Contents lists available at ScienceDirect

Chemical Engineering Research and Design

journal homepage: www.elsevier.com/locate/cherd

Model predictive control of phthalic anhydride synthesis in a fixed-bed catalytic reactor via machine learning modeling

Zhe Wu[a], *Anh Tran*[a], *Yi Ming Ren*[a], *Cory S. Barnes*[a], *Scarlett Chen*[a], *Panagiotis D. Christofides*[a,b,*]

[a] *Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA*

[b] *Department of Electrical and Computer Engineering, University of California, Los Angeles, CA, 90095-1592, USA*

ARTICLE INFO

Article history:
Received 22 December 2018
Received in revised form 7 February 2019
Accepted 11 February 2019
Available online 6 March 2019

Keywords:
Phthalic anhydride synthesis
Machine learning-based modeling
Model predictive control
CFD modeling
Recurrent neural network
Process control

ABSTRACT

This work proposes a general framework for linking a state-of-the-art computational fluid dynamics (CFD) solver, ANSYS Fluent, and other computing platforms using the lock synchronization mechanism in an effort to extend the utilities of CFD solvers from strictly modeling and design to also control and optimization applications. To demonstrate the effectiveness of the proposed approach, a challenging control problem in chemical engineering, i.e., maximizing the product yield and suppressing the hot-spot temperature in a fixed-bed reactor (FBR) with a highly exothermic reaction, is considered. Specifically, phthalic anhydride (PA) synthesis is chosen for this investigation because of its industrial significance and its extreme high exothermicity. Initially, a high-fidelity two-dimensional axisymmetric heterogeneous CFD model for an industrial-scale FBR is developed in ANSYS Fluent. Next, the CFD model is used to explore a wide operating regime of the FBR to create a database, from which recurrent neural network and ensemble learning techniques are used to derive a homogeneous ensemble regression model using a state-of-the-art application program interface (API), i.e., Keras. Then, a model predictive control (MPC) formulation that is designed to drive the process output to the desired set-point and suppress the magnitude of the hot-spot temperature to avoid catalyst deactivation is developed using the ensemble regression model. Subsequently, the CFD model, the ensemble regression model and the MPC are combined to create a closed-loop system by linking ANSYS Fluent to SciPy (a Python library used for scientific computing) via a message-passing interface (MPI) with lock synchronization mechanism. Finally, the simulation data generated by the closed-loop system are used to demonstrate the robustness and effectiveness of the proposed approach.

1. Introduction

Modeling, control and optimization of fixed-bed reactors (FBRs) has long been an active research area for both academia and industry as FBRs are building blocks of the petrochemical and refining industries. It is found that more than 90-95% of gas-solid catalytic reactors in existing production lines are FBRs. The prevalent application of FBRs in chemical process industries are because of their simplicity in design and operation compared with other reactor configurations, e.g., fluidized-bed reactors. However, despite the simple appearance, FBRs with highly exothermic reactions give rise to some of the most challenging control problems in chemical engineering owing to extreme nonlinearity, nonlinear spatially-distributed dynamics, moving hot spots, high risk of thermal runaway, constraints on manipulated

* Corresponding author.
E-mail address: pdc@seas.ucla.edu (P.D. Christofides).
https://doi.org/10.1016/j.cherd.2019.02.016
0263-8762/

inputs and state variables, and limited on-line measurements with high uncertainty and long delays (e.g., Wu and Huang, 2003). A standard framework to address the control aspect of FBRs to improve the process performance and to suppress the magnitude of the hot-spot temperature from a computational standpoint is well documented in literature. Specifically, it begins with the development of a detailed first-principles model for the FBR followed by the development of a corresponding computationally efficient reduced-order model, which is used in the development of a model-based control algorithm (Christofides, 2001). Subsequently, the first-principles model, the reduced-order model and the control algorithm are integrated to generate a closed-loop system based on which the performance of the algorithm is evaluated. For that reason, the first-principles model that accurately describes the dynamics of the physical FBR is essential to the development of control algorithms as well as the transition from a novel idea in academia to a cost-effective implementation in industry. The need for a computer-aided toolbox that can be used to expedite the development of high-fidelity first-principles models has been recognized by many and has led to the development of powerful computing platforms, i.e., computational fluid dynamic (CFD) solvers, such as ANSYS Fluent and OpenFoam, which allow process engineering researchers to overlook the nitty-gritty details of numerical methods for the solution of the first-principles model and to devote their complete attention to crafting more optimal reactor designs and developing algorithms to identify more cost-effective operating conditions. However, ANSYS Fluent and OpenFoam are developed to be stand-alone computing platforms, which means that the integration of CFD solvers with other computing platforms is often difficult, and they are yet to be standardized, as the source codes are often encrypted. Consequently, CFD solvers are often used solely for design purposes and not for real-time feedback control.

Recently, the community has made several efforts to expand the utilities of CFD solvers by studying the communication between CFD solvers and other computing platforms. For example, in Vaquerizo and Cocero (2018), a CFD solver was connected with Aspen Plus to obtain physical and chemical properties of compounds that do not exist in the ANSYS database. Additionally, in our previous works (Lao et al., 2016; Wu et al., 2017), an in-house optimization subroutine that solves a specific class of quadratic programming problems was developed via user-defined functions (UDF) such that ANSYS Fluent was able to compile the program of model predictive controller (MPC) and integrated it within the CFD model to form a closed-loop system. However, recognizing that MPC optimization problems may be non-convex and NP-hard for complex chemical reaction systems, a better solution is to integrate existing robust optimization solvers for solving large-scale nonlinear optimization problems with CFD solvers. For that reason, this work focuses on developing a general framework for linking a widely-used ANSYS Fluent CFD solver to various computing platforms using the lock synchronization mechanism that enables the development of a realistic closed-loop system in which a high-fidelity CFD model is used to represent the physical system.

Additionally, to derive a data-driven model to predict the nonlinear distributed dynamics of the FBR in MPC, we use recurrent neural networks and ensemble regression learning to derive a reduced-order model in our work due to an increasing interest in using machine learning methods to build prediction models across disciplines in academia among all existing data-driven modeling techniques (e.g., subspace identification, sparse proper orthogonal decomposition (Sidhu et al., 2018, 2018b) and machine learning methods). Machine learning can also be used in conjunction with other nonlinear control methods (e.g., universal Sontag's control law and feedback linearization), where a process model is needed and can be approximated by other data-driven models such as machine learning-based model. Therefore, in this work, we demonstrate the value and potential of machine learning methods in chemical engineering via an industrially-relevant example. Furthermore, we want to contribute to the common effort to expand to utilities of CFD solvers beyond design purposes by proposing a general integration framework for MPC and dynamic CFD simulation via message passage interface (MPI) to create closed-loop systems (i.e., MPC within CFD dynamic simulations) for the purpose of real-time control. Specifically, different from the MPI used to facilitate the communication between the host process and many computing process units (CPUs) in ANSYS Fluent parallel solver, the MPI we develop in the present work represents the communication between ANSYS Fluent CFD solver and stand-alone robust optimization subroutines that enables the development of a realistic closed-loop system in which a high-fidelity CFD model is used to represent the physical system.

The remainder of this manuscript is structured as follows: Section 2 provides a high-level description of a typical industrial-scale exothermic FBR used in the phthalic anhydride (PA) synthesis production (Orozco et al., 2010; Sarosh et al., 2018; Mülheims and Kraushaar-Czarnetzki, 2011), and Section 3 details the development of the heterogeneous CFD model for the FBR. Next, Section 4 discusses the model validation of the heterogeneous CFD model for the FBR and the study of the open-loop dynamics to map the sensitivity of the fluid temperature and composition inside the FBR to the jacket temperature profile. Section 5 presents the development of a homogeneous ensemble regression using the state-of-the-art application program interface (API), i.e., Keras (Chollet et al., 2015), that describes the nonlinear distributed dynamics of the FBR at several discrete locations is introduced. Then, Section 6.1 outlines the optimization problem of a model predictive controller (MPC) designed to drive the process output to the desired set-point and suppress the magnitude of the hot-spot temperature to avoid catalyst deactivation, and Section 6.2 proposes a general framework for linking ANSYS Fluent to *SciPy* library (Jones et al., 2001) in Python via a message-passing interface (MPI) embedded in UDF with lock synchronization mechanism so that the CFD model, the reduced-order model and the MPC can be integrated to create a closed-loop system. Finally, Section 7 provides the rigorous validation of the CFD model for the FBR, the assessment of the ensemble regression and the performance of the closed-loop system.

2. Chemical reactor description

Phthalic anhydride (PA) synthesis is chosen for this investigation since PA is one of the most important intermediates for a variety of polymer products such as polyesters, resins and plasticizers and is a key pharmaceutical ingredient of cellulose acetate phthalate, in addition to the extreme high exothermicity of its production reaction making reactor operation challenging. Specifically, PA is commonly produced by partial oxidation of o-xylene in great excess of air over V_2O_5/TiO_2 catalysts housed in a

multi-fixed-bed reactor, inside which the fixed-bed reactors (FBRs) are submerged in molten salt cooling jacket. The reactor design and the operating parameters of the production of PA are optimized to suppress the magnitude of the hot-spot temperatures to prevent explosion, catalyst deactivation and thermal runaway. Despite the off-line design efforts, the formation of hot spots in the first section of the FBRs is a known problem due to insufficient cooling, and therefore, the need of high-fidelity models that accurately simulate transient and steady-state response of the reactor is evident. This investigation outlines the modeling (Sections 3 and 5) and control (Section 6.1 and 6.2) of a FBR inside an industrial-scale fixed-bed reactor.

In the existing production line of PA, the FBR is typically 4 m in length and 2.5 cm in diameter and is packed with V_2O_5/TiO_2 catalyst particles. Additionally, the FBR in this study is equipped with an outer cooling jacket, which is divided into four noninteracting discrete zones along the FBR, each of which jacket temperature is independently regulated by a centralized controller as shown in Fig. 1. Specifically, the FBR is fed with a mixture of air and o-xylene that undergoes a highly exothermic conversion to produce PA leading to the formation of hot spots, which are regulated by the centralized controller to prevent catalyst deactivation and thermal runaway.

Previous studies (Orozco et al., 2010; Sarosh et al., 2018; Mülheims and Kraushaar-Czarnetzki, 2011) have used computational fluid dynamics (CFD) tools and first-principles modeling to create high-fidelity model for the FBR of this type to study the sensitivity of the hot-spot temperature and product yield to operating parameters such as the concentration of o-xylene in the feed, the feed flow rate and the jacket temperature profile. These studies have found that it is reasonable to assume that the FBR is symmetrical in the azimuthal direction, which allows for the generation of a computationally affordable two-dimensional (2D) axisymmetric in space heterogeneous CFD model with the same output of meaningful data as a computationally expensive 3D model. In remainder of this section, the development of a 2D axisymmetric heterogeneous CFD model of the FBR via ANSYS Fluent will be outlined.

3. Computational fluid dynamics modeling

3.1. Mesh generation

The mesh quality plays an important role in CFD modeling, where an accurate CFD model with high mesh quality can significantly save computational resources and obtain a converged solution with robustness. Due to the axisymmetric geometry property of the fixed-bed catalytic reactor, a two-dimensional (2D) axisymmetric structured mesh is constructed in ANSYS ICEM software as shown in Fig. 1. Specifically, the mesh discretizes the reactor volume into 85211 control volumes, within each of which the first principles model (as shown in Eq. (14)) of the FBR is numerically solved via second-order upwind finite volume methods. Additionally, high mesh density is applied near the boundary of the reactor to capture large spatial gradients in that region.With the high-quality structured mesh, evaluated based on the minimum orthogonal factor of 1 and maximum ortho skew of 0, the CFD simulations do not encounter convergence difficulty, and numerical solutions generated from the CFD model carry small numerical error (Tran et al., 2018). Additionally, a mesh independent study demonstrated that further increasing the mesh size does not improve the simulation results of CFD models but leads to higher computation time to convergence.

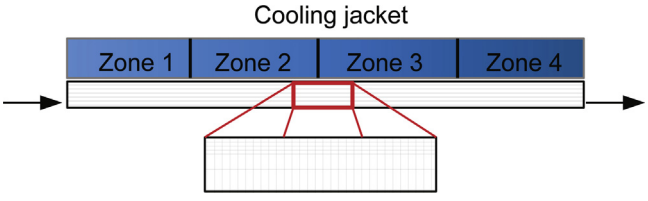


Fig. 1 – Two-dimensional axisymmetric reactor geometry with outer cooling jacket (blue) divided into four noninteracting discrete zones along the tube.

3.2. 2D axisymmetric heterogeneous CFD model

In this section, we outline the modeling strategy to customize ANSYS Fluent to create the 2D axisymmetric heterogeneous CFD model for the FBR, and we begin with the bed porosity profile. It has been found in previous experiments (De Klerk, 2003) that a bed porosity profile exhibits an oscillatory pattern with a decreasing amplitude as the distance to the reactor wall increases due to different packing structure induced by the wall effect; and this pattern is especially pronounced in FBRs with small tube-to-pellet ratio $N<10$. Therefore, in an effort to obtain an accurate simulation of the flow field profile in the FBR, the radial variation of the catalytic bed porosity must be accounted for, and in this CFD model, the bed porosity profile in the radial direction, encoded in a user-defined function (UDF), is modeled as follows (De Klerk, 2003):

$$\gamma(r) = 2.14r^2 - 2.53r + 1 \quad \text{when} \quad r \le 0.637 \tag{1}$$

$$= \gamma_b + 0.29\exp(-0.6r)[\cos(2.3\pi(r-0.16))] + 0.15\exp(-0.9r) \quad \text{when} \quad r > 0.637 \tag{2}$$

where $\gamma(r)$ and γ_b are the bed porosity at a particular position inside the bed and the bed porosity in the absence of wall effect, respectively, and r is the dimensionless distance from the wall and is computed as follows:

$$r = \frac{R-y}{d_p} \tag{3}$$

where R (m), y (m) and d_p (m) represent reactor radius, radial position relative to the reactor axis, and particle diameter, respectively. Specifically, at regions close to the reactor wall ($r \le 0.637$), in which the catalyst packing pattern is more structured, the bed porosity profile is modeled by Eq. (1). While at regions further from the wall ($r > 0.637$), in which the catalyst packing pattern is more random, the bed porosity profile with the oscillatory pattern is modeled by Eq. (2). By encoding an accurate description of the bed porosity profile in the CFD model, we have made the first step toward building a model that can simulate a representative flow field profile in the FBR.

In addition, when the reactants flow into the FBR with a nonuniformly packed catalytic bed and a low average porosity of 0.4, the flow field profile is expected to deviate from the plug flow profile and to have a complex pattern due to coupling effects with other transport phenomena, the flow superficial velocity is expected to decrease, and the pressure drop across the catalytic bed is expected to be significant. To address this

problem, ANSYS Fluent porous zone function is employed, which automatically generates and adds additional momentum source terms into the Navier–Stokes equation, to simulate the effects of the catalytic bed on the flow field profile, and its parameters, i.e., the permeability (α_b) and the inertial loss coefficient (C_2), encoded in a UDF are computed as follows:

$$\alpha_b = \frac{d_p}{150}\frac{(\gamma(r))^3}{(1-\gamma(r))^2} \tag{4}$$

$$C_2 = \frac{3.5}{d_p}\frac{1-\gamma(r)}{(\gamma(r))^3}, \tag{5}$$

and the Navier–Stokes equation can be written as shown in Eq. (14b) below.

In this study, the intrinsic partial oxidation rate of o-xylene to PA is modeled with a global heterogeneous kinetic model (Calderbank et al., 1977; Anastasov, 2003), in which the reaction rate expressions can be written as follows:

$$K_n = K_{0n}\exp\left(\frac{-E_n}{R_g T_s}\right) \quad \text{where} \quad n = 1, 2, \cdots, 5 \tag{6a}$$

$$r_1 = K_1 \alpha P_A^s \tag{6b}$$

$$r_2 = K_2 \alpha P_B^s \tag{6c}$$

$$r_3 = K_3 \alpha P_A^s \tag{6d}$$

$$r_4 = K_4 \alpha P_A^s \tag{6e}$$

$$r_5 = K_5 \alpha P_C^s \tag{6f}$$

$$\alpha = \frac{K_C P_{O_2}^s}{K_C P_{O_2}^s + (K_1 + 6.5K_3 + 3K_4)P_A^s + K_2 P_B^s + K_5 P_C^s} \tag{6g}$$

where A, B, C and D represent o-XL, o-tolualdehyde (o-TA), phthalide (PH), and PA, respectively, E_n (kJ kmol^{-1}), K_{0n} (kmol m^2 kg^{-1} s^{-1} N^{-1}), K_n (kmol m^2 kg^{-1} s^{-1} N^{-1}) and r_n (kmol kg^{-1} s^{-1}) are the activation energy, the pre-exponentional factor, the rate constant and the intrinsic reaction rate of the nth reaction, respectively, α is the fraction of unoccupied oxidized sites, P_i^s (kPa) is the partial pressure of species i on the catalyst (solid) surface, T_s (K) is the catalyst temperature and K_C (kmol m^2 kg^{-1} s^{-1} N^{-1}) is the catalyst reoxidation rate constant. In this kinetic model, r_6 is set to zero because within the operating regime optimized for the yield of PA, the oxidation rate of PA to CO_x has been found to be insignificant (Anastasov, 2003). The choice to integrate this kinetic model into the CFD model for the FBR is supported by experimental data reported in previous investigations. Specifically, Anastasov (2003) shows that when the first-principles model for the FBR with the same kinetic model is given boundary conditions at a similar operating regime to that considered in the present study, it generates simulation data that are in close agreement with the experimental data. Furthermore, it is important to note that the partial oxidation of o-xylene does not take place spontaneously as the reactants enter the FBR. In fact, the reactants must be transported from the bulk flow to the catalyst surface, and subsequently, diffuse into the catalyst pores and bind to catalyst active sites at which the partial oxidation of o-xylene occurs. Therefore, the effectiveness factor (η) of 0.34 is introduced to account for the effects of transport resistances between the free flowing fluid and the catalyst surface, and the observed reaction rates can be rewritten as follows:

$$r_n^o = \eta r_n(P_i^f), \quad n = 1, \cdots, 5. \tag{7}$$

In addition, it is equally important to recognize that the production of PA only occurs in the catalyst particles, and the reaction rates (Eq. (6)) are expressed in terms of the rate of change per unit mass of catalyst; therefore, prior to be integrated into the material and energy conservation equations of the FBR, the reaction rates are converted into the rate of change per unit volume of the FBR, i.e., kmol m^{-3} s^{-1}, as follows:

$$r_n^o = \eta\rho_s(1-\gamma(r))\, r_n(P_i^f). \tag{8}$$

where ρ_s (kg m^{-3}) is the density of the catalyst particles. Subsequently, the kinetic model is used to develop the source term ($S_{Y,i}$) for the material conservation equation for the species i as follows:

$$S_{Y,i} = \frac{M_i}{\gamma(r)}\sum_{n=1}^{5}\delta_{ni} r_n^o \tag{9}$$

where M_i (kg kmol^{-1}) is the molecular weight of the species i and δ_{ni} is the stoichiometric coefficient of species i in the reaction n. It is important to note that when a source term is introduced a CFD model in which ANSYS Fluent porous function is utilized, the source term is automatically multiplied by the bed porosity and is subsequently integrated in the corresponding governing equation as shown in Eqs. (14e, 14c) and (14d); therefore, a correction factor, $1/\gamma(r)$, must be used in the formulation of all source terms developed in this work as shown in Eq. (9).

To develop the heterogeneous CFD model for the FBR, we need to explicitly develop the first-principle model for the solid phase. In this work, we assume that the partial oxidation of o-xylene is a transport limited process, and the catalyst particles are isothermal. As a result, the first-principles model for the solid phase is reduced to the energy conservation equation shown below:

$$\frac{\partial}{\partial t}\left((1-\gamma(r))\rho_s C_{P,s} T_s\right) = \sum_{i=1}^{5} r_i^o \Delta H_i + \epsilon(1-\gamma(r))h_{fs}(T_f - T_s) \tag{10}$$

where $C_{P,s} = 836.0$ (J kg^{-1} K^{-1}) and $\epsilon = 540.0$ (m^2 m^{-3}) are the specific constant pressure heat capacity and the external surface per unit volume of catalyst particles, T_f (K) is the bulk temperature, h_{fs} (W m^{-2} K^{-1}) is the overall heat transfer coefficient between the bulk flow and the catalyst surface and H_i (J kmol^{-1}) is the enthalpy of the ith reaction. Specifically, the right hand side of Eq. (10) accounts for the rate of change in the solid (catalyst) temperature due to the partial oxidation of o-xylene and the heat transfer between the free flowing fluid and the catalyst particles, respectively. Next, to integrate the energy conservation equation for the solid phase into the CFD model, the solid temperature is defined as a user-defined scalar (UDS). Upon this, ANSYS Fluent automatically generates a generic convective-diffusive transport equation for this UDS in a porous media of the following form:

$$\frac{\partial(\gamma\rho\phi)}{\partial t} + \nabla\cdot(\gamma\rho\vec{v}\phi) = \nabla\cdot(\gamma\Gamma\nabla\phi) + \gamma S_\phi \tag{11}$$

where ϕ is a UDS, Γ is a diffusion coefficient, and S_ϕ is the source term, so that our task is to make use of UDFs to transform Eq. (11) into Eq. (10). Specifically, the convective and diffusive terms are set to zero, and two additional source terms accounting for the two mechanisms that change the solid temperature are formulated as follows:

$$S_{T_s,1} = \frac{1}{\gamma(r)} \sum_{i=1}^{5} r_i^o \Delta H_i \tag{12}$$

$$S_{T_s,2} = \frac{\epsilon(1-\gamma(r))h_{fs}(T_f - T_s)}{\gamma(r)}, \tag{13}$$

and the transient term in Eq. (11) is replaced by that of Eq. (10).

As a result of the above analysis, the 2D heterogeneous CFD model used to describe the behavior of a catalytic process taking place in the FBR, i.e., the continuity, momentum, energy and species material balances, are given by the following equations:

$$\frac{\partial}{\partial t}(\gamma\rho_f) + \nabla\cdot(\gamma\rho_f\vec{v}) = 0 \tag{14a}$$

$$\frac{\partial}{\partial t}(\gamma\rho_f\vec{v}) + \nabla\cdot(\gamma\rho_f\vec{v}\vec{v}) = -\gamma\nabla P + \nabla\cdot(\gamma\vec{\tau}) - \gamma\left(\frac{\gamma\mu}{\alpha_b}\vec{v} + \frac{\gamma^2 C_2}{2}\rho_f|\vec{v}|\vec{v}\right) \tag{14b}$$

$$\frac{\partial}{\partial t}(\gamma\rho_f C_{P,f} T_f) + \nabla\cdot(\gamma\vec{v}(\rho_f C_{P,f} T_f + P)) = \nabla\cdot\left[\gamma k_f \nabla T - \left(\sum_i h_i\vec{J}_i\right) + \vec{\tau}\cdot\vec{v}\right] - \gamma S_{T_s,2} \tag{14c}$$

$$\frac{\partial}{\partial t}((1-\gamma)\rho_s C_{P,s} T_s) = \gamma S_{T_s,1} + \gamma S_{T_s,2} \tag{14d}$$

$$\frac{\partial}{\partial t}(\gamma\rho_f Y_i) + \nabla\cdot(\gamma\rho_f\vec{v}Y_i) = -\nabla\cdot(\gamma\vec{J}_i) + \gamma S_{Y,i} \tag{14e}$$

with

$$\vec{\tau} = \mu\left[\left(\nabla\vec{v} + \nabla\vec{v}^T\right) - \frac{2}{3}\nabla\cdot\vec{v}I\right] \tag{14f}$$

$$\vec{J}_i = \rho_f D_{m,i}\nabla Y_i - D_{T,i}\frac{\nabla T}{T} \tag{14g}$$

$$Nu = 2 + 1.1Pr^{1/3}Re^{0.6} \tag{14h}$$

$$h_{fs} = \frac{k_f Nu}{d_p} \tag{14i}$$

boundary conditions :

$$z = 0; Y_i = Y_{i,0}, T_f = T_{f,0} \tag{14j}$$

$$r = 0; \frac{\partial\gamma\rho_f Y_i}{\partial r} = 0; \frac{\partial T_f}{\partial r} = 0 \tag{14k}$$

$$r = R; \frac{\partial\gamma\rho_f Y_i}{\partial r} = 0; -\lambda_r\frac{\partial T_f}{\partial r} = h_{fj}(T_f - T_j) \tag{14l}$$

where $\vec{v}$ (m s^{-1}), P (kPa), ρ_f (kg m^{-3}), $C_{P,f}$ (J kg^{-1} K^{-1}), k_f (W m^{-1} K^{-1}) and μ (m^2 s^{-1}) are the velocity, static pressure, specific constant pressure, density, heat capacity, thermal conductivity and molecular viscosity of the bulk flow, respectively, $\nabla\left(\sum_i h_i\vec{J}_i\right)$ represents the effect of enthalpy transport due to species diffusion, Y_i, $\vec{J}_i$ (kg m^{-2} s^{-1}), $D_{m,i}$ (m^2 s^{-1}) and $D_{T,i}$ (m^2 s^{-1} K^{-1}) are the mass fraction, mass diffusion flux, mass diffusion coefficient and thermal diffusion coefficient, respectively,

Table 1 – Operating parameters of the FBR reported in Anastasov (2003), in which $V_{f,0}$ (m^3 h^{-1}, STP), $T_{f,0}$ (K) and $C_{A,0}$ (g m^{-3}, STP) are the volumetric flow rate, temperature and concentration of o-xylene in the feed and T_{ji} (K) is the jacket temperature profile.

	Operating parameters
$V_{f,0}$	3.13
$T_{f,0}$	640
$C_{A,0}$	43.1
T_{ji}, i = 1, 2, 3, 4	640

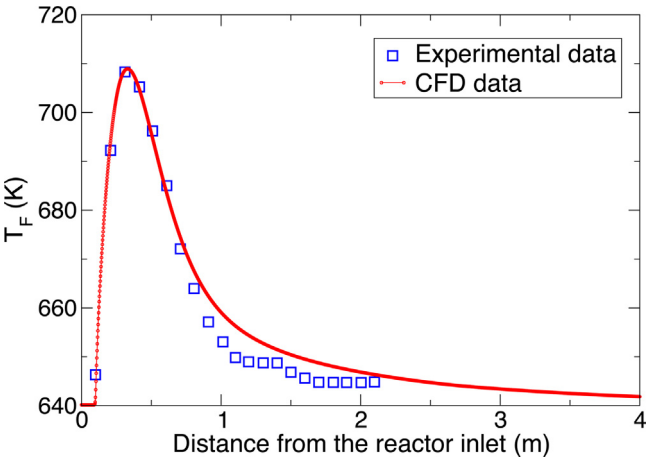


Fig. 2 – Comparison of the fluid temperature predicted by the 2D axisymmetric heterogeneous CFD model to the experimental data collected at the same operating parameters as listed in Table 1 and reported in Anastasov (2003).

T_j is the jacket temperature profile, h_{fj} the lumped heat transfer coefficient between the bulk flow and the cooling jacket, $\vec{\tau}$ and I are the stress tensor and unit tensor, respectively, Re, Pr and Nu are the Reynolds number, Prandtl number and Nusselt number, respectively.

4. Model validation and open-loop dynamics

In this section, the 2D axisymmetric heterogeneous CFD model for the FBR will be validated using experimental data reported in the literature. This section details the validation of the 2D axisymmetric heterogeneous CFD model for the FBR, in which the key process parameters, i.e., the hot-spot temperature and location, predicted by the CFD model are compared to those in the literature. To achieve a comprehensive and transparent analysis, the operating parameters reported in Anastasov (2003) as shown in Table 1 are used to develop the boundary conditions for the CFD model. Then, the simulation data predicted by the CFD model for the FBR are compared to experimental data reported in Anastasov (2003). Specifically, Fig. 2 shows that the CFD model predicts the steady-state temperature profile consistent to experimental data. In additional, the CFD model correctly identifies the hot-spot temperature and location, which are the two critical parameters that determine the usefulness of the CFD model in preventing catalyst deactivation and thermal runaway from a computational standpoint, with an absolute mean error of 0.64 K and 0.018 m, respectively. In addition, Fig. 3 verifies that our proposed modeling strategy for a heterogeneous CFD model, and especially the integration of the first-principles model for the solid

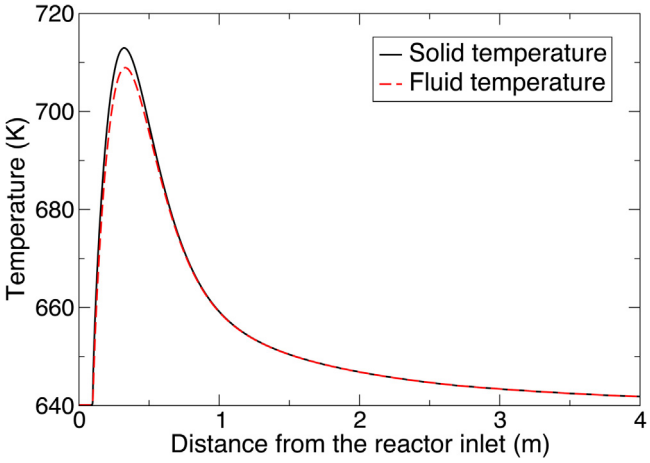


Fig. 3 – Comparison between the fluid temperature and solid temperature predicted by the 2D axisymmetric heterogeneous CFD model at the operating parameters as listed in Table 1.

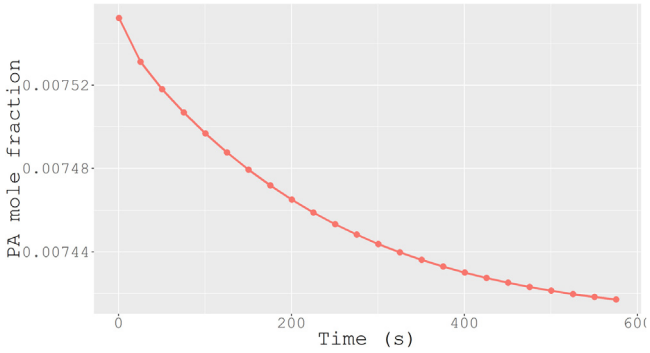


Fig. 4 – The trajectory of the radially-averaged PA concentration (denoted by x_D^{outlet}) at the reactor outlet for open-loop control with a desired set-point for PA concentration at the outlet $x_D^{set} = 7.420 \times 10^{-3}$ and pre-determined jacket temperatures $T_{j1} = 620$ K, $T_{j2} = 620$ K, $T_{j3} = 620$ K and $T_{j4} = 620$ K.

phase into the CFD model via UDS, is appropriate because the CFD model correctly simulates the exothermicity and heterogeneous reaction characteristics of the partial oxidation of o-xylene to PA inside the FBR. The above analysis indicates that the CFD model can be considered as a realistic representation for the FBR, and hence, it will be used to explore the space of operating parameters to identify a more cost-effective and safe operating regime. Specifically, the jacket temperatures of the four discrete noninteracting zones (which are chosen as the manipulated inputs of the centralized controller discussed in Section 6) are systematically adjusted independently within the range of ±20 K and a step change of 5 K from the nominal value of 640 K to sweep the space of the operating parameters. Once a new jacket temperature profile is proposed, it is applied to the CFD model that is originally at the steady-state corresponding to the operating parameters listed in Table 1 as a new boundary condition in a step-change input fashion. Then, the CFD model with the updated jacket temperature profile is integrated forward in time with the integration time step of 0.1 s until reaching the corresponding steady-state as shown in Figs. 4 and 5 . In this investigation, more than 4000 open-loop simulations are used to map the sensitivity of the fluid temperature and composition inside the FBR to the jacket temperature profile, and the simulation data are used to build

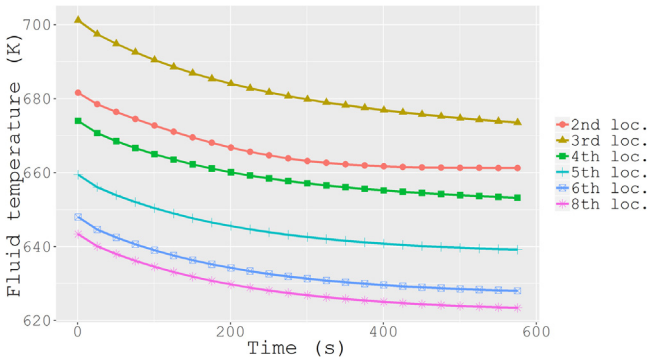


Fig. 5 – The trajectory of the radially-averaged fluid temperature (T_f) at a few selected tracking locations for open-loop control with $x_D^{set} = 7.420 \times 10^{-3}$ and pre-determined jacket temperatures $T_{j1} = 620$ K, $T_{j2} = 620$ K, $T_{j3} = 620$ K and $T_{j4} = 620$ K.

Table 2 – Tracking locations along the FBR such that at each tracking location, the fluid radially-averaged composition and temperature are recorded at the end of every CFD iteration time step.

Index	Distance from the reactor inlet (m)
1	0.1
2	0.2
3	0.4
4	0.7
5	1.0
6	2.0
7	3.0
8	4.0

the database for the purpose of training a reduced-order model that describes the nonlinear distributed dynamics of the FBR as a function of the jacket temperature profile. In detail, we setup eight tracking locations along the FBR as shown in Table 2 such that at each tracking location, information about the fluid radially-averaged composition and temperature is recorded at the end of every five CFD iteration time steps. It is noted that the first five are strategically placed in the first meter of the FBR from the reactor inlet to maximize our chance to capture the exact hot-spot temperature and location. Subsequently, the time-series database from open-loop simulations are used to create a large number of sequences, each of which contains descriptions about the distributed dynamics of the FBR over 50 CFD integration time steps, via a sliding window method. Additionally, data preprocessing is employed to remove data samples with repeated or similar dynamic behavior to avoid over-fitting those data during the training process.

5. Ensemble regression modeling

In this study, a recurrent neural network (RNN) learning algorithm (Lipton et al., 2015), an ensemble learning technique and a 10-fold cross validation are used to construct a homogeneous ensemble regression that uses the feed operating parameters, the jacket temperature profile and the real-time state measurements (i.e., the fluid temperature and composition) at eight discrete locations to predict the nonlinear distributed dynamics of the FBR. The motivation for using the ensemble regression is four-fold. First, the RNN learning algorithm, which is also known as the backpropagation through time (BPTT) algorithm, can easily derive RNNs that

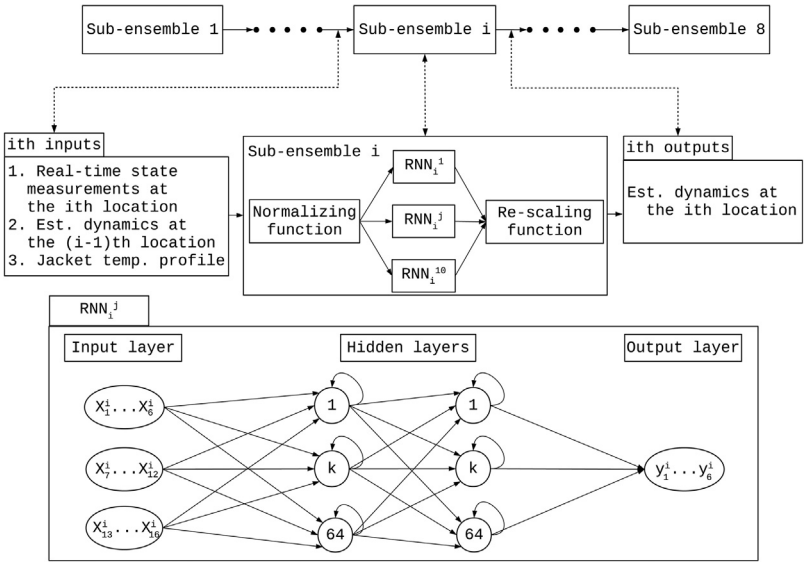


Fig. 6 – The schematic of ensemble learning with RNN models.

memorize the training data well, however, without the generalization capability, and therefore, by exposing the learning algorithm to different subsets of the training data, generated by 10-fold cross validation, effectively prevents it from overfitting the RNNs. Second, the learning algorithm is known to be non-convex and is an infamous example of an NP-hard problem, and hence, by using different starting initial weight matrices, the learning algorithm might be able to avoid getting trapped in a local minimum, and arrive at the set of weights that allows the RNN to accurately approximate the latent function that transforms input sequences to corresponding output sequences. Third, an ensemble regression is known to often outperform its individual constituents and simultaneously accounts for uncertainty in model selection (Mendes-Moreira et al., 2012). Fourth, the proposed approach creates a unique opportunity for us to demonstrate our general framework for linking ANSYS Fluent to several powerful API toolboxes that are typically incompatible. For those reasons, the ensemble regression is structured to consist of eight sub-ensembles corresponding to 8 tracking locations as shown in Table 2, and each of which is a 10 fold cross-validated committee of RNNs derived to predict the nonlinear distributed dynamics of the FBR as depicted in Fig. 6.

In this study, each sub-ensemble of the ensemble regression is designed to predict the trajectories of the state variables at a fixed location over a time interval equal to the sampling rate of the centralized controller, which can be an integer multiple of the CFD integration time step (in this work it is set to be 50 CFD integration time steps, which corresponds to a sequence of 10 consecutive state measurements). Specifically, as shown in Fig. 6, the sub-ensemble i uses the on-line measurements of the state variables at the designated location along the FBR (denoted by X_k^i, $k=1,\ldots,6$), the estimated trajectories of the state variables at an upstream location over the elapsed time (denoted by X_k^i, $k=7,\ldots,12$) and the jacket temperature profile (denoted by X_k^i, $k=13,\ldots,16$) to calculate its predictions, which are the average trajectories of those generated by its constituent RNNs. With this proposed design of the input, output and ensemble integration strategy of the sub-ensembles, the mathematical formula of a sub-ensemble i can be expressed as follows:

$$f_j^i\left(X_k^i\right) : \mathbf{R}^{10\times 6} \times \mathbf{R}^{10\times 6} \times \mathbf{R}^{10\times 4} \rightarrow \mathbf{R}^{10\times 6} \tag{15}$$

$$f^i(X_k^i) = \frac{1}{N_i}\sum_{j=1}^{N_i} f_j^i(X_k^i) \tag{16}$$

where $f_j^i(X_k^i)$ represents a j RNN in the sub-ensemble i. This ensemble integration strategy appears to be rudimentary; nonetheless, the performance of the sub-ensemble measured in terms of the absolute mean error is statistically better than that of its constituent RNNs.

The constituent RNNs of individual sub-ensembles are known as the 10-fold cross-validated committee; this is because that 10-fold cross validation is used to generate 10 different subsets of the training data, and each of which is subsequently used to train an RNN. Although the state-of-the-art API, Keras, can be readily used to create the 10-fold cross-validated committee, training RNNs remains a nontrivial task because the learning algorithm is known to suffer from the vanishing and exploding gradient problems. For that reason, the RNNs is designed to have two hidden recurrent layers consisting of 64 recurrent units that use the rectified linear unit function as the activation function (i.e., ReLu function, $g(x)=\max\{0, x\}$) and are initialized as proposed in Le et al. (2015). Specifically, Le et al. (2015) demonstrates that this initialization in which the recurrent and bias weight matrices are set to the identity matrix and zero, respectively, coupled with the use of the ReLu function allows the trained RNNs to have the consistently comparable performance with the standard long short term memory (LSTM) networks in various applications.

As a last note of this Section, we would like to detail the procedure encoded in the ensemble regression which is used to predict the nonlinear distributed dynamics of the FBR over a finite time interval in the future. Specifically, the sub-ensemble i normalizes the input signals, i.e., the feed

operating parameters, the real-time state measurements at the first location and the jacket temperature profile, using the statistics of the first training set and, subsequently, transmits the normalized input signals to its cross-validated committee of RNNs. It is critical to note that all input and output signals of individual RNNs are normalized based solely on the statistics of its training set to prevent data leakage from its test set. Next, the constituent RNNs simultaneously compute the trajectories of the normalized state variables over a finite time interval in the future, which are pooled to compute the respective average trajectories. Then, the first sub-ensemble generates its output signals by rescaling the average trajectories of the normalized state variables using the corresponding statistics. The output signals of the first sub-ensemble are used to infer the dynamics of the FBR at the corresponding location and as input signals to the second sub-ensemble in addition to the real-time state measurements at the second location and the jacket temperature profile as depicted in Fig. 6. All downstream sub-ensembles execute the same procedure, and the outputs of the sub-ensembles are used to estimate the nonlinear distributed dynamics of the FBR. As a result, the output of the ensemble regression model has a dimension of $8 \times 10 \times 6$, which corresponds to eight constituent sub-ensembles each of which predicts the trajectories of six state variables over a time interval of 10 sampling points.

6. Feedback control design

6.1. *Model predictive controller using ensemble regression model*

In this section, the ensemble regression model, derived to estimate the nonlinear distributed dynamics of the FBR over a time interval equal to 50 CFD integration time steps based on the real-time measurements of the state variables, the feed operating parameters and the cooling jacket temperature profile, is used to develop a model predictive controller (MPC). Specifically, the MPC with a sampling period equal to the prediction period of the ensemble regression model is formulated as an optimization problem to drive the concentration of PA at the reactor outlet to a desired set-point and to maintain the fluid temperature below a predetermined maximum allowable value at all times to prevent catalyst deactivation by changing the jacket temperature profile accounting for the constraints on it. For the purpose of ease and clarity in this discussion, we define the following notations: $\vec{X}$ represents the real-time measurements of the state variables at eight locations at the start of each MPC sampling period, x_D^{outlet} represents the radially-averaged mole fraction of PA at the reactor outlet at the end of each MPC sampling period, x_T represents the maximum fluid temperature along the FBR over each MPC sampling period, and u_i where i=1, 2, 3 and 4 represents the jacket temperature profile over each MPC sampling period. We note the corresponding estimated state variables can be obtained from the output of the ensemble regression model as follows:

$$f_X : \mathbf{R}^{8\times10\times6} \rightarrow \mathbf{R}^{8\times6} \; | f_X(\tilde{Y}(t_k)) = \tilde{X}(t_k) \tag{17}$$

$$f_D : \mathbf{R}^{8\times10\times6} \rightarrow \mathbf{R} \; | f_D(\tilde{Y}(t_k)) = \tilde{x}_D^{outlet}(t_k) \tag{18}$$

$$f_T : \mathbf{R}^{8\times10\times6} \rightarrow \mathbf{R} \; | f_T(\tilde{Y}(t_k)) = \tilde{x}_T(t_k) \tag{19}$$

where $\tilde{Y}(t_k)$ is the estimated trajectory of the state variables along the FBR from t_{k-1} to t_k predicted by the ensemble regression model to describe the distributed dynamics of the FBR. Additionally, the maximum fluid temperature and jacket temperatures are bounded by $x_T^{bound} := \{0 < x_T \le x_T^{max}\} \in \mathbf{R}$ and $U^{bound} := \{u_i^{min} \le u_i \le u_i^{max}\} \in \mathbf{R}^4$ (i=1, 2, 3, 4), respectively to avoid hot-spot formation. The optimization problem of the MPC is of the following form:

$$J = \min_{u \in S(\Delta)} \sum_{t_k}^{t_{k+N}} |\tilde{x}_D^{outlet}(t) - x_D^{set}|^2 \tag{20a}$$

$$\text{s.t.} \quad \tilde{Y}(t_{k+i+1}) = f_R(\tilde{X}(t_{k+i}), u(t_{k+i})), i = 0, 1, ..., N-1 \tag{20b}$$

$$\tilde{X}(t_{k+i}) = \vec{X}(t_k), i = 0 \tag{20c}$$

$$= f_X\left(\tilde{Y}(t_{k+i})\right), i = 1, ..., N-1$$

$$\tilde{x}_D^{outlet}(t_{k+i}) = f_D\left(\tilde{Y}(t_{k+i})\right), i = 1, ..., N-1 \tag{20d}$$

$$\tilde{x}_T(t_{k+i}) = f_T\left(\tilde{Y}(t_{k+i})\right), i = 1, ..., N-1 \tag{20e}$$

$$u(t_{k+i}) \in U^{bound}, i = 0, 1, ..., N-1 \tag{20f}$$

$$\tilde{x}_T(t_{k+i}) \in x_T^{bound}, i = 0, 1, ..., N-1 \tag{20g}$$

where $f_R(\cdot)$ represents the ensemble regression model, $S(\Delta)$ is the set of piecewise constant functions with period Δ and N is the number of sampling periods in the prediction horizon. In the optimization problem of Eq. (20), the objective function of Eq. (20a) is the sum of $|\tilde{x}_D^{outlet}(t) - x_D^{set}|^2$ over the prediction horizon. Eq. (20b) is the ensemble regression model $f_R(\cdot)$, which is used to predict the nonlinear distributed dynamics of the FBR (denoted by $\tilde{Y}$) based on the on-line measurements of the state variables, the feed operating parameters and the jacket temperature profile. It is noted that because the feed operating parameters are assumed to be constant in this investigation, they are omitted from Eq. (20b) for clarity. Eq. (20c) identifies the on-line measurements of the state variables, which are either the state measurements at $t = t_k$ from the CFD simulation (denoted by $\vec{X}(t_k)$) in the first prediction step, which corresponds to i=0, or the estimated state measurements derived from the outputs of the ensemble regression model (denoted by $f_X(\tilde{Y}(t_{k+i}))$) in all subsequent prediction steps. Eq. (20f) defines the input constraints over the entire MPC prediction horizon. Eq. (20g) defines state constraints over the MPC prediction horizon. Additionally, we assume that the states of the closed-loop system are measured at each MPC sampling time. After the optimal solution $u^*(t)$ of the optimization problem of Eq. (20) is obtained, only the first control action of $u^*(t)$ is sent to the control actuators to be applied over the next sampling period. Then, at the next instance of time $t_{k+1} := t_k + \Delta$, the optimization problem is solved again, and the horizon is rolled forward one sampling time. Under the MPC of Eq. (20), the closed-loop state x_D^{outlet} can be ultimately driven to its desired set-point x_D^{set} in the absence of model mismatch or operational disturbances (e.g., reactor feed disturbances) if there is sufficient control action to satisfy the constraints of Eq. (20f) and Eq. (20g). To account for model mismatch between the heterogeneous CFD model and the ensemble regression model and to eliminate the offset between the final

Algorithm

1: Import RNN Model f_R
2: Import state measurements $x(t_k)$
3: **function** MPC
4: **for** n←1 **to** N (prediction horizon)
5: **for** i←1 **to** num_of_models
6: **if** i=1: $x_{i-1}(t)=x_{initial}$
7: $x_i(t_{k+1})=f_R^{(i)}(x_i(t_k),u(t_k),x_{i-1}(t_{k+1}))$
8: Data processing (normalization)
9: Set temperature constraints x_T
10: **return** x_D^{outlet} and $\boldsymbol{J}=\boldsymbol{J}+(x_D^{outlet}-x_D^{set})^2$
11: **solve** MPC

Fig. 7 – The algorithm of integrating MPC with an ensemble regression model where N is the length of prediction horizon and i is the index of RNN models.

steady-state value of x_D^{outlet} and x_D^{set}, an additional deviation term $e(t_{k-1}) = \tilde{X}(t_{k-1}) - \vec{X}(t_{k-1})$ which represents the deviation between the actual state and the predicted state in the last step $t = t_{k-1}$ is applied in the predictive models of Eq. (20b) such that the model mismatch is accounted for in the predictive models (i.e., $\tilde{X}(t_{k+1}) = f_X(f_R(\tilde{X}(t_k), u(t_k))) + e(t_{k-1})$). Alternatively, offset can also be eliminated by integrating an integral feedback control scheme with MPC such that the overall control action can be obtained as: $u_{MPC+I}(t_k) = u_{MPC}(t_k) + u_I(t_k)$ (Wu et al., 2017).

In our work, the MPC is developed with the ensemble regression model via the optimization package *SciPy* in Python. The integration of the optimization problem of MPC and the ensemble regression model is demonstrated by the algorithm shown in Fig. 7. Specifically, at $t = t_k$ the ensemble regression model and the state measurements are imported to MPC. Subsequently, the MPC control actions are calculated by solving the optimization problem of Eq. (20). It is noted that in Eq. (20b), the ensemble regression model is used to predict the nonlinear distributed dynamics of the FBR (i.e., state trajectories at eight locations along the FBR over the prediction horizon $t = [t_k, t_{k+N})$) by taking on-line state measurements $X(t_k)$ and control actions $u(t)$, $t = [t_k, t_{k+N})$ as the inputs. In Fig. 7, it is further demonstrated that in the ensemble regression model (shown in Fig. 6), in addition to state measurements and control actions, each sub-ensemble also takes the predicted states $x(t_{k+1})$ from the upstream sub-ensemble (denoted by $f_R^{i-1}(\cdot)$ which corresponds to an $(i-1)$th location) as the inputs to predict the system dynamics at an ith location.

6.2. *Integrating MPC and dynamic CFD simulation*

In this section, an integrated framework of MPC and dynamic CFD simulation is developed via message passage interface (MPI) to implement MPC within CFD dynamic simulations. Since UDF supports C language only, the co-compilation of UDF and MPC solver depends on the specific programming language used. For example, if the optimization problem of MPC is solved in Python environment, then the MPC solver and UDF are compiled in Python and C environments, respectively, and a communication bridge between the MPC and ANSYS Fluent

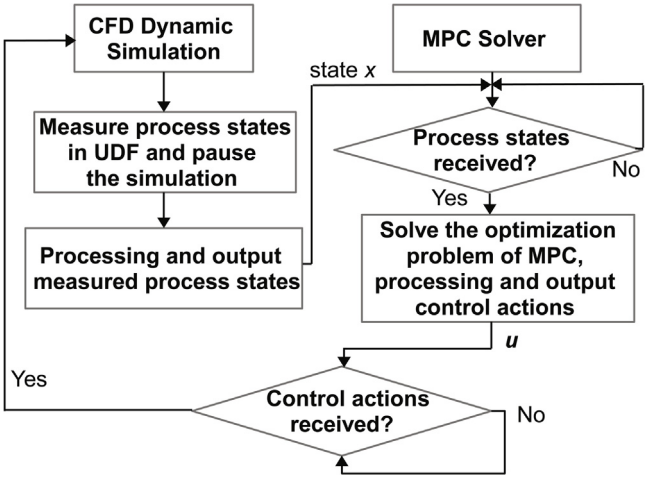


Fig. 8 – Basic structure of the proposed integrated MPC and CFD dynamic simulation via a lock synchronization mechanism.

UDF needs to be developed through a lock synchronization function embedded in the UDF.

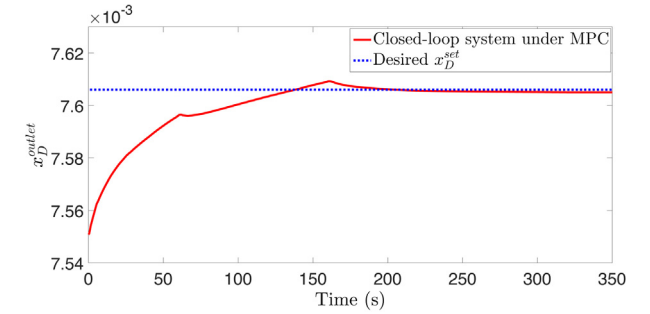
To accomplish the aforementioned data exchange, we present two methods as follows:

Method 1:

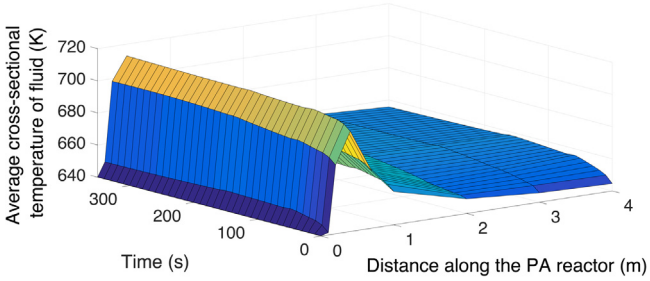
In Wu et al. (2017), a built-in MPC solver via a C subroutine was proposed to solve the constrained quadratic programming (QP) problem of MPC using an active-set method. Specifically, the optimization problem of Eq. (20) is first represented in the standard form of a QP problem, and the optimal solutions are calculated by iteratively solving equality-constrained QP subproblems. The advantage of this method is that the MPC solver is built in the same C environment within UDF, which facilitates the compilation and reduces computation time in exchanging data. However, the drawback of this method is that the algorithms of the MPC solver become complicated when the optimization problems of MPC are formulated to be non-convex and NP-hard for complex, large-scale chemical reaction systems. The development of a built-in MPC solver on a case-by-case basis becomes time-consuming and lacks robustness. Therefore, a second method that bridges a communication pathway between a robust existing algorithm for solving large-scale nonlinear optimization problems and CFD dynamic simulations is proposed.

Method 2:

The second method is to integrate MPC with dynamic CFD simulation by taking advantage of existing MPC solvers. As shown in Fig. 8, the lock synchronization mechanism is utilized to exchange data between the CFD simulation and the MPC solver. Specifically, at the end of each sampling period of CFD dynamic simulation, the measurements of process states are first obtained via UDF (e.g., functions C_T, F_YI are utilized to derive fluid temperature and PA concentration in ANSYS Fluent via user-defined functions (UDF)) and then sent to the MPC solver. Meanwhile, the CFD simulation is forced to wait for the control actions from the MPC solver. Then, the MPC solver is invoked to solve the optimization problem to derive control actions $u(t)$, for the next sampling period $t \in [t_k, t_{k+1})$ based on received process state measurements at $t = t_k$. Finally, the CFD simulation receives $u(t_k)$ and continues dynamic simulation until the next sampling time step. The above procedure is repeated with new measurements at the end of eachsampling period. Therefore, under the integration of MPC and dynamic CFD simulation, jacket temperatures



(a) The outlet PA concentration x_D^{outlet} profile.



(b) The fluid temperature profile.

Fig. 9 – The trajectories of the product concentration at the reactor outlet (Fig. 9a) and the fluid temperature profile (Fig. 9b) of the closed-loop system under MPC with $x_D^{set} = 7.606 \times 10^{-3}$.

(i.e., manipulated inputs calculated by MPC) can be adjusted at each sampling step such that x_D^{outlet} can be ultimately driven to its set-point in the absence of model mismatch or disturbances. Additionally, it should be noted that actual process states are measured in CFD simulations while normalized process states are utilized in RNN models and MPC solvers. Therefore, data processing function is also embedded in the MPC solver to convert it to normal value before sending back to CFD simulator.

Remark 1. The sampling period of MPC is set as the same sampling period in CFD dynamic simulation in our work such that control actions can be adjusted to better drive x_D^{outlet} to x_D^{set}. However, it should be noted that the sampling period of MPC can be different from the sampling period used in CFD dynamic simulations, i.e., CFD simulation is running in a sampling period Δ_{CFD} and MPC is activated each Δ_{MPC} sampling period, where Δ_{MPC} is an integer multiple of Δ_{CFD}. In this case, the CFD simulation still measures process states every Δ_{CFD} period, but will only send them to the MPC solver every Δ_{MPC} period. Then, the first control action of the optimal solutions $u^*(t)$ obtained in MPC will be sent back to the CFD simulation to be applied for the next Δ_{MPC} period. It is demonstrated that in general, an MPC with a short sampling period Δ_{MPC} can improve the performance of the closed-loop system by reducing oscillations. However, as a result, a long prediction horizon should be equipped with a short Δ_{MPC} to involve the dynamic behavior for a sufficient long period of time in predictive models, which will lead to longer computation time. Therefore, Δ_{MPC} and Δ_{CFD} need to be carefully determined to achieve a balanced trade-off between smooth dynamic responses and computational efficiency.

7. Closed-loop simulation results

In this section, the closed-loop performance of a fixed-bed catalytic reactor used in the phthalic anhydride synthesis under MPC is investigated. All simulation settings of the closed-loop simulations are the same as those used in the previously studied open-loop system with $\Delta_{CFD} = 0.1$ s. We demonstrate the case of set-point tracking under a disturbance-free environment. The initial steady-state is at $x_D^{outlet} = 7.550 \times 10^{-3}$, and the jacket temperatures are $T_{j1} = 640$ K, $T_{j2} = 640$ K, $T_{j3} = 640$ K and $T_{j4} = 640$ K, respectively. Since the ensemble regression model is developed based on the database generated from open-loop simulations in which the jacket temperatures of the four discrete noninteracting zones are systematically adjusted independently within the range of ±20 K and a step change of 5 K from the nominal value of 640 K, the set-point of the MPC is chosen to be the highest achievable value (i.e., $x_D^{set} = 7.606 \times 10^{-3}$) within the range observed in the database to guarantee a good performance of the closed-loop system.

It is demonstrated in Fig. 9a that radially-averaged concentration of PA at the outlet x_D^{outlet} is successfully driven to $x_D^{set} = 7.606 \times 10^{-3}$ under MPC, and it takes around 170 s for x_D^{outlet} to reach its final value. The dynamic performance of the closed-loop system under MPC improves significantly compared to an open-loop control as it takes around 300 s for the system to settle to $x_D^{set} = 7.606 \times 10^{-3}$. The corresponding manipulated input profiles are given in Fig. 10, where it is demonstrated that the jacket temperatures of the four zones stay at a high temperature (around 685 K) for the first 20 s to increase x_D^{outlet} quickly, and then the jacket temperatures decrease to their steady-state values such that x_D^{outlet} can approach its desired set-point x_D^{set} and ultimately settle down to x_D^{set}. Additionally, it is demonstrated in Fig. 9b that the fluid temperature profile is maintained below the maxi-

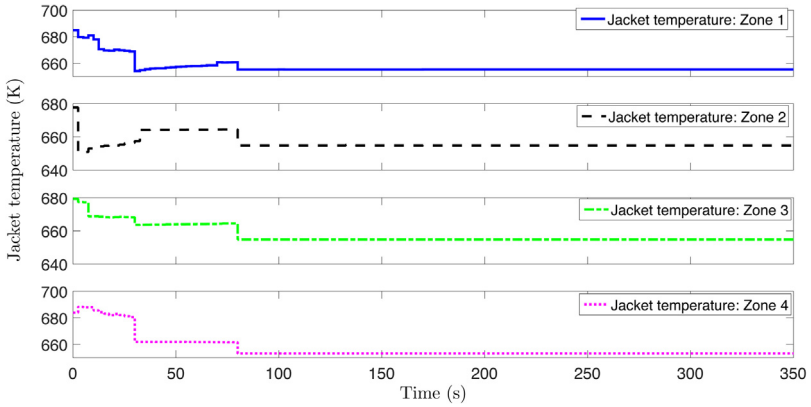


Fig. 10 – The jacket temperature profiles of the closed-loop system under MPC with $x_D^{set} = 7.606 \times 10^{-3}$.

mum allowable temperature 773 K under the constraints of Eq. (20g), which avoids the formation of hot-spots that could deactivate the catalyst.

Additionally, in our previous work Wu et al. (2019), an MPC based on a data-driven linear state-space model using the same dataset generated from CFD simulations was developed. It was demonstrated that in closed-loop simulation of tracking a certain set-point of PA concentration at outlet, there exist offsets for some of the set-points within the range of training dataset due to process/model mismatch, and therefore, an integral control term was added to the control action calculated by MPC to eliminate the offset. However, extensive closed-loop simulation results revealed that the neural network model-based MPC without additional integral or deviation control term is able to track the PA concentration at the outlet to any set-point within the range of training dataset with no offset due to the advantages of neural network model to capture nonlinearity of the system in a wide range of operating conditions and reduce process/model mismatch. Due to space limitations, in the present manuscript, we only included the closed-loop simulation results for the largest set-point change within the range included in the database.

8. Conclusion

In this work, a general framework for linking a CFD solver, ANSYS Fluent and other computing platforms using the lock synchronization mechanism was proposed to extend the utilities of CFD solvers from strictly modeling and design to also control and optimization applications. Specifically, a fixed-bed reactor (FBR) with a highly exothermic process was utilized to demonstrate the effectiveness of the proposed approach, under which the product yield can be maximized while suppressing the hot-spot temperature to avoid catalyst deactivation and thermal runaway. Initially, a high-fidelity two-dimensional axisymmetric heterogeneous CFD model for an industrial-scale FBR was developed in ANSYS Fluent. Then, the open-loop simulation of CFD model was conducted to create a database for a wide operating regime of the FBR, from which recurrent neural network and ensemble learning were used to derive a homogeneous ensemble regression model using the application program interface (API) Keras. Subsequently, based on the RNN models derived from ensemble regression, an MPC control scheme was developed to drive the process outputs to the desired set-points and suppress the magnitude of the hot-spot temperature to avoid catalyst deactivation. The CFD model, the ensemble regression and the MPC were integrated to create a closed-loop system by linking ANSYS Fluent to MPC optimizer in Python via a message-passing interface (MPI) with lock synchronization mechanism. Finally, the closed-loop simulation results demonstrated the robustness and effectiveness of the proposed approach.

Acknowledgments

Financial support from the National Science Foundation and the Department of Energy is gratefully acknowledged.

References

Anastasov, A.I., 2003. An investigation of the kinetic parameters of the o-xylene oxidation process carried out in a fixed bed of high-productive vanadia-titania catalyst. Chem. Eng. Sci. 58, 89–98.

Calderbank, P.H., Chandrasekharan, K., Fumagalli, C., 1977. The prediction of the performance of packed-bed catalytic reactors in the air-oxidation of o-xylene. Chem. Eng. Sci. 32, 1435–1443.

Chollet, F., et al., 2015. Keras, https://keras.io.

Christofides, P.D., 2001. Nonlinear and Robust Control of PDE Systems: Methods and Applications to Transport-Reaction Processes. Springer Science & Business Media.

De Klerk, A., 2003. Voidage variation in packed beds at small column to particle diameter ratio. AIChE J. 49, 2022–2029.

Jones, E., Oliphant, T., Peterson, P., et al., 2001. SciPy: Open source scientific tools for Python, URL: http://www.scipy.org/.

Lao, L., Aguirre, A., Tran, A., Wu, Z., Durand, H., Christofides, P.D., 2016. CFD modeling and control of a steam methane reforming reactor. Chem. Eng. Sci. 148, 78–92.

Le, Q.V., Jaitly, N., Hinton, G.E., 2015. A simple way to initialize recurrent networks of rectified linear units, arXiv preprint arXiv:1504.00941.

Lipton, Z.C., Berkowitz, J., Elkan, C., 2015. A critical review of recurrent neural networks for sequence learning, arXiv preprint arXiv:1506.00019.

Mendes-Moreira, J., Soares, C., Jorge, A., Sousa, J.F.D., 2012. Ensemble approaches for regression: A survey. ACM Computing Surveys (CSUR) 45, 10.

Mülheims, P., Kraushaar-Czarnetzki, B., 2011. Temperature profiles and process performances of sponge packings as compared to spherical catalysts in the oxidation of o-xylene to phthalic anhydride. Industrial & Engineering Chemistry Research 50, 9925-9935.

Orozco, G., Gomez, J.R., Sanchez, O.F., Gil, I.D., Duran, A., 2010. Effect of kinetic models on hot spot temperature prediction for phthalic anhydride production in a multitubular packed bed reactor. Can. J. Chem. Eng. 88, 224–231.

Sarosh, A., Hussain, A., Pervaiz, E., Ahsan, M., 2018. Computational fluid dynamics (CFD) analysis of phthalic anhydride's yield using lab synthesized and commercially available (V_2O_5/TiO_2) catalyst. Eng. Technol. Appl. Sci. Res. 8, 2821–2826.

Sidhu, H.S., Narasingam, A., Siddhamshetty, P., Kwon, J.S.I., 2018. Model order reduction of nonlinear parabolic pde systems with moving boundaries using sparse proper orthogonal decomposition: Application to hydraulic fracturing. Comput. Chem. Eng. 112, 92-100.

Sidhu, H.S., Siddhamshetty, P., Kwon, J.S.I., 2018b. Approximate dynamic programming based control of proppant concentration in hydraulic fracturing. Mathematics 6, 132.

Tran, A., Pont, M., Crose, M., Christofides, P.D., 2018. Real-time furnace balancing of steam methane reforming furnaces. Chem. Eng. Res. Design 134, 238–256.

Vaquerizo, L., Cocero, M., 2018. CFD-Aspen Plus interconnection method. Improving thermodynamic modeling in computational fluid dynamic simulations. Comput. Chem. Eng. 113, 152-161.

Wu, W., Huang, M.Y., 2003. Nonlinear inferential control for an exothermic packed-bed reactor: Geometric approaches. Chem. Eng. Sci. 58, 2023–2034.

Wu, Z., Aguirre, A., Tran, A., Durand, H., Ni, D., Christofides, P.D., 2017. Model predictive control of a steam methane reforming reactor described by a computational fluid dynamics model. Industrial & Engineering Chemistry Research 56, 6002-6011.

Wu, Z., Tran, A., Ren, Y., Barnes, C., Christofides, P.D., 2019. Computational fluid dynamics modeling and control of phthalic anhydride synthesis in a fixed-bed catalytic reactor. Proceedings of the American Control Conference, in press, Philadelphia, Pennsylvania.