Surrogate Modeling for Liquid–Liquid Equilibria Using a Parameterization of the Binodal Curve

Christian Kunde [1],*, **Tobias Keßler** [1], **Steffen Linke** [2], **Kevin McBride** [3],
Kai Sundmacher [2,3] **and Achim Kienle** [1,4],*

[1] Chair for Automation/Modelling, Otto von Guericke University Magdeburg, Universitätsplatz 2,
39106 Magdeburg, Germany; tobias.kessler@ovgu.de
[2] Chair for Process Systems Engineering, Otto von Guericke University Magdeburg, Universitätsplatz 2,
39106 Magdeburg, Germany; linke@mpi-magdeburg.mpg.de (S.L.); kai.sundmacher@ovgu.de or
sundmacher@mpi-magdeburg.mpg.de (K.S.)
[3] Max Planck Institute for Dynamics of Complex Technical Systems, Process Systems Engineering,
Sandtorstraße 1, 39106 Magdeburg, Germany; mcbride@mpi-magdeburg.mpg.de
[4] Max Planck Institute for Dynamics of Complex Technical Systems, Process Synthesis and Process Dynamics,
Sandtorstraße 1, 39106 Magdeburg, Germany
* Correspondence: christian.kunde@ovgu.de (C.K.); achim.kienle@ovgu.de or
kienle@mpi-magdeburg.mpg.de (A.K.)

Abstract: Computational effort and convergence problems can pose serious challenges when employing advanced thermodynamic models in process simulation and optimization. Data-based surrogate modeling helps to overcome these problems at the cost of additional modeling effort. The present work extends the range of methods for efficient data-based surrogate modeling of liquid–liquid equilibria. A new model formulation is presented that enables smaller surrogates with box-constrained input domains and reduced input dimensions. Sample data are generated efficiently by using numerical continuation. The new methods are demonstrated for the surrogate modeling and optimization of a process for the hydroformylation of 1-decene in a thermomorphic multiphase system.

1. Introduction

In chemical process design, advanced thermodynamic models offer increasing predictive accuracy for physical properties of mixtures. However, this correlates with increasingly complex models, which introduce serious challenges when used for process simulation and optimization. Problems may arise from embedded iterative solution procedures for the thermodynamic models, which may not converge to a valid solution or require a large number of iteration steps. The benefits of advanced thermodynamic models can be used by replacing them with surrogates that are computationally easy to handle [1].

The concept of surrogate modeling comprises a number of different approaches [2] that depend on the type of the original model and the desired properties of the surrogate. In chemical process engineering, data-driven surrogate modeling is usually employed [3]. This class of methods generally treats the original model as a black box with input–output data. The surrogate that replaces the original model is constructed such that it approximates the data and is easy to evaluate. These requirements define major steps of data-driven surrogate modeling.

In a first step, inputs and outputs have to be selected from the original model. This determines the properties of the input–output data and thereby affects both the accuracy and size of the surrogate

model. A larger input dimension increases the size of the surrogate and necessitates more sample data. A difficult input domain may require modeling with an additional surrogate [4]. In addition, output multiplicities have to be treated with special methods [5].

Several data-driven surrogate models are available, e.g., polynomials, neural networks, and Gaussian processes [2]. The choice for the surrogate model structure is usually not trivial and a number of different models should be tried. Tools for automatically selecting a surrogate model structure are available (e.g., [6]).

Parameter fitting for surrogate models requires sample data that fully characterize the input–output behavior. The additional cost for sample generation is reduced by employing efficient sampling methods. Adaptive sampling approaches that employ exploration and exploitation strategies are preferred over one-shot methods in recent literature (see [3,4] for references).

Although various strategies are available for each aspect of surrogate modeling, there is still significant potential for improvements, especially when problem-specific features are taken into account.

The present contribution extends the methods of surrogate modeling for liquid–liquid phase equilibria (LLE). This includes efficient sample generation by employing a numerical continuation method and an efficient surrogate formulation that has a box-constrained input domain and a reduced input dimension compared to established formulations. In Section 2, LLE modeling is introduced briefly and a description of surrogate modeling of LLE is given that summarizes recent work on the topic. Section 3 describes the methods of sample data generation and surrogate modeling proposed in this work. A case study based on a process for the hydroformylation of 1-decene in a thermomorphic multiphase system is used in Section 4 to demonstrate the proposed methods. A short conclusion is given in Section 5.

2. Background

This section summarizes the required background for LLE modeling and recent work on surrogate modeling of LLE.

2.1. Liquid–Liquid Equilibrium Modeling

The liquid–liquid equilibrium is described by a two-phase flash model comprising component mass balance conditions

$$N_i^{\text{in}} = N_i^{\text{I}} + N_i^{\text{II}}, \quad i = 1, \ldots, N_{\text{c}}, \tag{1}$$

and thermodynamic equilibrium conditions

$$T^{\text{I}} = T^{\text{II}} = T, \tag{2}$$

$$p^{\text{I}} = p^{\text{II}} = p, \tag{3}$$

$$\mu_i^{\text{I}} = \mu_i^{\text{II}}, \quad i = 1, \ldots, N_{\text{c}}. \tag{4}$$

Here, N_i^j is the mole amount, μ_i^j is the chemical potential, T^j is the temperature and p^j is the pressure in phase $j \in \{\text{I}, \text{II}\}$ for component i. The mole amounts of the feed mixture are denoted by N_i^{in} and the total number of components in the mixture is N_{c}.

The conditions for equal chemical potential can be reformulated for liquid–liquid equilibria as the isoactivity conditions

$$x_i^{\text{I}} \gamma_i^{\text{I}}(\boldsymbol{x}^{\text{I}}, T, p) = x_i^{\text{II}} \gamma_i^{\text{II}}(\boldsymbol{x}^{\text{II}}, T, p), \quad i = 1, \ldots, N_{\text{c}}, \tag{5}$$

with mole fractions x_i^j, vectors of mole fractions $\boldsymbol{x}^j$, and activity coefficients γ_i^j.

Iterative procedures are usually utilized to find a solution of the implicit equilibrium model, evaluating the activity coefficients provided by the thermodynamic model in each iteration. Therefore, using advanced models such as the perturbed-chain statistical associating fluid theory (PC-SAFT) [7] or universal quasichemical functional-group activity coefficients (UNIFAC) [8] directly for flowsheet simulation and optimization is unfavorable in terms of convergence behavior and computational effort. Furthermore, excluding undesired trivial solutions of the equation system given above requires additional effort. The following section describes the technique of replacing such expensive calculations by data-driven surrogate models that are computationally easier to handle.

2.2. Surrogate Modeling of LLE

Surrogate modeling aims at replacing a function f,

$$\{(x, f(x)) : x \in X\}, \tag{6}$$

with a *surrogate* function $\hat{f}$ that is similar to the original function f and exhibits favorable properties, e.g., much lower computational effort. More specifically, data-driven surrogate modeling generally uses samples $(x_k, f(x_k))$ to fit parameters of the surrogate $\hat{f}$ such that some performance criterion is satisfied. If an approximation of the original function $\hat{f} \approx f$ is desired, typical performance criteria include minimizing the mean squared error or mean absolute error with respect to the samples. Another performance criterion is the exact reproduction of the sample points $\hat{f}(x_k) = f(x_k)$, which can be used for instance in Kriging interpolation (e.g., [9]). In this sense, data-driven surrogate modeling is another name for nonlinear regression and interpolation methods. Applications for surrogate modeling include, but are not limited to, cases where the original function is expensive to evaluate, e.g., computational fluid dynamics simulations, or where a functional form may not be available, e.g., experimental results.

This section gives a summary of recent efforts [4,10] on using data-driven surrogate models for phase equilibrium calculations, particularly for LLE, to improve computational performance while preserving high model accuracy. For the sake of readability, some details of each approach are omitted here and we refer to the original publications for comprehensive descriptions. A broader overview on general surrogate modeling is available in the literature [2,3,11,12].

2.2.1. Selection of Inputs and Outputs

Distribution coefficients $K_i := N_i^{\mathrm{I}} / N_i^{\mathrm{in}}$ are introduced, which allows one to rewrite the equilibrium model as

$$N_i^{\mathrm{I}} = K_i N_i^{\mathrm{in}}, \quad i = 1, \ldots, N_{\mathrm{c}}, \tag{7}$$

$$K_i = f_i(x_1^{\mathrm{in}}, \ldots, x_{N_{\mathrm{c}}-1}^{\mathrm{in}}, T), \quad i = 1, \ldots, N_{\mathrm{c}}, \tag{8}$$

$$x_i^{\mathrm{in}} = \frac{N_i^{\mathrm{in}}}{\sum_{l=1}^{N_{\mathrm{c}}} N_l^{\mathrm{in}}}, \quad i = 1, \ldots, N_{\mathrm{c}}-1, \tag{9}$$

$$N_i^{\mathrm{II}} = N_i^{\mathrm{in}} - N_i^{\mathrm{I}}, \quad i = 1, \ldots, N_{\mathrm{c}}. \tag{10}$$

In the model comprising Equations (7)–(10), f_i is replaced by a surrogate $\hat{f}_i \approx f_i$. Kriging interpolation is used for the surrogates $\hat{f}_i$ in [4,10]. The inputs of the surrogate are $N_{\mathrm{c}} - 1$ mole fractions of the feed mixture and the temperature. The outputs comprise the N_{c} distribution coefficients. The pressure does not appear as an input since its influence on the liquid–liquid equilibrium is assumed to be negligible.

It should be emphasized that, given a feasible feed mixture and temperature, all quantities in Equations (7)–(10) can be calculated in a sequential fashion.

2.2.2. Description of the Input Domain X

A natural choice for the input domain, given molar fractions and the temperature as inputs, would be

$$\tilde{X} = \{(x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T) \in [0,1]^{N_c-1} \times \mathbb{R}^+ : x_1^{\text{in}} + \ldots + x_{N_c-1}^{\text{in}} \leq 1\}. \tag{11}$$

However, the distribution coefficient model in Equations (7)–(10) is only valid within the miscibility gap of the mixture. Therefore, box constraints are not suitable to accurately describe the complete input domain X of the surrogate $\hat{f}_i$. Instead, the input domain can be defined by a classifier $g(x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T)$, which is less than zero only within the miscibility gap:

$$X = \{(x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T) \in \tilde{X} : g(x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T) \leq \mathbf{0}\}. \tag{12}$$

Possible choices for the classifier model include a set of linear expressions $\boldsymbol{g} = \boldsymbol{A} (x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T)^\top - \boldsymbol{b}$ as in [10] or a scalar function $g(x_1^{\text{in}}, \ldots, x_{N_c-1}^{\text{in}}, T)$ as in [4]. Since the scalar classifier g in [4] is as difficult to calculate as the equilibrium model, it is approximated by another surrogate model $\hat{g} \approx g$, in this case a support vector machine.

2.2.3. Sample Data and Surrogate Generation

The selection of sample locations for data generation can be categorized in one-shot space-filling designs, which select locations according to pre-defined strategies, and adaptive sampling approaches, which exploit information from available sampling data to select new sampling locations. In [10], sample values are calculated by evaluating the equilibrium model at pre-defined sample locations generated by a space-filling design. A major contribution of Nentwich and Engell [4] is a method of adaptively choosing new sample locations based on available sample data to improve the quality of the approximation compared to one-shot designs with the same number of samples. In principle, new sample locations are chosen by maximizing a weighted sum of the minimum distance to existing sample locations and the predicted variance, i.e., by exploration and exploitation strategies. The surrogate model needs to be updated in each iteration of this method.

In both references, the sample generation takes a substantial amount of time. For a constant sample density, the number of required samples scales exponentially with the dimension of the input domain. In addition, the equilibrium model is solved at each sample location independently, i.e., the solution algorithm needs to converge starting from an independent initial guess.

3. Methods

The literature approaches to surrogate modeling of LLE rely on a regression model for the distribution of components between phases and an additional classification model to determine whether a given set of input values lies in the two-phase region. This section describes new approaches aiming at extending the pool of methods for surrogate modeling of LLE. For this, a model reformulation that enables smaller surrogate models and a method for efficiently calculating sample data are presented.

3.1. Parameterization of Binodal Curves

The strategy for reducing the computational effort for surrogates in LLE modeling presented here comprises two ideas. First, instead of modeling the entire miscibility gap, only the binodal curve is described by a surrogate, which reduces the surrogate input dimension by one. As shown in Figure 1 for a ternary mixture, the binodal curve has one dimension fewer than the miscibility gap, i.e., it is a line instead of an area. Note that for more than three components $N_c > 3$, the set of equilibrium compositions is a $(N_c - 2)$-dimensional surface. For a constant sample density, the number of required

sample points for a surrogate model depends exponentially on its input dimension. Reducing the number of inputs therefore improves computational tractability of the surrogate.

Second, instead of using molar fractions as inputs of the surrogate, the position on the binodal curve is introduced as a new coordinate, marked by t_1 in Figure 1. Each value of this coordinate corresponds to the endpoints of a tie line, i.e., the compositions of two phases in equilibrium. The parameterization variable t_1 can be defined from the border of the phase diagram to the critical point, if the data are available, or just on a section of the binodal curve. This removes the need for a complicated (surrogate) model of the input domain, since the parameterization of the binodal curve can be chosen such that box constraints are sufficient, i.e., $t_1 \in [0, 1]$.
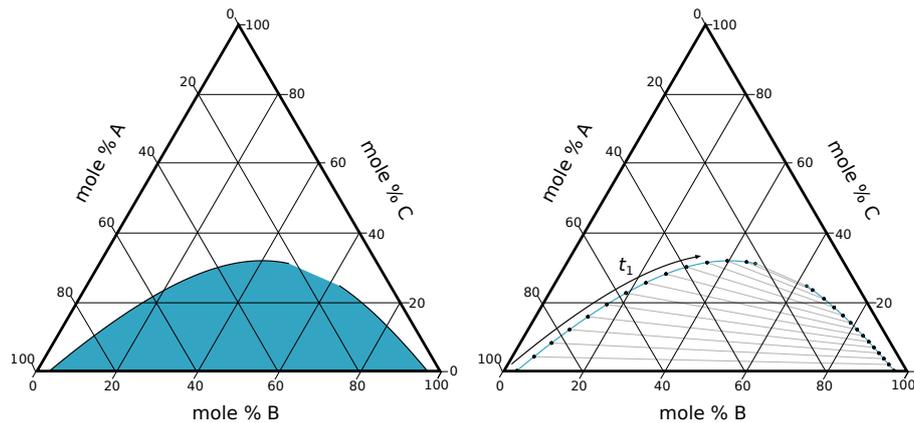
Figure 1. Sample region for liquid-liquid equilibrium (LLE) calculations of a ternary mixture, (**left**) when modeling the entire miscibility gap and (**right**) when modeling the binodal curve. The parameterization variable t_1 determines the position on the binodal curve, i.e., the pair of equilibrium compositions on the endpoints of the tie line.

Using these ideas, the equilibrium model is reformulated as follows.

$$N_i^{\text{in}} = N_i^{\text{I}} + N_i^{\text{II}}, \quad i = 1, \ldots, N_{\text{c}}, \tag{13}$$

$$\frac{N_i^j}{\sum_{l=1}^{N_{\text{c}}} N_l^j} = f_i^j(t_1, \ldots, t_{N_{\text{c}}-2}, T), \quad i = 1, \ldots, N_{\text{c}}-1, \quad j = \text{I}, \text{II} \tag{14}$$

Here, the parameterization variables for the binodal curve are denoted by $t_1, \ldots, t_{N_{\text{c}}-2}$. In the model comprising Equations (13) and (14), f_i^j is replaced by a surrogate $\hat{f}_i^j \approx f_i^j$. The inputs of the surrogate are the temperature and $N_{\text{c}} - 2$ variables for the parameterization of the binodal curve. The input domain for the surrogate functions $\hat{f}_i^j$ reads simply

$$X = [0, 1]^{N_{\text{c}}-2} \times \mathbb{R}^+. \tag{15}$$

The outputs comprise $2(N_{\text{c}} - 1)$ independent mole fractions. We decided to include all $2N_{\text{c}}$ molar fractions here, because eliminating one molar fraction by using the summation condition would make it harder to have the same error scaling for all components during surrogate fitting and it would increase the maximum error for the eliminated component in the final surrogate model.

In summary, the parameterization of binodal curves enables the generation of smaller surrogate models due to the reduction of the input dimension and the definition of the input domain by box constraints. The overall model is rendered implicit by this reformulation. Implicit models are generally harder to solve than explicit models. However, the reduced model size, and thereby the reduced number of nonlinear expressions that have to be evaluated, is favorable for the computational performance.

3.2. Numerical Continuation for Sample Calculation

Calculations of phase equilibria, especially for LLE, often require considerable computational effort due to iterative solution procedures and in some cases good initial guesses to calculate valid equilibrium compositions. Therefore, fast surrogates are employed for simulation and optimization. The original equilibrium model is only used during sample generation. However, the samples need to cover the entire miscibility gap to enable accurate surrogate modeling and their acquisition can take a considerable amount of time.

The computational cost of sample generation can be reduced by exploiting existing sample data. For instance, in [4], new sample locations are chosen such that they are most likely to improve the surrogate model, thus reducing the overall number of required sample points.

The basic idea followed in the present work is using numerical continuation methods (e.g., [13]) to improve convergence behavior of equilibrium calculations and thereby to reduce the computation time. Continuation methods are used to find solutions of an equation system that depend on a parameter, e.g., solutions of the equilibrium model depending on the feed composition. Varying this parameter, a prediction for the new solution is calculated based on previous solutions and then the prediction is corrected. Since initial guesses are improved using systematic predictions, the correction step converges more quickly to an accurate solution.

In an earlier work [14], the phase stability of an arbitrary point in the phase diagram is tested by tracking valid solutions of the equilibrium model from a known starting composition to the desired location using homotopy continuation. This strategy is adapted here to track solutions of the equilibrium model along the binodal curve by varying the feed composition.

We apply a dynamic method for minimization problems (e.g., [15]) to solve the equilibrium model for a given feed composition. The following ordinary differential equation system

$$(N_i^{\mathrm{I}})' = -k_{\mathrm{s}}\,(x_i^{\mathrm{I}}\,\gamma_i^{\mathrm{I}} - x_i^{\mathrm{II}}\,\gamma_i^{\mathrm{II}}), \quad i = 1,\ldots,N_{\mathrm{c}}, \tag{16}$$

$$(N_i^{\mathrm{II}})' = +k_{\mathrm{s}}\,(x_i^{\mathrm{I}}\,\gamma_i^{\mathrm{I}} - x_i^{\mathrm{II}}\,\gamma_i^{\mathrm{II}}), \quad i = 1,\ldots,N_{\mathrm{c}}, \tag{17}$$

converges to a valid solution if the initial guess of the phase compositions x_i^j is sufficiently close to that solution. Mole fractions are calculated by $x_i^j = \frac{N_i^j}{\sum_{l=1}^{N_{\mathrm{c}}} N_l^j}$ and the activity coefficients $\gamma_i^j = \gamma_i^j(x^j, T)$ are functions of composition and temperature. The scaling factor $k_{\mathrm{s}} := 1\,\mathrm{mol}$ ensures consistent units on both sides of the equations.

In an effort to track solutions along the binodal curve, the feed composition is varied along a line with a constant phase ratio. This trajectory, depicted in Figure 2, starts from a point on the border of the phase diagram and covers the entire miscibility gap of a ternary mixture. For all feed compositions along the trajectory, corresponding points on the binodal curve are obtained by solving the equilibrium model using Equations (16) and (17). In comparison to the previous approaches in [4,10], this strategy only requires sampling on a line in the miscibility gap instead of sampling in the whole area. If the starting point does not lie on the border of the phase diagram, the line has to be tracked in both directions from the starting point.

An efficient approach to characterize the overall phase behavior of a mixture by evaluating the convex envelope of the Gibbs energy, and thus to identify viable starting points within a miscibility gap, is given in [16].

The continuation method is implemented here by augmenting Equations (16) and (17) with additional expressions that offer convergence to the selected phase ratio and movement of the feed composition perpendicular to the current tie line. Each additional expression is chosen such that the dynamic it introduces is orders of magnitude slower than that of the equilibrium conditions. The resulting stiff ordinary differential equation system is solved using the MATLAB function *ode15s* [17].
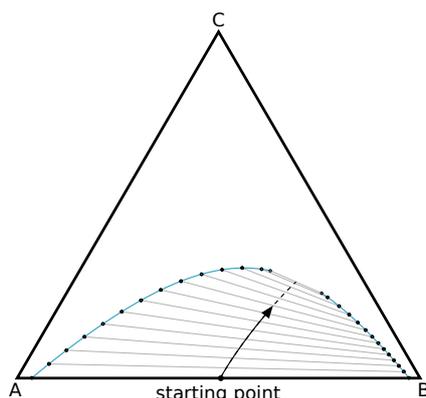
Figure 2. Sampling of the binodal curve: Trajectory of the feed composition at a constant phase ratio of 0.5 and corresponding equilibrium compositions on the binodal curve.

In the case of mixtures with more than three components, the equilibrium compositions form a $(N_c - 2)$-dimensional surface instead of a one-dimensional curve. In the present work, the continuation method is adapted to this case by fixing the molar fractions of additional components to constant values and repeat the calculations for all sets of fixed values.

In an alternative approach, instead of varying the feed composition, the compositions of both equilibrium phases could be directly tracked along the binodal curve using continuation methods [18] with Equations (16) and (17) as a possible choice for a robust corrector.

4. Results

This section demonstrates the application of the methods presented in the previous section to the modeling and optimization of a liquid–liquid extraction process.

4.1. Description of the Case Study

A multistage liquid–liquid extraction process adapted from [10] is used here as a case study for the methods presented above. In [10], the optimization of a process for the homogeneously catalyzed hydroformylation of 1-dodecene in a thermomorphic multiphase system is considered. The mixture is homogeneous at reaction conditions and biphasic at lower temperatures. This is used to separate the catalyst from the product in an extraction cascade and to recycle the catalyst back to the reactor. The extraction cascade considered here is highlighted in Figure 3, which shows a modified flowsheet of the overall process from [10].
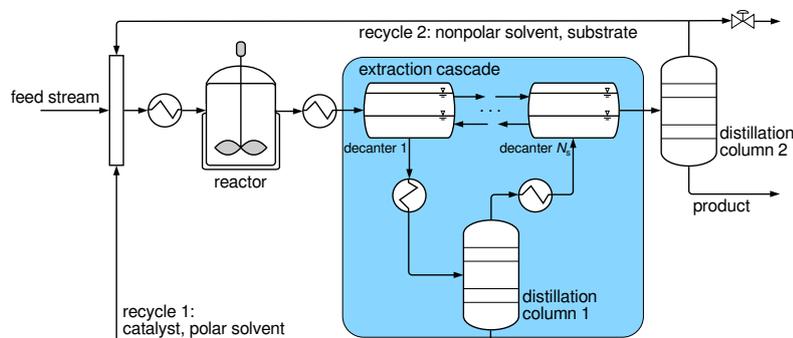
Figure 3. Modified hydroformylation process adapted from [10]. The highlighted area marks the extraction cascade with extraction solvent recycle considered in the present work.

In the present work, the mixture consists of the polar solvent dimethylformamide, the nonpolar solvent dodecane, the product n-undecanal and the substrate 1-decene. As in [10], the mixture includes

a catalyst, which is distributed between the two phases depending on their equilibrium composition. The catalyst concentration is higher in the more polar phase than in the less polar phase. Moreover, it is assumed that the catalyst has no influence on the phase equilibrium due to it having a very low concentration. Thus, the catalyst partition coefficient is modeled at infinite dilution and therefore it is added to the set of surrogate outputs, but not to the inputs. All extraction stages are operated at the same constant temperature $T = 298.15\,\text{K}$, as suggested by the results in [10].

The resulting equation system for a single extraction stage reads

$$\dot{n}_i^{\text{in}} = \dot{n}_i^{\text{I}} + \dot{n}_i^{\text{II}}, \quad i = 1, \ldots, N_{\text{c}}, \tag{18}$$

$$\dot{n}_{\text{cat}}^{\text{in}} = \dot{n}_{\text{cat}}^{\text{I}} + \dot{n}_{\text{cat}}^{\text{II}}, \tag{19}$$

$$\frac{\dot{n}_i^j}{\sum_{l=1}^{N_{\text{c}}} \dot{n}_l^j} = f_i^j(t_1, \ldots, t_{N_{\text{c}}-2}), \quad i = 1, \ldots, N_{\text{c}}, \quad j = \text{I, II}, \tag{20}$$

$$\log_{10}\left(\frac{\dot{n}_{\text{cat}}^{\text{I}} \sum_{l=1}^{N_{\text{c}}} \dot{n}_l^{\text{II}}}{\dot{n}_{\text{cat}}^{\text{II}} \sum_{l=1}^{N_{\text{c}}} \dot{n}_l^{\text{I}}} \right) = f_{\text{cat}}(t_1, \ldots, t_{N_{\text{c}}-2}), \tag{21}$$

with $\dot{n}_i^j$ denoting molar flows of component i in phase j and f_{cat} calculating the logarithmic partition coefficient $\log_{10} P_{12}$ for the catalyst between the less polar phase I and the more polar phase II. Table 1 lists the names of the components corresponding to the index i.

Table 1. Component labels.

Index i	Name	Description
1	dimethylformamide	polar solvent
2	dodecane	nonpolar solvent
3	n-undecanal	product
4	1-decene	substrate

In the model consisting of Equations (18)–(21), f_i^j and f_{cat} are replaced by a surrogate that comprises both the mole fractions $\hat{f}_i^j \approx f_i^j$ and the logarithmic partition coefficient $\hat{f}_{\text{cat}} \approx f_{\text{cat}}$. With $N_{\text{c}} = 4$, the inputs of the surrogate are two parameterization variables t_1 and t_2. The outputs are eight mole fractions for the equilibrium compositions in both phases as well as the catalyst partition coefficient.

The extraction cascade consists of $k = 1, .., N_{\text{s}}$ stages, numbered from $k = 1$ for the output stage of the more polar phase II to $k = N_{\text{s}}$ for the output stage of the less polar phase I. The stage $k = 1$ is also the feed stage. Some of the polar solvent is separated using distillation column 1 and recycled to the extraction cascade at stage $k = N_{\text{s}}$. The mass balances that describe the extraction cascade read

$$\dot{n}_{i,1}^{\text{in}} = \dot{n}_{i,2}^{\text{II}} + \dot{n}_i^{\text{feed}}, \quad i = 1, \ldots, N_{\text{c}}, \tag{22}$$

$$\dot{n}_{\text{cat},1}^{\text{in}} = \dot{n}_{\text{cat},2}^{\text{II}} + \dot{n}_{\text{cat}}^{\text{feed}}, \tag{23}$$

$$\dot{n}_{i,k}^{\text{in}} = \dot{n}_{i,k-1}^{\text{I}} + \dot{n}_{i,k+1}^{\text{II}}, \quad i = 1, \ldots, N_{\text{c}}, \quad k = 2, \ldots, N_{\text{s}}-1, \tag{24}$$

$$\dot{n}_{\text{cat},k}^{\text{in}} = \dot{n}_{\text{cat},k-1}^{\text{I}} + \dot{n}_{\text{cat},k+1}^{\text{II}}, \quad k = 2, \ldots, N_{\text{s}}-1, \tag{25}$$

$$\dot{n}_{i,N_{\text{s}}}^{\text{in}} = \dot{n}_{i,N_{\text{s}}-1}^{\text{I}} + \dot{n}_i^{\text{col1}}, \quad i = 1, \ldots, N_{\text{c}}, \tag{26}$$

$$\dot{n}_{\text{cat},N_{\text{s}}}^{\text{in}} = \dot{n}_{\text{cat},N_{\text{s}}-1}^{\text{I}}, \tag{27}$$

where $\dot{n}_i^{\text{col}}$ is the distillate molar flow of distillation column 1 that recycles the extraction solvent and $\dot{n}_i^{\text{feed}}$ is the feed flow of the extraction cascade for component i.

The mass balances of the distillation column are replaced by the approximation that the distillate of column 1 contains only the polar solvent $\dot{n}^{\text{col}} = \dot{n}_1^{\text{col}}$. The distillate flow is a degree of freedom for

this process. A larger distillate flow is able to extract more catalyst, but requires, e.g., more heat for vaporization, a larger distillation column, and larger extraction stages.

The costs considered for this process comprise the catalyst that is retained in the nonpolar product stream, the investment for the decanter vessels, and the operation and investment for distillation column 1. The catalyst costs are proportional to the amount of catalyst that is lost. The investment costs for the distillation column and the decanter vessels are calculated based on [19]. The sizing of the column utilizes the Fenske–Underwood–Gilliland correlations [20–23] with mean mixture and component properties. A simplified cost function is generated by fitting a polynomial function to data obtained by evaluating the cost calculation for a set of different compositions of the column feed and different values of the distillate flow rate at a distillate purity of 99 % (see [24] for more details). The overall cost J in $ year^{-1} for this process reads

$$J = \kappa_1 \sum_{k=1}^{N_s} \sum_{i=1}^{N_c} (\lambda_i \, \dot{n}_{i,k}^{\text{in}})^{\kappa_2} + \kappa_3 \, (\dot{n}_1^{\text{col1}})^2 + \kappa_4 \, \dot{n}_1^{\text{col1}} + \kappa_5 + \kappa_6 \, \dot{n}_{\text{cat},1}^{\text{I}}, \tag{28}$$

with parameters given in Table 2.

**Table 2.** Cost function parameters.

Parameter	Value	Unit	Parameter	Value	Unit
λ_1	0.0791506×10^{-3}	mol^{-1} s	κ_1	504155	$ year^{-1}
λ_2	0.233642×10^{-3}	mol^{-1} s	κ_2	0.586667	1
λ_3	0.210410×10^{-3}	mol^{-1} s	κ_3	-328.230	mol^{-2} s^2 $ year^{-1}
λ_4	0.195184×10^{-3}	mol^{-1} s	κ_4	27175.9	mol^{-1} s $ year^{-1}
			κ_5	10439.2	$ year^{-1}
			κ_6	4.44896×10^{12}	mol^{-1} s $ year^{-1}

4.2. Data Generation

The method presented in Section 3.2 is used to generate sample data for f_i^j. The activity coefficients are calculated utilizing a modified UNIFAC (Dortmund) model [8] with some parameters fit to experimental data as in [10]. Recall that, for four components, one molar fraction is required to be fixed to a constant value for each continuation run. Therefore, a set of binodal curves is calculated for different fixed values of the molar fraction of 1-dodecene in the less polar phase $x_4^{\text{I}} \in \{0, 0.01, \ldots, 0.25\}$. Each continuation run is initialized at the boundary of the phase diagram with zero n-undecanal, i.e., $x_3^{\text{I}} = x_3^{\text{II}} = 0$. The starting point is calculated by solving the dynamic equation system using ode15s in MATLAB with an arbitrary initial guess for the molar fractions of the polar solvent and the nonpolar solvent. Beginning at the calculated starting point, the feed composition is continued along the line of a constant phase ratio of 0.5. The continuation is stopped when the condition $x_1^{\text{I}} = x_2^{\text{I}}$ is met, i.e., when the amount of polar solvent in the less polar phase is equal to that of the nonpolar solvent. This stopping criterion is chosen here because the extraction of catalyst is ineffective beyond that point due to low values of the partition coefficient. An example of the resulting binodal curves is shown in Figure 4.

The total number of 26 continuation runs produce more than 4×10^4 samples with an accuracy of $\max_i(|x_i^{\text{I}} \gamma_i^{\text{I}} - x_i^{\text{II}} \gamma_i^{\text{II}}|) \leq 1 \times 10^{-9}$. The calculations require fewer than 2×10^5 calls to the UNIFAC implementation of the activity coefficient model $(\gamma_1^j, \ldots, \gamma_{N_c}^j) = f_{\text{UNIFAC}}(x^j, T, p)$ and a computation time of less than 1 min on a standard desktop computer without utilizing parallelization. For each binodal curve with a fixed value of x_4^{I}, this corresponds to more than 1500 samples and less than 2 s of computation time. Calculating each starting point requires approximately 1000 function calls to the activity coefficient model and each binodal curve requires fewer than 5000 function calls. In other words, the continuation method required fewer than four function calls per sample point. Note that

each evaluation of the right-hand-side of Equations (16) and (17) requires two function calls to the activity coefficient model, one for each phase.
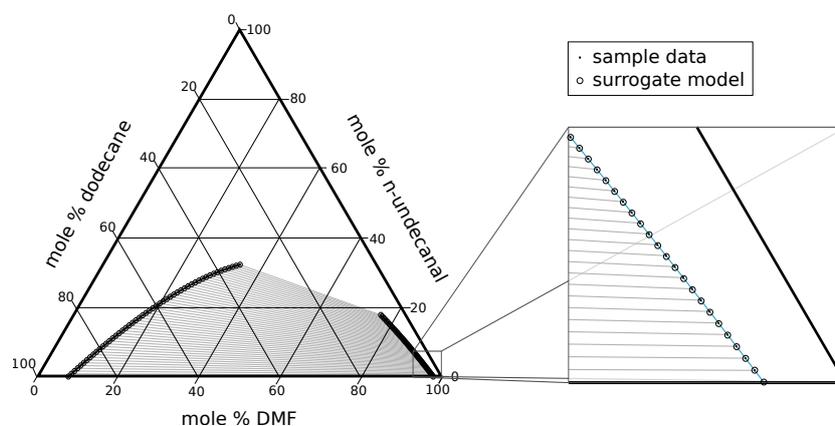
Figure 4. Results of sample data generation and surrogate fitting. The binodal curve is shown for $x_4^{\mathrm{I}} = 0$, i.e., for a mixture without 1-dodecene. The surrogate used for this figure has $N_{\mathrm{neurons}} = 10$ neurons in the hidden layer of the artificial neural network (ANN) (see Table 3).

The results from f_i^j are used in COSMOtherm [25] to calculate the logarithmic partition coefficient of the catalyst f_{cat} using the same parameterization and catalyst COSMO input file as in [10]. Continuation is not available for COSMOtherm. Therefore, the logarithmic partition coefficient needs to be calculated separately for each phase composition, which takes a few seconds per sample point. To limit the computational effort, only 51 equally spaced sample points are used for each binodal curve.

The overall dataset used for the surrogate generation comprises $51 \times 26 = 1326$ sample locations on a regular grid for the input variables $(t_1, t_2) \in [0,1]^2$. The variables (t_1, t_2) determine the equilibrium compositions of the two phases, i.e., the position on the two-dimensional binodal "curve". The value of the molar fraction of 1-dodecene in the less polar phase is defined as $x_4^{\mathrm{I}} := 0.25\, t_2$. At each sample location, the output vector contains nine values, namely the phase compositions and the logarithmic catalyst partition coefficient:

$$(x^{\mathrm{I}}, x^{\mathrm{II}}, \log_{10} P_{12})^\top = f(t_1, t_2), \tag{29}$$

with $f = (f_1^{\mathrm{I}}, \dots, f_{N_c}^{\mathrm{I}}, f_1^{\mathrm{II}}, \dots, f_{N_c}^{\mathrm{II}}, f_{\mathrm{cat}})^\top$. Because the definition for t_2 makes interpolating x_4^{I} redundant, only eight outputs are used for the surrogate model generation, which corresponds to $1326 \times 8 = 10{,}608$ sample values.

4.3. Surrogate Fitting

In the present work, the surrogate model is constructed as a sum of a second order polynomial function and a shallow artificial neural network (ANN) [26], i.e., an ANN with one hidden layer:

$$\hat{f}(t_1, t_2) = \hat{f}_{\mathrm{poly}}(t_1, t_2) + \hat{f}_{\mathrm{ANN}}(t_1, t_2). \tag{30}$$

The polynomial function captures most of the information contained in the data using only simple expressions. The ANN is employed to further reduce the interpolation error of the surrogate model. The second-order polynomial function

$$\hat{f}_{\mathrm{poly}}(t_1, t_2) = a_{00} + a_{10}\, t_1 + a_{01}\, t_2 + a_{11}\, t_1\, t_2 + a_{20}\, t_1^2 + a_{02}\, t_2^2 \tag{31}$$

is fitted to the original data in three steps. First, for each output f_i, all parameters a_{ij} of Equation (31) are determined by minimizing the sum of squared errors. Second, all terms with a low contribution to an output value are set to zero for that output value to reduce the number of nonlinear expressions, i.e.,

$$\text{if} \quad a_{ij \neq 00} \leq 0.05 \max_{kl \neq 00} |a_{kl}|, \quad \text{then} \quad a_{ij} := 0. \tag{32}$$

Third, the remaining parameter values are recalculated by minimizing the sum of squared errors. The polynomial function has a maximum of $6 \times 8 = 48$ nonzero parameters.

Afterwards, the shallow ANN $\hat{f}_{\text{ANN}}$ is fitted to the residual of the data $f(t_{1,k}, t_{2,k}) - \hat{f}_{\text{poly}}(t_{1,k}, t_{2,k})$. The model is fitted using the *train* command of MATLABs deep learning toolbox with the following options: the hyperbolic tangent as the activation function, the mean squared error as the performance function, data division using interleaved indices, Levenberg–Marquardt backpropagation, the regularization parameter equal to 1×10^{-11}, and a minimum gradient equal to 1×10^{-16}. The error weights $w_{\text{e},l}$ for the outputs l are chosen proportionally to the inverse squared data range over all samples $k \in \{1, \ldots, N_{\text{samples}}\}$

$$w_{\text{e},l} \propto \text{range}(f_l)^{-2}, \quad l = 1, \ldots, N_{\text{outputs}}, \tag{33}$$

with

$$\sum_{l=1}^{N_{\text{outputs}}} w_{\text{e},l} = N_{\text{outputs}}, \tag{34}$$

$$\text{range}(f_l) = \max_k f_l(t_{1,k}, t_{2,k}) - \min_k f_l(t_{1,k}, t_{2,k}), \quad l = 1, \ldots, N_{\text{outputs}}. \tag{35}$$

The remaining options are at default values. The number of neurons in the hidden layer N_{neurons} is varied from 5 to 15 to show the trade-off between the model accuracy and the number of parameters and nonlinear expressions.

The results of the parameter fitting are shown in Table 3 with the error metrics defined below. A part of the training algorithm relies on random number generation. The variations in model performance caused by this are not accounted for. The ANN training was executed once for each number of neurons N_{neurons} in the hidden layer.

Table 3. Performance metrics for the surrogate model. The sum of squared parameter values of the ANN is denoted by sumParANN^2.

N_{neurons}	# of Parameters	$e_{\text{norm}}^{\text{mean}}$	$e_{\text{norm}}^{\text{max}}$	$e_{\text{norm,poly}}^{\text{max}}$	sumParANN^2
5	103	0.00202	0.0163	0.1002	644.7
6	114	0.00227	0.0195	0.1002	2311
7	125	0.00127	0.0080	0.1002	667.7
8	136	0.00070	0.0083	0.1002	618.1
9	147	0.00058	0.0087	0.1002	311.6
10	158	0.00039	0.0038	0.1002	369.8
11	169	0.00054	0.0044	0.1002	291.9
12	180	0.00033	0.0034	0.1002	391.5
13	191	0.00030	0.0035	0.1002	357.7
14	202	0.00035	0.0064	0.1002	246.3
15	213	0.00022	0.0020	0.1002	254.9

The mean normalized error $e_{\text{norm}}^{\text{mean}}$ is the algebraic mean, over all outputs $l \in \{1, ..., N_{\text{outputs}}\}$ and samples $k \in \{1, \ldots, N_{\text{samples}}\}$, of the absolute error of an output value divided by the range of values for that output, i.e.,

$$e_{\text{norm}}^{\text{mean}} = \frac{1}{N_{\text{samples}} \, N_{\text{outputs}}} \sum_k \sum_l \frac{|\hat{f}_l(t_{1,k}, t_{2,k}) - f_l(t_{1,k}, t_{2,k})|}{\text{range}(f_l)}. \tag{36}$$

The maximum normalized error $e_{\text{norm}}^{\text{max}}$ is the maximum error of any output l over all samples k divided by the range of values for that output, i.e.,

$$e_{\text{norm}}^{\text{max}} = \max_l \left(\frac{\max_k |\hat{f}_l(t_{1,k}, t_{2,k}) - f_l(t_{1,k}, t_{2,k})|}{\text{range}(f_l)} \right). \tag{37}$$

The maximum normalized error offers a data-independent estimate of the surrogate quality, e.g., a surrogate with $e_{\text{norm}}^{\text{max}} \geq 0.5$ is not better than a constant function.

The error metric $e_{\text{norm,poly}}^{\text{max}}$ is the same as $e_{\text{norm}}^{\text{max}}$, but with the surrogate comprising only the polynomial function $\hat{f} = \hat{f}_{\text{poly}}$.

The surrogate fitting includes all molar fractions of a phase, although one of the molar fractions could be replaced by the summation condition. However, if the summation condition is used to replace one molar fraction, the worst case error for this quantity is the sum of absolute errors for all other molar fractions. This is increasingly relevant for a larger number of components. Including all molar fractions allows one to define suitable error weights for all quantities.

The results in Table 3 show that the surrogate can reproduce the data to a high degree of accuracy with a relatively low number of parameters. Based on the results for different numbers of neurons in the hidden layer, a surrogate model with $N_{\text{neurons}} = 10$ is used for the following surrogate based optimization. The mean error over all sample data and surrogate outputs normalized by the data range for this surrogate model is 0.00039 and the maximum normalized error is 0.0038. For 10608 sample values, this is achieved using 158 nonzero parameter values. There is no meaningful distinction between training set and validation set for this ratio of data points to number of parameters. The difference between surrogate model and sample data is also illustrated in Figure 4.

If only a second-order polynomial function is used as a surrogate, the maximum normalized error is 0.1002.

4.4. Optimization

In this section, the surrogate model is tested for its suitability to be used in process optimization problems. The process model introduced in Section 4.1 is used as a case study. A mixture containing four components plus a catalyst is considered for the liquid–liquid extraction at a fixed temperature and pressure. The aim is to find the optimal extraction solvent recycle such that the costs, comprising energy, investment, and catalyst loss, are minimized. Surrogate models are employed for the liquid–liquid equilibrium in each extraction stage to keep the computational burden of the optimization problem low.

The original phase equilibrium model is replaced by the surrogate model and slack variables δ that account for regression errors

$$f_i^j(t_1, t_2) := \hat{f}_i^j(t_1, t_2) + \delta_i^j, \quad i = 1, \ldots, N_{\text{c}}, \quad j = \text{I, II}, \tag{38}$$

$$f_{\text{cat}}(t_1, t_2) := \hat{f}_{\text{cat}}(t_1, t_2) + \delta_{\text{cat}}, \tag{39}$$

with the values of the slack variables lying between lower bounds $\underline{\delta}_i^j$ and upper bounds $\bar{\delta}_i^j$

$$\delta_i^j \in [\underline{\delta}_i^j, \bar{\delta}_i^j], \quad i = 1, \ldots, N_{\mathrm{c}}, \quad j = \mathrm{I}, \mathrm{II}, \tag{40}$$

$$\delta_{\mathrm{cat}} \in [\underline{\delta}_{\mathrm{cat}}, \bar{\delta}_{\mathrm{cat}}]. \tag{41}$$

The values of the upper and lower bounds are estimated by the maximum error of the surrogate model at the sample points $k \in \{1, \ldots, N_{\mathrm{samples}}\}$.

$$\bar{\delta}_i^j = \max_k f_i^j(t_{1,k}, t_{2,k}) - \hat{f}_i^j(t_{1,k}, t_{2,k}), \quad i = 1, \ldots, N_{\mathrm{c}}, \quad j = \mathrm{I}, \mathrm{II}, \tag{42}$$

$$\underline{\delta}_i^j = \min_k f_i^j(t_{1,k}, t_{2,k}) - \hat{f}_i^j(t_{1,k}, t_{2,k}), \quad i = 1, \ldots, N_{\mathrm{c}}, \quad j = \mathrm{I}, \mathrm{II}, \tag{43}$$

$$\bar{\delta}_{\mathrm{cat}} = \max_k f_{\mathrm{cat}}(t_{1,k}, t_{2,k}) - \hat{f}_{\mathrm{cat}}(t_{1,k}, t_{2,k}), \tag{44}$$

$$\underline{\delta}_{\mathrm{cat}} = \min_k f_{\mathrm{cat}}(t_{1,k}, t_{2,k}) - \hat{f}_{\mathrm{cat}}(t_{1,k}, t_{2,k}). \tag{45}$$

The optimization problem reads

$$\text{minimize} \quad J \tag{46}$$

$$\text{s.t.} \quad \text{Equations (18)–(28),} \tag{47}$$

$$\text{Equations (38)–(39),} \tag{48}$$

with domains for all variables set to appropriate values and additional redundant constraints that enable tighter bounds for deterministic global optimization. The hyperbolic tangent is formulated as $\tanh(x) = 1 - \frac{2}{1+\exp(2x)}$ as suggested in [26]. The feed stream $\dot{n}_i^{\mathrm{feed}}$ is given in Table 4. The number of stages is fixed to values between $N_{\mathrm{s}} = 1$ and $N_{\mathrm{s}} = 6$.

Table 4. Feed stream parameters.

Parameter	Value	Unit
$\dot{n}_1^{\mathrm{feed}}$	18.8	mol s^{-1}
$\dot{n}_2^{\mathrm{feed}}$	16.5	mol s^{-1}
$\dot{n}_3^{\mathrm{feed}}$	0.563	mol s^{-1}
$\dot{n}_4^{\mathrm{feed}}$	2.25	mol s^{-1}
$\dot{n}_{\mathrm{cat}}^{\mathrm{feed}}$	1×10^{-6}	mol s^{-1}

The nonlinear programming (NLP) problem is implemented in GAMS 26.1.0 [27] using the deterministic global solver BARON 18.11.12 [28] with CONOPT 4.09 [29,30] as the nonlinear programming subsolver and CPLEX 12.8.0 [31] as the mixed-integer programming subsolver, and it is solved on the same standard desktop computer as used in Section 4.2. All options are at default values except for the relative optimality criterion, which is set to 1×10^{-3}. The optimization is repeated for different numbers of stages N_{s}. The computation time for global optimization is the *total CPU time used* as reported by BARON. Performance variability [32] is not accounted for: Each optimization problem is solved once. The time and the iterations required to find a local solution are determined by solving each problem with CONOPT and using the total time elapsed as reported by GAMS. Default values chosen by GAMS are used as initial guesses for all variables. Although this initial point is always infeasible, the selected solvers found feasible solutions in all observed cases.

The results in Table 5 show that the cost of the process falls steeply until a minimum is reached at $N_{\mathrm{s}} = 4$. The cost increases slightly for $N_{\mathrm{s}} > 4$. This behavior is due to large solvent recycles being required for good catalyst retention at low numbers of stages and the relatively small costs of increasing the number of stages. The overall optimum at $N_{\mathrm{s}} = 4$ resembles a trade-off between the cost for catalyst loss and solvent recycle on one hand and the investment cost for additional stages on the other hand.

The solvent recycle for $N_s = 1$ is an exception. The feed of the single stage already contains a large amount of the extraction solvent. Increasing the solvent amount further has only a marginal effect on catalyst retention, which does not compensate for the additional cost of solvent recycle at some point.

Table 5. Optimization results.

N_s	Objective J in \$ year^{-1}	Solvent Recycle $\dot{n}_1^{col}$ in mol s^{-1}	Catalyst Retention $\dot{n}_{cat,1}^{II}/\dot{n}_{cat}^{feed}$	Iterations (CONOPT)	Solve Time (CONOPT)	Solve Time (BARON)
1	541,760	2.3524	0.9002	89	0.035 s	0.38 s
2	305,686	4.8854	0.9725	191	0.066 s	3.74 s
3	249,418	4.1435	0.9859	330	0.122 s	8.39 s
4	236,250	3.6599	0.9910	398	0.175 s	28.71 s
5	238,577	3.3459	0.9936	464	0.285 s	56.33 s
6	248,022	3.1299	0.9950	608	0.405 s	664.44 s

The surrogate model implementation is significantly faster than the original model considering that we observed computation times of a few seconds for a single COSMOtherm evaluation. The computation times using BARON show that the surrogate model formulation is also suitable for small to medium sized deterministic global optimization problems. CONOPT was able to find a global solution for each of the considered cases. BARON requires the computation time to prove globality by converging the lower bound.

5. Conclusions

In this work, we introduce a new surrogate formulation that uses a parameterization of the binodal curve to reduce the input dimension and simplify the input domain. We also propose using numerical continuation methods for sampling of LLE data based on advanced thermodynamic models. The presented methods reduce the time required for sample data generation and allow for highly accurate reproduction of the data using smaller surrogate models.

Parameterization of complex input domains and sampling by numerical continuation are general strategies that may be extended to other applications in future work. Another possible improvement is replacing the activity coefficient model currently used for LLE calculations with more predictive equation of state models. The computation time of global optimization may be further reduced by employing reduced-space formulations of the optimization problem as in [26,33,34].

Author Contributions: Conceptualization, C.K.; methodology, C.K.; software, C.K., T.K., K.M. and S.L.; investigation, C.K.; writing—original draft preparation, C.K.; writing—review and editing, C.K., T.K., K.M., A.K. and S.L.; visualization, C.K. and T.K.; supervision, A.K. and K.S.; and funding acquisition, A.K. and K.S.

References

1. Nentwich, C.; Engell, S. Application of surrogate models for the optimization and design of chemical processes. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 1291–1296. doi:10.1109/IJCNN.2016.7727346. [CrossRef]

2. Asher, M.J.; Croke, B.F.W.; Jakeman, A.J.; Peeters, L.J.M. A review of surrogate models and their application to groundwater modeling. *Water Resour. Res.* **2015**, *51*, 5957–5973. doi:10.1002/2015WR016967. [CrossRef]

3. McBride, K.; Sundmacher, K. Overview of Surrogate Modeling in Chemical Process Engineering. *Chem. Ing. Tech.* **2019**, *91*, 228–239. doi:10.1002/cite.201800091. [CrossRef]

4. Nentwich, C.; Engell, S. Surrogate modeling of phase equilibrium calculations using adaptive sampling. *Comput. Chem. Eng.* **2019**, *126*, 204–217. doi:10.1016/j.compchemeng.2019.04.006. [CrossRef]

5. Keßler, T.; Kunde, C.; McBride, K.; Mertens, N.; Michaels, D.; Sundmacher, K.; Kienle, A. Global optimization of distillation columns using explicit and implicit surrogate models. *Chem. Eng. Sci.* **2019**, *197*, 235–245. doi:10.1016/j.ces.2018.12.002. [CrossRef]

6. Boukouvala, F.; Floudas, C.A. ARGONAUT: AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems. *Optim. Lett.* **2017**, *11*, 895–913. doi:10.1007/s11590-016-1028-2. [CrossRef]

7. Gross, J.; Sadowski, G. Perturbed-Chain SAFT: An Equation of State Based on a Perturbation Theory for Chain Molecules. *Ind. Eng. Chem. Res.* **2001**, *40*, 1244–1260. doi:10.1021/ie0003887. [CrossRef]

8. Weidlich, U.; Gmehling, J. A modified UNIFAC model. 1. Prediction of VLE, h^{E}, and γ^{∞}. *Ind. Eng. Chem. Res.* **1987**, *26*, 1372–1381. doi:10.1021/ie00067a018. [CrossRef]

9. Daya Sagar, B.; Cheng, Q.; Agterberg, F. (Eds.) *Handbook of Mathematical Geosciences: Fifty Years of IAMG*; Springer International Publishing: Cham, Switzerland, 2018. doi:10.1007/978-3-319-78999-6. [CrossRef]

10. McBride, K.; Kaiser, N.M.; Sundmacher, K. Integrated reaction-extraction process for the hydroformylation of long-chain alkenes with a homogeneous catalyst. *Comput. Chem. Eng.* **2017**, *105*, 212–223. doi:10.1016/j.compchemeng.2016.11.019. [CrossRef]

11. Forrester, A.I.; Keane, A.J. Recent advances in surrogate-based optimization. *Prog. Aerosp. Sci.* **2009**, *45*, 50–79. doi:10.1016/j.paerosci.2008.11.001. [CrossRef]

12. Bhosekar, A.; Ierapetritou, M. Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Comput. Chem. Eng.* **2018**, *108*, 250–267. doi:10.1016/j.compchemeng.2017.09.017. [CrossRef]

13. Seydel, R. *Practical Bifurcation and Stability Analysis*, 3rd ed.; Springer: New York, NY, USA, 2010. doi:10.1007/978-1-4419-1740-9. [CrossRef]

14. Bausa, J.; Marquardt, W. Quick and reliable phase stability test in VLLE flash calculations by homotopy continuation. *Comput. Chem. Eng.* **2000**, *24*, 2447–2456. doi:10.1016/S0098-1354(00)00604-9. [CrossRef]

15. Brown, A.A.; Bartholomew-Biggs, M.C. Some effective methods for unconstrained optimization based on the solution of systems of ordinary differential equations. *J. Optim. Theory Appl.* **1989**, *62*, 211–224. doi:10.1007/BF00941054. [CrossRef]

16. Ryll, O.; Blagov, S.; Hasse, H. Convex envelope method for the determination of fluid phase diagrams. *Fluid Phase Equilibria* **2012**, *324*, 108–116. doi:10.1016/j.fluid.2012.04.002. [CrossRef]

17. *MATLAB Release*; The MathWorks, Inc.: Natick, MA, USA, 2018. Available online: https://www.mathworks.com/ (accessed on 14 October 2019).

18. Privat, R.; Jaubert, J.N.; Privat, Y. A simple and unified algorithm to solve fluid phase equilibria using either the gamma-phi or the phi-phi approach for binary and ternary mixtures. *Comput. Chem. Eng.* **2013**, *50*, 139–151. doi:10.1016/j.compchemeng.2012.11.006. [CrossRef]

19. Douglas, J. *Conceptual Design of Chemical Processes*; McGraw-Hill Book Company: Singapore, 1988; p. 601.

20. Fenske, M. Fractionation of Straight-Run Pennsylvania Gasoline. *Ind. Eng. Chem.* **1932**, *24*, 482–485. doi:10.1021/ie50269a003. [CrossRef]

21. Doherty, M.; Fidkowski, Z.; Malone, M.; Taylor, R. *Perry's Chemical Engineers' Handbook*; Chapter Distillation; McGraw-Hill: New York, NY, USA, 2008.

22. Gilliland, E. Multicomponent Rectification. *Ind. Eng. Chem.* **1940**, *32*, 1101–1106. doi:10.1021/ie50368a017. [CrossRef]

23. Eduljee, H. Equations replace Gilliland Plot. *Hydrocarb. Process.* **1975**, *54*, 120–122.

24. Keßler, T.; Kunde, C.; Linke, S.; McBride, K.; Sundmacher, K.; Kienle, A. Systematic Selection of Green Solvents and Process Optimization for the Hydroformylation of Long-Chain Olefines. *Process. Adv. Methods Process. Syst. Eng.* **2019**, forthcoming.

25. *COSMOtherm, C30, Release 1601*; COSMOlogic GmbH & Co. KG: Leverkusen, Germany. Available online: http://www.cosmologic.de (accessed on 14 October 2019).

26. Schweidtmann, A.M.; Mitsos, A. Deterministic Global Optimization with Artificial Neural Networks Embedded. *J. Optim. Theory Appl.* **2019**, *180*, 925–948. doi:10.1007/s10957-018-1396-0. [CrossRef]

27. *General Algebraic Modeling System (GAMS), 26.1.0*; GAMS Development Corporation: Fairfax, VA, USA. Available online: https://www.gams.com/ (accessed on 14 October 2019).

28. Kılınç, M.R.; Sahinidis, N.V. Exploiting integrality in the global optimization of mixed-integer nonlinear programming problems with BARON. *Optim. Methods Softw.* **2018**, *33*, 540–562. doi:10.1080/10556788.2017.1350178. [CrossRef]

29. Drud, A.S. CONOPT-A Large-Scale GRG Code. *ORSA J. Comput.* **1994**, *6*, 207–216. doi:10.1287/ijoc.6.2.207. [CrossRef]

30. Drud, A.S. CONOPT: A GRG Code for Large Sparse Dynamic Nonlinear Optimization Problems. *Math. Program.* **1985**, *31*, 153–191. doi:10.1007/BF02591747. [CrossRef]

31. *CPLEX Optimizer*; IBM Corporation: Armonk, NY, USA. Available online: https://www.cplex.com/ (accessed on 14 October 2019).

32. Koch, T.; Achterberg, T.; Andersen, E.; Bastert, O.; Berthold, T.; Bixby, R.E.; Danna, E.; Gamrath, G.; Gleixner, A.M.; Heinz, S.; et al. MIPLIB 2010. *Math. Program. Comput.* **2011**, *3*, 103–163. doi:10.1007/s12532-011-0025-9. [CrossRef]

33. Bongartz, D.; Mitsos, A. Deterministic global flowsheet optimization: Between equation-oriented and sequential-modular methods. *AIChE J.* **2019**, *65*, 1022–1034. doi:10.1002/aic.16507. [CrossRef]

34. Bongartz, D.; Mitsos, A. Deterministic global optimization of process flowsheets in a reduced space using McCormick relaxations. *J. Glob. Optim.* **2017**, *69*, 761–796. doi:10.1007/s10898-017-0547-4. [CrossRef]