# Using Machine Learning Aided Computational Fluid Dynamics to Facilitate Design of Experiments

Ziqing Zhao[1], Amanda Baumann[1], Emily M. Ryan[1,2*]

1. Department of Mechanical Engineering, Boston University, Boston, MA, 02215, USA
2. Division of Materials Science and Engineering, Boston University, Boston, MA, 02215, USA

*Corresponding author: ryanem@bu.edu

## Abstract:

The design of novel reactors and chemical processes requires an understanding of the fundamental chemical-physical processes at small spatial and temporal scales, and systematic scale up of these studies to investigate how the process will perform at industrial scales. The financial and temporal costs of these studies can be significant. The use of statistical machine learning-based methods can significantly reduce these costs. The use of design of experiments methods can help design an experimental plan that efficiently explores the design space using the fewest experiments possible. Computational methods such as computational fluid dynamics (CFD) are effective tools for detailed studies of the small-scale physics and are critical aids to facilitate and understand physical experiments. However, CFD methods can also be time consuming, often requiring hours or days of time on supercomputers. In this research, we investigate the combination of machine learning with reducing 3D CFD simulation to 2D by exploiting axial symmetry to facilitate design of experiments. Focusing on a 3D carbon dioxide ($CO_2$) capture reactor as an example, we demonstrate how machine learning and CFD can help facilitate modeling and design optimization. A 2D CFD is used to simulate the chemical-physical processes in the reactor and is then coupled with machine learning to develop a less computationally expensive model to accurately predict $CO_2$ adsorption. The learned model can be used to optimize the design of the reactor. This paper demonstrates the decrease in temporal and financial costs of designing industrial-scale chemical processes by combining reducing CFD dimension and machine learning. Equally importantly, this research demonstrates the significance of selecting a proper machine learning algorithm for different tasks by comparing performances of different machine learning algorithms.

**Key Words:** Computational Fluid Dynamics, Machine learning, Reduced Order Model, Black-box Optimization, Data Augmentation

## Introduction:

Chemical science and engineering research development relies on data, and data limitations can severely hinder the research process. How to utilize data efficiently and obtain more data cost-effectively becomes essential in data-limited research projects. Tools and methods, such as fractional factorial design and sequences of experiments in Design of experiments (DoE), have been developed to efficiently explore parameter space and utilize data effectively[1]. However, physical experiments can be financially and temporally costly, limiting what is achievable even with DoE. With the improvement of software and hardware, computational methods such as computational fluid dynamics (CFD), molecular dynamics (MD), and density functional theory (DFT) are reliable alternatives to obtain data at lower cost[2,3,4]. While computational methods can alleviate some data limitation problems in DoE, the methods are still too costly to produce large datasets. Therefore, more resources are needed to further reduce the data-limitation problem, and a promising pathway is to utilize machine learning (ML) methods[5,6]. This study introduces a novel method that first reduces a 3D CFD into 2D by exploiting the axial symmetry, and then uses a combination of ML and CFD to aid DoE, helping to relieve data-limitation issues.

While DoE has been successfully used for efficiently exploring parameter spaces, one of its main limitations is how to propose new experiments based on previous ones so as to maximize the amount of information learned or for optimization[7]. Bayesian optimization, a general black-box optimization tool, can potentially overcome these limitations when used in conjunction with DoE. Bayesian optimization builds a surrogate model with quantified uncertainty using previously explored datapoints. Then based on the surrogate model, Bayesian optimization can either 1) explore query points with high uncertainty to improve the surrogate model, or 2) exploit the surrogate model and evaluate the most promising query point. Balancing the trade-off between exploration and exploitation makes Bayesian optimization a promising tool to guide DoE[8,9,10]. For example, Vahid et al used Bayesian optimization guided DoE to successfully propose two new high-strength aluminum alloys[10].

However, applying Bayesian optimization directly to DoE can be unrealistic. When the experiment is temporally or financially costly, Bayesian optimization might not have enough datapoints to explore or exploit. Therefore, computational simulations such as CFD and DFT can be important aids to reduce the cost[11,12,13,14]. DFT can be used to calculate molecular properties given atomic parameters, and Bayesian optimization can use the DFT calculation to find optimal molecular structures for specific purposes. For example, Ando et al used Bayesian optimization and DFT to successfully find new molecules which could improve organic semiconductor carrier mobility[11]. Bayesian optimization as a general black-box optimization tool can be applied not only to DFT, but also to CFD. CFD can be used to calculate flow properties given boundary and initial conditions, and Bayesian optimization can use the CFD calculations to find optimal boundary and initial conditions for specific purposes. For example, Park et al used Bayesian optimization and CFD to maximize gas holdup and minimize power consumption in a gas-sparged stirred tank reactor[12].

Bayesian optimization combined with traditional computational methods such as CFD and DFT can be powerful tools to facilitate DoE. However, even computational simulations can be too costly. Without enough datapoints to explore or exploit, Bayesian optimization might not yield any useful results. The issue of limited data can be potentially solved by two methods. First, when symmetries present in the model, the dimension of the model can be reduced from 3D to 2D. The dimension reduction can greatly reduce the computational cost while still maintaining reasonable accuracy[15,16,17,18]. For example, as Li et al mentioned, it is a widely applied practice to regard flow in a cylindrical column fluid-bed riser as two-dimensional flow[18]. With less computational cost for each CFD simulation, more data can be generated. In addition, more data can be generated by augmenting data through training machine learning algorithms on traditional computational methods[19,20,21,22,23]. For example, Masoumi et al used CFD to predict the outlet water temperature of asphalt solar collectors, then used an artificial neural network to learn from the CFD results and predict outlet water temperature accurately at a much lower computational cost[19]. Essentially, a reduced order surrogate model was learned by a ML algorithm from a traditional computational method.

Building a reduced order surrogate model potentially solves the limited amount of data problem in optimization[24,25]. Machine learning algorithms can learn surrogate models to make accurate but much less computationally intensive predictions. The learned surrogate models can then be used to substitute traditional computational methods and can be called upon by a more "data hungry" optimization algorithm. For example, Owoyele et al used machine learning algorithms such as kernel ridge regression (KRR), support vector regression, and artificial neural networks to learn reduced order surrogate models of CFD for internal combustion engine designs. Then a genetic algorithm used the surrogate models to find the optimal design variable to reduce fuel consumption in the design variable domain[26].

Similarly, this research adopted the strategy of learning surrogate models from CFD with one ML algorithm and optimizing the design with the learned surrogate model using another ML algorithm. In this work we focused on a carbon capture reactor for point source capture. A solid sorbent carbon capture reactor is modeled where a porous material (Microlith) captures $CO_2$ through temperature swing adsorption, where $CO_2$ is adsorbed at low temperatures, and desorbed at higher temperatures[27]. The quantitative correlation between adsorption and desorption rate and temperature was studied using CFD and machine learning.

The chemical transport phenomena in the reactor were modeled with CFD based on experimental data. Then a surrogate model was developed with KRR which was used in conjunction with Bayesian optimization to optimize the reactor design. There are two main novel contributions of this research. Firstly, this paper is the first that compares the effectiveness of using KRR and NN to learn from CFD simulation results, illustrating the importance of selecting the appropriate ML algorithm for a given material science and engineering research task. In addition, this research is the first that achieves reduced order modeling through combining KRR and transforming a 3D CFD into 2D using axial symmetry. The reduced order modeling method in this research can be used in other types of research as well. KRR can be replaced by other appropriate ML method such as NN, random forest, etc. When different symmetries or other properties present in a specific application, other dimension reduction techniques such as thin layer approximation, planar symmetry, etc. can replace the axisymmetric assumption to transform the CFD model from 3D to 2D. Even in the cases where 3D CFD modeling is necessary, 2D simplification can serve as a good starting point for qualitative analysis with significantly lower computational cost[18]. ML methods can further reduce the computational cost while maintaining reasonable accuracy.

## Methods:

In this research, both CFD and ML are used to investigate $CO_2$ capture in a solid sorbent reactor. The CFD model initially simulates the chemical-physical processes in the reactor. The CFD results are then used to train ML algorithms for faster predictions, which are then used for DoE and optimization.

### Computational Fluid Dynamics Methods

The carbon capture reactor considered in this work is based on a jelly-roll design where a coated wire mesh (Microlith) is rolled around itself and $CO_2$ is fed into the center of the roll and flows radially outward (Figure 1(a)). $CO_2$ is adsorbed onto the Microlith as it flows through the jelly roll. The adsorption of $CO_2$ is governed by mass, momentum, energy, and species conservation equations, which is modeled using a two fluid modeling approach[28].

Mass conservation is solved in the solid (s) and gas phases (g) as,

$$\frac{\partial}{\partial t}(\epsilon_s \rho_s) + \nabla \cdot (\epsilon_s \rho_s \vec{v_s}) = R_s \tag{1}$$

$$\frac{\partial}{\partial t}(\epsilon_g \rho_g) + \nabla \cdot (\epsilon_g \rho_g \vec{v_g}) = R_g \tag{2}$$

where $\epsilon$ is the volume fraction, $\rho$ is the density, $\vec{v}$ is the velocity, and R is the reaction source term.

Momentum conservation is solved in each phase as,

$$\frac{\partial}{\partial t}(\epsilon_s \rho_s \vec{v_s}) + \nabla \cdot (\epsilon_s \rho_s \vec{v_s} \otimes \vec{v_s}) = -\epsilon_s \nabla P + \nabla \cdot \tau_s - I_{gs} + \epsilon_s \rho_s \vec{g} \tag{3}$$

$$\frac{\partial}{\partial t}(\epsilon_g \rho_g \vec{v_g}) + \nabla \cdot (\epsilon_g \rho_g \vec{v_g} \otimes \vec{v_g}) = -\epsilon_g \nabla P + \nabla \cdot \tau_g + I_{gs} + f_g + \epsilon_g \rho_g \vec{g} \tag{4}$$

where $P$ is the pressure, $\tau$ is the stress tensor, $I$ is the second invariant of the deviator of the strain rate tensor for gas, $\vec{g}$ is the gravitational constant, and f is the flow resistance caused by the porous media of the Microlith. In this work, the solid is closely packed, so the momentum conservation equation for the solid phase is omitted. The resistance to flow caused by the solid phase was computed using the Syamlal-O'Brien model with the MfiX default C1=0.8 and D1=2.65.

As the reaction is exothermic, it is important to calculate the heat exchange. Therefore, the energy conservation equation is solved in each phase as,

$$\epsilon_s \rho_s C_{ps} \left[\frac{\partial T_s}{\partial t} + \vec{v} \cdot \nabla T_s\right] = -\nabla q_s - \gamma_s (T_s - T_g) - \Delta H_s + \gamma_{Rs}(T_{Rs}^4 - T_s^4) \tag{5}$$

$$\epsilon_g \rho_g C_{pg} \left[\frac{\partial T_g}{\partial t} + \vec{v} \cdot \nabla T_g\right] = -\nabla q_g + \gamma_s (T_s - T_g) - \Delta H_g + \gamma_{Rg}(T_{Rg}^4 - T_g^4) \tag{6}$$

where $C_p$ is the specific heat, $T$ is the temperature, $q$ is the conductive heat flux, $\gamma_s$ is the Fluid-solids heat transfer coefficient corrected for interphase mass transfer, $\Delta H$ is the heat of reaction, $\gamma_R$ is the radiative heat transfer coefficient, and $T_{Rg}$ is the fluid phase radiation temperature. Since the temperatures of both solid and fluid phases are between 295 Kelvin and 350 Kelvin, only conduction and convection are considered while radiation is turned off in the CFD modeling. In other words, $\gamma_{Rg}$ and $\gamma_{Rs}$ are set to zero.

Finally, the species conservation in each phase is given by,

$$\frac{\partial}{\partial \mathrm{t}}(\epsilon_s \rho_s X_s) + \nabla \cdot (\epsilon_s \rho_s \vec{v} X_s) = \nabla \cdot (D\, \nabla X_s) + R_s \tag{7}$$

$$\frac{\partial}{\partial t}\left(\epsilon_g \rho_g X_g\right) + \nabla \cdot \left(\epsilon_g \rho_g \vec{v} X_g\right) = \nabla \cdot \left(D\, \nabla X_g\right) + R_g \tag{8}$$

where X is mass fraction of species, and *D* is the diffusion coefficient of the species. All equations above are from MFiX documentation[28].

The adsorption of $CO_2$ on the Microlith is modeled as,

$$\mathrm{CO_2(g)} + \mathrm{n} * \rightarrow \mathrm{CO_2(s)} \tag{9}$$

where *n* indicates the number of sites needed for the adsorption of each $CO_2$ molecule. The reaction is modeled as Langmuir adsorption,

$$\mathrm{R_g} = \mathrm{k C_{CO_2} C_{ML}^{n}} \tag{10}$$

where k is the adsorption reaction rate coefficient; $\mathrm{C_{CO_2}}$ and $\mathrm{C_{ML}}$ are the molar concentrations of $CO_2$ and Microlith respectively. In the modeling process, k and n are fit to experimental data.

The adsorption in the Microlith is reversible, therefore it is also important to consider desorption,

$$\mathrm{CO_2(s)} \rightarrow \mathrm{CO_2(g)} + \mathrm{n} * \tag{11}$$

where *n* indicates the number of sites freed up after desorption of each $CO_2$ molecule, same as adsorption. The desorption reaction rate is modeled as linearly proportional to reactant concentration,

$$\mathrm{R_g} = \mathrm{k_d C_{RM}} \tag{12}$$

where $\mathrm{k_d}$ is the desorption reaction rate coefficient; $C_{RM}$ is the molar concentrations of Microlith with adsorbed $CO_2$. In the modeling process, $\mathrm{k_d}$ is fit to experimental data.

The governing equations for the $CO_2$ reactor are modeled with the Department of Energy's Multiphase Flow with Interphase eXchanges (MFiX) CFD software[28], using the built-in two fluid model with the adsorption (Eq. (10)) and desorption (Eq. (12)) source terms added to the model through user defined functions.

Machine Learning and Bisection Method

Three ML algorithms are used in this research: kernel ridge regression (KRR), Bayesian optimization, and neural network (NN). Besides ML algorithms, the bisection method is also used.

KRR is a regression tool. It stems from linear regression. Given data input matrix X and measurement vector y, linear regression algorithms find a coefficient vector $\vec{k}$ that minimizes the error $\epsilon$

$$\epsilon = \frac{1}{n}\left|\left|X\vec{k} - \vec{y}\right|\right|^2 \tag{13}$$

where n is the dimension of the vector $\vec{y}$, and the norm || • || symbol indicates the $L_2$ norm, i.e. for any n-dimensional vector $\vec{z}$,

$$\left|\left|\vec{z}\right|\right|^2 = \sum_{i=1}^{n} z_i \cdot z_i \tag{14}$$

Due to the linearity of matrix multiplication, linear regression is limited to regressing linear functions. To regress non-linear functions, the kernel trick can be used. The core of the kernel trick is mapping the data matrix X to a higher dimensional space with a mapping function Φ. The regression task then becomes minimizing,

$$\epsilon = \frac{1}{n}\left|\left|\Phi(X)\vec{k} - \vec{y}\right|\right|^2 \tag{15}$$

One downside of adding the kernel trick is that the regression can be much more susceptible to noise. Therefore, it is necessary to prevent the regression algorithm from overfitting (i.e. regressing the noise).

To reduce overfitting, regularization can be used. More specifically, a penalty term $\lambda \,||\vec{k}\,||$ is added to the error, and the regression task becomes minimizing

$$\epsilon = \frac{1}{n}\left|\left|\Phi(X)\vec{k} - \vec{y}\right|\right|^2 + \lambda\left|\left|\vec{k}\,\right|\right|^2 \tag{16}$$

where λ is a positive real number. The penalty term deters the coefficient vector $\vec{k}$ from being too complicated, hence alleviating the overfitting problem. For a more detailed mathematical explanation of KRR, refer to Vovk[29].

Bayesian optimization is an iterative optimization tool that relies on Bayes' rule to find the maximum value of an unknown function. In each iteration, the algorithm suggests a new point in the parameter space to evaluate the unknown function. The optimization proceeds in two steps; first, a Gaussian process is used to build a surrogate model based on the observed datapoints, and then an acquisition function selects the most promising point to evaluate next. This process of building a Gaussian process surrogate model and selecting a query point using an acquisition function is repeated until a stopping criterion is met, such as reaching a predetermined number of iterations. The result is an optimized set of parameters that maximize the unknown function.

Gaussian process can be defined as a distribution over functions $f(\boldsymbol{x})$ defined by a mean function $m(\boldsymbol{x})$ and a covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$,

$$f(x) \sim \mathcal{GP}\left(m(x), k(\boldsymbol{x}, \boldsymbol{x}')\right) \tag{17}$$

Therefore, for any finite set $X = \{x_1, x_2, \ldots, x_n\}$, function values sampled on $X$ follow a multivariate Gaussian distribution,

$$f(X) \sim \mathcal{N}\left(m(X), k(X, X)\right) \tag{18}$$

To build a surrogate model using Gaussian process, the unknown function $g(x)$ is first evaluated on a finite set $X_1$. Then a surrogate model $f(x)$ predicts what $g(x)$ evaluates to on another finite set $X_2$, with $f(X_2)$ following a Gaussian conditional distribution $f(X_2) \mid g(X_1)$. More specifically,

$$f(X_2) \mid g(X_1) \sim \mathcal{N}\left(m\left(X_2 | \, g(X_1)\right), k(X_2 | \, g(X_1), X_2 | \, g(X_1))\right) \tag{19}$$

where $m\left(X_2 | \, f(X_1)\right)$ is the conditional mean function of $X_2$ given the evaluation on $X_1$, and $k(X_2 | \, f(X_1), X_2 | \, f(X_1))$ is the conditional covariance function[30]. Based on $m\left(X_2 | \, f(X_1)\right)$ and $k\left(X_2 | \, f(X_1), X_2 | \, f(X_1)\right)$, the Gaussian process model can estimate the predictive uncertainty in addition to making predictions, which the acquisition functions can take advantage of.

Acquisition functions are regarded as policies that determine new query points on which the unknown function $g(x)$ is evaluated in the optimization process. Acquisition functions balance between exploration and exploitation by taking into account the tradeoff between evaluating points with high uncertainty for $f(x)$ (exploration) and points with high expectation that $f(x)$ is maximal (exploitation). Popular acquisition functions include the expected improvement, upper confidence bound, and probability of improvement. Probability of improvement selects the point with the largest probability to make an improvement as the next query point. Expected improvement takes into account not only the probability to make an improvement but also the expected amount of improvement that can be made, and selects the query point that can make the largest expected improvement. Upper confidence bound encourages exploring query points with large uncertainties and at the same time rewards exploiting the current surrogate model. Compared to the unknown function $g(x)$, acquisition functions can be computed quickly, making them useful for finding optima of $g(x)$. For a more detailed explanation of Bayesian optimization, refer to Agnihotri et al[31].

NN is a versatile machine learning algorithm inspired by biological nervous systems. This research uses NN as a regression tool. Each NN consists of neurons, weights, biases, and activation functions. Neurons are in the format of multiple layers of tensors. Write the $i^{\text{th}}$ layer of neurons as $X_i$. $X_i$ is multiplied by the corresponding weight matrix $W_i$, followed by the addition of a bias tensor $B_i$. An activation function $f$ then operates on the result of multiplication and addition to create the next layer of neurons $X_{i+1}$. Mathematically, the computation at each layer of neurons is represented by the equation

$$X_{i+1} = f(W_i \cdot X_i + B_i) \tag{20}$$

Eq. (20) defines the forward propagation of the neural network. Namely each neuron in the later layer is computed through the neurons and weights of the previous layer until it reaches the output layer. The output layer can be compared to the ground truth (for example, the experimental results) and a loss function similar to Eq. (15) can be calculated. Namely

$$L(W,B) = \frac{1}{N}\left\|\mathrm{Y} - \hat{Y}\right\|^2 \tag{21}$$

where W represents all the weights, B represents all the biases, $N$ is the number of samples in the input tensor $X$, $Y$ is the ground truth tensor, and $\hat{Y}$ is the NN prediction. The goal of NN is adjusting W and B so that the loss $L(W,B)$ is minimized. The adjustment procedure is called the backpropagation [32]. Typically, backpropagation is achieved through gradient descent.

The bisection method [33] is a numerical technique used to approximate the root of a given function. Given an unknown function $g(x)$ and a value $y$, the bisection method can be applied to approximate the $x$ value that satisfies $g(x) = y$. The method works by first identifying two values $x_{over}$ and $x_{under}$ such that $g(x_{over}) > y$ and $g(x_{under}) < y$, and then iteratively updating $x_{over}$ and $x_{under}$ based on the evaluation of their middle point $g(\frac{1}{2}(x_{over} + x_{under}))$.

## Implementation

The objective of this study was to model and enhance the performance of $CO_2$ capture in a solid sorbent reactor designed in a jelly roll configuration, as illustrated in Figure 1(a). Flue gas, which had a high concentration of $CO_2$, entered the reactor through a central inlet and exits radially outwards. The sorbent is a Microlith designed by Precision Combustion, Inc. (PCI). The Microlith consists of the sorbent material coated on wires and stacked as shown in Figure1(b). The simulation domain, shown in Figure 2, was an axisymmetric domain where the inlet for the flue gas was at the bottom, and the outlet was at the top. The flue gas flowed through the Microlith (shown in dark grey in Figure 2) in the axial direction, and left the domain from the top outlet. This axisymmetric domain was used as a simplified representation of the jelly roll configuration to solve the 2D Navier-Stokes equations for $CO_2$ flow and adsorption in the jelly roll. Experimental data was provided by industry collaborator PCI.

(a)

(b)

Figure 1: (a) Jelly roll configuration. (b) Stacked structure of Microlith.

Initial experimental data used to develop the CFD model was for $CO_2$ adsorption at a single temperature. Tests were run for 25 minutes with a Microlith containing 2.27 kg of $CO_2$ sorbent material, an airflow rate of 60 L/s, and a $CO_2$ volume concentration of 15%. The concentration of $CO_2$ at the outlet was measured every 40 seconds. In the initial CFD simulations, the Microlith was modeled with a void fraction of 70% based on the experimental setup. The inlet velocity u was set to be u=0.238 m/s and the bed thickness h was set to be h=0.14 m. Both u and h matched the experimental setup. The rest of the initial conditions (ICs) and boundary conditions (BCs) were summarized in Table 1. The mesh was chosen to be 30 (horizontal direction) by 50 (vertical direction). A mesh convergence study was done and showed that the mean square differences of $CO_2$ concentration time series at the outlet between a 30 by 50 grid and a 60 by 100 grid was 0.0019 (less than 1.5% of maximum $CO_2$ concentration), and the mean square differences of $CO_2$ concentration time series at the outlet between a 60 by 100 grid and a 120 by 200 grid was 0.0008. In order to balance between accuracy and computational time, the 30 by 50 rectangular mesh was selected. Only the first 720 seconds of the experiment was modeled with CFD, since the $CO_2$ concentration to match experimental data in which $CO_2$ levels reached an equilibrium after 720 seconds. As the simulation time and input $CO_2$ concentration were the same across all simulations, time (x-axis) and $CO_2$ concentration (y-axis) were normalized to be between 0 and 1 for better performance of Bayesian optimization[34]. The time step was selected to be 0.001 seconds. A two-fluid model was chosen as the solver. The under-relaxation factor was set to 1 as the MFiX default.

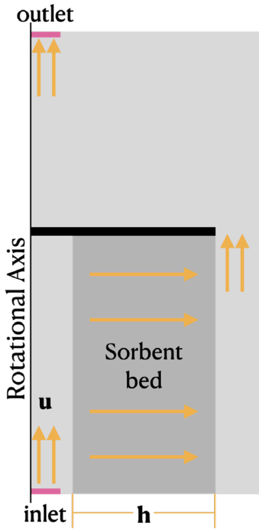


Figure 2: CFD simulation domain of an axisymmetric section of the Jelly roll configuration of Figure 1(a). The darker gray area indicates the Microlith where $CO_2$ is adsorbed. The arrows indicate the approximate airflow path. The barrier (black) obstructs air flow and directs it into the sorbent bed.

*Table 1:* ICs and BCs

| BCs | | ICs | |
|---|---|---|---|
| Left wall | Free-slip wall, adiabatic | Outlet Pressure | 101325 Pa |
| Right wall | No-slip wall, adiabatic | Fluid density | Ideal gas law |
| Bottom wall | No-slip wall, adiabatic | Fluid diffusivity | Diluted mixture approximation (air) |
| Top wall | No-slip wall, adiabatic | Fluid viscosity | Sutherland's law |
| Barrier | No-slip wall, adiabatic | Sorbent bed solid volume fraction | 0.3 |
| Inlet | Mass inflow | Sorbent bed solid density | 420kg/m$^3$ |
| Outlet | Pressure outflow | | |

The bisection method was initially used to determine the k and n values in the Langmuir model represented by Eq. (10).

Once the chemistry (i.e. k and n values) was determined for the CFD, 100 CFD cases were simulated in parallel to predict $CO_2$ concentration over time at the outlet of the reactor for various combinations of inlet velocity (u) and bed thickness (h). The simulation results were then used to train a KRR algorithm to predict $CO_2$ concentration at the outlet for each input pair of u and h.

The trained KRR was then further utilized with Bayesian optimization in a benchmarking case which varied inlet velocity (u) and bed thickness (h) to maximize the breakthrough time (the point at which the $CO_2$ concentration at the outlet reaches 10% of inlet concentration, since 5% to 10% is typically the industry standard for breakthrough). This correlates to the reactor design that would have the greatest capacity for $CO_2$ capture before it needs to be regenerated or replaced.

Finally, KRR and Bayesian optimization were applied on temperature varying data to fine tune the adsorption model by adding desorption into the modeling. The setup to collect temperature varying data was the same as before, except the inlet temperature was controlled to be at 22°C, 30°C, 40°C, and 50°C. Furthermore, the concentration of $CO_2$ at the outlet was measured every second instead of every 40 seconds, matching the changes in the experimental setup done by our industry collaborator, PCI.

## Results and Discussion:

### CFD Modeling of Experimental $CO_2$ Capture at a Single Temperature

$CO_2$ capture on the Microlith was modeled by a Langmuir model (Eqs. (9) and (10)). The reaction rate (k) and the adsorption sites (n) in Eq. (10) were calibrated using the bisection method to match the experimental data (Figure 3). A loss function $\epsilon(k,n)$ was defined to determine how well the simulation fits the experimental data given each k and n,

$$\epsilon(k,n) = \frac{1}{N} \| \vec{y}_{CFD} - \vec{y}_{EXP} \|^2 \qquad (22)$$

where N was the total number of datapoints in the experimental data of $CO_2$ concentration at the outlet, $\vec{y}_{CFD}$ was the CFD simulation time series data of $CO_2$ concentration at the outlet, and $\vec{y}_{EXP}$ was the experimental time series data of $CO_2$ concentration at the outlet.

Therefore, the bisection method was used to minimize the value of $\epsilon(k,n)$ of a domain where $k \in (0,1)$ and $n \in \{1,2,3,4\}$ For each n value, the best fitting k value was chosen after applying bisection method and denoted as $k_n$. Then the best fitting n and $k_n$ pair was chosen to minimize $\epsilon(k,n)$ across all four n values.

The best fitting k and n values were determined to be $k = 0.13$ and $n = 2$ with an $\epsilon(k, n)$ value 0.006. The chemical intuition behind the n value was that each oxygen atom in the $CO_2$ molecule occupies one site.

The best fitting CFD simulation (Figure 3) did not capture the experimental data in the beginning of the experiment, due to the initialization and heating of the bed at the start of the experiment. This problem was addressed in later subsections with more data and a more fine-tuned model. However, the matching of the CFD simulation with the experimental data after the heating phase provided confidence in the CFD model's chemical representation of the experiment and provided an initial bound for the k value as well as the precise n value for more sophisticated modeling for future data sets.

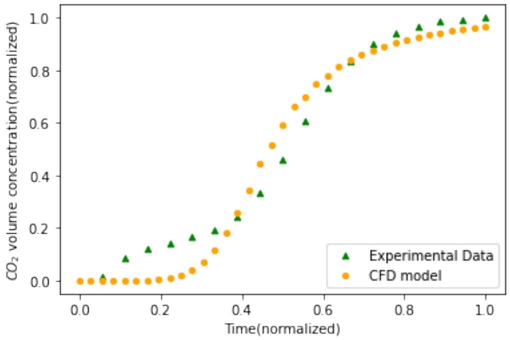


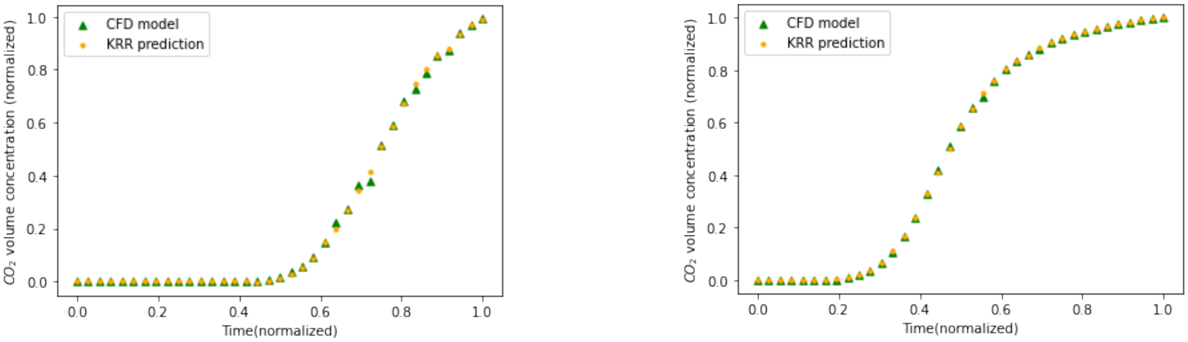
Figure 3: Best fitting result of CFD with the bisection method. The time and $CO_2$ concentration data are normalized.

## KRR Model Training

Once the chemistry was established, 100 CFD cases were simulated in parallel to predict $CO_2$ concentration over time at the outlet of the reactor for various combinations of inlet velocity (u) and bed thickness (h). The bounds of u and h are chosen to be $u \in (0.188\ m/s, 0.277\ m/s)$ and $h \in (0.134\ m, 0.211\ m)$, based on the experimental parameters. The simulation results were then used to train a KRR algorithm to predict $CO_2$ concentration at the outlet. Training was performed on 5% of the data, with the remaining 95% used to generate and evaluate predictions. The loss function was defined as,

$$\epsilon = \frac{1}{N} \left\| \overrightarrow{y'} - \vec{y} \right\|^2 \tag{23}$$

where N was the total number of datapoints within the time series data of $CO_2$ concentration at the outlet, $\overrightarrow{y'}$ was the KRR predicted time series data of $CO_2$ concentration at the outlet, and $\vec{y}$ was the CFD simulated time series data of $CO_2$ concentration at the outlet. The KRR training minimized the loss function $\epsilon$ on the training set.



(a)

(b)

Figure 4: Examples of CFD simulation compared to the KRR prediction trained on 100 CFD simulations where (a) shows the worst fitting case with mean square error 0.0058 and (b) shows the best fitting case with mean square error 0.0017. KRR prediction (yellow dots) was compared to the CFD simulation (green triangle). The axes are normalized.

As seen in Figure 4, the KRR model was able to accurately predict the CFD data. Figure 4 shows two example cases from the 100 scenarios that were run; similar plots can be made for the other 98 CFD simulations. All KRR predictions agreed well with the CFD data, with a mean root-mean-square error of 0.0029, indicating that KRR was able to accurately regress the Langmuir model in the given parameter range. Moreover, each CFD simulation took more than eight hours to complete, while the KRR surrogate model took less than a second. The KRR surrogate model provided an efficient and accurate alternative to CFD for analyzing the effects of changing parameters u and h on $CO_2$ concentration at the outlet. The fast, inexpensive, and accurate predictions of KRR surrogate model enabled fast optimization of the carbon reactor, as shown in the following subsection.

### Optimization of Reactor Design

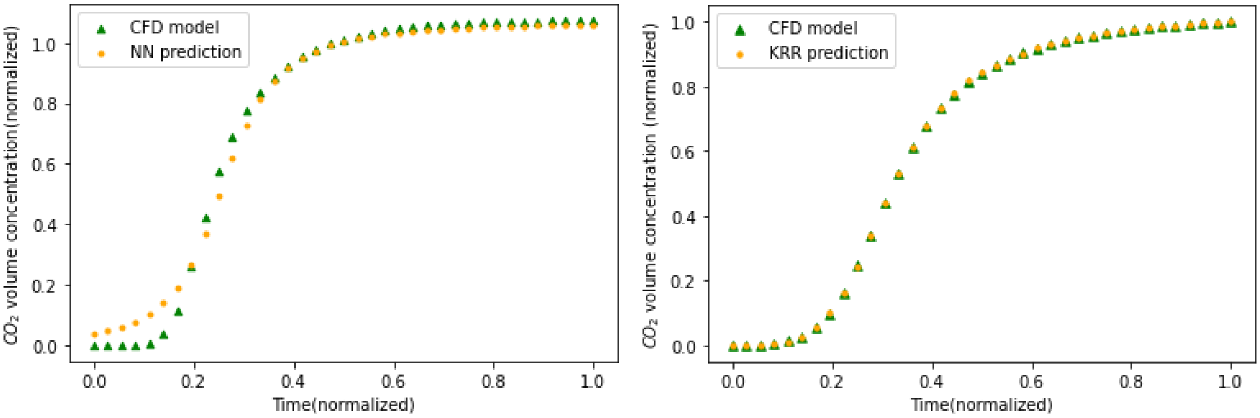
Following KRR training, Bayesian optimization was used in conjunction with KRR to identify the optimal u and h values within bounds $u \in (0.188\ m/s,\ 0.277\ m/s)$ and $h \in (0.134\ m,\ 0.211\ m)$ that would maximize the breakthrough time $t_{break\ through}$. These bounds were chosen based on the KRR training bounds, since KRR is better at interpolation than extrapolation.

The Bayesian optimization on $t_{break\ through}$ served as a benchmark. Due to the simple geometry of the jelly-roll model, it was expected that the thicker the bed, the longer it would take for $CO_2$ to reach the breakthrough threshold since more adsorbent would be in the bed. Meanwhile, the slower the inlet velocity, the longer it would take for $CO_2$ to reach the breakthrough threshold since less reactant was input into the bed each unit time. As a result, given the bound u and h, to optimize $t_{break\ through}$, the lower bound of u (i.e. u=0.188m/s) and upper bound of h (i.e. h=0.211m) needed to be chosen. Therefore, $t_{break\ through}$ could be used as a benchmark to test if Bayesian optimization was capable of finding the optimized design based on the black box KRR surrogate model. The Bayesian optimization result showed the optimal $t_{\text{break through}}$ = 374 seconds was achieve when u = 0.188 m/s and h = 0.201 m, agreeing well with the expected values after only 200 iterations and 10 randomly explored initial points, which boosted the confidence for further usage of Bayesian optimization with KRR in this research.

## Modeling with Temperature Varying Data

The success of KRR with Bayesian optimization in the benchmarking case to maximize $t_{\text{break through}}$ gave confidence for modeling additional $CO_2$ adsorption datasets at four different temperatures. At higher temperatures, it was necessary to account for $CO_2$ desorption from the Microlith. Therefore, the CFD model was expanded to include desorption (Eq. (12)). As was done in the adsorption only case, the reaction parameters were fit to experimental data. Two parameters were set to vary in the CFD models: k (adsorption reaction rate coefficient), and $k_d$ (desorption reaction rate coefficient). In total 231 CFD simulations were run, with the bounds on the two variables $k \in (0.08, 0.4)$ $k_d \in (0, 0.0044)$. The k and $k_d$ values for each simulation were selected by a Bayesian optimization that attempted to fit the experimental data at 22 °C (145 simulations) and 50 °C (86 simulations). Bisection method was not used here because the solution space was two-dimensional and bisection method could only work in one-dimensional space. As the CFD simulations were computationally expensive, Bayesian optimization was not able to find k and $k_d$ values that fit the experiment data well with limited amount of CFD simulation datapoints to explore.

Therefore, KRR and NN were used to regress the CFD simulations, so that KRR and NN could predict similar $CO_2$ concentration time series at the outlet to the CFD simulations but with less than 1% of computational cost. Both KRR and NN used randomly sampled datapoints from all CFD results for training and were tested on the 86 simulations of 50 °C. NN training used 10% of the total data while KRR training used 1% of total data. Figure 5 shows the NN and KRR regressions of CFD results. NN required ten times more data for training, but was much less able to regress the CFD, with a maximum root-mean-square error 0.042, almost 10 times higher than the maximum root-mean-square error of KRR, which was 0.0048. Therefore, KRR was selected as the regression tool for the remainder of the paper given its power of regressing more accurately with only 10% of training data that NN required.

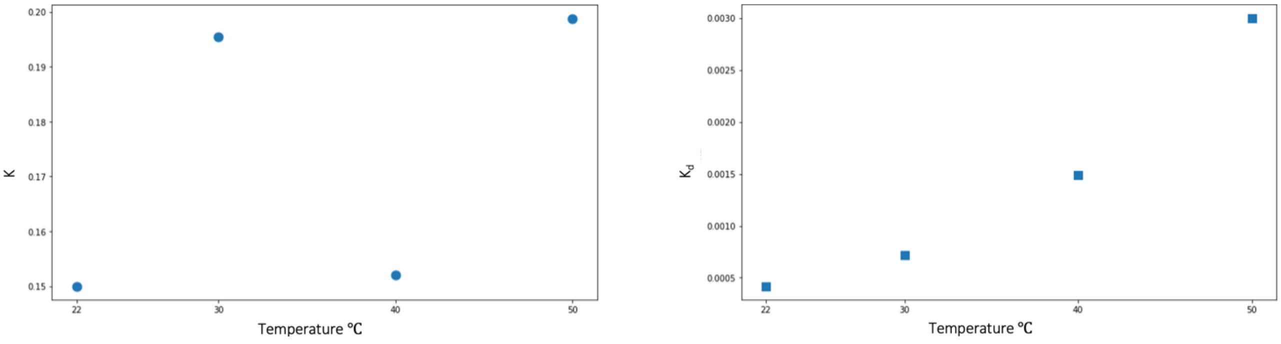


(a) (b)

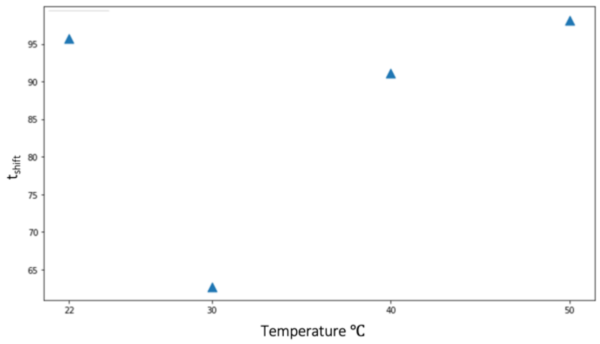
Figure 5: Examples of (a) NN and (b) KRR regressions of CFD. The green triangles are the CFD simulations. The yellow dots are NN prediction in (a) and KRR predations in (b). The axes are normalized.

The initial run of KRR of Bayesian optimization attempted to find the optimal values of three variables k, $k_d$, and another variable $t_{shift}$ to fit the experimental data of all four temperatures. The variable $t_{shift}$ was introduced to account for the initialization of the reactor as it came up to temperature and flow conditions. In comparison, CFD simulations did not require time to reach temperature and flow conditions. Therefore, the KRR simulations learned from CFD were shifted to the right on the time axis by $t_{shift}$, so that the KRR simulations could align better with the experimental data. The bounds of the three variables were set to be $k \in (0.08, 0.4)$, $k_d \in (0, 0.0044)$, and $t_{shift} \in (0\ s, 100\ s)$.

Figure 6 shows the best fitting $k$, $k_d$, and $t_{shift}$ by varying all three parameters for each temperature data. Only $k_d$ showed a clear trend. Both $k$ and $t_{shift}$ showed variances due to the stochastic nature of Bayesian optimization[35]. However, due to the nature of the chemical reaction in the $CO_2$ capture reactor, it was believed that the adsorption reaction rate was much less influenced by temperature, unlike the desorption reaction. Therefore, the average $k$ value of Figure 6(a) was taken as the adsorption reaction rate in future simulations, i.e. $k = 0.175$. Meanwhile, the unknown reaction period in the beginning of the reaction was believed to last a similar amount of time for each experiment. Therefore, the time shift was chosen to be $t_{shift} = 95\ s$.

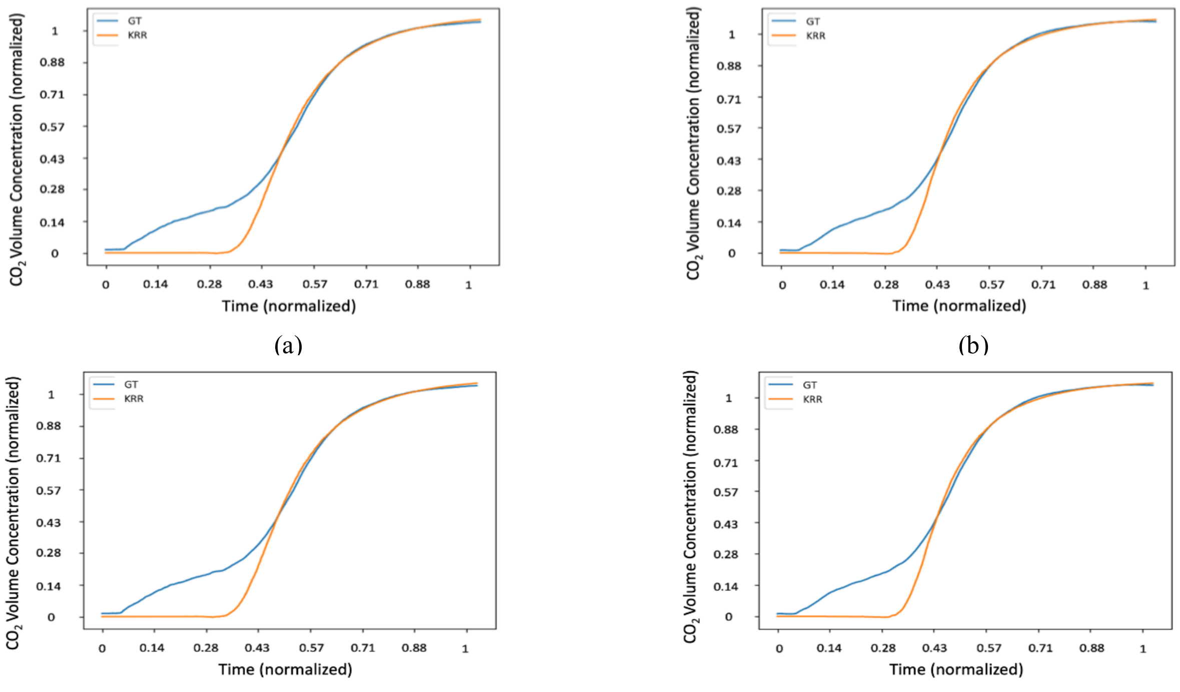


(a) (b)



(c)

Figure 6: Plots of k, $k_d$, and $t_{shift}$ in 22°C, 30°C, 40 °C, and 50 °C. (a) k vs temperature, (b) $k_d$ vs temperature, and (c) $t_{shift}$ vs temperature.



(a) (b)

(c) (d)

Figure 7: Plots of $CO_2$ concentration at the outlet vs time in different temperature. Both vertical and horizontal axes are normalized. The GT legend stands for ground truth, i.e. experimental results. The KRR legend stands for kernel ridge regression. Each plot represents the best fitting of each temperature data. (a)22 °C, (b)30 °C, (c)40 °C, and (d)50 °C.

In the second run of KRR of Bayesian optimization to fit the experimental data of all four temperatures, the values of k and $t_{shift}$ were fixed and only $k_d$ was varied to fit the experimental data for all temperatures. The choice of only varying $k_d$ corresponded with the physical system where the desorption process is only significant at elevated temperatures and would be expected to be a strong function of temperature. Figure 7 shows the fittings after fixing $k$ and $t_{shift}$. The fittings agreed well with the latter half of the experimental data of each temperature, with a maximum $L_2$ error of 0.019 for the latter half of the time series among the regressions for the four temperature data. The algorithm focused on fitting the latter half of the reaction because that is the time period when the reactor environment was steady and the adsorption and desorption reactions were mostly governed by Eq. (10) and Eq. (12).

As a verification, the $k_d$ of 30°C and 50°C were used to fit an Arrhenius equation in Figure 8. The 22°C and 40°C $k_d$ values lied closely to the Arrhenius equation predictions, indicating that likely the Bayesian optimization and KRR captured the physics in the experimental data.

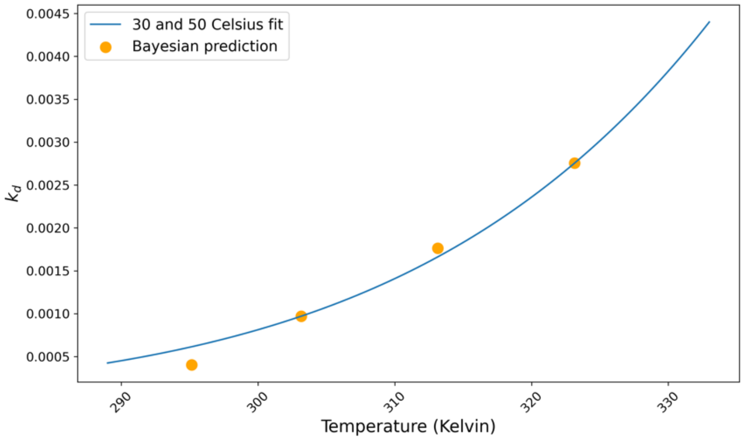


Figure 8: Arrhenius equation fitting based on the best fitting $k_d$ value of the 30 °C and 50 °C data. The blue curve is the Arrhenius fitting and the orange dots are the best fitting $k_d$ values.

## Conclusions:

This research showed the possibility and advantages of combining physics with machine learning to aid DoE. More specifically, this research demonstrated the possibility of using KRR aided CFD to model and optimize carbon capture reactors. With the fast and accurate predictions of a KRR surrogate model, a Bayesian optimizer can be used to optimize the carbon capture reactor design. Apart from optimizing product design, Bayesian optimizers can also be used with CFD to model unknown physical and chemical reactions with the aid of a KRR surrogate model.

The KRR surrogate model was able to learn from only 1% of the CFD training results, and regressed the test set with maximum root-mean-square error of 0.019. Moreover, while each CFD simulation costed about eight hours on 28 CPU-cores to complete, KRR generated each prediction within a second on just 1 CPU-core. This showed the strong capability of KRR to aid CFD after learning. With the fast and accurate prediction power of the KRR surrogate model, Bayesian optimization was used to optimize the reaction bed of the carbon capture reactor. In the benchmarking optimization test, the Bayesian optimizer used the KRR surrogate model to successfully pick the optimal design for the longest breakthrough time. Finally, the Bayesian optimization and KRR algorithms successfully guided CFD to improve modeling of the variable temperature experimental data with desorption added into the CFD simulations, fitting the experimental data with lower error.

To the authors' best knowledge, this research was the first one that compared the capability of NN and KRR to learn from CFD results. For material science and engineering research, it is vital to pick the correct machine learning tool so that the tool balances accuracy, computational cost, ease of use, etc. In the case of this research, KRR was a better tool compared to NN because it was more accurate, required less training data, and was simpler to implement. Selecting KRR as the regression tool made it effective to couple a regression tool with Bayesian optimization to learn and guide the modeling of unknown chemistry in CFD models. In addition, this research was the first to achieve reduced order modeling through exploiting axisymmetric property of the model and KRR. This paradigm of reducing the dimensions of a model and then using a ML algorithm to regress the reduced-dimension simulation results can be applied to other cases as well. The dimension reduction can be achieved through other methods such as thin layer assumption, planar symmetry, etc., and other ML regression tools such as NN, random forest, Gaussian process regression, etc. can replaced KRR.

The success of the machine learning guided CFD modeling in this research demonstrated that machine learning algorithms could be versatile and helpful tools in DoE. As was shown here, when combined with CFD, they could be used for regression to build inexpensive but accurate models. By making predictions budget-friendly, optimization of material properties becomes possible. The paradigm developed in this paper can be widely applied to other types of DoE research, such as battery optimization and catalyst selection. For different applications, other ML methods can replace KRR and Bayesian optimization algorithms, such as random forest and genetic algorithm; other computational methods can replace CFD, such as density functional theory and molecular dynamics. But the procedures for optimization remain the same. Namely, given a small amount of experimental data, apply a computational method to develop a model with or without the aid of ML to model the experimental data, then regress the model with ML (i.e. build a surrogate model) to reduce the computational cost while achieving similar accuracy, and finally use ML or other algorithms to optimize the system based on the ML surrogate model. The ML aided DoE has the potential to greatly reduce temporal and financial costs in the DoE process, making it feasible to conduct research in areas where data acquisition costs are limiting factors.

## Acknowledgements

This work was funded by the National Science Foundation (1932922, 2218729) and DOE through Precision Combustion Incorporated (PCI) (DE-SC0017221). Precision Combustion Inc. provided all experimental data used in this research. The author would like to thank Codruta Loebick and Benjamin Baird from PCI for access to PCI data and many helpful discussions on their carbon capture system.

References:

(1) Franceschini, G.; Macchietto, S. Model-Based Design of Experiments for Parameter Precision: State of the Art. *Chem. Eng. Sci.* **2008**, *63* (19), 4846–4872. https://doi.org/10.1016/j.ces.2007.11.034.

(2) Zhong, W.; Yu, A.; Zhou, G.; Xie, J.; Zhang, H. CFD Simulation of Dense Particulate Reaction System: Approaches, Recent Advances and Applications. *Chem. Eng. Sci.* **2016**, *140*, 16–43. https://doi.org/10.1016/j.ces.2015.09.035.

(3) Maginn, E. J.; Elliott, J. R. Historical Perspective and Current Outlook for Molecular Dynamics As a Chemical Engineering Tool. *Ind. Eng. Chem. Res.* **2010**, *49* (7), 3059–3078. https://doi.org/10.1021/ie901898k.

(4) *Density functional theory for chemical engineering: From capillarity to soft materials - Wu - 2006 - AIChE Journal - Wiley Online Library.* https://aiche.onlinelibrary.wiley.com/doi/full/10.1002/aic.10713 (accessed 2023-03-28).

(5) Frank, M.; Drikakis, D.; Charissis, V. Machine-Learning Methods for Computational Science and Engineering. *Computation* **2020**, *8* (1), 15. https://doi.org/10.3390/computation8010015.

(6) Keith, J. A.; Vassilev-Galindo, V.; Cheng, B.; Chmiela, S.; Gastegger, M.; Müller, K.-R.; Tkatchenko, A. Combining Machine Learning and Computational Chemistry for Predictive Insights Into Chemical Systems. *Chem. Rev.* **2021**, *121* (16), 9816–9872. https://doi.org/10.1021/acs.chemrev.1c00107.

(7) Thebelt, A.; Wiebe, J.; Kronqvist, J.; Tsay, C.; Misener, R. Maximizing Information from Chemical Engineering Data Sets: Applications to Machine Learning. *Chem. Eng. Sci.* **2022**, *252*, 117469. https://doi.org/10.1016/j.ces.2022.117469.

(8) Frazier, P. I.; Wang, J. Bayesian Optimization for Materials Design. In *Information Science for Materials Discovery and Design*; Lookman, T., Alexander, F. J., Rajan, K., Eds.; Springer Series in Materials Science; Springer International Publishing: Cham, 2016; pp 45–75. https://doi.org/10.1007/978-3-319-23871-5_3.

(9) Greenhill, S.; Rana, S.; Gupta, S.; Vellanki, P.; Venkatesh, S. Bayesian Optimization for Adaptive Experimental Design: A Review. *IEEE Access* **2020**, *8*, 13937–13948. https://doi.org/10.1109/ACCESS.2020.2966228.

(10) Vahid, A.; Rana, S.; Gupta, S.; Vellanki, P.; Venkatesh, S.; Dorin, T. New Bayesian-Optimization-Based Design of High-Strength 7xxx-Series Alloys from Recycled Aluminum. *JOM* **2018**, *70* (11), 2704–2709. https://doi.org/10.1007/s11837-018-2984-z.

(11) Ando, T.; Shimizu, N.; Yamamoto, N.; Matsuzawa, N. N.; Maeshima, H.; Kaneko, H. Design of Molecules with Low Hole and Electron Reorganization Energy Using DFT Calculations and Bayesian Optimization. *J. Phys. Chem. A* **2022**, *126* (36), 6336–6347. https://doi.org/10.1021/acs.jpca.2c05229.

(12) Park, S.; Na, J.; Kim, M.; Lee, J. M. Multi-Objective Bayesian Optimization of Chemical Reactor Design Using Computational Fluid Dynamics. *Comput. Chem. Eng.* **2018**, *119*, 25–37. https://doi.org/10.1016/j.compchemeng.2018.08.005.

(13) Zhang, Y.; Apley, D. W.; Chen, W. Bayesian Optimization for Materials Design with Mixed Quantitative and Qualitative Variables. *Sci. Rep.* **2020**, *10* (1), 4924. https://doi.org/10.1038/s41598-020-60652-9.

(14) Ge, H.; Bakir, A. H.; Yadav, S.; Kang, Y.; Parameswaran, S.; Zhao, P. CFD Optimization of the Pre-Chamber Geometry for a Gasoline Spark Ignition Engine. *Front. Mech. Eng.* **2021**, *6*.

(15) Abdi, R.; Krzaczek, M.; Tejchman, J. Comparative Study of High-Pressure Fluid Flow in Densely Packed Granules Using a 3D CFD Model in a Continuous Medium and a Simplified 2D DEM-CFD Approach. *Granul. Matter* **2021**, *24* (1), 15. https://doi.org/10.1007/s10035-021-01179-2.

(16) Vegini, A. A.; Meier, H. F.; Iess, J. J.; Mori, M. Computational Fluid Dynamics (CFD) Analysis of Cyclone Separators Connected in Series. *Ind. Eng. Chem. Res.* **2008**, *47* (1), 192–200. https://doi.org/10.1021/ie061501h.

(17) Li, T.; Gel, A.; Pannala, S.; Shahnam, M.; Syamlal, M. CFD Simulations of Circulating Fluidized Bed Risers, Part I: Grid Study. *Powder Technol.* **2014**, *254*, 170–180. https://doi.org/10.1016/j.powtec.2014.01.021.

(18) Li, T.; Pannala, S.; Shahnam, M. CFD Simulations of Circulating Fluidized Bed Risers, Part II, Evaluation of Differences between 2D and 3D Simulations. *Powder Technol.* **2014**, *254*, 115–124. https://doi.org/10.1016/j.powtec.2014.01.022.

(19) Masoumi, A. P.; Tajalli-Ardekani, E.; Golneshan, A. A. Investigation on Performance of an Asphalt Solar Collector: CFD Analysis, Experimental Validation and Neural Network Modeling. *Sol. Energy* **2020**, *207*, 703–719. https://doi.org/10.1016/j.solener.2020.06.045.

(20) Jinnouchi, R.; Asahi, R. Predicting Catalytic Activity of Nanoparticles by a DFT-Aided Machine-Learning Algorithm. *J. Phys. Chem. Lett.* **2017**, *8* (17), 4279–4283. https://doi.org/10.1021/acs.jpclett.7b02010.

(21) Ulissi, Z. W.; Medford, A. J.; Bligaard, T.; Nørskov, J. K. To Address Surface Reaction Network Complexity Using Scaling Relations Machine Learning and DFT Calculations. *Nat. Commun.* **2017**, *8* (1), 14621. https://doi.org/10.1038/ncomms14621.

(22) Kochkov, D.; Smith, J. A.; Alieva, A.; Wang, Q.; Brenner, M. P.; Hoyer, S. Machine Learning–Accelerated Computational Fluid Dynamics. *Proc. Natl. Acad. Sci.* **2021**, *118* (21), e2101784118. https://doi.org/10.1073/pnas.2101784118.

(23) Vinuesa, R.; Brunton, S. L. Enhancing Computational Fluid Dynamics with Machine Learning. *Nat. Comput. Sci.* **2022**, *2* (6), 358–366. https://doi.org/10.1038/s43588-022-00264-7.

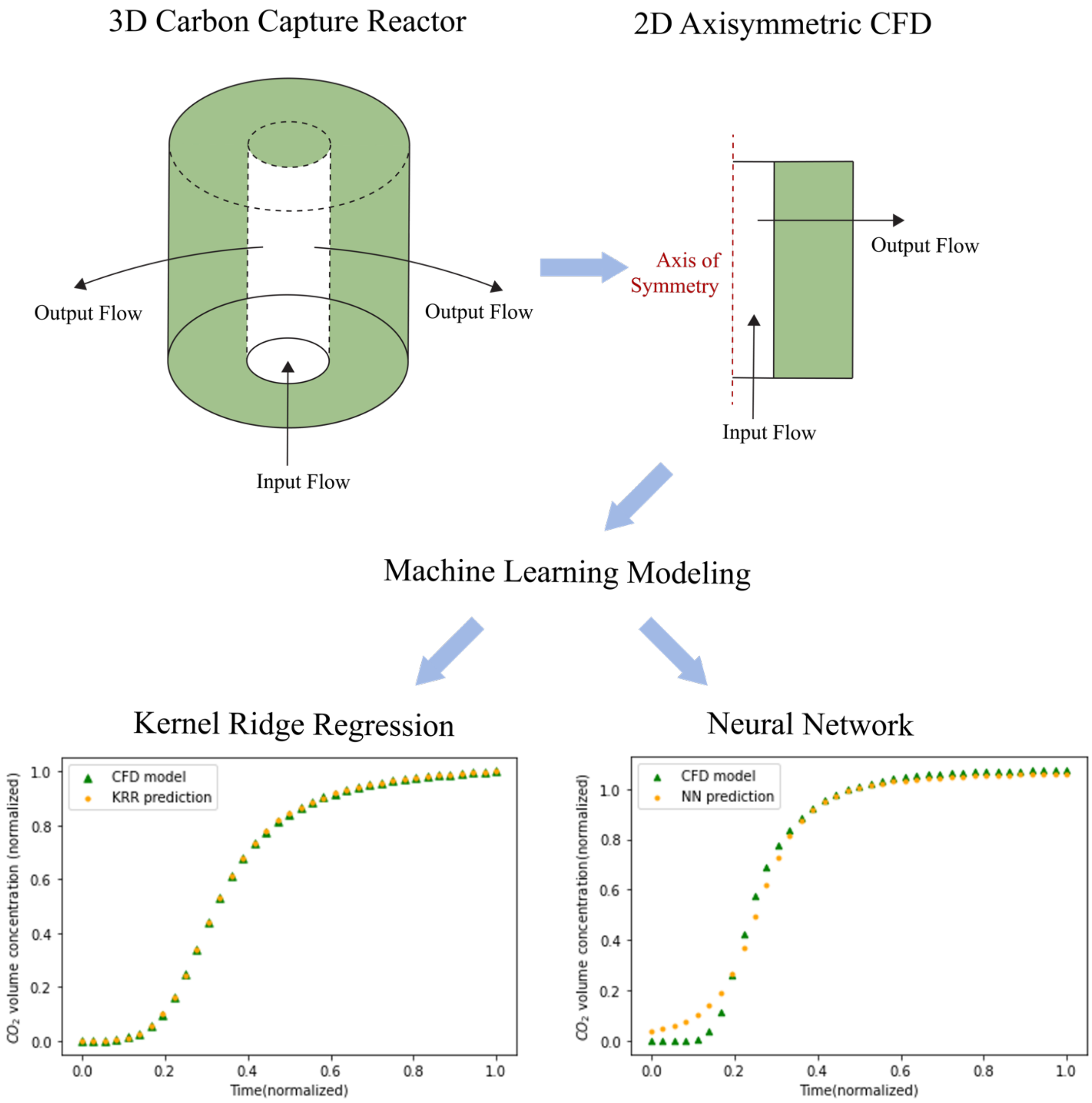
(24) Eldred, M.; Dunlavy, D. Formulations for Surrogate-Based Optimization with Data Fit, Multifidelity, and Reduced-Order Models. In *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*; American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2006-7117.

(25) Peitz, S.; Dellnitz, M. A Survey of Recent Trends in Multiobjective Optimal Control—Surrogate Models, Feedback Control and Objective Reduction. *Math. Comput. Appl.* **2018**, *23* (2), 30. https://doi.org/10.3390/mca23020030.

(26) Owoyele, O.; Pal, P.; Vidal Torreira, A.; Probst, D.; Shaxted, M.; Wilde, M.; Senecal, P. K. Application of an Automated Machine Learning-Genetic Algorithm (AutoML-GA) Coupled with Computational Fluid Dynamics Simulations for Rapid Engine Design Optimization. *Int. J. Engine Res.* **2022**, *23* (9), 1586–1601. https://doi.org/10.1177/14680874211023466.

(27) Hedin, N.; Andersson, L.; Bergström, L.; Yan, J. Adsorbents for the Post-Combustion Capture of CO2 Using Rapid Temperature Swing or Vacuum Swing Adsorption. *Appl. Energy* **2013**, *104*, 418–433. https://doi.org/10.1016/j.apenergy.2012.11.034.

(28) *MFIX*. NETL Multiphase Flow Science. https://mfix.netl.doe.gov/products/mfix/ (accessed 2023-03-28).
(29) Vovk, V. Kernel Ridge Regression. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik*; Schölkopf, B., Luo, Z., Vovk, V., Eds.; Springer: Berlin, Heidelberg, 2013; pp 105–116. https://doi.org/10.1007/978-3-642-41136-6_11.
(30) Soch, J. *Conditional distributions of the multivariate normal distribution*. The Book of Statistical Proofs. https://statproofbook.github.io/P/mvn-cond.html (accessed 2023-03-28).
(31) Agnihotri, A.; Batra, N. Exploring Bayesian Optimization. *Distill* **2020**, *5* (5), e26. https://doi.org/10.23915/distill.00026.
(32) Backprop.Pdf. https://www.cs.cornell.edu/courses/cs5740/2016sp/resources/backprop.pdf (accessed 2024-02-13).
(33) Thapliyal, P.; Tomar, K. Role of Bisection Method. *Int. J. Comput. Appl. Technol. Res.* **2014**, *3* (8).
(34) Singh, D.; Singh, B. Investigating the Impact of Data Normalization on Classification Performance. *Appl. Soft Comput.* **2020**, *97*, 105524. https://doi.org/10.1016/j.asoc.2019.105524.
(35) Frazier, P. I. A Tutorial on Bayesian Optimization. arXiv July 8, 2018. https://doi.org/10.48550/arXiv.1807.02811.


3D Carbon Capture Reactor
2D Axisymmetric CFD
Output Flow
Output Flow
Input Flow
Axis of Symmetry
Output Flow
Input Flow
Machine Learning Modeling
Kernel Ridge Regression
Neural Network
CFD model
KRR prediction
$CO_2$ volume concentration (normalized)
0.0
0.2
0.4
0.6
0.8
1.0
Time(normalized)
CFD model
NN prediction
$CO_2$ volume concentration(normalized)
Time(normalized)