Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Non-Dimensional Feature Engineering and Data-Driven Modeling for Microchannel Reactor Control

Calvin Tsay * **Michael Baldea** *

* *McKetta Department of Chemical Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: calvint@utexas.edu, mbaldea@che.utexas.edu).*

Abstract: Catalytic plate microchannel reactors (CPRs) are a promising means for modular hydrogen/fuels production from distributed natural gas resources. However, the equipment miniaturization presents challenges for process control, including spatially-distributed models, limited availability of measurements, and fast process time constants. In the present paper, we investigate the use of data-driven models—specifically, artificial neural networks (ANNs)—to estimate temperature "hotspots" within CPRs. We prescribe nonlinear transformations of the model inputs in the form of well-known dimensionless quantities (e.g., Reynolds number), and we show that these engineered features can improve the prediction capability of computationally parsimonious ANNs using a first-principles reactor model. Finally, we present a simulation case study that demonstrates the use of a trained ANN for inferential model predictive control.

Keywords: feature engineering, manufacturing processes, process control, chemical industry

1. INTRODUCTION

The modular manufacturing paradigm for chemical processing involves using decentralized, flexible production facilities to access distributed feedstocks (Baldea et al., 2017). For instance, "stranded" natural gas resources are located in remote areas and cannot be monetized via conventional means. Rather, they require distributed, small-scale gas-to-liquids (GTL) processes (e.g., steam-methane reforming followed by Fischer-Tropsch synthesis). These processes must be robust and efficient, as they are often deployed at a distance from support personnel.

One pathway towards process technologies that exhibit good "scale-down" performance (i.e., capital cost scales favorably as capacity is diminished) is *process intensification*, a design paradigm whereby transfer and transport limitations are minimized, e.g., by equipment miniaturization (Stankiewicz and Moulijn, 2000). In this context, catalytic plate microchannel reactors (CPRs) have been identified as a promising avenue for scale-down of GTL processing facilities (Chen et al., 2017). CPRs consist of alternating (co- or counter-current) millimeter-sized channels, separated by catalyst-coated plates. Microchannel reactors for steam-methane reforming are potentially an important intermediary step in small-scale, Fischer-Tropsch synthesis. In this application, the endothermic reforming reactions and the exothermic catalytic combustion of methane occur in adjacent channels, thermally coupling the two reacting spaces. Channel sizes in the order of millimeters minimize transfer and transport limitations. CPRs are thus an order of magnitude smaller than conventional reactors of comparable capacity and use significantly less catalyst by weight.

However, there are several control challenges for CPRs, mostly related to flow maldistribution and localized temperature maxima (known as "hotspots"). In particular, hotspots can be present at steady state or may arise during transient operation. They adversely affect reactor performance and jeopardize the integrity of the catalyst coating/support structures (Norton et al., 2006). Many literature studies have focused on design modifications that mitigate the formation of hotspots, including distributed fuel feed locations (Kolios et al., 2005), optimizing catalyst coating placement (Zanfir et al., 2011), and incorporating thermally activated flow control devices (Pattison et al., 2015). Fewer studies have focused on temperature control strategies, which are required to deal with fluctuations in reforming flow, inlet compositions, etc. Here, Pattison and Baldea (2013) investigated the insertion of phase-change material between reactor plates as a means of distributed temperature control, in conjunction with a higher-level supervisory controller derived via steady-state analysis. Odunsi et al. (2016) applied a similar hierarchical strategy to control temperature for a Fischer-Tropsch reactor.

A major impediment in the implementation of effective temperature/hotspot control strategies is the lack of relevant measurements in CPRs. In this paper, we employ artificial neural networks (ANNs) as reduced-order reactor models to predict maximum wall temperature (i.e., the magnitude of the hotspot) as a function of limited measurements. We show that performing nonlinear transformations on the input variables (i.e., feature engineering) based on dimensional analysis (in particular, using dimensionless numbers) improves the accuracy of the learned models when the goal is small model size, and thereby computational efficiency. Finally, we use a trained ANN for inferential model predictive control of a CPR.

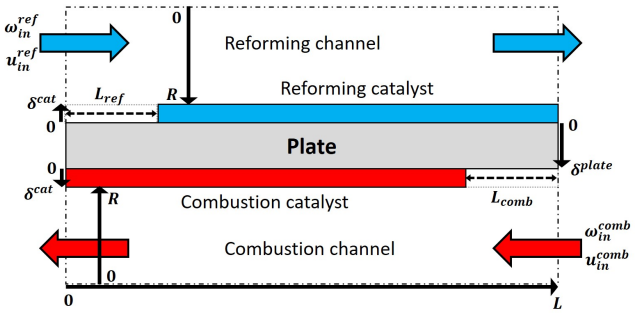

Fig. 1. Catalytic plate reactor model with counter-current reforming and combustion flows.

2. PHYSICAL SYSTEM CONSIDERED

This work considers the steam-methane reforming CPR system described and modeled by Zanfir et al. (2011), corresponding to a reactor comprising a stack of channels separated by catalytic plates. The primary model assumptions are: (i) end effects, or heat transfer between the bottom/top channels with the environment, are negligible at the stack level, and (ii) flow distribution between channels is substantially even. Given these assumptions, the dynamics of the CPR can be well-represented by modeling a single catalytic plate (with reforming catalyst coating on one side and combustion catalyst coating on the other), as well as the adjacent half-channels. The feed streams enter the two channels in a counter-current configuration, with nominal velocities of 4 m/s. Flow in both channels is assumed to be laminar, with fully developed velocity profiles (between infinite parallel plates) at all times.

2.1 Mathematical Model

The reactor is modeled in two dimensions (2-D), as shown in Figure 1. Symmetry boundary conditions are imposed at the centerlines of the channels. No-flux conditions are imposed at the channel outlets, and equal-flux conditions are imposed at the gas-solid interface. The gas phase in each channel and the solid wall are modeled using a 2-D geometry, while the catalyst layers are relatively thin and modeled in one dimension only (1-D). The complete model equations are provided in Zanfir et al. (2011).

The three reactions occurring in the reforming channel are:

$$CH_4 + H_2O \leftrightarrow CO + 3H_2;\ \Delta H = +206 \text{ kJ/mol} \quad (1)$$

$$CO + H_2O \leftrightarrow CO_2 + H_2;\ \Delta H = -41 \text{ kJ/mol} \quad (2)$$

$$CH_4 + 2H_2O \leftrightarrow CO_2 + 4H_2;\ \Delta H = +164 \text{ kJ/mol} \quad (3)$$

with (1) being steam-methane reforming, (2) water-gas shift, and (3) reverse methanation. Their rates are described using the kinetic model by Xu and Froment (1989). The (overall endothermic) reforming reactions are supported thermally by the heat from catalytic combustion of methane in the adjacent combustion channels:

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O;\ \Delta H = -803 \text{ kJ/mol} \quad (4)$$

The rate of combustion (4) is modeled using a first-order kinetic expression with respect to methane and an activation energy of 90 kJ/mol. Furthermore, the effects of homogeneous combustion, which has a non-negligible contribution at high temperature, are included.

2.2 Model Implementation

The reactor design was determined by solving a steady-state optimization problem to maximize the sum of the conversions in both channels, defined using the amounts of methane present at each channel inlet and outlet. Further details can be found in Zanfir et al. (2011); Pattison and Baldea (2013). Note that the resulting offset catalyst distribution is optimized for its steady-state performance and does not necessarily correspond to the reactor design that is the most operable or controllable. The optimized reactor design and model parameters are given in Table 1.

Table 1. CPR model parameters

Parameter	Value
Reactor length (L)	0.6 m
Channel height ($2R$)	2.0 mm
Catalyst thickness (δ^{cat})	20 μm
L_{ref}	9.0 cm
L_{comb}	15 cm
Inlet temperature	793.15 K
Reforming inlet composition (ω_{in}^{ref})	19.11 %CH_4, 72.18% H_2O, 2.94% CO_2, 0.29% H_2, 5.48% N_2
Combustion inlet composition (ω_{in}^{comb})	5.26% CH_4, 22.09% O_2 72.65% N_2

The model was implemented and simulated in gPROMS (Process Systems Enterprise, 1997-2019). The axial direction was discretized using a second-order finite difference scheme with 40 nodes in the reverse direction of flow (backward for the reforming channel and catalyst, forward for combustion). The solid wall plate was discretized in the axial direction using a central finite difference approximation. Orthogonal collocation on finite elements was used to discretize the transverse direction, motivated by the dispersive mass and heat transport. Third-order polynomials were used, with three finite elements for each domain. In total, the discretized model contains 6715 state and 29943 algebraic variables. The initial point used for dynamic simulations corresponds to the steady-state solution given nominal flow velocities ($u_{in}^{comb}, u_{in}^{ref} = 4$ m/s).

3. DATA SET GENERATION

3.1 Simulation Methodology

To generate a data set with dynamic excitations, the inlet velocities to both the reforming and combustion channels were represented as pseudo-random multi-level sequences, or PRMSs (Wang and Baldea, 2014), constructed based on a frequency analysis of the nominal system. We assume that changes in inlet velocities with higher frequencies than the corner frequency (10^{-1}) are filtered by the system due to the thermal inertia of the wall (Pattison and Baldea, 2013). Furthermore, we assume that a model-based, feed-forward controller (e.g., Section 5) should deal with disturbances on frequencies one order of magnitude lower than the corner frequency. This simplified analysis provides a range for the duration of PRMS steps, such that $(t_{\text{PRMS},i+1} - t_{\text{PRMS},i}) \in [50s, 350s], i = 1, ..., N-1$.

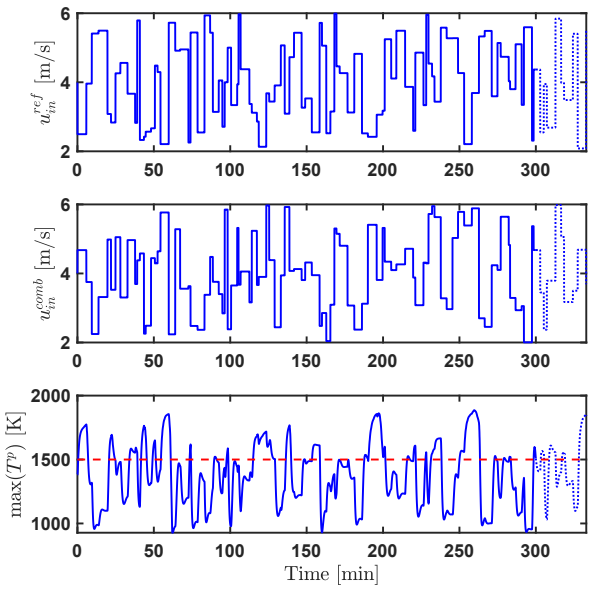

Fig. 2. Top: PRMS of reforming inlet velocity. Middle: PRMS of combustion inlet velocity. Bottom: corresponding response of maximum plate temperature, with upper limit (1500 K) marked by the dashed line.

Each PRMS was then generated by sampling independent random values between 2–6 m/s, corresponding to ±50% deviations from the nominal 4 m/s inlet flow to each channel. The same step durations were used for both flows:

$$u_{in}^{ref} = u_{\mathrm{PRMS},i}^{ref}; \forall\ t \in \left[t_{\mathrm{PRMS},i}, t_{\mathrm{PRMS},i+1}\right) \quad (5)$$

$$u_{in}^{comb} = u_{\mathrm{PRMS},i}^{comb}; \forall\ t \in \left[t_{\mathrm{PRMS},i}, t_{\mathrm{PRMS},i+1}\right) \quad (6)$$

where $u_{\mathrm{PRMS},i}^{ref}$ and $u_{\mathrm{PRMS},i}^{comb}$ are, respectively, the randomly sampled values of the inlet flow rate to the reforming and combustion channels at the i^{th} PRMS step.

Figure 2 shows the 100-level PRMS used to generate the data set for this work. We are interested in the response of the CPR system, in the form of the instantaneous maximum plate temperature, max(T^p). The response of max(T^p) to changes in u_{in}^{ref} and u_{in}^{comb} is shown in Figure 2. The equipment limit of 1500 K is exceeded at multiple points, as these data represent an open-loop system identification experiment (no controller was imposed). A more conservative input signal may be required for a real-world system identification experiment (e.g., via thermal imaging). In all, 20000 data points were generated as a combined training and test data set. The final 10% of the data set (plotted as dotted lines in Figure 2) was preserved for use as test data, and was not used in model training.

3.2 Measurements and Features

We assume that the inlet velocity to each channel is measured, as well as the gas temperature at each inlet (the temperature at each inlet changes slightly due to changes in the plate temperature). Furthermore, we assume that thermocouples are placed and measure the temperatures of the wall plate at the extremity covered with catalyst coating ($z = 0$ for combustion and $z = L$ for reforming). Finally, we assume that a thermocouple is placed at the end of the combustion catalyst zone ($z = L - L_{comb}$) to provide an internal measurement. Note that this is the only measured variable inside the reactor, owing to manufacturing challenges of constructing a CPR outfitted with internal sensors. The placement of the internal thermocouple is physically motivated, as the extremity of the combustion catalyst layer is where the catalytic combustion reaction ignites, and the temperature begins to rapidly increase in the combustion channel.

Sánchez-Maroño et al. (2005) showed the advantages of feature extraction using dimensionless combinations of measured variables, e.g., obtained using physical knowledge or the more general Π-Theorem (Buckingham, 1914), Furthermore, Rudolph (1997) proposed using such dimensionless features as inputs to "dimensionally homogenous" neural networks for improved accuracy and training properties. We therefore supplemented the above measurements with several dimensionless engineered features.

Several common dimensionless numbers can be computed easily from the CPR measurements, including the Reynolds number $Re = \frac{\rho u L}{\mu}$, Prandtl number $Pr = \frac{C_p \mu}{\kappa}$, and Schmidt number $Sc = \frac{\mu}{\rho D}$ (note that the inlet compositions are assumed constant). Dimensionless numbers also often appear as coefficients when energy and material balance equations are non-dimensionalized, and we further used the dimensionless numbers described for a similar system by Baldea and Daoutidis (2007): thermal Péclet number $Pe_T = RePr$, mass Péclet number $Pe_M = ReSc$, and Fourier number Fo. The values of these six ratios were computed at the inlet to each channel. For each wall-temperature measurement, the Damköhler number Da_j was computed for each reaction j (Baldea and Daoutidis, 2007). The appearance of these dimensionless ratios in non-dimensionalized equations supports that they may inform dynamic models when used as inputs, although they only involve (functions of) existing measured variables.

The full list of inputs/features is given in Table 2. The superscripts ref, $comb$, and p refer respectively to the reforming side, combustion side, and separating wall plate.

Table 2. Inputs to data-driven models

Location (z)	Measured Variables	Computed Variables
0	u^{ref}, T^{ref}, $T^{p,comb}$	Re^{ref}, Pr^{ref}, Sc^{ref}, Pe_T^{ref}, Pe_M^{ref}, Fo^{ref}, Da_1^{ref}, Da_2^{ref}, Da_3^{ref}
L	u^{comb}, T^{comb}, $T^{p,ref}$	Re^{comb}, Pr^{comb}, Sc^{comb}, Pe_T^{comb}, Pe_M^{comb}, Fo^{comb}, Da_1^{comb}
$L - L^{comb}$	$T^{p,comb}$	Da_1^{comb}

4. DATA-DRIVEN MODEL STRUCTURES FOR CPR

This section describes the training of ANNs to estimate max(T^p) using the simulated data set of measured and engineered features. We examined two classes of neural networks that are suitable for modeling time-series data, prioritizing models of very small size. Such reduced-dimensional models could be deployed in, e.g., "MPC-on-a-chip" applications, which are well-suited for fast controller response in modular production scenarios. The location of the temperature peak (and desired output variable, max(T^p)) varies during operation, and therefore measuring its value requires thermal imaging, distributed internal

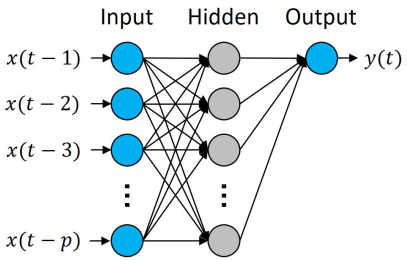

Fig. 3. Architecture of a time-delay neural network (TDNN) with a single hidden layer.

thermocouples, etc. Therefore, historical values of the output variable $\max(T^p)$ are typically unavailable in practice, and autoregressive terms/models were not considered.

4.1 Time-Delay Neural Networks

The Time-Delay Neural Network (TDNN) is a simple feedforward ANN for modeling time-series data (Waibel et al., 1989). The units at each layer are fully connected, and each hidden layer receives inputs from a "window" of outputs from the layer below. Figure 3 shows the architecture of a TDNN with a single input variable x and a single hidden layer. The window size is p, and each unit in the hidden layer receives the first p time-delayed values of x as inputs. In the case of multiple inputs, $x_1, ..., x_n$, each unit in the hidden layer receives $p \times n$ inputs, corresponding to p time-delayed values of each input. Position dependence is removed during training of the network parameters (i.e., the same weights are applied at all values of t), making the model shift invariant.

TDNNs were trained on the above data set using the MATLAB Deep Learning Toolbox (The MathWorks, Inc., 2018). The Levenberg-Marquardt algorithm was used with default stopping parameters, except the maximum number of epochs was increased to 10000 and the maximum number of validation fails to 500 (i.e., training stops if the validation error increases for 500 consecutive epochs). Input and output variables were scaled to be in $[-1, 1]$, and `tanh()` activation functions were used. Prediction mean squared error (MSE) was selected as the loss function. The parameters were initialized using random weights, and 20 TDNNs were trained for each model configuration. All models contained a single hidden layer, with the number of hidden units varying from one to eight. A sample time of 5 s was used to account for measurement time.

The performance of the 20 TDNN models in terms of MSE is shown in Figure 4, where it can be seen that $\max(T^p)$ can be modeled accurately using fairly low-dimensional TDNN models. Using all the variables in Table 2 (seven measured and 17 computed) improved the best MSE for 1–6 hidden units, compared to the case where only the seven measured variables are used. However, this improvement in accuracy could be attributed to increasing the number of model parameters, from 7×p to 17×p input-layer connections. To isolate the effect of adding dimensionless number inputs, we trained a model with the seven measured variables and the most important seven computed features. The most important features were identified as those with the largest weights in the model with all 17 computed variables. After this feature selection

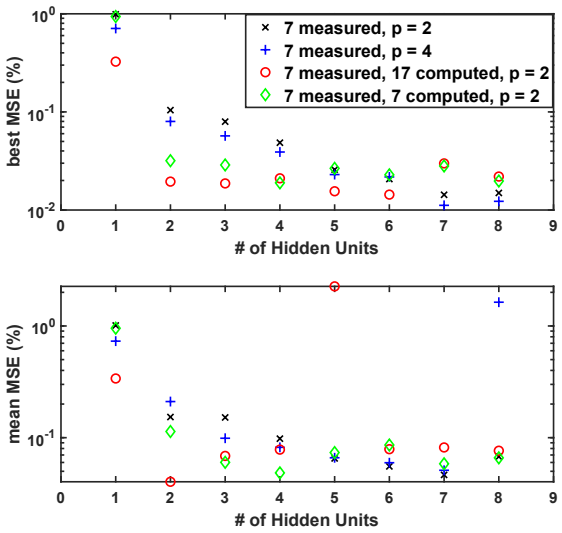

Fig. 4. Best (top) and mean (bottom) performance of TDNNs with varying number of hidden units.

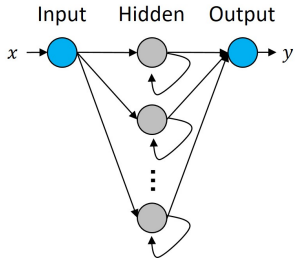

Fig. 5. Architecture of a recurrent neural network (RNN) with a single hidden layer. Recurrent connections are only shown from each unit to itself for simplicity.

process, the model with $p = 2$ includes the same number of input-layer connections (and trained parameters) as a model with the seven measured variables as inputs, and $p = 4$. The model with seven dimensionless inputs outperforms the two models with only measured variables in terms of best MSE at small model architectures (up to four hidden units). The mean MSE is also lower for these small models, suggesting they may also be easier to train.

4.2 Recurrent Neural Networks

The Recurrent Neural Network (RNN) is a more complicated ANN architecture, with an internal state, or "memory." An RNN includes *recurrent* connections, wherein the inputs to some hidden units are the time-delayed outputs of others, or themselves (Medsker and Jain, 1999). Figure 5 shows an RNN architecture with a single input variable x and a single hidden layer. Each hidden unit is depicted with a recurrent connection to only itself for simplicity. In reality, hidden units in RNNs typically have full recurrent connections, i.e., each hidden unit receives the previous p outputs from all hidden units as inputs. The same weights are again applied at all values of t during training, making the RNN shift invariant.

RNNs were trained on the same data set using the MATLAB Deep Learning Toolbox (The MathWorks, Inc., 2018). The Levenberg-Marquardt algorithm was again used with default stopping parameters, except the maxi-

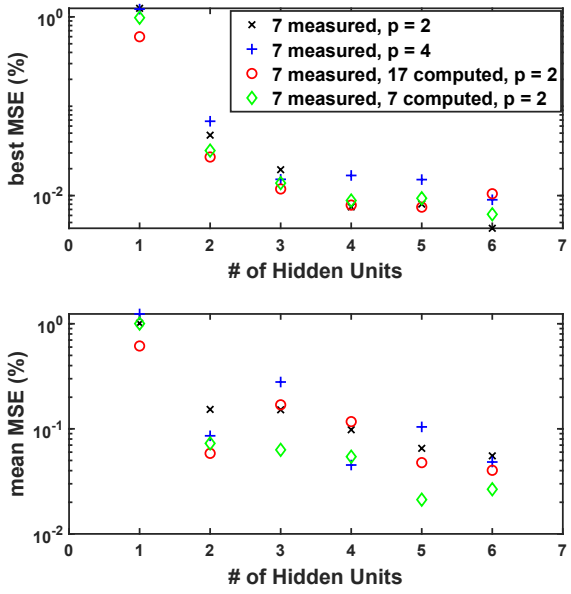

Fig. 6. Best (top) and mean (bottom) performance of RNNs with varying number of hidden units.

mum number of epochs was increased to 5000 and the maximum number of validation fails to 100. These parameters were shortened because RNNs require more computational effort to train; however, in our computations the models seemed to reach low MSE values quickly. `tanh()` activation functions and the prediction MSE loss function were again used. Ten RNNs were trained for each model configuration, using a sample time of 5 s and a single hidden layer. One to six units were considered in the hidden layer.

The performance of the 10 RNN models in terms of MSE is shown in Figure 6. The models shown correspond to same four input structures in Figure 4. Here, we found the effect of introducing dimensionless engineered features to be negligible. In other words, the prediction accuracy of the models is not improved significantly by adding the dimensionless numbers as inputs. This suggests that the RNNs are capable of learning the underlying nonlinear correlations present in the data and do not benefit from the introduction of simple engineered features. Indeed, the MSEs of the RNN models are lower than those of the corresponding TDNN models. Lastly, we note that the bulk of the connections (trained parameters) in RNN models are recurrent connections, and additional input variables have therefore lower relative impact on a model.

5. MODEL PREDICTIVE CONTROL OF CPR

A transient simulation study was performed to demonstrate the application of the above models to model predictive control (MPC). The controlled variable, $\max(T^p)$, must satisfy 1000 K $< \max(T^p) <$ 1500 K to avoid reactor extinction (lower bound) and catalyst/structural damage (upper bound). The same data set (Figure 2) was used to perform system identification, in order to obtain models that describe the effect of changes in the disturbance variable, $u_{\text{in}}^{\text{ref}}$, and the manipulated variable, $u_{\text{in}}^{\text{comb}}$, on $\max(T^p)$. A fourth-order, linear state-space model with ∼20% MSE was identified for each input.

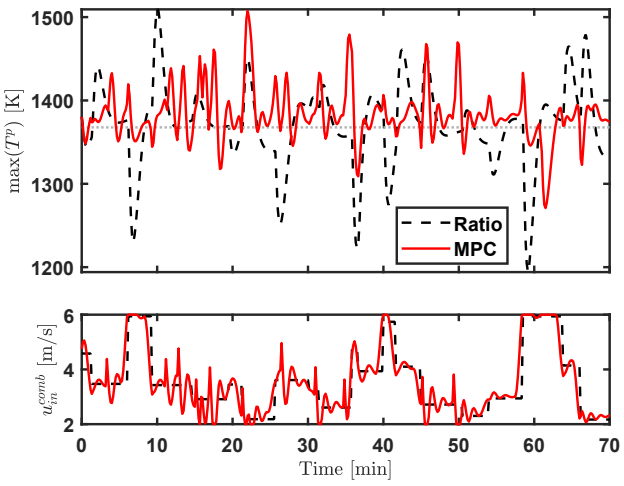

Fig. 7. Maximum reactor temperature during the disturbance sequence with model predictive control assuming $\max(T^p)$ is measured. The dotted line marks the MPC setpoint.

A feedforward ratio controller, $u_{\text{in}}^{\text{comb}} = R \times u_{\text{in}}^{\text{ref}}$, was selected as a benchmark. For the considered CPR, $R = 1$ based on the steady-state energy balance at nominal operation. A linear MPC system was implemented using the above state-space models in the MATLAB Model Predictive Control Toolbox (The MathWorks, Inc., 2018), using the default weights, a sample time of 10 s, a prediction horizon of six time samples (i.e., 1 min), and a control horizon of two time samples. A 70-min PRMS with the same temporal properties as in Section 3.1 was used to represent u_{in}^{ref}, and a constant setpoint for $\max(T^p)$ equal to its nominal steady-state value was imposed. The performances of the benchmark ratio controller and the MPC assuming perfect $\max(T^p)$ measurements are compared in Figure 7. MPC control actions (assumed instantaneous) were computed in 1.0 ± 0.3 ms (mean $\pm\ \sigma$) on a Windows 10 system with Intel Core i7-8700 CPU at 3.20 GHz. This scenario demonstrates the best-case MPC performance.

For the practical case where $\max(T^p)$ is not measured, its value can be instead inferred using an ANN. As TDNNs are much simpler than RNNs to evaluate, a TDNN with four hidden units was selected. Recall that dimensionless features improved TDNNs with 1–4 hidden units (Figure 4). A simple inferential control configuration similar to that of Bahar et al. (2004) was used, wherein $\max(T^p)$ is estimated—from the seven measured variables and the seven most important engineered features—at each sample time, and the estimate is passed to the MPC in place of a measurement. The same MPC was used otherwise, and the performance of this *inferential* MPC (denoted as IMPC) is shown in Figure 8 for the same disturbance sequence. The IMPC control actions (with ANN prediction) were computed in 11.8 ± 0.3 ms. Estimating $\max(T^p)$ with the simple TDNN only increased MPC computational time by ∼11 ms, making the 10 s sample time feasible.

Figure 8 reveals that IMPC response is more oscillatory; this may be attributed to inaccuracy in forecasting, leading the controller to take more aggressive actions even when $\max(T^p)$ is close to its setpoint, and could be addressed through further controller tuning. Nevertheless,

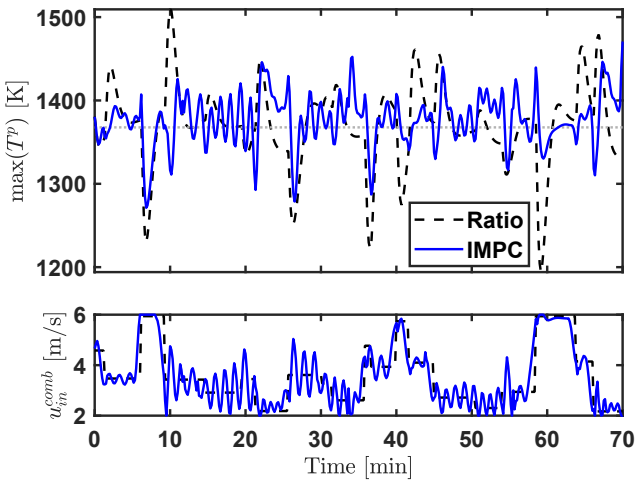

Fig. 8. Maximum reactor temperature under IMPC.

the mean deviation of max(T^p) from the nominal value (and MPC/IMPC setpoint) of 1367.7 K was 34.5 K under ratio control, 24.1 K under MPC, and 25.2 K under IMPC. The IMPC performed remarkably similar to MPC with perfect measurements, and both methods maintained max(T^p) much closer to its nominal value than the ratio controller did. In turn, this has important consequences for CPR performance: the mean deviation of reforming conversion from its nominal value of 96.93% was respectively 1.50%, 1.29%, and 1.24% under ratio control, MPC, and IMPC.

6. CONCLUSIONS

Microchannel reactors are a promising means for monetizing distributed resources, but present control challenges such as complex models, limited measurements, and fast response times. With model-based control in mind, this work studies ANN modeling techniques for predicting the evolution of temperature "hotspots." We prescribed nonlinear transformations of measured variables, in the form of common dimensionless quantities (e.g., Reynolds number). These engineered features improved the prediction accuracy for small TDNN models, a result we expect to be relevant in embedded model-based control applications. On the other hand, the benefit of introducing the engineered features was negligible for larger data-driven model structures, such as larger TDNN models or RNN models. This result suggests the ability of larger or more complex ANNs to discern and learn non-linear relationships present in the systems underlying the data, and make accurate output predictions. A simulated case study demonstrated the application of a small TDNN including dimensionless inputs to (inferential) model predictive control.

REFERENCES

Bahar, A., Özgen, C., Leblebicioğlu, K., and Halıcı, U. (2004). Artificial neural network estimator design for the inferential model predictive control of an industrial distillation column. *Ind. Eng. Chem. Res.*, 43, 6102–11.

Baldea, M. and Daoutidis, P. (2007). Dynamics and control of autothermal reactors for the production of hydrogen. *Chem. Eng. Sci.*, 62(12), 3218–30.

Baldea, M., Edgar, T.F., Stanley, B.L., and Kiss, A.A. (2017). Modular manufacturing processes: Status, challenges, and opportunities. *AIChE J.*, 63(10), 4262–72.

Buckingham, E. (1914). On physically similar systems; illustrations of the use of dimensional equations. *Phys. Rev.*, 4(4), 345.

Chen, J., Yan, L., Song, W., and Xu, D. (2017). Methane steam reforming thermally coupled with catalytic combustion in catalytic microreactors for hydrogen production. *Int. J. Hydrogen Energ.*, 42(1), 664–80.

Kolios, G., Glöckler, B., Gritsch, A., Morillo, A., and Eigenberger, G. (2005). Heat-integrated reactor concepts for hydrogen production by methane steam reforming. *Fuel Cells*, 5(1), 52–65.

Medsker, L. and Jain, L.C. (1999). *Recurrent neural networks: design and applications*. CRC press.

Norton, D., Wetzel, E.D., and Vlachos, D.G. (2006). Thermal management in catalytic microreactors. *Ind. Eng. Chem. Res.*, 45(1), 76–84.

Odunsi, A.O., O'Donovan, T.S., and Reay, D.A. (2016). Temperature stabilisation in Fischer–Tropsch reactors using phase change material (PCM). *Appl. Therm. Eng.*, 93, 1377–93.

Pattison, R.C. and Baldea, M. (2013). A thermal-flywheel approach to distributed temperature control in microchannel reactors. *AIChE J.*, 59(6), 2051–61.

Pattison, R.C., Donahue, M.M., Gupta, A.M., and Baldea, M. (2015). Localized temperature control in microchannel reactors using bimetallic thermally-actuated valves. *Ind. Eng. Chem. Res.*, 54(24), 6355–61.

Process Systems Enterprise (1997-2019). general PROcess Modeling System (gPROMS). www.psenterprise.com/products/gprom.

Rudolph, S. (1997). On topology, size and generalization of non-linear feed-forward neural networks. *Neurocomputing*, 16(1), 1–22.

Sánchez-Maroño, N., Fontenla-Romero, O., Castillo, E., and Alonso-Betanzos, A. (2005). Modelling engineering problems using dimensional analysis for feature extraction. In *Proc. Int. Conf. Artificial Neural Networks (ICANN)*, 949–54.

Stankiewicz, A.I. and Moulijn, J.A. (2000). Process intensification: transforming chemical engineering. *Chem. Eng. Prog.*, 96(1), 22–34.

The MathWorks, Inc. (2018). MATLAB Release R2018a. https://www.mathworks.com/products/matlab.html.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K.J. (1989). Phoneme recognition using time-delay neural networks. *IEEE Trans. Acoust.*, 37(3), 328–39.

Wang, S. and Baldea, M. (2014). Identification-based optimization of dynamical systems under uncertainty. *Comput. Chem. Eng.*, 64, 138–52.

Xu, J. and Froment, G.F. (1989). Methane steam reforming, methanation and water-gas shift: I. Intrinsic kinetics. *AIChE J.*, 35(1), 88–96.

Zanfir, M., Baldea, M., and Daoutidis, P. (2011). Optimizing the catalyst distribution for countercurrent methane steam reforming in plate reactors. *AIChE J.*, 57(9), 2518–28.