambda \right\} \Big|_{x=\Omega'} = h_{bds}(y, p, \lambda) \quad (1) \\
 & \quad y|_{t=0} = y_0 \\
 & \quad g_{cons}(y, p, \lambda) \leq 0
 \end{aligned}$$

where  $G(y, p, \lambda) : \mathbb{R}^{N_y} \times \mathbb{R}^{N_p} \times \mathbb{R}^{N_\lambda} \rightarrow \mathbb{R}$  denotes the objective cost function for the OCP, and the constraints are the PDE-based system equations with corresponding boundary and initial conditions.  $t \in \mathbb{R}$  is the time dimension while  $x \in \mathbb{R}^{N_x}$  are the space dimensions,  $y(t, x) : \mathbb{R} \times \mathbb{R}^{N_x} \rightarrow \mathbb{R}^{N_y}$  are the state variables,  $p \in \mathbb{R}^{N_p}$  the uncertain parameters,  $\lambda(t) : \mathbb{R} \rightarrow \mathbb{R}^{N_\lambda}$  are the manipulated variables,  $D$  is the differential operator,  $R(y, p, \lambda)$  are the nonlinear parts of PDEs,  $A$  is the operator on the boundary conditions,  $\Omega'$  is the boundary,  $h_{bds}(y, p, \lambda) : \mathbb{R}^{N_y} \times \mathbb{R}^{N_p} \times \mathbb{R}^{N_\lambda} \rightarrow \mathbb{R}^{N_y}$  are the function values for the boundary conditions,  $y_0 \in \mathbb{R}^{N_y}$  are the initial values and  $g_{cons}(y, p, \lambda) : \mathbb{R}^{N_y} \times \mathbb{R}^{N_p} \times \mathbb{R}^{N_\lambda} \rightarrow \mathbb{R}^{N_{con}}$  are the  $N_{con}$  general constraints for state variables, manipulated variables and uncertain parameters. Here, we assumed that the uncertain parameters  $p \in \mathbb{R}^{N_p}$  were time-invariant.

In general, the analytical solutions for the above Eq.(1) are unavailable. Therefore, computing the numerical solutions is more practical. If we transform the above continuous dynamic systems into discrete ones, the discretised dynamic systems can be formulated as below:

$$\begin{aligned} y'_{\tau+1} &= f_{dis}(y'_\tau, \lambda'_\tau, p) + w_\tau \\ z_{\tau+1} &= h_{dis}(y'_{\tau+1}, \lambda'_\tau, p) + v_\tau \\ y'_0 &= y_0 \end{aligned} \quad (2)$$

where  $y'(t) \in \mathbb{R}^{N_y}$  are the high dimensional discretised state variables,  $y'_\tau \in \mathbb{R}^{N_y}$  the state variables at discretised time step  $\tau \in \mathbb{N}$ ,  $\lambda'_\tau \in \mathbb{R}^{N_\lambda}$  the manipulated values at discretised time step  $\tau$ ,  $z_\tau \in \mathbb{R}^{N_z}$  the measurement values at discretised time step  $\tau$ .  $y'_0$  are the discretised initial values.  $f_{dis} : \mathbb{R}^{N_y} \times \mathbb{R}^{N_\lambda} \times \mathbb{R}^{N_p} \rightarrow \mathbb{R}^{N_y}$  and  $h_{dis} : \mathbb{R}^{N_y} \times \mathbb{R}^{N_\lambda} \times \mathbb{R}^{N_p} \rightarrow \mathbb{R}^{N_z}$  denote the discrete time nonlinear dynamic systems and output equations, respectively while  $w_\tau \in \mathbb{R}^{N_w}$  and  $v_\tau \in \mathbb{R}^{N_v}$  represent the time-varying additional noises to state variables and measurements, respectively, which follow zero-mean normal distributions.

Then the time-space infinite dimensional original problem Eq.(1) can be reduced as a time-space finite dimensional large-scale stochastic (due to parametric uncertainty  $p$ ) nonlinear programming (NLP) problem:

$$\begin{aligned} \min_{\lambda'_\tau} \quad & G'(y'_\tau, z_\tau, p, \lambda'_\tau) \\ \text{s.t.} \quad & y'_{\tau+1} = f_{dis}(y'_\tau, \lambda'_\tau, p) + w_\tau, \tau = 0, 1, 2, \dots, k \\ & z_{\tau+1} = h_{dis}(y'_{\tau+1}, \lambda'_\tau, p) + v_\tau, \tau = 0, 1, 2, \dots, k \\ & y'_0 = y_0 \\ & g'_{cons}(y'_\tau, p, z_\tau, \lambda'_\tau) \leq 0, \tau = 0, 1, 2, \dots, k \end{aligned} \quad (3)$$

where  $G'$  is the discretised objective cost function,  $g'_{cons}$  denotes the general constraints for the discrete time state variables, manipulated variables, output measurements and uncertain parameters, and  $k$  is the number of time horizons. The cost function  $G'$  and the constraints  $g'_{cons}$  will have a stochastic representation due to the general parametric uncertainty  $p$ . In the following parts, a combination of PCE and POD-RNN techniques is employed to deal with the large-scale stochastic nonlinear programming Eq.(3).

## B. Polynomial chaos expansion

To address parametric uncertainty, probabilistic approaches describe it by employing a probability density function (PDF) [49]. PCE, one of the most efficient UQ methods, uses only a few system samples to construct accurate stochastic surrogate models. The key idea of PCE is to represent an arbitrary random variable  $g$  with finite second-order moments as a function of random variables  $\theta$  [50]:

$$\begin{aligned} g(\theta) &= \sum a_b \Theta_b(\theta) \\ \Theta_b(\theta) &= \prod_{i=1}^{N_\theta} \Theta_{b_i}(\theta_i) \end{aligned} \quad (4)$$

where  $\theta \in \mathbb{R}^{N_\theta}$  are random variables such as the random parameters  $p$  in this work,  $g$  is quantities of interest such as the state variables and output values in this work;  $\Theta_b : \mathbb{R}^{N_\theta} \rightarrow \mathbb{R}$  are multivariate orthogonal polynomials from tensor products of univariate polynomials  $\Theta_{b_i} : \mathbb{R} \rightarrow \mathbb{R}$  and  $a_b \in \mathbb{R}$  are the corresponding coefficients,  $b \in \mathbb{N}^{N_\theta}$  are multidimensional summation indices and  $b_i$  denotes the degree of each univariate polynomial  $\Theta_{b_i}(\theta_i)$  of  $\theta_i$ .

For generalised polynomial chaos, the choice of orthogonal polynomials significantly depends on the types of the probabilistic distributions of random variables  $\theta$ . For example, Hermite polynomials would commonly be chosen for normal distributions. The equation below gives the expression for univariate Hermite polynomials  $\Theta_{b_i}^H$  with respect to the standard Gaussian distribution  $\theta_i^g$ .

$$\Theta_{b_i}^H(\theta_i^g) = (-1)^{b_i} \exp\left(\frac{1}{2}\theta_i^{g2}\right) \frac{d^{b_i}}{d\theta_i^{g b_i}} \exp\left(-\frac{1}{2}\theta_i^{g2}\right) \quad (5)$$

One of the most important properties of multivariate orthogonal polynomials is orthogonality, i.e. generalised polynomial chaos terms are orthogonal to each other:

$$\langle \Theta_{b1}(\theta), \Theta_{b2}(\theta) \rangle = \begin{cases} \langle \Theta_{b1}^2(\theta) \rangle & , b1 = b2 \\ 0 & , b1 \neq b2 \end{cases} \quad (6)$$

where  $\langle \Theta_{b1}^2(\theta) \rangle$  are often known as constants, whose values depend on the chosen polynomial family and multidimensional summation indice  $b1$ .

If the inner product  $\langle, \rangle$  of any two random variables  $g_2(\theta), g_1(\theta)$  are defined in corresponding probability space:

$$\langle g_1(\theta), g_2(\theta) \rangle = \int_{\Omega'} g_1(\theta) g_2(\theta) \pi(\theta) d\theta \quad (7)$$

where  $\Omega'$  is integral space of inner product,  $\pi(\theta)$  is the probability density function.

Then the inner product between arbitrary random variable  $g(\theta)$  and any multivariate orthogonal polynomial chaos  $\Theta_{b1}(\theta)$ :

$$\begin{aligned} \langle \Theta_{b1}(\theta), g(\theta) \rangle &= \int_{\Omega'} \Theta_{b1}(\theta) g(\theta) \pi(\theta) d\theta \\ &= \int_{\Omega'} \Theta_{b1}(\theta) \sum a_b \Theta_b(\theta) \pi(\theta) d\theta \\ &= \sum a_b \int_{\Omega'} \Theta_{b1}(\theta) \Theta_b(\theta) \pi(\theta) d\theta \\ &= \sum a_b \langle \Theta_{b1}(\theta), \Theta_b(\theta) \rangle \\ &= a_{b1} \langle \Theta_{b1}^2(\theta) \rangle \end{aligned} \quad (8)$$

According to Eqs.(7-8), any coefficient of the generalised polynomial chaos terms can be computed as follows:

$$\begin{aligned} a_{b1} &= \frac{\langle \Theta_{b1}(\theta), g(\theta) \rangle}{\langle \Theta_{b1}^2(\theta) \rangle} \\ &= \frac{\int_{\Omega'} \Theta_{b1}(\theta) g(\theta) \pi(\theta) d\theta}{\langle \Theta_{b1}^2(\theta) \rangle} \end{aligned} \quad (9)$$

Here we adopt non-intrusive projection and quadrature methods calculate the integral term as Eq.(10) due to the black-box characteristics of nonlinear dynamic systems.

$$\int_{\Omega'} \Theta_{b1}(\theta) g(\theta) \pi(\theta) d\theta \approx \sum_{kk=1}^{N_{kk}} w_{kk} \Theta_{b1}(\theta_{kk}) g(\theta_{kk}) \quad (10)$$

where  $N_{kk}$  is the number of quadrature points,  $\theta_{kk}$  and  $w_{kk}$  are the sampling points and the corresponding weights, respectively, according to the quadrature rules.

Then, the PCE-based stochastic surrogate models are tractable and can be utilised to estimate the probabilistic distributions of state variables and output values. The mean value and covariance of quantities of interest  $g(\theta)$  can be calculated as below [51]:

$$\begin{aligned} g(\theta) &\approx \sum_{b=0}^L a_b \Theta_b(\theta) \\ \mu_g &= \mathbb{E}(g(\theta)) \approx a_0 \\ \sigma_g &= \text{Var}(g(\theta)) \approx \sum_{b=1}^L a_b^2 \langle \Theta_b^2(\theta) \rangle \end{aligned} \quad (11)$$

where  $L$  denotes the tensor product of the truncated order of the arbitrary random variable  $g(\theta)$ .

The specific probability limits, such as the worst bounds in low confidence levels under the defined probability  $\mathbb{P}'$ , could be evaluated through the PDFs obtained for  $g(\theta)$  as follows:

$$\begin{aligned} g^p(\theta) &= F_{cdf}^{-1}(\mathbb{P}') \\ \mathbb{P}'(g(\theta) < g^p(\theta)) &= F_{cdf}(g^p(\theta)) \end{aligned} \quad (12)$$

where  $g^p(\theta)$  are probability bounds for the random variable  $g(\theta)$ , and  $F_{cdf}$  the cumulative distribution function (CDF). In this work, we focus on the mean values and lower-upper bounds of state variables and output values as follows:

$$\begin{aligned} \mu_{y'_\tau} &= \mathbb{E}(y'_\tau) \\ \alpha_{y'_\tau}^{lo} &= F_{cdf, y'_\tau}^{-1}(\frac{1}{2}\beta) \\ \alpha_{y'_\tau}^{up} &= F_{cdf, y'_\tau}^{-1}(1 - \frac{1}{2}\beta) \\ \mu_{z_\tau} &= \mathbb{E}(z_\tau) \\ \alpha_{z_\tau}^{lo} &= F_{cdf, z_\tau}^{-1}(\frac{1}{2}\beta) \\ \alpha_{z_\tau}^{up} &= F_{cdf, z_\tau}^{-1}(1 - \frac{1}{2}\beta) \end{aligned} \quad (13)$$

where  $\mu_{y'_\tau}$ ,  $\alpha_{y'_\tau}^{lo}$ ,  $\alpha_{y'_\tau}^{up}$  are the mean value, lower and upper bounds of the discrete-time state variable  $y'_\tau$ , respectively;  $\mu_{z_\tau}$ ,  $\alpha_{z_\tau}^{lo}$ ,  $\alpha_{z_\tau}^{up}$  are then mean value, lower and upper bounds of discrete-time output values  $z_\tau$ , respectively;  $F_{y'_\tau}$  and  $F_{z_\tau}$  are the CDFs of  $y'_\tau$  and  $z_\tau$ ;  $\beta$  is the small parameter value

to compute lower and upper bounds with a high confidence level. In this work,  $\beta$  is set to be 0.05 with a 95 % level of confidence.

In this manner, the above large-scale stochastic nonlinear programming Eq.(3) can be transformed into the following Eq.(14):

$$\begin{aligned} \min_{\lambda'_\tau} \quad & G''(\mu_{y'_\tau}, \alpha_{y'_\tau}^{lo}, \alpha_{y'_\tau}^{up}, \mu_{z_\tau}, \alpha_{z_\tau}^{lo}, \alpha_{z_\tau}^{up}, \lambda'_\tau) \\ s.t. \quad & \mu_{y'_{\tau+1}} = f_\mu(\mu_{y'_\tau}, \alpha_{y'_\tau}^{lo}, \alpha_{y'_\tau}^{up}, \lambda'_\tau) + w_\tau, \tau = 0, 1, 2, \dots, k \\ & \alpha_{y'_{\tau+1}}^{lo} = f_{\alpha^{lo}}(\mu_{y'_\tau}, \alpha_{y'_\tau}^{lo}, \alpha_{y'_\tau}^{up}, \lambda'_\tau) + w_\tau, \tau = 0, 1, 2, \dots, k \\ & \alpha_{y'_{\tau+1}}^{up} = f_{\alpha^{up}}(\mu_{y'_\tau}, \alpha_{y'_\tau}^{lo}, \alpha_{y'_\tau}^{up}, \lambda'_\tau) + w_\tau, \tau = 0, 1, 2, \dots, k \\ & \mu_{z_{\tau+1}} = h_\mu(\mu_{y'_{\tau+1}}, \alpha_{y'_{\tau+1}}^{lo}, \alpha_{y'_{\tau+1}}^{up}, \lambda'_\tau) + v_\tau, \tau = 0, 1, 2, \dots, k \\ & \alpha_{z_{\tau+1}}^{lo} = h_{\alpha^{lo}}(\mu_{y'_{\tau+1}}, \alpha_{y'_{\tau+1}}^{lo}, \alpha_{y'_{\tau+1}}^{up}, \lambda'_\tau) + v_\tau, \tau = 0, 1, 2, \dots, k \\ & \alpha_{z_{\tau+1}}^{up} = h_{\alpha^{up}}(\mu_{y'_{\tau+1}}, \alpha_{y'_{\tau+1}}^{lo}, \alpha_{y'_{\tau+1}}^{up}, \lambda'_\tau) + v_\tau, \tau = 0, 1, 2, \dots, k \\ & \mu_{y'_0} = y_0 \\ & g''_{cons}(\mu_{y'_\tau}, \alpha_{y'_\tau}^{lo}, \alpha_{y'_\tau}^{up}, \mu_{z_\tau}, \alpha_{z_\tau}^{lo}, \alpha_{z_\tau}^{up}, \lambda'_\tau) \leq 0, \tau = 0, 1, 2, \dots, k+1 \end{aligned} \quad (14)$$

where  $G''$ : the deterministic objective formulation of the previous stochastic one  $G'$  through stochastic moments and bounds  $\mu_{y'_\tau}$ ,  $\alpha_{y'_\tau}^{lo}$ ,  $\alpha_{y'_\tau}^{up}$ ,  $\mu_{z_\tau}$ ,  $\alpha_{z_\tau}^{lo}$ ,  $\alpha_{z_\tau}^{up}$ .  $f_\mu, f_{\alpha^{lo}}, f_{\alpha^{up}}, h_\mu, h_{\alpha^{lo}}, h_{\alpha^{up}}$  and  $g''_{cons}$  are transformed formulations in similar manners.

### C. Proper orthogonal decomposition

Although the fast PCE-based uncertainty propagation method decreases the computational complexity of calculating stochastic moments and bounds, the reduced problem as illustrated in Eq.(14), is still black-box, high-dimensional and non-convex. Double model techniques, proper orthogonal decomposition (POD) and recurrent neural networks (RNNs), would be employed to capture the high-dimensional dynamics of stochastic moments and bounds (state variables and output values  $\mu_{y'_\tau}$ ,  $\alpha_{y'_\tau}^{lo}$ ,  $\alpha_{y'_\tau}^{up}$ ,  $\mu_{z_\tau}$ ,  $\alpha_{z_\tau}^{lo}$ ,  $\alpha_{z_\tau}^{up}$ ) obtained from the fast PCE method. The key idea of the POD method is to project the high-dimensional dynamics of the computed stochastic moments/bounds onto a low-dimensional subspace, which requires efficient sampling methods in order to collect enough representative snapshots for a range of parameter values. Details about the theory and applications of POD can be found in the literature [52]. Here Latin hypercube (LHC) sampling on the space of design variables  $\lambda'$ , combining PCE methods for discretised nonlinear dynamic system Eq.(2), was used to get high-dimensional dynamic data sets ( $D \in \mathbb{R}^{N_{\lambda'} \times N}$ ,  $[Y_{\mu_y} \in \mathbb{R}^{m \times N'_{y'}}, Y_{\alpha_y^{lo}} \in \mathbb{R}^{m \times N'_{y'}}, Y_{\alpha_y^{up}} \in \mathbb{R}^{m \times N'_{y'}}, Y_{\mu_z} \in \mathbb{R}^{m \times N'_{z'}}, Y_{\alpha_z^{lo}} \in \mathbb{R}^{m \times N'_{z'}}, Y_{\alpha_z^{up}} \in \mathbb{R}^{m \times N'_{z'}}]$ ), where  $m \in \mathbb{N}$  is the number of discrete interval points, which for distributed parameter systems tends to be a large number, and  $N \in \mathbb{N}$  is the number of samples,  $N'_{y'} = N_{y'} * N * k$  is the number of discrete time points for discretised variable  $y'$  while  $N'_{z'} = N_{z'} * N * k$  denotes the number of discrete time points for discretised variable  $z$ ,  $Y_{\mu_y}, Y_{\alpha_y^{lo}}, Y_{\alpha_y^{up}}, Y_{\mu_z}, Y_{\alpha_z^{lo}}, Y_{\alpha_z^{up}}$  are the high-dimensional dynamic data sets of stochastic moments and bounds  $\mu_y, \alpha_y^{lo}, \alpha_y^{up}, \mu_z, \alpha_z^{lo}, \alpha_z^{up}$ , respectively, through the

methods of PCE and snapshots. LHC method could systematically generate samples, covering the whole design space and maximizing the difference among the produced samples. Given relatively large number of samples, LHC strategy could fill the design space to represent the dynamic features of complex systems. More details about sampling techniques and discussions can be found in the literature [43].

For the high-dimensional dynamic data set  $Y_{\mu_y}$  over a finite spatial interval  $\Omega'' \in \mathbb{R}$ , the POD method aims to choose a "small" set of dominant POD dynamic modes  $P'_{\mu_y} = (p_1, p_2, \dots, p_{a_{\mu_y}}) \in \mathbb{R}^{a_{\mu_y} \times N'_{y'}}$  ( $a_{\mu_y} \in \mathbb{N}$  is the number of POD modes) through projecting the high-dimensional dynamics of  $Y_{\mu_y}$  onto the subspace of the  $a_{\mu_y}$  POD modes  $\mathbb{P}'$ .

$$U_1 = P'_{\mu_y} Y_{\mu_y} \quad (15)$$

Here  $U \in \mathbb{R}^{a_{\mu_y} \times N'_{y'}}$  is the projection of the original dynamic data  $Y_{\mu_y}$  onto  $\mathbb{P}'$  and  $P'_{\mu_y}$  is the orthogonal projector.

In the POD method, the matrix  $\mathbb{P}'$  could be constructed by the covariance matrix  $C_y \in \mathbb{R}^{N'_{y'} \times N'_{y'}}$  of the output data  $Y_{\mu_y}$ :

$$C_y = \frac{1}{N'_{y'} - 1} Y_{\mu_y} Y_{\mu_y}^T \quad (16)$$

Here we seek to maximise variance and minimise covariance between data, i.e. maximising its diagonal elements, while minimising the off-diagonal elements of  $C_y$ .

This is equivalent to implementing singular value decomposition (SVD) on  $C_y$ :

$$C_y = Z^T Z = \left( \frac{1}{\sqrt{N'_{y'} - 1}} Y_{\mu_y}^T \right)^T \left( \frac{1}{\sqrt{N'_{y'} - 1}} Y_{\mu_y}^T \right) = V_1 D_1 V_1^T \quad (17)$$

where  $D_1 \in \mathbb{R}^{N'_{y'} \times N'_{y'}}$  is a diagonal matrix whose diagonal elements are the eigenvalues of  $Z^T Z$  and  $V_1$  is the orthogonal matrix whose columns are the eigenvectors of  $Z^T Z$ , which as can be easily shown are equivalent to the POD modes of  $Y_{\mu_y}$ . In fact, we can keep the first  $a_{\mu_y}$  POD modes corresponding to the  $a_{\mu_y}$  dominant eigenvalues of  $C_y$ , where usually  $a_{\mu_y} \ll N'_{y'}$ , hence  $V \in \mathbb{R}^{N'_{y'} \times a_{\mu_y}}$  and  $D_1$  now contains only the  $a_{\mu_y}$  most dominant eigenvalues of the system,  $D_1 \in \mathbb{R}^{a_{\mu_y} \times a_{\mu_y}}$ . We can then set  $P'_{\mu_y} = V_1^T$  and perform data reduction through the projection in Eq.(15). The original dynamic data sample,  $Y_{\mu_y}$  can be reconstructed from the projected data:

$$Y_{\mu_y} = P'_{\mu_y}^T U_1 \quad (18)$$

And the inverse projection model (reconstruction models) could be obtained as:

$$\mu_{y'\tau} = P'_{\mu_y}^T u_{\mu_y} \quad (19)$$

where  $u_{\mu_y}$  is the reduced low-dimensional variables.

In the similar manner, the inverse projection model (reconstruction models) for  $\alpha_{y'\tau}^{lo}, \alpha_{y'\tau}^{up}, \mu_{z\tau}, \alpha_{z\tau}^{lo}, \alpha_{z\tau}^{up}$  could be com-

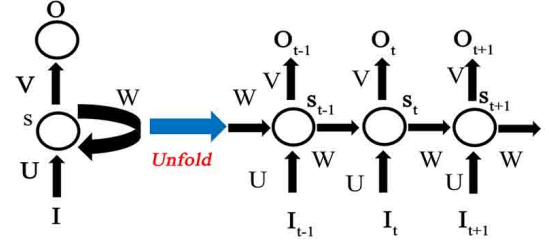


Fig. 1. From fold RNN to unfold RNN

puted:

$$\begin{aligned} \alpha_{y'\tau}^{lo} &= P'_{\alpha_y^{lo}}^T u_{\alpha_y^{lo}} \\ \alpha_{y'\tau}^{up} &= P'_{\alpha_y^{up}}^T u_{\alpha_y^{up}} \\ \mu_{z\tau} &= P'_{\mu_z}^T u_{\mu_z} \\ \alpha_{z\tau}^{lo} &= P'_{\alpha_z^{lo}}^T u_{\alpha_z^{lo}} \\ \alpha_{z\tau}^{up} &= P'_{\alpha_z^{up}}^T u_{\alpha_z^{up}} \end{aligned} \quad (20)$$

where  $P'_{\alpha_y^{lo}}, P'_{\alpha_y^{up}}, P'_{\mu_z}, P'_{\alpha_z^{lo}}, P'_{\alpha_z^{up}}$  are the corresponding orthogonal projectors to  $\alpha_{y'\tau}^{lo}, \alpha_{y'\tau}^{up}, \mu_{z\tau}, \alpha_{z\tau}^{lo}, \alpha_{z\tau}^{up}$ , respectively while  $u_{\alpha_y^{lo}}, u_{\alpha_y^{up}}, u_{\mu_z}, u_{\alpha_z^{lo}}, u_{\alpha_z^{up}}$  are the corresponding reduced variables.

#### D. Recurrent neural network

Previous POD model strategy has been employed to project the high-dimensional nonlinear dynamics of stochastic moments and bounds  $\mu_{y'\tau}, \alpha_{y'\tau}^{lo}, \alpha_{y'\tau}^{up}, \mu_{z\tau}, \alpha_{z\tau}^{lo}, \alpha_{z\tau}^{up}$  onto the subspaces of the low-dimensional dominant dynamics of  $u_{\mu_y}, u_{\alpha_y^{lo}}, u_{\alpha_y^{up}}, u_{\mu_z}, u_{\alpha_z^{lo}}, u_{\alpha_z^{up}}$  through the methods of snapshots. Then recurrent neural networks (RNNs) are adopted to capture the temporal coefficients on the low-dimensional subspace. In general, RNNs can be depicted as a folded computational graph, illustrated in Fig.(1), which is unfolded into a series of connected nodes. The mathematical formulation is given in the equation:

$$\begin{aligned} s_\tau &= f_{NN}(U I_\tau + W s_{\tau-1} + B_1) \\ O_\tau &= g_{NN}(V s_\tau + C_1) \end{aligned} \quad (21)$$

where  $s_\tau$  are the time series of the low-dimensional variables  $u$  while  $I_\tau$  are the time series of the manipulated variables  $\lambda'_\tau$ , and  $O_\tau$  are the output time series.  $f_{NN}$  and  $g_{NN}$  are RNN functions and output functions with weights  $U, W, V$  and biases  $B_1, C_1$ , respectively. Here we choose the purely linear activation function for the output function  $g_{NN}$  since this common choice allows large adjustments for output values.

Then multiple layers are set for RNNs because shallow NNs are easier to be over-fitted, which also requires large-scale structures (exponentially larger number of nodes in one hidden layer) to represent the nonlinear dominant dynamics, resulting in intensive computations for the upper-level optimisation and control. Since the traditional NN activation functions such as the sigmoid and the hyperbolic tangent functions, are highly non-convex, leading to multiple local optima, here the continuous piece-wise affine (PWA) activation function ReLU

is adopted, since it can reformulate the trained RNN-based optimisation problem into an MILP problem, which can be applied in an open-loop control mode and be solved using the advanced MILP solver CPLEX [44]. Here ReLU-RNNs are employed to represent the low-dimensional dynamics of  $u_{\mu_y}, u_{\alpha_y^{lo}}, u_{\alpha_y^{up}}, u_{\mu_z}, u_{\alpha_z^{lo}}, u_{\alpha_z^{up}}$ . Deep rectifier NN based MILP problems have been formulated in previous studies [53]. However, combination of uncertainty quantification, model reduction and deep rectifier NNs has not been reported in the literature. The mathematical equations of deep rectifier NNs for  $u_{\mu_y} = (u_{\mu_y,1}, u_{\mu_y,2}, \dots, u_{\mu_y,a_{\mu_y}}) \in \mathbb{R}^{a_{\mu_y}}$  can be reformulated into PWA functions through the big-M method [54] as the equation  $F_{u_{\mu_y}}(\lambda'_\tau, u_{\mu_y}) = 0$ .

In a similar manner, the trained RNN models for  $u_{\alpha_y^{lo}}, u_{\alpha_y^{up}}, u_{\mu_z}, u_{\alpha_z^{lo}}, u_{\alpha_z^{up}}$  can be computed as

$$\begin{aligned} F_{u_{\alpha_y^{lo}}}(\lambda'_\tau, u_{\alpha_y^{lo}}) &= 0 \\ F_{u_{\alpha_y^{up}}}(\lambda'_\tau, u_{\alpha_y^{up}}) &= 0 \\ F_{u_{\mu_z}}(\lambda'_\tau, u_{\mu_z}) &= 0 \\ F_{u_{\alpha_z^{lo}}}(\lambda'_\tau, u_{\alpha_z^{lo}}) &= 0 \\ F_{u_{\alpha_z^{up}}}(\lambda'_\tau, u_{\alpha_z^{up}}) &= 0 \end{aligned} \quad (22)$$

Therefore Eq.(14) can be transformed into the following large-scale deterministic programming problem:

$$\begin{aligned} \min_{\lambda'_\tau} \quad & G''(\mu_{y'\tau}, \alpha_{y'\tau}^{lo}, \alpha_{y'\tau}^{up}, \mu_{z\tau}, \alpha_{z\tau}^{lo}, \alpha_{z\tau}^{up}, \lambda'_\tau) \\ \text{s.t.} \quad & F_{u_{\mu_y}}(\lambda'_\tau, u_{\mu_y}) = 0 \\ & F_{u_{\alpha_y^{lo}}}(\lambda'_\tau, u_{\alpha_y^{lo}}) = 0 \\ & F_{u_{\alpha_y^{up}}}(\lambda'_\tau, u_{\alpha_y^{up}}) = 0 \\ & F_{u_{\mu_z}}(\lambda'_\tau, u_{\mu_z}) = 0 \\ & F_{u_{\alpha_z^{lo}}}(\lambda'_\tau, u_{\alpha_z^{lo}}) = 0 \\ & F_{u_{\alpha_z^{up}}}(\lambda'_\tau, u_{\alpha_z^{up}}) = 0 \\ & g''_{cons}(P_{\mu_y}^{T'} u_{\mu_y}, P_{\alpha_y^{lo}}^{T'} u_{\alpha_y^{lo}}, P_{\alpha_y^{up}}^{T'} u_{\alpha_y^{up}}, \\ & P_{\mu_z}^{T'} u_{\mu_z}, P_{\alpha_z^{lo}}^{T'} u_{\alpha_z^{lo}}, P_{\alpha_z^{up}}^{T'} u_{\alpha_z^{up}}, \lambda'_\tau) \leq 0, \\ & t = 0, 1, 2, \dots, k+1 \end{aligned} \quad (23)$$

So far, reduced surrogate models are built to represent the high-dimensional dynamics of stochastic moments and/or bounds for large-scale distributed parameter systems. The detailed PCE-POD-RNN model construction algorithm is summarised in Algorithm 1 below:

#### E. Nonlinear model predictive control

Eq.(23) gives the open-loop formulation of model reduction based OCP problems, which could be used as an important part of NMPC-based scheme because the solutions of closed-loop NMPC would mainly depend on the online solution of the open-loop problem Eq.(23). Here we assume only box constraints for  $g''_{cons}$ , which commonly include the bounds for manipulated variables and statistics of state and measurement

#### Algorithm 1 PCE-POD-RNN model construction algorithm

**Inputs:**  $N_1, N_2, w_\tau, v_\tau$

**Outputs:**  $P'_{\mu_y}, P'_{\alpha_y^{lo}}, P'_{\alpha_y^{up}}, P'_{\mu_z}, P'_{\alpha_z^{lo}}, P'_{\alpha_z^{up}}, F_{u_{\mu_y}}, F_{u_{\alpha_y^{lo}}}, F_{u_{\alpha_y^{up}}}, F_{u_{\mu_z}}, F_{u_{\alpha_z^{lo}}}, F_{u_{\alpha_z^{up}}}$

- 1: Generate  $N_1$  Latin hypercube samples of  $\lambda$  and collect the distributed dynamic trajectories with system and output noises  $w_\tau$  and  $v_\tau$
- 2: Compute model coefficients of PCE method for time-space trajectories as Eq.(9)
- 3: Generate  $N_2$  Monte Carlo samples (enough) of uncertain parameters  $P$
- 4: Compute distributed stochastic moments and/or bounds (quantities of interest) through the computed PCE models as Eq.(11-13)
- 5: Compute each POD global projectors for every stochastic moment and/or bound as Eq.(15-20)
- 6: Compute each RNN surrogate in the computed low-dimensional POD subspace as Eq.(21)
- 7: Check the accuracy of the generated POD-RNN models using fresh samples and dynamic trajectories
- 8: **if** Model is accurate **then**
- 9:   Go to Step 13
- 10: **else**
- 11:   Go back to Step 1 and add more samples or regenerate LHC samples
- 12: **end if**
- 13: **return** POD projectors and RNN models

variables as follows:

$$\begin{aligned} \underline{\mu_{y'}} &\leq P_{\mu_y}^{T'} u_{\mu_y} \leq \overline{\mu_{y'}} \\ \underline{\alpha_{y'}^{lo}} &\leq P_{\alpha_y^{lo}}^{T'} u_{\alpha_y^{lo}} \leq \overline{\alpha_{y'}^{lo}} \\ \underline{\alpha_{y'}^{up}} &\leq P_{\alpha_y^{up}}^{T'} u_{\alpha_y^{up}} \leq \overline{\alpha_{y'}^{up}} \\ \underline{\mu_z} &\leq P_{\mu_z}^{T'} u_{\mu_z} \leq \overline{\mu_z} \\ \underline{\alpha_z^{lo}} &\leq P_{\alpha_z^{lo}}^{T'} u_{\alpha_z^{lo}} \leq \overline{\alpha_z^{lo}} \\ \underline{\alpha_z^{up}} &\leq P_{\alpha_z^{up}}^{T'} u_{\alpha_z^{up}} \leq \overline{\alpha_z^{up}} \\ \underline{\lambda'} &\leq \lambda' \leq \overline{\lambda'} \\ \tau &= 0, 1, 2, \dots, k+1 \end{aligned} \quad (24)$$

where  $\underline{\mu_{y'}}, \underline{\alpha_{y'}^{up}}, \underline{\mu_z}, \underline{\alpha_z^{up}}, \underline{\lambda'}, \underline{\mu_{y'}}, \overline{\alpha_{y'}^{lo}}, \overline{\alpha_{y'}^{up}}, \overline{\mu_z}, \overline{\alpha_z^{lo}}, \overline{\alpha_z^{up}}, \overline{\lambda'}$  are lower and upper bounds of state variables and manipulated variables, respectively.

The objective function  $G''$  generally aims to minimize the quadratic functions of desired set-points and/or maximize the quantity of interest. In the following case study, we consider maximizing the exit concentration(s) of products, leading to a shrinking-horizon NMPC. In this way, the general model reduction-based open loop problems Eq.(23) can be reformulated into large-scale MILP problems, solved by the advanced MILP solver CPLEX 12.0.

Moreover, multiple validated RNN models can be employed to estimate the current statistical moments or bounds of the state variables and quantities of interest, leading to convenient

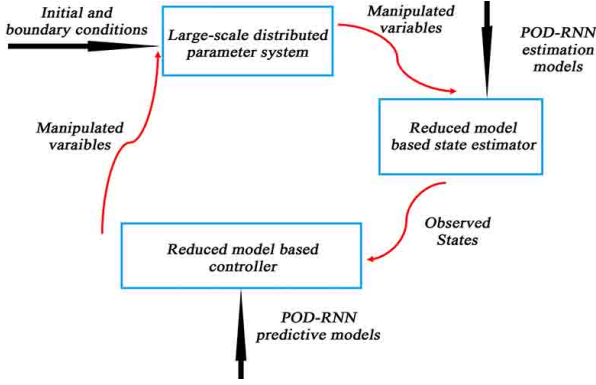
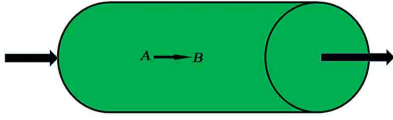


Fig. 2. Block diagram of reduced models based observer and NMPC structure

Fig. 3. An exothermic tubular reactor with reaction  $A \rightarrow B$ 

online state estimation using off-line computational models. Then a closed-loop scheme is implemented as shown in Fig.(2)

### III. CASE STUDIES

In this section, we use a tubular chemical reactor and a two-phase packed bed bioreactor to validate our PCE-POD-RNN-NMPC framework.

#### A. Chemical tubular reactor

This is a chemical engineering application: a tubular reactor as illustrated in Fig.(3), where an exothermic reaction takes place. The reactor model consists of 2 differential equations in dimensionless form as follows:

$$\begin{aligned} \frac{\partial C}{\partial t} &= \frac{1}{Pe_1} \frac{\partial^2 C}{\partial y_1^2} - \frac{\partial C}{\partial y_1} + Da(1-C)\exp(T/(1+T/\gamma)) \\ \frac{\partial C}{\partial t} &= \frac{1}{LePe_2} \frac{\partial^2 T}{\partial y_1^2} - \frac{1}{Le} \frac{\partial T}{\partial y_1} - \frac{\beta}{Le} T + \\ &BDa(1-C)\exp(T/(1+T/\gamma)) + \frac{\beta}{Le} T_w \end{aligned}$$

b.c.

$$\begin{aligned} \frac{\partial C}{\partial y_1} - Pe_1 C &= 0, \quad \frac{\partial T}{\partial y_1} - Pe_2 T = 0, \quad \text{at } y_1 = 0 \\ \frac{\partial C}{\partial y_1} &= 0, \quad \frac{\partial T}{\partial y_1} = 0, \quad \text{at } y_1 = 1 \end{aligned} \quad (25)$$

Here  $C$  and  $T$  are the dimensionless concentration and temperature, respectively.  $Da$  denotes the Damköhler number,  $Le$  is the Lewis number,  $Pe_1$  is the Peclet number for mass transport and  $Pe_2$  for heat transport,  $\beta$  a dimensionless heat transfer coefficient,  $C$  is the dimensionless adiabatic temperature rise,  $\gamma$  the dimensionless activation energy and  $y_1$  the dimensionless longitudinal coordinate. The system parameters

are  $Pe_1 = 5$ ,  $Pe_2 = 5$ ,  $Le = 1$ ,  $Da = 0.1$ ,  $\beta = 1.5$ ,  $\gamma = 10$ ,  $B = 12$ ;  $T_w$  is the adiabatic wall temperature of the cooling problem with the cooling zone. A simulator of the model in Eq.(25) was built with additional system and output noises, and solved through the *pde* solver in MATLAB with 200 space discretisation nodes and was subsequently used in inputs/outputs (black-box) mode. Here inputs include the manipulated variable  $T_w$  and two uncertain parameters  $Da \sim N(0.08, 0.008^2)$  and  $B \sim N(8, 0.8^2)$  while the outputs are 400 distributed concentration/temperature profiles. The system and outputs noises satisfy  $w_\tau \sim N(0, \text{diag}(0.00001 * 1 \in \mathbb{R}^{400}))$  and  $v'_\tau \sim N(0, \text{diag}(0.000001 * 1 \in \mathbb{R}^{400}))$ , respectively. Meanwhile the reporting time sampling interval was 0.4 for the dynamic simulator.

We aim to improve the performance of this chemical production process through an efficient control strategy under uncertainty based on the constructed black-box simulator. Specifically, The objective of the controller design is to maximize  $\mathbb{E}(C_{exit})$ , the mean value of concentration at the exit, satisfying the upper bound constraints of temperature  $T^{up}(y_1, t)$  across the whole reactor. Here Latin hypercube sampling ( $N_1 = 20$ ) was employed to collect enough representative trajectory samples through the black-box simulator. Second order polynomial chaos and 4000 realisations ( $N_2$ ) of uncertainty distributions were used to compute the high-dimensional dynamics of statistical moments  $\mathbb{E}(C(y_1, t))$  and bounds  $T^{up}(y_1, t)$ . Then double model reduction, First POD and then RNNs, was employed to generate the reduced models for online NMPC. Here only 2 dominate POD modes could capture 99.8 % of the system energy for both  $\mathbb{E}(C(y_1, t))$  and  $T^{up}(y_1, t)$ , which were then represented by the same size RNNs, 2 hidden layers (15 neurons, 15 neurons) ReLU-based RNNs. All computations were implemented in MATLAB R2019a on a Desktop (Intel Core(TM) i7-8700 CPU 3.2 292 GHz, 16 GB memory, 64-bit operating system, Windows 10).

To avoid over-fitting and under-fitting, the data set was randomly divided into a training, a validation and a test set with respective size ratios of 0.70: 0.15 : 0.15. The MATLAB Neural Network Toolbox was utilised to fit the weights  $U, W, V$  and biases  $B_1, C_1$  by minimizing the mean squared errors (MSE) on the training set using Levenberg-Marquardt algorithm and the early stopping procedure. To obtain the preferable NN structure (numbers of neurons and hidden layers), the training process was repeated until desired accuracy was obtained by adding more nodes and layers. Fig.(4) gives the predicted time profiles of  $\mathbb{E}(C(y_1, t))$  and  $T^{up}(y_1, t)$  at the exit of the tubular reactor while Fig.(5) compares the space profiles at steady state. Both time and space profiles show that the POD-RNN reduced models can approximate the true time-space dynamic process with a high accuracy.

Then the accurate reduce POD-RNN models could be employed into the model based state estimator and control strategy. Therefore the general open-loop OCP Eq.(23) could be applied as formulation Eq.(26) for the above chemical reaction process, which could be reformulated into MILP problems. The MILP problems would be iteratively solved by

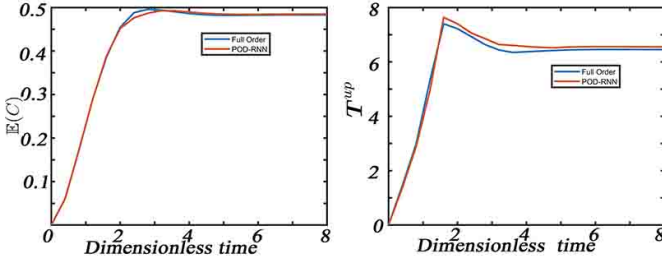


Fig. 4. Comparison of full order and POD-RNN time profiles of  $\mathbb{E}(C(y1,t))$  and  $T^{up}(y1,t)$  at the exit of the tubular reactor

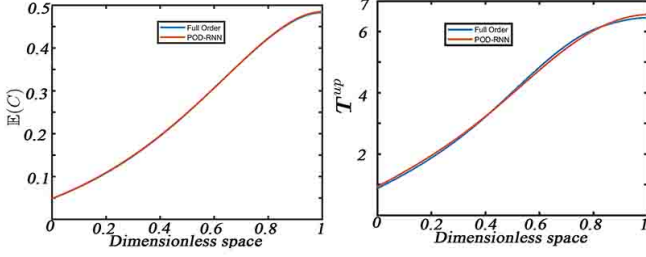


Fig. 5. Comparison of full order and POD-RNN space profiles of  $\mathbb{E}(C(y1,t))$  and  $T^{up}(y1,t)$  at steady state

CPLEX 12.0 in GAMS.

$$\begin{aligned}
 & \max_{T_{w,t}} \mathbb{E}(C_{exit}) \\
 & \text{s.t. RNN Equations;} \\
 & P'_{T_i} u_{T_i} \leq 4 - \varepsilon_1 \\
 & 0 \leq T_{w,t} \leq 2 \\
 & -\varepsilon_2 \leq T_{w,t} - T_{w,t-1} \leq \varepsilon_2 \\
 & t_n = 1, 2, \dots, k
 \end{aligned} \quad (26)$$

where  $T^{up}_t(y1)$  is the high-dimensional discrete upper bounds of distributed temperature at time horizon  $t$ .  $T_{w,t}$  is the discrete manipulated variable, the temperature of cooling zone.  $u_{T_i}^{up}$  denotes the low-dimensional variables of the projected  $T^{up}_t(y1)$  through the projector  $P'_{T_i}$ .  $\varepsilon_1$  is a relaxation parameter to decrease the impact of approximation errors.  $\varepsilon_2$  is a limitation parameter for smoothing the jump of manipulated variable. Here  $\varepsilon_1 = \varepsilon_2 = 0.1$ .

Therefore the reactor system, reduced model based state estimator and above open-loop controller make up of the closed-loop control scheme as Fig.(2) for this chemical reaction process. Running the close-loop control system, the close-loop optimal profiles were automatically generated as the dynamic profiles of the manipulated variable  $T_w$  in Fig.(6), time dynamic profiles Fig.(7) at the end of the tubular reactor and space profiles Fig.(8) at steady state for both  $\mathbb{E}(C(y1,t))$  and  $T^{up}(y1,t)$ . Then the robustness of the computational framework was tested through running the systems under the computed optimal control policy and 200 random realisations of uncertain parameters. The resulting concentration and temperature profiles are compared in Fig.(9) (time profile) at the exit of the reactor and Fig.(10) (space profile) at steady state. The time profiles of Fig.(9) illustrate that the random

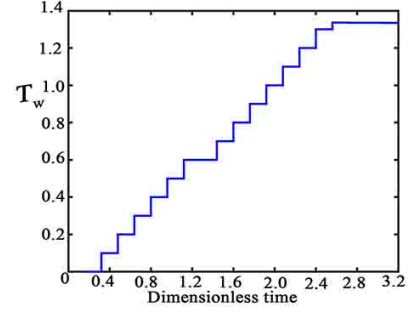


Fig. 6. Optimal control policy of heating temperature  $T_w$

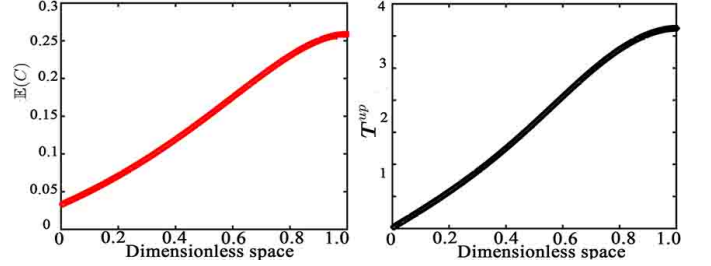


Fig. 7. Predictive space profiles of  $\mathbb{E}(C(y1,t))$  and  $T^{up}(y1,t)$  at the steady state under the optimal control policy

time trajectories of  $C(y1,t)$  at the exit surround the predicted  $\mathbb{E}(C(y1,t))$  as expected, while the trajectories of  $T(y1,t)$  mostly appear below the predicted  $T^{up}(y1,t)$ , except for short periods near  $t=0$ . One of possible reason for this discrepancy is that the errors coming from the model reduction steps are relatively big compared with the small state values at the initial stage.

Meanwhile, the behaviour of spatial profiles is much better. Across the whole length of reactor, spatially distributed concentrations fluctuate around the predicted  $\mathbb{E}(C(y1,t))$  profile. Meanwhile the spatially distributed temperature trajectories always appear below the predicted  $T^{up}(y1,t)$ . Furthermore, all time-space profiles obey the rigorous constraints  $T^{up}(y1,t) = P'_{T_i} u_{T_i} \leq 4$  for the upper bound variables  $T^{up}(y1,t)$ .

### B. Packed-bed bioreactor with immobilised cells

Here we extend the model-reduction based robust NMPC strategy to a more complex two-phase (two physical scales) packed bed-bioreactor shown in Fig.(11). Here a fermentation

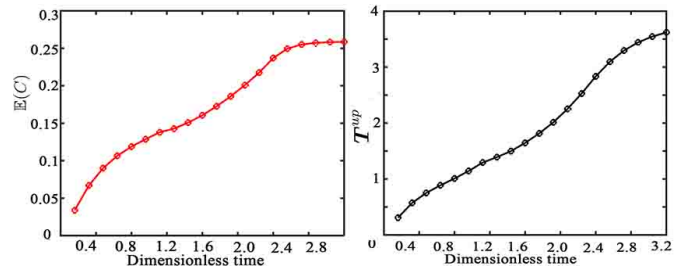


Fig. 8. Predictive time profiles of  $\mathbb{E}(C(y1,t))$  and  $T^{up}(y1,t)$  at the exit under the optimal control policy

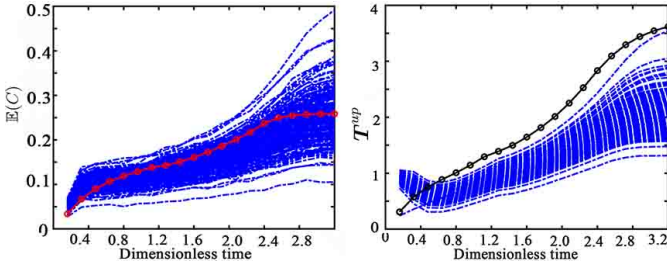


Fig. 9. Comparison between random time trajectories (blue dash) and statistic moments (red/black solid) of  $C(y1,t)$  and  $T(y1,t)$  at the exit under the optimal control policy

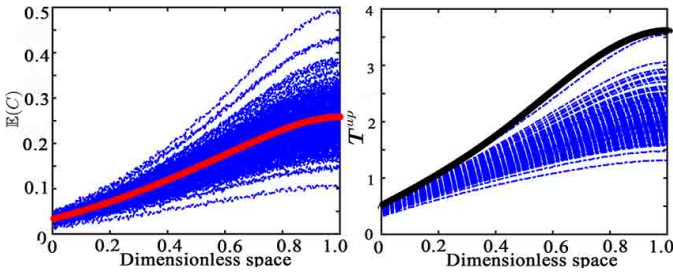


Fig. 10. Comparison between random space trajectories (blue dash) and statistic moments (red/black solid) of  $C(y1,t)$  and  $T(y1,t)$  at the steady state under the optimal control policy

process of glycerol to succinic, acetic and formic acid was optimally controlled. A two-phase model was built for the heterogeneous process, where one-dimensional PDEs Eq.(27) describe the transport-phenomena in the bulk phase of the reactor, while another set of steady-state PDEs Eq.(28) illustrate the reaction behaviours in the bead phase.

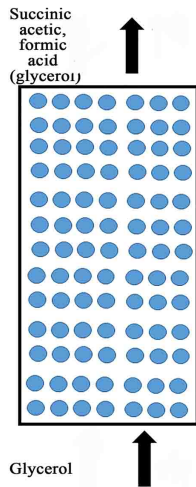


Fig. 11. Schematic diagram of packed bed bioreactor with immobilised cells

$$\begin{aligned}
 \frac{\partial x_{gly}}{\partial t} &= D_{gly} \frac{\partial^2 x_{gly}}{\partial y1^2} - \frac{v}{\epsilon} \frac{\partial x_{gly}}{\partial y1} - R_{tot_{gly}} \\
 \frac{\partial x_{sa}}{\partial t} &= D_{sa} \frac{\partial^2 x_{sa}}{\partial y1^2} - \frac{v}{\epsilon} \frac{\partial x_{sa}}{\partial y1} + R_{tot_{sa}} \\
 \frac{\partial x_{aa}}{\partial t} &= D_{aa} \frac{\partial^2 x_{aa}}{\partial y1^2} - \frac{v}{\epsilon} \frac{\partial x_{aa}}{\partial y1} + R_{tot_{aa}} \\
 \frac{\partial x_{fa}}{\partial t} &= D_{fa} \frac{\partial^2 x_{fa}}{\partial y1^2} - \frac{v}{\epsilon} \frac{\partial x_{fa}}{\partial y1} + R_{tot_{fa}} \\
 x_{gly} \Big|_{y1=0} &= x_{gly0}, \frac{\partial x_{gly}}{\partial y1} \Big|_{y1=L} = 0 \\
 x_{sa} \Big|_{y1=0} &= 0, \frac{\partial x_{sa}}{\partial y1} \Big|_{y1=L} = 0 \\
 x_{aa} \Big|_{y1=0} &= 0, \frac{\partial x_{aa}}{\partial y1} \Big|_{y1=L} = 0 \\
 x_{fa} \Big|_{y1=0} &= 0, \frac{\partial x_{fa}}{\partial y1} \Big|_{y1=L} = 0
 \end{aligned} \tag{27}$$

where  $x_{gly}$ ,  $x_{sa}$ ,  $x_{aa}$ ,  $x_{fa}$  are the concentrations of glucose, succinic, acetic and formic acid, respectively;  $\epsilon$  denotes the void fraction of the packed bed;  $v$  is the velocity of the fluid while  $R_{tot_{gly}}$ ,  $R_{tot_{sa}}$ ,  $R_{tot_{aa}}$  and  $R_{tot_{fa}}$  are the reaction source terms, respectively;  $D_{gly}$ ,  $D_{sa}$ ,  $D_{aa}$ ,  $D_{fa}$  are the diffusion coefficients of species in the fluid, respectively.

$$\begin{aligned}
 \frac{d}{dr} \left[ r^2 D'_{gly} \frac{\partial x'_{gly}}{\partial r} \right] - r^2 \cdot r_{gly} &= 0 \\
 \frac{d}{dr} \left[ r^2 D'_{sa} \frac{\partial x'_{sa}}{\partial r} \right] + r^2 \cdot r_{sa} &= 0 \\
 \frac{d}{dr} \left[ r^2 D'_{aa} \frac{\partial x'_{aa}}{\partial r} \right] + r^2 \cdot r_{aa} &= 0 \\
 \frac{d}{dr} \left[ r^2 D'_{fa} \frac{\partial x'_{fa}}{\partial r} \right] + r^2 \cdot r_{fa} &= 0 \\
 x'_{gly} \Big|_{y1=0} &= x_{gly}, \frac{\partial x'_{gly}}{\partial y1} \Big|_{y1=L} = 0 \\
 x'_{sa} \Big|_{y1=0} &= x_{sa}, \frac{\partial x'_{sa}}{\partial y1} \Big|_{y1=L} = 0 \\
 x'_{aa} \Big|_{y1=0} &= x_{aa}, \frac{\partial x'_{aa}}{\partial y1} \Big|_{y1=L} = 0 \\
 x'_{fa} \Big|_{y1=0} &= x_{fa}, \frac{\partial x'_{fa}}{\partial y1} \Big|_{y1=L} = 0
 \end{aligned} \tag{28}$$

where  $D'_{gly}$ ,  $D'_{sa}$ ,  $D'_{aa}$ ,  $D'_{fa}$  are the diffusion coefficients of species in the beads, respectively;  $r_{gly}$ ,  $r_{sa}$ ,  $r_{aa}$ ,  $r_{fa}$  are the

reaction rates of species in the beads, respectively.

$$\begin{aligned}
 R_{toti} &= \int_0^R r_i 4\pi R^2 dR \cdot \rho_{bead} (1 - \varepsilon), \forall i \in \{gly, sa, aa, fa\} \\
 r_i &= (\alpha_i \mu + \beta_i) \cdot X_{cons}, \forall i \in \{gly, sa\} \\
 r_i &= \beta_i \cdot X_{cons}, \forall i \in \{fa, aa\} \\
 \mu &= \mu_{max} \cdot \frac{x'_{gly}}{K_{S_{gly}} + x'_{gly} + \frac{x'^2_{gly}}{K_{I_{gly}}}} \cdot \frac{x_{CO_2}}{K_{x_{CO_2}} + x_{CO_2}} \cdot \left(1 - \frac{x'_{sa}}{SA^*}\right)^{n_{SA}}
 \end{aligned} \quad (29)$$

where  $R$  is the radius of alginate beads,  $X_{cons}$  denotes the biomass concentration inside the alginate beads;  $\rho_{bead}$  is the density of beads;  $\mu$ ,  $\alpha_i$  and  $\beta_i$  denote the coefficients of reaction rates of species;  $x_{CO_2}$  is the concentration of  $CO_2$ ;  $K_{I_{gly}}$ ,  $K_{S_{gly}}$ ,  $K_{x_{CO_2}}$ ,  $SA^*$  and  $n_{SA}$  are the coefficients of intrinsic kinetics.

Here  $v=0.1$ ;  $L=20$ ,  $D_{gly}=0.01$ ,  $D_{sa}=0.01$ ,  $D_{aa}=0.02$ ,  $D_{fa}=0.02$ ,  $\varepsilon=0.55$ ,  $\rho_{bead}=2.12$ ,  $R=0.15$ ,  $X_{cons}=0.21$ ,  $\alpha_{gly}=2.39$ ,  $\alpha_{sa}=4.5$ ,  $\beta_{gly}=0.187$ ,  $\beta_{sa}=0.21$ ,  $\beta_{fa}=0.011$ ,  $\beta_{aa}=0.0055$ ,  $\mu_{max}=0.2568$ ,  $K_{S_{gly}}=5.4$ ,  $K_{I_{gly}}=119.99$ ,  $x_{CO_2}=0.03$ ,  $K_{x_{CO_2}}=0.03$ ,  $SA^*=45.6$ ,  $n_{SA}=5$ ,  $D'_{sa}=0.00989$ ,  $D'_{fa}=0.01835$ ,  $D'_{aa}=0.01384$ ;

A simulator of the model in Eqs.(27-29) was built with additional system and output noises, and solved through the ODE 113 solver in MATLAB with 100 space discretisation nodes and was subsequently used in inputs/outputs (black-box) mode. Here inputs include the manipulated variable nominal substrate glycerol concentration  $\bar{x}_{gly0}$  and two uncertain parameters the true substrate glycerol concentration  $x_{gly0} \sim U(\bar{x}_{gly0} - 2.5, \bar{x}_{gly0} + 2.5)$  and the diffusion coefficients of glycerol in bead phase  $D'_{gly} \sim U(0.008, 0.012)$ . while the outputs are 400 distributed reactant/product (glycerol, succinic, acetic and formic acid) concentration profiles. The system and outputs noises follow  $w_\tau \sim N(0, \text{diag}(0.000001 * 1 \in \mathbb{R}^{400}))$  and  $v_\tau \sim N(0, \text{diag}(0.000001 * 1 \in \mathbb{R}^{400}))$ , respectively. Meanwhile the reporting time sampling interval was 15 hours for the dynamic simulator.

We aim to improve the performance of this biochemical production process through an efficient control strategy under uncertainty based on the built black-box simulator. Specifically, the objective of controller design is to maximize the expected main product (succinic acid) concentration  $\mathbb{E}(x_{sa}^{exit}(t))$  at the exit while observing upper bound concentration ( $x_{aa}^{up}(y1, t)$  and  $x_{fa}^{up}(y1, t)$ ) constraints of byproducts acetic and formic acid in the whole bioreactor length. Similar to the settings for the previous tubular reactor case study, Latin hypercube sampling ( $N_1 = 20$ ) to collect enough representative samples using the constructed system simulator in black-box mode. Here second order polynomial chaos and 4000 realisations ( $N_2$ ) of uncertain parameters were used to perform the uncertainty propagation procedure. Then a single RNN (2 hidden layers, 15 neurons and 15 neurons) model was employed to capture the dynamics of the mean value of succinic acid at the exit, while POD-RNN double models were built for time-space upper bound profiles of acetic and formic acid. Only 2 dominant POD modes are enough to capture 99.8 % energy of acetic and formic acid dynamics. In addition, two RNNs (2 hidden layers with 15 neurons and 15 neurons, and 2 hidden layers with

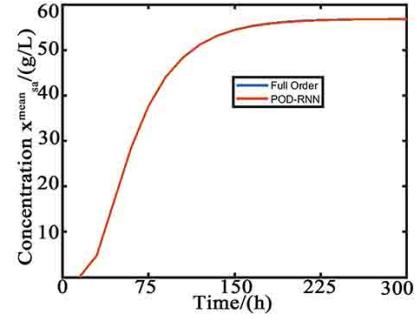


Fig. 12. Comparison of expected succinic acid concentration from full systems and predictive POD-RNN models at the exit

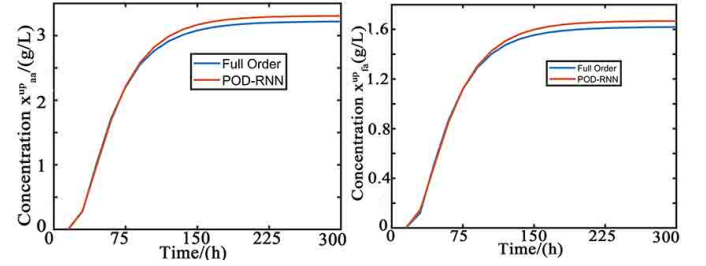


Fig. 13. Comparison of upper bound acetic and formic acid concentrations from full systems and predictive POD-RNN models at the exit of the reactor

20 neurons and 20 neurons) were utilised to construct the low-dimensional dynamic models for acetic and formic acid, respectively. The training process setting was similar to the previous implementation for the chemical tubular reactor. Then Fig.(12-14) compare the product concentrations from the full system and predictive POD-RNN models. Fig.(12) illustrates that the predicted expected succinic acid concentrations are extremely close to the ones computed from the original system, indicating the high accuracy of the reduced model prediction. In addition, there are only small errors (less than 3 % on the stable stage) between the predictions of the reduced and the full model as can be seen in Fig.(13-14))h accuracy of the POD-RNN models for the computation of acetic and formic acid profiles. In general, the constructed reduced surrogate RNN model could accurately predict the complex bioprocess behaviours. Then the accurate reduce POD-RNN models could be employed into the model based state estimator and control strategy. Therefore the general open-loop OCP Eq.(23) could be applied as formulation Eq.(30) for the above

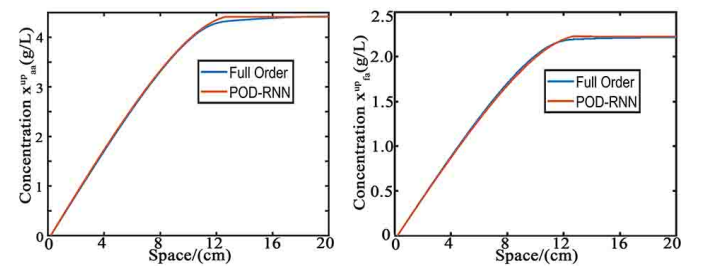


Fig. 14. Comparison of upper bound acetic and formic acid concentrations from full systems and predictive POD-RNN models at the steady state

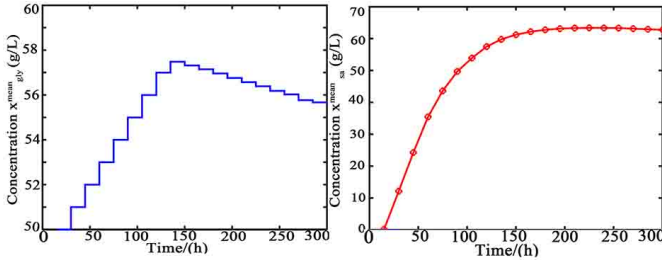


Fig. 15. Diagram of the optimal control policy and expected succinic acid concentration at the exit

fermentation process, which could be reformulated into MILP problems.

$$\begin{aligned}
 & \max_{\bar{x}_{gly0}(t)} \mathbb{E}(x_{sa}^{exit}) \\
 & \text{s.t. RNN} \quad \text{Equations;} \\
 & \quad P'_{x_{aa}}{}^T u_{x_{aa}}^{up} \leq 4.1 - \epsilon'_1 \\
 & \quad P'_{x_{fa}}{}^T u_{x_{fa}}^{up} \leq 2.1 - \epsilon'_3 \\
 & \quad 50 \leq \bar{x}_{gly0}(t) \leq 70 \\
 & \quad -\epsilon'_2 \leq \bar{x}_{gly0}(t) - \bar{x}_{gly0}(t-1) \leq \epsilon'_2 \\
 & \quad t = 1, 2, \dots, k
 \end{aligned} \tag{30}$$

where  $\epsilon'_1 = 0.1$  and  $\epsilon'_3 = 0.1$  are the relaxation values to decrease the impact of approximation errors.  $\epsilon'_2 = 1$  is an additional limitation value for smoothing the jump of manipulated variable.  $u_{x_{aa}}^{up}$  and  $u_{x_{fa}}^{up}$  are the low dimensional variables of the projected  $x_{aa}^{up}$  and  $x_{fa}^{up}$  with the projectors  $P'_{x_{aa}}$  and  $P'_{x_{fa}}$ , respectively.

Therefore the reactor system, reduced model based state estimator and above open-loop controller make up of the closed-loop control scheme as Fig.(2) for this biochemical fermentation process. Running the close-loop control system, the close-loop optimal profiles were automatically generated. Fig.(15) depicts the generated optimal control policy of the manipulated variable substrate glycerol concentration and the dynamic profile of the expected predicted succinic acid at the exit under the control policy. The substrate concentration first went up quickly and then gradually decreased beyond 150 hours, possibly in order to satisfy the requirements for byproduct concentrations. Meanwhile, the computed succinic acid concentration went through initial fast dynamics and then reached steady state. In addition, Fig.(16-17) illustrate the time-space profiles of byproducts acetic and formic acid with similar dynamic behaviours. Time profiles at the exit of the reactor (Fig.(16)), showed that the byproduct concentrations grew up quickly, and then became more stable with a little bit final decrease. While space profiles at steady-state stage (Fig.(17)) illustrate that the byproduct concentrations increased dramatically along with the reactor and then went down slightly, indicating the nonlinear spatial distributions of byproduct concentrations.

Then the robustness of the proposed control framework was tested through running the systems under 100 random realisations of uncertain parameters within the computed op-

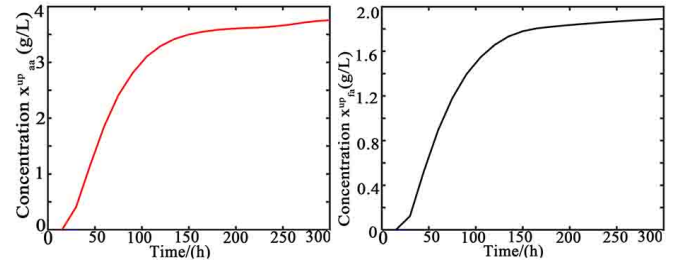


Fig. 16. Comparison upper bound acetic and formic acid concentrations at the end of reactor under optimal control policy

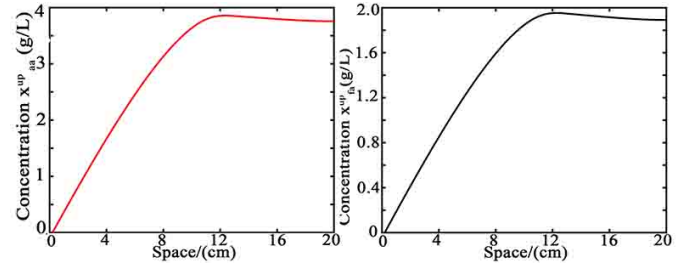


Fig. 17. Comparison upper bound acetic and formic acid concentrations at the steady state under optimal control policy

timal control policy. The resulting succinic acid concentration profile is illustrated in Fig.(18). Generally, the computed concentration profiles are as expected, with the exception of a small overestimation during the fast dynamic period and a small underestimation for the steady state value, which may be caused by the model reduction steps.

Moreover, the computed acetic and formic acid concentration are displayed in Fig.(19-20). Almost all of random trajectories followed the dynamic profiles below the predicted upper bounds except for an extremely small initial period for the formic acid and a small period at the end of the exponential phase for both formic and acetic acids. Meanwhile, the gap between the predicted bounds and the random trajectory profiles was not distinct, especially when the profiles were close to the exit of reactor or steady state, which may be caused by the model reductions. That indicates that the additional relaxation parameters  $\epsilon_1$  and  $\epsilon_2$  are necessary to reduce the impact of model errors on the rigorous product quality constraints. Without the relaxation parameters, the limitation constraints of byproducts may be destroyed due to the errors from model reductions. Furthermore, all of the 100 random time-space trajectories could satisfy the upper bound constraints  $x_{aa}^{up}(y1, t) = P'_{x_{aa}}{}^T u_{x_{aa}}^{up} \leq 4.1, x_{fa}^{up}(y1, t) = P'_{x_{fa}}{}^T u_{x_{fa}}^{up} \leq 2.1$ , implying the rigorous product quality constraint could be guaranteed with the relation parameters under the computed optimal control policy.

#### IV. CONCLUSION

This work has proposed a robust model predictive control strategy for large-scale distributed systems under parametric uncertainty. Firstly, the PCE method is used to efficiently compute the high-dimensional dynamics of statistic moments and probabilistic bounds for time-space state variables and

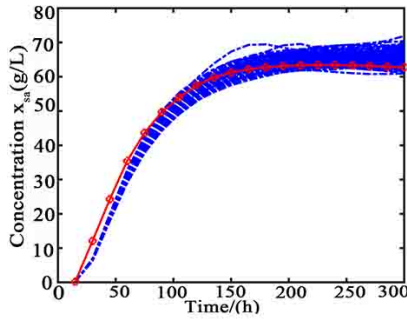


Fig. 18. Comparison between random space trajectories (blue dash) and predictive mean value of succinic concentration (red solid)  $x_{sa}$  at the end of reactor under the optimal control policy

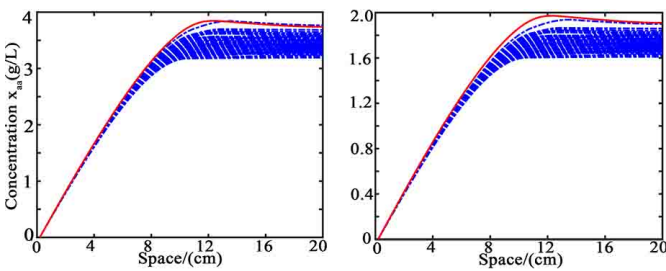


Fig. 19. Comparison between random space trajectories (blue dash) and upper bounds (red/black solid) of  $x_{aa}(y,t)$  and  $x_{fa}(y,t)$  at the steady state under the optimal control policy

measurement outputs. Then, the double model reductions, POD and RNN, are employed to construct simple but accurate predictive models for the nonlinear dynamics of statistic moments and upper/lower bounds. Next the MILP models are used to reformulate online optimisation scheme and solved globally within the NMPC. The proposed methodology is verified by a typical chemical tubular reactor and a packed bed bioreactor with immobilised cells for production. The two cases shows that the proposed NMPC strategy could efficiently improve process production and also satisfy the requirements of process safety (time-space temperature constraints) and product quality (time-space byproduct constraints). In the future, less conservative strategy would be exploited to further enhance the process performances. Moreover, online adaptive control was considered when offline samples are not enough.

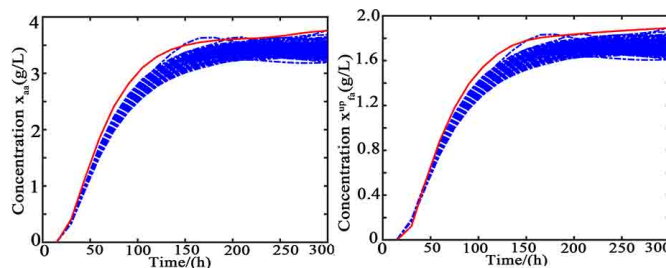


Fig. 20. Comparison between random time trajectories (blue dash) and upper bounds (red/black solid) of  $x_{aa}(y,t)$  and  $x_{fa}(y,t)$  at the exit under the optimal control policy

## REFERENCES

- [1] N. Ahmed, "Distributed parameter systems," *Encyclopedia of physical science and technology*, vol. 1, p. 277, 1987.
- [2] M. Tao, K. Guo, Z. Huang, H. Liu, and C. Liu, "A hybrid optimization method to design shapes of three-dimensional flow channels," *Chemical Engineering Research and Design*, vol. 114, pp. 190–201, 2016.
- [3] S. Park and Y. Li, "Integration of biological kinetics and computational fluid dynamics to model the growth of nanochloropsis salina in an open channel raceway," *Biotechnology and bioengineering*, vol. 112, no. 5, pp. 923–933, 2015.
- [4] G.-L. Yang, W.-H. Zhou, and F. Liu, "Simulation of flow field of high-pressure water-jet from nozzle with fluent [j]," *Journal of Lanzhou University of Technology*, vol. 34, no. 2, pp. 49–52, 2008.
- [5] T. J. Sullivan, *Introduction to uncertainty quantification*. Springer, 2015, vol. 63.
- [6] G. Stephanopoulos and C. Han, "Intelligent systems in process engineering: A review," *Computers & Chemical Engineering*, vol. 20, no. 6–7, pp. 743–791, 1996.
- [7] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.
- [8] C. Multiphysics, "Introduction to comsol multiphysics@," *COMSOL Multiphysics*, Burlington, MA, accessed Feb, vol. 9, p. 2018, 1998.
- [9] A. Fluent, "Ansys fluent," *Academic Research. Release*, vol. 14, 2015.
- [10] Z. Nagy and R. D. Braatz, "Distributional uncertainty analysis using power series and polynomial chaos expansions," *Journal of Process Control*, 2007.
- [11] M. C. Dunn, B. Shotorban, and A. Frendi, "Uncertainty quantification of turbulence model coefficients via latin hypercube sampling method," *Journal of fluids engineering*, vol. 133, no. 4, 2011.
- [12] F. Nobile, R. Tempone, and C. G. Webster, "A sparse grid stochastic collocation method for partial differential equations with random input data," *SIAM Journal on Numerical Analysis*, vol. 46, no. 5, pp. 2309–2345, 2008.
- [13] —, "An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data," *SIAM Journal on Numerical Analysis*, vol. 46, no. 5, pp. 2411–2442, 2008.
- [14] G. Lin, A. M. Tartakovsky, and D. M. Tartakovsky, "Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids," *Journal of Computational Physics*, vol. 229, no. 19, pp. 6995–7012, 2010.
- [15] Z. K. Nagy and R. D. Braatz, "Worst-case and distributional robustness analysis of finite-time control trajectories for nonlinear distributed parameter systems," *IEEE Transactions on Control Systems Technology*, vol. 11, no. 5, pp. 694–704, 2003.
- [16] —, "Open-loop and closed-loop robust optimal control of batch processes using distributional and worst-case analysis," *Journal of process control*, vol. 14, no. 4, pp. 411–422, 2004.
- [17] M. Eldred, "Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design," in *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No.*, 2009, p. 2274.
- [18] E. Bradford and L. Imsland, "Output feedback stochastic nonlinear model predictive control for batch processes," *Computers & Chemical Engineering*, vol. 126, pp. 434–450, 2019.
- [19] D. Chaffart and L. A. Ricardez-Sandoval, "Robust dynamic optimization in heterogeneous multiscale catalytic flow reactors using polynomial chaos expansion," *Journal of Process Control*, vol. 60, pp. 128–140, 2017.
- [20] —, "Robust optimization of a multiscale heterogeneous catalytic reactor system with spatially-varying uncertainty descriptions using polynomial chaos expansions," *The Canadian Journal of Chemical Engineering*, vol. 96, no. 1, pp. 113–131, 2018.
- [21] G. Kimaev and L. A. Ricardez-Sandoval, "A comparison of efficient uncertainty quantification techniques for stochastic multiscale systems," *AIChE Journal*, vol. 63, no. 8, pp. 3361–3373, 2017.
- [22] W. H. Schilders, H. A. Van der Vorst, and J. Rommes, *Model order reduction: theory, research aspects and applications*. Springer, 2008, vol. 13.
- [23] A. Theodoropoulou, R. A. Adomaitis, and E. Zafiriou, "Model reduction for optimization of rapid thermal chemical vapor deposition systems," *IEEE Transactions on Semiconductor Manufacturing*, vol. 11, no. 1, pp. 85–98, 1998.

- [24] W. Xie, I. Bonis, and C. Theodoropoulos, "Off-line model reduction for on-line linear mpc of nonlinear large-scale distributed systems," *Computers & chemical engineering*, vol. 35, no. 5, pp. 750–757, 2011.
- [25] —, "Data-driven model reduction-based nonlinear mpc for large-scale distributed parameter systems," *Journal of Process Control*, vol. 35, pp. 50–58, 2015.
- [26] P. D. Christofides and P. Daoutidis, "Finite-dimensional control of parabolic pde systems using approximate inertial manifolds," *Journal of mathematical analysis and applications*, vol. 216, no. 2, pp. 398–420, 1997.
- [27] P. D. Christofides, "Robust control of parabolic pde systems," *Chemical Engineering Science*, vol. 53, no. 16, pp. 2949–2965, 1998.
- [28] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, C. Theodoropoulos *et al.*, "Equation-free, coarse-grained multi-scale computation: Enabling microscopic simulators to perform system-level analysis," *Communications in Mathematical Sciences*, vol. 1, no. 4, pp. 715–762, 2003.
- [29] E. Luna-Ortiz and C. Theodoropoulos, "An input/output model reduction-based optimization scheme for large-scale systems," *Multi-scale Modeling & Simulation*, vol. 4, no. 2, pp. 691–708, 2005.
- [30] I. Bonis and C. Theodoropoulos, "Model reduction-based optimization using large-scale steady-state simulators," *Chemical engineering science*, vol. 69, no. 1, pp. 69–80, 2012.
- [31] P. Petsagkourakis, I. Bonis, and C. Theodoropoulos, "Reduced order optimization of large-scale nonlinear systems with nonlinear inequality constraints using steady state simulators," *Industrial & Engineering Chemistry Research*, vol. 57, no. 30, pp. 9952–9963, 2018.
- [32] C. Theodoropoulos, "Optimisation and linear control of large scale nonlinear systems: a review and a suite of model reduction-based techniques," in *Coping with Complexity: Model Reduction and Data Analysis*. Springer, 2011, pp. 37–61.
- [33] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [34] L. D. Chambers, *Practical handbook of genetic algorithms: complex coding systems*. CRC press, 2019, vol. 3.
- [35] C. A. Floudas, I. G. Akrotirianakis, S. Caratzoulas, C. A. Meyer, and J. Kallrath, "Global optimization in the 21st century: Advances and challenges," *Computers & Chemical Engineering*, vol. 29, no. 6, pp. 1185–1202, 2005.
- [36] R. Misener and C. A. Floudas, "Antigone: algorithms for continuous/integer global optimization of nonlinear equations," *Journal of Global Optimization*, vol. 59, no. 2-3, pp. 503–526, 2014.
- [37] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical Programming*, vol. 103, no. 2, pp. 225–249, 2005.
- [38] D. Rehfeldt and T. Koch, "Scip-jack—a solver for stp and variants with parallelization extensions: An update," in *Operations Research Proceedings 2017*. Springer, 2018, pp. 191–196.
- [39] C. A. Henao and C. T. Maravelias, "Surrogate-based superstructure optimization framework," *AIChE Journal*, vol. 57, no. 5, pp. 1216–1232, 2011.
- [40] J. D. Smith, A. A. Neto, S. Cremaschi, and D. W. Crunkleton, "Cfd-based optimization of a flooded bed algae bioreactor," *Industrial & Engineering Chemistry Research*, vol. 52, no. 22, pp. 7181–7188, 2012.
- [41] A. M. Schweidtmann and A. Mitsos, "Deterministic global optimization with artificial neural networks embedded," *Journal of Optimization Theory and Applications*, vol. 180, no. 3, pp. 925–948, 2019.
- [42] D. T. Doncevic, A. M. Schweidtmann, Y. Vaupel, P. Schäfer, A. Caspari, and A. Mitsos, "Dcplexeterministic global nonlinear model predictive control with neural networks embedded," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 5273–5278, 2020.
- [43] M. Tao, P. Petsagkourakis, J. Li, and C. Theodoropoulos, "Model reduction based global optimization for large-scale steady state nonlinear systems," 2021.
- [44] I. ILOG, "Cplex 12.0 users manual," *ILOG Inc., Mountain View, CA*, 2006.
- [45] H. Degachi, W. Chagra, and M. Ksouri, "Global optimization method for model predictive control based on wiener model," in *2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*. IEEE, 2015, pp. 1–6.
- [46] J. Katz, I. Pappas, S. Avraamidou, and E. N. Pistikopoulos, "Integrating deep learning models and multiparametric programming," *Computers & Chemical Engineering*, vol. 136, p. 106801, 2020.
- [47] X. Wang, V. Mahalec, and F. Qian, "Globally optimal nonlinear model predictive control based on multi-parametric disaggregation," *Journal of Process Control*, vol. 52, pp. 1–13, 2017.
- [48] C. Long, P. Polisetty, and E. Gatzke, "Globally optimal nonlinear model predictive control," *IFAC Proceedings Volumes*, vol. 37, no. 9, pp. 83–88, 2004.
- [49] A. Stanton, D. Wiegand, and G. Stanton, "Probability reliability and statistical methods in engineering design," *eBook*, 2000.
- [50] D. Xiu and G. E. Karniadakis, "The wiener–askey polynomial chaos for stochastic differential equations," *SIAM journal on scientific computing*, vol. 24, no. 2, pp. 619–644, 2002.
- [51] J. A. Paulson, A. Mesbah, S. Streif, R. Findeisen, and R. D. Braatz, "Fast stochastic model predictive control of high-dimensional systems," in *53rd IEEE Conference on decision and Control*. IEEE, 2014, pp. 2802–2809.
- [52] Y. Liang, H. Lee, S. Lim, W. Lin, K. Lee, and C. Wu, "Proper orthogonal decomposition and its applications—part i: Theory," *Journal of Sound and vibration*, vol. 252, no. 3, pp. 527–544, 2002.
- [53] B. Grimstad and H. Andersson, "Relu networks as surrogate models in mixed-integer linear programs," *Computers & Chemical Engineering*, vol. 131, p. 106580, 2019.
- [54] P. Belotti, L. Liberti, A. Lodi, G. Nannicini, and A. Tramontani, "Disjunctive inequalities: applications and extensions," *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

# Chapter 5

## Uncertainty quantification for distributed parameter systems: deterministic and stochastic cases

### 5.1 Introduction

A robust model predictive control framework for large-scale distributed parameter systems under parametric uncertainty has been presented in Chapter 4. Two practical case studies were employed to validate the computational advantages of the proposed framework. The proposed framework employed polynomial chaos expansion to account for uncertainty quantification. Polynomial chaos expansion (PCE) methods can efficiently deal with many parametric uncertainties with relatively cheap computational resources. However, the PCE method requires prior knowledge about the type of uncertainty distribution to choose the suitable polynomials. Wrong choice of polynomials may lead to loss of computational accuracy. Furthermore, the PCE method would not be so accurate for highly complex systems due to its inherent low order characteristics. This chapter therefore, focuses on developing a novel uncertainty propagation method to deal with complex distributed parameter systems.

Although the traditional Monte Carlo method could accurately account for uncertainty quantification without prior knowledge of uncertainty distribution, the computational cost is high for large-scale distributed parameter systems. Derived from this issue, this work aims to develop a novel uncertainty propagation approach, where “equation-free” techniques are

employed to speed up the propagation process of the Monte Carlo related uncertainty quantification algorithms.

The proposed model reduction-based uncertainty quantification approach can efficiently compute the distributional steady states for both deterministic and stochastic cases under general uncertainty distributions. For deterministic large-scale systems, large volumes of uncertain realisations are generated to approximate different kinds of uncertain distributions. The recursive projection method is then employed to accelerate the propagation of large-scale dynamic systems to steady states, which significantly decreases the required computational resources. For stochastic complex systems, including lifting and restriction operations are coupled with equation-free techniques, to compute the coarse-scale steady states using only small sets of data from the stochastic simulators.

The proposed reduced model-based uncertainty analysis algorithms are validated through a typical chemical tubular reactor, the FitzHugh-Nagumo model and a stochastic catalytic surface reaction case for both single parametric and multivariate uncertainties. Deterministic cases show that the proposed algorithms improve the computational speed by almost one order of magnitude compared to the traditional Monte Carlo method, while the computational accuracy is much better than that of power series and polynomial chaos expansion methods. The stochastic case illustrates that the presented method could efficiently compute the coarse-scale distributional uncertainty with a slight decrease in computational time.

The main possible reason why less gains are obtained for stochastic problems than for deterministic ones is the relatively small number of spatially distributed areas and lattice size due to the limitation of our computational resources. Small number (only four) of spatially distributed areas results in less computational accelerations while the "healing time" of lifting errors and process stochastics require relatively large "coarse-scale" time step to capture slow dynamics, leading to more computational burdens on the RPM accelerations and lifting-restriction operations.

## 5.2 Author's contribution

**Min Tao**

Conceptualization-Equal, Methodology-Equal, Validation-Lead, Writing-original draft-Lead

**Constantinos Theodoropoulos**

Conceptualization-Equal, Funding acquisition-Lead, Methodology-Equal, Project administration-Lead, Supervision-Lead, Writing-review and editing-Lead

# UNCERTAINTY QUANTIFICATION FOR DISTRIBUTED PARAMETER SYSTEMS: DETERMINISTIC AND STOCHASTIC CASES\*

MIN TAO AND CONSTANTINOS THEODOROPOULOS<sup>†</sup>

**Abstract.** A novel uncertainty quantification (UQ) methodology is proposed for both deterministic and stochastic distributed parameter systems, combining Monte Carlo (MC) simulations with "equation-free" (EF) techniques. The highly accurate MC method is first utilised to generate numerous sample realizations for parametric uncertainty quantification, and then, the large number of MC simulations are propagated to their corresponding steady states through EF-based acceleration. We demonstrate UQ computations on both deterministic and stochastic cases: an illustrative deterministic tubular reactor with single and multivariate uncertain parameters, the FitzHugh-Nagumo model with multivariate uncertain parameters and a stochastic heterogeneous catalytic surface reaction example with one uncertain parameter, where equation-free lifting and restriction operations are performed for information exchanges between the fine-scale evolution and the coarse-scale observables. The results show that our proposed UQ approach can achieve a good computational performance with less computations than the standard MC methods and higher accuracy than power series and polynomial chaos expansions for deterministic cases, and capture the coarse-scale distributional uncertainty with a slight decrease in computational time for the stochastic case.

**Key words.** uncertainty quantification, distributed parameter systems, Monte Carlo, equation-free, multiscale

**AMS subject classifications.** 65N12, 65N55, 65Y99

**1. Introduction.** Complex distributed parameter systems, which can be described through continuum models in the form of partial differential equations [10] cover a wide range of applications in engineering and science. Arising from the inherent system stochasticity and/or insufficient prior knowledge about the process [17], uncertainty results in mismatch between model results and true process observations. In particular, random parameters in process models including initial and boundary conditions could result in distinct spatiotemporal behaviours [25], which may significantly impact the computation-based decision making for design and operations. Thus, efficient uncertainty quantification techniques need to be developed to robustly address uncertainty propagation in large-scale distributed parameter systems.

Uncertainty quantification (UQ) techniques are typically performed using system models to generate multiple realisations from probability distributions of uncertain parameters [17]. Then the propagated results are used to compute stochastic moments for quantities of interest. Based on the consequences of the law of large numbers [7], standard Monte Carlo (MC) methods can produce samples to accurately approximate random distributions. Moreover, standard MC and its extensions [14, 1] can generate a sequence of random samples, converging to a complex target of probability distributions. However, large number of samples from MC methods will result in intensive computations, especially for large-scale applications.

Computational challenges drive the developments for efficient uncertainty quantification computational tools. One class of methods exploits efficient sampling ideas such as the Latin hypercube sampling method [15] and sparse grids [4], reducing the

---

\*Preprint Submitted to SIAM Journal.

**Funding:** The University of Manchester and China Scholarship Council joint scholarship (file no. 201706250031)

<sup>†</sup>Department of Chemical Engineering and Analytical Science, The University of Manchester, M139PL, UK ([min.tao@postgrad.manchester.ac.uk](mailto:min.tao@postgrad.manchester.ac.uk) AND [k.theodoropoulos@manchester.ac.uk](mailto:k.theodoropoulos@manchester.ac.uk)).

number of sample points while preserving the computational accuracy. To reduce the variance from standard Monte Carlo estimators, the Multilevel Monte Carlo (MLMC) method [6], an efficient sampling method, utilises discretisation with different step sizes, leading to much fewer samples at the fine discretisation level. The quasi-Monte Carlo sampling method [3] employs deterministic quasi-random sequences to obtain MC samples in a systematic manner, which converges to probability distributions much faster than the standard MC method. Although the number of samples needed to describe random parameters can be reduced, leading to faster computation speeds, a large number of realisations are still necessary to support enough computational accuracy.

Another class of methods aim to construct efficient low-order closed-form models to replace the original expensive system models, including popular power series expansion (PSE) [13] and polynomial chaos expansions (PCE) [23] methods. Power series expansion, as a perturbation method, utilises sensitivity information to rebuild the relationship between output observations and uncertain parameters, which only requires a small number of system evaluations for the low-order sensitivity information. The PSE approach can address parametric uncertainty with desirable precision for slightly perturbed systems, where small fluctuations on quantities of interest occur around average values with respect to random parameters. However, computational results highly depend on the size of the perturbation steps, needing extra computations to enhance computational stability. Moreover, costly high-order sensitivities are required to preserve computational accuracy for strongly nonlinear systems. Polynomial chaos expansion, firstly introduced in [21], represents the observable variables with a series of polynomials with uncertain parameters, and then truncates the exact representation into finite order polynomials [22]. Low-order PCE models, with the coefficients of polynomials computed by the collocation method or Galerkin projection, can accurately account for the impacts of typical probabilistic distributions. Extensions and ramifications of PCE approach can deal with more complex uncertainty problems [20]. However, the PCE method requires a prior knowledge of the type of uncertainty distributions, because "suitable" orthogonal polynomials need to be chosen that correspond to the specific uncertainty distribution, which significantly affects computational accuracy [22].

The *equation-free* (EF) approach was proposed to enable black-box time-stepping simulators to efficiently perform a number of computational tasks such as coarse integration, coarse bifurcation analysis, optimisation and control [9, 19, 18, 2] for a wide range of deterministic and stochastic dissipative systems. A key component is to employ a computational algorithm, called the recursive projection method (RPM) [16], to accelerate the convergence of microscopic/macroscopic integrators toward steady state through adaptively computing the low-dimensional subspaces of the (typically) few dominant modes of the systems. Through only input-output information from deterministic or stochastic simulators, numerical tasks for unavailable (black-box) models could be completed with desirable computational speeds.

Previous works have exploited an equation-free uncertainty quantification framework in combination with general polynomial chaos [24, 26]. In this context, evolving time coefficients of PCE could be computed and utilised to perform continuum-level steady state and bifurcation analysis around the unavailable atomistic level simulator through short bursts of direct model simulations. A combination of multi-scale simulations and the PCE method was built for a computational UQ framework to perform continuum-level UQ tasks involving parametric uncertainties of the atomistic-level simulations. The EF-PCE approach is an accelerated multi-scale PCE tech-

nique exploiting only existing input/output microscopic simulations. However, this uncertainty quantification method still requires a prior knowledge of the type of uncertainty distributions due to the adopted PCE formulation, which also decides the upper bounds of the computational accuracy for the EF-PCE approach.

In this work, we present a novel equation-free uncertainty quantification approach based on MC method and its extensions for continuum-level tasks. The key idea is to accelerate the computations towards the respective steady states around the using the dynamic MC integrators through the equation-free technique exploiting the short-term dynamics of appropriately initialised MC simulations. The EF-MC approach can significantly reduce the computational requirements of MC-related UQ methods, but also inherits the main advantages of MC methods, e.g., high accuracy and capability to deal with general uncertainty distributions. The proposed EF uncertainty quantification approach is applicable for both deterministic and stochastic distributed parameter systems. The difference is that additional multi-scale techniques that link microscopic simulations and "coarse-scale" observations need to be combined into the computational framework for the stochastic cases.

Here two deterministic cases are demonstrated including a chemical tubular reactor with multivariate uncertainty (Damkohler number and adiabatic temperature rise) and a FitzHugh-Nagumo model with two uncertain parameters. In addition, a stochastic lattice kinetic Monte Carlo simulator for oxidation of CO on a catalytic surface with a single uncertain operation parameter is also shown as the stochastic case.

The paper is organized as follows: Section 2 outlines the uncertainty quantification problems for distributed parameter systems. Section 3 introduces the standard Monte Carlo algorithm. Then Section 4 and 5 give two equation-free Monte Carlo algorithms for deterministic and stochastic cases with corresponding applications, respectively. Section 6 presents the deterministic and stochastic case studies and finally, conclusions and future works are discussed.

**2. Problem formulation.** In this work, a model reduction-based uncertainty quantification framework is presented on the continuum-level descriptions for both deterministic and stochastic distributed parameter systems. The continuum-level equations with accompanying boundary and initial conditions (the governed PDEs for deterministic cases generally exist while the corresponding "coarse-scale" PDEs for stochastic cases only *conceptually* exist without closed-form equations [9]) could be described as follows:

$$\begin{aligned}
 (2.1) \quad & \frac{\partial \mathbf{y}}{\partial t} = D \left\{ \frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p} \right\} + R(\mathbf{p}, \mathbf{y}) \\
 & A \left\{ \frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p} \right\} \Big|_{\mathbf{x}=\Omega} = h_{bds}(\mathbf{p}, \mathbf{y}) \\
 & \mathbf{y}|_{t=0} = \mathbf{y}_0(\mathbf{p})
 \end{aligned}$$

Here  $t \in \mathbb{R}$  denotes time dimension,  $\mathbf{x} \in \mathbb{R}^{N_x}$  is spatial dimensions  $N_x$  the number of spatial dimensions,  $N_x=1,2$ , or 3.  $D \in \mathbb{R}$  is the spatial differential operator,  $\mathbf{p} \in \mathbb{R}^{N_p}$  are the uncertainty parameters and  $\mathbf{y} \in \mathbb{R}^{N_y}$  a set of state variables,  $R(\mathbf{p}, \mathbf{y}) : \mathbb{R}^{N_p} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}^{N_y}$  are the nonlinear terms.  $h_{bds}(\mathbf{p}, \mathbf{y}) : \mathbb{R}^{N_p} \times \mathbb{R}^{N_y} \rightarrow \mathbb{R}^{N_y}$  are the right hand sides of the boundary conditions,  $A$  is the operator of the boundary condition equations,  $\Omega$  are the boundaries and  $\mathbf{y}_0 \in \mathbb{R}^{N_y}$  are the initial values of the state variables. Considering steady state analysis and assuming that  $\mathbf{y}(t, \mathbf{x}) \rightarrow \mathbf{y}(\mathbf{x})$ ,

and  $\partial \mathbf{y} / \partial t = 0$ , the above equations become:

$$(2.2) \quad \begin{aligned} 0 &= D\left\{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p}\right\} + R(\mathbf{p}, \mathbf{y}) \\ A\left\{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p}\right\} \Big|_{\mathbf{x}=\Omega} &= h_{bds}(\mathbf{p}, \mathbf{y}) \end{aligned}$$

In practical situations, random parameters  $\mathbf{p}$  can significantly affect the values of the state variables. So the objective of the proposed UQ algorithms is to compute the probabilistic solutions of the above PDEs with random parameters. In general, the unavailability of the system equations inside commercial software prohibits the use of direct model-based techniques for deterministic cases. For micro/mesoscale-level simulators only evolution rules exist, while the corresponding coarse-scale PDEs only exist conceptually without explicit equations. Here we consider the cases where only input/output dynamic simulators (macroscopic or microscopic) are utilised to perform UQ tasks, exploiting the dissipativity of the underlying systems to apply model reduction techniques. These techniques take advantage of the separation of times-scales dissipative systems exhibit, to compute (typically) low-dimensional eigendirections of the dominant slow dynamics [11]. The large-scale time integrator or dynamic models resulting from spatial discretisation could be written into inputs-outputs fashion with a time interval  $tt$ :

$$(2.3) \quad \mathbf{Y}^{n+1} = F(\mathbf{Y}^n, \mathbf{p}^i; tt)$$

where  $n$  is the number of iteration time steps, uncertain realisation  $\mathbf{p}^i$   $\mathbf{Y}^n \in \mathbb{R}^m$  are the  $m$  spatially discretised state variables at time step  $n$ ,  $F(\mathbf{Y}^n, \mathbf{p}^i; tt) : \mathbb{R}^m \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^m$ , denotes a system time-stepper over a time interval  $tt$  on  $m$  spatial grid points. In the above computational formulation, inputs are the current values of state variables  $\mathbf{Y}^n$  and uncertain realisation  $\mathbf{p}^i$  while the outputs  $\mathbf{Y}^{n+1}$  are the values of state variables in next time step  $n + 1$ . Steady-state solutions are often computed simply through using the above computational formulation Eq.(2.3) to integrate the dynamic inputs/outputs systems for large times.

**3. Monte Carlo-related UQ algorithm.** Sampling methods, such as Monte Carlo method and its extensions, use adequate samples to represent propagated random outputs. These methods are particularly useful when the explicit equations of complex black-box systems are unavailable and no information about the type of the parametric uncertainty exists. From the point of view of computational accuracy and ability to handle the general uncertainty, the direct Monte Carlo method could efficiently perform UQ tasks for complex large-scale systems.

Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling (or high-quality pseudo-random sampling) and statistical analysis to obtain sample based numerical results. Specifically, Monte Carlo methods are popularly used in numerical integration problems for UQ calculations of complex large-scale systems, which are based on the consequences of the law of large numbers. Here  $\mathbf{y}(t, \mathbf{x}, \mathbf{p})$  are the quantities of interest, for which we wish to compute statistical moments with respect to uncertain parameters  $\mathbf{p}$  in the domain of  $\Omega'$  as follows:

$$(3.1) \quad \begin{aligned} \mathbb{E}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})) &= \int_{\Omega'} \mathbf{y}(t, \mathbf{x}, \mathbf{p}) \pi(\mathbf{p}) d\mathbf{p} \\ \mathbb{V}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})) &= \int_{\Omega'} (\mathbf{y}(t, \mathbf{x}, \mathbf{p}) - \mathbb{E}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})))^2 \pi(\mathbf{p}) d\mathbf{p} \end{aligned}$$

177 where  $\mathbb{E}(\mathbf{y}(t, \mathbf{x}, \mathbf{p}))$  and  $\mathbb{V}(\mathbf{y}(t, \mathbf{x}, \mathbf{p}))$  are the mean value and variance of  $\mathbf{y}(t, \mathbf{x}, \mathbf{p})$   
 178 with respect to the probability density  $\pi$  of  $\mathbf{p}$ , respectively. However, here we deal  
 179 with large-scale black-box systems, implying that explicit equations of quantities of  
 180 interest  $\mathbf{y}(t, \mathbf{x}, \mathbf{p})$  are not available. According to the law of large numbers, enough  
 181 samples generated from the Monte Carlo method to approximate the random domain  
 182  $\Omega'$ , can be employed to calculate the above quantities. Assuming  $\{\mathbf{p}^i\}_{i=1}^M (\mathbf{p}^1, \mathbf{p}^2, \dots$   
 183  $\mathbf{p}^M)$  is a sequence of i.i.d. (independent and identically distributed)  $M$  samples from  
 184 the probability density  $\pi$  of parametric uncertainty distribution, which could be gen-  
 185 erated through efficient pseudo random number (instead of "truly random numbers")  
 186 generators and statistical transformation methods, then

$$\begin{aligned}
 \mathbb{E}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})) &= \lim_{M_1 \rightarrow \infty} \frac{1}{M_1} \sum_{i=1}^{M_1} \mathbf{y}(t, \mathbf{x}, \mathbf{p}^i) \\
 \mathbb{V}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})) &= \lim_{M_1 \rightarrow \infty} \frac{1}{M_1} \sum_{i=1}^{M_1} (\mathbf{y}(t, \mathbf{x}, \mathbf{p}^i) - \mathbb{E}(\mathbf{y}(t, \mathbf{x}, \mathbf{p})))^2
 \end{aligned}
 \tag{3.2}$$

188 Although the above Eq.(3.2) only holds at the limit of  $M_1 \rightarrow \infty$ , we are able to make  
 189 approximation of the left hand side with a finite value of  $M$ . High order statistical  
 190 moments can be computed by a similar manner.

191 The above formulation Eq.(3.2) can be employed to compute statistical moments  
 192 with respect to uncertain parameters  $\mathbf{p}$ . The Monte Carlo-based UQ algorithm could  
 193 be seen as algorithm

**Algorithm 3.1** EF-MC uncertainty quantification algorithm for deterministic case

**Input:** random parameters  $\mathbf{p}$ ,  $N_{m1}$  maximum number of random samples for  $\mathbf{p}$ , initial conditions  $\mathbf{y}_0$ , initial basis  $\mathbf{Z}_0$ , integration time horizon  $t_0$ , time interval  $tt$  for the input/output integrator and tolerance  $\sigma$

**Output:** statistic moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$

```

1: Generate  $N_2$  ( $N_2 \leq N_{m1}$ ) i.i.d. samples for parametric uncertainty through the
   Monte Carlo method
2: Rearrange the sequence of the  $N_2$  MC samples according to the greedy algorithm
   4.1
3:  $i \leftarrow 1$ 
4: while  $i \leq N_2$  do
5:   Compute the initial conditions  $\mathbf{y}_0(\mathbf{p}^i)$  from Eq.(4.7)
6:   Compute the initial basis  $\mathbf{Z}_0(\mathbf{p}^i)$  from Eq.(4.8)
7:   Perform time integration for time horizon  $t_0$ 
8:   Check the steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$  ?
9:   if Yes then
10:     $i \leftarrow i + 1$ , go to the next sample
11:   else
12:     $kk \leftarrow 1$ 
13:    Perform Newton-Picard iterations as in Eq.(4.6)
14:    Check the steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$  ?
15:    if Yes then
16:       $i \leftarrow i + 1$ , go to the next sample
17:    else
18:      if  $kk \geq N_{m2}$  then
19:        Recompute the basis  $\mathbf{Z}$ 
20:         $kk \leftarrow 1$ 
21:      end if
22:       $kk \leftarrow kk + 1$ 
23:      Repeat steps 13-24
24:    end if
25:  end if
26: end while
27: Compute the statistical moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$  from Eq.(3.2)
28: return  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$ 

```

However, the general input-output dynamic integrators need to be running for long time to perform steady state UQ tasks. The computational challenges motivate our developing equation-free Monte Carlo algorithms in the following sections.

**4. Equation-free Monte Carlo Algorithm 1: deterministic case.** The above Monte Carlo-based UQ algorithms generate a large number of MC samples  $(\mathbf{p}^i)_{i \in M}$  and then directly perform steady state UQ tasks under these uncertain realisations  $(\mathbf{p}^i)_{i \in M}$ . This section introduces equation-free techniques to accelerate the computations of Monte Carlo-based UQ algorithms for deterministic distributed parameter systems, where the UQ calculations are performed directly on the deterministic dynamic input/output simulator.

For dissipative systems we can take advantage of the separation of time scales which can be observed as a gap in the eigspectrum of the system as in Fig.1 where the bulk of the eigenvalues (the blue points) are close to the origin, while a small

207 cluster of eigenvalues (the red points) relatively close to the boundary of the unit  
 208 circle. This gap in the eigenspectrum is exploited as a model reduction technique,  
 209 by adaptively capturing the low-dimensional subspace of the few dominant eigen-  
 210 values. This is achieved by the recursive projection method (RPM) [16] which works  
 211 "around" the time integrator Eq.(2.3) to accelerate the computations of steady states.  
 212 RPM is employed to decompose the solution space  $\mathbb{R}^m$  into two sub-spaces: the low-  
 213 dimensional  $\mathbb{P} \in \mathbb{R}^k$ , slow subspace corresponding to the red eigenvalues in Fig.1, and  
 214 its orthogonal complement  $\mathbb{Q} \in \mathbb{R}^{m-k}$ , corresponding to the blue eigenvalues in Fig.1:

$$215 \quad (4.1) \quad \mathbb{R}^m = \mathbb{P} \oplus \mathbb{Q}$$

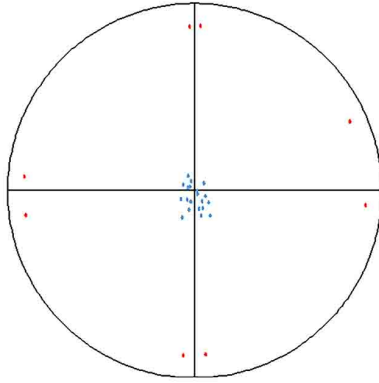


FIG. 1. Eigenspectrum of a discrete-time dissipative system, exhibiting a timescale gap

216 Let  $\mathbf{Z} \in \mathbb{R}^{m \times k}$  be an orthonormal basis for the slow subspace  $\mathbb{P}$ , which can be  
 217 efficiently computed by matrix-free algorithms, such as the Arnoldi method. Then  
 218 the following equalities hold:

$$219 \quad (4.2) \quad \begin{aligned} \mathbf{Z}\mathbf{Z}^T &= P \\ \mathbf{Z}^T\mathbf{Z} &= I_k \in \mathbb{R}^{k \times k} \end{aligned}$$

220 where  $P \in \mathbb{R}^{m \times m}$  is the orthogonal projection operator onto the subspace  $\mathbb{P}$ ,  $m$  is the  
 221 index of the dimensionality of  $\mathbb{P}$ .

$$222 \quad (4.3) \quad Q = I_m - P$$

223 where  $Q \in \mathbb{R}^{m \times m}$  is the high dimensional orthogonal complement of the projector  $P$   
 224 onto the subspace  $\mathbb{Q}$ .

225 The low dimensional Jacobians on the subspace  $\mathbb{P}$  can then be approximated  
 226 by directional perturbations as following. Firstly, the matrix-vector product can be  
 227 computed

$$228 \quad (4.4) \quad F_{\mathbf{Y}}\mathbf{Z} = \frac{(F(\mathbf{Z} + \epsilon\mathbf{Z}) - F(\mathbf{Z} - \epsilon\mathbf{Z}))}{2\epsilon}$$

229 where  $\epsilon$  is a small perturbation on the direction of  $\mathbf{Z}$ . Then the low-order Jacobian  
 230  $H \in \mathbb{R}^{k \times k}$  is given by:

$$231 \quad (4.5) \quad H = \mathbf{Z}^T F_{\mathbf{Y}}\mathbf{Z}$$

232 The idea of RPM is to perform simultaneous iterations: Newton iterations on the  
 233 low-order subspace  $\mathbb{P}$  of the dominant modes and Picard iterations on the orthogo-  
 234 nal complement  $\mathbb{Q}$ . Assuming a numerically stable time integration, the main RPM  
 235 steps include the solution space decomposition (Eq.(4.3)), Newton-Picard iterations  
 236 Eq.(4.6) on the two subspaces and a Picard iteration on the final sum to check con-  
 237 vergence  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p}^i; tt)\| \leq \sigma$ :

$$\begin{aligned}
 & p_1^0 = P\mathbf{Y}^0(\mathbf{p}^i), q_1^0 = Q\mathbf{Y}^0(\mathbf{p}^i) \\
 & p_1^{n+1} = p_1^n + (I_N - \mathbf{Z}\mathbf{H}\mathbf{Z}^T)^{-1} \times (PF(\mathbf{Y}^n, \mathbf{p}^i; tt) - p_1^n) \\
 & q_1^{n+1} = QF(\mathbf{Y}^n, \mathbf{p}^i; tt) \\
 & \mathbf{Y}^{n+1} = p_1^{n+1} + q_1^{n+1}
 \end{aligned}
 \tag{4.6}$$

239 where  $\sigma$  is the tolerance,  $p_1^n$  and  $q_1^n$  denote the projection parts of  $\mathbf{Y}^n$  onto two sub-  
 240 spaces at the  $n$ th iteration.

241 One important step of our equation-free Monte Carlo algorithm is to recycle infor-  
 242 mation between systems under different uncertain realisations, i.e. exploit similarity  
 243 between systems to provide good initialisations. We assume that the more similar-  
 244 ity of systems could be exploited if the (Euclidean) distance between two neighbor  
 245 samples under probability space of uncertain parameters is smaller. Therefore the  
 246 precondition operation is to rearrange the sequence of the generated MC samples  
 247 with the objective being minimizing the distances among neighbor samples under  
 248 probability space. In this work, we adopted a greedy algorithm 4.1 for rearranging  
 249 the sequence of generated MC samples. The key idea is to find the sample with the  
 250 minimum distance to the current sample as the next sample in the new sequence of  
 251 sample sets.

---

**Algorithm 4.1** Greedy algorithm for rearranging the sample sequence

---

**Input:** i.i.d. samples  $\{\mathbf{p}^i\}_{i=1}^M$  with the initial sequence of  $M$  samples from the Monte  
 Carlo method

**Output:** i.i.d. samples  $\{\mathbf{p}^i\}_{i=1}^M$  with a rearranged sequence of  $M$  samples

- 1: Find the sample with the minimum coordinate value(s) as the first sample  $\mathbf{p}^1$  in  
 the rearranged sequence of samples  $\{\mathbf{p}^i\}_{i=1}^M$
  - 2:  $i \leftarrow 1$
  - 3:  $M^0 \leftarrow \emptyset$
  - 4: **while**  $i \leq M$  **do**
  - 5:   Put the current sample  $\mathbf{p}^i$  into the set  $M^0$
  - 6:   Compute the Euclidean distances between the current samples with all samples  
 $\{\mathbf{p}^i\}_{i=1}^M$  except ones in the set  $M^0$
  - 7:   Find the sample with the minimum distance to the current sample as the next  
 sample in the new sequence of sample set  $M'$
  - 8:    $i = i + 1$
  - 9: **end while**
  - 10: **return** i.i.d. samples  $\{\mathbf{p}^i\}_{i=1}^M$  with a rearranged sequence of  $M$  samples
- 

252 It should be noted that the above greedy algorithm 4.1 could generate a new se-  
 253 quence of sample  $\{\mathbf{p}^i\}_{i=1}^M$  with relatively small distance between two neighbor samples  
 254 for relatively small probability space (such as one or two dimensions in this work).  
 255 However, the greedy algorithm could hardly ensure the small distances for complex

256 high-dimensional probability space cases. More efficient rearrangement operations  
 257 should be combined for more complex uncertainty distributions, e.g. cluster methods  
 258 such as K-mean approach could be adopted to first classify the generated MC sam-  
 259 ples into multiple sample clusters and then perform the above algorithm 4.1 for each  
 260 cluster. With the rearranged sequence of samples, similarity between systems under  
 261 neighbor uncertain realisations could be then exploited. Here first-order continuation  
 262 was adopted among adjacent samples. The initial guess  $\mathbf{y}_0(\mathbf{p}^i)$  can be computed as  
 263 follows:

$$(4.7) \quad \mathbf{y}_0(\mathbf{p}^i) = \begin{cases} \mathbf{y}_{ss}(\mathbf{p}^{i-1}) + \frac{\|\mathbf{p}^i - \mathbf{p}^{i-1}\|}{\|\mathbf{p}^{i-1} - \mathbf{p}^{i-2}\|} (\mathbf{y}_{ss}(\mathbf{p}^{i-1}) - \mathbf{y}_{ss}(\mathbf{p}^{i-2})), & i = 3, 4, \dots, M \\ \mathbf{y}_{ss}(\mathbf{p}^{i-1}), & i = 2 \\ \mathbf{y}_0, & i = 1 \end{cases}$$

265 where  $\mathbf{y}_0(\mathbf{p}^i)$  denotes the initial value for the system with uncertain realisation  $\mathbf{p}^i$  ( $i =$   
 266  $1, 2, \dots, M$ ),  $\mathbf{y}_{ss}(\mathbf{p}^i)$  is the corresponding steady-state value,  $\mathbf{y}_0$  is the given initial value.  
 267 The initial values of basis  $\mathbf{Z}_0(\mathbf{p}^i)$  can be calculated though a similar manner of com-  
 268 puting  $\mathbf{y}_0(\mathbf{p}^i)$  as follows.

$$(4.8) \quad \mathbf{Z}_0(\mathbf{p}^i) = \begin{cases} \mathbf{Z}_0(\mathbf{p}^{i-1}) + \frac{\|\mathbf{p}^i - \mathbf{p}^{i-1}\|}{\|\mathbf{p}^{i-1} - \mathbf{p}^{i-2}\|} (\mathbf{Z}_0(\mathbf{p}^{i-1}) - \mathbf{Z}_0(\mathbf{p}^{i-2})), & i = 3, 4, \dots, M \\ \mathbf{Z}_0(\mathbf{p}^{i-1}), & i = 2 \\ \mathbf{Z}_0, & i = 1 \end{cases}$$

270 The above has introduced how to employ the RPM procedure and good initialisa-  
 271 tions to accelerate the computations of previous Monte Carlo-related UQ algorithms.  
 272 The whole equation-free Monte Carlo algorithm in the deterministic steady state UQ  
 273 context could be summarised below as Algorithm 4.2.

**Algorithm 4.2** EF-MC uncertainty quantification algorithm for deterministic case

**Input:** random parameters  $\mathbf{p}$ ,  $N_{m1}$  maximum number of random samples for  $\mathbf{p}$ ,  $N_{m2}$  maximum number of Newton-Picard iterations, initial conditions  $\mathbf{y}_0$ , initial basis  $\mathbf{Z}_0$ , integration time horizon  $t_0$ , time interval  $tt$  for the input/output integrator and tolerance  $\sigma$

**Output:** statistic moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$

```

1: Generate  $N_2$  ( $N_2 \leq N_{m1}$ ) i.i.d. samples for parametric uncertainty through the
   Monte Carlo method
2: Rearrange the sequence of the  $N_2$  MC samples according to the greedy algorithm
   4.1
3:  $i \leftarrow 1$ 
4: while  $i \leq N_2$  do
5:   Compute the initial conditions  $\mathbf{y}_0(\mathbf{p}^i)$  from Eq.(4.7)
6:   Compute the initial basis  $\mathbf{Z}_0(\mathbf{p}^i)$  from Eq.(4.8)
7:   Perform time integration for time horizon  $t_0$ 
8:   Check the steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$  ?
9:   if Yes then
10:     $i \leftarrow i + 1$ , go to the next sample
11:   else
12:     $kk \leftarrow 1$ 
13:    Perform Newton-Picard iterations as in Eq.(4.6)
14:    Check the steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$  ?
15:    if Yes then
16:       $i \leftarrow i + 1$ , go to the next sample
17:    else
18:      if  $kk \geq N_{m2}$  then
19:        Recompute the basis  $\mathbf{Z}$ 
20:         $kk \leftarrow 1$ 
21:      end if
22:       $kk \leftarrow kk + 1$ 
23:      Repeat steps 13-24
24:    end if
25:  end if
26: end while
27: Compute the statistical moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$  from Eq.(3.2)
28: return  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$ 

```

The equation-free Monte Carlo (EF-MC) algorithm 4.1 could significantly accelerate the steady-state UQ computations for large-scale deterministic distributed parameter systems. Some improvements could be used to further enhance the computational efficiency for more complex cases. More advanced rearrangement approach for the sample sequence could be employed to provide good initialisations for the complex uncertainty cases. Meanwhile, the fixed size of slow subspace was employed in this work. Adaptively computing the low-dimensional the dominant subspace under different uncertain realisations may improve the computational speeds.

**5. Equation-free Monte Carlo Algorithm 2: stochastic case.** In this section, we present a novel equation-free Monte Carlo algorithm for stochastic cases, where only a fine-scale stochastic simulator as in Eq.(5.1) is available for coarse-scale

285 UQ tasks as in Eq.(2.1).

$$286 \quad (5.1) \quad \frac{\partial \mathcal{Y}}{\partial t} = f(\mathcal{Y}, \mathbf{p}^i; tt)$$

287 where  $\mathcal{Y}$  denotes spatially distributed microscopic states over the different sub-lattice  
 288 areas  $\delta x$ ,  $f$  is a fine-scale simulator evolving over a fine-scale time interval  $tt$  and  
 289 different sub-lattice areas  $\delta x$  under uncertain realisation  $\mathbf{p}^i$ .

290 Since coarse-scale PDEs (Eq.(2.1)) only *conceptually* exist without closed-form  
 291 equations, equation-free lifting and restriction techniques are required to perform  
 292 information exchange in a multi-scale context as Fig.2 between the fine-scale stochastic  
 293 simulator and coarse-scale simulations for coarse-level UQ computations. Similar  
 294 to deterministic cases, coarse time-steppers in the formulation Eq.(2.3) need to be  
 295 constructed for stochastic cases, which would contain lifting operation, fine-scale sim-  
 296 ulations and restriction operation []. Two operators, lifting ( $\mu$ ) and restriction ( $\mathcal{M}$ ),  
 297 are defined as follows Eq.(5.2,5.3) to build a communication bridge between the dis-  
 298 tributed microscopic states  $\mathcal{Y}$  and coarse values  $\mathbf{Y}$ :

$$299 \quad (5.2) \quad \mu : \mathbf{Y}^n(\mathbf{p}^i) \mapsto \mathcal{Y}^n(\mathbf{p}^i) = \mu[\mathbf{Y}^n(\mathbf{p}^i)]$$

300

$$301 \quad (5.3) \quad \mathcal{M} : \mathcal{Y}^n(\mathbf{p}^i) \mapsto \mathbf{Y}^n(\mathbf{p}^i) = \mathcal{M}[\mathcal{Y}^n(\mathbf{p}^i)]$$

302 where  $n$  is the number of iteration coarse time steps. (Lifting operator is one-to-  
 303 many mapping from a coarse to a fine-scale state, i.e. reinitialization of the fine-  
 304 scale model consistent with a prescribed coarse state while restriction operator is the  
 305 inverse operation of lifting operator, mapping from a fine scale to a coarse state, i.e.  
 306 observation (estimation) of the coarse state from the corresponding fine-scale state.)  
 307 Here a coarse time-stepper would be constructed as the algorithm 5.1 for spatially  
 308 distributed stochastic systems as Fig.2.

---

**Algorithm 5.1** Coarse time-stepper for spatially distributed stochastic systems over  
 a fine-scale time interval  $tt$  and different sub-lattice areas  $\delta x$

---

**Input:** Initial distributed coarse-scale conditions  $\mathbf{Y}^n(\mathbf{p}^i)$ ,  $N_0$  number of repeated  
 stochastic simulations, a fine-scale simulator  $f$  over a fine-scale time interval  $tt$   
 and sub-lattice areas  $\delta x$

**Output:** Next time step coarse-scale conditions  $\mathbf{Y}^{n+1}(\mathbf{p}^i)$

- 1: Lift each of distributed coarse values onto  $N_0$  different fine-scale distributions  
 across each sub-lattice area  $\delta x$  as Eq.(5.2)
  - 2: Perform microscopic evolution from the lifted  $N_0$  fine-scale states through the  
 fine-scale simulator over a fine-scale time interval  $tt$  and the sub-lattice areas  $\delta x$   
 as Eq.(5.1)
  - 3: Restrict  $N_0$  fine-scale states at the next time-step states into distributed coarse-  
 scale values  $\mathbf{Y}^{n+1}(\mathbf{p}^i)$  as Eq.(5.3)
  - 4: **return** Next time step coarse-scale conditions  $\mathbf{Y}^{n+1}(\mathbf{p}^i)$
- 

309 The built inputs/outputs coarse time-stepper d

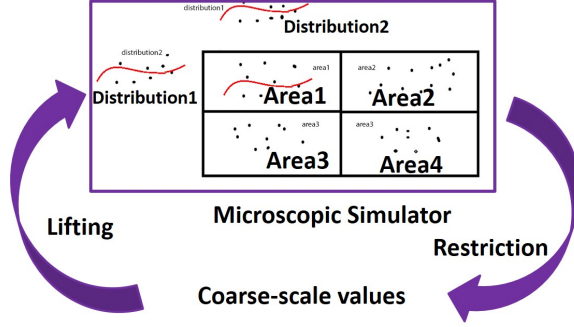


FIG. 2. Schematic of a coarse-scale lifting and restricting operations in a spatially distributed stochastic system

The direct Monte Carlo method for uncertainty quantification tasks requires multi-level computations across microscopic stochastic simulation level, coarse-scale time and space level, and uncertain parameter level, which is extremely expensive. In this work, we try to accelerate the multi-level computations through equation-free strategies, similar to the Algorithm 4.2. The main difference here is that on-the-fly lifting and restriction operations are introduced to perform coarse-scale computations with only stochastic simulators available. The whole computational procedure is illustrated in detail as Algorithm 5.2.

---

**Algorithm 5.2** EF-MC uncertainty quantification algorithm for stochastic cases

---

**Input:** random parameters  $\mathbf{p}$ ,  $N_{m1}$  maximum number of random samples for  $\mathbf{p}$ ,  $N_{m2}$  maximum number of Newton-Picard iterations,  $N_0$  number of repeated stochastic simulations, initial conditions  $\mathbf{y}_0$ , initial basis  $\mathbf{Z}_0$ , integration time horizon  $t_0$ , coarse-scale time interval  $tt$  and tolerance  $\sigma$

**Output:** statistical moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$

- 1: Generate  $N_2$  ( $N_2 \leq N_{m1}$ ) i.i.d. samples through the Monte Carlo method or its extensions to approximate the uncertainty distributions
- 2: Rearrange the sequence of the  $N_2$  MC samples according to the distance to each other
- 3:  $k \leftarrow 1$
- 4: **while**  $k \leq N_2$  **do**
  - 5: Compute the coarse-scale initial conditions  $\mathbf{y}_0^k$  as in Eq.(4.7)
  - 6: Lift the coarse-scale initial conditions  $\mathbf{y}_0^k$  onto  $N_0$  stochastic simulations
  - 7: Compute the coarse-scale initial basis  $\mathbf{Z}_0^k$  in the similar manner
  - 8: Perform  $t_0$  coarse-scale time integration onto  $N_0$  stochastic simulations
  - 9: Restrict  $N_0$  stochastic simulations to compute coarse-scale states
  - 10: Check coarse-scale steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$  ? through lifting-restriction on  $N_0$  stochastic simulations with  $tt$  time interval
  - 11: **if** Yes **then**
    - 12:  $k \leftarrow k + 1$ , go to the next sample
  - 13: **else**
    - 14:  $kk \leftarrow 1$
    - 15: Perform coarse-scale Newton-Picard iterations as Eq.(4.6)
    - 16: Lift the coarse-scale conditions onto  $N_0$  stochastic simulations
    - 17: Perform  $t_0$  coarse-scale time integration onto  $N_0$  stochastic simulations

```

345 18:   Restrict  $N_0$  stochastic simulations to compute coarse-scale states
346 19:   Check coarse-scale steady state convergence condition  $\|\mathbf{Y}^n - F(\mathbf{Y}^n, \mathbf{p})\| \leq \sigma$ 
347   ? through lifting-restriction on  $N_0$  stochastic simulations with  $tt$  time interval
348
349 20:   if Yes then
350 21:      $k \leftarrow k + 1$ , go to the next sample
351 22:   else
352 23:     if  $kk \geq N_{m2}$  then
353 24:       Recompute the basis  $\mathbf{Z}$ 
354 25:        $kk \leftarrow 1$ 
355 26:     end if
356 27:      $kk \leftarrow kk + 1$ 
357 28:     Go to step 13
358 29:   end if
359 30: end if
360 31: end while
361 32: Compute coarse-scale statistical moments  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$  as in Eq.(3.2)
362 33: return  $\mathbb{E}(\mathbf{Y}(\mathbf{p}))$  and  $\mathbb{V}(\mathbf{Y}(\mathbf{p}))$ 

```

363 **6. Case studies.** Practical case studies were employed to verify the above EF-  
364 MC uncertainty quantification algorithms 4.2 and 5.2, with three case studies: A (de-  
365 terministic) chemical tubular reactor, the (deterministic) FitzHugh-Nagumo model,  
366 and the (stochastic) catalytic oxidation of CO on a lattice catalyst surface. The com-  
367 putational results show that the important initialisation step Eq.4.7 could speed up  
368 the UQ computations of our EF-MC algorithms by three to five times. For com-  
369 parison purposes, we perform the same initialisation step in both the standard MC  
370 and our EF-MC algorithms for all the computational cases below. All computational  
371 algorithms were implemented in MATLAB R2019a on a Desktop (Intel Core(TM)  
372 i7-8700 CPU 3.2 GHz, 16 GB memory, 64-bit operating system, Windows 10).

373 **6.1. Deterministic case study 1: chemical tubular reactor.** A chemical  
374 tubular reactor[8] with an exothermic first-order reaction, was utilised as an applica-  
375 tion to illustrate deterministic EF-MC uncertainty quantification algorithm 4.2. The  
376 mathematical formulation of the corresponding reaction-transport processes is given  
377 as follows along with the boundary conditions:

$$\begin{aligned}
\frac{\partial C}{\partial t} &= \frac{1}{Pe_1} \frac{\partial^2 C}{\partial x^2} - \frac{\partial C}{\partial x} + D_a(1 - C) \exp\left(\frac{T}{1 + \frac{T}{\gamma}}\right) \\
\frac{\partial T}{\partial t} &= \frac{1}{Le Pe_2} \frac{\partial^2 T}{\partial x^2} - \frac{1}{Le} \frac{\partial T}{\partial x} - \frac{T}{Le \beta} + \frac{B}{Le} D_a(1 - C) \exp\left(\frac{T}{1 + \frac{T}{\gamma}}\right) + \frac{\beta T_w}{Le} \\
(6.1) \quad \frac{\partial C}{\partial x} - Pe_1 C &= 0 \\
\frac{\partial T}{\partial x} - Pe_2 T &= 0, \quad \text{at } x = 0 \\
\frac{\partial C}{\partial x} &= 0 \\
\frac{\partial T}{\partial x} &= 0, \quad \text{at } x = 1
\end{aligned}$$

379 Here  $C$  and  $T$  are two state variables, the dimensionless concentration and temper-  
380 ature respectively, and  $x$  is the dimensionless longitudinal coordinate. The system's

parameters are the Damkohler number  $D_a$ , the Lewis number  $Le$ , the Peclet number for mass transport  $Pe_1$ , the Peclet number for heat transport  $Pe_2$ , the dimensionless heat transfer coefficient  $\beta$ , the dimensionless adiabatic temperature rise  $B$ , the dimensionless adiabatic wall temperature  $T_w$  and the dimensionless activation energy  $\gamma$ .

The above two PDEs were spatially discretised on 250 computational nodes by central finite differences, resulting in a system of ODEs with 500 unknown variables. An explicit fourth-order Runge–Kutta method was chosen to integrate the dynamic system with the time step being  $1.0e - 06$  seconds. The dynamic simulator was then used as a black-box, input-output time integrator, where inputs were the system parameters and outputs the distributed dimensionless temperatures and concentrations. This work studied how the uncertainty in the system parameters affects the output concentration in this reaction-transport process.  $C_{exit}$  is the dimensionless output concentration,  $\mu(C_{exit})$  is the mean value of the uncertain output concentration and  $\sigma(C_{exit})$  is the standard variance value of the uncertain output concentration. For illustration purposes, a simple case with single parametric uncertainty was firstly studied. Here system parameters were  $Le = 1.0, Pe_1 = Pe_2 = 5.0, \gamma = 20.0, \beta = 1.50, B = 12.0, T_w = 0.0$ , while the distributional uncertainty parameter was the Damkohler number  $D_a$ , satisfying  $D_a \sim N(0.1, 0.01^2)$ .

The above standard distribution needs to be approximated by a large number of Monte Carlo samples. As illustrated in Eq.(3.2), only an infinite number of discrete samples can completely represent the true distribution. Here a finite number of MC samples was utilised to replace the distribution. Independent experiments were performed to choose a suitable number of MC samples, achieving a good trade-off between computational cost and accuracy. Four tests with 1000, 2000, 3000, 4000 ( $N_{m1}$  maximum number of random samples) samples respectively were conducted. The errors for the mean values and variances are firstly decreasing and then becoming stable as the number of samples is increasing. Thus 3000 samples were chosen for the subsequent computations.

Using the generated 3000 samples to represent the distribution of the uncertain parameter  $D_a$ , it is necessary to resort the sequences of samples for subsequent uncertainty propagation. Here only one uncertain parameter with the realisations round the normal value 0.1, one-dimensional rearrangement is enough to provide good initial values for most samples. It should be noted that more efficient rearrangement operations should be adopted for more complex uncertainty distributions. For example, cluster methods such as K-mean approach could be adopted to classify these samples into multiple clusters before resorting samples. Except for rearranging samples and computation methods Eq.(4.7) for initialisation, the equation-free method displayed in Eq.(4.5) was used to accelerate the computation to steady state for each sample. One of the most important components of the method is the computation of the low-dimensional projected Jacobian matrix. Here matrix-free Arnoldi iterations were utilised to adaptively compute the dominant eigendirections of the system. Consequently, only a few short-time dynamic simulations, using the black-box simulator were required. In this work, the dimension  $m$  of projected Jacobian was fixed to be 10 while 60 iterations were set to be the limit number  $N_{m2}$  for checking whether Newton-Picard iterations were convergent or not. Still the computation for the low-dimensional Jacobian is the most expensive part of the whole procedure. Two ways to overcome this were used. One was to run the short-time ( $t_0$  was 0.05 seconds) dynamic solver before the Newton-Picard iterations, avoiding the costly Jacobian computations and leading to fast computations for the fast dynamic cases. The time horizon  $tt$  was

0.01 seconds. The other one was to adaptively compute the low-dimensional basis. The low-dimensional basis was updated only when the previous basis could not make the Newton-Picard iterations convergent in the desirable number of iterations, resulting in much less computations. Computational practice in this work shows that the previous basis is almost always accurate enough for later computations because adjacent samples are close after rearrangement operations. Thus the presented adaptive method may be more suitable for uncertainty quantification. The tolerance  $\sigma$  was set to be  $1.0e - 06$ .

To verify the computational efficiency of the proposed EF-MC uncertainty quantification method, the standard MC method, second-order PSE, second-order PCE and our novel algorithm 4.2 was used for the single parametric uncertainty case. For the standard MC method, advanced precondition techniques, including the same sample rearrangement and first-order continuation were employed. Independent experiments need to be performed to choose suitable parameters for the low-order PSE and PCE methods. The perturbation step size can significantly affect the computational results of the PSE method. Thus the independent experiments with perturbation step size being 0.001, 0.01, 0.03, respectively were constructed, leading to the chosen perturbation step size to be 0.01. Similarly, 9 points of the original system were used for the PCE method after independent experiments with the number of points being 6, 9, 12, respectively.

Tab.1 gives the comparison of the computational results and costs among the standard MC, PSE, PCE and EF-MC methodologies. As displayed in Tab.1, the standard MC method utilising the original dynamic solver is the most time-consuming among the four uncertainty propagation methods. Compared with the standard MC method, equation-free strategies significantly accelerated the computations with more than 87% saving and achieved almost identical results as displayed in Tab.1, with less than 0.001% error. Although PSE and PCE based methods seem to take much less computational times, as shown in Tab.1, choosing the proper perturbation size and number of points for system simulations increased the actual total costs. In fact, the total cost including independent experiments was 0.24 hours for the PSE method while it was 0.40 hours for the PCE-based method. Compared with PSE, PCE produced more accurate mean values, lower and upper bounds. However, both PSE and PCE methods generated much smaller variances as can be seen in Tab.1. Generally, the EF-MC uncertainty quantification algorithm achieved much less computational cost than the standard MC method but higher degree of computational accuracy than the low-order PSE and PCE methods.

TABLE 1  
Uncertainty analysis for a single parameter (95% confidence level) for a number of uncertainty propagation methods.

Models	Lower bound ( $C_{exit}$ )	Upper bound ( $C_{exit}$ )	$\mu(C_{exit})$	$\sigma(C_{exit})$	Computational time (hours)
MC	0.1204	0.3136	0.1963	0.0999	5.32
EF-MC (Relative)	0.1204 Error: $\leq 0.0009\%$	0.3136 Error: $\leq 0.0004\%$	0.1963 Error: $\leq 0.0006\%$	0.0999 Error: $\leq 0.001\%$	0.67
PSE (Relative)	0.1266 Error: 5.15%	0.2856 Error: 8.93%	0.1869 Error: 4.79%	0.0418 Error: -58.16%	0.07
PCE (Relative)	0.1215 Error: -0.91%	0.3018 Error: -3.61%	0.1876 Error: -4.43%	0.0478 Error: -52.15%	0.15

Then uncertainty analysis for multivariate cases was performed to further investigate the efficiency of the EF-MC procedure. Here the system parameters were  $Le = 1.0, Pe1 = Pe2 = 5.0, \gamma = 20.0, \beta = 1.50, T_w = 0.0$ , while the two uncertain parameters were the Damkohler number  $Da$  and the adiabatic temperature rise  $B$ . Firstly, independent parameters were studied, satisfying the distributions:  $Da \sim N(0.08, 0.008^2)$  and  $B \sim N(8, 0.8^2)$ . Then the dependent cases were investigated with the different covariances being  $\begin{pmatrix} 0.008^2 & 0.002 \\ 0.002 & 0.8^2 \end{pmatrix}, \begin{pmatrix} 0.008^2 & 0.005 \\ 0.005 & 0.8^2 \end{pmatrix}, \begin{pmatrix} 0.008^2 & 0.006 \\ 0.006 & 0.8^2 \end{pmatrix}$ . After independent experiments, the number of samples was set to be 4000. The standard Monte Carlo method was utilised to generate 4000 random samples and then the samples were rearranged. MC and EF-MC methods were used to propagate the parametric uncertainty. Tab.2 shows the comparison of computational results between EF-MC procedure and the standard MC method. For all the computational cases, EF-MC method could reduce more than 90% computational time compared with the standard MC method but still kept almost identical results with the standard MC method with less than 0.00105 % error. Moreover, the mean value and standard variance increased slightly with stronger correlation between the uncertain parameters.

The computational results between the accelerated procedure and the direct MC simulation are almost the same, illustrating the high accuracy of the accelerated EF-MC procedure. However, the computational time with accelerated procedure can be reduced by more than 90% for all the simulation cases, verifying its efficiency. Moreover, the correlation between the random  $Da$  and  $B$  slightly affects the mean value and standard variance of the final output concentration, while the correlation of random parameters changes the PDF a lot, especially when the correlation is strong.

**6.2. Deterministic case study 2: FitzHugh-Nagumo Model.** This section would apply the EF-MC uncertainty analysis method to the two-variable reaction-diffusion FitzHugh-Nagumo model [19] shown below:

$$\begin{aligned}
 \frac{\partial x_1}{\partial t} &= \frac{\partial^2 x_1}{\partial x^2} + x_1 - x_2 - x_1^3 \\
 \frac{\partial x_2}{\partial t} &= D \frac{\partial^2 x_2}{\partial x^2} + \epsilon(x_1 - t_1 x_2 - t_0) \\
 \frac{\partial x_1}{\partial x} &= \frac{\partial x_2}{\partial x} = 0, \quad \text{at } x = 0 \\
 \frac{\partial x_1}{\partial x} &= \frac{\partial x_2}{\partial x} = 0, \quad \text{at } x = L
 \end{aligned}
 \tag{6.2}$$

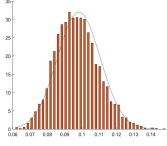
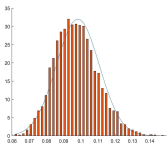
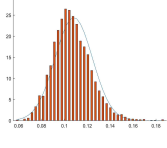
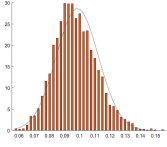
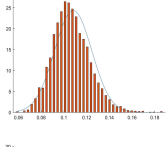
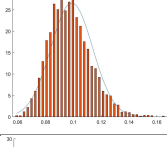
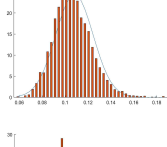
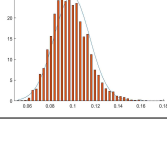
where  $x_1, x_2$  are the two state variables,

$D, t_1, t_0$  the process parameters, and  $L$  the length of the spatial domain  $x$ . Here  $D = 4.0, L = 20, t_0 = -0.03$ , and  $\epsilon$  and  $t_1$  are two uncertain parameters satisfying  $\epsilon \sim N(0.4, 0.04^2)$  and  $t_1 \sim N(2, 0.1^2)$

The two PDE equations were discretised by spatial 200 nodes and then solved by the MATLAB pdepe tool. The reporting time horizon  $tt$  was set to 0.5 seconds. The MATLAB simulator was used as a black-box dynamic simulator with inputs being the process parameters and the time horizon, and outputs being the values of the two variables along the domain.

TABLE 2

Computational results of multivariate uncertainty (95% confidence level) for standard MC and EF-MC method.

Covariances	Methods	$\mu(C_{exit})$	$\sigma(C_{exit})$	Probability density functions	Computational time (hours)
$\begin{pmatrix} 0.008^2 & 0.00 \\ 0.00 & 0.8^2 \end{pmatrix}$	MC	0.0982	0.0124		34.96
	EF-MC	0.0982	0.0124		1.31
$\begin{pmatrix} 0.008^2 & 0.002 \\ 0.002 & 0.8^2 \end{pmatrix}$	MC	0.0985	0.0139		34.24
	EF-MC	0.0985	0.0139		1.33
$\begin{pmatrix} 0.008^2 & 0.005 \\ 0.005 & 0.8^2 \end{pmatrix}$	MC	0.0988	0.0149		34.10
	EF-MC	0.0988	0.0149		1.36
$\begin{pmatrix} 0.008^2 & 0.006 \\ 0.006 & 0.8^2 \end{pmatrix}$	MC	0.0992	0.0156		34.01
	EF-MC	0.0992	0.0156		1.36

Similar to the previous tubular reactor case, 4000 random samples were generated for the two uncertain parameters to approximate the uncertainty distributions. Other algorithm parameters were set to be the same as in the tubular reactor case. Then

both standard Monte Carlo and EF-MC propagation methods were used for steady-state uncertainty propagation with tolerance  $\sigma$  set to be  $1.0e - 06$ . Tab.3 and Fig.3 give the comparison of computational results from the two methods. At nominal parameter values  $\epsilon=0.4$  and  $t_1=2$ , the outputs at  $x = L$  stayed at their nominal values  $\hat{x}_1=-0.4471$  and  $\hat{x}_2=-0.2089$ . The results were really close to the mean values from the computational cases with parametric uncertainty but still could not reflect the uncertain fluctuations of outputs when parametric uncertainty was present, indicating the high demand for uncertainty quantification. In terms of computational efficiency, the proposed EF-MC method took only 2.14 computational hours, more than 90% time saving compared with the standard MC method. Moreover, the almost same mean value, covariance, lower and upper bounds of the state variables  $x_1, x_2$  as seen in Tab.3 and the similar distributions from the two methods depicted in Fig.3 implies the high accuracy of the proposed EF-MC uncertainty propagation method. The possible explanation could be that the final Picard iterations on the full computational space attract the lower-order approximation results onto the accuracy steady state trajectory although low-order approximations would produce errors. Meanwhile, the standard variance of state variable  $\hat{x}_2$  is much smaller than the variance of  $\hat{x}_1$  while fluctuation of  $\hat{x}_2$  in the distribution figure is smaller than that of  $\hat{x}_1$ , indicating that  $\hat{x}_2$  is less sensitive than  $\hat{x}_1$  to parametric uncertainty. In terms of computational cost and accuracy, the proposed EF-MC method is more powerful over the standard MC method.

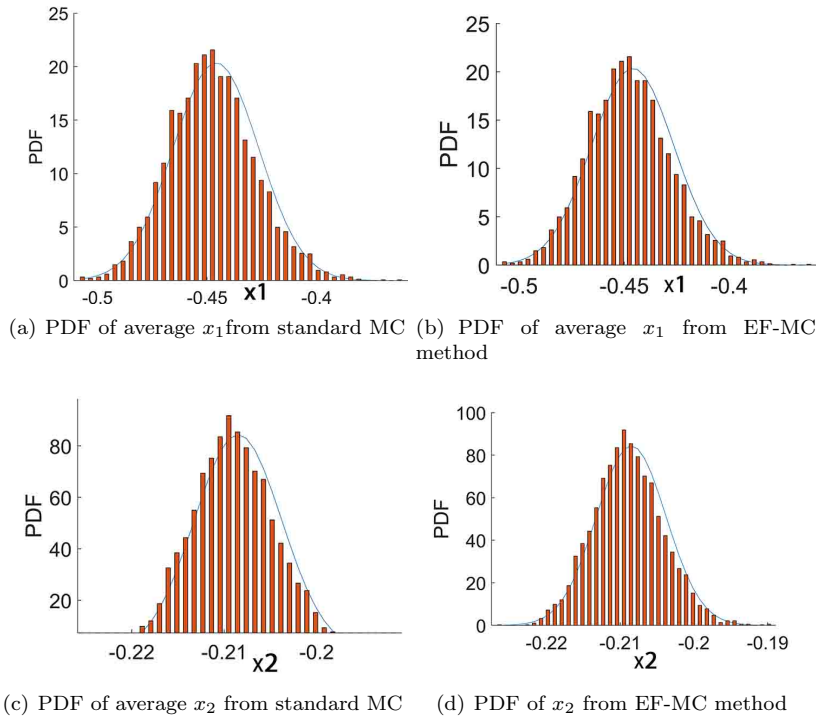
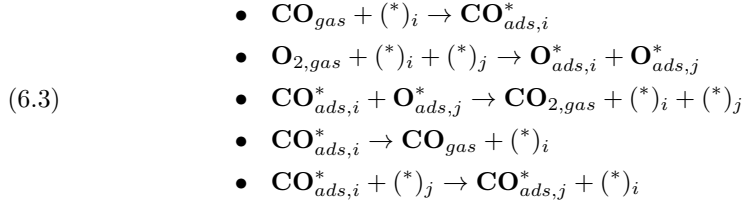


FIG. 3. Probability density functions of the average steady-state values of  $x_1$  and  $x_2$  at the end

TABLE 3  
*Uncertainty analysis for Fitzhugh Naguma model (95% confidence level)*

Methods	Lower bound	Upper bound	$\mu$	$\sigma$	Computational time (hours)
Standard MC $\hat{x}_1$	-0.5069	0.3585	-0.4460	0.0196	27.13
EF-MC $\hat{x}_1$	-0.5069	0.3585	-0.4460	0.0196	2.14
Standard MC $\hat{x}_2$	-0.2264	-0.1889	-0.2085	0.0047	27.13
EF-MC $\hat{x}_2$	-0.2264	-0.1889	-0.2085	0.0047	2.14

**6.3. Stochastic case study: oxidation of CO on a lattice catalytic surface.** In many chemical and biochemical processes, fine-scale stochastic phenomena significantly impact system performances, which requires microscopic-level simulations to exploit process characteristics. Moreover, uncertain parameters would always occur in the complex processes, such as operating conditions at the microscopic level. This section applies the proposed EF-MC uncertainty quantification algorithm 5.2 to a distributed lattice kinetic Monte Carlo process [12]: a catalytic surface reaction . The surface elementary reactions are as follows:



Here  $\text{CO}_{gas}$ ,  $\text{O}_{2,gas}$ ,  $\text{CO}_{2,gas}$  are gas molecules in the bulk gas phase around the surface,  $(*)_i$  and  $(*)_j$  are two types of empty absorbed sites, while  $\text{CO}_{ads,i}^*$  and  $\text{O}_{ads,j}^*$  are adsorbed species on the surface sites  $i$  and  $j$ , respectively. The reaction mechanism involves the adsorption of gas molecules  $\text{CO}_{gas}$  and  $\text{O}_{2,gas}$  onto vacant sites  $(*)_i$  and  $(*)_j$ , a reactive event between two adsorbed molecules  $\text{CO}_{ads,i}^*$  and  $\text{O}_{ads,j}^*$ , desorption of the absorbed molecule  $\text{CO}_{ads,i}^*$  into the gas molecule  $\text{CO}_{gas}$ , and diffusion of the absorbed molecule  $\text{CO}_{ads,i}^*$  to a neighboring empty site  $(*)_j$ . The diffusion process is slow and could be negligible. Hence only the first four elementary reactions were considered in this work. The coarse-scale distributed parameter system was focused on the average coverages of absorbed the molecules in the sub-lattice areas as shown in Fig.2 . For the same sub-lattice coverages, the spatial distributions of molecules could be different due to the inherent stochasticity as in Fig.2 .

In this work,  $200 \times 200$  size lattices were employed with only four sub-lattice areas in the whole reaction surface. It should be noted that this is only a test to validate the proposed EF-MC uncertainty analysis method 5.2, which is still applicable for the cases with more sub-lattices. The lattice kinetic Monte Carlo method [5] was utilised to simulate the above reaction mechanism. This is an inherently input/output dynamic simulator where inputs are uncertain parameters while outputs are the eight coarse-scale coverage values of  $\text{CO}^*$  and  $\text{O}^*$  in the 4 sub-lattices.

For test purposes, one uncertain parameter, the partial pressure of gaseous  $\text{CO}_2$ , was considered satisfying the distribution:  $P_{\text{CO}_2} \sim N(600, 80^2)$ . Since the stochastic simulator for coarse-scale tasks is really expensive, 500 ( $N_2$ ) Monte Carlo samples were obtained to approximate the distribution. Furthermore, an obvious characteristic of the above stochastic simulator is the noise within the output responses. Multiple

copies need be used to reduce the output noise. Here the  $N_0$  number of repeated stochastic simulations was set to be 1000. The 1000 stochastic simulations were employed to compute the coarse-scale coverages of  $\mathbf{CO}^*$  and  $\mathbf{O}^*$  through the restriction operation, which provides a precondition for further projection based acceleration

to generate approximate coarse-scale coverages toward steady state.

Then the approximate steady-state coarse-scale coverages would be lifted onto the random distributions, which need to take a "healing time" to change from random distributions into "equilibrated distributions" and then travel on the normal dynamic trajectories within the stochastic simulator.

The coarse-scale time step should be large enough to capture the healing time. Here, the coarse-scale reporting time horizon  $tt$  was chosen to be  $5.0e - 05$ . The integration time horizon  $t_0$  was also set to be  $5.0e - 05$ .  $m = 8$  the full space Newton-Raphason iteration

was more accuracy to capture the whole dynamics.

The tolerance  $\sigma$  was set to be  $1.0e - 03$ . Tab.4 shows a comparison of results from both standard MC and EF-MC methods. The difference was less than 0.3% for the average coverage values for different sub-lattices, which may be due to the process stochasticity. The EF-MC method reduces the computational time by more than 3%, much less than the value in the above deterministic cases. This is mainly because of the large coarse-time step to capture the slow dynamics. If more copies were used, smooth dynamics could possibly be obtained with smaller coarse-time steps, leading to better computational savings, more accurate coarse-scale values but more computational resources.

The resulting output coverage distributions in the sub-lattice areas are given as follows. Fig.4,5 depict the fluctuations around the normal values of the coverage in different sub-lattices for both  $\mathbf{CO}^*$  and  $\mathbf{O}^*$ , implying the significant impact of parametric uncertainty. Meanwhile, the distributions for the adsorbed molecules are different from each other, which indicates the necessity to consider the effect of the spatial distribution. In the future, a larger number of sub-lattices can be used to better investigate coarse spatial distribution.

TABLE 4  
Uncertainty analysis for lattice catalytic surface reactions

Areas	Methods	$\mu(\mathbf{CO}^*)$	$\sigma(\mathbf{CO}^*)$	$\mu(\mathbf{O}^*)$	$\sigma(\mathbf{O}^*)$	Computational time (hours)
Area1	MC	0.4511	0.0924	0.5071	0.0960	184.32
	EF-MC	0.4528	0.0923	0.5052	0.0958	178.50
Area2	MC	0.4551	0.0883	0.5027	0.0914	184.32
	EF-MC	0.4560	0.0895	0.5017	0.0927	178.50
Area3	MC	0.4538	0.0905	0.5041	0.0938	184.32
	EF-MC	0.4539	0.0902	0.5040	0.0935	178.50
Area4	MC	0.4549	0.0895	0.5028	0.0927	184.32
	EF-MC	0.4555	0.0881	0.5022	0.0911	178.50

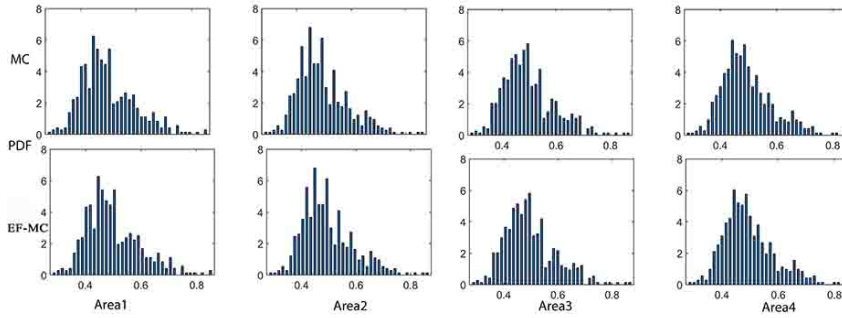


FIG. 4. Probability density function of a coarse-scale steady-state  $CO^*$  coverage value in different sub-lattices

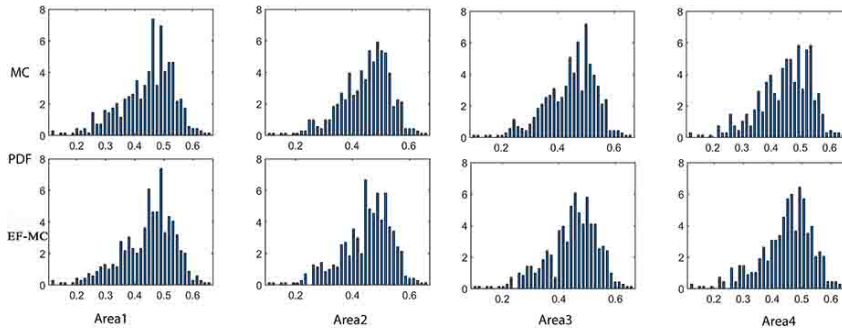


FIG. 5. Probability density function of a coarse-scale steady-state  $O^*$  coverage value in different sub-lattices.

**7. Conclusions.** This paper proposes novel equation-free Monte Carlo uncertainty quantification algorithms for dynamic black-box simulators of large-scale distributed parameter systems. Equation-free techniques can quickly accelerate steady state computations of the black-box dynamic simulators. Deterministic practical examples including a chemical tubular reactor and the FitzHugh-Nagumo model were employed, demonstrating that the proposed EF-MC algorithm 4.2 can reduce the computational costs up to 90% with highly accurate results. While the stochastic case study of the oxidation reactions of CO on a lattice catalytic surface shows that the EF-MC algorithm 5.2 could accurately compute the coarse-scale output distributions with slightly less computations. In such stochastic cases, the "healing-time" during lifting-restriction operations increase computation costs in spite of projection based acceleration. In the future, multivariate cases and higher spatial distribution for the stochastic problems should be employed to investigate the efficiency of our EF-MC algorithms.

**Acknowledgments.** The financial support of the University of Manchester and China Scholarship Council joint scholarship (file no. 201706250031) for MT's PhD studies is gratefully acknowledged

## REFERENCES

- [1] M. BETANCOURT, *A conceptual introduction to hamiltonian monte carlo*, arXiv preprint arXiv:1701.02434, (2017).
- [2] I. BONIS, W. XIE, AND C. THEODOROPOULOS, *Multiple model predictive control of dissipative pde systems*, IEEE Transactions on Control Systems Technology, 22 (2013), pp. 1206–1214.
- [3] R. E. CAFLISCH, *Monte carlo and quasi-monte carlo methods*, Acta numerica, 7 (1998), pp. 1–49.
- [4] D. DŨNG, *Sampling and cubature on sparse grids based on a b-spline quasi-interpolation*, Foundations of Computational Mathematics, 16 (2016), pp. 1193–1240.
- [5] I. S. FRAGKOPOULOS AND C. THEODOROPOULOS, *Multi-scale modelling of electrochemically promoted systems*, Electrochimica Acta, 150 (2014), pp. 232–244.
- [6] M. B. GILES, *Multilevel monte carlo path simulation*, Operations research, 56 (2008), pp. 607–617.
- [7] P.-L. HSU AND H. ROBBINS, *Complete convergence and the law of large numbers*, Proceedings of the National Academy of Sciences of the United States of America, 33 (1947), p. 25.
- [8] K. F. JENSEN AND W. H. RAY, *The bifurcation behavior of tubular reactors*, Chemical Engineering Science, 37 (1982), pp. 199–222.
- [9] I. G. KEVREKIDIS, C. W. GEAR, J. M. HYMAN, P. G. KEVREKIDIS, O. RUNBORG, C. THEODOROPOULOS, ET AL., *Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis*, Communications in Mathematical Sciences, 1 (2003), pp. 715–762.
- [10] A. W. LEUNG, *Systems of nonlinear partial differential equations: applications to biology and engineering*, vol. 49, Springer Science & Business Media, 2013.
- [11] R. LOZANO, B. BROGLIATO, O. EGELAND, AND B. MASCHKE, *Dissipative systems*, in Dissipative Systems Analysis and Control, Springer, 2000, pp. 111–166.
- [12] A. G. MAKEEV, D. MAROUDAS, AND I. G. KEVREKIDIS, *“coarse” stability and bifurcation analysis using stochastic simulators: Kinetic monte carlo examples*, The Journal of chemical physics, 116 (2002), pp. 10083–10091.
- [13] H. POHJANPALO, *System identifiability based on the power series expansion of the solution*, Mathematical biosciences, 41 (1978), pp. 21–33.
- [14] C. P. ROBERT AND G. CASELLA, *The metropolis—hastings algorithm*, in Monte Carlo Statistical Methods, Springer, 1999, pp. 231–283.
- [15] M. D. SHIELDS AND J. ZHANG, *The generalization of latin hypercube sampling*, Reliability Engineering & System Safety, 148 (2016), pp. 96–108.
- [16] G. M. SHROFF AND H. B. KELLER, *Stabilization of unstable procedures: the recursive projection method*, SIAM Journal on numerical analysis, 30 (1993), pp. 1099–1120.
- [17] T. J. SULLIVAN, *Introduction to uncertainty quantification*, vol. 63, Springer, 2015.
- [18] C. THEODOROPOULOS AND E. LUNA-ORTIZ, *A reduced input/output dynamic optimisation method for macroscopic and microscopic systems*, in Model reduction and coarse-graining approaches for multiscale phenomena, Springer, 2006, pp. 535–560.
- [19] C. THEODOROPOULOS, Y.-H. QIAN, AND I. G. KEVREKIDIS, *“coarse” stability and bifurcation analysis using time-steppers: A reaction-diffusion example*, Proceedings of the National Academy of Sciences, 97 (2000), pp. 9840–9843.
- [20] X. WAN AND G. E. KARNIADAKIS, *An adaptive multi-element generalized polynomial chaos method for stochastic differential equations*, Journal of Computational Physics, 209 (2005), pp. 617–642.
- [21] N. WIENER AND A. WINTNER, *The discrete chaos*, American Journal of Mathematics, 65 (1943), pp. 279–298.
- [22] D. XIU, *Numerical methods for stochastic computations*, Princeton university press, 2010.
- [23] D. XIU AND G. E. KARNIADAKIS, *The wiener–askey polynomial chaos for stochastic differential equations*, SIAM journal on scientific computing, 24 (2002), pp. 619–644.
- [24] D. XIU, I. G. KEVREKIDIS, AND R. GHANEM, *An equation-free, multiscale approach to uncertainty quantification*, Computing in science & engineering, 7 (2005), pp. 16–23.
- [25] M. G. ZIMMERMANN, S. O. FIRLE, M. A. NATIELLO, M. HILDEBRAND, M. EISWIRTH, M. BÄR, A. K. BANGIA, AND I. G. KEVREKIDIS, *Pulse bifurcation and transition to spatiotemporal chaos in an excitable reaction-diffusion model*, Physica D: Nonlinear Phenomena, 110 (1997), pp. 92–104.
- [26] Y. ZOU AND I. G. KEVREKIDIS, *Uncertainty quantification for atomistic reaction models: An equation-free stochastic simulation algorithm example*, Multiscale Modeling & Simulation, 6 (2008), pp. 1217–1233.

# Chapter 6

## Robust steady-state optimisation of large-scale distributed systems under uncertainty

### 6.1 Introduction

A model reduction based uncertainty analysis framework for large-scale distributed parameter systems has been presented in Chapter 5. Three practical production cases, including both deterministic and stochastic examples, were employed to validate the computational advantages of the proposed framework. The proposed framework uses "equation-free" techniques to speed up steady-state uncertainty propagation through complex dynamic systems. This chapter focuses on constructing an efficient robust steady state optimisation framework for large-scale distributed parameter systems under parametric uncertainty.

Firstly, the uncertainty propagation techniques from Chapter 5 are employed to account for parametric uncertainty for complex large-scale dynamic systems. Through a few short-term time integrations using the dynamic simulators, the distributed output steady states are easily computed for complex systems. Nevertheless, complex system under parametric uncertainty are still computationally expensive for further optimisation operations, especially for computing derivative information within the popular gradient-based optimisation approaches. Moreover, accurate gradient information may be hard to generate due to the unknown underlying equations and/or parameters of the complex systems with parameter uncertainty.

Complex systems under uncertainty are more likely to present high stochastic characteristics, which may lead to wrong derivatives and hence sub-optimal results. Furthermore, almost all derivative based optimisation methods will tend to be stuck on the local optima, missing the true global solution.

To deal with these issues, a Bayesian global optimisation strategy was adopted in this work. Using prior knowledge, Bayesian optimisation can find good solutions through the trade-off between exploitation and exploration operations. Exploitation operations aim to search better solutions as much as possible while exploration procedures seek the unknown optimisation domains to globally optimise the objective cost functions. Furthermore, here Kriging surrogate models are employed to represent the complex system under uncertainty, which can efficiently predict the system performance and calculate the uncertainties in closed form to speed up process computations.

The proposed robust steady state optimisation framework was tested through a typical chemical tubular reactor under single and multivariate parametric uncertainties. In the previous chapter, the developed model reduction-based uncertainty propagation method has been validated and its ability to accelerate the computation by one magnitude compared with the traditional Monte Carlo method was demonstrated. Here, the test cases on the gradient-based optimisation method show that wrong derivative information can lead to zigzag optimisation paths, which may result in more computational cost and worse solutions. The Bayesian optimisation approach used here, is able to compute better solutions than the derivative-based methods through the same limited number of expensive function evaluations. The computational results also illustrate that Bayesian optimisation procedure can escape from the neighborhood of local optima to perform global search for better optimal solutions.

## **6.2 Author's contribution**

**Min Tao**

Conceptualization-Equal, Methodology-Equal, Validation-Lead, Writing-original draft-Lead

**Constantinos Theodoropoulos**

Conceptualization-Equal, Funding acquisition-Lead, Methodology-Equal, Project administration-Lead, Supervision-Lead, Writing-review and editing-Lead

# Robust steady-state optimisation for large-scale distributed parameter systems with uncertainty

Min Tao<sup>a</sup>, Constantinos Theodoropoulos<sup>a,\*</sup>

<sup>a</sup>*Department of Chemical Engineering and Analytical Science, The University of Manchester, M13 9PL, UK*

---

## Abstract

Complex distributed parameter systems usually exhibit uncertainty, derived from inherent stochastic and/or incomplete knowledge of the underlying processes, leading to substantial model-plant mismatch. In this work, a robust steady-state optimisation framework is proposed for large-scale distributed systems with parametric uncertainty. Our equation-free Monte Carlo uncertainty quantification algorithm [1] is utilised to efficiently account for uncertainty distribution. Nevertheless, large-scale systems including uncertainty are costly often simulated through black-box models, including stochastic noise and are highly non-convex, all of which significantly affects computational speed and solution quality of derivative-based optimisation algorithms. Here a Kriging model-based Bayesian optimisation strategy is adopted to compute a globally optimal solution for large-scale systems including uncertainty with a limited number of system evaluations. The performance of the proposed robust steady-state optimisation framework is demonstrated through practical chemical tubular reactor cases.

*Keywords:* Robust optimisation, Distributed parameter systems, Uncertainty, Equation-free, Bayesian optimisation, Kriging model

---

## 1. Introduction

Large-scale industrial processes are commonly modelled by distributed-parameter systems [2], exhibiting uncertainty due to the inherent stochastics

---

\*Corresponding author

and/or incomplete disclosure of physical phenomena [3, 4]. Uncertainty often causes mismatch between model outputs and true process observations, requiring uncertainty quantification (UQ) to address the relevant challenges. However, UQ methods traditionally require a large number of repeated computationally expensive evaluations especially when large-scale distributed parameter systems (DPS) are involved. Furthermore, optimisation-based intelligent decision-making can significantly improve the performance of industrial production [5, 6, 7, 8]. Nevertheless, uncertainty in large-scale DPS presents complex characteristics, such as high stochasticity and non-convexity, which in addition to the frequent lack of explicit availability of system equations, prevents the implementation of many existing optimisation solvers including gradient based methods and stochastic search approaches. Thus, robust optimisation of DPS under uncertainty is an important practical engineering challenge.

Uncertainty quantification aims to measure the impact of uncertainty on quantities of interest [9]. Explicit analytical solutions may be obtained if system models are available and simple enough. However, process models are mostly black-box for large-scale complex DPS. Even if the system models are explicit, direct analytical solutions cannot be computed for the highly complex models. Fortunately, uncertainty propagation techniques can be employed for practical large-scale problems through simulating black-box system models for a large number of realizations. The large volume of propagated results can be then used to calculate the probability distributions of quantities of interest. For example, numerous samples from standard Monte Carlo methods can accurately represent uncertainty distributions, leading to highly accurate propagated results. However, computing a large number of samples for large-scale systems would be a heavy burden for computing [10]. The computational challenges drive the development of fast-computing UQ methods. One class of such UQ methods is efficient sampling methods including Latin hypercube [11], sparse grid [12] and multilevel Monte Carlo [13, 14] sampling methods. These efficient sampling methods use relatively fewer representative samples to be propagated, which significantly reduces computational times. However, smaller numbers of samples result in loss of computational accuracy. Another class of UQ methods is low-order model based Monte Carlo methods, such as power series and polynomial chaos expansions [10, 15, 16] methods. These low-order model-based methods construct efficient surrogate models to replace the original costly large-scale models, leading to much faster propagation processes [15]. However, loss of compu-

tational accuracy may be caused due to the errors between the approximate models and the original models. Moreover, some of the lower-order methods are limited to certain types of uncertainty distributions. For example, polynomial chaos expansion method requires prior knowledge of the type of uncertainty distributions to choose suitable polynomials for the approximate models [17].

A recent novel "equation-free" Monte Carlo (EF-MC) steady-state uncertainty quantification methodology has been developed for large-scale distributed parameter systems [1]. The EF-MC algorithm is an improved Monte Carlo-based UQ algorithm, which firstly generates a large number of samples by the standard Monte Carlo sampling method or its extensions, and then propagates the samples using the available dynamic models with efficient initialisation procedures. The key step of EF-MC algorithm is the recursive projection method (RPM) [18], exploiting the dissipativity of systems [19] to accelerate propagation computations of dynamic systems to steady states for each sample. Computational results show that the EF-MC uncertainty quantification technique not only inherits the advantages of standard Monte Carlo methods, i.e. high accuracy and more generalisation to different types of uncertainty distributions, but also achieves significant computational savings. Here the EF-MC uncertainty quantification method is utilised to account for parametric uncertainty.

Nevertheless, optimisation of large-scale systems including uncertainty is still challenging. Since large-scale systems with uncertainty are commonly black-box, derivatives as the key elements for gradient-based optimization algorithms, need to be computed by numerical approximations [20]. However, computational noise including truncation and round-off errors can significantly decrease the computational accuracy of derivatives, leading to bad and/or local convergence performance. Moreover, the computation of derivatives, even of low-order ones, is costly due to the multiple evaluations of the systems with uncertainty. Furthermore, traditional gradient-based optimisation techniques can be easily stuck at the neighborhood of local optima, which are possibly much worse than the true globally optimal solutions. Derivative-free stochastic optimisation algorithms seem to be possible solutions to handle the ill-conditioned derivatives and the global optima computation issues. However, these global search algorithms such as simulation annealing [21] and genetic algorithms [22], require large number of function evaluations, which would cause huge computational costs when dealing with expensive systems including uncertainty.

An efficient global optimization method [23], a Bayesian optimisation strategy [24, 25] and its extensions [26, 27] have been developed successfully for expensive black-box systems. This method employs surrogate Kriging models [25, 28] to predict the mean values and the related uncertainty for locally searching the minimums and global exploring the design space. By controlling exploitation and exploration operations, good optimal solutions can be obtained with limited function evaluations. Here, the Bayesian global optimisation strategy combining Kriging surrogate models and expected improvement functions was utilised. The objective of this work is to propose a robust optimisation framework for large-scale distributed systems with uncertainty, combining the EF-MC uncertainty quantification algorithm and the Bayesian global optimisation strategy. It should be noted that this is an extension of our previous findings [29].

The rest of the paper is organized as follows: in Section 2, the detailed robust optimisation framework is introduced. In Section 3, single parameter and multivariate uncertainty cases for a tubular reactor are used to verify this framework. Finally, conclusions and future work are discussed in Section 4.

## 2. Robust steady-state optimisation framework

Here we consider the following large-scale spatially distributed system model with uncertainty:

$$\begin{aligned} \frac{\partial \mathbf{y}}{\partial t} &= D\left\{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p}, \mathbf{d}\right\} + R(\mathbf{y}, \mathbf{p}, \mathbf{d}) \\ A\left\{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}, \frac{\partial^2 \mathbf{y}}{\partial \mathbf{x}^2}, \dots, \frac{\partial^n \mathbf{y}}{\partial \mathbf{x}^n}, \mathbf{p}, \mathbf{d}\right\} \Big|_{\mathbf{x}=\Omega'} &= h_{bds}(\mathbf{y}, \mathbf{p}, \mathbf{d}) \\ \mathbf{y}|_{t=0} &= \mathbf{y}_0(\mathbf{p}) \end{aligned} \quad (1)$$

Eq.(1) gives the governing PDE as well as the corresponding initial and boundary conditions.  $D$  and  $A$  denote dissipative spatial differential operators,  $\mathbf{d} \in \mathbb{R}^{n_d}$  are the design variables,  $\mathbf{p} \in \mathbb{R}^{n_p}$  are the uncertain parameters and  $\mathbf{y} \in \mathbb{R}^{n_y}$  are the state variables,  $R(\mathbf{y}, \mathbf{p}, \mathbf{d}) : \mathbb{R}^{n_y} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{n_y}$  are the nonlinear functions,  $\Omega'$  are the system boundaries,  $h_{bds}(\mathbf{y}, \mathbf{p}, \mathbf{d}) : \mathbb{R}^{n_y} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{n_y}$  are the right hand sides of the boundary conditions.  $\mathbf{y}_0 : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_y}$  are the initial values.  $t \in \mathbb{R}$  and  $\mathbf{x} \in \mathbb{R}^{n_x}$  are the time and space coordinates, respectively.

Typically, the spatiotemporal PDEs are spatially discretised into large-scale ODE models, and then solved through time integration schemes using commercial software [30, 31, 32]. The resulting input-output iterations in time using black-box simulators can be expressed into the following Picard iteration formulation:

$$\mathbf{Y}^{n+1} = F(\mathbf{Y}^n, \mathbf{p}, \mathbf{d}; tt) \quad (2)$$

where  $n \in \mathbb{N}$  is the number of the current iteration step,  $\mathbf{Y}^n \in \mathbb{R}^N$  denotes the spatial variables at time step  $n$ ,  $F(\mathbf{Y}^n, \mathbf{p}, \mathbf{d}) : \mathbb{R}^N \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^N$ , is the time-stepper over a reporting time horizon  $tt$ . Then the steady-state conditions using time integration are reached for  $n \rightarrow \infty$ . Here, a tolerance  $\eta \in \mathbb{R}^+$  is introduced to identify the steady-state condition  $\|\mathbf{Y}^{n+1} - \mathbf{Y}^n\| \leq \eta$  for practical computations.

Then the general robust steady-state optimisation problem for PDE-based distributed parameter systems could be formulated as the following stochastic programming due to uncertain parameters  $\mathbf{p}$ :

$$\begin{aligned} \min_{\mathbf{d}} \quad & G(\mathbf{Y}^{ss}, \mathbf{p}, \mathbf{d}) \\ \text{s.t.} \quad & \\ & \mathbf{Y}^{ss} = f(\mathbf{p}, \mathbf{d}) \\ & g_{cons}(\mathbf{Y}^{ss}, \mathbf{p}, \mathbf{d}) \leq 0 \end{aligned} \quad (3)$$

where  $\mathbf{Y}^{ss} \in \mathbb{R}^N$  denotes the steady-state solutions (also the quantities of interest) satisfying  $\|\mathbf{Y}^{n+1} - \mathbf{Y}^n\| \leq \eta$ ,  $f(\mathbf{p}, \mathbf{d}) : \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^N$  is the unknown map from the uncertain parameters  $\mathbf{p}$  and design variables  $\mathbf{d}$  to steady-state solutions  $\mathbf{Y}^{ss}$ .  $G(\mathbf{Y}^{ss}, \mathbf{p}, \mathbf{d}) : \mathbb{R}^N \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}$  is the objective cost function for the robust optimisation problem, and  $g_{cons}(\mathbf{Y}^{ss}, \mathbf{p}, \mathbf{d}) : \mathbb{R}^N \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{N_{con}}$  are the  $N_{con}$  general constraints for quantities of interest, design variables and uncertain parameters.

In the following parts, a combination of the EF-MC uncertainty quantification technique and the Kriging model based Bayesian optimisation strategy is employed to deal with the above stochastic nonlinear programming Eq.(3).

### 2.1. "Equation-free" Monte Carlo uncertainty propagation method

To address parametric uncertainty, probabilistic approaches describe it by employing a probability density function (PDF) [33]. "Equation-free" Monte Carlo uncertainty propagation method [1], was proposed to efficiently perform steady-state UQ tasks for large-scale distributed parameter systems.

The key idea is to employ the recursive projection method (RPM) [18, 34] to accelerate the computations of the Monte Carlo-related UQ algorithm.

### 2.1.1. Monte Carlo-related UQ algorithm

The MC-based steady-state UQ algorithm employs repeated random sampling (or high-quality pseudo-random sampling) and statistical analysis to compute statistics of quantities of interest through Monte Carlo based numerical integration as below:

$$\begin{aligned}\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})) &= \int_{\Omega'} \mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}) \pi(\mathbf{p}) d\mathbf{p} \\ \mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})) &= \int_{\Omega'} (\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}) - \mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})))^2 \pi(\mathbf{p}) d\mathbf{p}\end{aligned}\tag{4}$$

where  $\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  and  $\mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  denotes the expected value and the variance of quantities of interest  $\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})$  with respect to the probability density  $\pi$  of uncertain parameters  $\mathbf{p}$  in the random domain  $\Omega'$ . Here  $\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})$  is the steady-state solution of the Picard iterations Eq.(2) without explicit solution formulations. Therefore, the sample-based Monte Carlo method can be utilised to compute the above integration in Eq.(4). If  $\{\mathbf{p}^i\}_{i=1}^M$  is a sequence of i.i.d.  $M$  samples from the probability density function of  $\mathbf{p}$ , then the zero moment of quantities of interest can be calculated by Eq.(5). The consequences of the law of large numbers [35] can provide mathematical guarantees on the convergence of the Monte Carlo methods when  $M \rightarrow \infty$ . Here a finite value of  $M$  is adopted to approximately compute the left hand side. Other statistical moments can be computed by a similar sample based manner.

$$\begin{aligned}\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d})) &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{i=1}^M \mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d}) \\ \mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d})) &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{i=1}^M (\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d}) - \mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d})))^2\end{aligned}\tag{5}$$

Since the quantities of interest  $\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d})$  are steady-state solutions computed through Eq.(2), good initialisation could significantly accelerate the computations. Here we adopt a greedy algorithm and the first-order continuation method Eq.(6) to transfer information between systems under different uncertain realisations  $\mathbf{p}^i$  to provide good initialisations for system evaluations.

More details can be seen in [1] .

$$\mathbf{Y}^0(\mathbf{p}^i, \mathbf{d}) = \begin{cases} \mathbf{Y}^{ss}(\mathbf{p}^{i-1}, \mathbf{d}) + \frac{\|\mathbf{p}^i - \mathbf{p}^{i-1}\|}{\|\mathbf{p}^{i-1} - \mathbf{p}^{i-2}\|} (\mathbf{Y}^{ss}(\mathbf{p}^{i-1}, \mathbf{d}) - \mathbf{Y}^{ss}(\mathbf{p}^{i-2}, \mathbf{d})), & i = 3, 4 \dots M \\ \mathbf{Y}^{ss}(\mathbf{p}^{i-1}, \mathbf{d}), & i = 2 \\ \mathbf{Y}^0, & i = 1 \end{cases} \quad (6)$$

Here  $\mathbf{Y}^0(\mathbf{p}^i, \mathbf{d})$  denotes the initial value for the system with uncertain realisation  $\mathbf{p}^i (i = 1, 2 \dots M)$ ,  $\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d})$  is the corresponding steady-state value,  $\|\cdot\|$  denotes the Euclidean distance,  $\mathbf{Y}^0$  is the spatially discretised initial value of  $\mathbf{y}_0$ .

### 2.1.2. The recursive projection method

The above Monte Carlo-based UQ method employs a large number of samples to approximate the probability distributions of uncertain parameters  $\mathbf{p}$ , which ensures the high accuracy and generalised ability to handle different types of uncertainty distributions. However, the computational costs of standard Monte Carlo are extremely intensive if directly propagating these samples for large-scale distributed systems. Fortunately, the recursive projection method [18, 34], a key step of the EF-MC uncertainty propagation method, can be employed to achieve significant computational savings [29].

The recursive projection method focuses on dissipative systems, and exploits the separation of their timescales to fast and slow ones, which can be seen in the eigenspectrum of linearized discrete-time systems. Fig.1 shows the eigenspectrum as such a dissipative system exhibiting separation of scales. On the one hand, the small number of slow modes, corresponding to the eigenvalues close to the boundary of the unit circle depicted in Fig.1 represent the slow dynamics, which are also the dominant dynamics of the system. On the other hand, the large cluster of fast modes, corresponding to the eigenvalues around the origin in Fig.1, represent the fast dynamics of the dissipative systems.

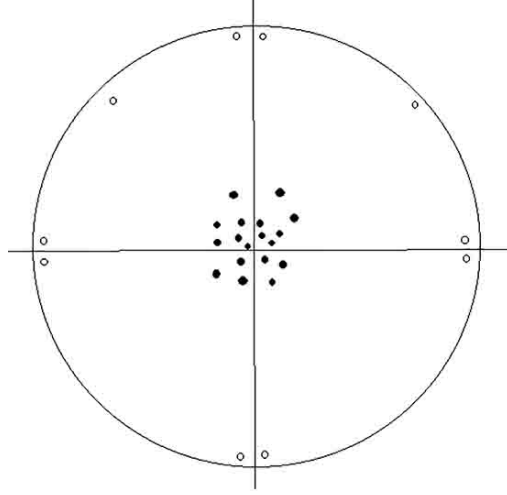


Figure 1: Eigenspectrum of a discrete-time dissipative system, exhibiting a clear separation of modes

Let  $\mathbf{P}$  be the low-dimensional subspace corresponding to the slow modes of the discrete-time dissipative systems, and  $\mathbf{Q}$  be its orthogonal complement, so

$$\mathbf{P} \oplus \mathbf{Q} = \mathbb{R}^m \quad (7)$$

where  $\mathbf{P} \in \mathbb{R}^k$  and  $\mathbf{P} \in \mathbb{R}^{m-k}$ ,  $m$  being the dimension of the subspace  $\mathbf{P}$ . Let  $P$  and  $Q$  be the projectors onto the subspace  $\mathbf{P}$  and  $\mathbf{Q}$  respectively, then

$$\begin{aligned} Q &\in \mathbb{R}^{m \times m} = I_m - P \\ P &= ZZ^T \in \mathbb{R}^{m \times m} \end{aligned} \quad (8)$$

where  $Z \in \mathbb{R}^{m \times k}$  is the orthogonal basis for the subspace  $\mathbf{P}$ , which can be efficiently computed by matrix-free Arnoldi iteration methods.

RPM performs Newton iterations on the low-dimensional subspace  $\mathbf{P}$  and Picard iterations on its complement  $\mathbf{Q}$  until the convergence conditions ( $\|\mathbf{Y} - F(\mathbf{Y}, \mathbf{p}, \mathbf{d})\| \leq \eta$ ) are satisfied as follows:

$$\begin{aligned} p^0 &= PY^0(\mathbf{p}, \mathbf{d}), q^0 = QY^0(\mathbf{p}, \mathbf{d}) \\ p^{n+1} &= p^n + (I - ZHZ^T)^{-1} \times (PF(\mathbf{Y}^n, \mathbf{p}, \mathbf{d}) - p^n) \\ q^{n+1} &= QF(\mathbf{Y}^n, \mathbf{d}) \\ \mathbf{Y}^{n+1} &= p^{n+1} + q^{n+1} \end{aligned} \quad (9)$$

where  $\eta \in \mathbb{R}^+$  is the tolerance,  $p^n$  and  $q^n$  denote the projection parts of  $\mathbf{Y}^n$  onto two subspaces at the  $n$ th iteration. In the above iteration process Eq.(9), the low-order Jacobian  $H \in \mathbb{R}^{k \times k}$  as illustrated in Eq.11 can be computed through the matrix-vector product  $\mathbf{F}_{\mathbf{Y}}Z$  obtained by directional perturbations in Eq.10

$$\mathbf{F}_{\mathbf{Y}}Z = \frac{(\mathbf{F}(\mathbf{Y} + \epsilon Z) - \mathbf{F}(\mathbf{Y} - \epsilon Z))}{2\epsilon} \quad (10)$$

$$H = Z^T \mathbf{F}_{\mathbf{Y}}Z \quad (11)$$

where  $\epsilon \in \mathbb{R}$  a small perturbation on the direction of  $Z$ . The initial values can be calculated though first-order continuation [1] .

Therefore, the whole equation-free Monte Carlo algorithm can be summarised below as Algorithm 1. More details can be referred to the literature [1].

---

**Algorithm 1** EF-MC uncertainty quantification algorithm

---

**Input:** random parameters  $\mathbf{p}$ ,  $N_{m1}$  maximum number of random samples for  $\mathbf{p}$ ,  $N_{m2}$  maximum number of Newton-Picard iterations, initial conditions  $\mathbf{y}_0$ , initial basis  $\mathbf{Z}_0$ , integration time horizon  $t_0$ , the input/output integrator over a time interval  $tt$  and tolerance  $\sigma$

**Output:** statistical moments  $\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  and  $\mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$

- 1: Generate  $M$  ( $M \leq N_{m1}$ )  $\{\mathbf{p}^i\}_{i=1}^M$  i.i.d. samples to compute parametric uncertainty through the Monte Carlo method
- 2:  $i \leftarrow 1$
- 3: Find the sample with the minimum coordinate value(s)
- 4:  $M^0 \leftarrow \emptyset$
- 5: **while**  $i \leq M$  **do**
- 6:   Put the current sample  $\mathbf{p}^i$  into the set  $M^0$
- 7:   Compute the Euclidean distances between the current samples with all samples  $\{\mathbf{p}^i\}_{i=1}^M$  except ones in the set  $M^0$
- 8:   Find the sample with the minimum distance to the current sample as the next sample in the new sequence of sample set  $\{\mathbf{p}^i\}_{i=1}^M$
- 9:    $i = i + 1$
- 10: **end while**
- 11:  $i \leftarrow 1$
- 12: **while**  $i \leq N_2$  **do**
- 13:   Compute the initial conditions  $\mathbf{Y}^0(\mathbf{p}^i, \mathbf{d})$  from Eq.(6)

```

14:   Compute the initial basis  $\mathbf{Z}_0(\mathbf{p}^i, \mathbf{d})$ 
15:   Perform time integration for time horizon  $t_0$ 
16:    $kk \leftarrow 0$ 
17:   Compute  $\mathbf{Y}^{n+1}$  at the next time step through time integration  $\mathbf{Y}^{n+1} =$ 
       $F(\mathbf{Y}^n, \mathbf{p}^i, \mathbf{d}; tt)$ 
18:   while  $\|\mathbf{Y}^{n+1} - \mathbf{Y}^n\| > \eta$  do
19:     if  $kk \geq N_{m2}$  then
20:       Recompute the basis  $\mathbf{Z}(\mathbf{p}^i, \mathbf{d})$ 
21:        $kk \leftarrow 0$ 
22:     end if
23:     Perform Newton-Picard iterations as in Eq.(9)
24:      $kk \leftarrow kk + 1$ 
25:     Compute  $\mathbf{Y}^{n+1}$  at the next time step through  $\mathbf{Y}^{n+1} = F(\mathbf{Y}^n, \mathbf{p}^i, \mathbf{d}; tt)$ 
26:   end while
27:    $\mathbf{Y}^{ss}(\mathbf{p}^i, \mathbf{d}) = \mathbf{Y}^n(\mathbf{p}^i, \mathbf{d})$ 
28:    $i \leftarrow i + 1$ 
29: end while
30: Compute the statistical moments  $\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  and  $\mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  from
      Eq.(5)
31: return  $\mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  and  $\mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$ 

```

---

Then the statistical moments  $E_Y(\mathbf{d}) = \mathbb{E}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  and  $V_Y(\mathbf{d}) = \mathbb{V}(\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d}))$  can be computed through the Algorithm 1 under different design points of  $\mathbf{d}$ . Thus, the above stochastic programming Eq.(3) could be transformed into the following deterministic nonlinear programming (NLP) formulation:

$$\begin{aligned}
& \min_{\mathbf{d}} \quad G'(E_Y, V_Y, \mathbf{d}) \\
& \quad s.t. \\
& \quad E_Y \in \mathbb{R}^N = f_1(\mathbf{d}) \\
& \quad V_Y \in \mathbb{R}^N = f_2(\mathbf{d}) \\
& \quad g'_{cons}(E_Y, V_Y, \mathbf{d}) \leq 0
\end{aligned} \tag{12}$$

where  $G'(E_Y, V_Y, \mathbf{d})$  and  $g'_{cons}(E_Y, V_Y, \mathbf{d})$  are the deterministic formulations of  $G$  and  $g_{cons}$ .

If we eliminate the equality constraints  $E_Y = f_1(\mathbf{d})$  and  $V_Y = f_2(\mathbf{d})$ , then

the above problem Eq.(12) could be transformed into the following:

$$\begin{aligned} \min_{\mathbf{d}} \quad & G'(f_1(\mathbf{d}), f_2(\mathbf{d}), \mathbf{d}) \\ \text{s.t.} \quad & \\ & g'_{cons}(f_1(\mathbf{d}), f_2(\mathbf{d}), \mathbf{d}) \leq 0 \end{aligned} \tag{13}$$

Generally, the above problem Eq.(13) could be reformulated into unconstrained nonlinear programming problems if  $g'_{cons}$  only includes simple bound constraints. However, the relationship functions  $f_1$  and  $f_2$  are unknown (may exploit complex non-linearity) and expensive to evaluate due to costly uncertainty propagation processes, requiring efficient optimisation strategies to deal with the problem Eq.(13). It should be noted that we only use the mean value and variance in this work. High order statistical moments and bounds can be computed through the similar manner Eq.(5) based on the propagated distributions of  $\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})$ .

## 2.2. Bayesian global optimisation

Although the EF-MC uncertainty propagation algorithm speeds up the computations of steady states  $\mathbf{Y}^{ss}(\mathbf{p}, \mathbf{d})$  for large-scale distributed systems with uncertainty, the above optimisation problems 13 are still challenging due to the unavailable relationships expensive  $f_1$  and  $f_2$ . Black-box map functions between the statistics (quantities of interest) and the design variables  $\mathbf{d}$  may exploit the complex nonlinear characteristics e.g. non-convexity and non-smoothness, which can lead to the ill-conditioned derivatives and difficulties to compute global optima for derivative-based optimisation algorithms [36, 37]. Moreover, it is hard to employ the efficient surrogate models such as artificial neural networks (ANNs) because it would require a large number of expensive samples.

Fortunately, Bayesian optimisation strategy can efficiently handle black-box systems since it does not require any function information and can often find a good solution within the limited number of function evaluations. The key components of Bayesian optimisation strategy are a probabilistic surrogate model to predict the expected output value and variance (uncertainty) of the predictions, and an acquisition function including the predictive information of mean value and variance to compute the next sampling point. Moreover, the surrogate model would be updated with additional samples (next sampling points) to give more accurate predictive information. Here

we adopt the popular Kriging model as the surrogate model and the expected improvement function as the acquisition function.

### 2.2.1. Kriging model

Originated from the geological research [38], Kriging models have been applied as common surrogate models across multiple disciplines [39, 40, 41, 42]. In modern data science, Kriging models are also used as Gaussian process regression [43]. Compared with other surrogate models, Kriging models have been found to be really efficient for expensive nonlinear problems [44].

Different from other surrogate models, the outputs of Kriging models are random variables including mean value and variance due to their statistical background. The mean value can be used to estimate the nominal function value while the variance can be utilised to predict the uncertainty or the errors, which enables more advantages of Kriging models over other surrogate models. Here, we want to employ Kriging model to build the maps  $f_1$  and  $f_2$  from  $\mathbf{d}$  to the  $G' \in \mathbb{R}$  with an assumption that  $g'_{cons}$  being simple bounds constraints.

The prior knowledge of Gaussian process model indicates the random output variable  $G'(\cdot): \mathbb{R}^{n_d} \rightarrow \mathbb{R}$  at any location point of design space  $\mathbf{d} \in \mathbb{R}^{n_d}$  with mean value  $\mu$  and variance  $\sigma^2$ , a Gaussian distribution for the finite random variables can be as follows:

$$\mathcal{G}(\mathbf{d}) \sim N(\boldsymbol{\mu}(\mathbf{d}), K(\mathbf{d}, \mathbf{d})) \quad (14)$$

where  $\mathcal{G}(\mathbf{d}) = (G'(\mathbf{d}_1), \dots, G'(\mathbf{d}_{N'})) : \mathbb{R}^{n_d \times N'} \rightarrow \mathbb{R}^{N'}$  denotes the results of the given inputs  $\mathbf{d} = [\mathbf{d}_1, \dots, \mathbf{d}_{N'}]^T \in \mathbb{R}^{n_d \times N'}$ ,  $N'$  is the number of function evaluations,  $\boldsymbol{\mu}(\mathbf{d}) = (\mu(\mathbf{d}_1), \dots, \mu(\mathbf{d}_{N'})) : \mathbb{R}^{n_d \times N'} \rightarrow \mathbb{R}^{N'}$  is the mean values, and  $K(\mathbf{d}, \mathbf{d}) : \mathbb{R}^{n_d \times N'} \rightarrow \mathbb{R}^{N' \times N'}$  is the covariance with  $K_{ij} = k'(\mathbf{d}_i, \mathbf{d}_j)$  as the kernel function.

In this work, the smooth and stationary squared-exponential kernel [43] was used, assuming that the function values  $G'(\mathbf{d}_1)$  and  $G'(\mathbf{d}_2)$  are close when the locations  $\mathbf{d}_1$  and  $\mathbf{d}_2$  of design space are close:

$$k'(\mathbf{d}_i, \mathbf{d}_j) = \sigma_f^2 \exp\left(-\frac{1}{2\mathbf{l}^2}(\mathbf{d}_i - \mathbf{d}_j)^T(\mathbf{d}_i - \mathbf{d}_j)\right) \quad (15)$$

where  $\sigma_f^2$  and  $\mathbf{l}$  are kernel parameters, denoting the covariance magnitude and characteristic length-scale parameters, which could be estimated through inference method such as maximum likelihood estimation (MLE). MLE method

seeks the above parameter values to maximize the probability of the function values  $\rho(\mathcal{G}|_{\sigma, \mathbf{l}, \mathbf{d}})$ , which is equal to the log likelihood of the observed points as following [45, 46]:

$$\log \rho(\mathcal{G}|_{\sigma, \mathbf{l}, \mathbf{d}}) = -\frac{n_d}{2} \log 2\pi - \frac{1}{2} \log |K_{\sigma, \mathbf{l}}(\mathbf{d}, \mathbf{d})| - \frac{1}{2} (\mathcal{G} - \boldsymbol{\mu}(\mathbf{d}))^T (K_{\sigma, \mathbf{l}}(\mathbf{d}, \mathbf{d}))^{-1} (\mathcal{G} - \boldsymbol{\mu}(\mathbf{d})) \quad (16)$$

where  $(\mathbf{d}, \mathcal{G})$  is the observed dataset.

Ignoring constant terms, the log likelihood could be simplified into the equation below:

$$\log \rho'(\mathcal{G}|_{\sigma, \mathbf{l}, \mathbf{d}}) = -\frac{1}{2} \log |K_{\sigma, \mathbf{l}}(\mathbf{d}, \mathbf{d})| - \frac{1}{2} (\mathcal{G} - \boldsymbol{\mu}(\mathbf{d}))^T (K_{\sigma, \mathbf{l}}(\mathbf{d}, \mathbf{d}))^{-1} (\mathcal{G} - \boldsymbol{\mu}(\mathbf{d})) \quad (17)$$

Then the hyperparameters  $\sigma, \mathbf{l}$  could be estimated by maximising Eq.(17), which could be employed to compute the prediction at new samples points.

Meanwhile, the current variables and the unknown variables  $G'(\mathbf{d}_*)$  given new location  $\mathbf{d}_*$  satisfy a joint Gaussian distribution:

$$\begin{bmatrix} \mathcal{G}(\mathbf{d}) \\ G'(\mathbf{d}_*) \end{bmatrix} \sim N \left( \begin{bmatrix} \boldsymbol{\mu}(\mathbf{d}) \\ \boldsymbol{\mu}(\mathbf{d}_*) \end{bmatrix}, \begin{bmatrix} K & k'(\mathbf{d}, \mathbf{d}_*) \\ k'(\mathbf{d}_*, \mathbf{d}) & k'(\mathbf{d}_*, \mathbf{d}_*) \end{bmatrix} \right) \quad (18)$$

Then the posterior predictive  $G'(\mathbf{d}_*)$  would satisfy the Gaussian distributions as below:

$$G'(\mathbf{d}_*)|_{\mathcal{G}(\mathbf{d}), \mathbf{d}, \mathbf{d}_*} \sim N(\mu_*, \sigma_*^2) \quad (19)$$

where  $\mu_* = \boldsymbol{\mu}(\mathbf{d}_*) + k'(\mathbf{d}_*, \mathbf{d}) K^{-1} (\mathcal{G}(\mathbf{d}) - \boldsymbol{\mu}(\mathbf{d}))$ ,  $\sigma_*^2 = k'(\mathbf{d}_*, \mathbf{d}_*) - k'(\mathbf{d}_*, \mathbf{d}) K^{-1} k'(\mathbf{d}, \mathbf{d}_*)$ .

Here, we could choose a zero mean function such that  $\boldsymbol{\mu}(\mathbf{d}) = \mathbf{0}$ , which could be achieved by normalizing the dataset beforehand, leading to more straight computations. Latin hypercube sampling (LHC) method is employed to produce these sample sets  $(\mathbf{d}, \mathcal{G})$  around the feasible search space. The general LHC sampling method is employed to choose the samples according to two rules, spanning the whole design space and maximizing the difference among the samples [47]. Specifically, the sample domain is divided into  $N'$  sub-intervals, where sample points are generated randomly to represent the whole sub-regions.

Then hyperparameters  $\sigma, \mathbf{l}$  could be estimated through the Eq.(17) and the collected dataset  $(\mathbf{d}, \mathcal{G})$ . Here, the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) [48] was used with 10 restarting times to avoid local optima for the first computations. The recursive optimisation procedure of

updating the parameters would employ one starting point, the previous computational result. Thus, surrogate Kriging models could be constructed to predict the function information  $G'$  in any location  $\mathbf{d}$  including the expected mean value and variance (uncertainty), which would be which would be then employed by the acquisition function for calculating the location of the next sample point or find a better solution over the current best observe.

### 2.2.2. Expected improvement function

In Bayesian global optimisation, acquisition functions are the powerful tools to quantify the predictive information of mean value and variance, computing the next sampling point. Acquisition functions are also used as active learning functions in machine learning areas [49, 50]. The utilisation of acquisition functions provides the capability of the Bayesian optimisation procedure to achieve a good balance between local exploitation and global exploration. The local exploitation ability is derived from exploiting the information of mean value  $\mu_*$  to search a minimum function value, while the global exploration capability relies on the information of variance  $\sigma_*^2$  to explore more uncertain feasible space. The popular acquisition functions include probability of improvement and expected improvement (EI) functions, which has been discussed in literature [51]. More improved functions can be seen in these works [52, 53, 54, 55]. Here EI function with Kriging model was used to deal with the trade-off between exploitation and exploration as below [23].

$$I(\mathbf{d}_* |_{\mathbf{d}, G'(\mathbf{d})}) = \max(G'(\mathbf{d}_*) - G'(\mathbf{d}^+) - j, 0) |_{\mathbf{d}, G'(\mathbf{d})} \quad (20)$$

where  $I(\mathbf{d}_* |_{\mathbf{d}, G'(\mathbf{d})})$  is the improvement function. Since  $G'(\mathbf{d})$  is a random variable,  $I(\mathbf{d})$  is also a random variable.  $G'(\mathbf{d}^+)$  is the maximum function value for the history samples  $(\mathbf{d}, G'(\mathbf{d}))$  and  $\mathbf{d}^+$  is the location for the sample with the maximum function value, i.e.  $\mathbf{d}^+ = \operatorname{argmax}_{\mathbf{d} \in \mathbf{d}} G'(\mathbf{d})$ ,  $j$  is a parameter that determines the explore-exploit trade-off during the Bayesian optimisation process. Higher  $j$  values imply less importance of the posterior mean value but more importance of variance, leading to more exploration to reduce the uncertainty. In this work,  $j$  was set to be 0.01. Then the probability of the improvement function could be denoted as below:

$$\varrho(I(\mathbf{d}_* |_{\mathbf{d}, G'(\mathbf{d})})) = \Phi \left( \frac{\mu_* - G'(\mathbf{d}^+) - j}{\sigma_*} \right), \sigma_* > 0 \quad (21)$$

where  $\Phi$  is the cumulative distribution function of the standard normal distribution,  $\mu_*$  and  $\sigma_*$  are the mean value and variance of the posterior prediction  $G'(\mathbf{d}_*)$ , respectively.

And the expectation of the above EI function Eq.(20) is expressed:

$$EI(\mathbf{d}_*|\mathbf{d}, G'(\mathbf{d})) = \mathbb{E}_{\mathbf{d}, G'(\mathbf{d})} \max(G'(\mathbf{d}_*) - G'(\mathbf{d}^+) - j, 0) \quad (22)$$

The close-form equations of the expectation improvement can be expressed as following [23]:

$$EI(\mathbf{d}_*|\mathbf{d}, G'(\mathbf{d})) = \begin{cases} (\mu_* - G'(\mathbf{d}^+) - j)\Phi\left(\frac{\mu_* - G'(\mathbf{d}^+) - j}{\sigma_*}\right) + \sigma_*\phi\left(\frac{\mu_* - G'(\mathbf{d}^+) - j}{\sigma_*}\right) & , \sigma_* > 0 \\ 0 & , \sigma_* = 0 \end{cases} \quad (23)$$

where  $\phi$  are the probability density function of a standard normal distribution. The  $EI$  function includes two terms, the first one representing the relative importance of exploitation while the second one indicating the weight of exploration.

Then the optimisation procedure would be employed to compute the location  $\mathbf{d}_*$  for the next sample pair  $(\mathbf{d}_*, G'(\mathbf{d}_*))$ . Here the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) [48] was used with 20 restarting times to avoid local optima.

So far, the whole procedure of the robust steady-state optimisation framework have been introduced, which could be summarised as the following Algorithm 2.

---

**Algorithm 2** Robust steady-state optimisation of distributed parameter systems under uncertainty

---

**Input:** random parameters  $\mathbf{p}$ ,  $N_{m1}$  maximum number of random samples for  $\mathbf{p}$ ,  $N_{m2}$  maximum number of Newton-Picard iterations, initial conditions  $\mathbf{y}_0$ , initial basis  $\mathbf{Z}_0$ , integration time horizon  $t_0$ , the input/output integrator over a time interval  $tt$  and tolerance  $\sigma$ ,  $N'$  the number of initial samples,  $N''$  the limited number of iterations (expensive evaluations),  $N'''$  the maximum number of continuous iterations without the objective improvements and the convergent parameter  $\sigma$ , the maximum computational time  $H_t$

**Output:** the best solution pair  $(\mathbf{d}_{op}, G'(\mathbf{d}_{op}))$ , the iteration sample pairs  $(\mathbf{d}_{k1}, G'_{k1})|_{k1=1, \dots, i-1}$  and the best history sample pairs  $(\mathbf{d}^{k1}, G'^{k1})|_{k1=0, \dots, i-1}$ ,

- the number of optimisation iterations ( $i - 1$ ), the computational time  $t_i$ , the current value of  $ii$
- 1: Generate  $N'$  LHC initial inputs  $\mathbf{d}_0$  to cover the design space  $\mathbf{d}$
  - 2: Compute the corresponding results  $\mathcal{G}_0(\mathbf{d}_0)$  of the given  $\mathbf{d}_0$  through the EF-MC **Algorithm** 1 and the transformed function formulation  $G''(E_Y, V_Y, \mathbf{d})$
  - 3:  $i \leftarrow 0, ii \leftarrow 0, t_0 \leftarrow 0$
  - 4: Calculate the hyperparameters  $\sigma_0, \mathbf{l}_0$  of the initial Kriging model through maximising Eq.(17) with the collected dataset  $(\mathbf{d}_0, \mathcal{G}_0)$
  - 5: Record the current computational time  $t_1$  after the initial computations
  - 6: Record the current best sample pair  $(\mathbf{d}^0, G'^0)$  after the initial computations
  - 7:  $i \leftarrow 1$
  - 8: **while**  $i \leq N'' \ \& \ ii \leq N''' \ \& \ t_i \leq H_t$  **do**
  - 9:   Compute the location  $\mathbf{d}_i$  of the next sample by maximising Eq.(23) with the collected dataset  $(\mathbf{d}_{i-1}, \mathcal{G}_{i-1})$
  - 10:   Compute the corresponding  $G'_i(\mathbf{d}_i)$  of  $\mathbf{d}_i$  through the EF-MC **Algorithm** 1 and the function formulation  $G'(E_Y, V_Y, \mathbf{d})$
  - 11:   Put the current sample pair  $(\mathbf{d}_i, G'_i)$  into the collected dataset  $(\mathbf{d}_i, \mathcal{G}_i)$
  - 12:   **if**  $|G'_i - G'^{i-1}| \leq \sigma$  **then**
  - 13:      $ii \leftarrow ii + 1$
  - 14:   **else**
  - 15:      $ii \leftarrow 0$
  - 16:   **end if**
  - 17:   Record the current best sample pair  $(\mathbf{d}^i, G'^i)$  among the sample set  $(\mathbf{d}_i, \mathcal{G}_i)$
  - 18:   Calculate the updated hyperparameters  $\sigma_i, \mathbf{l}_i$  of the initial Kriging model through maximising Eq.(17) with the collected dataset  $(\mathbf{d}_i, \mathcal{G}_i)$
  - 19:    $i = i + 1$
  - 20:   Record the current computational time  $t_i$  after the this iteration
  - 21: **end while**
  - 22: Record the current best sample pair  $(\mathbf{d}^i, G'^i)$  as the best solution pair  $(\mathbf{d}_{op}, G'(\mathbf{d}_{op}))$
  - 23: **return** the best solution pair  $(\mathbf{d}_{op}, G'(\mathbf{d}_{op}))$ , the iteration sample pairs  $(\mathbf{d}_{k'}, G'_{k'})_{k'=1, \dots, i-1}$  and the best history sample pairs  $(\mathbf{d}^{k'}, G'^{k'})_{k'=0, \dots, i-1}$ , the number of optimisation iterations ( $i - 1$ ), the computational time  $t_i$ , the current value of  $ii$
- 

In this work, the above Bayesian optimisation Algorithm 2 deals with the

simple bound constraints  $g'_{cons}$  and can compute a good solution  $(\mathbf{d}_{op}, G'(\mathbf{d}_{op}))$  with the limited computational resources (computational time  $H$ , iteration steps  $N'$ ). For the cases with complex  $g'_{cons}$ , the corresponding expected improved functions should be introduced [56] and combined into the objective to satisfying the complex nonlinear constraints, which can also employed the above robust optimisation procedure.

### 3. Case studies

In order to prove the high efficiency of the proposed robust steady-state optimisation framework, a typical tubular reactor, displayed in Fig.2, was tested. For comparison purposes, the traditional gradient based optimisation method in MATLAB was utilised to solve the same problem Eq.(13) with the same EF-MC uncertainty propagation method as the above Algorithm 2. For all computational cases below, the gradient based optimisation method and the EF-MC uncertainty propagation method were running in MATLAB R2019a while the Algorithm 2 was implemented in Python 3.7.3/PyCharm 2018.3.5 on a Desktop (Intel Core(TM) i7-8700 CPU 3.2 GHz, 16 GB memory, 64-bit operating system, Windows 10).

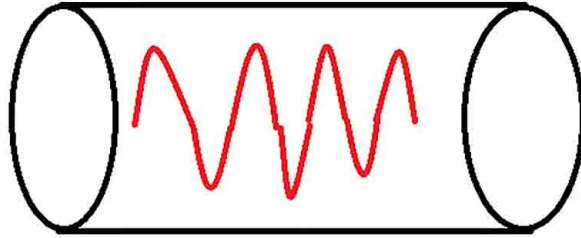


Figure 2: Tubular reactor with cooling zones

An exothermic first-order irreversible reaction happens  $A \rightarrow B$  in a tubular reactor [57, 29]. We consider three cooling zones on the jacket of the reactor, temperatures of which could be controlled independently. The governed

mathematical equations are below.

$$\begin{aligned}
\frac{\partial x_1}{\partial t} &= \frac{1}{Pe_1} \frac{\partial^2 x_1}{\partial y^2} - \frac{\partial x_1}{\partial y} + D_a(1 - x_1) \exp\left(\frac{x_2}{1 + \frac{x_2}{\gamma}}\right) \\
\frac{\partial x_2}{\partial t} &= \frac{1}{L_e Pe_2} \frac{\partial^2 x_2}{\partial y^2} - \frac{1}{L_e} \frac{\partial x_2}{\partial y} - \frac{x_2}{L_e \beta} + \frac{B}{L_e} D_a(1 - x_1) \exp\left(\frac{x_2}{1 + \frac{x_2}{\gamma}}\right) + \frac{\beta x_3}{L_e} \\
x_3(y) &= \sum_{i=1}^3 (H(y - y_{i-1}) - H(y - y_i)) x_{3i}
\end{aligned} \tag{24}$$

Here  $x_1$  and  $x_2$  are two state variables, the dimensionless concentration and dimensionless temperature respectively, and  $y$  is the dimensionless longitudinal coordinate. Systems parameters are the Damkohler number  $D_a$ , the Lewis number  $L_e$ , the Peclet number for mass transport  $Pe_1$ , the Peclet number for heat transport  $Pe_2$ , a dimensionless heat transfer coefficient  $\beta$ , the dimensionless adiabatic temperature rise  $B$ , the dimensionless adiabatic wall temperatures  $x_3$  of cooling zones and the dimensionless activation energy  $\gamma$ . The above two PDEs are accompanied with the boundary conditions:

$$\begin{aligned}
\frac{\partial x_1}{\partial y} - Pe_1 x_1 &= 0 \\
\frac{\partial x_2}{\partial y} - Pe_2 x_2 &= 0, \quad \text{at } y = 0 \\
\frac{\partial x_1}{\partial y} &= 0 \\
\frac{\partial x_2}{\partial y} &= 0, \quad \text{at } y = 1
\end{aligned} \tag{25}$$

The above two PDEs were spatially discretised on 250 ( $N$ ) computational nodes by central finite difference method, and then integrated in time by explicit fourth-order Runge-Kutta method. Then the whole computational codes were made into a input-output black-box simulator, where only the input-output data from the simulator could be used. The reporting time horizon was set to be  $1e - 06$  second ( $tt$ ) for the black-box simulator. Inputs include fixed system parameters and uncertain parameters while outputs would be the dimensionless temperatures and concentrations.  $x_{1_{exit}}$  is the dimensionless output concentration at the end,  $\mu(x_{1_{exit}})$  ( $E_Y$ ) is the mean value of the uncertain output concentration and  $\sigma(x_{1_{exit}})$  ( $V_Y$ ) is the standard

variance value of the uncertain output concentration. In this work, the above optimisation framework was studied to robustly find the good operation temperatures for the cooling zones to maximize the dimensionless concentration  $x_1$  under parametric uncertainty.

### 3.1. Single parametric uncertainty

As an illustrative example, a simple case with single parametric uncertainty was firstly studied. Here, system parameters  $Le = 1.0$ ,  $Pe1 = Pe2 = 5.0$ ,  $\gamma = 20.0$ ,  $\beta = 1.50$ ,  $B = 12.0$  are fixed while the distributional uncertain parameter  $\mathbf{p}$  is Damkohler number  $D_a$ , which could exploit the rich parametric behaviours of the dynamic systems. The parametric uncertainty follows a normal distribution here.

$$D_a \sim N(0.1, 0.01^2) \quad (26)$$

The mathematical formulation could be seen as Eq.(27). In this optimisation problem, the objective is a weighted function, which aims to obtain the larger output product concentration with smaller fluctuations at the end of exit under single parametric uncertainty. Here the fixed weight factor  $r$  between the two terms was set to be 0.6. The design variables are three temperatures of the three cooling zones. The constraints include the black-box simulator equations with steady-state uncertainty quantification, box-constraints for the design variables and a normal distribution for the parametric uncertainty.

$$\begin{aligned} \max_{x_3} \quad & r\mu(x_{1exit}) - (1-r)\sigma(x_{1exit}) \\ \text{s.t. } & \text{black-box simulator Eq.(24,25) with steady-state } UQ \\ & D_a \sim N(0.1, 0.01^2) \\ & 0 \leq x_{3i} \leq 4, \quad i = 1, 2, 3 \end{aligned} \quad (27)$$

Through eliminating the equality constraints and the uncertainty propagation steps, then the above problem Eq.(27) could be transformed into the following unconstrained nonlinear programming as the general formulation Eq.(13):

$$\begin{aligned} \max_{x_3} \quad & r\mu(x_3) - (1-r)\sigma(x_3) \\ \text{s.t.} \quad & \\ & 0 \leq x_{3i} \leq 4, \quad i = 1, 2, 3 \end{aligned} \quad (28)$$

Using EF-MC uncertainty propagation method (algorithm 1) with the black-box dynamic solver, the Bayesian robust steady-state optimisation algorithm 2 was constructed for the problem (Eq. (28)). As a part of uncertainty propagation process, 3000 ( $M$ ) MC samples were used, the dimensionality of subspace was fixed to be 10 ( $m$ ) and the limited number of Newton-Picard iteration was set to be 60 ( $N_{m2}$ ). Previous research [1] shows that EF-MC uncertainty propagation algorithm could accelerate the UQ computations by around one magnitude compared with the standard MC method. The convergent condition was set to be the maximum iteration size being 100 ( $N''$ ) or the maximum computational time being 160 ( $H_t$ ) hours, or no improvement for continuous 10 ( $N'''$ ) iterations. Specifically for this case, Latin hypercube sampling method was first utilised to collect 8 ( $N'$ ) initial samples, taking 9.75 ( $t_0$ ) hours. Then the initial samples were used to construct the surrogate Kriging model 17, compute the location  $\mathbf{d}_*$  of the new sample point 23, calculate the function value  $G'(\mathbf{d}_*)$  and check the convergence  $|G'_i - G'^{i-1}| \leq \sigma$  as the optimisation algorithm 2. The optimisation process, as showed in Fig.3, converged to the optimal objective being 0.5999 within 33 ( $i - 1$ ) iterations and 30.91 ( $t_i$ ) hours. For comparative purposes, the 8 ( initial samples are took as the initial points for the derivative-based optimisation algorithms. Here, we used the efficient interior-point algorithms in *fmincon* function of MATLAB, with the maximum computational time set to be 160 ( $H_t$ ) hours. The results could be seen in Fig.4 and Tab.1.

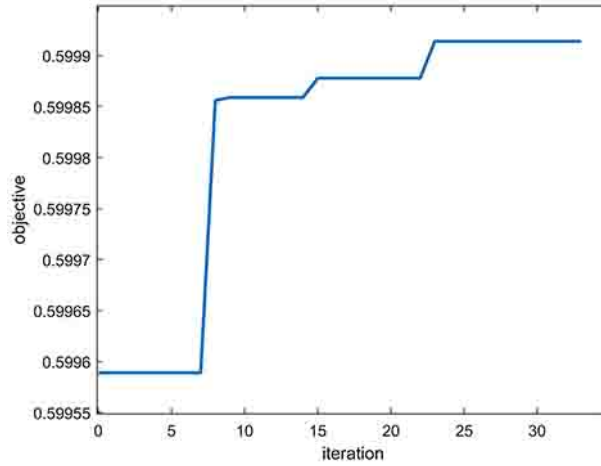
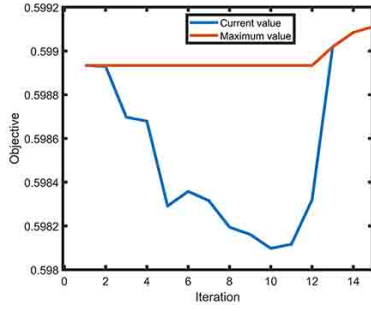


Figure 3: Bayesian optimisation of cooling process with single parametric uncertainty

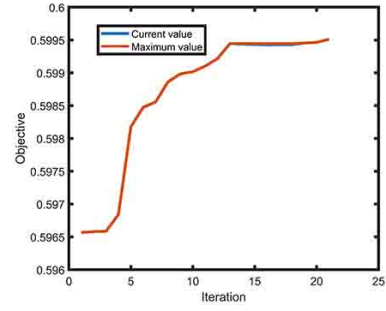
As showed in Tab.1, the Bayesian global optimisation method took more iterations but less function evaluations to reach the convergent condition, finding a quite good solution, while the other eight gradient-based optimisation processes took less iterations, exceeding the maximum computational time due to more function evaluations for computing first-order or/and higher order derivatives. The around 75% computational saving implies the high computational efficiency of the Bayesian optimization method. Meanwhile, the Bayesian optimisation method requires more initial samples to obtain the global information of the feasible search space while the gradient-based optimisation method need only one initial point. Here, the gradient-based optimisations employed the different initial guesses from the same initial sample sets for the Bayesian optimisation procedure. However, all the solutions from the gradient-based method is worse than the one found from the Bayesian optimisation approach. There are two possible reasons. The first one is that the computed derivatives are inaccurate, leading to wrong search directions or zigzag optimized processes. For example, the gradient-based optimisation process with first initial point took many iteration steps to search worse solutions, resulting in a waste of computational costs. The other one is that the Bayesian optimization method can skip the basin of local convergence because of the exploration ability of acquisition function while the gradient-based method can only search around the local optima. That may be a reasonable explanation for the appearance of multiple stable stages during the Bayesian optimisation process while less stable stages within gradient-based search. In terms of the computational accuracy and cost, the Bayesian robust optimisation algorithm 2 is superior over the gradient-based method for the large-scale distributed systems with uncertainty. It should be noted the EF-MC uncertainty quantification method, achieving near 90% computational saving [1] compared with the traditional MC method, could scalably reduce the computations for all the optimisation processes.

Table 1: Comparative results from different optimisation process

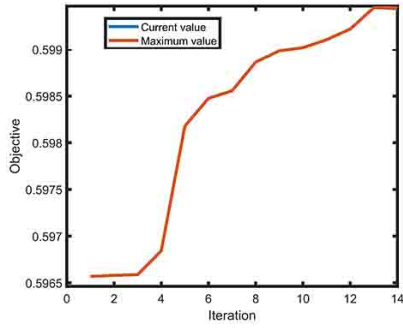
<b>Optimization method</b>	<b>Initial samples</b>	<b>Number iteration</b>	<b>Number function evaluation</b>	<b>Optimal value</b>	<b>Computational time (h)</b>
Bayesian	8 samples	33	41	0.5999	40.66
Gradient	1st sample	14	78	0.5991	160
Gradient	2nd sample	20	104	0.5995	160
Gradient	3rd sample	13	69	0.5994	160
Gradient	4th sample	14	82	0.5991	160
Gradient	5th sample	9	54	0.5993	160
Gradient	6th sample	18	96	0.5997	160
Gradient	7th sample	15	93	0.5995	160
Gradient	8th sample	17	112	0.5996	160



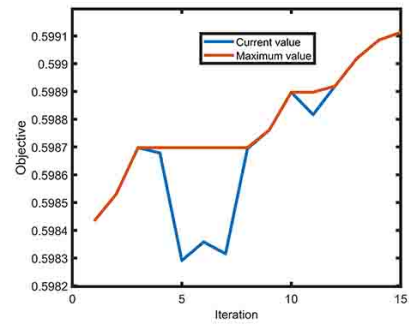
(a) Optimisation with first point



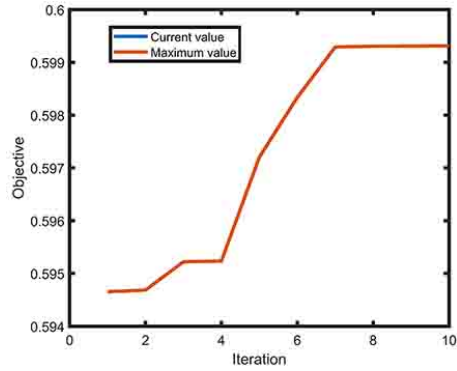
(b) Optimisation with second point



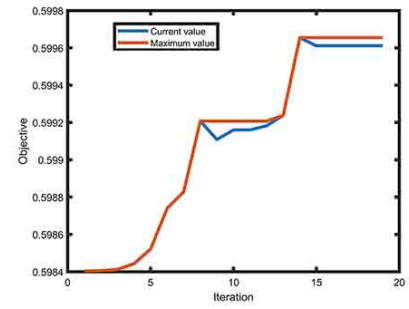
(c) Optimisation with third point



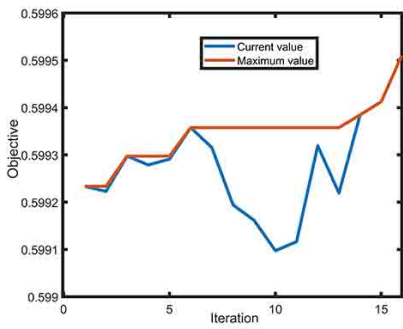
(d) Optimisation with fourth point



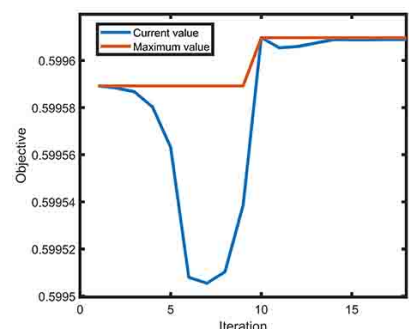
(e) Optimisation with fifth point



(f) Optimisation with sixth point



(g) Optimisation with seventh point



(h) Optimisation with eighth point

Figure 4: Gradient-based optimisation for cooling a tubular reactor with single parametric uncertainty

### 3.2. Multivariate uncertainty

Due to the high computational efficiency in single parametric uncertainty case, the above robust steady-state optimisation framework would be extended to solve a multivariate uncertainty case.

Here system parameters  $Le = 1.0, Pe1 = Pe2 = 5.0, \gamma = 20.0, \beta = 1.50$  are still fixed into the black-box simulator, while the two distributional uncertain parameters are Damkohler number  $D_a$  and adiabatic temperature rise  $B$ . The independent parameters was studied, satisfying the following normal distributions:

$$\begin{aligned} D_a &\sim N(0.08, 0.008^2) \\ B &\sim N(8, 0.8^2) \end{aligned} \tag{29}$$

In this optimisation problem (Eq. (30)), the objective is still a weighted function, which aims to obtain larger output product concentration but smaller fluctuations at the end of the reactor under multivariate uncertainty. Here, the fixed weight  $r$  between the two objective terms is set to be 0.35. The design variables are still the three temperatures of the three different cooling zones. The constraints includes the black-box simulator equations with steady-state uncertainty quantification, box-constraints for the design variables and the two normal distributions for the multivariate uncertainty. Similar to the previous case, the problem (Eq. (30)) could be transformed into a black-box unconstrained nonlinear programming manner (Eq. (31)), which can be solved by the gradient optimisation method and the above Bayesian robust steady-state optimisation algorithm 2.

$$\begin{aligned} \max_{x_3} \quad & r\mu(x_{1exit}) - (1-r)\sigma(x_{1exit}) \\ \text{s.t. } & \text{black-box simulator Eq.(24, 25) with steady-state } UQ \\ & D_a \sim N(0.08, 0.008^2) \\ & B \sim N(8, 0.8^2) \\ & 0 \leq x_{3i} \leq 2, \quad i = 1, 2, 3 \end{aligned} \tag{30}$$

$$\begin{aligned} \max_{x_3} \quad & r\mu(x_3) - (1-r)\sigma(x_3) \\ \text{s.t.} \quad & \\ & 0 \leq x_{3i} \leq 2, \quad i = 1, 2, 3 \end{aligned} \tag{31}$$

Similar to the previous case study, here 4000 ( $M$ ) MC samples, after the independent computational experiments, were used as the preconditions for

performing uncertainty propagation algorithm 1 with the inputs/outputs integrator. Additionally, the dimensionality size of subspace was fixed to be 10 ( $m$ ) while the limited number of Newton-Picard iteration was set to be 60 ( $N_{m2}$ ). Then both of the Bayesian optimisation procedure (algorithm 2 ) and the gradient based optimisation method (*fmincon* function in MATLAB) were implemented with the uncertainty propagation procedure (algorithm 1). As for the Bayesian optimisation process, the convergent condition was ensured if any of the three conditions was satisfied, including the exceeding the maximum iteration size being 100 ( $N''$ ) or the maximum computational time being 160 ( $H_t$ ) hours, and no improvement after continuous 10 ( $N'''$ ) iterations. Latin hypercube sampling method was utilised to collect 8 ( $N'$ ) initial samples, taking 17.75 ( $t_0$ ) hours. The information of sample sets was exploited to compute the hyperparameters for the Kriging model, which was then adopted to compute the locations of the next samples and the corresponding function values. As displayed in Fig.5, the Bayesian optimisation computations finally converged to the optimal solution within 16 ( $i - 1$ ) iterations, taking 29.79 ( $t_i$ ) hours. For the comparative purposes, the 8 initial samples were took as the initial points for the derivative-based optimisation algorithms. Here the *fmincon* function in MATLAB was used to perform the gradient-based optimisation tasks with the maximum computational time being 160 hours. The results could be seen in Fig.6 and Tab.2.

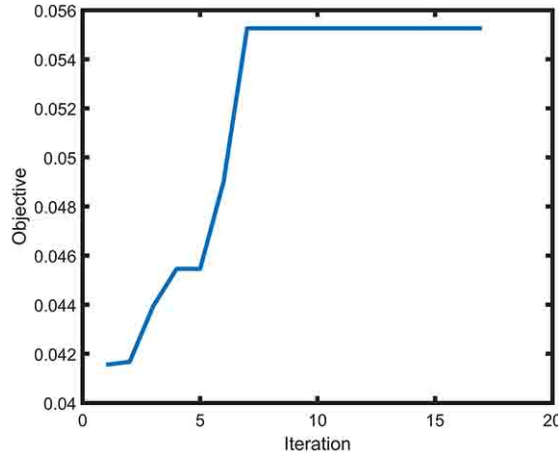
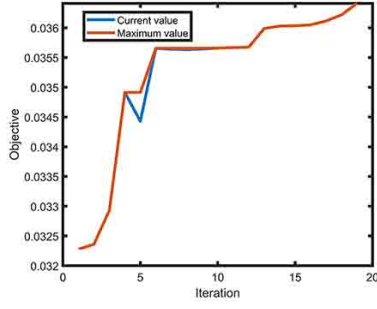


Figure 5: Bayesian optimisation of cooling process with multivariate uncertainty

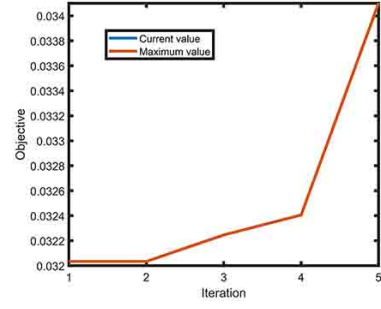
As showed in Tab.2, the Bayesian global optimisation method took the similar number of iteration steps but less function evaluations while the other eight gradient-based optimisation processes took more function evaluations due to the demand of computing derivatives or unreliable derivatives and the resulting expensive optimisation routines. The around 70% computational saving implies the high computational efficiency of the Bayesian optimization algorithm 2. Furthermore, the eight gradient-based optimisation procedures utilised the different initial guesses from the same initial sample sets of the Bayesian optimisation routine. However, all the solutions from the gradient-based method are much worse than the solution from Bayesian optimisation. The possible reason is that all the gradient-based optimisations can only touch the local optima possibly missing better optima , while the Bayesian optimisation procedure could globally explore the uncertain space. Another possible reason would be the limited computational time. For example, the gradient-based optimisation algorithm with the second point maybe spend much computational cost on the computing the derivatives, leading to an early stop although the current gradients can guide the optimisation routine toward a better solution. Generally, the Bayesian strategy based robust optimisation framework showed superior advantages over the gradient-based one in terms of both the computational cost and the solution quality.

Table 2: Comparative results from different optimisation process under multivariate uncertainty

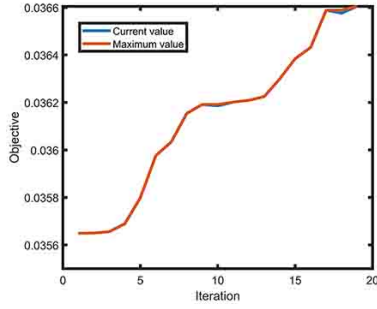
<b>Optimisation method</b>	<b>Initial samples</b>	<b>Number iteration</b>	<b>Number function evaluation</b>	<b>Optimal value</b>	<b>Computational time (h)</b>
Bayesian	8 samples	16	24	0.0553	47.54
Gradient	1st sample	18	84	0.0364	160
Gradient	2nd sample	4	50	0.0341	160
Gradient	3rd sample	18	80	0.0366	160
Gradient	4th sample	16	77	0.0365	160
Gradient	5th sample	10	47	0.0354	160
Gradient	6th sample	12	52	0.0284	160
Gradient	7th sample	13	57	0.0278	160
Gradient	8th sample	10	44	0.0547	160



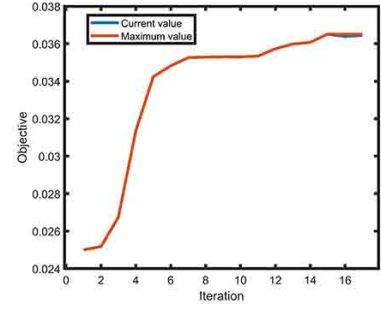
(a) Optimisation with 1th point



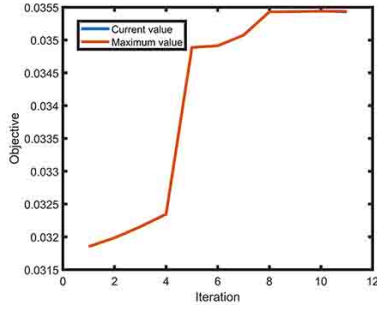
(b) Optimisation with 2nd point



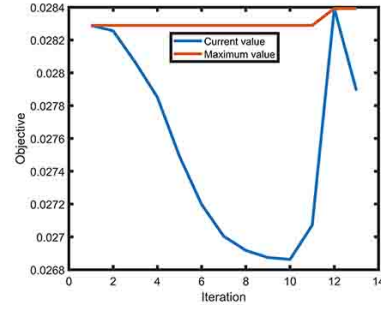
(c) Optimisation with 3rd point



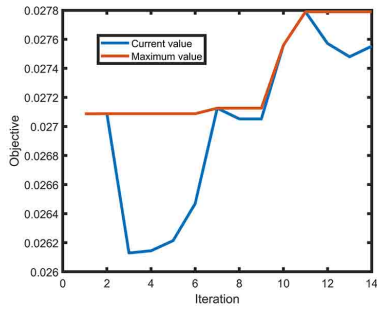
(d) Optimisation with 4th point



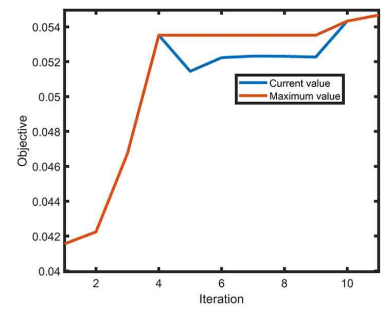
(e) Optimisation with 5th point



(f) Optimisation with 6th point



(g) Optimisation with 7th point



(h) Optimisation with 8th point

Figure 6: Gradient-based optimisation processes for a cooling tubular reactor under multivariate uncertainty

## 4. Conclusion

This paper proposes a robust steady-state optimisation framework for large-scale distributed systems with uncertainty. The proved EF-MC uncertainty analysis method was first employed to deal with parametric uncertainty. The accelerated system including uncertainty was then utilised by the proposed Bayesian global optimisation strategy (algorithm 2), combining the surrogate Kriging model and the expected improvement function. Finally, a practical cooling tubular reactor with single and multivariate uncertainty cases were adopted to show that the proposed robust optimisation framework could reduce the computational cost up to 70% and achieve better solutions compared with the traditional gradient-based methods. In the future, the robust optimisation framework will be applied to more practical large-scale computational fluid dynamic and multi-scale cases.

## References

- [1] M. Tao, Global optimisation, model predictive control and uncertainty quantification methodologies for distributed parameter systems, in: Chapter 5, PhD thesis, The University of Manchester, 2021, p. 104.
- [2] N. Ahmed, Distributed parameter systems, Encyclopedia of physical science and technology 1 (1987) 277.
- [3] T. J. Sullivan, Introduction to uncertainty quantification, volume 63, Springer, 2015.
- [4] D. Xiu, Numerical methods for stochastic computations: a spectral method approach, Princeton university press, 2010.
- [5] D. L. Ma, S. H. Chung, R. D. Braatz, Worst-case performance analysis of optimal batch control trajectories, AIChE journal 45 (1999) 1469–1476.
- [6] D. Chaffart, S. Rasoulia, L. A. Ricardez-Sandoval, Distributional uncertainty analysis and robust optimization in spatially heterogeneous multiscale process systems, AIChE Journal 62 (2016) 2374–2390.
- [7] D. Chaffart, L. A. Ricardez-Sandoval, Robust dynamic optimization in heterogeneous multiscale catalytic flow reactors using polynomial chaos expansion, Journal of Process Control 60 (2017) 128–140.

- [8] D. E. Shen, R. D. Braatz, Polynomial chaos-based robust design of systems with probabilistic uncertainties, *AIChE Journal* 62 (2016) 3310–3318.
- [9] R. Ghanem, D. Higdon, H. Owhadi, *Handbook of uncertainty quantification*, volume 6, Springer, 2017.
- [10] Z. Nagy, R. D. Braatz, Distributional uncertainty analysis using power series and polynomial chaos expansions, *Journal of Process Control* 17 (2007) 229–240.
- [11] J. C. Helton, F. J. Davis, Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems, *Reliability Engineering & System Safety* 81 (2003) 23–69.
- [12] F. Nobile, R. Tempone, C. G. Webster, A sparse grid stochastic collocation method for partial differential equations with random input data, *SIAM Journal on Numerical Analysis* 46 (2008) 2309–2345.
- [13] M. B. Giles, Multilevel monte carlo methods, *Acta Numerica* 24 (2015) 259.
- [14] S. Heinrich, Multilevel monte carlo methods, in: *International Conference on Large-Scale Scientific Computing*, Springer, pp. 58–67.
- [15] G. Kimaev, L. A. Ricardez-Sandoval, A comparison of efficient uncertainty quantification techniques for stochastic multiscale systems, *AIChE Journal* 63 (2017) 3361–3373.
- [16] F. Nobile, R. Tempone, C. G. Webster, An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data, *SIAM Journal on Numerical Analysis* 46 (2008) 2411–2442.
- [17] D. Xiu, G. E. Karniadakis, Supersensitivity due to uncertain boundary conditions, *International journal for numerical methods in engineering* 61 (2004) 2114–2138.
- [18] G. M. Shroff, H. B. Keller, Stabilization of unstable procedures: the recursive projection method, *SIAM Journal on numerical analysis* 30 (1993) 1099–1120.

- [19] R. Lozano, B. Brogliato, O. Egeland, B. Maschke, Dissipative systems, in: *Dissipative Systems Analysis and Control*, Springer, 2000, pp. 111–166.
- [20] A. R. Conn, K. Scheinberg, L. N. Vicente, *Introduction to derivative-free optimization*, SIAM, 2009.
- [21] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, Optimization by simulated annealing, *science* 220 (1983) 671–680.
- [22] L. D. Chambers, *Practical handbook of genetic algorithms: complex coding systems*, volume 3, CRC press, 2019.
- [23] D. R. Jones, M. Schonlau, W. J. Welch, Efficient global optimization of expensive black-box functions, *Journal of Global optimization* 13 (1998) 455–492.
- [24] M. Pelikan, D. E. Goldberg, *Bayesian optimization algorithm: From single level to hierarchy*, University of Illinois at Urbana-Champaign Urbana, IL, 2002.
- [25] J. P. Kleijnen, Kriging metamodeling in simulation: A review, *European journal of operational research* 192 (2009) 707–716.
- [26] D. Huang, T. T. Allen, W. I. Notz, N. Zeng, Global optimization of stochastic black-box systems via sequential kriging meta-models, *Journal of global optimization* 34 (2006) 441–466.
- [27] P. Hennig, C. J. Schuler, Entropy search for information-efficient global optimization, *The Journal of Machine Learning Research* 13 (2012) 1809–1837.
- [28] M. L. Stein, *Interpolation of spatial data: some theory for kriging*, Springer Science & Business Media, 2012.
- [29] M. Tao, C. Theodoropoulos, Uncertainty analysis and model reduction based global optimisation of distributed large-scale systems, in: *Computer Aided Chemical Engineering*, volume 48, Elsevier, 2020, pp. 1975–1980.
- [30] C. Multiphysics, *Introduction to comsol multiphysics®*, COMSOL Multiphysics, Burlington, MA, accessed Feb 9 (1998) 2018.

- [31] A. Fluent, Ansys fluent, Academic Research. Release 14 (2015).
- [32] H. Jasak, A. Jemcov, Z. Tukovic, et al., Openfoam: A c++ library for complex physics simulations, in: International workshop on coupled methods in numerical dynamics, volume 1000, IUC Dubrovnik Croatia, pp. 1–20.
- [33] A. Stanton, D. Wiegand, G. Stanton, Probability reliability and statistical methods in engineering design, eBook (2000).
- [34] E. Luna-Ortiz, C. Theodoropoulos, An input/output model reduction-based optimization scheme for large-scale systems, *Multiscale Modeling & Simulation* 4 (2005) 691–708.
- [35] P.-L. Hsu, H. Robbins, Complete convergence and the law of large numbers, *Proceedings of the National Academy of Sciences of the United States of America* 33 (1947) 25.
- [36] D. P. Bertsekas, W. Hager, O. Mangasarian, *Nonlinear programming*, Athena Scientific Belmont, MA, 1998.
- [37] J. Nocedal, S. Wright, *Numerical optimization*, Springer Science & Business Media, 2006.
- [38] D. G. Krige, A statistical approach to some basic mine valuation problems on the witwatersrand, *Journal of the Southern African Institute of Mining and Metallurgy* 52 (1951) 119–139.
- [39] S. Jeong, Y. Minemura, S. Obayashi, Optimization of combustion chamber for diesel engine using kriging model, *Journal of Fluid Science and Technology* 1 (2006) 138–146.
- [40] S. Kawai, K. Shimoyama, Kriging-model-based uncertainty quantification in computational fluid dynamics, in: 32nd AIAA Applied Aerodynamics Conference, p. 2737.
- [41] O. W. Tsui, N. C. Coops, M. A. Wulder, P. L. Marshall, Integrating airborne lidar and space-borne radar via multivariate kriging to estimate above-ground biomass, *Remote Sensing of Environment* 139 (2013) 340–352.

- [42] S. Sakata, F. Ashida, M. Zako, Kriging-based approximate stochastic homogenization analysis for composite materials, *Computer methods in applied mechanics and engineering* 197 (2008) 1953–1964.
- [43] C. E. Rasmussen, Gaussian processes in machine learning, in: *Summer school on machine learning*, Springer, pp. 63–71.
- [44] T. W. Simpson, J. Poplinski, P. N. Koch, J. K. Allen, Metamodels for computer-based engineering design: survey and recommendations, *Engineering with computers* 17 (2001) 129–150.
- [45] K. P. Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.
- [46] C. K. Williams, C. E. Rasmussen, *Gaussian processes for machine learning*, volume 2, MIT press Cambridge, MA, 2006.
- [47] M. Stein, Large sample properties of simulations using latin hypercube sampling, *Technometrics* 29 (1987) 143–151.
- [48] D. C. Liu, J. Nocedal, On the limited memory bfgs method for large scale optimization, *Mathematical programming* 45 (1989) 503–528.
- [49] P. Boyle, *Gaussian processes for regression and optimisation*, Victoria University of Wellington, 2007.
- [50] E. Brochu, V. M. Cora, N. De Freitas, A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, *arXiv preprint arXiv:1012.2599* (2010).
- [51] D. R. Jones, A taxonomy of global optimization methods based on response surfaces, *Journal of global optimization* 21 (2001) 345–383.
- [52] M. Schonlau, *Computer experiments and global optimization*, University of Waterloo, 1997.
- [53] M. Schonlau, W. J. Welch, D. R. Jones, Global versus local search in constrained optimization of computer models, *Lecture Notes-Monograph Series* (1998) 11–25.

- [54] M. Sasena, P. Papalambros, P. Goovaerts, Metamodeling sampling criteria in a global optimization framework, in: 8th Symposium on Multidisciplinary Analysis and Optimization, p. 4921.
- [55] M. J. Sasena, Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations, Ph.D. thesis, University of Michigan Ann Arbor, MI, 2002.
- [56] M. Abdolshah, A. Shilton, S. Rana, S. Gupta, S. Venkatesh, Expected hypervolume improvement with constraints, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, pp. 3238–3243.
- [57] K. F. Jensen, W. H. Ray, The bifurcation behavior of tubular reactors, *Chemical Engineering Science* 37 (1982) 199–222.

# Chapter 7

## Conclusions and future work

### 7.1 Conclusions

Complex distributed parameter systems could cover a wide range of practical physical applications, including chemical and biochemical reactors. However, intelligent operations on these practical applications, such as optimisation, control and uncertainty quantification, are still facing many computational challenges. This thesis is devoted to developing several computational methodologies to overcome some important challenges for large-scale distributed parameter systems.

Previous work in the literature has developed a model reduction based local optimisation methodology (Bonis & Theodoropoulos, 2012). Chapter 3 aims to present a model reduction based global optimisation computational framework for large-scale distributed parameter systems. Firstly, a large number of inputs/outputs samples would be generated using systematical data collection strategies. Then the off-line double model reductions, principal component analysis and artificial neural networks, would be implemented to generate the simple but accurate reduced surrogate models, which would be employed by the advanced general-purpose global optimisation solvers. To further improve the performance of the initial optimisation framework, two different reformulation techniques (piecewise linear affine techniques and deep neural networks with ReLU activation function) were adopted to transform the original nonlinear programming into the less complex mixed integer linear programming (MILP), speeding up the global optimisation procedure. The proposed computational framework, especially the second improvement (deep neural network with ReLU activation function) can find the approximate global optima with desirable accuracy and computational speed for sev-

eral validated practical examples.

To intelligently control the performance of large-scale distributed systems, a robust non-linear model predictive control strategy is presented in Chapter 4 to address the computational challenges including process uncertainty and intensive computations. Model predictive control was introduced to deal with the plant-model mismatch, recursively updating the output responses to eliminate some of process uncertainty such as the slight measurement noise. However, substantial uncertainties such as the important parametric uncertainty, would significantly affect the output responses. Here, the popular polynomial chaos expansion method was adopted to deal with parametric uncertainty, which could efficiently compute the statistical moments and probabilistic bounds of quantities of interest for large-scale systems. Moreover, the double model construction techniques, proper orthogonal decomposition and recurrent neural networks, were employed to represent the high-dimensional dynamics of the computed statistical moments and probabilistic bounds. The reduced surrogate models could be reformulated into a set of sequential MILP sub-problems (similar to Chapter 3), which could be globally solved to find the approximate global optima within the control procedure. Computational results of the practical chemical and biochemical examples show that the proposed robust control framework can improve process productions satisfying the rigorous time-space process constraints (quality requirements and safety requirements).

The popular polynomial chaos expansion method (In Chapter 4) requires the prior knowledge about the type of uncertainty distribution to choose the suitable polynomials, and may be not so accurate for high complex systems due to the inherent low order characteristics. Chapter 5 focuses on developing efficient uncertainty propagation methods for both deterministic and stochastic distributed parameter systems. The key idea is to employ the "Equation-free" approaches to accelerate the computations of large-scale dynamic systems to steady states within the traditional Monte Carlo-related uncertainty quantification algorithm. Specifically, the recursive projection method would efficiently compute the distributional steady states or coarse-scale steady states for large-scale systems while the lifting and restriction operations perform the bridge between the stochastic simulator and coarse-scale computations. Deterministic practical examples including a chemical tubular reactor and the FitzHugh-Nagumo model were employed, demonstrating that the proposed uncertainty quantification algorithm can reduce the computational costs up to 90% with highly accurate results. While the stochas-

tic case study of the oxidation reactions of CO on a lattice catalytic surface shows that the proposed algorithm could accurately compute the coarse-scale output distributions with slightly less computations.

Finally, Chapter 6 aim to develop a robust steady-state optimisation framework for large-scale distributed systems under parametric uncertainty. The equation-free Monte Carlo uncertainty propagation algorithms in Chapter 5 would be employed to address the parametric uncertainty. The accelerated system including uncertainty was then utilised by the proposed Bayesian global optimisation strategy, combining the surrogate Kriging model and the expected improvement function. Finally, a practical cooling tubular reactor with single and multivariate uncertainty cases were adopted to show that the proposed robust optimisation framework could reduce the computational cost up to 70% and achieve better solutions compared with the traditional gradient-based methods.

## **7.2 Future work**

Regarding future work, several extensions could be explored as follows:

Firstly, the model reduction based global optimisation framework (Chapter 3) assumes enough representative samples as a basis to construct the reduced order models. Smart sampling methods to achieve optimal trade-off between quality and quantity are important for improving both efficiency and accuracy, as well as verification methods to guarantee the accuracy of the computed solutions(Botoeva et al., 2020). Moreover, global optimisation even using reduced surrogate models is still computationally expensive. Advanced data techniques and MILP algorithms (R. Anderson et al., 2018) may further improve computational efficiency of this optimisation framework.

Secondly, a less conservative strategy will be exploited in the robust model predictive control strategy (Chapter 4), to further enhance the process performances. Moreover, the proposed control policy assumes that the offline samples are enough to build accurate predictive models. Online control with adaptive models will be considered when the offline samples are not enough to support the high-fidelity surrogate models.

Thirdly, the proposed novel equation-free Monte Carlo (EF-MC) uncertainty quantifica-

tion algorithms (Chapter 5) only test the stochastic case with single distributional uncertainty. Multivariate cases for the stochastic problems will be employed to investigate the efficiency of our EF-MC algorithms.

Fourthly, the robust steady-state optimisation framework (Chapter 6) are employed for two PDEs based reaction process. In this future, this computational framework will be applied to more complex large-scale computational fluid dynamic and multi-scale cases.

# Bibliography

- Adjiman, C. S., Dallwig, S., Floudas, C. A., & Neumaier, A. (1998). A global optimization method,  $\alpha$ BB, for general twice-differentiable constrained nlp—i. theoretical advances. *Computers & Chemical Engineering*, 22(9), 1137–1158.
- Adjiman, C. S., & Floudas, C. A. (1996). Rigorous convex underestimators for general twice-differentiable problems. *Journal of Global Optimization*, 9(1), 23–40.
- Ahmed, N. (1987). Distributed parameter systems. *Encyclopedia of physical science and technology*, 1, 277.
- Al-Dunainawi, Y., & Abbod, M. F. (2016). Evolutionary based optimisation of multivariable fuzzy control system of a binary distillation column. *2016 UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim)*, 127–132.
- Alkhatib, A., & Babaei, M. (2016). Applying the multilevel monte carlo method for heterogeneity-induced uncertainty quantification of surfactant/polymer flooding. *SPE Journal*, 21(04), 1192–1203.
- Anderson, D. F., & Higham, D. J. (2012). Multilevel monte carlo for continuous time markov chains, with applications in biochemical kinetics. *Multiscale Modeling & Simulation*, 10(1), 146–179.
- Anderson, J. D., & Wendt, J. (1995). *Computational fluid dynamics* (Vol. 206). Springer.
- Anderson, R., Huchette, J., Tjandraatmadja, C., & Vielma, J. P. (2018). Strong convex relaxations and mixed-integer programming formulations for trained neural networks. *arXiv preprint arXiv:1811.01988*.
- Armaou, A., & Christofides, P. D. (2002). Dynamic optimization of dissipative pde systems using nonlinear order reduction. *Chemical engineering science*, 57(24), 5083–5114.

- Armaou, A., I. Siettos, C., & G. Kevrekidis, I. (2004). Time-steppers and ‘coarse’ control of distributed microscopic processes. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 14(2), 89–111.
- Armaou, A., & Varshney, A. (2004). Dynamic optimization of dissipative pdes using control vector parameterization: Application to gan thin film epitaxy. *Proceedings of the 2004 American Control Conference*, 1, 279–286.
- Bajaj, I., & Hasan, M. F. (2020). Global dynamic optimization using edge-concave underestimator. *Journal of Global Optimization*, 77(3), 487–512.
- Bansal, V., Perkins, J., Pistikopoulos, E., Ross, R., & Van Schijndel, J. (2000). Simultaneous design and control optimisation under uncertainty. *Computers & Chemical Engineering*, 24(2-7), 261–266.
- Bertsekas, D. P. (1976). On penalty and multiplier methods for constrained minimization. *SIAM Journal on Control and Optimization*, 14(2), 216–235.
- Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3), 464–501.
- Bhattacharyya, S. (2017). Robust control under parametric uncertainty: An overview and recent results. *Annual Reviews in Control*, 44, 45–77.
- Biegler, L. T. (1984). Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers & chemical engineering*, 8(3-4), 243–247.
- Biegler, L. T. (2010). *Nonlinear programming: Concepts, algorithms, and applications to chemical processes*. SIAM.
- Biegler, L. T., Grossmann, I. E., & Westerberg, A. W. (1997). Systematic methods for chemical process design.
- Biegler, L. T., & Zavala, V. M. (2009). Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization. *Computers & Chemical Engineering*, 33(3), 575–582.

- Bird, R. B., Stewart, W. E., & Lightfoot, E. N. (2006). Transport phenomena, revised 2 nd edition.
- Boggs, P. T., & Tolle, J. W. (1989). A strategy for global convergence in a sequential quadratic programming algorithm. *SIAM journal on Numerical Analysis*, 26(3), 600–623.
- Bonis, I., & Theodoropoulos, C. (2012). Model reduction-based optimization using large-scale steady-state simulators. *Chemical engineering science*, 69(1), 69–80.
- Bonis, I., Xie, W., & Theodoropoulos, C. (2013). Multiple model predictive control of dissipative pde systems. *IEEE Transactions on Control Systems Technology*, 22(3), 1206–1214.
- Botoeva, E., Kouvaros, P., Kronqvist, J., Lomuscio, A., & Misener, R. (2020). Efficient verification of relu-based neural networks via dependency analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34, 3291–3299.
- Boudart, M., & Djéga-Mariadassou, G. (2014). *Kinetics of heterogeneous catalytic reactions*. Princeton University Press.
- Boukouvala, F., & Floudas, C. A. (2017). Argonaut: Algorithms for global optimization of constrained grey-box computational problems. *Optimization Letters*, 11(5), 895–913.
- Boukouvala, F., Hasan, M. F., & Floudas, C. A. (2017). Global optimization of general constrained grey-box models: New method and its application to constrained pdes for pressure swing adsorption. *Journal of Global Optimization*, 67(1-2), 3–42.
- Bradford, E., & Imsland, L. (2019). Output feedback stochastic nonlinear model predictive control for batch processes. *Computers & Chemical Engineering*, 126, 434–450.
- Brennen, C. E., & Brennen, C. E. (2005). Fundamentals of multiphase flow.
- Byrd, R. H., & Nocedal, J. (1990). An analysis of reduced hessian methods for constrained optimization. *Mathematical Programming*, 49(1), 285–323.
- Caspari, A., Offermanns, C., Schäfer, P., Mhamdi, A., & Mitsos, A. (2019). A flexible air separation process: 2. optimal operation using economic model predictive control. *AIChE Journal*, 65(11), e16721.

- Chaffart, D., Rasoulilian, S., & Ricardez-Sandoval, L. A. (2016). Distributional uncertainty analysis and robust optimization in spatially heterogeneous multiscale process systems. *AIChE Journal*, 62(7), 2374–2390.
- Chaffart, D., & Ricardez-Sandoval, L. A. (2017). Robust dynamic optimization in heterogeneous multiscale catalytic flow reactors using polynomial chaos expansion. *Journal of Process Control*, 60, 128–140.
- Chaffart, D., & Ricardez-Sandoval, L. A. (2018). Robust optimization of a multiscale heterogeneous catalytic reactor system with spatially-varying uncertainty descriptions using polynomial chaos expansions. *The Canadian Journal of Chemical Engineering*, 96(1), 113–131.
- Chambers, R. G., & Quiggin, J. (2000). *Uncertainty, production, choice, and agency: The state-contingent approach*. Cambridge University Press.
- Chan, S. H. (1996). *Transport phenomena in combustion* (Vol. 1). Taylor & Francis.
- Charpentier, J., & McKenna, T. (2004). Managing complex systems: Some trends for the future of chemical and process engineering. *Chemical Engineering Science*, 59(8-9), 1617–1640.
- Chibante, R. (2010). *Simulated annealing: Theory with applications*. BoD–Books on Demand.
- Christofides, P. D., & Daoutidis, P. (1997). Finite-dimensional control of parabolic pde systems using approximate inertial manifolds. *Journal of mathematical analysis and applications*, 216(2), 398–420.
- Čižniar, M., Fikar, M., & Latifi, M. (2008). Design of constrained nonlinear model predictive control based on global optimisation. *Computer aided chemical engineering* (pp. 563–568). Elsevier.
- Constantin, P., & Foias, C. (1988). *Navier-stokes equations*. University of Chicago Press.
- Curtarolo, S., & Ceder, G. (2002). Dynamics of an inhomogeneously coarse grained multiscale system. *Physical Review Letters*, 88(25), 255504.

- Davis, E., & Ierapetritou, M. (2009). A kriging based method for the solution of mixed-integer nonlinear programs containing black-box functions. *Journal of Global Optimization*, 43(2), 191–205.
- Degachi, H., Chagra, W., & Ksouri, M. (2015). Global optimization method for model predictive control based on wiener model. *2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*, 1–6.
- Demirhan, C. D., Boukouvala, F., Kim, K., Song, H., Tso, W. W., Floudas, C. A., & Pistikopoulos, E. N. (2020). An integrated data-driven modeling & global optimization approach for multi-period nonlinear production planning problems. *Computers & Chemical Engineering*, 141, 107007.
- Di Pillo, G., & Grippo, L. (1989). Exact penalty functions in constrained optimization. *SIAM Journal on control and optimization*, 27(6), 1333–1360.
- Dittrich, W., & Reuter, M. (1994). *Classical and quantum dynamics*. Springer.
- Dolan, W., Cummings, P., & LeVan, M. (1989). Process optimization via simulated annealing: Application to network design. *AIChE Journal*, 35(5), 725–736.
- Doncevic, D. T., Schweidtmann, A. M., Vaupel, Y., Schäfer, P., Caspari, A., & Mitsos, A. (2020). Deterministic global nonlinear model predictive control with neural networks embedded. *IFAC-PapersOnLine*, 53(2), 5273–5278.
- Dunn, M. C., Shotorban, B., & Frendi, A. (2011). Uncertainty quantification of turbulence model coefficients via latin hypercube sampling method. *Journal of fluids engineering*, 133(4).
- Dussault, J.-P. (1995). Numerical stability and efficiency of penalty algorithms. *SIAM Journal on Numerical Analysis*, 32(1), 296–317.
- Edger, T., & Himmelblau, D. (2001). Optimization of chemical process.
- Eldred, M. (2009). Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design. *50th AIAA/ASME/ASCE/AHS/ASC*

- Esposito, W. R., & Floudas, C. A. (1998). Parameter estimation in nonlinear algebraic models via global optimization. *Computers & chemical engineering*, 22, S213–S220.
- Evans, M. R. (2012). Modelling ecological systems in a changing world. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 367(1586), 181–190.
- Florian, A. (1992). An efficient sampling scheme: Updated latin hypercube sampling. *Probabilistic engineering mechanics*, 7(2), 123–130.
- Floudas, C. A. (1995). *Nonlinear and mixed-integer optimization: Fundamentals and applications*. Oxford University Press.
- Floudas, C. A. (2013). *Deterministic global optimization: Theory, methods and applications* (Vol. 37). Springer Science & Business Media.
- Floudas, C. A., & Gounaris, C. E. (2009). A review of recent advances in global optimization. *Journal of Global Optimization*, 45(1), 3–38.
- Fluent, A. (2015). Ansys fluent. *Academic Research. Release*, 14.
- Furlonge, H., Pantelides, C., & Sørensen, E. (1999). Optimal operation of multivessel batch distillation columns. *AIChE Journal*, 45(4), 781–801.
- Garcia, C. E., Prett, D. M., & Morari, M. (1989). Model predictive control: Theory and practice—a survey. *Automatica*, 25(3), 335–348.
- Ghanem, R., Higdon, D., & Owhadi, H. (2017). *Handbook of uncertainty quantification* (Vol. 6). Springer.
- Giles, M. B. (2008). Multilevel monte carlo path simulation. *Operations research*, 56(3), 607–617.
- Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25), 2340–2361.
- Gordon, G., & Tibshirani, R. (2012). Karush-kuhn-tucker conditions. *Optimization*, 10(725/36), 725.

- Gould, N. I. M. (1989). On the convergence of a sequential penalty function method for constrained minimization. *SIAM Journal on Numerical Analysis*, 26(1), 107–128.
- Grimstad, B., & Andersson, H. (2019). Relu networks as surrogate models in mixed-integer linear programs. *Computers & Chemical Engineering*, 131, 106580.
- Hasan, M. F. (2018). An edge-concave underestimator for the global optimization of twice-differentiable nonconvex problems. *Journal of Global Optimization*, 71(4), 735–752.
- Hasmady, S., Hatakeyama, T., Wacker, M. P., Fushinobu, K., & Okazaki, K. (2009). Treatment of heterogeneous electrocatalysis in modeling transport-reaction phenomena in pefcs. *Thermal science and engineering*, 17(4), 147–156.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504–507.
- Holland, J. H. (1992). Genetic algorithms. *Scientific american*, 267(1), 66–73.
- Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5), 359–366.
- Horst, R., & Pardalos, P. M. (2013). *Handbook of global optimization* (Vol. 2). Springer Science & Business Media.
- Houska, B., & Chachuat, B. (2019). Global optimization in hilbert space. *Mathematical programming*, 173(1), 221–249.
- Icardi, M., Boccardo, G., & Tempone, R. (2016). On the predictivity of pore-scale simulations: Estimating uncertainties with multilevel monte carlo. *Advances in Water Resources*, 95, 46–60.
- Jasak, H., Jemcov, A., Tukovic, Z., et al. (2007). Openfoam: A c++ library for complex physics simulations. *International workshop on coupled methods in numerical dynamics*, 1000, 1–20.
- Jones, D. A., Margolin, L. G., & Titi, E. S. (1995). On the effectiveness of the approximate inertial manifold—a computational study. *Theoretical and Computational Fluid Dynamics*, 7(4), 243–260.

- Kamien, M. I., & Schwartz, N. L. (2012). *Dynamic optimization: The calculus of variations and optimal control in economics and management*. courier corporation.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95-international conference on neural networks*, 4, 1942–1948.
- Keßler, T., Kunde, C., Mertens, N., Michaels, D., & Kienle, A. (2019). Global optimization of distillation columns using surrogate models. *SN applied Sciences*, 1(1), 1–8.
- Ketcheson, D. I., Macdonald, C. B., & Gottlieb, S. (2009). Optimal implicit strong stability preserving runge–kutta methods. *Applied Numerical Mathematics*, 59(2), 373–392.
- Kevrekidis, I. G., Gear, C. W., & Hummer, G. (2004). Equation-free: The computer-aided analysis of complex multiscale systems. *AIChE Journal*, 50(7), 1346–1355.
- Khatibi, S., Cassol, G. O., & Dubljevic, S. (2021). Model predictive control of a non-isothermal axial dispersion tubular reactor with recycle. *Computers & Chemical Engineering*, 145, 107159.
- Kim, S. H., & Boukouvala, F. (2020). Surrogate-based optimization for mixed-integer non-linear problems. *Computers & Chemical Engineering*, 140, 106847.
- Kim, S. Y., & Bagajewicz, M. (2017). Global optimization of heat exchanger networks. part 2: Stages/substages superstructure with variable cp. *Industrial & Engineering Chemistry Research*, 56(20), 5958–5969.
- Kimaev, G., & Ricardez-Sandoval, L. A. (2018). Multilevel monte carlo for noise estimation in stochastic multiscale systems. *Chemical Engineering Research and Design*, 140, 33–43.
- Kleijn, C., Dorsman, R., Kuijlaars, K., Okkerse, M., & van Santen, H. v. (2007). Multi-scale modeling of chemical vapor deposition processes for thin film technology. *Journal of crystal growth*, 303(1), 362–380.
- Ko, D., Siriwardane, R., & Biegler, L. T. (2005). Optimization of pressure swing adsorption and fractionated vacuum pressure swing adsorption processes for co2 capture. *Industrial & engineering chemistry research*, 44(21), 8084–8094.

- Kramer, M. W. (1999). Motivation to reduce uncertainty: A reconceptualization of uncertainty reduction theory. *Management communication quarterly*, 13(2), 305–316.
- Krink, T., VesterstrOm, J. S., & Riget, J. (2002). Particle swarm optimisation with spatial particle extension. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, 2, 1474–1479.
- Lang, Y.-d., Malacina, A., Biegler, L. T., Munteanu, S., Madsen, J. I., & Zitney, S. E. (2009). Reduced order model based on principal component analysis for process simulation and optimization. *Energy & Fuels*, 23(3), 1695–1706.
- Lawler, E. L., & Wood, D. E. (1966). Branch-and-bound methods: A survey. *Operations research*, 14(4), 699–719.
- Lazinica, A. (2009). *Particle swarm optimization*. BoD–Books on Demand.
- Leimkuhler, B., & Matthews, C. (2016). *Molecular dynamics*. Springer.
- Leite, J., & Topping, B. (1999). Parallel simulated annealing for structural optimization. *Computers & Structures*, 73(1-5), 545–564.
- Levenspiel, O. (1999). *Chemical reaction engineering*. John Wiley & Sons.
- Lewis, F. L., Xie, L., & Popa, D. (2017). *Optimal and robust estimation: With an introduction to stochastic control theory*. CRC press.
- Li, D.-a., Hao, H., Ji, G., & Zhao, J. (2015). An adaptive clustering algorithm based on improved particle swarm optimisation in wireless sensor networks. *International Journal of High Performance Computing and Networking*, 8(4), 370–380.
- Lu, Q., Kumar, R., & Zavala, V. M. (2020). Mpc controller tuning using bayesian optimization techniques. *arXiv preprint arXiv:2009.14175*.
- Luna-Ortiz, E., & Theodoropoulos, C. (2005). An input/output model reduction-based optimization scheme for large-scale systems. *Multiscale Modeling & Simulation*, 4(2), 691–708.
- Ma, D. L., & Braatz, R. D. (2001). Worst-case analysis of finite-time control policies. *IEEE Transactions on Control Systems Technology*, 9(5), 766–774.

- Malik, M. R., Isaac, B. J., Coussement, A., Smith, P. J., & Parente, A. (2018). Principal component analysis coupled with nonlinear regression for chemistry reduction. *Combustion and Flame*, 187, 30–41.
- Marian, R. M., Luong, L. H., & Abhary, K. (2006). A genetic algorithm for the optimisation of assembly sequences. *Computers & Industrial Engineering*, 50(4), 503–527.
- Mathelin, L., Hussaini, M. Y., Zang, T. A., & Bataille, F. (2004). Uncertainty propagation for a turbulent, compressible nozzle flow using stochastic methods. *AIAA journal*, 42(8), 1669–1676.
- Mattheij, R. M., Rienstra, S. W., & Boonkkamp, J. T. T. (2005). *Partial differential equations: Modeling, analysis, computation*. SIAM.
- McCabe, W. L., Smith, J. C., & Harriott, P. (1993). *Unit operations of chemical engineering* (Vol. 5). McGraw-hill New York.
- McKeown, J., Meegan, D., & Sprevak, D. (1990). *An introduction to unconstrained optimization*. CRC Press.
- Meyer, C. A., Floudas, C. A., & Neumaier, A. (2002). Global optimization with nonfactorable constraints. *Industrial & engineering chemistry research*, 41(25), 6413–6424.
- Mirjalili, S. (2019). Particle swarm optimisation. *Evolutionary algorithms and neural networks* (pp. 15–31). Springer.
- Misener, R., & Floudas, C. A. (2014). Antigone: Algorithms for continuous/integer global optimization of nonlinear equations. *Journal of Global Optimization*, 59(2-3), 503–526.
- Mitsos, A., Chachuat, B., & Barton, P. I. (2009). McCormick-based relaxations of algorithms. *SIAM Journal on Optimization*, 20(2), 573–601.
- Mitten, L. (1970). Branch-and-bound methods: General formulation and properties. *Operations Research*, 18(1), 24–34.
- Morrison, D. D., Riley, J. D., & Zancanaro, J. F. (1962). Multiple shooting method for two-point boundary value problems. *Communications of the ACM*, 5(12), 613–614.

- Mühlenbein, H., Schomisch, M., & Born, J. (1991). The parallel genetic algorithm as function optimizer. *Parallel computing*, 17(6-7), 619–632.
- Nagy, Z. K., & Braatz, R. (2007). Distributional uncertainty analysis using power series and polynomial chaos expansions. *Journal of Process Control*, 17(3), 229–240.
- Najman, J., & Mitsos, A. (2019). Tighter mccormick relaxations through subgradient propagation. *Journal of Global Optimization*, 75(3), 565–593.
- Nocedal, J., & Wright, S. J. (2006). Sequential quadratic programming. *Numerical optimization*, 529–562.
- Noor, M., Kadirgama, K., & Rahman, M. (2011). Particle swarm optimisation prediction model for surface roughness. *International Journal of Physical Sciences*, 6(13), 3082–3090.
- Oduguwa, V., & Roy, R. (2002). Bi-level optimisation using genetic algorithm. *Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*, 322–327.
- Olufsen, M. S., & Nadim, A. (2003). On deriving lumped models for blood flow and pressure in the systemic arteries. *Computational fluid and solid mechanics 2003* (pp. 1786–1789). Elsevier.
- Petsagkourakis, P., Bonis, I., & Theodoropoulos, C. (2018). Reduced order optimization of large-scale nonlinear systems with nonlinear inequality constraints using steady state simulators. *Industrial & Engineering Chemistry Research*, 57(30), 9952–9963.
- Pires, J. C. M., Martins, F. G., Sousa, S., Alvim-Ferraz, M., & Pereira, M. (2008). Selection and validation of parameters in multiple linear and principal component regressions. *Environmental Modelling & Software*, 23(1), 50–55.
- Pontryagin, L. S. (1987). *Mathematical theory of optimal processes*. CRC press.
- Protter, P. E. (2005). Stochastic differential equations. *Stochastic integration and differential equations* (pp. 249–361). Springer.

- Puranik, Y., Kiliç, M., Sahinidis, N. V., Li, T., Gopalakrishnan, A., Besancon, B., & Roba, T. (2016). Global optimization of an industrial gas network operation. *AIChE Journal*, 62(9), 3215–3224.
- Rasoulilian, S., & Ricardez-Sandoval, L. A. (2014). Uncertainty analysis and robust optimization of multiscale process systems with application to epitaxial thin film growth. *Chemical Engineering Science*, 116, 590–600.
- Rawlings, J. B., Mayne, D. Q., & Diehl, M. (2017). *Model predictive control: Theory, computation, and design* (Vol. 2). Nob Hill Publishing Madison, WI.
- Reguera, D., Rubi, J., & Vilar, J. (2005). The mesoscopic dynamics of thermodynamic systems.
- Roberge, D. M., Ducry, L., Bieler, N., Cretton, P., & Zimmermann, B. (2005). Microreactor technology: A revolution for the fine chemical and pharmaceutical industries? *Chemical Engineering & Technology: Industrial Chemistry-Plant Equipment-Process Engineering-Biotechnology*, 28(3), 318–323.
- Rosner, D. E. (2012). *Transport processes in chemically reacting flow systems*. Courier Corporation.
- Ruuth, S. (2006). Global optimization of explicit strong-stability-preserving runge-kutta methods. *Mathematics of Computation*, 75(253), 183–207.
- Sahinidis, N. V. (1996). Baron: A general purpose global optimization software package. *Journal of global optimization*, 8(2), 201–205.
- Sales, B. C., Turner, J., & Maple, M. (1982). Oscillatory oxidation of co over pt, pd and ir catalysts: Theory. *Surface Science*, 114(2-3), 381–394.
- Schefflan, R. (2016). *Teach yourself the basics of aspen plus*. John Wiley & Sons.
- Schilders, W. H., Van der Vorst, H. A., & Rommes, J. (2008). *Model order reduction: Theory, research aspects and applications* (Vol. 13). Springer.
- Schwarm, A. T., & Nikolaou, M. (1999). Chance-constrained model predictive control. *AIChE Journal*, 45(8), 1743–1752.

- Schweidtmann, A. M., Bongartz, D., Grothe, D., Kerkenhoff, T., Lin, X., Najman, J., & Mitsos, A. (2021). Deterministic global optimization with gaussian processes embedded. *Mathematical Programming Computation*, 1–29.
- Schweidtmann, A. M., Bongartz, D., Huster, W. R., & Mitsos, A. (2019). Deterministic global process optimization: Flash calculations via artificial neural networks. *Computer aided chemical engineering* (pp. 937–942). Elsevier.
- Schweidtmann, A. M., & Mitsos, A. (2019). Deterministic global optimization with artificial neural networks embedded. *Journal of Optimization Theory and Applications*, 180(3), 925–948.
- Shroff, G. M., & Keller, H. B. (1993). Stabilization of unstable procedures: The recursive projection method. *SIAM Journal on numerical analysis*, 30(4), 1099–1120.
- Silva, A. P., Ravagnani, M. A., & Biscaia Jr, E. C. (2009). Particle swarm optimisation applied in retrofit of heat exchanger networks. *Computer aided chemical engineering* (pp. 1035–1040). Elsevier.
- Smith, J. D., Neto, A. A., Cremaschi, S., & Crunkleton, D. W. (2013). Cfd-based optimization of a flooded bed algae bioreactor. *Industrial & Engineering Chemistry Research*, 52(22), 7181–7188.
- Smith, R. (2005). *Chemical process: Design and integration*. John Wiley & Sons.
- Song, G., Cheng, L., Chao, Y., Yang, K., & Liu, Z. (2017). Emerging nanotechnology and advanced materials for cancer radiation therapy. *Advanced Materials*, 29(32), 1700996.
- Spall, J. C. (2005). *Introduction to stochastic search and optimization: Estimation, simulation, and control* (Vol. 65). John Wiley & Sons.
- Srivastava, R., Neuffer, W., Grano, D., Khan, S., Staudt, J., & Jozewicz, W. (2005). Controlling nox emission from industrial sources. *Environmental progress*, 24(2), 181–197.
- Sullivan, T. J. (2015). *Introduction to uncertainty quantification* (Vol. 63). Springer.
- Tawarmalani, M., & Sahinidis, N. V. (2005). A polyhedral branch-and-cut approach to global optimization. *Mathematical programming*, 103(2), 225–249.

- Teles, J. P., Castro, P. M., & Matos, H. A. (2013). Multi-parametric disaggregation technique for global optimization of polynomial programming problems. *Journal of Global Optimization*, 55(2), 227–251.
- Theodoropoulos, C. (2011). Optimisation and linear control of large scale nonlinear systems: A review and a suite of model reduction-based techniques. *Coping with complexity: Model reduction and data analysis*, 37–61.
- Theodoropoulos, C., & Luna-Ortiz, E. (2006). A reduced input/output dynamic optimisation method for macroscopic and microscopic systems. *Model reduction and coarse-graining approaches for multiscale phenomena* (pp. 535–560). Springer.
- Theodoropoulou, A., Adomaitis, R. A., & Zafiriou, E. (1998). Model reduction for optimization of rapid thermal chemical vapor deposition systems. *IEEE Transactions on Semiconductor Manufacturing*, 11(1), 85–98.
- Tsoo, C., Estrin, D. A., & Singer, S. J. (1990). Electronic energy shifts of a sodium atom in argon clusters by simulated annealing. *The Journal of chemical physics*, 93(10), 7187–7200.
- Van Laarhoven, P. J., & Aarts, E. H. (1987). Simulated annealing. *Simulated annealing: Theory and applications* (pp. 7–15). Springer.
- Vassiliadis, V. S., Sargent, R. W., & Pantelides, C. C. (1994). Solution of a class of multistage dynamic optimization problems. 1. problems without path constraints. *Industrial & Engineering Chemistry Research*, 33(9), 2111–2122.
- Vinter, R. B., & Vinter, R. (2010). *Optimal control*. Springer.
- Vose, M. D. (1999). *The simple genetic algorithm: Foundations and theory*. MIT press.
- Wächter, A., Biegler, L., Lang, Y., & Raghunathan, A. (2002). Ipopt: An interior point algorithm for large-scale nonlinear optimization.
- Wang, Q. (1997). Using genetic algorithms to optimise model parameters. *Environmental Modelling & Software*, 12(1), 27–34.

- Wang, Y.-N., Xu, Y.-Y., Xiang, H.-W., Li, Y.-W., & Zhang, B.-J. (2001). Modeling of catalyst pellets for fischer- tropesch synthesis. *Industrial & engineering chemistry research*, 40(20), 4324–4335.
- Wei, Z., Li, X., Xu, L., & Tan, C. (2012). Optimization of operating parameters for low no x emission in high-temperature air combustion. *Energy & Fuels*, 26(5), 2821–2829.
- Wiener, N. (1938). The homogeneous chaos. *American Journal of Mathematics*, 60(4), 897–936.
- Willcox, K., & Peraire, J. (2002). Balanced model reduction via the proper orthogonal decomposition. *AIAA journal*, 40(11), 2323–2330.
- Wolkenhauer, O., Auffray, C., Brass, O., Clairambault, J., Deutsch, A., Drasdo, D., Gervasio, F., Preziosi, L., Maini, P., Marciniak-Czochra, A., et al. (2014). Enabling multiscale modeling in systems medicine. *Genome medicine*, 6(3), 1–3.
- Wright, S. J. (1997). *Primal-dual interior-point methods*. SIAM.
- Xie, W., Bonis, I., & Theodoropoulos, C. (2015). Data-driven model reduction-based non-linear mpc for large-scale distributed parameter systems. *Journal of Process Control*, 35, 50–58.
- Xiong, Z., & Zhang, J. (2005). Neural network model-based on-line re-optimisation control of fed-batch processes using a modified iterative dynamic programming algorithm. *Chemical Engineering and Processing: Process Intensification*, 44(4), 477–484.
- Xiu, D. (2010). *Numerical methods for stochastic computations*. Princeton university press.
- Xiu, D., & Karniadakis, G. E. (2002). The wiener–askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2), 619–644.
- Xu, X., Liang, X., & Ren, J. (2007). Optimization of heat conduction using combinatorial optimization algorithms. *International journal of heat and mass transfer*, 50(9-10), 1675–1682.

- Zhang, Y., Ding, Y., & Christofides, P. D. (2019). Multiscale computational fluid dynamics modeling of thermal atomic layer deposition with application to chamber design. *Chemical Engineering Research and Design*, 147, 529–544.
- Zheng, H., Ricardez-Sandoval, L., & Budman, H. (2020). Robust estimation and economic predictive control for dynamic metabolic flux systems under probabilistic uncertainty. *Computers & Chemical Engineering*, 140, 106918.
- Zimmerman, W. B. (2006). *Multiphysics modeling with finite element methods* (Vol. 18). World Scientific Publishing Company.
- Zissis, G. (2002). Transport processes in multicomponent plasma. *Plasma Physics and Controlled Fusion*, 44(10), 2283.
- Zou, Y., & Kevrekidis, I. G. (2008). Uncertainty quantification for atomistic reaction models: An equation-free stochastic simulation algorithm example. *Multiscale Modeling & Simulation*, 6(4), 1217–1233.