Digital Chemical Engineering 14 (2025) 100200

Contents lists available at ScienceDirect

Digital Chemical Engineering

journal homepage: www.elsevier.com/locate/dche

Original article

Classifier surrogates to ensure phase stability in optimisation-based design of solvent mixtures

Tanuj Karia, Gustavo Chaparro, Benoît Chachuat, Claire S. Adjiman *

Department of Chemical Engineering, The Sargent Centre for Process Systems Engineering, Institute for Molecular Science and Engineering, Imperial College London, London, SW7 2AZ, United Kingdom

ARTICLE INFO

Dataset link: https://github.com/tk3016/SI_CBS_CAMbD.git

Keywords:
Computer-aided mixture/blend design
Phase stability
Mixed-integer nonlinear programming
Surrogate-based optimisation
Classifier
Artificial neural network

ABSTRACT

The ability to guarantee a single homogeneous liquid phase is a key consideration in computer-aided mixture/blend design (CAM^{b}D). In this article, we investigate the use of a classifier surrogate of the phase stability condition within a CAM^{b}D optimisation model for designing solvent mixtures with guaranteed phase stability properties. We show how to develop such classifiers for describing multiple candidate mixtures over a range of compositions and temperatures based on the generation of phase stability data using thermodynamic models such as UNIFAC. We test the approach on two solvent design case studies and illustrate its effectiveness in enabling the *in silico* design of stable mixtures, simultaneously providing a probability of phase stability as an interpretable metric.

1. Introduction

The number of phases (vapour, liquid or solid) present in a mixture at equilibrium under given conditions of temperature, pressure and global composition is an important consideration in the design of manufacturing processes and chemical products. The unexpected appearance of a new phase can impair the performance of a process or product, lead to wasted economic and material resources, or even call for a complete redesign (Xu et al., 2005). The ability to assess phase stability reliably is thus critical for digital design, yet remains particularly challenging when liquid–liquid demixing can occur.

For a mixture at a given total composition, temperature and pressure, the stability of a postulated equilibrium state – that is, a set of possible equilibrium phases with their associated compositions – can be determined using a thermodynamic model and the tangent plane distance criterion (Baker et al., 1982; Michelsen, 1982). The criterion entails the minimisation over all possible compositions of the distance between the Gibbs free energy surface and a plane that is tangent to that surface at the composition of the postulated phases. Where this difference is negative, the postulated equilibrium state cannot be stable. An alternative stability criterion, the dual extremum principle (Mitsos and Barton, 2007), requires the solution of a max–min problem derived from Lagrangian duality. Both criteria enable a rigorous assessment of phase stability provided the corresponding optimisation problems are solved to global optimality. But because describing the phase behaviour of real mixtures requires highly nonlinear thermodynamic models, these optimisation problems are inherently nonconvex and thus challenging to solve to global optimality. Much effort has been devoted to their reliable solution, with a variety of heuristic methods (Sun and Seider, 1995; Nichita et al., 2002; Elhassan et al., 1998; Pan and Firoozabadi, 1998; Pereira et al., 2012; Fonseca-Pérez et al., 2021) and deterministic methods (Stadtherr et al., 1995; McDonald and Floudas, 1997; Hua et al., 1999; Mitsos and Barton, 2007; Pereira et al., 2010) having been developed, often in the context of combined phase equilibrium and phase stability calculations.

A further difficulty arises from the context in which one wishes to test for phase stability. Phase stability is often considered part of a broader effort, for instance in process optimisation, product design or parameter estimation of thermodynamic models, where a specific type of phase behaviour is sought. Due to the nature of phase stability criteria, incorporating a rigorous phase stability constraint within an overarching optimisation problem results in a bi-level program (Mitsos and Barton, 2007; Mitsos et al., 2009), with nonconvexity present both in the lower- and upper-level programs. The simplest approach to mitigate these computational challenges is to drop the phase stability constraint in the optimisation problem and conduct a *post hoc* phase stability check. A drawback with this approach, however, is that there is no straightforward way of updating the optimisation problem in a subsequent iteration if the optimal mixture were to fail the stability test.

* Corresponding author.
E-mail address: c.adjiman@imperial.ac.uk (C.S. Adjiman).

https://doi.org/10.1016/j.dche.2024.100200
Received 11 July 2024; Received in revised form 13 November 2024; Accepted 25 November 2024
Available online 20 December 2024
2772-5081/© 2024 Published by Elsevier Ltd on behalf of Institution of Chemical Engineers (IChemE). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

In the area of computer-aided molecular and blend design (CAM^{b}D), Conte et al. (2011) proposed a systematic methodology for screening of binary mixtures that takes phase stability into account. They identified four types of Gibbs mixing functions for binary mixtures, which they used to determine whether candidate mixtures are miscible. A drawback of their approach was that partially miscible binary mixtures were discarded entirely, although some of these mixtures exhibit stability at certain compositions. Other authors have integrated phase stability as a *post hoc* test for the candidate blends identified via optimisation-based CAM^{b}D. For instance, Mah et al. (2019) considered the design of solvents to enhance the properties of bio-oil and conducted a post-optimal stability check on the candidate mixtures using the tangent plane distance criterion. Where a mixture was found to be partially miscible its phase behaviour was further analysed to determine if small changes in composition could result in a homogeneous liquid phase, otherwise the addition of binding agents was considered to achieve the desired phase behaviour. Watson et al. (2021) introduced another *post hoc* test in the form of a PT-flash algorithm with a phase stability calculation. Where a candidate mixture was found to be partially miscible, the limits of stability were computed and two revised optimisation formulations were solved based on each limit of stability to determine whether a mixture with the same components but different compositions would offer good performance. Note, however, that both approaches make the assumption that a change in composition for a mixture to achieve miscibility does not significantly impact its performance relative to other candidate mixtures.

A more direct approach would be one based on the consideration of phase stability directly as a problem constraint. To mitigate the computational challenges inherent to this approach, simplified stability conditions have been proposed. For the design of binary mixtures, Buxton et al. (1999) imposed a positivity constraint on the second derivative of the molar Gibbs free energy of the mixture with respect to one mole fraction at the target composition. While this constraint eliminates any mixture that is unstable, it still allows for metastable mixtures to be selected. A similar strategy was adopted by Karunanithi et al. (2005) and Karunanithi et al. (2006). In the case of multicomponent mixtures with up to three components, Jonuzaj et al. (2016) and Muhieddine et al. (2022) introduced one such positivity constraint for each component pair, resulting in three constraints. Such pairwise constraints do not fully eliminate the possibility of designing unstable mixtures, as certain ternary mixtures with fully miscible binary pairs can still undergo demixing (refer to Figure 10 in Binous and Bellagi (2021) for an example), so a *post hoc* test remains necessary. Dahmen and Marquardt (2017) considered the design of fuel blends and pathways with a large number of components and introduced a necessary condition in the form of a positive-definite constraint on the Hessian matrix of the Gibbs free energy of mixing at the mixture composition. They also performed a *post hoc* test to rule out metastability using the tangent plane distance criterion.

To remove the need for a *post hoc* check entirely, approaches to embedding the rigorous conditions for phase stability as a constraint within an overarching optimisation problem have also been investigated. In the context of parameter estimation, Mitsos et al. (2009) and Bollas et al. (2009) solved bi-level programs with the phase stability constraint defining the lower-level program. In the context of process design, Karacasulu et al. (2021) demonstrated the benefits of using a rigorous phase stability constraint for solving design problems embedding flash calculations for liquid–liquid equilibrium, through reformulating the bi-level program as a semi-infinite program. From a practical perspective, this approach has so far been limited to problems with a fixed total composition as it would otherwise lead to generalised semi-infinite programs whose numerical solution is even more challenging.

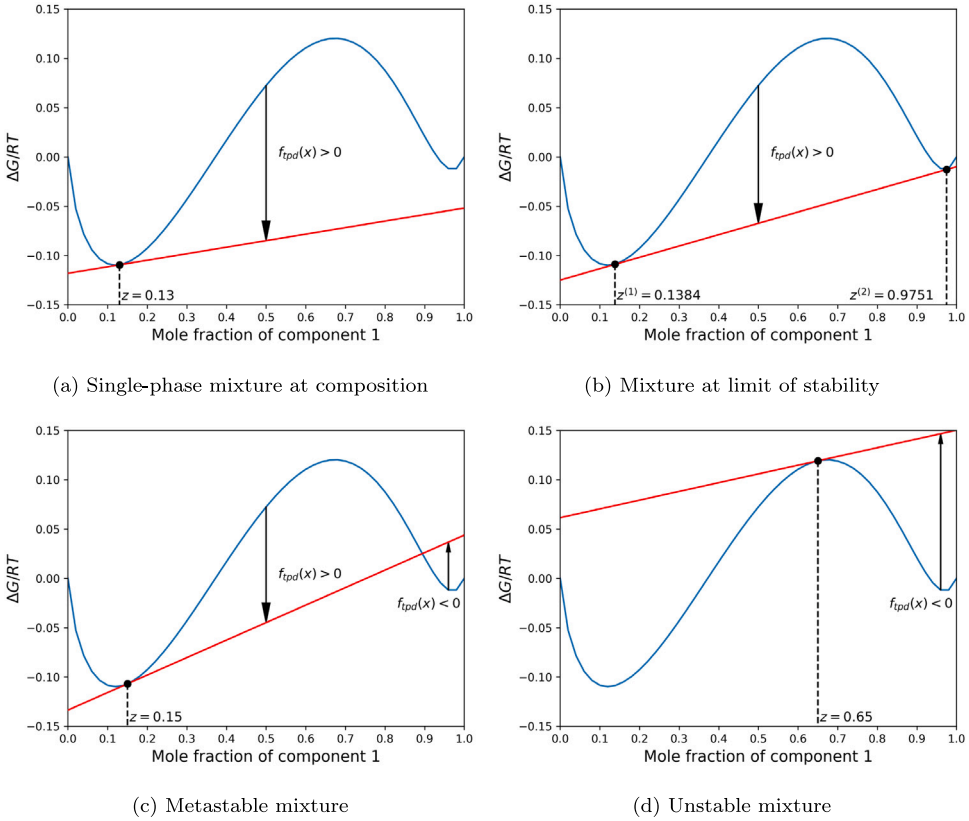
Given that phase stability constraints continue to pose a challenge for numerical optimisation, there has been growing interest in the use of machine learning to create surrogates of the rigorous stability criteria. Schmitz et al. (2006) reported the first use of a feedforward artificial neural network (ANN) classifier to quickly evaluate the number and type of stable phases for a ternary mixture of ethanol, ethylacetate and water over a range of temperatures where the mixture exhibits a heterogeneous azeotrope. Gaganis and Varotsis (2012) developed a method to distinguish between stable and unstable mixtures in two-phase equilibrium using support vector classifiers and applied it to mixtures with up to eight components across a range of pressures and temperatures. More recently, Kashinath et al. (2018) and Poort et al. (2019) extended the use of ANN surrogates for phase stability by including phase equilibrium calculations, while Kunde et al. (2019) reported the use of an ANN model of a liquid–liquid phase boundary to conduct process simulation.

To the best of our knowledge, the benefits of using a surrogate constraint of the rigorous phase stability criterion within optimisation problems have not yet been investigated, possibly due to the added numerical complexity associated with setting up and solving such optimisation problems. The recent development of several reformulations of the equations describing ANNs to facilitate their integration as constraints within optimisation problems (Grimstad and Andersson, 2019; Schweidtmann and Mitsos, 2019; Anderson et al., 2020; Tsay et al., 2021; Yang et al., 2022) opens up this possibility. Herein, we propose using a surrogate constraint based on an ANN classifier to replace the rigorous phase stability constraint in an optimisation problem. We focus on applications in solvent mixture design, where the identity of the solvents is described by discrete optimisation variables that impact the stability of the phase under consideration. This makes the representation of phase stability especially challenging as the phase boundaries vary not only with the state of the system (composition, temperature, and pressure) but also with the identity of the components in the mixture. This is in contrast with previous work where the number and identities of the components in the mixture were set and the surrogate model could then be trained on the phase behaviour of the single mixture of interest. The proposed methodology is exemplified for the UNIFAC model (Fredenslund et al., 1975) but is, in principle, applicable to any parametric thermodynamic model as we make use of the relationships between the solvent (component) identities and the thermodynamic model parameters. The classifiers are thus trained by taking a continuous representation of the solvent identities, mixture composition and temperature as inputs to predict the probability of there being a homogeneous liquid phase as the output. The performance of the classifier is tested on two solvent design case studies related to the crystallisation of ibuprofen (Jonuzaj et al., 2016; Watson et al., 2021), with up to three components and three phases present – one or two liquid phases and/or a solid phase. We demonstrate that by using a classifier-based surrogate modelling framework, we can account for the presence of liquid–liquid and solid–liquid–liquid phase regions directly within the optimisation framework to reduce the likelihood of designing solvent mixtures that undergo liquid–liquid demixing at the initial and final states of crystallisation.

The rest of the paper is organised as follows. The background on phase stability is explained in Section 2. The methodology for developing a classifier-based surrogate model for phase stability calculations and their integration within the design problem are outlined in Section 3. The performance of the surrogate models on the two selected case studies is discussed in Section 4. Finally, Section 5 concludes the paper.

2. Preliminaries

This section summarises the phase stability criteria used in our work, including the rigorous tangent plane distance (TPD) criterion and a number of approximate formulations commonly adopted in CAM^{b}D. It also details the numerical procedures and software used for rigorous phase stability calculations.



(a) Single-phase mixture at composition

(b) Mixture at limit of stability

(c) Metastable mixture

(d) Unstable mixture

Fig. 1. Geometrical interpretation of TPD criterion for a binary mixture based on the distance between the Gibbs free energy curve (in blue) and the tangent when the mole fraction of component 1 is equal to z (in red), over all mole fractions. (a) Case of a stable single-phase mixture at $z = 0.1300$, where the TPD is non-negative for all mole fractions and has a single global minimum. (b) Case of a mixture at liquid–liquid equilibrium at $z = 0.1384$, where the TPD is non-negative for all mole fractions and has two global minima, $z^{(1)} = 0.1384$ and $z^{(2)} = 0.9751$, that correspond to the mole fractions of component 1 in the two liquid phases. (c) Case of a metastable mixture at $z = 0.1500$, where the tangent plane intersects the Gibbs free energy curve at some mole fractions, so that the minimum TPD is negative. (d) Case of an unstable mixture at $z = 0.6500$, where the Gibbs free energy of mixing is positive. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2.1. Conditions for phase stability

According to the Gibbs phase rule, a multi-component mixture with N_c components under given conditions (T, P) of temperature and pressure can exhibit up to N_c+2 stable phases. Its phase stability is governed as a consequence of the second law of thermodynamics, which imposes that an isolated system at given $(T, P, \mathbf{n})$, where $\mathbf{n} := [n_1 \ \cdots \ n_{N_c}]^T$ are the mole numbers of each component, will approach equilibrium as its total Gibbs free energy G decreases, with the equilibrium state corresponding to the global minimum of G. It is possible to check that a phase, defined by its mole fraction vector $\mathbf{z}$, such that $z_i = \frac{n_i}{\sum_{c=1}^{N_c} n_c}$, $i = 1, \ldots, N_c$, is stable at T, P by applying the tangent plane distance criterion (Baker et al., 1982):

$$\forall \mathbf{x} \in X, \quad f_{\mathrm{TPD}}(\mathbf{x}) := \sum_{c=1}^{N_c} x_c \left[\mu_c(T, P, \mathbf{x}) - \mu_c(T, P, \mathbf{z})\right] \geq 0 \tag{1}$$

with $X := \{\mathbf{x} \in [0,1]^{N_c} \mid \sum_{c=1}^{N_c} x_c = 1\}$, and μ_c the chemical potential of component c.

The integration of the TPD criterion within a multiphase fluid equilibrium and stability algorithm was first proposed by Michelsen (1982). It was also proven by Michelsen (1982) and Baker et al. (1982) that the TPD criterion (1) is both necessary and sufficient for phase stability. Geometrically, $f_{\mathrm{TPD}}(\mathbf{x})$ represents the distance between the Gibbs free energy surface and the tangent plane to this surface at the global composition $\mathbf{z}$. This geometrical interpretation of the TPD criterion is illustrated in Fig. 1 for a binary mixture, where a phase at composition $\mathbf{z}$ is stable when the minimum TPD is equal to zero (Fig. 1a & b) and metastable or unstable when the minimum TPD is negative (Fig. 1c & d).

Ascertaining whether the TPD is non-negative for all mole fractions calls for the solution of a global optimisation problem (e.g., McDonald and Floudas, 1995; Nichita and Gomez, 2009), such as:

$$\begin{aligned} f^{\star}_{\mathrm{TPD}} \ := \ & \min_{\mathbf{x} \in [0,1]^{N_c}} \ f_{\mathrm{TPD}}(\mathbf{x}) \\ & \text{s.t.} \ \sum_{c=1}^{N_c} x_c = 1 \end{aligned} \tag{2}$$

Embedding the TPD criterion (Eq. (2)) as a constraint into an optimisation-based design problem thus results in a bi-level program, with problem (2) as the lower-level program and the additional constraint $f^{\star}_{\mathrm{TPD}} \geq 0$. Note that it is possible for the solution of the resulting problem to correspond to a case where the global phase is the only stable phase (Fig. 1(a)) or to a case where the global phase is in equilibrium with one or more incipient stable phases (Fig. 1(b)). In both cases, the global phase has a phase fraction of 1, satisfying the requirement that it is a homogeneous liquid. In practice, it would be desirable to maintain a safety margin away from the region of liquid–liquid demixing and this would be investigated during the verification phase which takes place after solution of the design problem. Regardless, the phase stability check is often simplified in CAM$^{\mathrm{b}}$D to avoid solving bi-level design problems, by focusing on the phase stability of one or more binary mixtures present in the system.

In the case of a binary liquid mixture comprising components a and b with respective mole fractions z_{a} and z_{b} at given (T, P), the change in the Gibbs free energy of the system on mixing a and b, ΔG_{mix}, is given by (Smith et al., 2001):

$$\Delta G_{\mathrm{mix}}(T, P, \mathbf{z}) = G^{\mathrm{E}}(T, P, \mathbf{z}) + RT(z_{\mathrm{a}} \ln z_{\mathrm{a}} + z_{\mathrm{b}} \ln z_{\mathrm{b}}) \tag{3}$$

where $\mathbf{z} := [z_a \; z_b]^T$, with $z_a + z_b = 1$, G^E is the excess Gibbs free energy measuring the deviation of the Gibbs free energy from ideal behaviour, and R is the universal gas constant. Necessary conditions for the binary mixture to be stable or metastable are given by (Smith et al., 2001):

$$\Delta G_{\text{mix}}(T, P, z_a) \leq 0 \tag{4}$$

$$\frac{\partial^2 \Delta G_{\text{mix}}}{\partial z_a^2}(T, P, z_a) \geq 0 \tag{5}$$

where z_b has been eliminated as an independent variable by setting $z_b = 1 - z_a$. Constraint (4) ensures that the mixture will attain a lower ΔG_{mix} than two pure phases (Smith et al., 2001), while constraint (5) is frequently enforced in a CAM^{b}D problem to increase the likelihood of the designed phase being stable (Buxton et al., 1999; Dahmen and Marquardt, 2017). Although it prevents the design of unstable mixtures, the latter constraint cannot distinguish between stable and metastable mixtures (Wasylkiewicz et al., 1996; Bausa and Marquardt, 2000; Bollas et al., 2009). This discrimination may only be achieved through integrating the rigorous TPD criterion (Eq. (1)) in the CAM^{b}D problem (Conte et al., 2011).

When designing multi-component mixtures with three or more components, one could decide to apply constraint (5) to all binary pairs (Jonuzaj et al., 2016). Importantly, however, these pairwise constraints would neither yield a necessary condition nor a sufficient condition in the sense that: (i) an unstable binary mixture may be stabilised via the addition of one or more components; or (ii) a mixture in which all binary pairs are fully miscible may present regions of liquid–liquid immiscibility. We will compare the use of the restricted stability criterion (5) with the use of a surrogate model of the rigorous stability criterion (1) for multi-component mixtures later in the paper.

2.2. Phase stability calculations using the UNIFAC model

The calculation of either the chemical potential or the Gibbs free energy of mixing, as used in the stability constraints (1), (4) and (5), requires the selection of a thermodynamic model. In the context of CAM^{b}D, it is desirable to use a predictive model in the form of a group-contribution method for relating molecular structure to thermodynamic properties. While the proposed methodology is in principle applicable to any parametric thermodynamic model, we adopt the UNIFAC activity coefficient model (Fredenslund et al., 1975) herein.

The UNIFAC model represents a component c in terms of a specific set $\mathbb{G}$ of atom groups, such as CH_3, CH or OH, and the number of occurrences $v_g^{(c)}$ of each group $g \in \mathbb{G}$. The UNIFAC model is independent of pressure, and the activity coefficients depend on the temperature T and molar composition $\mathbf{x}$ only. The activity coefficient of component c is determined as the sum of a combinatorial contribution, γ_c^C, and a residual contribution, γ_c^R:

$$\ln \gamma_c = \ln \gamma_c^C + \ln \gamma_c^R \tag{6}$$

$$\ln \gamma_c^C = 1 - J_c + \ln J_c - 5q_c \left(1 - \frac{J_c}{L_c} + \ln \frac{J_c}{L_c}\right) \tag{7}$$

$$\ln \gamma_c^R = q_c \left[1 - \sum_{g \in \mathbb{G}} \left(\kappa_g \frac{\xi_{c,g}}{w_g} - e_{c,g} \ln \frac{\xi_{c,g}}{w_g}\right)\right] \tag{8}$$

$$J_c = \frac{r_c}{\sum_{c'=1}^{N_c} r_{c'} x_{c'}} \tag{9}$$

$$L_c = \frac{q_c}{\sum_{c'=1}^{N_c} q_{c'} x_{c'}} \tag{10}$$

$$\kappa_g = \frac{\sum_{c'=1}^{N_c} x_{c'} q_{c'} e_{c',g}}{\sum_{c'=1}^{N_c} x_{c'} q_{c'}}, \quad g \in \mathbb{G} \tag{11}$$

$$\omega_g = \sum_{g' \in \mathbb{G}} \kappa_g \tau_{g,g'}, \quad g \in \mathbb{G} \tag{12}$$

$$\xi_{c,g} = \sum_{g' \in \mathbb{G}} e_{c,g'} \tau_{g,g'}, \quad g \in \mathbb{G} \tag{13}$$

$$e_{c,g} = \frac{v_g^{(c)} Q_g}{q_c}, \quad g \in \mathbb{G} \tag{14}$$

$$r_c = \sum_{g \in \mathbb{G}} v_g^{(c)} R_g \tag{15}$$

$$q_c = \sum_{g \in \mathbb{G}} v_g^{(c)} Q_g \tag{16}$$

$$\tau_{g,g'} = \exp\left(-\frac{a_{g,g'}}{T}\right), \quad g, g' \in \mathbb{G} \tag{17}$$

where Q_g and R_g are the van der Waals surface area and volume parameters for atom group $g \in \mathbb{G}$, respectively, and $a_{g,g'}$ is the interaction parameter between groups g and g'. The values of these parameters are taken from Poling et al. (2001) and are also available in the Dortmund Data Bank website (Dortmund Data Bank, 2024).

The focus throughout this paper is on mixtures with up to three phases present, one or two liquid phases plus one solid phase. The homogeneity of a given liquid phase (i.e., whether it splits into two liquid phases) may be determined based solely on the TPD criterion. Since the UNIFAC model (Eqs. (6)–(17)) predicts the activity coefficients of all components in a postulated phase, it is useful to express the stability criterion in terms of activity coefficients. The molar Gibbs free energy of a postulated phase is related to the chemical potentials of all components by:

$$G(T, P, \mathbf{x}) = \sum_{c=1}^{N_c} x_c \, \mu_c(T, P, \mathbf{x}) \tag{18}$$

where μ_c is the chemical potential of component c. The TPD function can be recast using activity coefficients (Atkins et al., 2023) as:

$$f_{\text{TPD}}(\mathbf{x}) = \sum_{c=1}^{N_c} x_c \left[\ln x_c + \ln \gamma_c(T, P, \mathbf{x}) - \ln z_c - \ln \gamma_c(T, P, \mathbf{z})\right] \tag{19}$$

Similarly, stability constraint (5) for a binary mixture can be expressed in terms of activity coefficients (Smith et al., 2001):

$$\frac{\partial \ln \gamma_a(T, P, z_a)}{\partial z_a} + \frac{1}{z_a} \geq 0. \tag{20}$$

This constraint is referred to as the restricted stability condition (RSC) in the remainder of this paper.

We use the open-source python library PhasePy (Chaparro and Mejía, 2020) to calculate the TPD function with UNIFAC. To increase the likelihood of locating a globally minimum $f_{\text{TPD}}^\star$ when solving the optimisation problem (2), PhasePy incorporates a multi-start heuristic based on initialising the gradient-based local minimisation solver at pure phase compositions as well as using multiple random initial starting points (Chaparro and Mejía, 2020). The TPD is minimised until three local minima – which may not be unique – have been identified. This multi-start heuristic is computationally inexpensive in the case of binary or ternary mixtures.

Stability is then inferred from the composition corresponding to the lowest tangent plane distance:

A stable phase: $f_{\text{TPD}}^\star = 0$

A metastable or unstable phase: $f_{\text{TPD}}^\star < 0$

We note that if there are two distinct minima with $f_{\text{TPD}}^\star = 0$, the phase is at the limit of stability and there exists an incipient liquid phase at a different composition. Nevertheless, we consider the phase as stable in such cases.

2.3. Testing the stability of a liquid phase in the presence of a solid

When a solid phase coexists alongside a fluid, an isothermal–isobaric flash (PT-flash) calculation needs to be conducted (Prausnitz

et al., 1980; Gupta et al., 1991; Michelsen and Mollerup, 2007; Pereira et al., 2010).

The library PhasePy includes an implementation of the modified multiphase flash method (Gupta et al., 1991) to compute phase stability and equilibrium in the presence of multiple fluid phases, using both activity coefficients models and cubic equations of state. This method calculates the equilibrium ratio, K_c^{π} as:

$$K_c^{\pi} := \frac{x_c^{\pi}}{x_c^{\circ}} = \frac{\hat{\phi}_c^{\circ}}{\hat{\phi}_c^{\pi}} \tag{21}$$

where x_c^{π} and $\hat{\phi}_c^{\pi}$ are the mole fraction and the specific fugacity of component c in phase π, respectively. The superscript $^{\circ}$ denotes a reference phase, which must correspond to a stable phase for the multiphase flash method to converge. The solution of the multiphase flash problem (Eqs. (25)–(29) in Gupta et al., 1991) determines the phase fractions, β_{π}, and the stability variables, θ_{π} for all phases $\pi = 1, \ldots, \Pi$: stable phases are those corresponding to positive phase fractions and stability variables equal to zero, $\beta_{\pi} > 0, \theta_{\pi} = 0$.

As part of this work, the PT-flash solver in PhasePy is extended to handle the presence of solid phases, alongside the fluid phases (slle solver). In the extension it is assumed the solid phase consists of a pure component c only, and its fugacity coefficient, $\phi_c^{(\mathrm{S})}$ is calculated as (Coutinho et al., 1995; Prausnitz et al., 1998):

$$\ln \phi_c^{(\mathrm{S})} = \ln \phi_c^{(\mathrm{L})}\Big|_{x_c=1} - \frac{\Delta H_c^{\mathrm{fus}}}{R}\left(\frac{1}{T} - \frac{1}{T_c^{\mathrm{m}}}\right) \tag{22}$$

where $\phi_c^{(\mathrm{L})}\Big|_{x_c=1}$ is the liquid fugacity coefficient of pure component c, $\Delta H_c^{\mathrm{fus}}$ the enthalpy of fusion of the solid form of c (with reference to a specific polymorph if relevant), and T_c^{m} the corresponding melting temperature. Note that the entropic contribution is neglected in Eq. (22), thereby assuming negligible heat capacity differences between the liquid and solid phases – although this contribution could easily be added if the required data were available (Prausnitz et al., 1998). The equilibrium ratio for a pure solid phase of component c with respect to the reference phase is given by:

$$K_c^{(\mathrm{S})} := \begin{cases} \dfrac{x_c^{(\mathrm{S})}}{x_c^{\circ}} = \dfrac{\hat{\phi}_c^{\circ}}{\phi_c^{(\mathrm{S})}}, & \text{if component } c \text{ can freeze} \\ 0, & \text{otherwise} \end{cases} \tag{23}$$

This extra equilibrium ratio is added to the formulation of the extended modified multiphase flash, while appending extra components β_{π} and θ_{π} to the phase fraction and stability vectors to represent the solid phase.

The solution of the multiphase flash in PhasePy is initialised will all phase fractions $\boldsymbol{\beta}$ set to $1/\Pi$, with Π the number of phases, and all stability variables $\boldsymbol{\theta}$ set to zero. The compositions of the liquid phases are obtained by minimising the TPD criterion (Eq. (19)), while the composition of the (pure) solid phase remains constant. The multiphase flash algorithm is primarily based on accelerated successive substitution (Fig. 2), until the equilibrium ratios are all within a specified tolerance (ε). The successive substitution approach has the advantage of being numerically stable and robust even when poor initial guesses are provided; but this may come at the cost of a slow convergence rate, especially near critical conditions (Michelsen and Mollerup, 2007). With the default setting in PhasePy up to five accelerated successive substitution cycles are performed, where successive substitution is accelerated after 20 substitution cycles for higher numerical stability. The maximum number of successive substitution iterations is set to $N_{ss}^{\max} = 100$. If the calculation has not converged after performing so many iterations, the slle solver switches to a direct minimisation of the Gibbs free energy of the system using SciPy's built-in L-BFGS-B method (Virtanen et al., 2020). The slle solver attempts to compute a solid–liquid–liquid equilibrium by default, but since it handles phase equilibrium and stability simultaneously, it can converge to liquid–liquid equilibrium, solid–liquid equilibrium, or a single liquid phase.

The presence of a stable liquid phase, without or with a separate solid phase, may be directly inferred from the multiphase flash results:

$$\begin{aligned} \text{A single, stable liquid phase } (\mathrm{L}_1): &\quad \theta_{\mathrm{L}_1} = 0,\ \theta_{\mathrm{L}_2} > 0,\ \theta_{\mathrm{S}} > 0 \\ \text{A stable liquid phase } (\mathrm{L}_1) & \\ \text{and a solid phase } (\mathrm{S}): &\quad \theta_{\mathrm{L}_1} = \theta_{\mathrm{S}} = 0,\ \theta_{\mathrm{L}_2} > 0 \\ \text{Two stable liquid phases } (\mathrm{L}_1, \mathrm{L}_2): &\quad \theta_{\mathrm{L}_1} = \theta_{\mathrm{L}_2} = 0,\ \theta_{\mathrm{S}} > 0 \\ \text{Two stable liquid phases } (\mathrm{L}_1, \mathrm{L}_2) & \\ \text{and a solid phase } (\mathrm{S}): &\quad \theta_{\mathrm{L}_1} = \theta_{\mathrm{L}_2} = \theta_{\mathrm{S}} = 0 \end{aligned}$$

3. Phase stability modelling and inference using a data classifier

This section details the design and training of a classifier to infer if a given mixture does not undergo liquid–liquid demixing. It then explains how to embed such a classifier within the optimisation formulation for CAM$^{\mathrm{b}}$D. The classifier is developed for ternary mixtures that contain two solvents and one (fixed) solute over a range of temperatures. The two solvents are chosen from a list of N_s candidates, so that the classifier should be applicable to all possible solvent combinations. The classifier addresses the following problem: "Given a mixture of 3 components, with two components chosen from a given list, at total composition, $\mathbf{z}$ and temperature T, determine whether the liquid phase is homogeneous or undergoes demixing".

3.1. Classifier input parameterisation

A natural choice for the classifier inputs consists of the identities of the components in the mixture, along with their respective mole fractions and the mixture temperature. However, using one-hot encoding (Brownlee, 2017; Kim and Boukouvala, 2020) to handle categorical inputs such as the component identities would result in a large number of binary input variables to the classifier, requiring as many inputs as there are candidate solvents in the set $\mathbb{S}$, for each component. Moreover, relaxing these binary variables as $[0, 1]$ continuous inputs (Holmström et al., 2008; Rashid et al., 2013; Müller et al., 2013; Müller, 2016) would require the same number of input parameters.

To limit the number of input variables in the classifier and draw on the benefits of a continuous surrogate, we exploit the fact that the set of components in the mixture to be designed is much smaller than the set of candidate solvents and we take advantage of the parametric nature of the UNIFAC model to identify the solvents in terms of continuous variables. Specifically, both UNIFAC parameters r_c (Eq. (15)) and q_c (Eq. (16)) are used to map the identities of the solvents, given that a (r_c, q_c) combination will usually be unique to a candidate solvent. For instance, the combination of volume and surface area parameters, $q_{\mathrm{H_2O}} = 1.4$ and $r_{\mathrm{H_2O}} = 0.92$, is unique to water. Note also that while r_c and q_c are treated as continuous variables during the training phase of a classifier, the trained classifier will only be evaluated at (r_c, q_c) combinations that match solvents in the set $\mathbb{S}$ during the CAM$^{\mathrm{b}}$D optimisation phase. This is because the evaluation of the UNIFAC model also requires the specification of the interaction energy parameters $\tau_{g,g'}$ (Eq. (17)) for all the atom group combinations present in the mixture, and these are not included as classifier inputs.

3.2. Classifier data generation

For a family of ternary mixtures comprising a given solute and two solvents a and b that can be chosen from a solvent list, with up to two liquid phases plus one solid phase present, the classifier inputs comprise $[q_{\mathrm{a}}\ q_{\mathrm{b}}\ r_{\mathrm{a}}\ r_{\mathrm{b}}\ z_{\mathrm{a}}\ z_{\mathrm{b}}\ T]^T$, where the temperature input T is optional. The classifier output is a scalar describing the likelihood of the corresponding mixture exhibiting a homogeneous liquid phase. The construction of a dataset for training, validation and testing of the classifier requires factoring for all possible permutations among the N_{s}

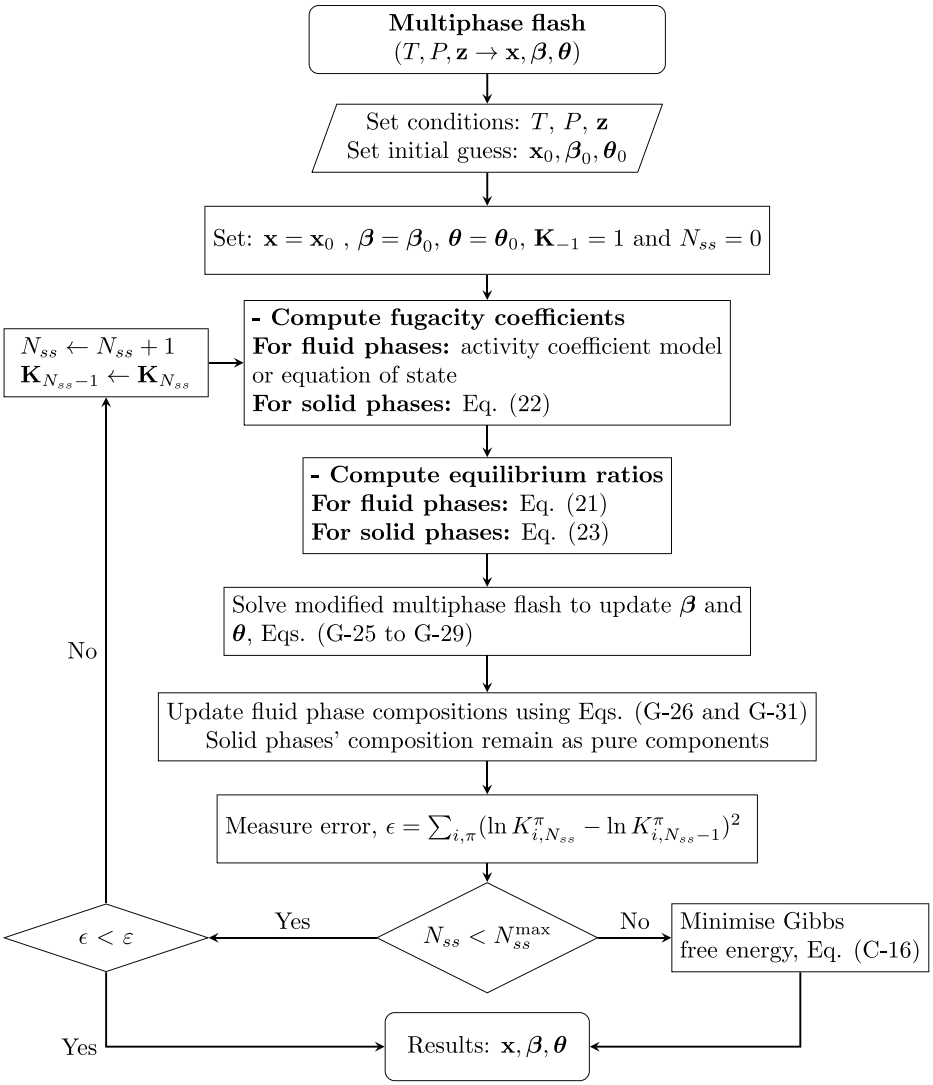


Fig. 2. Multiphase flash calculation procedure using accelerated successive substitution. Equations starting with G- and C- refer are in reference to the articles by Gupta et al. (1991) and Chaparro and Mejía (2020), respectively.

solvent candidates; that is a total of $\frac{1}{2}N_s(N_s - 1)$ solvent combinations.

For a given solvent pair (a, b), the corresponding UNIFAC parameters $(r_\mathsf{a}, q_\mathsf{a})$ and $(r_\mathsf{b}, q_\mathsf{b})$ are easily calculated using Eqs. (15) and (16). Then, we use space-filling sampling based on Sobol' sequences (Sobol', 1967) to generate the mole fraction $(z_\mathsf{a}, z_\mathsf{b})$ samples – including normalisation to ensure that $\sum_{i=1}^{N_c} z_i = 1$ – and, if required, the temperature T samples as well. We choose a large enough sample size to allow the constructed classifier to produce accurate phase stability predictions over the whole range of compositions and temperatures (c.f. Section 3.4). Furthermore, we consider the same set of generated compositions and temperatures for each solvent pair, which is computationally tractable for fewer than a dozen solvent candidates.

For each sample, we carry out a phase stability calculation to predict if the given solvent identities, mole fractions and temperature correspond to a state with a single homogeneous liquid phase ($y_{\text{misc}} = 1$) or if the liquid phase might undergo demixing ($y_{\text{misc}} = 0$). The phase stability calculation is done via PT-flash accounting for a putative solid phase and up to two liquid phases, as discussed in Section 2.3. To prepare the dataset, samples with $\theta_{\text{L}_1} = 0$ and $\theta_{\text{L}_2} > 0$ are labelled as $y_{\text{misc}} = 1$ and the others are labelled as $y_{\text{misc}} = 0$.

Lastly, the generated dataset is randomly shuffled and split into training, validation and testing subsets with a proportion of 0.70: 0.15: 0.15 respectively.

3.3. Classifier surrogate architecture

The proposed classifier surrogates are designed as feedforward ANNs, with one or more hidden layers and nonlinear activation functions, which we choose due to their properties as universal approximators (Hornik et al., 1989). The ANN used is expressed as follows:

$$\tilde{\alpha}_{0,n} = \frac{\alpha_n - \mu_{\alpha_n}}{\sigma_{\alpha_n}}, \quad n = 1 \dots N^0_{\text{neu}} \tag{24}$$

$$\tilde{\alpha}_{\ell,n} = \tanh\left({\mathbf{w}_{\ell-1,n}}^T \tilde{\boldsymbol{\alpha}}_{\ell-1} + b_{\ell-1,n}\right), \quad \ell = 1 \dots N_{\text{hid}}, \; n = 1 \dots N^{\ell}_{\text{neu}} \tag{25}$$

$$p_{\text{misc}} = \text{sigmoid}\left({\mathbf{w}_{N_{\text{hid}},1}}^T \tilde{\boldsymbol{\alpha}}_{N_{\text{hid}}} + b_{N_{\text{hid}},1}\right) \tag{26}$$

In this formulation, the classifier inputs, $\boldsymbol{\alpha} := [q_\mathsf{a}\ q_\mathsf{b}\ r_\mathsf{a}\ r_\mathsf{b}\ z_\mathsf{a}\ z_\mathsf{b}\ T]^T$ are normalised (Eq. (24)) to help speed up the training process (LeCun et al., 2012). The mean μ_{α_n} and standard deviation σ_{α_n} of each input element, $n = 1 \dots N^0_{\text{neu}}$, are computed over the training data set (70% of the complete data set). The normalised inputs $\tilde{\boldsymbol{\alpha}} \in \mathbb{R}^{N^0_{\text{neu}}}$ are then propagated (Eq. (25)) through all neurons $n = 1 \dots N^{\ell}_{\text{neu}}$ of all hidden layers $\ell = 1 \dots N_{\text{hid}}$, for given weight vectors $\mathbf{w}_{\ell,n} \in \mathbb{R}^{N^{\ell}_{\text{neu}}}$ and biases $b_{\ell,n}$. The hyperbolic tangent activation function is selected over other popular choices such as the rectifier linear unit (ReLU) which is nonsmooth and whose reformulation can introduce additional binary variables in the optimisation models (Grimstad and Andersson, 2019;

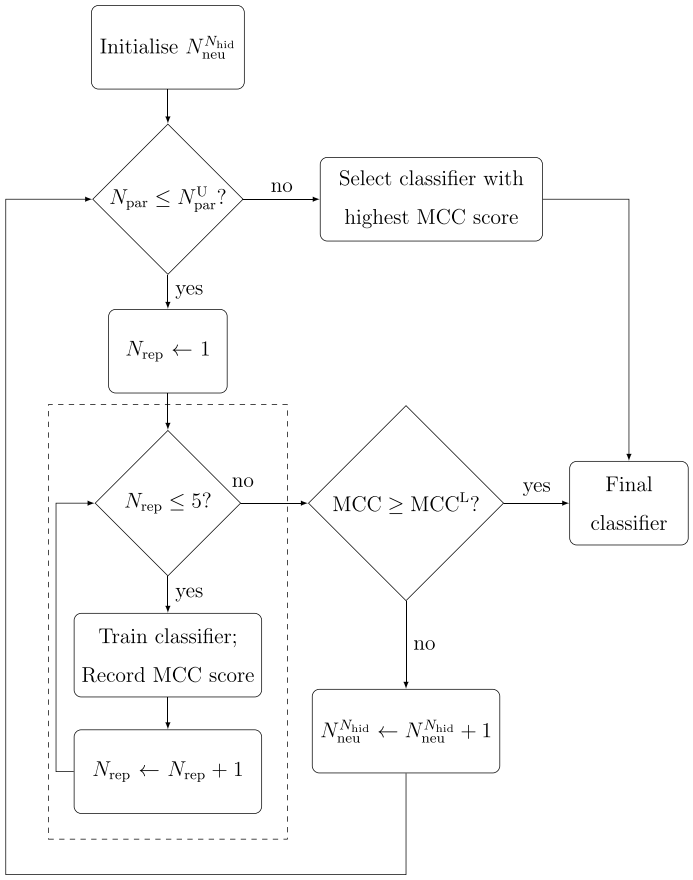


Fig. 3. Flowchart of the training procedure where the inner loop is shown in the dashed box.

Tsay et al., 2021). For a binary classifier, the output layer (Eq. (26)) consists of a single neuron, $N_{\text{neu}}^{N_{\text{hid}}+1} = 1$. The sigmoid activation function, $\text{sigmoid}(x) := \frac{1}{1+\exp(-x)}$ returns the scalar p_{misc}, interpreted herein as a probability (Niculescu-Mizil and Caruana, 2005; Guo et al., 2017); specifically, a probability for the mixture encoded by the inputs $\boldsymbol{\alpha}$ to exhibit a homogeneous liquid phase.

Finally, we apply thresholding (James et al., 2023) to convert the continuous value of $p_{\text{misc}} \in [0,1]$ into a binary outcome y_{misc} by choosing an appropriate lower bound $p_{\text{misc}}^{\text{L}} \in (0,1)$. Satisfaction of the surrogate constraint:

$$p_{\text{misc}} \geq p_{\text{misc}}^{\text{L}} \tag{27}$$

therefore describes a state with a homogeneous liquid phase ($y_{\text{misc}} = 1$).

3.4. Classifier training

Training of a classifier surrogate, (Eqs. (24)–(26)) for the sampled data $(\boldsymbol{\alpha}^{(s)}, y_{\text{misc}}^{(s)})_{1\leq s\leq N_{\text{train}}}$ entails a two-step process (Fig. 3):

Inner loop. For a given parameterisation $\{N_{\text{neu}}^{\ell}, \ell = 1 \dots N_{\text{hid}}\}$, a classifier surrogate comprises a total of $N_{\text{par}} := \sum_{\ell=0}^{N_{\text{hid}}} (N_{\text{neu}}^{\ell}+1) N_{\text{neu}}^{\ell+1}$ weight and bias parameters. We employ logistic regression (Hosmer et al., 2013) to fit these parameters, with binary cross-entropy (James et al., 2023) chosen as the loss function:

$$\min \sum_{s=1}^{N_{\text{train}}} -y_{\text{misc}}^{(s)} \ln\left(p_{\text{misc}}^{(s)}\right) - \left(1 - y_{\text{misc}}^{(s)}\right) \ln\left(1 - p_{\text{misc}}^{(s)}\right) \tag{28}$$

where $p_{\text{misc}}^{(s)}$ is the probability predicted by the classifier for the s^{th} input sample $\boldsymbol{\alpha}^{(s)}$, compared to the corresponding ground truth $y_{\text{misc}}^{(s)} \in \{0,1\}$. We solve the loss minimisation problem using the stochastic gradient descent method Adam (Kingma and Ba, 2014) and perform $N_{\text{rep}} = 5$ repeats from randomly initialised weights and biases to improve the likelihood of obtaining a (near-)optimal classifier.

Outer loop. The classifier design itself is optimised in an outer loop, with a view to minimising the number of neurons required to achieve a certain performance target. This iteration seeks to strike a balance between underfitting and overfitting, as, when embedding an ANN within an optimisation problem, ANNs with fewer neurons typically result in a problem that is faster to solve to global optimality (Schweidtmann and Mitsos, 2019). We quantify the performance of a trained classifier for a given parameterisation $\{N_{\text{neu}}^{\ell}, \ell = 1 \dots N_{\text{hid}}\}$ by means of a confusion matrix (James et al., 2023), as illustrated in Fig. 4. We specifically use the Matthews correlation coefficient (Chicco and Jurman, 2020), $\text{MCC} \in [-1,1]$, with 1 and -1 indicating perfect classification and perfect misclassification, respectively[1]:

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \tag{29}$$

Unlike other metrics, the MCC score accounts for all four elements of a confusion matrix and is insensitive to the number of positive *vs.* negative samples not being in equal proportions.

The numbers of neurons N_{neu}^{ℓ} in each hidden layer, $\ell = 1 \dots N_{\text{hid}}$ are incremented after each iteration. This outer loop is interrupted as soon as a minimal threshold MCC^{L} on the MCC score is met, or upon exceeding the maximum number of neurons, $N_{\text{par}}^{\text{U}}$, as defined by the user and set to 600 for this study.

[1] The MCC is well-defined when the numbers of true positives and true negatives are both non-zero.

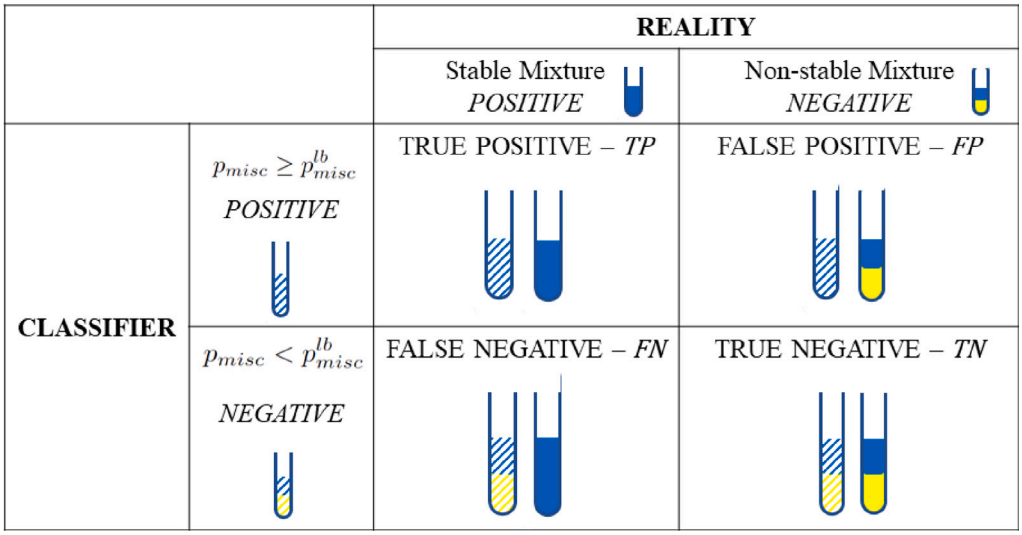


Fig. 4. Confusion matrix describing the four elements of the matrix – numbers of true positive (TP), false positive (FP), true negative (TN) and false negative (FN) – in the context of stability of mixtures where reality is visualised with a test tube and solid fill and the classifier prediction by a test tube with hashed fill. A single colour in the test tube represents that the liquid phase is homogeneous, while two colours denote that it undergoes liquid–liquid demixing.

3.5. Implementation details

The classifier training procedure (Fig. 3), and subsequent embedding of a trained classifier into a solvent mixture optimisation model, are coordinated from Python notebooks, which are provided as part of the Electronic Supplementary Information (ESI).

The composition and temperature inputs to train the classifier surrogates are sampled using the Sobol' sequence generator in SciPy (Virtanen et al., 2020), before passing to PhasePy (Chaparro and Mejía, 2020) as modified in this work for performing the phase stability calculations. The training of the ANN classifiers is conducted with Tensorflow (v2.12.10) (Abadi et al., 2015). The outer training loop incrementing the number of neurons in each layer is interrupted as soon as an MCC score greater than 0.99 is achieved on the validation data set. Both the data generation and model training are carried out on a laptop with an Intel® Core i7-1165G7 CPU @1.69 GHz with 8 GB of RAM running Windows 11.

The integration of the classifier surrogates into a solvent mixture design problem is automated using the GAMS Python API to pass the weight and bias parameters. The resulting optimisation models are solved in the GAMS modelling environment (v42) using the global mixed-integer nonlinear programming (MINLP) solver SCIP (v8.0) (Bestuzheva et al., 2023) or the local MINLP solver SBB (v42.3) (GAMS Development Corporation, 2023). Since SCIP does not include an implementation of the tanh function the recommended reformulation, $\tanh(x) = 1 - \frac{2}{1+\exp(2x)}$, is implemented instead (Schweidtmann and Mitsos, 2019). A ranked list of the top five best performing mixtures is generated by iteratively adding integer cuts to eliminate previous solutions, as detailed in Appendix D. All the optimisation models are solved using a dual 12-core Intel® Xeon® CPU E5-2650 v4 @2.20 GHz with 196 GB of RAM running Centos 7.8. Multi-threading is enabled and all 24 threads are utilised.

4. Results and discussion

The solvent design problem is solved to generate a list of candidate solvents and associated operating conditions that are likely to offer good performance in the process. This approach can be used to guide the experimental exploration of the space of possible solvents, ultimately accelerating process development by reducing the time and amount of material required. To assess the effectiveness of surrogate classifiers at enforcing phase stability in optimisation-based design of solvent mixtures, we choose to focus on the design of binary solvent mixtures in the context of crystallisation and select two case studies from the literature. The presence of a single liquid phase within any crystallisation unit is indeed crucial during the manufacturing of active pharmaceutical ingredients (APIs) such as ibuprofen, where liquid–liquid demixing is known to be detrimental to product quality (Yu et al., 2021; Dohrn et al., 2021).

Table 1
Wall time (in seconds) taken to carry out 18,432 phase stability calculations and to train the CBS model for Case Study 1. The final classifier corresponds to the chosen single-layer architecture with 8 neurons.

Model	Data generation (s)	Classifier training (s)	Final architecture
CBS	1622	3193	8

4.1. Case study 1

The objective of the first case study is to design a binary solvent mixture that maximises the solubility of ibuprofen at a set temperature of 300 K. The case is based on the model developed by Jonuzaj et al. (2016). It entails the selection of two solvents to ensure a single, homogeneous liquid phase in equilibrium with ibuprofen. A set of nine solvents is considered, thereby leading to 36 binary mixture candidates. In addition to determining the identities of the two solvents in the mixture, their respective mole fractions are also optimised.

Mathematical programming formulations for this CAM^{b}D problem are detailed in Appendix A, using either the restricted stability condition (Eq. (20)) or the classifier-based surrogate to enforce phase stability. In the latter case, an ANN classifier with a single hidden layer, i.e., $N_{\text{hid}} = 1$, is preferred over a deeper ANN architecture to reduce the optimisation times on solving the model (Schweidtmann and Mitsos, 2019).

4.1.1. Classifier training

For each of the 36 candidate mixture combinations, $2^9 = 512$ Sobol' samples are generated, resulting in a total of 18,432 samples. Phase stability calculations are conducted for every sample, to generate the required dataset to train the classifier-based surrogate (CBS) model. The wall times taken to complete the phase stability calculations for all the samples and to train the corresponding classifier are reported in Table 1. Overall, both the data generation and training steps are

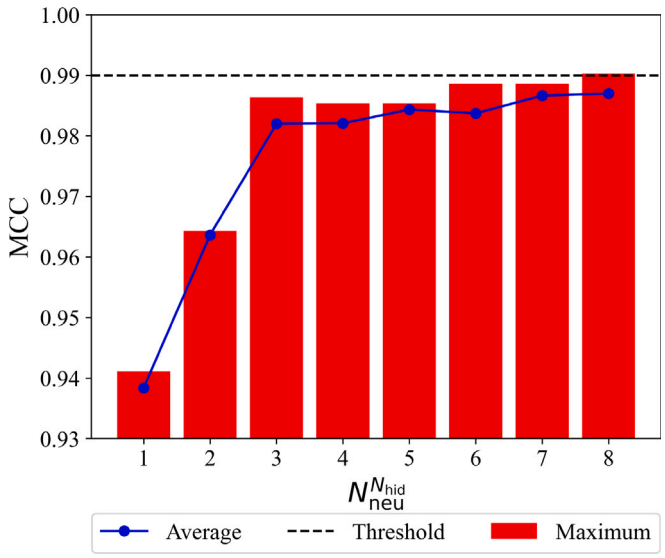


Fig. 5. MCC scores against number of neurons in the hidden layer $N_{neu}^{N_{hid}}$ for Case Study 1. All MCC scores are calculated on the validation set. Blue points/lines: average MCC score over the 5 runs of logistic regression. Red bars: maximum MCC score attained over the 5 runs. Black dashed lines: MCC threshold score. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Confusion matrices calculated on the training, validation and testing datasets for the CBS model in Case Study 1. The columns labelled '% of reality' report the proportion of classifier predictions that are consistent with the rigorous phase stability calculations.

		Training dataset		Validation dataset		Testing dataset	
		Number	% of reality	Number	% of reality	Number	% of reality
CBS	TP	11,144	99.8	2408	99.8	2380	99.7
	FP	12	0.69	2	0.57	3	0.79
	FN	19	0.17	4	0.17	7	0.29
	TN	1723	99.3	350	99.4	375	99.2

completed within an acceptable time frame compared to solving the CAM^{b}D optimisation models themselves (see results and discussion below).

Fig. 5 illustrates the average and maximum MCC scores in terms of the number of neurons in the hidden layer $N_{neu}^{N_{hid}}$, as incremented in the outer loop of the training procedure (Fig. 3). The CBS meets the specified threshold ($MCC^L \geq 0.99$) with eight neurons. The confusion matrices calculated on the training, validation and testing datasets for the CBS model are reported in Table 2, confirming that the proportion of misclassified samples (either false negative or false positive) is below 2%.

4.1.2. Solution of CAM^b D problem

The top five performing mixtures computed by the global solver GAMS-SCIP for the CAM^{b}D optimisation models with CBS are compared to those computed with the restricted stability condition (RSC) in Table 3. A *post hoc* phase stability check is further carried out using PhasePy at the optimal mixture composition of each solution to confirm its validity. The wall time taken to solve each optimisation problem, up to a time limit of 10,800 s, is shown in Fig. 7.

All optimal mixtures in Table 3 are found to exhibit a homogeneous liquid phase at the reported compositions. These compositions lie at the solid–liquid equilibrium (SLE) boundary since the ibuprofen solubility conditions ((A.5) & (A.16)) are enforced in the optimisation models. The chloroform + water mixture, which is identified as the top performing design (IC1) both for RSC and CBS, is particularly

Table 3
Ranked lists of top five performing mixtures for the CAM^{b}D problem in Case Study 1, embedding either the restricted stability condition (RSC) or the classifier-based surrogates (CBS). The solvent identities (a, b), their respective mole fractions (z_a, z_b, the mole fraction of ibuprofen (z_{ibu}, objective function), and the stability probability predicted by the CBS (p_{misc}) are reported. The rightmost column shows the outcome of the *post hoc* phase stability check, with a tick where the liquid phase is homogeneous and a cross where two liquid phases are present at equilibrium.

Model	IC	a	b	z_a (–)	z_b (–)	z_{ibu} (–)	p_{misc}(%)	Check
RSC	1	chloroform	water	0.610	0.052	0.338	–	✓
	2	methanol	chloroform	0.143	0.523	0.334	–	✓
	3	ethanol	chloroform	0.108	0.567	0.325	–	✓
	4	2-propanol	chloroform	0.091	0.586	0.323	–	✓
	5	chloroform	toluene	0.674	0.010	0.316	–	✓
CBS	1	chloroform	water	0.614	0.050	0.336	50.00	✓
	2	methanol	chloroform	0.143	0.523	0.334	100.00	✓
	3	ethanol	chloroform	0.108	0.567	0.325	100.00	✓
	4	2-propanol	chloroform	0.091	0.586	0.323	100.00	✓
	5	chloroform	toluene	0.674	0.010	0.316	100.00	✓

interesting (Fig. 6). At the mixture composition obtained with RSC, the UNIFAC model predicts a positive Gibbs free energy of mixing when simply considering the binary chloroform + water mixture, indicative of an immiscible mixture (see stability condition, Eq. (4)). The formulation proposed by Jonuzaj et al. (2016) with the RSC only considers the binary solvent system and does not account for the impact of the presence of the solid on the overall phase stability of the system. In reality, the addition of ibuprofen in the mixture has a stabilising effect so that, over a small range of compositions, it is possible to observe a single homogeneous liquid phase. This behaviour is illustrated in Fig. 6(a), where the grey lines denote the boundaries between different regions of the ternary phase diagram of chloroform + water + ibuprofen. On using CBS, the small region of miscibility where a high solubility of ibuprofen is achieved can be accurately identified. These miscibility predictions are illustrated with the probability map overlays on the ternary phase diagrams in Fig. 6(b). The reader is referred to Appendix C for further discussions about how ibuprofen addition may impact the phase stability of a binary solvent mixture.

The CBS model is trained on ternary mixture data accounting for the presence of ibuprofen in both solid and liquid phases. The phase boundaries separating regions where one liquid phase exists from those where two liquid phases are present on Fig. 6(b) are again captured accurately by CBS and regions of high probability are found to be consistent with single liquid phase regions. With the CBS model, the maximum solubility of ibuprofen is found near the liquid–liquid phase boundary for the ternary mixture. The solution obtained with RSC is also close to the phase boundary, although this is merely coincidental due to the fact that RSC does not include a constraint on the sign of the Gibbs free energy of mixing. We shall see in Case Study 2 that the use of RSC can lead to non-stable phases.

The optimal solubility values z_{ibu} obtained with the CBS model is slightly lower than that obtained with RSC for the chloroform + water solvent mixture (Table 3). This is attributed to the default threshold value p_{misc} of 50% being slightly conservative, preventing the optimiser from getting closer to the true liquid–liquid phase boundary, where the probability constraint, Eq. (27), becomes violated. Overall, the use of CBS can successfully prevent the occurrence of solutions where the SLE configuration is not stable. This demonstrates the potential of embedding classifier surrogates in CAM^{b}D problems as a means to identify solvent mixtures that have a high probability of being physically realisable.

From a design perspective, the chloroform + water mixture offers a slightly higher solubility of ibuprofen compared to the methanol + chloroform mixture. However, based on the prediction from the CBS model, there is a 50% probability that the chloroform + water mixture exhibits a homogeneous liquid phase at the composition reported in Table 3. In contrast, the CBS model predicts that the methanol +

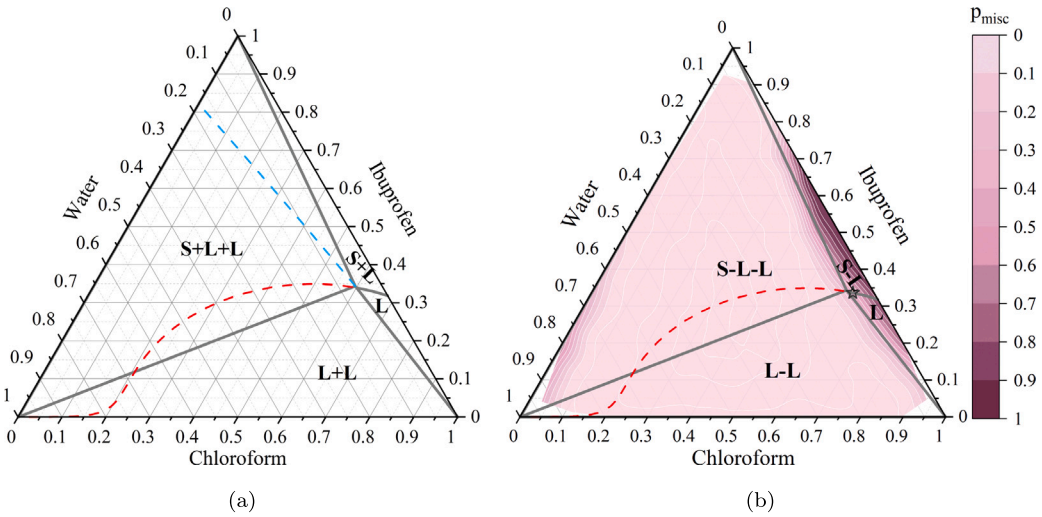


Fig. 6. Ternary phase diagram for (chloroform + water +ibuprofen) mixture at 300 K. (a) Phase boundaries (solid grey lines) between stable phases, along with metastable portion of the SLE boundary (dashed red line) and metastable portion of the LLE boundary (dashed blue line). The probability map and the optimal solution by the star symbol is shown for (b) CBS.

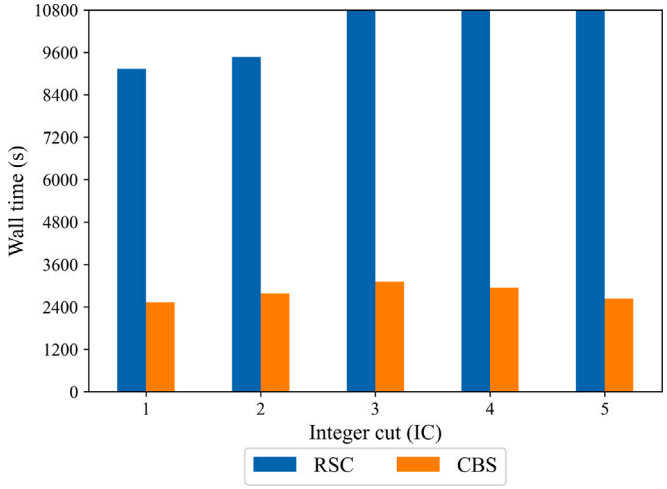


Fig. 7. Wall time comparison using GAMS-SCIP to solve the CAM^{b}D problems of Case Study 1, embedding either the restricted stability condition (RSC) or the classifier-based surrogate (CBS).

chloroform is practically guaranteed to exhibit a homogeneous liquid phase at the reported composition. If the consideration of homogeneity is of greater importance, the designer may focus the experimental verification efforts on the methanol + chloroform mixture. By using RSC, one cannot derive this insight while comparing the potential of in-silico design candidates.

Lastly, the computational effort required to solve the CAM^{b}D optimisation models is summarised in Fig. 7. The solution of the model with RSC embedded takes about 90% of the time limit to converge for the first two runs (IC1–IC2), and then fails to converge within the time limit for the final three runs (IC3–IC5), whereas for CBS all runs (IC1–IC5) complete within 24%–28% of the time limit. This clearly highlights the potential reduction in computational effort needed to certify global optimality with the CBS. Moreover, the models with CBS embedded provide a higher degree of reliability when factoring in the phase behaviour of the overall ternary system, which the RSC constraint does not consider. Even taking into account the time taken to generate the data and train the CBS models, the proposed formulations are found to offer better performance; even more so when generating a ranked list of mixtures as the data generation and classifier training only need to be performed once.

4.2. Case study 2

The objective of this case study is to design a binary solvent mixture that maximises the yield of ibuprofen from a crystallisation process that may include cooling (a change in temperature), the use of anti-solvent (a change in solvent mass and composition), or a hybrid of the two. The case is based on the model developed by Watson et al. (2021), where the approach to design focuses on the thermodynamic, rather than kinetic, aspects of crystallisation, and hence, only information about the initial and final states of the system is considered. The solvents (and anti-solvents) are selected from eight candidate molecules, leading to 56 potential binary solvent pairs to choose from as part of the optimisation. For simplicity, the UNIFAC model is used to estimate the activity coefficients rather than the SAFT-γ Mie group-contribution approach (Papaioannou et al., 2014), as used by Watson et al. (2021). The decision variables for this optimisation are the identities of solvents, their compositions at the initial and final stages, and the temperatures at the initial and final stages. Alongside maximising the crystal yield, lower limits are imposed on the solvent consumption to prevent the formation of a highly viscous slurry (Jonuzaj et al., 2020). Restrictions are also imposed to prevent the process from operating too close to the melting or boiling point of a solvent.

Mathematical programming formulations for this CAM^{b}D problem are detailed in Appendix B, with either the restricted stability condition (Eq. (20)) or the classifier-based surrogate to enforce phase stability. Due to the added dependence of the data on temperature, an ANN classifier with a deeper architecture is required: $N_{hid} = 3$ is considered here, consisting of $N_{neu}^{N_{hid}}$+6, $N_{neu}^{N_{hid}}$+3 and $N_{neu}^{N_{hid}}$ neurons in the first, second and third layer, respectively. When allowing process temperatures to be optimised alongside the solvent mixture selection, the resulting optimisation models are, therefore, significantly more challenging to solve than those in Case Study 1.

4.2.1. Classifier training

In this case study, for each of the 56 candidates, $2^{12} = 4096$ Sobol' samples are generated, leading to a total of 229,376 samples, and phase stability calculations are carried out at each sample. The greater number of samples compared to Case Study 1 is to account for the additional complexity caused by the added temperature dependence in the phase stability predictions. Some of the candidate solvent mixtures undergo liquid–liquid separation upon addition of ibuprofen in certain

Table 4
Wall time (in seconds) taken to carry out 229,376 phase stability calculations and to train CBS for Case Study 2. The final classifier corresponds to the chosen three-layer architecture with 15, 12 and 9 neurons in the first, second and third hidden layers, respectively.

Model	Data generation (s)	Classifier training (s)	Final architecture
CBS	5326	4369	{15,12,9}

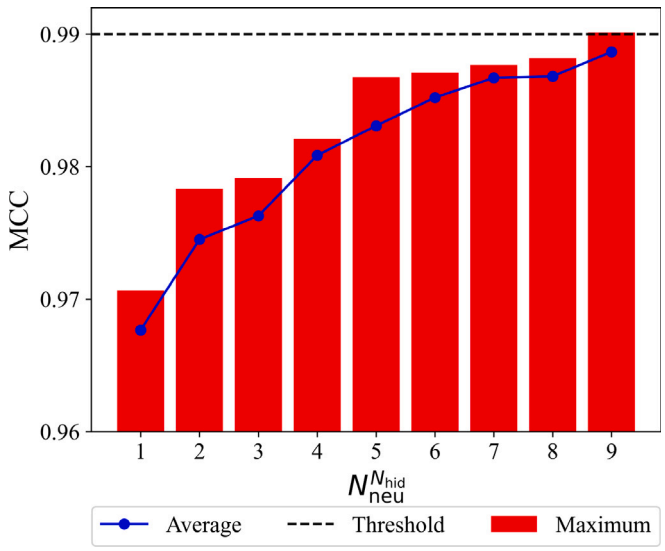


Fig. 8. MCC scores against number of neurons $N_{neu}^{N_{hid}}$ incremented in the outer loop of the training procedure for CBS in Case Study 2. All MCC scores are calculated on the validation set. Blue points/lines: average MCC score over the 5 runs of logistic regression. Red bars: maximum MCC score attained over the 5 runs. Black dashed lines: MCC threshold score.

Table 5
Confusion matrices calculated on the training, validation and testing datasets for CBS model in Case Study 2. The columns labelled '% of reality' report the proportion of classifier predictions that are consistent with the rigorous phase stability calculations.

		Training dataset		Validation dataset		Testing dataset	
		Number	% of reality	Number	% of reality	Number	% of reality
CBS	TP	132,768	99.8	28,429	99.83	28,460	99.8
	FP	236	0.86	49	0.83	63	1.08
	FN	200	0.15	47	0.17	54	0.19
	TN	27,078	99.1	5821	99.1	5770	98.9

temperature ranges. For instance, a mixture of water + ethanol + ibuprofen has been reported to exhibit a lower solution critical temperature, with no liquid–liquid separation at temperatures below 298.15 K and liquid–liquid separation at 313.15 K (Manrique and Martinez, 2007; Rashid et al., 2014). This behaviour is also predicted by the UNIFAC model although the extent of demixing is overestimated relative to the experimental data.

The wall times taken to generate the samples and train the classifier surrogates are reported in Table 4, with the progress of the outer loop as a function of the number of neurons $N_{neu}^{N_{hid}}$ (Fig. 3) illustrated in Fig. 8. Convergence is achieved for $N_{neu}^{N_{hid}\star} = 9$. The performance of the trained classifiers, assessed through the corresponding confusion matrices (Table 5), confirms that the proportion of incorrect classifications is around or below 1% in all cases.

4.2.2. Solution of CAM^{b}D problem

None of the CAM^{b}D problems can be solved to global optimality using off-the-shelf solvers such as GAMS-BARON (v 23.1.5) (Nick Sahinidis, 2023) and GAMS-SCIP (v8.0) within 10 h of computational wall time. This prompts the use of the local MINLP solver GAMS-SBB (v 42.3) instead. The results obtained with GAMS-SBB, when embedding either RSC or CBS to ensure a homogeneous liquid phase, are reported in Table 6. Integer cuts (Appendix D) are applied in each case to generate a list of (local) solutions, presented here in the order they are generated rather than in order of decreasing yield. All of these solutions are verified by carrying out rigorous *post hoc* phase stability checks with PhasePy on the optimised mixtures both in their initial and final states. Among the solutions reported, only solution RSC/IC1 corresponds to anti-solvent crystallisation without cooling (identical initial and final temperatures). Solutions RSC/IC4, RSC/IC5, CBS50/IC2 and CBS85/IC2 and CBS85/IC4 correspond to cooling crystallisation (unchanged z_a: z_b ratio between the initial and final stages). The rest of the solutions use a combined cooling and anti-solvent crystallisation (with solvent b added as the anti-solvent, as specified in the model equations).

Recall that for the solutions obtained with RSC embedded, phase stability is enforced in an approximate manner in part because the presence of ibuprofen is neglected. The impact of this approximation is clearly visible in the results as all of the pairs but ethanol + water are found to undergo demixing at the initial and/or final state after the *post hoc* stability check (columns labelled 'Check' in Table 6). It is critical, therefore, to describe the phase behaviour of the ternary system more accurately to ensure feasible mixtures and process conditions are identified.

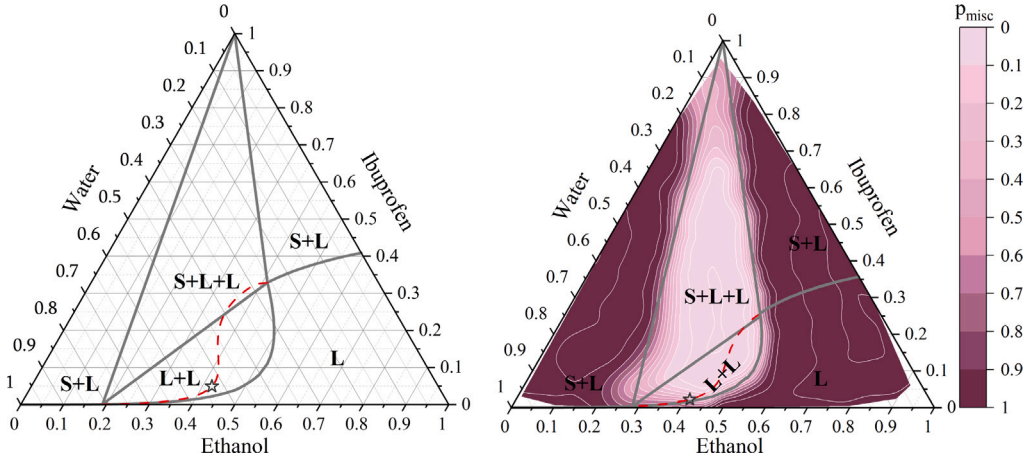
A larger set of feasible designs, i.e., that exhibit a homogeneous liquid phase at the initial and final states, can be obtained with the classifier-based surrogate (CBS) model. On solving the model with CBS embedded when p^L_{misc} is set as 0.50, denoted as CBS50, three out of the five designs pass the *post hoc* stability check, a major improvement compared to RSC. The three stable solutions generated with CBS50 include the high-yield ethanol + water pair, as well as the 1-propanol + n-heptane and acetone + n-heptane pairs with lower yields around 50%–55%. It is also worth noting that only those solutions with a stability probability $p_{misc} > 83\%$ correspond to stable homogeneous mixtures. This indicates a high sensitivity of the accuracy of the classifier to the proximity of the mixture to the phase boundary.

Since it is desirable to operate a safe distance away from regions of liquid–liquid demixing in practice, the use of a higher probability threshold is advisable here. Therefore, we investigate the effect of selecting a higher threshold of $p^L_{misc} = 0.85$, denoted as CBS85, chosen to be greater than the predicted probability of 0.8293 obtained for the 1-propanol + water pair (CBS50/IC3), which was found to fail the post hoc test at the initial state. The list of solvent mixtures determined with CBS for the new threshold, shown in Table 6, are all confirmed to be stable. In addition to the ethanol + water, acetone + n-heptane and 1-propanol + n-heptane pairs that had already been identified, the new pairs consist of 1-pentanol + n-heptane and 1-butanol + n-heptane. Notice that the water + ethanol pair is no longer on the list, which suggests that the local solver is unable to find a feasible operating point. However, separate optimisation runs lead to the successful identification of a stable solution with a yield of around 64% for the water + ethanol pair.

To gain a better understanding of the importance of the training data on the reliability of the classifier surrogates, we investigate the mixture of water and ethanol in more detail. A process that makes use of this mixture and combines cooling and anti-solvent crystallisation, with water as the anti-solvent, is found to achieve the greatest yield and is identified with the various optimisation formulations – it is reported as ethanol + water pair in Table 6 and is found to be homogeneous during the *post hoc* check. On the other hand, cooling crystallisation processes that use the same mixture as a solvent (RSC/IC4 and CBS50/IC2), with a water:ethanol molar ratio of about 1.25 for RSC/IC4 and 1.4 for CBS50/IC2, are identified but are found to fail the *post hoc* check – these are reported as water + ethanol pairs in Table 6. Ternary phase diagrams for the water + ethanol + ibuprofen mixture are shown in Fig. 9(a) at 318.15 K (initial state for RSC/IC4), and in Fig. 9(b) at 313.96 K (initial state for CBS50/IC2). The

Table 6
Top five performing mixtures for the CAM^{b}D problem in Case Study 2, embedding either the restricted stability condition (RSC) or the classifier-based surrogate (CBS) model. The CBS model is solved with p^{L}_{misc} values set at 0.50 and 0.85, denoted as CBS50 and CBS85, respectively. The * superscript in column IC highlights the best stable (local) solution identified using GAMS-SBB. Reported for each optimal solution are the solvent identities (a, b), their respective mole fractions (z_a, z_b) at the initial (ini) and final (fin) states, the temperature of the initial state (T^{ini}), the solvent use both on a mass and a volume basis, the percentage yield of ibuprofen (Y_{ibu}, objective function). The percentage probability of stability (p_{misc}) of the solvent pairs predicted by the CBS at the ini and fin states are also reported. The two rightmost columns show the outcome of the *post hoc* phase stability check at the ini and fin states, with a tick where the liquid phase is homogeneous and a cross where two liquid phases are present at equilibrium. For all solutions except for CBS50/IC3, $T^{fin} = 293.15$ K. For CBS50/IC3, $T^{fin} = 295.82$ K.

Model	IC	a	b	z_a (–)		z_b (–)		T^{ini}	Solvent use		Y_{ibu}	p_{misc}(%)		Check	
				ini	fin	ini	fin	(K)	(g/g)	(mL/g)	(%)	ini	fin	ini	fin
	1	acetone	water	0.53	0.10	0.32	0.90	293.15	3.73	4.00	99.88	N/A	N/A	✗	✓
	2*	ethanol	water	0.68	0.10	0.14	0.90	302.70	3.77	4.00	99.86	N/A	N/A	✓	✓
RSC	3	1-propanol	water	0.34	0.10	0.56	0.90	318.15	3.75	4.00	99.72	N/A	N/A	✗	✗
	4	water	ethanol	0.53	0.55	0.42	0.44	318.15	3.52	4.15	81.13	N/A	N/A	✗	✓
	5	water	acetone	0.60	0.62	0.35	0.37	318.15	3.50	4.11	87.27	N/A	N/A	✗	✗
	1*	ethanol	water	0.52	0.10	0.34	0.90	308.50	3.77	4.00	99.86	98.73	100.00	✓	✓
	2	water	ethanol	0.57	0.58	0.41	0.42	313.96	9.59	11.26	66.31	50.00	99.99	✗	✓
CBS50	3	1-propanol	water	0.48	0.48	0.42	0.47	303.04	3.59	4.26	51.92	82.93	100.00	✗	✓
	4	1-propanol	n-heptane	0.17	0.10	0.44	0.78	315.92	3.50	5.06	50.00	100.00	100.00	✓	✓
	5	acetone	n-heptane	0.10	0.09	0.66	0.81	309.17	3.50	5.08	54.93	100.00	99.99	✓	✓
	1*	ethanol	water	0.52	0.10	0.34	0.90	308.50	3.77	4.00	99.86	98.73	100.00	✓	✓
	2	n-heptane	acetone	0.72	0.81	0.08	0.09	306.74	3.50	5.08	54.93	100.00	100.00	✓	✓
CBS85	3	1-pentanol	n-heptane	0.11	0.09	0.59	0.81	312.30	3.50	5.05	53.78	100.00	100.00	✓	✓
	4	1-butanol	n-heptane	0.08	0.09	0.72	0.81	306.86	3.50	5.06	53.07	100.00	100.00	✓	✓
	5	1-propanol	n-heptane	0.09	0.09	0.69	0.81	307.63	3.50	5.07	52.33	100.00	100.00	✓	✓



(a) RSC at 318.15 K (b) CBS50 at 313.96 K

Fig. 9. Ternary phase diagrams for (water, ethanol, ibuprofen) mixture at various temperatures. The subplots show the phase boundaries (solid grey lines) between stable phases, along with metastable portion of the SLE boundary (dashed red line) and metastable portion of the LLE boundary (dashed blue line). The stability probability (colour map) on the subplots on the right is for CBS. The reported optimal solutions (star symbol) correspond to: (a) RSC at 318.15 K; (b) CBS50 at 313.96 K.

optimal compositions identified in each case are indicated by a star on the diagrams. At both temperatures, the phase boundaries (solid grey curves) indicate that the mixture exhibits a large region in which the liquid phase undergoes demixing, with or without a solid phase, despite the fact that the binary water + ethanol mixture is miscible under all conditions. The region of demixing becomes larger as the temperature increases. The probability map overlay on the phase diagram (Fig. 9(b)) shows that CBS describes the phase diagram well, yet with greatest uncertainty at the boundary between two-liquid-phase regions (L+L, S+L+L) and one-liquid-phase regions (L, S+L).

As expected, the solutions always lie on the solid–liquid equilibrium curve (S+L). Focusing on the water + ethanol pair (cooling crystallisation), the RSC/IC4 solution lies within the large liquid–liquid phase region which is observed at 318.15 K. For the CBS50/IC2 solution, the demixing region is smaller at 313.96 K; and while the solution fails the *post hoc* stability check, it is found to lie very close to the phase boundary, suggesting that this may be due to classification inaccuracies near the phase boundary and to the low probability threshold used. Unfortunately, at the temperatures where the presence of ibuprofen induces liquid–liquid separation, much of the SLE curve becomes metastable and many ethanol:water molar ratios become impractical. Only solvent mixtures with 60 mol% ethanol or more do not undergo demixing, but the high ibuprofen solubility at such high concentrations of ethanol does not allow a reasonable yield to be achieved, thereby making cooling crystallisation a poor choice.

Finally, we examine the computational effort needed to solve the different optimisation model formulations for Case Study 2. All but one problem is solved within 15 s of wall time using GAMS-SBB (Fig. 10). The CBS50 case takes the longest to solve for IC4, possibly due to the initial guess provided to the local solver from the solution of the previous integer cut iteration. By contrast, the RSC cases are solved the quickest. However, since the RSC constraints cannot ensure that the optimised designs can be realised physically, the higher computational burden associated with CBS seems a reasonable price to pay, even after accounting for the time taken for sample generation and model training.

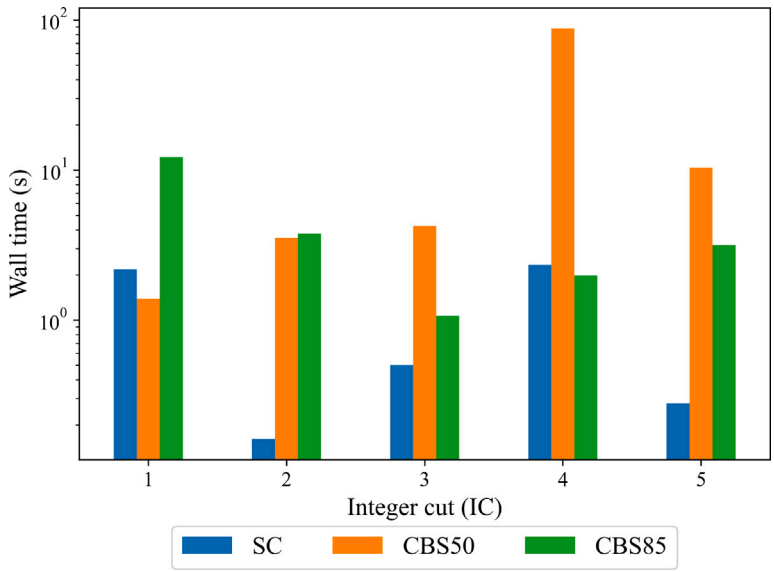


Fig. 10. Wall time (on a $\log_{10}$ scale) comparison using GAMS-SBB to solve the CAM^{b}D problems of Case Study 2, embedding either the restricted stability condition (RSC) or the classifier-based surrogate (CBS). CBS50 and CBS85 denote where p^L_{misc} is set as 0.50 and 0.85 respectively.

5. Conclusions

Through this paper, we proposed the use of classifier surrogates to ensure the phase stability or miscibility of mixtures obtained by computer-aided mixture/blend design (CAM^{b}D). While machine learning models were previously used to represent phase behaviour, these studies were restricted to specific mixtures with fixed components and not incorporated within optimisation models. Herein, each classifier surrogate is applicable to a set of mixtures (36 mixtures for Case study 1 and 56 mixtures for Case study 2), over all compositions and, where relevant, over a range of temperatures too. Embedding such classifier surrogates within an overarching optimisation problem reduces the likelihood of selecting mixtures that are not stable. By carrying out multiple phase stability calculations for mixtures from a given set of solvents (design space), we showed that it is possible to train (single- or multi-layer) ANN classifiers that accurately describe phase boundaries. We investigated the performance of the classifier surrogates on two solvent design case studies. We showed that by embedding the classifiers trained on rigorous phase stability data in optimisation-based design and using a sufficiently high probability threshold, the optimiser could rule out the non-stable mixtures. Employing the classifier in the optimisation model also provides an interpretable metric, which in turn helps to discriminate between the designs more effectively.

Our findings from the case studies also suggest that there is potential to further improve the accuracy and reduce the computational effort in developing and using classifiers as phase stability surrogates. This could possibly be achieved by considering adaptive sampling strategies during the data generation step to generate denser data close to the phase boundary. Adaptive sampling could lead to a reduction in the number of samples required to train the classifiers, while providing greater accuracy in the design regions of interest. From an optimisation standpoint, it would be interesting to conduct a comparative assessment of different classification models such as support vector machines, decision trees or random forest, and to benchmark the performance of these models for phase stability in the context of optimisation-based CAM^{b}D. This could serve as a prelude to developing more general classifier surrogates that can be applied to mixtures not included in the training dataset, thereby expanding the CAM^{b}D design space.

We also plan to investigate the application of the proposed classifier surrogate methodology to other thermodynamic models such as the SAFT-γ Mie group contribution equation of state (Papaioannou et al., 2014). A noteworthy feature of classifier surrogates is that, while the choice of thermodynamic model may have an impact on the computational cost of data generation, it is not expected to affect the complexity of the ANN. Regardless of the thermodynamic model used, the cost reductions observed when solving the surrogate-based design problems are thus expected to be retained. Another extension of our work would be to consider the possible appearance of a vapour phase and/or mixtures with more than three components. This is expected to require minimal changes to the procedure as the training data retains its binary nature (i.e., whether the global phase is stable or not) regardless of the number of phases and components or of additional complexities in the phase behaviour (e.g., azeotropes). We anticipate that the main impact will be on data generation, which will require the introduction of an appropriate model for the vapour phase. In addition, if there are more than three components, more training data may be needed to provide sufficient coverage of the variable space and the resulting ANN may be larger. Finally, as part of future work, we aim to investigate the applicability of this approach on multi-unit solvent and process design problems such as the ones developed in Muhieddine et al. (2022), considering a larger number of solvents and supporting the end-to-end design of manufacturing processes.

CRediT authorship contribution statement

Tanuj Karia: Writing – original draft, Validation, Software, Methodology, Data curation, Conceptualization. **Gustavo Chaparro:** Writing – original draft, Software, Methodology, Data curation. **Benoît Chachuat:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Conceptualization. **Claire S. Adjiman:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Claire Adjiman reports financial support was provided by Engineering and Physical Sciences Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

TK is grateful to the Department of Chemical Engineering, Imperial College London, United Kingdom for a scholarship to support his PhD. GCH thanks the support provided by the President's PhD Scholarship, Imperial College London, United Kingdom. BC and CSA gratefully acknowledge funding by the Engineering and Physical Sciences Research Council (EPSRC), United Kingdom under grant EP/W003317/1. The authors are grateful to Dr. Suela Jonuzaj and Dr. Mohamad Muhieddine for assistance with developing the GAMS model for Case Study 2. The authors would like to thank Professor Alexander Mitsos and Dr. Thomas Bernet for insightful discussions.

Appendix A. Case study 1

Model with restricted stability condition. A complete mathematical formulation of Case Study 1 that uses the restricted stability condition (20) is presented in Eqs. (A.1)–(A.9) below.

$$\max_{\mathbf{x},\mathbf{y}} \quad x_{\mathsf{ibu}} \tag{A.1}$$

$$\text{s.t.} \quad \sum_{s\in\mathbb{S}} y_{c,s} = 1, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{A.2}$$

$$y_{\mathsf{a},s} + y_{\mathsf{b},s'} \le 1, \quad s,s' \in \mathbb{S} : \mathrm{ord}(s') \le \mathrm{ord}(s) \tag{A.3}$$

$$\sum_{c\in\mathbb{C}} x_c = 1 \tag{A.4}$$

$$\ln x_{\mathsf{ibu}} + \ln \gamma_{\mathsf{ibu}}(T,\mathbf{x},\mathbf{y}) = \frac{H^{\mathrm{fus}}_{\mathsf{ibu}}}{R}\left(\frac{1}{T^{\mathrm{m}}_{\mathsf{ibu}}} - \frac{1}{T}\right) \tag{A.5}$$

$$x_{\mathsf{a:b}} = \frac{x_{\mathsf{a}}}{x_{\mathsf{a}} + x_{\mathsf{b}}} \tag{A.6}$$

$$\frac{\partial \ln \gamma_{\mathsf{a:b}}(T, x_{\mathsf{a:b}}, \mathbf{y})}{\partial x_{\mathsf{a:b}}} + \frac{1}{x_{\mathsf{a:b}}} \ge 0 \tag{A.7}$$

$$x_c \in [0,1], \quad c \in \mathbb{C} \tag{A.8}$$

$$y_{c,s} \in \{0,1\}, \quad c \in \{\mathsf{a},\mathsf{b}\}, \ s \in \mathbb{S} \tag{A.9}$$

The problem entails the selection of two solvent molecules (a and b) with ibuprofen (ibu), leading to the ternary mixture components $\mathbb{C} := \{\mathsf{a},\mathsf{b},\mathsf{ibu}\}$. A set of nine solvent candidates is considered, $\mathbb{S} := \{\mathsf{methanol}, \mathsf{ethanol}, \mathsf{2-propanol}, \mathsf{acetone}, \mathsf{MIBK}, \mathsf{ethylacetate}, \mathsf{chloroform}, \mathsf{toluene}, \mathsf{water}\}$, which can be expressed in terms of the following 14 atom groups, given in UNIFAC notation by $\mathbb{G} := \{\mathsf{CH3}, \mathsf{CH2}, \mathsf{CH}, \mathsf{aCH}, \mathsf{aCCH3}, \mathsf{aCCH2}, \mathsf{aCCH}, \mathsf{OH}, \mathsf{CH3OH}, \mathsf{H2O}, \mathsf{CH3CO}, \mathsf{CH3COO}, \mathsf{COOH}, \mathsf{CHCl3}\}$. The values of the UNIFAC parameters Q_g, R_g and $a_{g,g'}$ are taken from Poling et al. (2001). The binary decision variables, $y_{c,s}$ with $c \in \{\mathsf{a},\mathsf{b}\}$ and $s \in \mathbb{S}$ encode the two selected solvents in the mixture, with the constraints (A.2) enforcing exactly two solvents and the constraints (A.3) ensuring that distinct solvents are selected and breaking symmetry among candidate solvent pairs.

The objective function (A.1) seeks to maximise the solubility of ibuprofen at a set temperature $T = 300$ K in the blend. The continuous decision variables x_c, with $c \in \mathbb{C}$ comprise the mole fractions of the three components, that sum up to 1 per constraint (A.4).

The solubility of ibuprofen in the solvent mixture is determined by the solid–liquid equilibrium condition in the constraint (A.5), with the enthalpy of fusion of ibuprofen $\Delta H^{\mathrm{fus}}_{\mathsf{ibu}} = 25.5$ kJ mol^{-1} and the corresponding melting temperature $T^{\mathrm{m}}_{\mathsf{ibu}} = 347.15$ K (Gracin and Rasmuson, 2002). The logarithm of the activity coefficient of ibuprofen, $\ln\gamma_{\mathsf{ibu}}$ is estimated using the UNIFAC model (cf. Section 2.2) for the ternary mixture $\mathbb{C}$, after replacing (Eqs. (15)–(16)) for the expressions below to account for the solvent selection encoded by the binary variables $\mathbf{y}$:

$$r_c = \sum_{s\in\mathbb{S}}\sum_{g\in\mathbb{G}} \upsilon^{(s)}_g R_g y_{c,s} \tag{A.10}$$

$$q_c = \sum_{s\in\mathbb{S}}\sum_{g\in\mathbb{G}} \upsilon^{(s)}_g Q_g y_{c,s} \tag{A.11}$$

Lastly, the inequality constraint (A.7) enforces the restricted stability condition (RSC), where $x_{\mathsf{a:b}}$ denotes the mole fraction of solvent a in a binary mixture of a and b only (Eq. (A.6)), that is, omitting the ibuprofen component. This constraint requires an expression for the derivative of (the logarithm of) the activity coefficient of solvent a in a binary mixture with b, $\gamma_{\mathsf{a:b}}$, where the mole fraction of b has been eliminated as an independent variable with $1-x_{\mathsf{a:b}}$ (cf. Section 2.1). Following the approach by Jonuzaj et al. (2016), approximate derivative expressions are used herein rather than the exact derivative expressions of the UNIFAC model, to improve computational tractability.

Model with classifier-based stability surrogate. A complete mathematical formulation of Case Study 1 that uses the classifier-based stability surrogate is presented in Eqs. (A.12)–(A.24) below.

$$\max_{\mathbf{x},\mathbf{y}} \quad x_{\mathsf{ibu}} \tag{A.12}$$

$$\text{s.t.} \quad \sum_{s\in\mathbb{S}} y_{c,s} = 1, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{A.13}$$

$$y_{\mathsf{a},s} + y_{\mathsf{b},s'} \le 1, \quad s,s' \in \mathbb{S} : \mathrm{ord}(s') \le \mathrm{ord}(s) \tag{A.14}$$

$$\sum_{c\in\mathbb{C}} x_c = 1 \tag{A.15}$$

$$\ln x_{\mathsf{ibu}} + \ln \gamma_{\mathsf{ibu}}(T,\mathbf{x},\mathbf{y}) = \frac{H^{\mathrm{fus}}_{\mathsf{ibu}}}{R}\left(\frac{1}{T^{\mathrm{m}}_{\mathsf{ibu}}} - \frac{1}{T}\right) \tag{A.16}$$

$$r_c = \sum_{s\in\mathbb{S}}\sum_{g\in\mathbb{G}} \upsilon^{(s)}_g R_g y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{A.17}$$

$$q_c = \sum_{s\in\mathbb{S}}\sum_{g\in\mathbb{G}} \upsilon^{(s)}_g Q_g y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{A.18}$$

$$\tilde{\alpha}_{0,n} = \frac{\alpha_n - \mu_{\alpha_n}}{\sigma_{\alpha_n}}, \quad n = 1 \dots N^0_{\mathrm{neu}} \tag{A.19}$$

$$\tilde{\alpha}_{1,n} = 1 - \frac{2}{1+\exp\left(2\mathbf{w}_{0,n}{}^{\intercal}\tilde{\boldsymbol{\alpha}}_0 + 2b_{0,n}\right)}, \quad n = 1 \dots N^1_{\mathrm{neu}} \tag{A.20}$$

$$p_{\mathrm{misc}} = \frac{1}{1+\exp\left(-\mathbf{w}_{1,1}{}^{\intercal}\tilde{\boldsymbol{\alpha}}_1 - b_{1,1}\right)} \tag{A.21}$$

$$p_{\mathrm{misc}} \ge p^{\mathrm{L}}_{\mathrm{misc}} \tag{A.22}$$

$$x_c \in [0,1], \quad c \in \mathbb{C} \tag{A.23}$$

$$y_{c,s} \in \{0,1\}, \quad c \in \{\mathsf{a},\mathsf{b}\}, \ s \in \mathbb{S} \tag{A.24}$$

The objective and constraints (A.12)–(A.16) remain identical to those defined in the counterpart model with restricted stability condition. In Case Study 1, the classifier inputs correspond to $\boldsymbol{\alpha} := [q_{\mathsf{a}} \ q_{\mathsf{b}} \ r_{\mathsf{a}} \ r_{\mathsf{b}} \ x_{\mathsf{a}} \ x_{\mathsf{b}}]^{\intercal}$, with r_c and q_c determined through the constraints (A.17)–(A.18). Each input is normalised using the mean μ_{α_n} and standard deviation σ_{α_n} from the training data set (Eq. (A.19)), then processed through the hidden layer of the classifier (Eq. (A.20)), where the hyperbolic tangent activation function is reformulated as $\tanh(x) := 1 - \frac{2}{1+\exp(2x)}$ (cf. Section 3.5). The stability constraint (A.22) defines a lower bound, $p^{\mathrm{L}}_{\mathrm{misc}}$ on the probability of miscibility, p_{misc} predicted by the classifier (Eq. (A.21)).

Appendix B. Case study 2

Model with restricted stability condition. A complete mathematical formulation of Case Study 2 that uses the restricted stability condition (20) is presented in Eqs. (B.1)–(B.24) below.

$$\max_{T^{\xi},\mathbf{x}^{\xi},\mathbf{y}} \quad 1 - \frac{n^{\mathrm{fin}}_{\mathsf{ibu}}}{n^{\mathrm{ini}}_{\mathsf{ibu}}} \tag{B.1}$$

$$\text{s.t.} \quad \frac{\sum_{c\in\{\mathsf{a},\mathsf{b}\}} m^{\mathrm{fin}}_c}{m^{\mathrm{ini}}_{\mathsf{ibu}} - m^{\mathrm{fin}}_{\mathsf{ibu}}} \ge 3.5 \tag{B.2}$$

$$\frac{\sum_{c\in\{\mathsf{a},\mathsf{b}\}} m^{\mathrm{fin}}_c/\rho_c}{m^{\mathrm{ini}}_{\mathsf{ibu}} - m^{\mathrm{fin}}_{\mathsf{ibu}}} \ge 4 \tag{B.3}$$

$$\sum_{s\in\mathbb{S}} y_{c,s} = 1, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.4}$$

$$y_{\mathsf{a},s} + y_{\mathsf{b},s} \le 1, \quad s \in \mathbb{S} \tag{B.5}$$

$$\rho_c = \sum_{s\in\mathbb{S}} \rho_s\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.6}$$

$$T^{\mathrm{m}}_s \ge T^{\mathrm{ini}} + T^{\mathrm{off}} - 600\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.7}$$

$$T^{\mathrm{b}}_s \le T^{\mathrm{fin}} - T^{\mathrm{off}} + 600\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.8}$$

$$T^{\mathrm{ini}} \le T^{\mathrm{m}}_s - T^{\mathrm{off}} + 600\left(1 - \sum_{c\in\{\mathsf{a},\mathsf{b}\}} y_{c,s}\right), \quad s \in \mathbb{S} \tag{B.9}$$

$$T^{\mathrm{fin}} \ge T^{\mathrm{b}}_s + T^{\mathrm{off}} - 600\left(1 - \sum_{c\in\{\mathsf{a},\mathsf{b}\}} y_{c,s}\right), \quad s \in \mathbb{S} \tag{B.10}$$

$$T^{\mathrm{ini}} \ge T^{\mathrm{fin}} \tag{B.11}$$

$$n^{\mathrm{ini}}_{\mathsf{a}} = n^{\mathrm{fin}}_{\mathsf{a}} \tag{B.12}$$

$$n^{\mathrm{ini}}_{\mathsf{b}} \le n^{\mathrm{fin}}_{\mathsf{b}} \tag{B.13}$$

$$n^{\mathrm{ini}}_{\mathrm{ibu}} \ge n^{\mathrm{fin}}_{\mathrm{ibu}} \tag{B.14}$$

$$\sum_{c\in\mathbb{C}} x^{\xi}_c = 1, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.15}$$

$$x^{\xi}_c = \frac{n^{\xi}_c}{\sum_{c'\in\mathbb{C}} n^{\xi}_{c'}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ c \in \{\mathsf{a},\mathsf{b}\} \tag{B.16}$$

$$m^{\xi}_c = n^{\xi}_c \sum_{s\in\mathbb{S}} M_s\, y_{c,s}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ c \in \{\mathsf{a},\mathsf{b}\} \tag{B.17}$$

$$m^{\xi}_{\mathrm{ibu}} = n^{\xi}_{\mathrm{ibu}}\, M_{\mathrm{ibu}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.18}$$

$$\ln x^{\xi}_{\mathrm{ibu}} + \ln \gamma_{\mathrm{ibu}}(T^{\xi}, \mathbf{x}^{\xi}, \mathbf{y}) = \frac{H^{\mathrm{fus}}_{\mathrm{ibu}}}{R}\left(\frac{1}{T^{\mathrm{m}}_{\mathrm{ibu}}} - \frac{1}{T^{\xi}}\right), \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.19}$$

$$x^{\xi}_{\mathsf{a}:\mathsf{b}} = \frac{x^{\xi}_{\mathsf{a}}}{x^{\xi}_{\mathsf{a}} + x^{\xi}_{\mathsf{b}}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.20}$$

$$\frac{\partial \ln \gamma_{\mathsf{a}:\mathsf{b}}}{\partial x_{\mathsf{a}:\mathsf{b}}}(T^{\xi}, x^{\xi}_{\mathsf{a}:\mathsf{b}}, \mathbf{y}) + \frac{1}{x^{\xi}_{\mathsf{a}:\mathsf{b}}} \ge 0, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.21}$$

$$T^{\mathrm{ini}} \in [T^{\min}, T^{\max}], \quad T^{\mathrm{fin}} = T^{\min} \tag{B.22}$$

$$x^{\xi}_c \in [0,1], \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ c \in \mathbb{C} \tag{B.23}$$

$$y_{c,s} \in \{0,1\}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.24}$$

The problem entails the selection of solvent and antisolvent molecules (a and b, respectively) that constitute the solvent blend. These are chosen from a list of eight solvent candidates, $\mathbb{S} := \{\mathrm{water}, \mathrm{n-pentane}, \mathrm{n-heptane}, \mathrm{ethanol}, 1-\mathrm{propanol}, \mathrm{i-butanol}, 1-\mathrm{pentanol}, \mathrm{acetone}\}$, which can be expressed in terms of the following nine atom groups, given in UNIFAC notation by $\mathbb{G} := \{\mathrm{CH3}, \mathrm{CH2}, \mathrm{CH}, \mathrm{aCH}, \mathrm{aCCH2}, \mathrm{aCCH}, \mathrm{CH3CO}, \mathrm{COOH}, \mathrm{H2O}\}$. The values of the UNIFAC parameters Q_g, R_g and $a_{g,g'}$ are taken from Poling et al. (2001). The selected solvents are encoded with the binary decision variables $y_{c,s}$, with $c \in \{\mathsf{a},\mathsf{b}\}$ and $s \in \mathbb{S}$ (Eq. (B.24)). The constraints (B.4) enforce exactly one solvent and one antisolvent, while the constraints (B.5) ensure that the selected solvent and antisolvent are different.

The continuous decision variables T^{ξ}, with $\xi \in \{\mathrm{ini},\mathrm{fin}\}$ represent the initial and final temperatures of the crystallisation process, along with the initial and final molar compositions of the binary solvent mixture $\mathbf{x}^{\xi}$ that sum up to 1 per constraint (B.15). The corresponding variable ranges are specified in Eqs. (B.22)–(B.23), with $T^{\min} = 293.15$ K and $T^{\max} = 318.15$ K.

The objective function (B.1) seeks to maximise the crystal yield of ibuprofen, ibu in the blend. n^{ξ}_{ibu} refers to the number of moles of ibuprofen, in either initial or final states, as constrained by (B.12)–(B.14). The operability constraints (B.2)–(B.3) enforce, respectively, a minimum solvent consumption on a mass basis of 3.5 $\mathrm{g_{solvent}\ g^{-1}_{crystal}}$ and on a volume basis of 4 $\mathrm{mL_{solvent}\ g^{-1}_{crystal}}$. The required densities of the candidate solvents are determined through the constraint (B.6). The constraints (B.16)–(B.18) furthermore relate the mole fractions x^{ξ}_c, numbers of moles n^{ξ}_c, and masses m^{ξ}_c of solvent, antisolvent and ibuprofen. The initial and final temperatures T^{ξ} are furthermore constrained by (B.7)–(B.10) based on the (experimental) melting point T^{m}_c and bubble point T^{b}_c of each solvent candidate, where $T^{\mathrm{off}} = 10$ K is a temperature offset chosen to prevent the process from operating too close to a solvent phase change, and 600 acts as a big-M value. In addition, the constraint (B.11) enforces a lower final temperature after cooling.

Lastly, the constraints (B.19)–(B.21) are akin to (A.5)–(A.7) in Case Study 1. They determine the solubility of ibuprofen in the solvent blend and enforce the restricted stability condition (RSC) for a single liquid phase.

Model with classifier-based stability surrogate. A complete mathematical formulation of Case Study 2 that uses the classifier-based stability surrogate is presented in Eqs. (B.25)–(B.52) below.

$$\max_{T^{\xi},\mathbf{x}^{\xi},\mathbf{y}} \quad 1 - \frac{n^{\mathrm{fin}}_{\mathrm{ibu}}}{n^{\mathrm{ini}}_{\mathrm{ibu}}} \tag{B.25}$$

$$\text{s.t.} \quad \frac{\sum_{c\in\{\mathsf{a},\mathsf{b}\}} m^{\mathrm{fin}}_c}{m^{\mathrm{ini}}_{\mathrm{ibu}} - m^{\mathrm{fin}}_{\mathrm{ibu}}} \ge 3.5 \tag{B.26}$$

$$\frac{\sum_{c\in\{\mathsf{a},\mathsf{b}\}} m^{\mathrm{fin}}_c/\rho_c}{m^{\mathrm{ini}}_{\mathrm{ibu}} - m^{\mathrm{fin}}_{\mathrm{ibu}}} \ge 4 \tag{B.27}$$

$$\sum_{s\in\mathbb{S}} y_{c,s} = 1, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.28}$$

$$y_{\mathsf{a},s} + y_{\mathsf{b},s} \le 1, \quad s \in \mathbb{S} \tag{B.29}$$

$$\rho_c = \sum_{s\in\mathbb{S}} \rho_s\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.30}$$

$$T^{\mathrm{m}}_s \ge T^{\mathrm{ini}} + T^{\mathrm{off}} - 600\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.31}$$

$$T^{\mathrm{b}}_s \le T^{\mathrm{fin}} - T^{\mathrm{off}} + 600\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.32}$$

$$T^{\mathrm{ini}} \le T^{\mathrm{m}}_s - T^{\mathrm{off}} + 600\left(1 - \sum_{c\in\{\mathsf{a},\mathsf{b}\}} y_{c,s}\right), \quad s \in \mathbb{S} \tag{B.33}$$

$$T^{\mathrm{fin}} \ge T^{\mathrm{b}}_s + T^{\mathrm{off}} - 600\left(1 - \sum_{c\in\{\mathsf{a},\mathsf{b}\}} y_{c,s}\right), \quad s \in \mathbb{S} \tag{B.34}$$

$$T^{\mathrm{ini}} \ge T^{\mathrm{fin}} \tag{B.35}$$

$$n^{\mathrm{ini}}_{\mathsf{a}} = n^{\mathrm{fin}}_{\mathsf{a}} \tag{B.36}$$

$$n^{\mathrm{ini}}_{\mathsf{b}} \le n^{\mathrm{fin}}_{\mathsf{b}} \tag{B.37}$$

$$n^{\mathrm{ini}}_{\mathrm{ibu}} \ge n^{\mathrm{fin}}_{\mathrm{ibu}} \tag{B.38}$$

$$\sum_{c\in\mathbb{C}} x^{\xi}_c = 1, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.39}$$

$$x^{\xi}_c = \frac{n^{\xi}_c}{\sum_{c'\in\mathbb{C}} n^{\xi}_{c'}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ c \in \{\mathsf{a},\mathsf{b}\} \tag{B.40}$$

$$m^{\xi}_c = n^{\xi}_c \sum_{s\in\mathbb{S}} M_s\, y_{c,s}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ c \in \{\mathsf{a},\mathsf{b}\} \tag{B.41}$$

$$m^{\xi}_{\mathrm{ibu}} = n^{\xi}_{\mathrm{ibu}}\, M_{\mathrm{ibu}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.42}$$

$$\ln x^{\xi}_{\mathrm{ibu}} + \ln \gamma_{\mathrm{ibu}}(T^{\xi}, \mathbf{x}^{\xi}, \mathbf{y}) = \frac{H^{\mathrm{fus}}_{\mathrm{ibu}}}{R}\left(\frac{1}{T^{\mathrm{m}}_{\mathrm{ibu}}} - \frac{1}{T^{\xi}}\right), \quad \xi \in \{\mathrm{ini},\mathrm{fin}\} \tag{B.43}$$

$$r_c = \sum_{s\in\mathbb{S}} \sum_{g\in\mathbb{G}} v^{(s)}_g R_g\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.44}$$

$$q_c = \sum_{s\in\mathbb{S}} \sum_{g\in\mathbb{G}} v^{(s)}_g Q_g\, y_{c,s}, \quad c \in \{\mathsf{a},\mathsf{b}\} \tag{B.45}$$

$$\tilde{\alpha}^{\xi}_{0,n} = \frac{\alpha^{\xi}_n - \mu_{\alpha_n}}{\sigma_{\alpha_n}}, \quad \xi \in \{\mathrm{ini},\mathrm{fin}\},\ n = 1 \ldots N^0_{\mathrm{neu}} \tag{B.46}$$

$$\tilde{\alpha}^{\xi}_{\ell,n} = 1 - \frac{2}{1+\exp\left(2\mathbf{w}_{\ell-1,n}{}^{\intercal}\tilde{\boldsymbol{\alpha}}^{\xi}_{\ell-1} + 2b_{\ell-1,n}\right)}, \quad \xi \in \{\mathsf{ini},\mathsf{fin}\},\ \ell = 1\ldots 3,\ n = 1\ldots N^{\ell}_{\mathrm{neu}} \tag{B.47}$$

$$p^{\xi}_{\mathrm{misc}} = \frac{1}{1+\exp\left(-\mathbf{w}_{N_{\mathrm{hid}},1}{}^{\intercal}\tilde{\boldsymbol{\alpha}}^{\xi}_{N_{\mathrm{hid}}} - b_{N_{\mathrm{hid}},1}\right)}, \quad \xi \in \{\mathsf{ini},\mathsf{fin}\} \tag{B.48}$$

$$p^{\xi}_{\mathrm{misc}} \geq p^{\mathrm{L}}_{\mathrm{misc}}, \quad \xi \in \{\mathsf{ini},\mathsf{fin}\} \tag{B.49}$$

$$T^{\mathsf{ini}} \in [T^{\min}, T^{\max}], \quad T^{\mathsf{fin}} = T^{\min} \tag{B.50}$$

$$x^{\xi}_{c} \in [0,1], \quad \xi \in \{\mathsf{ini},\mathsf{fin}\},\ c \in \mathbb{C} \tag{B.51}$$

$$y_{c,s} \in \{0,1\}, \quad c \in \{\mathsf{a},\mathsf{b}\},\ s \in \mathbb{S} \tag{B.52}$$

The objective and constraints (B.25)–(B.43) remain identical to those defined in the counterpart model with restricted stability condition. In Case Study 2, the classifier inputs correspond to $\boldsymbol{\alpha}^{\xi} := [q_{\mathsf{a}}\ q_{\mathsf{b}}\ r_{\mathsf{a}}\ r_{\mathsf{b}}\ x^{\xi}_{\mathsf{a}}\ x^{\xi}_{\mathsf{b}}\ T^{\xi}]^{\intercal}$, with r_c and q_c determined through the constraints (B.44)–(B.45). Each input is normalised using the mean μ_{α_n} and standard deviation σ_{α_n} from the training data set (Eq. (B.46)), then processed through the three hidden layers of the classifier (Eq. (B.47)), where the hyperbolic tangent activation function is reformulated as $\tanh(x) := 1 - \frac{2}{1+\exp(2x)}$ (cf. Section 3.5). The stability constraint (B.49) defines a lower bound, $p^{\mathrm{L}}_{\mathrm{misc}}$ on the probability of miscibility, p^{ξ}_{misc} predicted by the classifier for both the initial and final states (Eq. (B.48)).

Appendix C. Impact of ibuprofen addition on phase stability of a solvent mixture

Adding ibuprofen to a binary solvent mixture may impact its phase stability in four different ways:

- The solvent mixture is originally completely miscible and ibuprofen addition induces a two-liquid phase region; e.g., methanol + water and acetone + water mixtures
- The solvent mixture is originally completely immiscible and ibuprofen addition stabilises the mixture for certain compositions; e.g., toluene + water and ethanol + water mixtures
- The solvent mixture is originally partially miscible and ibuprofen addition induces a larger two-liquid phase region; e.g., 2-propanol + water mixture
- The solvent mixture is originally partially miscible and ibuprofen addition induces a smaller two-liquid phase region; e.g., chloroform + water and n-pentane + ethanol mixtures

Refer to the datasets `Case1_tpd_calc_bin.csv` and `Case1_tpd_calc_slle.csv` for Case Study 1 and to the datasets `Case2_tpd_calc_bin.csv` and `Case2_tpd_calc_slle.csv` for Case Study 2 that are provided in the ESI for further information.

Appendix D. Generating top performing mixtures

Integer cuts can be added to the CAM$^{\mathrm{b}}$D models to generate a ranked list of top performing mixtures. A binary mixture encoded by $\mathbf{y}^{(i)}$ is excluded by appending the following integer cut (Balas and Jeroslow, 1972):

$$\sum_{(c,s)\in\mathbb{B}} y_{c,s} - \sum_{(c,s)\notin\mathbb{B}} y_{c,s} \leq 1 \tag{D.1}$$

with $\mathbb{B}^{(i)} := \{(c,s) : y^{(i)}_{c,s} = 1\}$.

Data availability

Data underlying this article referred to as Electronic supplementary information (ESI) in the main article can be accessed on Github at https://github.com/tk3016/SI_CBS_CAMbD.git.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. Software available from tensorflow.org.

Anderson, R., Huchette, J., Ma, W., Tjandraatmadja, C., Vielma, J.P., 2020. Strong mixed-integer programming formulations for trained neural networks. Math. Program. 183, 3–39. http://dx.doi.org/10.1007/s10107-020-01474-5.

Atkins, P., de Paula, J., Keeler, J., 2023. Atkins' Physical Chemistry, twelfth ed. Oxford University Press, Oxford.

Baker, L.E., Pierce, A.C., Luks, K.D., 1982. Gibbs energy analysis of phase equilibria. Soc. Pet. Eng. J. 22, 731–742. http://dx.doi.org/10.2118/9806-PA.

Balas, E., Jeroslow, R., 1972. Canonical cuts on the unit hypercube. SIAM J. Appl. Math. 23, 61–79. http://dx.doi.org/10.1137/0123007.

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

Bestuzheva, K., Chmiela, A., Müller, B., Serrano, F., Vigerske, S., Wegscheider, F., 2023. Global optimization of mixed-integer nonlinear programs with SCIP 8. http://dx.doi.org/10.48550/arXiv.2301.00587, arXiv preprint arXiv:2301.00587.

Binous, H., Bellagi, A., 2021. Calculation of ternary liquid-liquid equilibrium data using arc-length continuation. Eng. Rep. 3, 012296. http://dx.doi.org/10.1002/eng2.12296, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/eng2.12296. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/eng2.12296. e12296 ENG-2019-05-0285.R1.

Bollas, G.M., Barton, P.I., Mitsos, A., 2009. Bilevel optimization formulation for parameter estimation in vapor-liqui(-liquid) phase equilibrum problems. Chem. Eng. Sci. 64, 1768–1783. http://dx.doi.org/10.1016/j.ces.2009.01.003.

Brownlee, J., 2017. Why one-hot encode data in machine learning. URL: https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/. (Accessed 07 September 2023).

Buxton, A., Livingston, A.G., Pistikopoulos, E.N., 1999. Optimal design of solvent blends for environmental impact minimization. AIChE J. 45, 817–843. http://dx.doi.org/10.1002/aic.690450415.

Chaparro, G., Mejía, A., 2020. Phasepy: A python based framework for fluid phase equilibria and interfacial properties computation. J. Comput. Chem. 41, 2504–2526. http://dx.doi.org/10.1002/jcc.26405.

Chicco, D., Jurman, G., 2020. The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics 21, 1–13. http://dx.doi.org/10.1186/s12864-019-6413-7.

Conte, E., Gani, R., Ng, K.M., 2011. Design of formulated products: A systematic methodology. AIChE J. 57, 2431–2449. http://dx.doi.org/10.1002/aic.12458.

Coutinho, J.A., Andersen, S.I., Stenby, E.H., 1995. Evaluation of activity coefficient models in prediction of alkane solid–liquid equilibria. Fluid Phase Equilib. 103, 23–39. http://dx.doi.org/10.1016/0378-3812(94)02600-6.

Dahmen, M., Marquardt, W., 2017. Model-based formulation of biofuel blends by simultaneous product and pathway design. Energy Fuels 31, 4096–4121. http://dx.doi.org/10.1021/acs.energyfuels.7b00118.

Dohrn, S., Luebbert, C., Lehmkemper, K., Kyeremateng, S.O., Degenhardt, M., Sadowski, G., 2021. Solvent mixtures in pharmaceutical development: Maximizing the API solubility and avoiding phase separation. Fluid Phase Equilib. 548, 113200. http://dx.doi.org/10.1016/j.fluid.2021.113200.

Dortmund Data Bank, 2024. Parameters of the original unifac model. https://www.ddbst.com/published-parameters-unifac.html. (Accessed 31 May 2024).

Elhassan, A.E., Tsvetkov, S.G., Craven, R.J., Stateva, R.P., Wakeham, W.A., 1998. A rigorous mathematical proof of the area method for phase stability. Ind. Eng. Chem. Res. 37, 1483–1489. http://dx.doi.org/10.1021/ie970265v.

Fonseca-Pérez, R., Bonilla-Petriciolet, A., Tapia-Picazo, J., Jaime-Leal, J., 2021. A reconsideration on the resolution of phase stability analysis using stochastic global optimization methods: Proposal of a reliable set of benchmark problems. Fluid Phase Equilib. 548, 113180. http://dx.doi.org/10.1016/j.fluid.2021.113180.

Fredenslund, A., Jones, R.L., Prausnitz, J.M., 1975. Group-contribution estimation of activity coefficients in nonideal liquid mixtures. AIChE J. 21, 1086–1099. http://dx.doi.org/10.1002/aic.690210607.

Gaganis, V., Varotsis, N., 2012. Non-iterative phase stability calculations for process simulation using discriminating functions. Fluid Phase Equilib. 314, 69–77. http://dx.doi.org/10.1016/j.fluid.2011.10.021.

GAMS Development Corporation, 2023. GAMS – the solver manuals: SBB. URL: https://www.gams.com/42/docs/S_SBB.html.

Gracin, S., Rasmuson, Å.C., 2002. Solubility of phenylacetic acid, p-hydroxyphenylacetic acid, p-aminophenylacetic acid, p-hydroxybenzoic acid, and ibuprofen in pure solvents. J. Chem. Eng. Data 47, 1379–1383. http://dx.doi.org/10.1021/je0255170.

Grimstad, B., Andersson, H., 2019. ReLU networks as surrogate models in mixed-integer linear programs. Comput. Chem. Eng. 131, 106580. http://dx.doi.org/10.1016/j.compchemeng.2019.106580.

Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q., 2017. On calibration of modern neural networks. In: Precup, D., Teh, Y.W. (Eds.), Proceedings of the 34th International Conference on Machine Learning. pp. 1321–1330, URL: http://proceedings.mlr.press/v70/guo17a/guo17a.pdf.

Gupta, A.K., Raj Bishnoi, P., Kalogerakis, N., 1991. A method for the simultaneous phase equilibria and stability calculations for multiphase reacting and non-reacting systems. Fluid Phase Equilib. 63, 65–89. http://dx.doi.org/10.1016/0378-3812(91)80021-M.

Holmström, K., Quttineh, N.H., Edvall, M.M., 2008. An adaptive radial basis algorithm (ARBF) for expensive black-box mixed-integer constrained global optimization. Optim. Eng. 9, 311–339. http://dx.doi.org/10.1007/s11081-008-9037-3.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Netw. 2, 359–366. http://dx.doi.org/10.1016/0893-6080(89)90020-8.

Hosmer, Jr., D.W., Lemeshow, S., Sturdivant, R.X., 2013. Applied Logistic Regression. Vol. 398. John Wiley & Sons.

Hua, J.Z., Maier, R.W., Tessier, S.R., Brennecke, J.F., Stadtherr, M.A., 1999. Interval analysis for thermodynamic calculations in process design: a novel and completely reliable approach. Fluid Phase Equilib. 158, 607–615. http://dx.doi.org/10.1016/S0378-3812(99)00105-3.

James, G., Witten, D., Hastie, T., Tibshirani, R., Taylor, J., 2023. An Introduction to Statistical Learning with Applications in Python, first ed. In: Springer Texts in Statistics, Springer International Publishing, Cham.

Jonuzaj, S., Akula, P.T., Kleniati, P.M., Adjiman, C.S., 2016. The formulation of optimal mixtures with generalized disjunctive programming: A solvent design case study. AIChE J. 62, 1616–1633. http://dx.doi.org/10.1002/aic.15122.

Jonuzaj, S., Watson, O.L., Ottoboni, S., Price, C.J., Sefcik, J., Galindo, A., Jackson, G., Adjiman, C.S., 2020. Computer-aided solvent mixture design for the crystallisation and isolation of mefenamic acid. In: Pierucci, S., Manenti, F., Bozzano, G.L., Manca, D. (Eds.), 30th European Symposium on Computer Aided Process Engineering. In: Computer Aided Chemical Engineering, vol. 48, Elsevier, pp. 649–654. http://dx.doi.org/10.1016/B978-0-12-823377-1.50109-9.

Karacasulu, K., Jungen, D., Djelassi, H., Najman, J., Bongartz, D., Grave, L., Mitsos, A., 2021. Design with equilibrium processes embedded: Global optimization with guaranteed phase stability. In: 2020 Virtual AIChE Annual Meeting Proceedings, 2020 AIChE Annual Meeting, Online, 16 Nov 2020–20 Nov 2020. American Institute of Chemical Engineers, New York, NY, p. 218d, URL: https://publications.rwth-aachen.de/record/810235.

Karunanithi, A.T., Achenie, L.E., Gani, R., 2005. A new decomposition-based computer-aided molecular/mixture design methodology for the design of optimal solvents and solvent mixtures. Ind. Eng. Chem. Res. 44, 4785–4797. http://dx.doi.org/10.1021/ie049328h.

Karunanithi, A.T., Achenie, L.E., Gani, R., 2006. A computer-aided molecular design framework for crystallization solvent design. Chem. Eng. Sci. 61, 1247–1260. http://dx.doi.org/10.1016/j.ces.2005.08.031.

Kashinath, A., Szulczewski, M.L., Dogru, A.H., 2018. A fast algorithm for calculating isothermal phase behavior using machine learning. Fluid Phase Equilib. 465, 73–82. http://dx.doi.org/10.1016/j.fluid.2018.02.004.

Kim, S.H., Boukouvala, F., 2020. Surrogate-based optimization for mixed-integer nonlinear problems. Comput. Chem. Eng. 140, 106847. http://dx.doi.org/10.1016/j.compchemeng.2020.106847.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. http://dx.doi.org/10.48550/arXiv.1412.6980, arXiv preprint arXiv:1412.6980.

Kunde, C., Keßler, T., Linke, S., McBride, K., Sundmacher, K., Kienle, A., 2019. Surrogate modeling for liquid–liquid equilibria using a parameterization of the binodal curve. Processes 7 (753), http://dx.doi.org/10.3390/pr7100753.

LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.R., 2012. Efficient BackProp, second ed. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 9–48. http://dx.doi.org/10.1007/978-3-642-35289-8_3, (Chapter 1).

Mah, A.X.Y., Chin, H.H., Neoh, J.Q., Aboagwa, O.A., Thangalazhy-Gopakumar, S., Chemmangattuvalappil, N.G., 2019. Design of bio-oil additives via computer-aided molecular design tools and phase stability analysis on final blends. Comput. Chem. Eng. 123, 257–271. http://dx.doi.org/10.1016/j.compchemeng.2019.01.008.

Manrique, J., Martinez, F., 2007. Solubility of ibuprofen in some ethanol+ water cosolvent mixtures at several temperatures. Lat. Am. J. Pharm. 26, 344–354.

McDonald, C.M., Floudas, C.A., 1995. Global optimization for the phase stability problem. AIChE J. 41, 1798–1814. http://dx.doi.org/10.1002/aic.690410715.

McDonald, C.M., Floudas, C.A., 1997. GLOPEQ: A new computational tool for the phase and chemical equilibrium problem. Comput. Chem. Eng. 21, 1–23. http://dx.doi.org/10.1016/0098-1354(95)00250-2.

Michelsen, M.L., 1982. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. http://dx.doi.org/10.1016/0378-3812(82)85001-2.

Michelsen, M., Mollerup, J., 2007. Thermodynamic Models: Fundamentals & Computational Aspects. Tie-Line Publications.

Mitsos, A., Barton, P.I., 2007. A dual extremum principle in thermodynamics. AIChE J. 53, 2131–2147. http://dx.doi.org/10.1002/aic.11230.

Mitsos, A., Bollas, G.M., Barton, P.I., 2009. Bilevel optimization formulation for parameter estimation in liquid–liquid phase equilibrium problems. Chem. Eng. Sci. 64, 548–559. http://dx.doi.org/10.1016/j.ces.2008.09.034.

Muhieddine, M.H., Viswanath, S.K., Armstrong, A., Galindo, A., Adjiman, C.S., 2022. Model-based solvent selection for the synthesis and crystallisation of pharmaceutical compounds. Chem. Eng. Sci. 264, 118125. http://dx.doi.org/10.1016/j.ces.2022.118125.

Müller, J., 2016. MISO: mixed-integer surrogate optimization framework. Optim. Eng. 17, 177–203. http://dx.doi.org/10.1007/s11081-015-9281-2.

Müller, J., Shoemaker, C.A., Piché, R., 2013. SO-MI: A surrogate model algorithm for computationally expensive nonlinear mixed-integer black-box global optimization problems. Comput. Oper. Res. 40, 1383–1400. http://dx.doi.org/10.1016/j.cor.2012.08.022.

Nichita, D.V., Gomez, S., 2009. Efficient location of multiple global minima for the phase stability problem. Chem. Eng. J. 152, 251–263. http://dx.doi.org/10.1016/j.cej.2009.05.011.

Nichita, D.V., Gomez, S., Luna, E., 2002. Phase stability analysis with cubic equations of state by using a global optimization method. Fluid Phase Equilib. 194, 411–437. http://dx.doi.org/10.1016/S0378-3812(01)00779-8.

Nick Sahinidis, 2023. GAMS – the solver manuals: BARON. URL: https://www.gams.com/42/docs/S_BARON.html.

Niculescu-Mizil, A., Caruana, R., 2005. Predicting good probabilities with supervised learning. In: Proceedings of the 22nd International Conference on Machine Learning. Association for Computing Machinery, New York, NY, USA, pp. 625–632. http://dx.doi.org/10.1145/1102351.1102430.

Pan, H., Firoozabadi, A., 1998. Complex multiphase equilibrium calculations by direct minimization of gibbs free energy by use of simulated annealing. SPE Reserv. Eval. Eng. 1, 36–42. http://dx.doi.org/10.2118/37689-PA.

Papaioannou, V., Lafitte, T., Avendaño, C., Adjiman, C.S., Jackson, G., Müller, E.A., Galindo, A., 2014. Group contribution methodology based on the statistical associating fluid theory for heteronuclear molecules formed from Mie segments. J. Chem. Phys. 140, 054107. http://dx.doi.org/10.1063/1.4851455.

Pereira, F.E., Jackson, G., Galindo, A., Adjiman, C.S., 2010. A duality-based optimisation approach for the reliable solution of (P, T) phase equilibrium in volume-composition space. Fluid Phase Equilib. 299, 1–23. http://dx.doi.org/10.1016/j.fluid.2010.08.001.

Pereira, F.E., Jackson, G., Galindo, A., Adjiman, C.S., 2012. The HELD algorithm for multicomponent, multiphase equilibrium calculations with generic equations of state. Comput. Chem. Eng. 36, 99–118. http://dx.doi.org/10.1016/j.compchemeng.2011.07.009.

Poling, B.E., Prausnitz, J.M., O'Connell, J.P., 2001. Properties of Gases and Liquids, fifth ed. McGraw-Hill Education, New York.

Poort, J.P., Ramdin, M., van Kranendonk, J., Vlugt, T.J., 2019. Solving vapor–liquid flash problems using artificial neural networks. Fluid Phase Equilib. 490, 39–47. http://dx.doi.org/10.1016/j.fluid.2019.02.023.

Prausnitz, J., Anderson, T., Grens, E., Eckert, C., Hsieh, R., O'Connel, J., 1980. Computer Calculations for Multicomponent Vapor-Liquid and Liquid-Liquid Equilibria. Prentice-Hall, Inc.

Prausnitz, J.M., Lichtenthaler, R.N., De Azevedo, E.G., 1998. Molecular Thermodynamics of Fluid-Phase Equilibria. Pearson Education.

Rashid, K., Ambani, S., Cetinkaya, E., 2013. An adaptive multiquadric radial basis function method for expensive black-box mixed-integer nonlinear constrained optimization. Eng. Optim. 45, 185–206. http://dx.doi.org/10.1080/0305215X.2012.665450.

Rashid, A., White, E.T., Howes, T., Litster, J.D., Marziano, I., 2014. Effect of solvent composition and temperature on the solubility of ibuprofen in aqueous ethanol. J. Chem. Eng. Data 59, 2699–2703. http://dx.doi.org/10.1021/je400819z.

Schmitz, J.E., Zemp, R.J., Mendes, M.J., 2006. Artificial neural networks for the solution of the phase stability problem. Fluid Phase Equilib. 245, 83–87. http://dx.doi.org/10.1016/j.fluid.2006.02.013.

Schweidtmann, A.M., Mitsos, A., 2019. Deterministic global optimization with artificial neural networks embedded. J. Optim. Theory Appl. 180, 925–948. http://dx.doi.org/10.1007/s10957-018-1396-0.

Smith, J.M., Van Ness, H.C., Abbott, M.M., 2001. Introduction to Chemical Engineering Thermodynamics, sixth ed. In: McGraw-Hill Chemical Engineering series, McGraw-Hill, Boston.

Sobol', I., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput. Math. Math. Phys. 7, 86–112. http://dx.doi.org/10.1016/0041-5553(67)90144-9.

Stadtherr, M.A., Schnepper, C.A., Brennecke, J.F., 1995. Robust Phase Stability and Analysis using Interval Methods. In: AIChE Symposium Series, American Institute of Chemical Engineers, New York, NY, pp. 356–359, 1971-c2002.

Sun, A.C., Seider, W.D., 1995. Homotopy-continuation method for stability analysis in the global minimization of the Gibbs free energy. Fluid Phase Equilib. 103, 213–249. http://dx.doi.org/10.1016/0378-3812(94)02579-P.

Tsay, C., Kronqvist, J., Thebelt, A., Misener, R., 2021. Partition-based formulations for mixed-integer optimization of trained ReLU neural networks. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (Eds.), Advances in Neural Information Processing Systems. Curran Associates, Inc., pp. 3068–3080, URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/17f98ddf040204eda0af36a108cbdea4-Paper.pdf.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, I., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., Vijaykumar, A., Bardelli, A.P., Rothberg, A., Hilboll, A., Kloeckner, A., Scopatz, A., Lee, A., Rokem, A., Woods, C.N., Fulton, C., Masson, C., Häggström, C., Fitzgerald, C.,

Nicholson, D.A., Hagen, D.R., Pasechnik, D.V., Olivetti, E., Martin, E., Wieser, E., Silva, F., Lenders, F., Wilhelm, F., Young, G., Price, G.A., Ingold, G.L., Allen, G.E., Lee, G.R., Audren, H., Probst, I., Dietrich, J.P., Silterra, J., Webber, J.T., Slavič, J., Buchner, J., Kulick, J., Schönberger, J.L., de Miranda Cardoso, J.V., Reimer, J., Harrington, J., Rodríguez, J.L.C., Nunez-Iglesias, J., Kuczynski, J., Tritz, K., Thoma, M., Newville, M., Kümmerer, M., Bolingbroke, M., Tartre, M., Pak, M., Smith, N.J., Nowaczyk, N., Shebanov, N., Pavlyk, O., Brodtkorb, P.A., Lee, P., McGibbon, R.T., Feldbauer, R., Lewis, S., Tygier, S., Sievert, S., Vigna, S., Peterson, S., More, S., Pudlik, T., Oshima, T., Pingel, T.J., Robitaille, T.P., Spura, T., Jones, T.R., Cera, T., Leslie, T., Zito, T., Krauss, T., Upadhyay, U., Halchenko, Y.O., Vázquez-Baeza, Y., 2020. SciPy 1.0: fundamental algorithms for scientific computing in python. Nature Methods 17, 261–272. http://dx.doi.org/10.1038/s41592-019-0686-2, arXiv:1907.10121.

Wasylkiewicz, S.K., Sridhar, L.N., Doherty, M.F., Malone, M.F., 1996. Global stability analysis and calculation of liquid- liquid equilibrium in multicomponent mixtures. Ind. Eng. Chem. Res. 35, 1395–1408. http://dx.doi.org/10.1021/ie950049r.

Watson, O.L., Jonuzaj, S., McGinty, J., Sefcik, J., Galindo, A., Jackson, G., Adjiman, C.S., 2021. Computer aided design of solvent blends for hybrid cooling and antisolvent crystallization of active pharmaceutical ingredients. Organic Process Res. Dev. 25, 1123–1142. http://dx.doi.org/10.1021/acs.oprd.0c00516.

Xu, G., Haynes, W.D., Stadtherr, M.A., 2005. Reliable phase stability analysis for asymmetric models. Fluid Phase Equilib. 235, 152–165. http://dx.doi.org/10.1016/j.fluid.2005.06.016.

Yang, D., Balaprakash, P., Leyffer, S., 2022. Modeling design and control problems involving neural network surrogates. Comput. Optim. Appl. 1–42. http://dx.doi.org/10.1007/s10589-022-00404-9.

Yu, Z.Q., Zhang, F.K., Tan, R.B.H., 2021. Liquid–liquid phase separation in pharmaceutical crystallization. Chem. Eng. Res. Des. 174, 19–29. http://dx.doi.org/10.1016/j.cherd.2021.07.028.