INL/CON-23-71414-Revision-0

Data-Driven Model Predictive Control for Temperature Management of Heat Pipe Microreactor

July 2023

Changing the World's Energy Future

Linyu Lin, Joseph Eugene Oncken, Vivek Agarwal

INL/CON-23-71414-Revision-0

Data-Driven Model Predictive Control for Temperature Management of Heat Pipe Microreactor

Linyu Lin, Joseph Eugene Oncken, Vivek Agarwal

July 2023

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Data-Driven Model Predictive Control for Temperature Management of Heat-Pipe Microreactor

Linyu Lin, Joseph Oncken, Vivek Agarwal

Idaho National Laboratory, Idaho Falls, ID

ABSTRACT

To enable the self-regulating capability of heat pipe (HP) microreactors, an anticipatory control strategy through model predictive control (MPC) could proactively respond to potential disturbances and deviations in operating setpoints. However, a key factor prohibiting the widespread adoption of MPCs in nuclear applications is the effort and computational costs associated with learning and calibrating first-principles-based process models when the target system is complex and when there are gaps between modeled and target reactor systems. In this paper, we demonstrate data-driven MPC using three approaches for modeling the system dynamics, including a linear state-space model, feedforward neural network, and recurrent neural networks long short-term memory. We present the development and validation process of each model and compare the performance of data-driven MPCs in controlling the temperatures of selected HPs at the evaporator and condenser regions in a 37-HP-monolith system. Our results show that, qualitatively, all data-driven MPCs are producing similar control actions, while quantitatively, with artificial neural nets (especially feedforward neural nets), MPC can better follow drastic changes in setpoints with smallest errors.

Keywords: data-driven model predictive control, machine learning

1. INTRODUCTION

To successfully deploy microreactors [1], self-regulating is a key feature that distinguish microreactors from large reactors. This operational feature requires a simple and responsive design, semi-autonomous operation, load-following capability, and seamless integration with renewables within microgrids.

To establish a technical basis for self-regulating microreactors, an anticipatory control strategy can proactively respond to anomalies and disturbances in anticipation of potential deviations from operating setpoints set by users. Anticipatory strategies are preferred in nuclear reactors rather than reactive or feedback systems because of the high thermal inertia and the actuator speed restrictions in reactor power maneuvers. This work uses model predictive control (MPC) for predicting system transients and computing a sequence of control actions that optimize the plant's future behaviors. MPC is attractive to nuclear applications because of its ability to directly and explicitly handle the multiple and complex constraints present in nuclear reactor systems. Meanwhile, in comparison to classical feedback controllers, such as the proportional-integral-derivative controller, MPC's capability to track time-varying reference trajectories under different constraints can be monitored by controlling the model accuracy of system dynamics. As a result, MPC is better suited to nuclear reactor controls given that requirements on reactor maneuvering and the controller's tracking capabilities are strictly defined.

Moreover, to ensure the adaptability of a control system from a single heat pipe (HP) system to an HP microreactor and support various operational features of microreactors, the MPC system, including process models and optimizer, should be modular and scalable as the system becomes more complex [2]. This study investigates MPCs with data-driven surrogates for the process model since high-fidelity models require a significant number of computations. Compared to mechanistic models, data-driven models show better adaptability in matching the actual data when there are gaps between modeled and targeted HP microreactors. The gaps refer to general differences between the modeled and target system, including differences in geometry, physics, phenomena, initial and boundary conditions, etc. Such gaps are caused by a lack of data from the prototype system and incomplete modeling of the complex physical system, which is multiphysics and multiscale in nature. Such adaptability is important in supporting microreactor research and developing different sub-areas, where different microreactor designs and operational features need to be tested.

Although data-driven MPCs have been applied to building energy management, robotics, and autonomous racing, their uses in nuclear applications are still in the early stage and requires investigation. This work is innovative in developing and applying data-driven MPC to support the self-regulating features of HP-cooled microreactors. Meanwhile, considering the requirements on reactor maneuvering and the controller's tracking capabilities, this study further presents comparative studies for MPC systems with different data-driven approaches for accurately representing HP dynamics in the standalone HP test and a 37-HP-monolith system (HPs mounted by monolith block) at the same time. Because of a lack of experimental data from physical facilities, we begin with a description of a high-fidelity simulator model of the single HP and a 37-HP-monolith system in Section 2. We introduce the transient results from the simulator for the training and validation of process models with different approaches. Section 3 defines the finite receding horizon control problem with MPC framework. Section 4 describes the design and demonstration stages for MPC systems with a linear regression model, feedforward neural network (FNN), and recurrent neural network (RNN). We compare the MPC performance with different modeling approaches in Section 5 and conclude the paper with a summary of the results and a brief discussion of future work in Section 6.

2. Simulator Models

Microreactors are a special class of small reactor systems, typically under 50 MW electricity [1]. Compared to small modular reactors, microreactors are very small in size and target nonconventional nuclear markets, including remote communities, deep space power, planetary surface power, disaster relief missions, etc. Among various microreactor designs, HP-cooled microreactors use HP elements to cool the core. The system structure is greatly simplified by omitting the main pipeline, circulating pump, and auxiliary equipment. In a generic HP-cooled microreactor design, the fission heat is transferred to heat pipes through the structural materials with high thermal conductivity [3]. The heat is then transported to an energy conversion system from the cold end of HPs. Compared to other microreactor designs, HP microreactors have a passive primary coolant, minimal moving parts, and low-pressure operations. Meanwhile, the HP design is tightly coupled, and the reactivity is sensitive to dimensional and material changes from the Doppler effect and stainless-steel monolith swelling [3]. As a result, maintaining temperatures at designated setpoints is critical to enable the stable and self-regulating operations of HP microreactors. Since there are no operating HP microreactors, this work uses Multiphysics Object-Oriented Simulation Environment (MOOSE) based tools as high-fidelity simulators for a generic non-nuclear 37-HP system design, which includes HP and stainless-steel monolith block models. Without any design-based self-regulating features, the target non-nuclear system requires (semi-) autonomous control system to respond to external disturbances accurately and reliably. In the simulated HP system, the fuel rods that generate fission powers are replaced by electrical heater rods.

2.1. HP Models

The HPs in a microreactor usually consist of stainless-steel encasements filled with a liquid-metal working fluid. At the evaporator region (hot end) of the HP, thermal heat from the fuel pins is absorbed, evaporating the working fluid. The vapor then travels axially through the vapor core, and the working fluid is condensed at the condenser region (cold end) of the HP, transporting a large amount of heat from the evaporator to the condenser. Meanwhile, a wick structure inside the HP drives the condensed working fluid back to the hot end of the HP via capillary force.

Based on the working principle of the HP, the entire domain can be discretized into three regions in the axial direction (i.e., the condenser, adiabatic region, and evaporator). To ensure high heat transfer rates by HPs, the surface temperatures of HPs at the evaporator region (denoted T_e) and condenser region (denoted T_c) should be accurately controlled [4]. In this work, we use a MOOSE-based simulation tool, named Sockeye [5], to generate high-fidelity representations of HP transients. Sockeye solves a thermal conduction equation to determine the material temperature distribution along the axial and radial directions of an HP:

$$\rho c_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = 0 \tag{1}$$

where ρ, c_p, and k are material properties (e.g., density, specific heat capacity, and thermal conductivity for HP working fluids and structural materials respectively) and T is the temperature solved by MOOSE numerical schemes. There are 100 and 19 elements in the HP axial and radial directions, respectively. Verification and validation results of Sockeye HP model can be found in reference [5].

2.2. Heat-Pipe System with Monolith

A standalone monolith block consists of a single HP surrounded by six heater rods in a hexagonal stainless-steel housing, while the 37-HP system consists of a hexagonal stainless-steel housing containing 37 HPs and 54 heater rods. All HPs are placed in monolith holes, where heat is transferred from the monolith block to HP evaporator regions through a gap conductance model. Meanwhile, the condenser regions of all HPs are connected with an energy conversion system, and heat is removed from the system. This work assumes that the heat transfer between the HP condensers and energy conversion system is controlled by a convective heat transfer coefficient (HTC). We assumed that the length of the HP evaporator region is the same as the thickness of monolith and that the HP evaporator temperatures in each hole are the same as the inner surface temperatures of surrounding monolith. Figure 1 shows the mesh structure for a 37-HP system and schematic configurations for the heat conduction simulations.

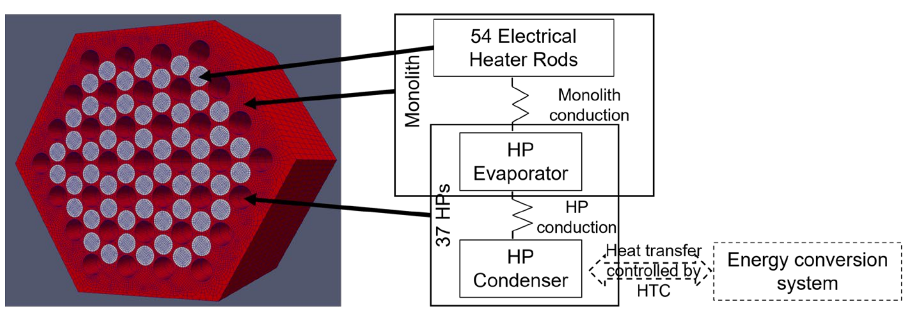

Figure 1. (Left) Mesh for a 37-HP system, where the stainless-steel monolith is red, heaters are white, and holes are HPs. (Right) Schematic configurations for the heat conduction simulations.

2.3. Transient Simulations

To develop models for system dynamics, transient simulations of HPs and monolith are generated from high-fidelity MOOSE-based models separately since the coupled simulations for the 37-HP system are computationally expensive. Moreover, the gap between training transients from the single HP and target 37-HP enables adaptability tests for data-driven models. For the single HP system, the transient simulation is generated by perturbing heat inputs at the HP evaporator and heat outputs at the HP condenser based on two uniform distributions. The power outputs are manipulated by controlling convective HTC to the atmosphere, and for each perturbation, the simulation will run for 200 seconds until a steady state is established. Figure 2 compares the simulation results of evaporator temperatures from four perturbations with different number of samples.

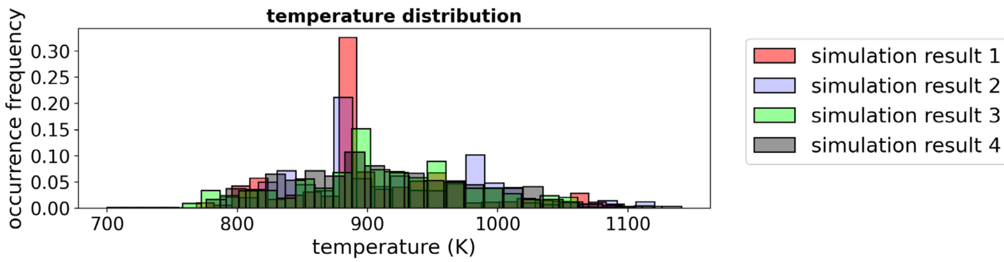

Figure 2. Distribution of evaporator temperatures.

Since the amount of data has significant impacts on the performance of data-driven models [6], this work generates four transient simulation results for a single HP with 20 (simulation1), 50 (simulation2), 100 (simulation3), and 200 (simulation4) perturbations, each of which has randomly generated heat inputs and outputs and lasts for 200 seconds. As the number of perturbations increases from 20 to 200, the length of transients increases from 5,000 to 40,000 seconds, and temperature distributions show larger variances. For the 37-HP monolith system, the transient is generated by perturbing the heater power inputs and HTC at the HP condenser. To evaluate the adaptability of data-driven models, the surrogate process models are developed using data from the single HP system and tested on the central and outermost HPs in the 37-HP monolith system.

3. Model Predictive Control

The essence of MPC is to optimize, over the manipulatable inputs, forecasts of process behaviors subject to equality and inequality constraints. The forecasting is performed using a process model (i.e., a predictor) over a finite time interval of length P. Equations (2) and (3) show a generic process model f for a 37-HP monolith system:

$$(x_{mono})_{k+1|j} = f\big((x_{mono})_{k|j}, u_{k|j}\big) \tag{2}$$

$$(x_{HP})_{k+1|j} = f\big((x_{mono})_{k+1|j}, (x_{HP})_{k|j}, u_{k|j}\big) \tag{3}$$

where x_{mono} and x_{HP} stand for the state variables of monolith block and HP system, respectively, x_{mono} represents unmanipulatable heat fluxes from monolith to HPs at the corresponding monolith hole, and x_{HP} represents unmanipulatable evaporator and condenser temperatures of selected HPs. Since the HP temperatures are assumed to be the same as the inner surface temperature of corresponding monolith hole, x_{HP} will depend on x_{mono}. u is the manipulatable variable, denoted as a control action in this work, and used in both Equation (2) and (3). Control actions include heater power inputs in W/m^3, denoted as Q, and HTC coefficient in W/m^2-K. j represents the time step, at which the MPC is activated to find the optimal control action, and $k = 1, \dots, P$ represents the steps ahead of j that system states are predicted. For convenience, we use subscript $k|j$ to denote k steps ahead of the real-time j, and the initial states of x when $k = 0$ ($x_{0|j}$ or x_j) are obtained from real-time measurements from the (simulated) microreactor.

Based on the system-state predictions over the prediction horizon, optimization can be performed over a control sequence $U = [u_{1|j}, \dots, u_{N|j}]$ by minimizing the summation J of stage cost function l over the entire prediction horizon, where J^* stands for the minimized summation of the stage cost function.

$$J^* = \min_U \left[\sum_{k=1}^{P} l(x_{k|j}, u_{k|j}) \right] \quad \textbf{(4)}$$

subject to

$$x_{k+1|j} = f(x_{k|j}, u_{k|j})$$

$$\boldsymbol{U} = [\boldsymbol{u}_{\boldsymbol{1}|\boldsymbol{j}}, \dots, \boldsymbol{u}_{\boldsymbol{P}|\boldsymbol{j}}] \in \mathbf{U}_{\boldsymbol{i}} \textbf{ for all } \boldsymbol{i} = \mathbf{1}, \dots, \boldsymbol{n}_{\boldsymbol{c}_{\boldsymbol{u}}}$$

$$\boldsymbol{X} = [\boldsymbol{x}_{\boldsymbol{1}|\boldsymbol{j}}, \dots, \boldsymbol{x}_{\boldsymbol{P}|\boldsymbol{j}}] \in \mathbf{X}_{\boldsymbol{i}} \textbf{ for all } \boldsymbol{i} = \mathbf{1}, \dots, \boldsymbol{n}_{\boldsymbol{c}_{\boldsymbol{x}}}$$

$$x_{0|j} = x_j$$

This work uses a weighted quadratic function as the stage cost function to approximate the actual multi-objective, sparse, and nondifferentiable cost function. $\mathbf{U}_i$ and $\mathbf{X}_i$ represent constraints on the control actions and state variables, respectively. n_{c_u} and n_{c_x} are the numbers of constraints for control actions and state variables, respectively. We use SciPy [7] optimization packages for solving Equation (4) and create a Python script for transferring information and driving both programs. Previous work [8] demonstrated MPC in managing surface temperatures of a single HP, while this work aims to test data-driven MPC for a 37-HP test article.

4. Data-Driven Model Predictive Control

Since the first-principles-based high-fidelity model requires significant computational resources, a surrogate process model is needed to represent the dynamics of selected state variables. Based on the transient data described in Section II, we investigate three data-driven approaches for developing a surrogate process model in MPC: a linear state-space model calibrated by the sparse identification of nonlinear dynamics with control (SINDYc), FNN, and RNN with long short-term memory (LSTM) units. During the design stage, each model is validated in making one-step and recursive temperature predictions. They are further tested in finding the actual control actions when connected to optimizers.

4.1. Model Predictive Control with Linear State-Space Model

SINDYc is a regression-based method for generating governing dynamic equations using a set of training data from a complex dynamic system [9]. SINDYc reduces to dynamic mode decomposition if formulated in discrete-time, with only linear functions, and without a sparsity-promoting L1 penalty term. To represent the transient data described in previous sections, we used a linear state-space model, shown in Equation (5), and calibrated the coefficients matrix A, B, and C based on Equation (6) by SINDYc [9]:

$$\frac{dx}{dt} = Ax(t) + Bu(t) + C \quad \textbf{(5)}$$

$$\Theta = \min_{\hat{\xi}_K} \frac{1}{2} \left| x'_k - \hat{\xi}_k \Theta^T(x_k, u_k) \right|_2^2 + \lambda \left| \hat{\xi}_k \right|_1 \quad \textbf{(6)}$$

where ξ are sparsity coefficients for a dynamic system, λ is the penalty coefficient for the sparsity-promoting term, Θ is the coefficient matrix (A, B, and C) of SINDYc surrogate in Equation (5), and x'_k represents state variables advancing one step ahead.

4.2. Model Predictive Control with Neural Networks

This work evaluates the MPC performance with FNN and RNN as the surrogate process models. Unlike the state-space model, neural networks (NNs) are used as multistep predictors in MPC, which skip the sequential state propagation and directly predict designated steps into the future. One of the major benefits is that the multistep predictor allows for simple bounds on the prediction error. Meanwhile, as nonparametric models, NN-based predictors require zero knowledge on the model forms and parameters and are able to express highly nonlinear transients. However, the major cost of having an accurate NN-based predictor is the increased model complexity with a large number of parameters [10]. Besides, process model validations are needed to ensure the accuracy and scalability of NN surrogates in representing the target transients beyond the training domain.

The FNN topology consists of a set of neurons linked together in a number of layers. At every time step t, a number of state variables $x(t)$ for the monolith block and HPs at different locations together with control actions $u(t)$ are fed into the NNs to predict the state variable at next time step. A two-layer FNN with 10 neurons in each layer and rectified linear unit activation function is trained using the Keras machine learning package [11]. Meanwhile, considering the advantages of an RNN in capturing and predicting time-dependent transients, this work investigates an RNN with an LSTM cell. Compared to vanilla RNN cells, LSTM can learn long-term dependency and deal with vanishing gradient problems. At sequence step t in the forget cell, LSTM first decides on information to be forgotten from the input state $x(t)$ and the previous hidden states with a sigmoid activation function. Next, the input gate uses a sigmoid activation function to decide on values to be updated, while a new candidate value is created with a tangential activation function. These values are combined as the new cell state and fed to the output cell. The output cell decides the output values with a sigmoid activation function and updates the hidden state based on the output gate and new cell state. A two-layer RNN with 10 LSTM units each layer and tanh activation function are trained using Keras. A similar loss function to FNN and backpropagation is adopted to update weights and biases within each LSTM unit.

4.3. Process Model Validation

To evaluate the performance of different data-driven approaches, this work tests the performance of each data-driven process model in predicting unseen testing data, making single-step and recursive predictions, and finding control actions that minimize deviations from reference trajectories of HP temperatures. For single and multistep validation, the testing data are separated from the transient simulations 1, 2, 3, and 4 based on time, where only the first 80% of transients are used in the training process. For finding control actions, the testing data are extracted from simulations with 37 HPs and monolith block. Since surrogate models are developed based on transient simulations from a single HP simulation, this is considered a scalability test, where scalability is defined as expanding the predictive capabilities of a surrogate for a single HP to meet application requirements for controlling HP temperatures in a 37-HP test article. Equation (7) summarizes the root mean squared error (RMSE) metrics in single-step and recursive predictions. For a single-step prediction, $\hat{y}_t$ equals the actual monolith and HP state variables x_t described in Equation (2) and (3). For recursive predictions, $\hat{y}_t$ is equal to model-predicted state variables $\hat{y}_t = \hat{x}_t = f(\hat{x}_{t-1}, u_{t-1})$ with initial conditions of $x_0 = \begin{bmatrix} T_e \\ T_c \end{bmatrix}_{t=0}$ and $u_0 = \begin{bmatrix} Q \\ HTC \end{bmatrix}_{t=0}$. Equation (8) evaluates MPC's ability to find the optimal control actions u^* that have HP temperatures following the predefined trajectory x_{SP} in the testing data from the 37-HP system, where the actual control actions u_a are known. In a closed-loop MPC system, such control actions are not known.

$$\hat{x}_{t+1} = \begin{bmatrix} \hat{T}_e \\ \hat{T}_c \end{bmatrix}_{t+1} = f(\hat{y}_t, u_t) \tag{7}$$

$$\epsilon_{\hat{x}} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}$$

$$u_{t+1}^* = \begin{bmatrix} Q^* \\ HTC^* \end{bmatrix} = \underset{u^* \in E,I}{\operatorname{argmin}} \left[\sum_{i=1}^{P} (\hat{x}_{t+i} - (x_{SP})_{t+i})^2 \right]$$

$$\varepsilon_{u^*} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}((u_a)_i - (u^*)_i)^2} \tag{8}$$

Table I summarizes the performance of SINDYc, FNN, and RNN-based data-driven models from the testing data. The goal is to evaluate the adaptability of data-driven models from different approaches when there are gaps between modeled and target systems. SINDYc shows the best performance in all tests. To evaluate the impact of the lengths of transient data on the performance of an FNN and RNN, we developed four models for each approach with the same model configurations but based on simulation 1, 2, 3, and 4 described in Section II-C. Both RNN and FNN errors in recursive predictions and finding actual actions are reduced significantly when longer transients are used as training data. To evaluate the impacts of different simulations, we calculated Pearson correlation coefficients between NN prediction errors and number of data points in each simulation. The average Pearson correlation coefficient value is -0.79 and 0.77 for FNN and RNN, respectively, and it means that more data result in a better NN performance with strong linear correlations.

Table I. Summary of model validation results for evaporator (T_e) and condenser (T_c) temperatures in RMSE based on four transient simulations.

	Single-step predictions		Recursive predictions		Actual actions	
	T_e	T_c	T_e	T_c	Q	HTC
SINDYc	3.3	2.1	3.3	2.1	11.3	4.87
RNN (simulation 1)	0.3	0.1	62.1	39.8	1.4×10^2	5.7×10^2
RNN (simulation 2)	2.3	1.4	47.8	26.2	1.0×10^2	77.73
RNN (simulation 3)	16.3	9.8	32.6	21.6	52.43	79.45
RNN (simulation 4)	24.0	14.8	31.7	20.6	29.19	39.18
FNN (simulation 1)	6.6	3.2	51.7	34.2	1.0×10^4	8.92×10^3
FNN (simulation 2)	0.3	0.005	48.6	27.0	5.2×10^3	6.08×10^3
FNN (simulation 3)	0.2	0.3	54.8	31.6	4.4×10^2	2.49×10^2
FNN (simulation 4)	0.2	0.08	57.1	31.8	4.8×10^2	2.41×10^2

5. Comparative Study

To evaluate the MPC performance with different data-driven models, we couple MPCs with the MOOSE-based 37-HP system simulator with the same objective function and box constraints for all cases. Unlike process model validations, which are performed based on transients from a single HP, the MPC is connected with the 37-HP system described in Section 2.2.

5.1. Simulation Settings

Figure 3 shows the reference trajectories for evaporator and condenser temperatures of the central HP. The selection of these setpoints is arbitrary and does not fall into the training domain. The objective function is to minimize squared errors between the actual and setpoint temperatures at the evaporator and condenser regions of central HPs. The sampling time in the MPC is 1 second, and the entire transient lasts 800 seconds. The length of the prediction horizon P is 10 seconds, and the length of the control horizon is 5 seconds, beyond which the control actions are assumed constant. Box constraints are added to control actions, where the absolute changing speed for heater inputs and HTC is required to be less than 50 W/m^3-s and 10 W/m^2-K-s, respectively. The same settings are applied to all MPCs except that different modeling approaches for system dynamics are used in the optimization problem.

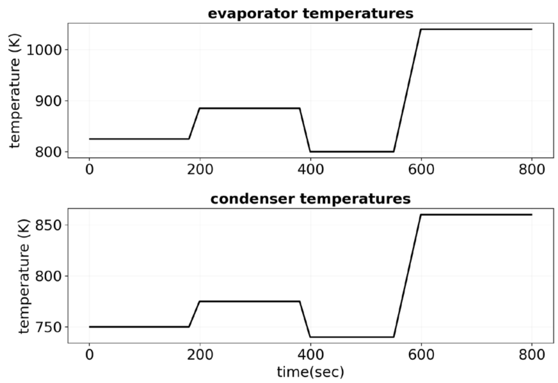

Figure 3. Reference trajectories for the HP evaporator and condenser temperatures at the central location.

Based on the heater inputs and HTC, the data-driven models predict the temperature distribution of the central and outermost HPs. We used Sequential Least Squares Programming in the SciPy optimizer for minimizing an objective function under constraints, using FNN and RNN based on data4.

5.2. Results

Optimal control actions for MPCs with all three data-driven methods are shown in Figure 4. Transients before 150 seconds and after 750 seconds are truncated for better visualizations. The chosen temperature trajectories require both the heater inputs and HTC to be manipulated.

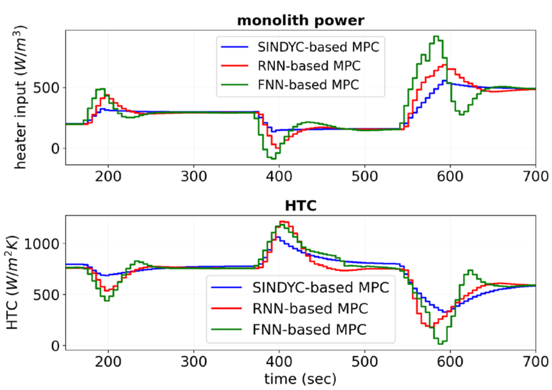

Figure 4. Comparisons of optimal actions from FNN-, LSTM- and SINDYc-based MPCs for heater inputs (top) and the HTC coefficient (bottom).

Qualitatively, all MPCs result in similar solutions. When there are changes in setpoints at 200, 400, and 600 seconds, NN-based MPCs result in larger changing speeds in control actions than SINDYc. This is caused by the fluctuated predictions from highly nonlinear NN models, while the SINDYc model tends to produce a smoother transient with better accuracy. Figure 5 plots the residuals errors between actual and reference temperature trajectories at evaporator and condenser regions of the central HP. The quantitative results in Table II suggest that a SINDYc-based MPC results in biggest deviations from the reference trajectory in RMSE, while the FNN-based MPC results in fluctuated deviations but the smallest overall RMSE. Although the nonlinearity of NN models result in larger validation errors, NN-based MPCs show better tracking capabilities because of the large changing speeds in control actions. Moreover, the receding horizon control strategy by the MPC mitigate error accumulation issues by NN models and further improve the accuracy at the demonstration stage.

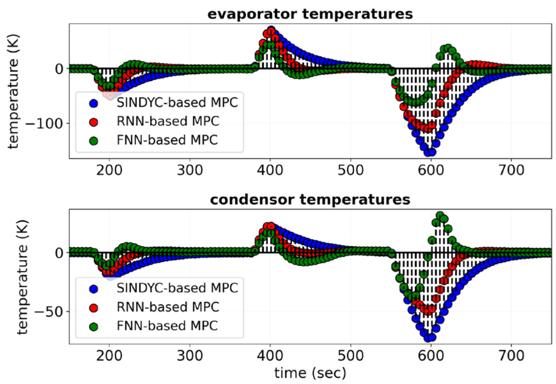

Figure 5. Residual error $T_{actual} - T_{SP}$ for FNN-, LSTM-, and SINDYc-based MPCs in tracking reference setpoints of evaporator and condenser temperatures of the central HP.

Overall, although SINDYc produces stabler and more accurate validation results, the inherent variance of NN models results in larger changes in control actions, and the NN-based MPCs better track the drastic changes in setpoints.

Table II. Summary of deviations between achieved and target HP temperatures.

	Root Mean Squared Errors (RMSE)	
	Evaporator temperatures (T_e)	**Condenser temperatures (T_c)**
SINDYc-based MPC	39.50	17.89
RNN-based MPC	27.54	11.63
FNN-based MPC	16.03	8.56

6. Conclusion

This work presents data-driven MPCs with three data-driven algorithms for finite receding horizon control using a SINDYc-calibrated linear state-space model, FNN, and RNN with LSTM cells. The training and testing transients are generated by perturbing inputs to the simulator model of a single HP and 37-HP system. By separating the control actions and uncontrollable state variables, three process models are validated in making single-step and recursive predictions and in finding the actual control actions in testing data. By selecting different transients from the training data, we also evaluate the adaptability of data-driven models because there are gaps between transients from the single HP system and the transient from the 37-HP system. At the design stage, SINDYc shows the best performance in recursive predictions and in finding the actual control actions. FNN shows the largest prediction errors, while all NNs can be improved by having more training data.

This work further develops MPCs in following arbitrary changes in temperature setpoints for the central HP in a 37-HP system and compares the performance given data-driven surrogates by different modeling approaches. We demonstrate that all data-driven MPCs result in similar control actions and can follow changes in temperature setpoints while avoiding the efforts and computational burdens associated with developing a physics-based system model. Although NNs have larger prediction errors than the state-space model, they are better at capturing nonlinear behaviors, which result in fluctuated predictions and larger changing speed in control actions. As a result, NN-based MPCs show better capabilities in following drastic changes in temperature setpoints because of such inherent variances. Our current and future works focus on demonstrating the capability of different data-driven MPCs on a complex microreactor system using simulation data from high-fidelity multiphysics simulator models. In addition, we aim to expand the operational scenarios and perform validations and uncertainty quantification for data-driven models based on experimental data.

ACKNOWLEDGMENTS

This work is supported through Idaho National Laboratory's Laboratory Directed Research and Development Program under Department of Energy Idaho Operations Office contract no. DE-AC07-05ID14517. This research made use of Idaho National Laboratory computing resources.

REFERENCES

[1] Idaho National Laboratory, "A microreactor program plan for the department of energy (INL/EXT-20-58919)," Idaho National Laboratory, Idaho Falls, ID, 2022.

[2] L. Lin, P. Athe, P. Rouxelin, M. Avramova, A. Gupta, R. Youngblood, J. Lane and N. Dinh, "Development and assessment of a nearly autonomous management and control system for advanced reactors," *Annals of Nuclear Energy,* vol. 150, p. 107861, 2021.

[3] B. Yan, C. Wang and L. Li, "The technology of micro heat pipe cooled reactor: a review," *Annals of Nuclear Energy,* vol. 135, p. 106948, 2020.

[4] H. N. Chadhry, B. R. Hughes and S. A. Ghani, "A review of heat pipe systems for heat recovery and renewable energy applications," *Renewable and Sustainable Energy Reviews,* vol. 16, no. 4, pp. 2249-2259, 2012.

[5] J. E. Hansel, R. A. Berry, D. Andrs, M. S. Kunick and R. C. Martineau, "Sockeye: A One-Dimensional, Two-Phase, Compressible Flow Heat Pipe Application," *Nuclear Technology,* vol. 7, pp. 1096-1117, 2021.

[6] L. Wang, L. Lin and N. Dinh, "Data coverage assessment on neural network based digital twins for autonomous control system," *Annals of Nuclear Energy,* vol. 182, p. 109568, 2023.

[7] P. Virtanen and e. al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods,* vol. 17, no. 3, pp. 261-272, 2020.

[8] L. Lin, J. Oncken, V. Agarwal, C. Permann, A. Gribok, T. McJunkin, S. Eggers and R. Boring, "Development and assessment of a model predictive controller enabling anticipatory control strategies for a heat-pipe system," *Progress in Nuclear Energy,* vol. 156, p. 104527, 2023.

[9] S. Brunton, J. Proctor and J. Kutz, "Sparse identification of nonlinear dynamics with control (SINDYc)," *IFAC-PapersOnLine,* vol. 49, no. 18, pp. 710-715, 2016.

[10] J. Köhler, K. Wabersich, J. Berberich and M. Zeilinger, "State space models vs. multi-step predictors in predictive control: Are state space models complicating safe data-driven designs?," *IEEE 61st Conference on Decision and Control (CDC),* pp. 491-498, 2022.

[11] F. Chollet and others, "Keras (https://github.com/fchollet/keras)," GitHub, 2015.