



Managing uncertainty in data-driven simulation-based optimization

Gordon Hüllen, Jianyuan Zhai, Sun Hye Kim, Anshuman Sinha, Matthew J. Realff, Fani Boukouvala*

Georgia Institute of Technology, School of Chemical & Biomolecular Engineering, 311 Ferst Dr., Atlanta, GA, 30332, USA

ARTICLE INFO

Article history:

Received 24 February 2019

Revised 15 July 2019

Accepted 17 July 2019

Available online 29 August 2019

Keywords:

Surrogate modeling
Simulation optimization
Direct air capture
Neural networks
Polynomial interpolation

ABSTRACT

Optimization using data from complex simulations has become an attractive decision-making option, due to ability to embed high-fidelity, non-linear understanding of processes within the search for optimal values. Due to lack of tractable algebraic equations, the link between simulations and optimization is oftentimes a surrogate metamodel. However, several forms of uncertainty exist within the cycle that links simulation data, to metamodels, to optimization. Uncertainty may originate from parameters of the simulation, or the form and fitted parameters of the metamodel. This paper reviews different literatures that are relevant to surrogate-based optimization and proposes different strategies for handling uncertainty, by combining machine learning with stochastic programming, robust optimization, and discrepancy modeling. We show that incorporating uncertainty management within simulation-based optimization leads to more robust solutions, which protect the decision-maker from infeasible solutions. We present the results of our proposed approaches through a case study for direct-air capture through temperature swing adsorption.

© 2019 Published by Elsevier Ltd.

1. Introduction

Rapid developments in computational capabilities and numerical methods have led to substantial advances in computer-aided decision-making using simulations in many areas of chemical engineering, including Process Systems Engineering (PSE). Our advancing ability to simulate phenomena from a molecular level, all the way to the process and enterprise levels has motivated many researchers to use multiscale models for design, optimization, control and operations. Detailed simulations include large systems of ordinary or partial differential equations, computational fluid dynamic simulations, discrete element method models or even Monte-Carlo simulations, and many more (Amaran et al., 2014; Boukouvala et al., 2017; Chuang et al., 2018; Cozad et al., 2014; Dias et al., 2018; Dowling et al., 2014; Eslick et al., 2014; Lucidi et al., 2016; Ma et al., 2018; Marques et al., 2017; Palmer and Realff, 2002; Wang and Ierapetritou, 2017). Simulations are used to model multiphase or particle, multicomponent flows within complex geometries, oftentimes under dynamically changing initial or boundary conditions. However, as the complexity of simulations increases, this creates a new challenge, namely the inability to directly optimize systems with commercially available deterministic optimization

solvers (Amaran et al., 2014; Dowling et al., 2014; Eason and Biegler, 2016). This challenge has led to an increasing interest in the area of data-driven optimization techniques that search for globally or locally optimal solutions in the absence of the true derivatives of the original model.

In the past decade, data-driven optimization (DDO) applications are becoming common in many engineering fields (Anna et al., 2017; Boukouvala et al., 2017; Ibrahim et al., 2018; Lucidi et al., 2016; Ma et al., 2018; Marques et al., 2017; Na et al., 2017; Negrellos-Ortiz et al., 2016; Quirante and Caballero, 2016; Rossger and Richter, 2018; Wang et al., 2017; Zadeh et al., 2016; Zhong et al., 2019). Specifically, in chemical engineering, DDO has been applied to carbon capture systems (Eslick et al., 2014), pressure swing adsorption (Boukouvala et al., 2017), ammonia production process simulation (Palmer and Realff, 2002), reactive flow systems (Rossger and Richter, 2018), cryogenic air separation (Negrellos-Ortiz et al., 2018), crude-oil distillation (Ibrahim et al., 2018), continuous pharmaceutical manufacturing modeling (Boukouvala and Ierapetritou, 2013; Marques et al., 2017; Wang et al., 2017), and many more. In these applications, closed-form or explicit algebraic relationships and derivatives are either unavailable or impractical to obtain. There are many different types of data-driven optimization methods, and several recent review articles provide very comprehensive comparisons (Audet and Kokkolaras, 2016; Bhosekar and Ierapetritou, 2018; Boukouvala et al., 2016; Rios and Sahinidis, 2013). One of the most popular categories of data-driven optimization is referred to as “model-based” or

* Corresponding author.

E-mail address: fani.boukouvala@chbe.gatech.edu (F. Boukouvala).

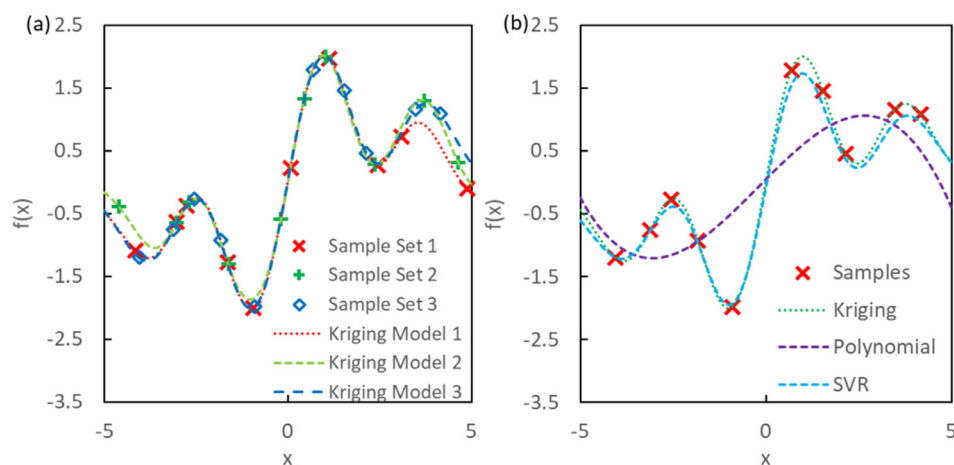


Fig. 1. (a) Curves of the same surrogate model (Kriging) trained with different samples and (b) curves of different surrogate model types (Kriging, Polynomial, Support Vector Regression) trained with the same deterministic sample points.

“surrogate-based”. This category includes a critical step after data collection, namely the training of intermediate surrogate models that serve as the computationally cheap algebraic approximations of the important input-output relationships embedded within the optimization problem. This paper will focus on surrogate-based optimization techniques that have become very popular in the recent literature, which is also due to the increasing interest in Machine Learning in engineering.

There are currently many open challenges in surrogate-based optimization. These include: (a) the selection of the most efficient sampling strategy; (b) the selection of the best surrogate model; (c) the efficient training and validation of the surrogate models; and finally (d) the development of efficient algorithms that adaptively sample-fit-validate data, to find the best optimum with minimum computational cost and sampling requirements. The literature is split between the use of local model-based methods, such as trust-region methods, e.g. Powell’s method (Powell, 2002), and global model-based methods, such as the Sequential Design for Optimization (Cox and John, 1992) and Efficient Global Optimization (Jones et al., 1998) methods. Both classes of methods use local or global surrogate models to expedite the search for an optimum. Two recent review articles outline the differences between the state-of-the-art methods on this topic (Audet and Kokkolaras, 2016; Boukouvala et al., 2016). Undoubtedly, no method has been proven to significantly outperform all others across a large set of benchmark problems, while the performance of each method is highly dependent on (a) the problem characteristics, (b) the amount and quality of data available, and (c) the selected surrogate modeling technique. Computational studies of the current existing data-driven optimization software have verified that the performance is problem dependent (Amaran et al., 2014; Boukouvala and Ierapetritou, 2013; Rios and Sahinidis, 2013). Especially when surrogate models are employed for optimization, researchers have devoted great effort on finding the best surrogate model (Boukouvala and Floudas, 2017; Cozad et al., 2014; Eason and Cremaschi, 2014; Garud et al., 2018; Garud et al., 2017, 2018; Strau and Skogestad, 2017; Wilson and Sahinidis, 2017); however, the presence of uncertainty and limitations in sampling make this a very challenging problem. Specifically, training one type of surrogate model with samples collected from different sampling strategies may result in slightly different realizations of the surrogate model (Fig. 1a). Even in the situation where the same samples are used to train different types of surrogate models, the realizations may also vary from each other significantly (Fig. 1b).

Although there is a very prolific open debate on the development of the best strategy for sampling and surrogate model-

ing and comparison of methods that manage to get closest to the best-known global optimum, there has been less discussion on the effects of uncertainty when using surrogate-based optimization algorithms. Recently, Wang et al. used the smoothing effects of Gaussian process metamodels to optimize simulations with homoscedastic input uncertainty (Wang et al., 2018), while Wang et al. used a modified Expected Improvement criterion and Kriging metamodels to optimize a stochastic simulation of a pharmaceutical production system (Wang and Ierapetritou, 2018). Recent work on Bayesian optimization under uncertainty has shown that by conditioning the prior distribution on the observations, a Bayesian approach can be used to locate robust solutions of sample-based systems (Beland and Nair, 2017; Bogunovic et al., 2018). A local stochastic trust-region based algorithm was recently presented by Shashaani et al. (2018), which aims to locate local optima of unconstrained optimization problems subject to noise. Similarly, Audet et al. modified their mesh adaptive direct-search algorithm to handle noisy black-box data, in order to robustly converge to locally optimal solutions (Audet et al., 2018). Eslick et al. present a framework that aims to perform uncertainty quantification, sensitivity analysis and optimization of carbon capture systems through a user-friendly interface (Eslick et al., 2014). Bertsimas et al. have developed a local gradient-descent optimization algorithm to identify locally robust solutions of unconstrained simulation based optimization problems (Bertsimas et al., 2010b), or constrained problems with known functional forms (Bertsimas et al., 2010a).

In this work, we aim to highlight the importance of considering uncertainty when performing surrogate-based simulation-based optimization, which is a topic rarely discussed in the surrogate-based optimization literature. We focus our work on techniques that are applicable for simulation-based problems for which prior distribution assumptions, or knowledge of the simulation model equations are not available. Specifically, first we ask the question: Can simulation-based data-driven optimization be treated as a deterministic problem in engineering case studies? We argue that there are several forms of uncertainty that are embedded within data-driven optimization problems that need to be considered, and we show this through a simulation-based case study.

First, one of the most commonly referred to sources of uncertainty in simulation-based optimization is caused by numerical methods and numerical precision. However, this form of uncertainty is negligible in state-of-the-art computer systems and software. A second major source of uncertainty is introduced from the uncertain simulation parameters that are estimated using experimental data (e.g., mass transfer coefficients, kinetic parameters, etc.). This form of uncertainty is often not considered in bench-

marking of surrogate-based optimization solvers, treating the data-generating simulation as a deterministic model. Finally, even if the simulation is deterministic, a last form of uncertainty is introduced during the selection of the form and the training of the parameters of the surrogate model. An example of this type of uncertainty is shown in Fig. 1, where in (a) slightly different samples are used to fit the same type of surrogate model (i.e., Kriging) and in (b) the same samples are used to fit different types of surrogate models (i.e., Kriging, polynomial and Support Vector Regression (SVR)). Through this simple example, it becomes evident that small variations in sample locations, and the choice of the surrogate model, introduce uncertainty in the final output predictions of the meta-model, and would potentially lead to different optimal solutions.

All three forms of this uncertainty may not be present in a specific application simultaneously, however, in a surrogate-based optimization approach the latter form of uncertainty is always introduced. As a result, we argue that uncertainty is unavoidably embedded within the problem of data-driven surrogate-based optimization. We show this through a case study that suffers from two forms of uncertainty: (a) the simulation is stochastic due to the presence of an uncertain parameter, and (b) the best form and parameters of the surrogate model to represent the simulation are unknown. The case study discusses direct air capture (DAC) by passing air through solid adsorbents inside a monolithic structure and removing the CO₂ through pressure swing adsorption. The DAC simulation model outputs are affected by an estimated mass transfer coefficient parameter.

Finally, we present several ideas for handling uncertainty when using surrogates for optimization that are general enough to be adapted to different metamodeling techniques. The techniques proposed in this work use existing concepts from the robust and stochastic optimization and discrepancy modeling literatures. The main contribution of this paper is two-fold: (a) this paper serves as a brief review and comparison of different literatures that are rarely discussed together, namely surrogate-based optimization, robust and stochastic optimization and discrepancy modeling, and (b) the main novelty of this paper is the adaptation and compre-

hensive comparison of existing concepts from diverse fields within a surrogate-based optimization framework under uncertainty. Specifically, we employ a Robust Counterpart (RC) approach to develop “Robust Surrogates” from an ensemble of deterministic surrogates and compare this approach with a sample average approximation formulation, as well as a discrepancy modeling approach. We discuss the advantages and disadvantages of different types of surrogate modeling and uncertainty handling methods. Overall, we show that the obtained robust solutions are more conservative, but maintain feasibility at different realizations of the uncertain parameters, which is not the case for deterministic solutions.

The remainder of the paper is structured as follows. First, the motivating case study is described in the Motivating Case Study: Design of Direct-Air Capture section. Following, we discuss sampling and surrogate modeling methods (Surrogate Modeling Methods) and focus on two types that were used in this work, namely polynomial approximations and artificial neural networks. Next, three ideas for handling uncertainty within a surrogate-based optimization framework are discussed in section Methods for Handling Uncertainty. The results are then presented for several combinations of surrogate models and uncertainty management methods and all solutions are validated against the actual simulation (Results). We conclude the paper with some discussion and future perspectives (Conclusions).

2. Motivating case study: design of direct-air capture

Our direct air capture (DAC) design aims at removing CO₂ directly from ambient air using solid adsorbents via pressure swing adsorption. The solid adsorbent is coated inside monolithic channel in the form of films. For this case study, we have used a metal organic framework (MOF), MIL-101(Cr)-PEI-800, as the solid adsorbent. Details about the isotherm behavior of this adsorbent at ambient CO₂ conditions (~ 400 ppm) can be found in [Sinha et al. \(2017\)](#). The Direct Air Capture process is performed in two steps: adsorption and desorption, as shown in Fig. 2. During the adsorption step, ambient air is passed through the front end of the chan-

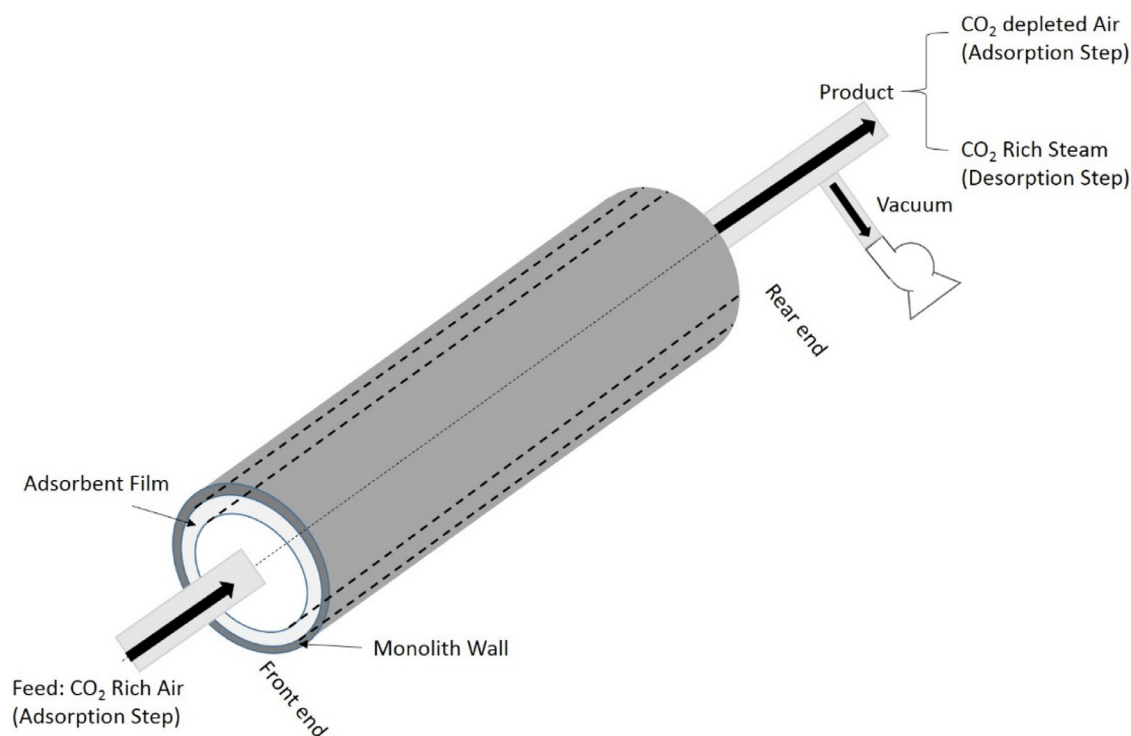


Fig. 2. Schematic of direct air capture process.

nel with the help of blowers. CO₂ adsorbs in the MOF film as the air flows inside the channel and CO₂ depleted air is emitted from the rear end of the channel. During the desorption step, the pressure from the rear end of the channel is decreased below atmospheric pressure with the help of vacuum pump and the front end of the channel is closed. After this brief depressurization, steam is then used to heat the channel, this causes CO₂ to desorb from the adsorbent. The steam also sweeps the CO₂ down the channel lowering its partial pressure due to steam dilution and eventually removing it from the channel.

The DAC design is modeled by formulating coupled heat and mass balance equations governing the concentration and temperature dynamics of the different process components. The detailed governing equations of this model have been presented in [Sinha et al. \(2017\)](#) and are included in the Supplementary Information of this paper. The model tracks the gas phase concentration of three adsorbates, CO₂, N₂ and O₂. The distribution of temperature within the gas, adsorbent and monolithic wall are also modeled through the heat balance. The velocity of the gas inside the channel is modeled using the Hagen Poiseuille equation. Specific boundary and initial conditions for the adsorption and desorption steps are required to model the operation of the temperature swing adsorption process. The rate of adsorption of CO₂ is approximated by a linear driving force model, where the equilibrium concentration of adsorbed CO₂ is determined by isotherm equations. The rate of adsorption also depends on the mass transfer coefficient k (h⁻¹) that needs to be estimated. Determination of mass transfer coefficient requires estimation of adsorbent film thickness and effective diffusivity of CO₂ inside the adsorbent. Adsorbent film thickness determination involves uncertainty due to irregular adsorbent film coating. Similarly, determination of effective diffusivity introduces sources of uncertainty due to human errors in experimental runs and inherently through the orientation and thickness of MOF crystals on the adsorbent surface. Hence, mass transfer coefficient is an uncertain parameter in the DAC model.

In this case study, we have used adsorption time (t_{ads}) and desorption time (t_{des}) as inputs and productivity, recovery, purity and energy as the output design variables. Definition of the output design variables are provided in [Eqs. \(1\)–\(4\)](#).

$$\text{Productivity} = \frac{\text{Amount of CO}_2 \text{ recovered as product (mol)}}{\text{mass of adsorbent (kg)} \times \text{total time (h)}} \quad (1)$$

$$\text{Recovery} = \frac{\text{Amount of CO}_2 \text{ recovered as product (kg)}}{\text{Amount of CO}_2 \text{ fed as input (kg)}} \quad (2)$$

$$\text{Purity} = \frac{\text{Moles of CO}_2 \text{ recovered as product}}{\text{Total moles of gas recovered as product}} \quad (3)$$

$$\text{Energy} = E_{\text{Blower}} + E_{\text{Vacuum}} \quad (4)$$

where E_{Blower} is the energy consumed in the blowers that push the ambient air inside the monolithic channel during adsorption. E_{Vacuum} is the energy required to operate the vacuum pump during desorption.

[Fig. 3](#) shows the process flow for the DAC simulation. The input variables (t_{ads} and t_{des}) are fed to the model to obtain the out-

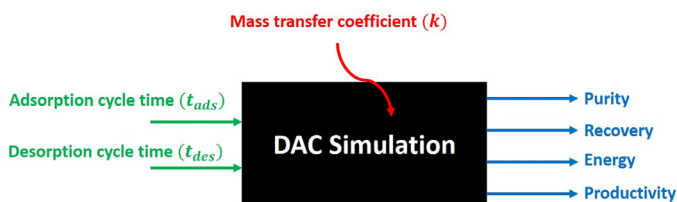


Fig. 3. Process flow showing input and output variables along with uncertain parameter.

put values of productivity, recovery, purity and energy at varying mass transfer coefficient, k . The model is implemented in gPROMS ("gPROMS," 1997–2018).

Based on the above, the optimization problem we propose to solve in this work is as follows:

$$\begin{aligned} &\max \text{Productivity}(t_{ads}, t_{des}, k) \\ &\text{subject to:} \\ &\text{Purity}(t_{ads}, t_{des}, k) \geq 95 \\ &\text{Recovery}(t_{ads}, t_{des}, k) \geq 15 \\ &\text{Energy}(t_{ads}, t_{des}, k) \leq 0.2 \\ &20 \leq t_{ads}, t_{des} \leq 250 \end{aligned} \quad (P1)$$

All of the equations in formulation (P1) rely on the simulation outlined above. In addition, all of the constraints and the objective function are expected to be highly nonlinearly correlated with the two decision variables and the uncertain parameter. One approach to optimize this system would be through discretization of the system of equations and the formulation of a large nonlinear programming problem. However, in this work we will follow a surrogate-based optimization approach, based on which we will collect data from the simulation and approximate the objective function and all of the constraints in P1 with surrogate models.

3. Surrogate modeling methods

In this section, we discuss the problem of fitting a surrogate model $f^{(m)}(b^{(m)}, x)$, where $x \in \mathbb{R}^D$ and $b^{(m)}$ is the vector of parameters fitted to a set of $X - Y_m$ input-output data. X is a $[N \times D]$ matrix of $n = 1, \dots, N$ observations in $d = 1, \dots, D$ dimensions. Y is a $[N \times M]$ matrix of $m = 1, \dots, M$ outputs measured at the samples in X , and Y_m is the vector of matrix Y corresponding to output m . Each function for each output m may differ in the form of f^m and the parameter vector b^m .

There are many different types of surrogate models or meta-models that have been used to approximate data from computer simulations. The types of surrogate models can be broadly categorized into: (a) explicit regression functions with functional forms that are fixed a-priori (e.g., linear, quadratic and generalized linear regression) and (b) implicit regression functions that are based on linear or nonlinear kernel transformations, such as Gaussian Process Models, Neural Networks, Support Vector Regression and more ([Hastie et al., 2009](#)). Although all surrogate models have parameters that need to be fitted based on observed data, the first category requires that one predefines the terms of the regression function, while the second category includes generic nonlinear universal approximators, whose structure and model parameters don't have a physical meaning and could change depending on the amount of data available.

This work does not aim to perform a thorough comparison between all different types of surrogate functions, as this has been done in recent work on a variety of problems ([Bhosekar and Ierapetritou, 2018](#); [Davis et al., 2018](#); [Garud et al., 2018, 2018](#)). In this work, we employ two different types of surrogate models that have convergent approximating qualities: (a) Neural networks, and (b) Sparse-Grid polynomial interpolation models. For each type, we use well established sampling strategies, training and validation procedures, to ensure that we identify a surrogate model structure and the optimal parameters that predict the collected data with minimized validation error. We compare the performance of these two different types of models and then employ various strategies to formulate surrogate-based optimization under uncertainty problems.

3.1. Polynomial approximations based on sparse grids

One approach for surrogate modeling is polynomial interpolating models using a set of fixed samples that lie on a Sparse Grid, or else a Smolyak Grid (Smolyak, 1963). The Sparse Grid and polynomial fitting method has received significant attention in the integration (Bungartz and Dirnstorfer, 2003, 2004; Dung, 2016; Gerstner and Griebel, 1998; Peherstorfer et al., 2015; Tang et al., 2016), and approximation theory (Gajda, 2005; Plaskota and Wasilkowski, 2004; Xu, 2015) literatures. One of the strengths of this approach is the existence of convergent approximation error bounds as the number of samples increases under mild smoothness assumptions (Waskowski and Wozniakowski, 1995). In fact, it has been shown that interpolating functions fitted using these grids converge to the true black-box function with a more tractable rate and a weaker dependence on dimensionality. These approximations have recently gained some popularity in the field of surrogate-based optimization (Grimstad and Sandnes, 2016; Hulsman and Reith, 2013; Kieslich et al., 2018; Novak and Ritter, 1996; Valentin and Pfluger, 2016).

The first decision in building a SG is the selection of the sampling basis points, which are roots or extrema of the orthogonal polynomials used to build the final surrogate function. In this work we use extrema of Chebyshev polynomials as our basis points. Once the basis is selected, a multidimensional grid can be constructed as a tensor product of one-dimensional basis points. A multidimensional SG can be constructed for different degrees of polynomial exactness, or else approximation level (μ). As μ increases, more points are added to the grid (Fig. 4); the size of the interpolating polynomial terms and parameters increases; as does its approximation accuracy. One very important property of SGs is their 'nesting' nature, which means that every level of approximation contains all of the points of the previous level (Fig. 4). Finally, in order to fit the multidimensional polynomial based on the level and grid points collected, Lagrange interpolation is used to identify the parameters of the surrogate function. Details about how to construct these grids and surrogate functions has been documented in the literature, and algorithmic packages exist for their

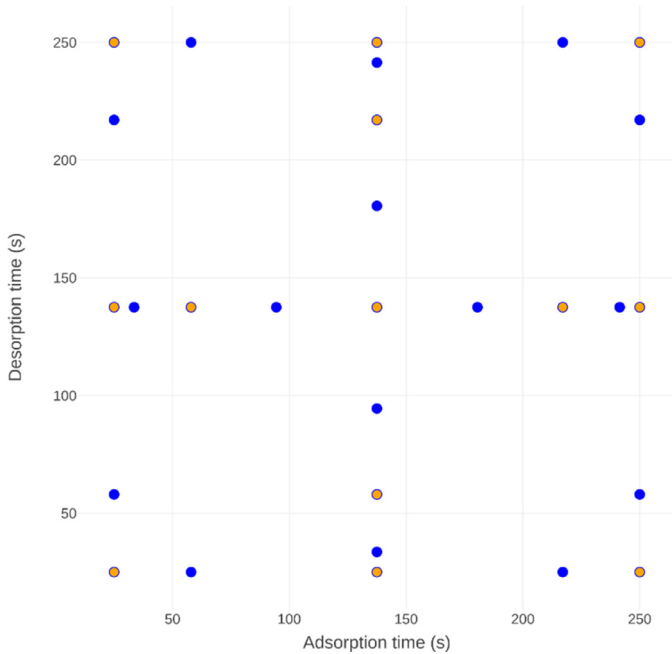


Fig. 4. Sparse grid designs for up to level $\mu = 3$. Orange points represent points of prior level $\mu = 2$, blue points and orange points are part of level $\mu = 3$ (Eq. (5)). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

generation (Harding, 2016; Judd et al., 2014; Kieslich et al., 2018; Novak et al., 2010; Peherstorfer et al., 2015; Valentin and Pfluger, 2016).

Unlike many other types of popular machine learning methods for development of approximations, this approach has no flexibility on the sampling scheme and functional form. As a result, these methods are only suited for simulation-based optimization studies, where one can precisely control the location of the samples. Once an algorithm exists to create the grid points and basis functions for different levels μ , the simulation is inquired at the simulation points, and the parameters are fitted by solving a system of linear equations (Fig. 5). A grid of $D=2$ and levels $\mu = 2$ and $\mu = 3$ is shown in Fig. 4, along with their associated functional forms (Eq. (5)). As seen in Fig. 4, a Sparse Grid in 2 dimensions has $N=13$ points for level of approximation $\mu = 2$ and $N=29$ points for $\mu = 3$. In order to validate the approximation error of the fitted functions, we calculate the approximation error on sample points that do not lie on the SG (training set). In this work, we use points collected based on a Latin Hypercube Design as validation points (validation set), and these will be discussed in the next section.

$$f_{SG_3} = b_0 + b_1x_1 + b_2(2x_1^2 - 1) + b_3x_2 + b_4(2x_2^2 - 1) + b_5(4x_1^3 - 3x_1) + b_6(1 - 8x_1^2 + 8x_1^4) + b_7x_1x_2 + b_8x_2(2x_1^2 - 1) + b_9x_1(2x_2^2 - 1) + b_{10}(2x_2^2 - 1)(2x_1^2 - 1) + b_{11}(4x_2^3 - 3x_2) + b_{12}(1 - 8x_2^2 + 8x_2^4) + b_{13}(5x_1 - 20x_1^3 + 16x_1^5) + b_{14}(18x_1^2 - 48x_1^4 + 32x_1^6 - 1) + b_{15}(56x_1^3 - 112x_1^5 + 64x_1^7 - 7x_1) + b_{16}(1 - 32x_1^2 + 160x_1^4 - 256x_1^6 + 128x_1^8) + b_{17}x_2(4x_1^3 - 3x_1) + b_{18}x_2(1 - 8x_1^2 + 8x_1^4) + b_{19}(4x_1^3 - 3x_1)(2x_2^2 - 1) + b_{20}(2x_2^2 - 1)(1 - 8x_1^2 + 8x_1^4) + b_{21}x_1(4x_2^3 - 3x_2) + b_{22}(2x_1^2 - 1)(4x_2^3 - 3x_2) + b_{23}x_1(1 - 8x_2^2 + 8x_2^4) + b_{24}(2x_1^2 - 1)(1 - 8x_2^2 + 8x_2^4) + b_{25}(5x_2 - 20x_2^3 + 16x_2^5) + b_{26}(18x_2^2 - 48x_2^4 + 32x_2^6 - 1) + b_{27}(56x_2^3 - 112x_2^5 + 64x_2^7 - 7x_2) + b_{28}(1 - 32x_2^2 + 160x_2^4 - 256x_2^6 + 128x_2^8) \quad (5)$$

3.2. Artificial neural network (ANN) models based on Latin hypercube sampling (LHS)

Even though the basic concept of ANNs has been known since the middle of the 20th century, ANNs have recently gained popularity, as they started to outperform other machine learning techniques (Schmidhuber, 2015). ANNs have favorable properties for a use as surrogate models: ANN evaluation is fast and therefore, ANNs are suitable for application in process design and optimization (Hoskins and Himmelblau, 1988). In this work, we used multi-layer feedforward networks (FFN) for surrogate modeling, which is a class of ANNs that can approximate any Borel measurable function to any desired degree of accuracy (Hornik et al., 1989). FFNs have been applied to different tasks in the field of chemical engineering, for example, CO₂ capture processes (Nuchitprasittichai and Cremaschi, 2013), Fischer-Tropsch synthesis (Fernandes, 2006), Biodiesel production (Yuste and Dorado, 2006), pressure swing adsorption processes (Lewandowski et al., 1998) and some of the most common unit operations (Henao and Maravelias, 2011).

ANNs are inspired by the way natural neurons process information. First, the incoming values x_d are weighted by a factor w_{md} , which is fitted for each connection between two neurons. Those weighted values are then summed together the so-called bias term (b_m). Second, this sum is exposed to an activation function f to gain



Fig. 5. Steps for building and fitting SG polynomial functions.

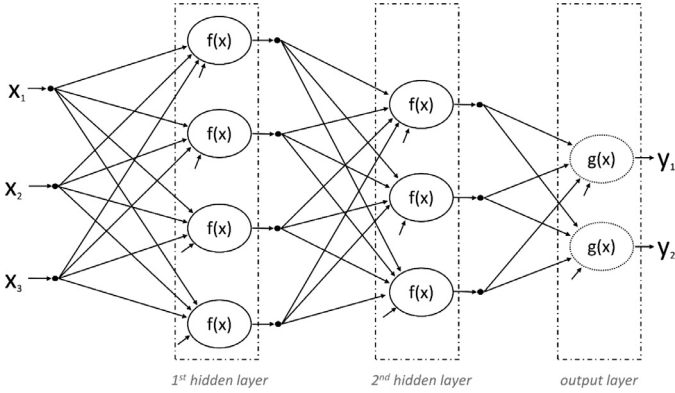


Fig. 6. Example of a 3-4-3-2-shaped FFN mapping from $X \in \mathbb{R}^3$ to $Y \in \mathbb{R}^2$.

the neurons output y_m , as shown in Eq. (6):

$$y_m = f\left(b_m + \sum_d w_{md} x_d\right) \quad (6)$$

Using nonlinear activation functions enables the FFN to represent complex nonlinear relationships between inputs and outputs. A schematic of how artificial neurons are organized in a FFN can be found in Fig. 6. A neuron receives the outputs from all neurons in the anterior layer and passes its output to all neurons in the posterior layer. The last layer in the direction of calculation is called output layer, while all previous layers, except the first input layer, are hidden layers. For many years, logistic functions like the hyperbolic tangent or the sigmoid function serve as the most common activation functions (LeCun et al., 2012). Recently neurons with different activation functions, such as rectifier liner units ReLUs and exponential linear units ELUs, have become more popular (Clevert et al., 2016).

When fitting ANN models, we have more flexibility with respect to selecting the sampling design. In this work, Latin Hypercube Sampling (LHS) is used, which is a popular space-filling sampling technique for multidimensional spaces (McKay et al., 1979). Based on LHS, a specified number of sample points is positioned in the space such that the minimum distance between each pair of points is maximized. Based on this criterion, LHS designs are known to span the space relatively well and as a result clustering of points that can be caused by random sampling is avoided. A LHS design with $N = 39$ points used in this work is shown in Fig. 7. The total number of points is selected such that a fraction of the points can be used for training ($N = 29$), and the rest used for validation. The size of the training and validation set is selected to match the number of samples used to train and validate the polynomial surrogate models. In this case study we found that different set of training and validation points through cross validation, led to different accuracy in the best fitted FFN surrogate models. This is probably caused by the relatively low number of data points available which changed the composition of the training set substantially between runs. We have observed that in certain cases, when randomly splitting the validation and training sets, we encounter the case where some validation points are placed outside the region that is spanned by the training set. This may cause

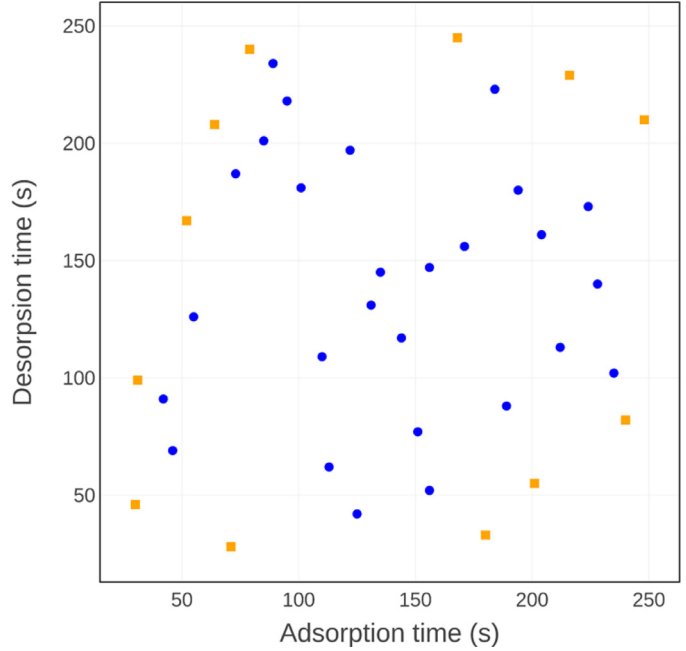


Fig. 7. Total set of Latin Hypercube samples collected for training and validation of ANNs. Dominated points are shown in blue circles, non-dominated points are shown in orange squares. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

high validation errors, resulting in the disadvantageous behavior of early stopping, since the validation points are basically extrapolation points. Therefore, for FFN fitting, the splitting of training and validation was not done fully randomly, but in a way that ensured that “non-dominated” points of the dataset were always part of the training data (Fig. 7). In this case a point is “dominated” if there is at least one other point with higher or lower values for adsorption or desorption time in any of the four combinations of axis and directions. Hence non-dominated points are all points that do not fit this requirement in at least one of the four possible combinations of directions. In addition, the split into training and validation sets is done so that in every quarter of the LHS dataset the fraction of validation points is equal. This is done to avoid possible clustering of the validation points in on area.

The number of neurons and layers, the training algorithm and the activation function all have an impact on the quality of the obtained FFN. In this work, we used a modified Genetic Algorithm to identify a good set of hyperparameters for training, because hyperparameters influence each other and training is overall an optimization problem. Two different approaches were investigated, a multiple input multiple output (MiMo) FFN where all outputs are calculated simultaneously and individual multiple input single output (MiSo) FFN for each of the four outputs. In preparation of training five different kinds of hyperparameters need to be specified: the number of layers, the number of neurons in each layer (which can vary in each layer), the activation function of the hidden and output layers and the training algorithm for the network. In all cases, we ensure that the shape of a FFN is chosen

in a way that the degrees of freedom in the FFN are lower or equal the number of training points available. The training procedure of these structures was done in an automated framework that was developed by integrating several state-of-the-art algorithms in the scikit-learn and Keras packages on a TensorFlow backend (Chollet, 2015; Abadi et al., 2015; Pedregosa et al., 2011) and details regarding the training parameters and settings are provided in the Results. Overall, the training and validation procedure for FFN models is a more involved procedure to ensure that the best model is found and overfitting is avoided, especially in this case of limited data ("small data"). However, the high-accuracy and high-flexibility of FFN models make them attractive, especially for higher dimensions, when large data sets are available and particularly when the samples cannot be selected.

4. Methods for handling uncertainty

In this section we present various ways to manage the different forms of uncertainty outlined earlier. We present three different ideas: (a) formulating a robust counterpart surrogate-based optimization formulation; (b) formulating a stochastic formulation using an ensemble of surrogates; and (c) fitting the uncertainty in the form of a discrepancy to correct a nominal surrogate model. These three ideas are not tied to the specific surrogate models presented in this work, and therefore they represent general methodologies that can be applied for a wide range of studies, if similar datasets were available. However, we will describe why some of these approaches are only applicable with certain general classes of surrogate models, based on the form of the surrogate model. At this point it is important to mention that there is a large body of work for quantification of uncertainty using model-based approaches, or reachability analysis methods for dynamical systems with uncertainty (Scott and Barton, 2013; Shen and Scott, 2017). However, here we discuss approaches for optimization under uncertainty, assuming that the simulation is too complex and therefore is treated as a black-box problem.

4.1. Robust optimization

Robust optimization formulations aim to find a single optimal solution that is feasible for all the realizations within an uncertain parameter space. Early work of this concept proposed performing perturbations of a nominal problem and formulating a linear programming problem that would result to an optimal solution that is feasible for all possible perturbations (Ben-Tal et al., 2009; Ben-Tal and Nemirovski, 1999; Soyster, 1973). This idea has evolved over the years resulting to several recent advances which aim to find probabilistic bounds of optimization problems which suffer from parametric uncertainty (Ben-Tal and Nemirovski, 1999; Ghaoui and Lebret, 1997; Guzman et al., 2016; Li et al., 2011; Li et al., 2012; Matthews et al., 2018; Yuan et al., 2016). Accounting for the uncertainty embedded in mostly all real-life applications, significant effort has been devoted recently to combine data-driven techniques for characterizing the uncertainty in robust optimization (Ben-Tal et al., 2009; Birge and Louveaux, 2011). In robust optimization, the uncertain parameters belong to a given uncertainty set and the robust solution is known to be immunized against uncertainty (Ben-Tal et al., 2009). While most robust optimization studies have been focused on deriving standard uncertainty sets from available data (Ben-Tal et al., 2009; Ben-Tal et al., 2013; Bertsimas et al., 2018; Li et al., 2019), recently the use of data and machine learning has been proposed to find more customized uncertainty sets (Ning and You, 2018; Shang et al., 2017). In other recent related work, surrogate models have been used to find the convex subregions (Zhang et al., 2015) and the uncertainty sets from available data (Ning and You, 2018; Shang et al., 2017). The work of Bertsimas

et al. (2010a, 2010b) is the most relevant work to this paper. This work tackles the challenge of parameter and model mismatch uncertainty in simulation-based optimization, through a robust local search algorithm that requires the simulation to provide input-output values and the gradient of the objective function. In this work we use the definition of the robust formulation presented by Bertsimas et al. to build a robust counterpart surrogate model, while global optimization is used to optimize the surrogate-based formulations and the gradients of the simulation are not required.

One of the major decisions in robust counterpart optimization is the selection of the parameter uncertainty set, which significantly affects the conservativeness of the final solution (Ben-Tal et al., 2009; Bertsimas et al., 2018; Guzman et al., 2016; Li et al., 2011). In this work we formulate the constrained surrogate-based optimization problem as a robust counterpart formulation with box uncertainty, where the uncertain parameters are the fitted parameters of the surrogate models (P2).

$$\begin{aligned} \max \quad & \sum_{n=1}^N \tilde{b}_{n,Prod} g_n(x) \\ \text{subject to:} \quad & \sum_{n=1}^N \tilde{b}_{n,m} g_n(x) \leq c_m \quad m = \{Rec, Pur, Ene\} \end{aligned} \quad (P2)$$

where $\tilde{b}_{n,m}$ represent the uncertain parameters of the surrogate models for the objective function (productivity) and constraints (recovery, purity, energy), respectively. Parameters c_m represent the right-hand side of the feasibility constraints that are set to a known deterministic value. One of the pre-requisites of this formulation is that the uncertain parameters must participate linearly in the model, which makes this approach applicable only with a subset of the surrogate models. As a result, we can only formulate the above Robust Counterpart (RC) surrogate problem with polynomial functions, since neural network functions is a convoluted nonlinear function with respect to the parameters. Consequently, functions g_n represent the basis functions of the polynomials shown in Eq. (5). In RC optimization, uncertain parameters are represented as the sum of the nominal value and their perturbation, as shown in Eq. (7).

$$\tilde{b}_{n,m} = b_{n,m} + \xi_{n,m} \hat{b}_{n,m} \quad \forall n = 1, \dots, N \quad m = \{Prod, Pur, Rec, Ene\} \quad (7)$$

where $b_{n,m}$ represent the nominal value, $\hat{b}_{n,m}$ represent a positive perturbation and $\xi_{n,m}$ represent independent random variables which are subject to uncertainty that is bounded by the selected uncertainty set. The final RC formulation takes different forms depending on the uncertainty set that is selected to describe the uncertain parameters. By writing the objective function as an uncertain constraint and using a box uncertainty set, (P2) can be written as follows:

$$\begin{aligned} \max z \\ \text{subject to:} \\ z - \sum_{n=1}^N b_{n,Prod} g_n(x) + \Psi \sum_{n=1}^N \hat{b}_{n,Prod} |g_n(x)| \leq 0 \\ \sum_{n=1}^N b_{n,m} g_n(x) + \Psi \left[\sum_{n=1}^N \hat{b}_{n,m} |g_n(x)| \right] \leq c_m \quad m = \{Rec, Pur, Ene\} \end{aligned} \quad (P3)$$

where Ψ represents the uncertain set parameter. For the simplest case of box uncertainty within an interval, $\Psi = 1$. One of the advantages of robust optimization when compared to stochastic optimization is that the size of the robust formulation does not in-

crease from the nominal formulation in size or complexity. Details about how we calculate $b_{n,m}$ and $\hat{b}_{n,m}$ for the surrogate-based direct-air capture problem, given the data that is available, will be provided in the Results.

4.2. Stochastic programming

Stochastic programming is another commonly used technique of decision-making under uncertainty, which is introduced in the 1950s by Dantzig (1955). In the contrast to robust optimization, the uncertain parameters in stochastic programming are assumed to follow a probability distribution obtained from historical data or prior knowledge (Birge and Louveaux, 2011). The different realizations of the uncertain parameters are scenarios which are used to formulate the stochastic optimization problem. The objective function of stochastic programming is to optimize the expectation of the productivity, subject to constraints that represent multiple scenarios. Therefore, to calculate the expectation of the productivity, a probability distribution assumption is needed. Given the data that has been collected, the key assumption made here is that all ten scenarios are identically-distributed in the DAC process, which is essentially equivalent to the sample average approach (Birge and Louveaux, 2011). In order to formulate a stochastic programming surrogate-based formulation, the different scenarios are the different realizations of the simulation outputs for different values of mass transfer coefficient k . In other words, the stochastic formulation aims to find the expected optimal solution when the parameters of the surrogate models are uncertain. More specifically, the expectation of the productivity $E(P_i)$ can be expressed as:

$$E(P_i) = \sum_{i=1}^N p_i P_i \quad (8)$$

where P_i is the productivity for each instance i , and $p_i = \frac{1}{N}$. $N = 10$ is the number of scenarios. The full stochastic formulation (P4) is:

$$\begin{aligned} \max_{t_{ads}, t_{des}} \quad & \frac{1}{N} \sum_{i=1}^N P_i(t_{ads}, t_{des}, k_i) \\ \text{s.t.} \quad & 15 \leq \frac{1}{N} \sum_{i=1}^N \text{recovery}_i(t_{ads}, t_{des}, k_i) \leq 100 \\ & 95 \leq \frac{1}{N} \sum_{i=1}^N \text{purity}_i(t_{ads}, t_{des}, k_i) \leq 100 \\ & 0 \leq \frac{1}{N} \sum_{i=1}^N \text{energy}_i(t_{ads}, t_{des}, k_i) \leq 0.2 \\ & 25 \leq t_{ads} \leq 250 \\ & 25 \leq t_{des} \leq 250 \end{aligned} \quad (P4)$$

Notably, this is a simple one-stage stochastic programming problem with t_{ads} and t_{des} as the two variables. If more data is collected, or additional knowledge about the system becomes available, different assumptions regarding the distribution or the weights of the objective function can be used.

4.3. Discrepancy modeling

The final approach for handling uncertainty differs from the above two, because it does not provide a single robust or expected optimal solution, but it can be used to develop a “map” of the uncertainty in the form of a surrogate model. This approach also allows us to combine the two surrogate modeling approaches (i.e., polynomials and Neural Networks) in the form of a hybrid model. The concept of discrepancy modeling can be found in different variations in different fields which use different terminology.

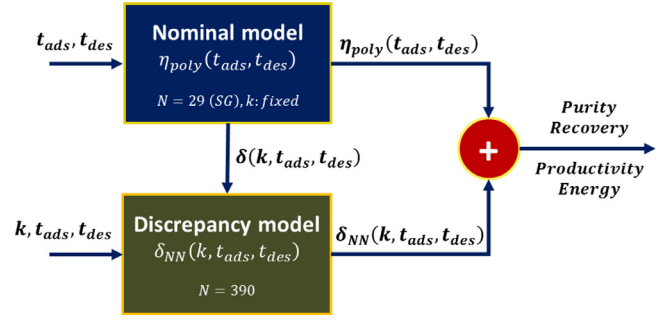


Fig. 8. Structure of hybrid nominal/discrepancy model structure.

Kennedy and O'Hagan discuss the concept of lumping many different forms of uncertainty, such as parameter uncertainty, model inadequacy, residual variability, and code uncertainty into a discrepancy model term ($\delta(x)$) (Kennedy and O'Hagan, 2001). The discrepancy model is essentially a model of the nominal model ($\eta(x)$) error, without accounting for the randomly distributed measurement error (ε):

$$f(x) = \eta(x) + \delta(x) + \varepsilon \quad (9)$$

This concept is based on the realization that all models are approximations of real systems and uncertainty exists in many forms. All of the aforementioned forms of uncertainty can be lumped into the modeling of the “systematic error”. This idea has been extended and applied into modeling of dynamical systems, by combining a reduced-model error with a dynamic discrepancy model (Li et al., 2017). This concept shares some similarities with the concept of hybrid modeling that has been used in the PSE literature extensively (von Stosch et al., 2014). Specifically, researchers have proposed several hybrid modeling structures (i.e., serial or parallel) that combine inaccurate or incomplete mechanistic models with black-box models that capture the residuals between the model and the experimental data (Duarte et al., 2004; Thompson and Kramer, 1994; Van Can et al., 1996). Particularly Neural Network models have been found to be very useful in modeling such residuals between a model and experimental observations.

We merge concepts from the area of hybrid modeling and discrepancy modeling, as our last proposed alternative for handling uncertainty for data-driven surrogate-based optimization. We develop a hybrid model that can capture the nominal effects, but also provide corrections of the nominal predictions depending on the uncertainty caused by the mass transfer coefficient k . In our approach, the nominal model is a polynomial model, $\eta_{poly}(t_{ads}, t_{des})$, fitted only with data from a single (nominal) k value (Fig. 8). Our discrepancy model, $\delta_{NN}(t_{ads}, t_{des}, k)$, is a FFN model that is fitted using as inputs all of the remaining available data, while the output is the error between the nominal prediction and the simulation data (Eq. (10)). Since this is a simulation-based case study, we can assume that the measurement error is negligible.

$$f_{hybrid}(t_{ads}, t_{des}) = \eta_{poly}(t_{ads}, t_{des}) + \delta_{NN}(t_{ads}, t_{des}, k) \quad (10)$$

The discrepancy model $\delta_{NN}(t_{ads}, t_{des}, k)$ is essentially a map of the effects of the uncertain parameter k on the predicted outputs. Undoubtedly, an alternative approach would use all of the data, treating the uncertain parameter k as an input variable, and fit a single 3-variable model directly in one pass. However, since k is not a controllable input, the hybrid model structure of Eq. (10) aims to represent the typical case where one would have a nominal or mechanistic model available, and thus the goal would be to model the trends of the residuals or the systematic bias of that model in a second stage. In addition, it has been claimed that if the nominal model is sufficiently accurate, the map of the residuals is a smoother function with less variability, and as a result

Table 1
Applied hyperparameter combinations for FFN training.

Model	Surrogate type	Shape	Training algorithm	Hidden layer activation	Output layer activation
All (MiMo)	MiMo	2–8–6–4	Nadam	tanh	linear
Energy	MiSo	2–7–1	Adam	tanh	linear
Productivity	MiSo	2–4–2–1	Adam	tanh	linear
Purity	MiSo	2–4–2–1	Nadam	tanh	linear
Recovery	MiSo	2–7–1	Nadam	tanh	linear

an easier output to fit using a surrogate model. We will test this hypothesis in the results, by comparing the quality of approximation and optimization via the proposed hybrid model and the single black-box three-variable model, where t_{ads} , t_{des} and k are all treated as inputs.

5. Results

5.1. Quality of approximations

In this section, we will compare the performance of the two surrogate modeling techniques used to approximate the simulation data. We have performed all of the studies in this work by collecting the following data from the simulation: (1) 29 sparse grid samples for 10 realizations of k (290 points); (b) 39 Latin Hypercube points for 10 realizations of k (390 points); (c) 9 validation points for 10 realizations of k (90 points), (d) 8 points outside the experimental region for extrapolation (80 points). A figure of all of the sampling locations is provided in Supplementary Information. The first set of points was used for training the polynomial approximations, the second was used for training and validation of the FFN models and the last category was not used for any training or testing procedure, and thus served as the validation set of both surrogate model types.

One of the key findings of this comparative work is that the selection of sampling sizes, training, validation and test sets played a significant effect on the final results and conclusions, especially because we are operating in a “small-data” regime. The fitting and validation procedure of polynomial models was described previously. Training and validation of FFNs is performed in epochs, where one epoch represents one iteration of the training algorithm in which the FFN is exposed to training points and the weights are adjusted according to the output errors. During each epoch, the number of training points that are used before a weight update is performed can be tuned. The choices are after every point, after all points or after a certain number of points, and this parameter is referred as mini batch size (Ruder, 2016). In this work a maximum number of 100,000 epochs with 29 points of training data (to be directly comparable with the polynomial surrogate models) using a mini batch size of four. To prevent overfitting 10 additional points of validation data are used, which corresponds to a validation split of 25.6%. If in 10,000 epochs the error on the validation points remains constant, then early stopping is employed to terminate the training. The minimum validation error is taken as a measure of quality for the obtained FFN. For each surrogate model developed, the training was done 40 times with the same set of hyperparameters and the best obtained FFN was selected for further use as surrogate model.

The identified hyperparameter combinations of the optimally identified FFN for the Multiple Input-Multiple Output (MiMo) model, and the Multiple Input-Single Output (MiSo) models are shown in Table 1. The identified set of hyperparameters for one value of the mass transfer coefficient is assumed to be suitable for all other FFNs representing the same input-output relation at different values of k .

Despite their very different structure and sample locations, both surrogate model types were able to capture the four different out-

Table 2
Relative average and maximum absolute Errors (%) for validation points.

	Purity	Recovery	Productivity	Energy
FFN – MiMo (rMAE)	0.029	0.809	0.574	0.681
FFN – MiSo (rMAE)	0.028	0.385	0.977	0.669
SG – Poly (rMAE)	0.020	0.522	1.232	2.177
FFN – MiMo (rMaxAE)	0.157	5.233	3.231	5.084
FFN – MiSo (rMaxAE)	0.248	1.998	8.871	5.811
SG – Poly (rMaxAE)	0.053	1.279	2.922	7.596

Table 3
Relative average and maximum absolute errors (%) for extrapolation points.

	Purity	Recovery	Productivity	Energy
FFN – MiMo (rMAE)	1.925	20.727	45.430	15.049
FFN – MiSo (rMAE)	1.715	10.028	41.041	15.289
SG – Poly (rMAE)	0.801	12.708	19.179	13.631
FFN – MiMo (rMaxAE)	19.567	351.267	197.381	52.265
FFN – MiSo (rMaxAE)	19.407	78.971	173.083	45.574
SG – Poly (rMaxAE)	3.562	43.811	57.778	53.769

puts of Productivity, Recovery, Purity and Energy with sufficient accuracy. In Table 2, we report the relative mean absolute error (rMAE) and relative maximum absolute error (rMaxAE) for the three different types of surrogate models and all outputs for the same validation points. In Table 3, we report the same error measures for 8 sample points collected outside the [25 – 250] range to test the extrapolating capabilities of the models. When it comes to predictions within the experimental region, we observe that the rMAE is below 1% for most models and FFNs perform better for three out of the four outputs. However, polynomial surrogates typically have reduced absolute maximum errors between the surrogate and the simulation. This result validates the theory of Sparse-Grid polynomial interpolation, which states the maximum error between the true function and the surrogate model converges to zero, as the level of approximation increases. Based on the results shown in Table 3, we observe that polynomials tend to perform better in the case of extrapolation, apart from two exceptions. This result could be because Sparse Grid points cover the extreme points of the search space. Overall, we observe that the performance of these surrogates is not promising for extrapolation and should not be used for predictions outside the range of the original experimental design. In Figs. 9 and 10 we show the surrogate model predictions for both polynomial functions and FFN for a nominal $k = 18.6 \text{ h}^{-1}$. As expected, the plots look similar and we observe that all of the four outputs are nonlinearly dependent on the two inputs.

5.2. Deterministic optimization results

If the uncertainty of k was not considered and one performed a surrogate-based optimization study for the nominal k value, a deterministic optimal solution would be obtained from each of the surrogate models (Table 4). Both surrogate formulations were solved in GAMS using BARON (Tawarmalani, 2005). The FFN models were solved in Pyomo (Hart et al., 2017), which generates a reduced-space formulation of the FFN models (Schweidtmann and Mitsos, 2019) that is solved in GAMS using the same solver. Despite

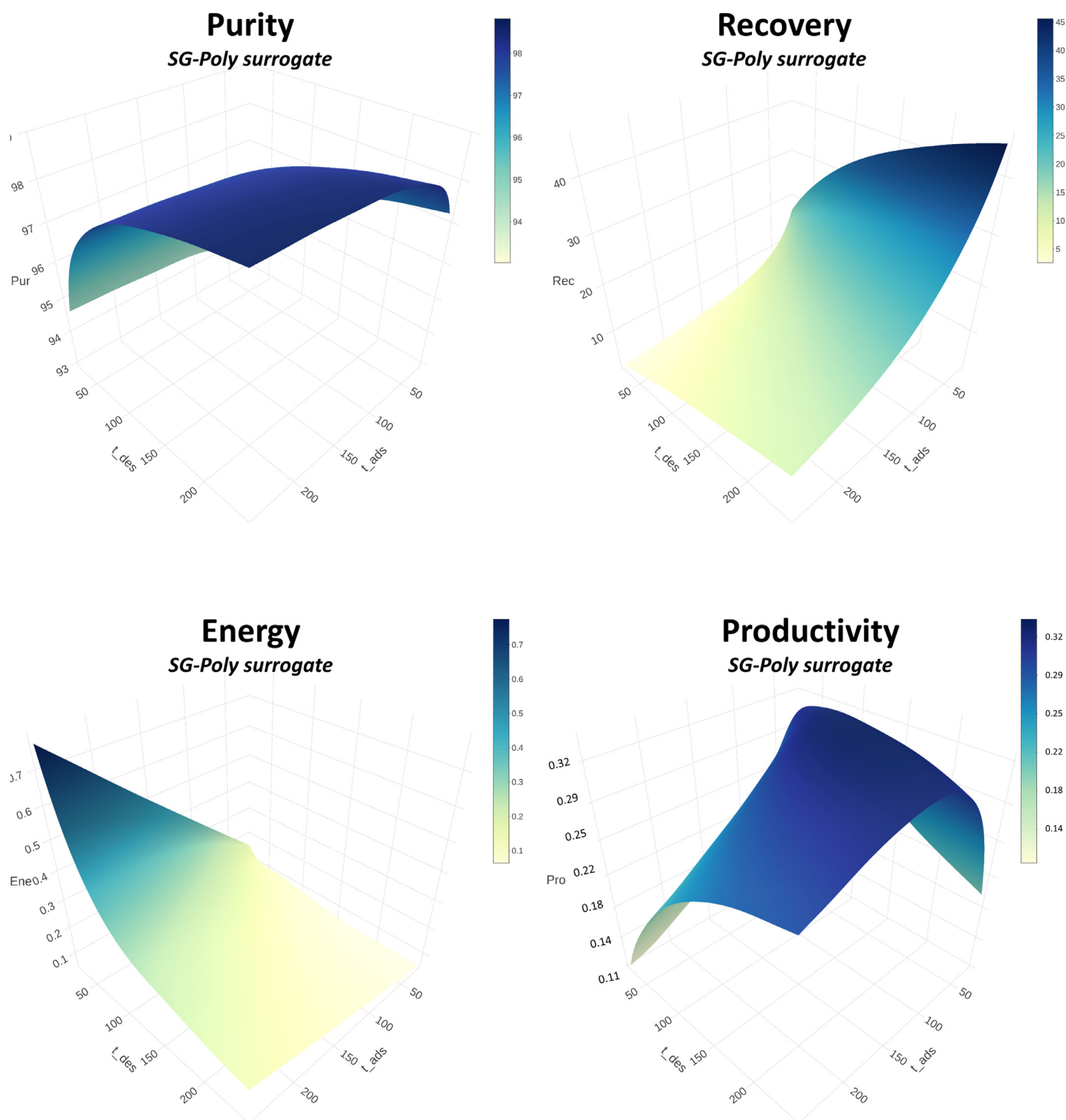


Fig. 9. Surrogate functions using Sparse Grids and polynomials for nominal $k = 18.6 \text{ h}^{-1}$.

the fact that both surrogate models are accurate, and the same global optimizer is used to optimize them, we observe that the optimal solution varies significantly when using a different surrogate model. In addition, since these results are based on a nominal k value, we should not expect these solutions to be feasible when k varies. In fact, by fixing the values of t_{ads} and t_{des} to the values in Table 4 for the polynomial and FFN respectively, and evaluating the prediction of the outputs for different values of k , we found that recovery and energy constraints were violated in almost half of the realizations of k .

In addition to the variability in optimal solutions that may be observed when using different surrogate models, we expect to obtain different optimal solutions when individually optimizing different surrogate models for each k value. This analysis was performed using polynomial surrogate models and the results are shown in Table 5, where we see that adsorption time varies within $[70.6 - 101.4]$ and desorption time varies within $[128 - 195]$. This variation in solutions validates our assumption that k plays a significant role and that the effects of k in the optimal solutions are also nonlinear. We observed similar variations when optimizing

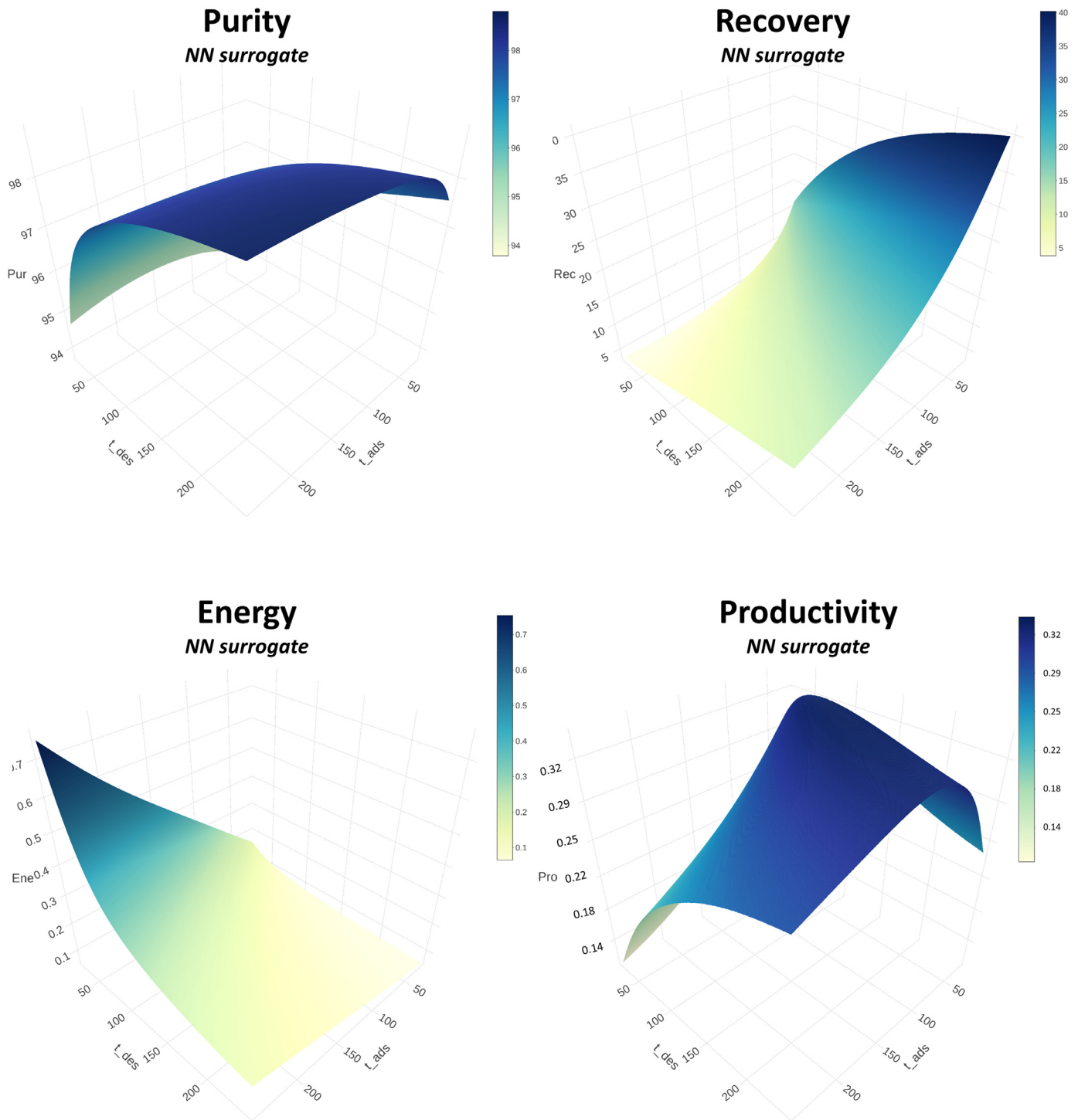


Fig. 10. Surrogate functions using Hypercube samples and neural networks for nominal $k = 18.6 \text{ h}^{-1}$.

the FFN models. In the next sections we will provide the results obtained when accounting for uncertainty and we will compare the obtained results with this nominal deterministic case.

5.3. Robust optimization with polynomial surrogate models

Solving the robust surrogate-based formulation (P3) can be considered as solving of a problem comprised of surrogate functions that are under estimations of the different surrogate realizations. Due to the nonlinearity of the problem, we cannot guarantee that

these are valid under-estimators of all of the realizations of the uncertain parameter space. However, by using the most conservative uncertainty set, namely interval uncertainty, we aim to get a conservative underestimation of the outputs we are approximating. This is shown in Fig. 11, where we overlay the polynomial surrogate models for the objective function of productivity for all values of the uncertain mass transfer coefficient, as well as the robust-counterpart surrogate model which bounds all of the realizations from below. The optimal solution of formulation (P3) is expected to highly increase the probability of obtaining a solution that will

satisfy all constraints throughout the uncertain parameter space of k . Also, it is important to note that through this formulation, we are grouping the effects of both forms of uncertainty (k and surrogate modeling) together.

One of the challenges of formulation (P3) is the calculation of the nominal ($b_{n,m}$) and deviation parameters ($\hat{b}_{n,m}$). Here, we used the average of each of the polynomial term parameters over all realizations of k as our nominal parameters. The deviation parameters are calculated using the standard deviation of the optimal parameters of individual surrogate models. We carefully studied the behavior of the optimal surrogate parameter distributions and observed that they are narrow bell-shaped distributions. This can be explained by the fact that although k influences the surrogate functions, the trends remain similar and thus the optimized parameter values tend to cluster towards similar values. Selecting the deviation parameter has a significant effect on the conservativeness of the solution. In this work, we used a deviation equal to the standard deviation of the optimized parameters for different k values, which resulted to a reasonable bound of all of our four outputs (Fig. 11). Using the above statistics, the robust formulation (P3) was globally optimized using BARON and the optimal solution was identified as $t_{ads} = 48$ s, $t_{des} = 138$ sec, which differs from the deterministic solutions obtained in Table 4.

We validated this solution (Table 6) by running the simulation for all realizations of k that were initially used to build the models, and three additional k values that were not used to train any models: $k_{new} = [7.2, 14.4, 21.6]$. The most important result to observe in Table 6 is that no constraints are violated for any of the original or new mass transfer coefficients. In addition, the maximum objective function obtained is $0.372 \frac{\text{mol CO}_2}{\text{kg Sorbent} \times \text{h}}$, the minimum is 0.198 and the average is 0.305. These values are lower but comparable to the values of Table 5, when each surrogate model was optimized individually.

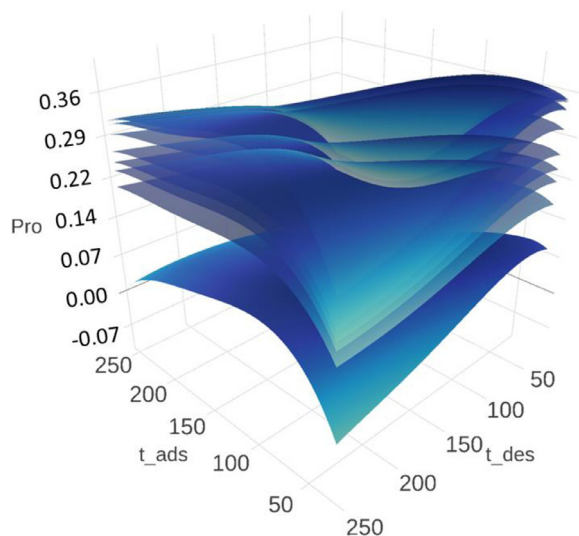


Fig. 11. Productivity polynomial surrogate models for all realizations of k bounded below by robust counterpart surrogate polynomial.

5.4. Stochastic optimization with surrogate models

Unlike the Robust surrogate approach, one advantage of formulation (P4) is that it can be solved for any surrogate model. As a result, two versions of (P4) were solved, namely one with all of the constraints and the objective are approximated by polynomial functions, and the second where all functions in P4 are FFN models. We performed this analysis to compare the performance of the different surrogate models, but nothing prohibits us from combining scenarios obtained by different types of surrogate models. The solution of the stochastic formulation does not provide a worst-case solution like in the case of the robust formulation, but it provides an expected maximum productivity. The expected solution when using polynomial models is: $(t_{ads}, t_{des}) = (73, 176)$, where the expected productivity is 0.327, while the optimal solution when the formulation is comprised solely on Neural Networks is: $(t_{ads}, t_{des}) = (78, 132)$, where the expected productivity is 0.328.

From the above two solutions we observe that the adsorption times are very similar for both problems, while desorption times differ. However, overall both formulations have led to relatively similar solutions, compared to the variability in solutions obtained when solving separate deterministic problems (Table 5). This is a promising indication of considering uncertainty when using surrogate models, since the result becomes less prone to the variation caused by all the different forms of uncertainty. However, variation in the solutions still exists, and this might be because of lack of sufficient data (or scenarios).

Finally, we validated the results of the solution obtained by the stochastic formulations, by running the simulation for the same k values as in the robust optimization study. This allows us to check whether the stochastic solution violates any constraints for different k values, and to calculate the accuracy of the expected productivity. As can be seen in Table 7, the obtained solution using the polynomial stochastic formulation does not violate any constraints and the average productivity is 0.315 (4% relative error from predicted expected productivity). It is also important to note that the solution obtained using the polynomial stochastic programming formulation has a higher average than the one obtained by the robust formulation. This is expected due to the conservativeness of the robust counterpart approach. However, the solution obtained using the FFN-based stochastic formulation has an expected productivity of 0.322 (2% relative error from predicted expected productivity), but violates the recovery constraint in 3 out of the 13 simulated scenarios. We hypothesize that this may be caused by the fact that polynomial models are less flexible than FFN models, and thus a larger number of scenarios of FFN models would be necessary to obtain more consistent and reliable sample average approximations.

5.5. Discrepancy modeling coupling polynomial approximations with neural networks

In this final section, we present the results for a hybrid model comprised of a nominal polynomial surrogate model and a FFN model of the error between the nominal model predictions and the simulated results. In order to develop this hybrid model, we used the 29 SG points for a nominal k to develop a model for predicting the purity, recovery, productivity and energy as a function of

Table 4

Deterministic global optimization results using different surrogate formulations for nominal k .

Model	t_{ads} (s)	t_{des} (s)	Recovery (predicted) (%)	Productivity (predicted) ($\frac{\text{mol CO}_2}{\text{kg Sorbent} \times \text{h}}$)	Energy (predicted) ($\frac{\text{MJ}}{\text{mole CO}_2}$)	Purity (predicted) (%)
(P1) Poly	88.7	165	20.4	0.340	0.106	98.104
(P1) FFN	76.0	132	19.98	0.341	0.112	97.751

Table 5

Globally optimal solutions of polynomial surrogates when different each realization of k is fitted and optimized individually.

$k (\frac{1}{h})$	t_{ads} (s)	t_{des} (s)	Recovery (%)	Productivity ($\frac{\text{mole CO}_2}{\text{kg Sorbent} \times \text{h}}$)	Energy ($\frac{\text{MJ}}{\text{mole CO}_2}$)	Purity (%)
30.9	70.6	128.0	23.82	0.387	0.10	97.65
18.6	88.7	165.0	20.40	0.341	0.11	98.10
31.1	72.6	133.6	23.92	0.386	0.09	97.72
5.7	79.6	195.0	15.00	0.209	0.15	97.74
9.6	101.4	176.8	15.44	0.271	0.14	98.14
25.9	74.8	140.5	23.04	0.372	0.10	97.81
29.9	85.6	162.7	23.42	0.386	0.09	98.05
26.1	75.3	141.9	23.11	0.372	0.09	97.83
12.1	96.3	171.4	17.15	0.296	0.12	98.14
8.0	97.0	182.0	15.00	0.252	0.14	98.06

Table 6

Simulation results for robust optimal solution (t_{ads}, t_{des}) = (48, 138).

$k (\frac{1}{h})$	Recovery (%)	Productivity ($\frac{\text{mole CO}_2}{\text{kg Sorbent} \times \text{h}}$)	Energy ($\frac{\text{MJ}}{\text{mole CO}_2}$)	Purity (%)
30.9	31.1	0.372	0.078	97.4
18.6	27.0	0.322	0.090	97.4
31.1	31.2	0.372	0.078	97.4
5.7	16.6	0.198	0.146	96.8
9.6	21.1	0.252	0.115	97.2
25.9	29.8	0.355	0.081	97.5
29.9	30.9	0.369	0.078	97.4
26.1	29.9	0.356	0.081	97.5
12.1	23.3	0.277	0.104	97.3
8.0	19.5	0.233	0.124	97.1
7.2	18.6	0.222	0.130	97.0
14.4	24.8	0.296	0.098	97.4
21.6	28.3	0.338	0.085	97.4

t_{ads} and t_{des} . The nominal model is one out of the ten models, for which approximation results were provided in Table 2. Once this model is fitted, we use all of the available Latin Hypercube data for all realizations of ($N = 39 \times 10$ points), and calculate the error (or discrepancy) between the nominal model prediction and the simulated output ($\delta(k, t_{ads}, t_{des})$) (Fig. 8). These errors are now the outputs of the discrepancy model ($\delta_{NN}(k, t_{ads}, t_{des})$), which are approximated using a FFN. The error models were trained and validated and we were able to identify optimal structures and hyperparameters that fit the errors with sufficient accuracy. This was the first indication that a systematic model can be fitted by a surrogate approximation. Indicative surface plots of the discrepancy model for a fixed $k = 30.9 \text{ h}^{-1}$ are shown in Fig. 12.

In order to validate the hybrid model, we calculate the error between its predictions and the simulation, for the same validation points of Table 2 (Table 8). We observe that the errors are comparable to the errors shown in Table 2, and thus the accuracy of this model for the entire search space of k is comparable to the model obtained by modeling each k individually. As a result, we can conclude that the discrepancy model can be used as a map of the error between the nominal case and different realizations of the uncertain parameter in the input space, and could be used to formulate various optimization problems. One of the main advantages of this approach is that it provides the ability to predict the discrepancy, or else the correction we need to add to our nominal predictions, for k values that have not been simulated.

Although this approach aims to develop a “correction” map, which will be additive to the prediction of a deterministic nominal model, one may argue that a simpler approach would be to treat the uncertain parameter as an additional input and fit a single three-input surrogate model. In order to assess the predictive ability of this larger model, we used all of the simulated data for all k values to train a single MiMo NN model with three inputs. Surprisingly, the size of the FFN for the 3-variable input model, is a slightly more accurate surrogate model (Table 8), and it is a network with 2 layers but an increased number of nodes (3–15–22–4). The optimal discrepancy FFN is a deeper network with less nodes in each layer (3–10–8–6–4). When comparing the total number of parameters of the discrepancy FFN model to the single 3-variable FFN model, the latter requires more fitted parameters. However, if the polynomial parameters are considered, the overall hybrid model (Poly+FFN) contains more parameters than the single-state 3-variable FFN. This is a surprising result that could be

Table 7

Simulated results for stochastic formulation using polynomials (t_{ads}, t_{des}) = (73, 176) (Poly) and NN (t_{ads}, t_{des}) = (78, 132) (NN). Constraint violations shown in bold.

$k (\frac{1}{h})$	Recovery (%)		Productivity ($\frac{\text{mole CO}_2}{\text{kg Sorbent} \times \text{h}}$)		Energy ($\frac{\text{MJ}}{\text{mole CO}_2}$)		Purity (%)	
	Poly	NN	Poly	NN	Poly	NN	Poly	NN
30.9	27.62	22.62	0.381	0.388	0.08	0.10	98.05	97.75
18.6	24.20	19.82	0.334	0.340	0.09	0.11	98.07	97.77
31.1	27.66	22.65	0.381	0.389	0.08	0.10	98.05	97.75
5.7	15.10	12.54	0.208	0.215	0.15	0.18	97.61	97.28
9.6	19.10	15.72	0.263	0.270	0.12	0.14	97.92	97.61
25.9	26.51	21.70	0.365	0.372	0.08	0.10	98.07	97.77
29.9	27.43	22.46	0.378	0.385	0.08	0.10	98.05	97.75
26.1	26.57	21.75	0.366	0.373	0.08	0.10	98.07	97.77
12.1	20.96	17.20	0.289	0.295	0.11	0.13	98.00	97.69
8.0	17.67	14.58	0.244	0.250	0.13	0.15	97.83	97.52
7.2	16.87	13.95	0.233	0.239	0.13	0.16	97.77	97.45
14.4	22.30	18.27	0.307	0.314	0.10	0.12	98.04	97.74
21.6	25.28	20.69	0.348	0.355	0.09	0.11	98.07	97.77

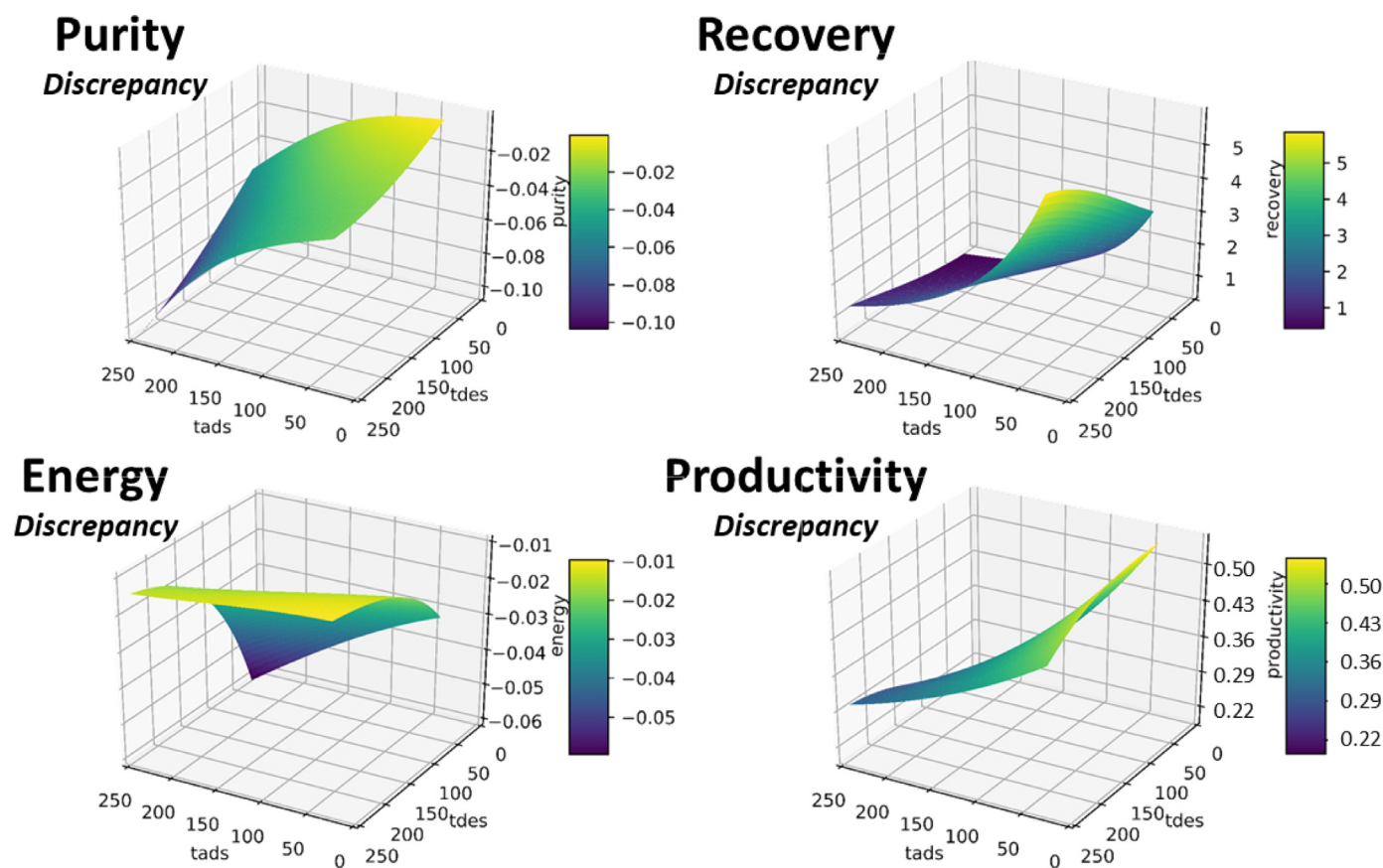


Fig. 12. Discrepancy models fitted using neural networks. The surfaces are shown for fixed $k = 30.9 \text{ h}^{-1}$.

Table 8

Relative absolute average and maximum errors (%) of hybrid discrepancy model over all values of k .

	Purity	Recovery	Productivity	Energy
Hybrid Discrepancy Model (rMAE)	0.035	1.616	1.400	3.124
Hybrid Discrepancy Model (rMaxAE)	0.096	3.534	4.052	7.260
3-input FFN Model (rMAE)	0.006	0.554	0.192	0.716
3-input FFN Model (rMaxAE)	0.020	2.268	0.614	2.069

explained by the fact that the residual mapping is a nonlinear response surface, and thus it is harder to fit than the smoother mapping between the 3 variables and the original outputs. However, the hybrid approach is still a promising approach that leads to an accurate overall model, especially when a widely-accepted nominal model pre-exists.

6. Conclusions

In this work we discuss the challenges caused by different forms of uncertainty in simulation-based optimization; advocate the need to consider uncertainty when performing surrogate-based optimization; and propose three basic ideas to manage this uncertainty. Our motivating case study is the optimization of a simulation for direct air capture by maximizing the productivity of the process, while satisfying purity, recovery and energy constraints. The two decision variables of the system are adsorption and desorption times, while the model outputs are nonlinearly dependent on the value of a mass transfer coefficient that is an uncertain simulation parameter. Due to the complexity of the simulation and lack of algebraic equations connecting the inputs to the outputs,

this problem is optimized following a surrogate-based optimization approach, based on which input-output data is collected, surrogate models are fitted and subsequently optimized. We use two very different types of surrogate models in this work (i.e., Sparse Grid polynomials and Neural Network models) to approximate the input-output data, and validate our obtained solutions with the rigorous simulation.

We show that both the uncertainty of the mass transfer coefficient and the selection of the surrogate model lead to variability in the obtained optimal values. Most importantly, we observe that often the solutions that are obtained when treating this problem as a deterministic case study are infeasible when simulated for different mass transfer coefficients. We compare three ideas for mitigating the effects of uncertainty by combining existing literature on robust optimization, stochastic optimization and discrepancy modeling, with surrogate approximations. The results show that we can obtain more conservative but feasible solutions when formulating a surrogate-based robust optimization problem, and a slightly less conservative solution when formulating a surrogate-based stochastic formulation. Finally, we show that we can accurately combine different surrogate models to capture both nominal effects but also

the discrepancy between the nominal and simulation model accurately. We observe that all three methods have merits and limitations, which are discussed throughout the paper.

The analysis performed in this work mainly aims to stress the issue of uncertainty embedded within surrogate-based optimization, which is not discussed enough in the PSE and black-box optimization literatures. The proposed techniques can serve as a guide for selecting the most appropriate approach to manage uncertainty in a variety of simulation-based optimization studies, depending on the chosen surrogate model, the availability of the data and the size of the problem. The results in this paper also aim to point to a more general debate for the simulation-based optimization literature; on whether investing on more sampling to achieve incremental improvements in deterministic optimal results is more valuable than sampling to characterize the unavoidable embedded uncertainty. In the future, it will be of high interest to compare the employed methods with recently proposed methods for Bayesian optimization under uncertainty, which have been used in other applications for robust optimization and discrepancy modeling.

Acknowledgments

The authors would like to thank the PSE 2018 organizing committee for the kind invitation to talk about this work. FB, SHK and JZ acknowledge funding from NSF CBET (1805724). FB, SHK and MJR acknowledge funding from RAPID/AICHE/DOE. AS, MJR acknowledge funding from NSF CBET (1336386).

Supplementary materials

Supplementary data associated with this article can be found, in the online version, at [10.1016/j.compchemeng.2019.106519](https://doi.org/10.1016/j.compchemeng.2019.106519).

References

- Amaran, S., Sahinidis, N.V., Sharda, B., Bury, S.J., 2014. Simulation optimization: a review of algorithms and applications. *4OR* 12, 301–333.
- Anna, H.R.S., Barreto, A.G., Tavares, F.W., de Souza, M.B., 2017. Machine learning model and optimization of a PSA unit for methane-nitrogen separation. *Comput. Chem. Eng.* 104, 377–391.
- Audet, C., Ihaddadene, A., Le Digabel, S., Tribes, C., 2018. Robust optimization of noisy blackbox problems using the mesh adaptive direct search algorithm. *Optim. Lett.* 12, 675–689.
- Audet, C., Kokkolaras, M., 2016. Blackbox and derivative-free optimization: theory, algorithms and applications. *Optim. Eng.* 17, 1–2.
- Beland, J.N., Nair, P.B., 2017. Bayesian optimization under uncertainty. In 31st Conference on Neural Information Processing Systems (NIPS) Long Beach, CA.
- Ben-Tal, A., Ghaoui, L.E., Nemirovski, A., 2009. *Robust Optimization*. Princeton University Press.
- Ben-Tal, A., Hertog, D.D., Waegenaere, A.D., Melenberg, B., Rennen, G., 2013. Robust solutions of optimization problems affected by uncertain probabilities. *Manage. Sci.* 59, 341–357.
- Ben-Tal, A., Nemirovski, A., 1999. Robust solutions of uncertain linear programs. *Oper. Res. Lett.* 25, 1–13.
- Bertsimas, D., Gupta, V., Kallus, N., 2018. Data-driven robust optimization. *Math. Program.* 167, 235–292.
- Bertsimas, D., Nohadani, O., Teo, K.M., 2010a. Nonconvex robust optimization for problems with constraints. *INFORMS J. Comput.* 22, 44–58.
- Bertsimas, D., Nohadani, O., Teo, K.M., 2010b. Robust optimization for unconstrained simulation-based problems. *Oper. Res.* 58, 161–178.
- Bhosekar, A., Ierapetritou, M., 2018. Advances in surrogate based modeling, feasibility analysis, and optimization: a review. *Comput. Chem. Eng.* 108, 250–267.
- Birge, J.R., Louveaux, F., 2011. *Introduction To Stochastic Programming*. Springer publishing company, Incorporated.
- Bogunovic, I., Scarlett, J., Jegelka, S., Cevher, V., 2018. Adversarially robust optimization with gaussian processes. In 32nd Conference on Neural Information Processing Systems (NIPS) Montreal, Canada.
- Boukouvala, F., Floudas, C.A., 2017. ARGONAUT: algorithms for global optimization of constrained grey-box computational problems. *Optim. Lett.* 11, 895–913.
- Boukouvala, F., Hasan, M.M.F., Floudas, C.A., 2017. Global optimization of general constrained grey-box models: new method and its application to constrained PDEs for pressure swing adsorption. *J. Global Optim.* 67, 3–42.
- Boukouvala, F., Ierapetritou, M.G., 2013. Surrogate-Based optimization of expensive flowsheet modeling for continuous pharmaceutical manufacturing. *J. Pharm. Innov.* 8, 131–145.
- Boukouvala, F., Misener, R., Floudas, C.A., 2016. Global optimization advances in mixed-integer nonlinear programming, MINLP, and constrained derivative-free optimization. *cdfo. Eur. J. Oper. Res.* 252, 701–727.
- Bungartz, H.J., Dirnstorfer, S., 2003. Multivariate quadrature on adaptive sparse grids. *Computing* 71, 89–114.
- Bungartz, H.J., Dirnstorfer, S., 2004. Higher order quadrature on sparse grids. In: Bubak, M., DickVanAlbada, G., Sloot, P.M.A., Dongarra, J.J. (Eds.). In: *Computational Science - ICCS 2004, Proceedings* (Vol. 3039, pp. 394–401).
- Chuang, Y.C., Chen, T., Yao, Y., Wong, D.S.H., 2018. Transfer learning for efficient meta-modeling of process simulations. *Chem. Eng. Res. Des.* 138, 546–553.
- Clevert, D.-A., Unterthiner, T., & Hochreiter, S. (2016). Fast and accurate deep network learning by Exponential Linear Units (ELUs). In arXiv preprint.
- Cox, D.D., John, S., 1992. A statistical method for global optimization. In: *Proceedings 1992 IEEE International Conference on Systems, Man, and Cybernetics*, 1242, pp. 1241–1246.
- Cozad, A., Sahinidis, N.V., Miller, D.C., 2014. Learning surrogate models for simulation-based optimization. *AIChE J.* 60, 2211–2227.
- Dantzig, G.B., 1955. Linear programming under uncertainty. *Manage. Sci.* 1, 197–206.
- Davis, S.E., Cremaschi, S., Eden, M.R., 2018. Efficient surrogate model development: impact of sample size and underlying model dimensions. In: Eden, M.R., Ierapetritou, M.G., Towler, G.P. (Eds.). In: *Computer Aided Chemical Engineering*, 44. Elsevier, pp. 979–984.
- Dias, L.S., Pattison, R.C., Tsay, C., Baldea, M., Ierapetritou, M.G., 2018. A simulation-based optimization framework for integrating scheduling and model predictive control, and its application to air separation units. *Comput. Chem. Eng.* 113, 139–151.
- Dowling, A.W., Eason, J.P., Ma, J., Miller, D.C., Biegler, L.T., 2014. Coal oxycombustion power plant optimization using first principles and surrogate boiler models. *Energy Procedia* 63, 352–361.
- Duarte, B., Saraiva, P., Pantelides, C., 2004. Combined mechanistic and empirical modelling. *Int. J. Chem. Reactor Eng.* 2.
- Dung, D., 2016. Sampling and cubature on sparse grids based on a B-spline quasi-interpolation. *Found. Comput. Math.* 16, 1193–1240.
- Eason, J., Cremaschi, S., 2014. Adaptive sequential sampling for surrogate model generation with artificial neural networks. *Comput. Chem. Eng.* 68, 220–232.
- Eason, J.P., Biegler, L.T., 2016. A trust region filter method for glass box/black box optimization. *AIChE J.* 62, 3124–3136.
- Eslick, J.C., Ng, B., Gao, Q., Tong, C.H., Sahinidis, N.V., Miller, D.C., 2014. A framework for optimization and quantification of uncertainty and sensitivity for developing carbon capture systems. *Energy Procedia* 63, 1055–1063.
- Fernandes, F.A.N., 2006. Optimization of Fischer-Tropsch synthesis using neural networks. *Chem. Eng. Technol.* 449–453.
- Francois Chollet, Keras, Github Repository, Retrieved from: Github repository, <https://github.com/fchollet/keras>, 2015.
- Gajda, P., 2005. Smolyak's algorithm for weighted L-1-approximation of multivariate functions with bounded rth mixed derivatives over R-d. *Numer Algorithms* 40, 401–414.
- Garud, S.S., Karimi, I.A., Brownbridge, G.P.E., Kraft, M., 2018a. Evaluating smart sampling for constructing multidimensional surrogate models. *Comput. Chem. Eng.* 108, 276–288.
- Garud, S.S., Karimi, I.A., Kraft, M., 2017. Smart sampling algorithm for surrogate model development. *Comput. Chem. Eng.* 96, 103–114.
- Garud, S.S., Karimi, I.A., Kraft, M., 2018b. LEAPS2: learning based evolutionary assistive paradigm for surrogate selection. *Comput. Chem. Eng.* 119, 352–370.
- Gerstner, T., Griebel, M., 1998. Numerical integration using sparse grids. *Numer Algorithms* 18, 209–232.
- Ghaoui, L.E., Lebret, H., 1997. Robust solutions to least-squares problems with uncertain data. *SIAM J. Matrix Anal. Appl.* 18, 1035–1064.
- gPROMS, 1997–2018. Process Systems Enterprise.
- Grimstad, B., Sandnes, A., 2016. Global optimization with spline constraints: a new branch-and-bound method based on B-splines. *J. Global Optim.* 65, 401–439.
- Guzman, Y.A., Matthews, L.R., Floudas, C.A., 2016. New a priori and a posteriori probabilistic bounds for robust counterpart optimization: I. Unknown probability distributions. *Comput. Chem. Eng.* 84, 568–598.
- Harding, B., 2016. Adaptive sparse grids and extrapolation techniques. In: Garcke, J., Pfleger, D. (Eds.). In: *Sparse Grids and Applications*, 109, pp. 79–102 Stuttgart 2014.
- Hart, W.E., Laird, C.D., Watson, J.-P., Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., Sirola, J.D., 2017. *Pyomo - Optimization Modeling in Python*, 67, second ed. Springer.
- Hastie, T., Tibshirani, R., Friedman, J., 2009. *The Elements of Statistical Learning* New York. Springer, NY.
- Henao, C.A., Maravelias, C.T., 2011. Surrogate-based superstructure optimization framework. *Process Syst. Eng.* 1216–1232.
- Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. *Neural Netw.* 359–366.
- Hoskins, J.C., Himmelblau, D.M., 1988. Artificial neural network models of knowledge representation in chemical engineering. *Comput. Chem. Eng.* 881–890.
- Hulsmann, M., Reith, D., 2013. SpaGROW-A derivative-free optimization scheme for intermolecular force field parameters based on sparse grid methods. *Entropy* 15, 3640–3687.
- Ibrahim, D., Jobson, M., Li, J., Guillen-Gosalbez, G., 2018. Optimization-based design of crude oil distillation units using surrogate column models and a support vector machine. *Chem. Eng. Res. Design* 134, 212–225.
- Jones, D.R., Schonlau, M., Welch, W.J., 1998. Efficient global optimization of expensive black-box functions. *J. Global Optim.* 13, 455–492.

- Judd, K.L., Maliar, L., Maliar, S., Valero, R., 2014. Smolyak method for solving dynamic economic models: lagrange interpolation, anisotropic grid and adaptive domain. *J. Econ. Dyn. Control* 44, 92–123.
- Kennedy, M.C., O'Hagan, A., 2001. Bayesian calibration of computer models. *J. R. Stat. Soc. B* 63, 425–464.
- Kieslich, C.A., Boukouvala, F., Floudas, C.A., 2018. Optimization of black-box problems using Smolyak grids and polynomial approximations. *J. Global Optim.* 71, 845–869.
- LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.-R., 2012. Efficient BackProp. In: *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 9–48.
- Lewandowski, J., Lemcoff, N.O., Palosaari, S., 1998. Use of neural networks in the simulation and optimization of pressure swing adsorption processes. *Chem. Eng. Technol.* 593–597.
- Li, K., Mahapatra, P., Bhat, K.S., Miller, D.C., Mebane, D.S., 2017. Multi-scale modeling of an amine sorbent fluidized bed adsorber with dynamic discrepancy reduced modeling. *Reaction Chem. Eng.* 2, 550–560.
- Li, X., Li, Z., Zhu, S., 2019. Data-driven robust mean-CVaR portfolio selection under distribution ambiguity AU - Kang, Zhilin. *Quant. Finance* 19, 105–121.
- Li, Z., Ding, R., Floudas, C.A., 2011. A comparative theoretical and computational study on robust counterpart optimization: I. Robust linear optimization and robust mixed integer linear optimization. *Ind. Eng. Chem. Res.* 50, 10567–10603.
- Li, Z., Tang, Q., Floudas, C.A., 2012. A comparative theoretical and computational study on robust counterpart optimization: II. Probabilistic guarantees on constraint satisfaction. *Ind. Eng. Chem. Res.* 51, 6769–6788.
- Lucidi, S., Maurici, M., Paulon, L., Rinaldi, F., Roma, M., 2016. A derivative-free approach for a simulation-based optimization problem in healthcare. *Optim. Lett.* 10, 219–235.
- Ma, Y.N., Chen, X., Biegler, L.T., 2018. Monte-Carlo-simulation-based optimization for copolymerization processes with embedded chemical composition distribution. *Comput. Chem. Eng.* 109, 261–275.
- Marques, C.M., Moniz, S., de Sousa, J.P., Barbosa-Povoa, A.P., 2017. A simulation-optimization approach to integrate process design and planning decisions under technical and market uncertainties: a case from the chemical-pharmaceutical industry. *Comput. Chem. Eng.* 106, 796–813.
- Martin Abadi, A. A., Paul Barham, Eugene Brevdo, Zhifeng Chen, C. C., Greg S. Corrado, Andy Davis, Jeffrey Dean, M. D., Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, G. I., Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, M. K., Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, S. M., Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, I. S., Kunal Talwar, Paul Tucker, Vincent Vanhoucke, V. V., Fernanda Viégas, Oriol Vinyals, P. W., Martin Wattenberg, Martin Wicke, Yuan Yu, & X. Z. TensorFlow: large-scale machine learning on heterogeneous systems. In: *arXiv preprint arXiv:1603.04467* (2016).
- Matthews, L.R., Guzman, Y.A., Floudas, C.A., 2018. Generalized robust counterparts for constraints with bounded and unbounded uncertain parameters. *Comput. Chem. Eng.* 116, 451–467.
- McKay, M.D., Beckman, R.J., Conover, W.J., 1979. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 21, 239–245.
- Na, J., Lim, Y., Han, C., 2017. A modified direct algorithm for hidden constraints in an LNG process optimization. *Energy* 126, 488–500.
- Negrellos-Ortiz, I., Flores-Tlacuahuac, A., Gutierrez-Limon, M.A., 2016. Product dynamic transitions using a derivative-free optimization trust-region approach. *Ind. Eng. Chem. Res.* 55, 8586–8601.
- Negrellos-Ortiz, I., Flores-Tlacuahuac, A., Gutierrez-Limon, M.A., 2018. Dynamic optimization of a cryogenic air separation unit using a derivative-free optimization approach. *Comput. Chem. Eng.* 109, 1–8.
- Ning, C., You, F., 2018. Data-driven decision making under uncertainty integrating robust optimization with principal component analysis and kernel smoothing methods. *Comput. Chem. Eng.* 112, 190–210.
- Novak, E., Ritter, K., 1996. Global Optimization Using Hyperbolic Cross Points, 7.
- Novak, E., Wozniakowski, H., Novak, E., Wozniakowski, H., 2010. Smolyak/Sparse Grid Algorithms, 12.
- Nuchitprasittichai, A., Cremaschi, S., 2013. An algorithm to determine sample sizes for optimization with. *Process Syst. Eng.* 805–812.
- Palmer, K., Realff, M., 2002. Metamodeling approach to optimization of steady-state flowsheet simulations: model generation. *Chem. Eng. Res. Design* 80, 760–772.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830.
- Peherstorfer, B., Kowitz, C., Pflüger, D., Bungartz, H.J., 2015. Selected recent applications of sparse grids. *Numer. Math.-Theory Methods. Appl.* 8, 47–77.
- Plaskota, L., Wasilkowski, G.W., 2004. Smolyak's algorithm for integration and L_1 -approximation of multivariate functions with bounded mixed derivatives of second order. *Numer. Algorithms* 36, 229–246.
- Powell, M.J.D., 2002. UOBYQA: unconstrained optimization by quadratic approximation. *Math. Program* 92, 555–582.
- Quirante, N., Caballero, J.A., 2016. Large scale optimization of a sour water stripping plant using surrogate models. *Comput. Chem. Eng.* 92, 143–162.
- Rios, L.M., Sahinidis, N.V., 2013. Derivative-free optimization: a review of algorithms and comparison of software implementations. *J. Global Optim.* 56, 1247–1293.
- Rossger, P., Richter, A., 2018. Performance of different optimization concepts for reactive flow systems based on combined CFD and response surface methods. *Comput. Chem. Eng.* 108, 232–239.
- Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- Schmidhuber, J., 2015. Deep learning in neural networks: an overview. *Neural Netw.* 85–117.
- Schweidtmann, A.M., Mitsos, A., 2019. Deterministic global optimization with artificial neural networks embedded. *J. Optim. Theory Appl.* 180, 925–948.
- Scott, J.K., Barton, P.I., 2013. Bounds on the reachable sets of nonlinear control systems. *Automatica* 49, 93–100.
- Shang, C., Huang, X., You, F., 2017. Data-driven robust optimization based on kernel learning. *Comput. Chem. Eng.* 106, 464–479.
- Shashaani, S., Hashemi, F.S., Pasupathy, R., 2018. ASTRO-DF: a class of adaptive sampling trust-region algorithms for derivative-free stochastic optimization. *Siam J. Optim.* 28, 3145–3176.
- Shen, K., Scott, J.K., 2017. Rapid and accurate reachability analysis for nonlinear dynamic systems by exploiting model redundancy. *Comput. Chem. Eng.* 106, 596–608.
- Sinha, A., Darunte, L.A., Jones, C.W., Realff, M.J., Kawajiri, Y., 2017. Systems design and economic analysis of direct air capture of CO₂ through temperature vacuum swing adsorption using MIL-101(Cr)-PEI-800 and mmen-Mg₂(dobpdc) MOF adsorbents. *Ind. Eng. Chem. Res.* 56, 750–764.
- Smolyak, S.A., 1963. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR* 4, 123.
- Soyster, A.L., 1973. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* 21, 1154–1157.
- Strau, J., Skogestad, S., 2017. Use of latent variables to reduce the dimension of surrogate models. In: Espuna, A., Graells, M., Puigjaner, L. (Eds.). In: 27th European Symposium on Computer Aided Process Engineering, Pt A, 40A, pp. 445–450.
- Tang, J.J., Ni, F., Ponci, F., Monti, A., 2016. Dimension-adaptive sparse grid interpolation for uncertainty quantification in modern power systems: probabilistic power flow. *IEEE Trans. Power Syst.* 31, 907–919.
- Tawarmalani, M.A.N.V.S., 2005. A polyhedral branch-and-cut approach to global optimization. *Math. Program.* 103, 225–249.
- Thompson, M.L., Kramer, M.A., 1994. Modeling chemical processes using prior knowledge and neural networks. *AIChE J.* 40, 1328–1340.
- Valentin, J., Pflüger, D., 2016. Hierarchical gradient-based optimization with B-Splines on sparse grids. In: Garcke, J., Pflüger, D. (Eds.). In: *Sparse Grids and Applications*, 109, pp. 315–336 Stuttgart 2014.
- Van Can, H.J.L., Hellinga, C., Luyben, K.C.A.M., Heijnen, J.J., Te Braake, H.A.B., 1996. Strategy for dynamic process modeling based on neural networks in macroscopic balances. *AIChE J.* 42, 3403–3418.
- von Stosch, M., Oliveira, R., Peres, J., Feyo de Azevedo, S., 2014. Hybrid semi-parametric modeling in process systems engineering: past, present and future. *Comput. Chem. Eng.* 60, 86–101.
- Wang, H., Yuan, J., Ng, S.H., 2018. Informational approach to global optimization with input uncertainty for homoscedastic stochastic simulation. In: 2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), pp. 1396–1400.
- Wang, Z.L., Escotet-Espinoza, M.S., Singh, R., Ierapetritou, M., 2017. Surrogate-based optimization for pharmaceutical manufacturing processes. In: Espuna, A., Graells, M., Puigjaner, L. (Eds.). In: 27th European Symposium on Computer Aided Process Engineering, Pt C, 40C, pp. 2797–2802.
- Wang, Z.L., Ierapetritou, M., 2017. A novel surrogate-based optimization method for black-box simulation with heteroscedastic noise. *Ind. Eng. Chem. Res.* 56, 10720–10732.
- Wang, Z.L., Ierapetritou, M., 2018. Constrained optimization of black-box stochastic systems using a novel feasibility enhanced Kriging-based method. *Comput. Chem. Eng.* 118, 210–223.
- Wasilkowski, G.W., Wozniakowski, H., 1995. Explicit cost bounds of algorithms for multivariate tensor product problems. *J. Complex.* 11, 1–56.
- Wilson, Z.T., Sahinidis, N.V., 2017. The ALAMO approach to machine learning. *Comput. Chem. Eng.* 106, 785–795.
- Xu, G.Q., 2015. On weak tractability of the Smolyak algorithm for approximation problems. *J. Approx. Theory* 192, 347–361.
- Yuan, Y., Li, Z., Huang, B., 2016. Robust optimization under correlated uncertainty: formulations and computational study. *Comput. Chem. Eng.* 85, 58–71.
- Yuste, A.J., Dorado, M.P., 2006. A neural network approach to simulate biodiesel production from waste olive oil. *Energy Fuels* 399–402.
- Zadeh, P.M., Mehmani, A., Messac, A., 2016. High fidelity multidisciplinary design optimization of a wing using the interaction of low and high fidelity models. *Optim. Eng.* 17, 503–532.
- Zhang, Q., Grossmann, I.E., Heuberger, C.F., Sundaramoorthy, A., Pinto, J.M., 2015. Air separation with cryogenic energy storage: optimal scheduling considering electric energy and reserve markets. *AIChE J.* 61, 1547–1558.
- Zhong, W., Qiao, C., Peng, X., Li, Z., Fan, C., Qian, F., 2019. Operation optimization of hydrocracking process based on Kriging surrogate model. *Control Eng. Pract.* 85, 34–40.