Group Contribution-based Property Modeling for Chemical Product Design: A Perspective in the AI Era

Vipul Mann[a], Rafiqul Gani[b,c], Venkat Venkatasubramanian[a,*]

[a] *Department of Chemical Engineering, Columbia University, New York, NY, 10027, USA*
[b] *PSE for SPEED Company, Ordrup Jagtvej 42D, DK-2920, Charlottenlund, Denmark*
[c] *Sustainable Energy and Environment Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China*

Abstract

We provide a perspective of the challenges and opportunities for the group contribution approach for property prediction modeling with respect to their use in the design of chemical-based products in the modern era of artificial intelligence. In particular, we discuss issues related to the correct formulation of the product design problem, representation of molecular structures for property prediction as well as generation of product candidates, regression of property model parameters, and the integration of property related data and models with product design methods and tools using several conceptual examples. The need for developing appropriate hybrid AI models is described and recommendations for future work are presented.

Keywords: group contribution; property prediction; hybrid modeling; artificial intelligence; chemical product design

Symbols

$\mathbf{c}_x$	vector of regressed contribution for group x (first, second, or third)
C_{pL}	heat of capacity, liquid
C_{pV}	heat of capacity, vapor
$\mathbf{e}_i$	prediction errors attributable to measurement errors or modeling inaccuracies for the i^{th} property

*Corresponding author
Email addresses: vm2583@columbia.edu (Vipul Mann), rgani2018@gmail.com (Rafiqul Gani), venkat@columbia.edu (Venkat Venkatasubramanian)

$\mathbf{f}(.)$	regression function that maps input molecular descriptors to target property
F, S, T	superscripts in equation 2 indicating first, second, and third order group contributions
H_L	liquid enthalpy
H_{sp}	Hansen solubility parameter
H_{vap}	heat of vaporization
H_V	vapor enthalpy
$K(.)$	kernel function to transform data to high dimensions
$LC50$	represents concentration in water that kills half of Fathead Minnow in 96 hours (or Daphnia Magna in 48 hours)
$LD50$	represents the amount of chemical (mass of the chemical per body weight of rat) which when orally ingested kills half of rats
$LogP$	logarithm of the Octanol-Water partition coefficient
M_s	miscibility index ($M_s = 1$ implies miscible mixture, $M_s = 2$ implies immiscible mixture
M_w	molecular weight
nf, ns, nt	first, second, and third order group types in equation 2
$\mathbf{n}_x$	number of occurrences of group x (first, second, or third order) in equation 2
P_c	critical pressure
$\hat{p}_{j,i}$	predicted value for property of interest i for the j^{th} chemical
$\mathbf{p}_j$	vector of properties for chemical or system j
pKa	negative log of acid dissociation constant
P	Pressure (conditions at which properties are to be estimated)
P^{sat}	saturation pressure
$\mathbf{s}_j$	vector of molecular structural variables that describe the chemical (for molecular design)

T	Temperature (conditions at which properties are to be estimated)
T_b	normal boiling point
T_{bub}	bubble point temperature
T_c	critical temperature
T_{dew}	dew point temperature
w_k	tunable parameter to control regression of model parameters
$\mathbf{x}_i$	vector of molecular descriptors for i^{th} molecule
$\mathbf{x}_j$	vector of compositions of the chemicals present in the system for which the properties are needed
$\hat{y}$	predicted value for variable y
$\boldsymbol{\beta}_i$	vector of model parameters for property i
γ_i^L	liquid phase activity coefficient of compound i in phase L
δ_P	Hildebrandt solubility parameter
η	viscosity
$\boldsymbol{\Theta}$	matrix of regression coefficients
λ, λ^*	Lagrange multipliers
ρ	density
AI	artificial intelligence
Evap	evaporation rate
GC	group contribution
GWP	global warming potential
HLB	hydrophilic-lipophilic balance
HLD	hydrophilic-lipophilic deviation
ML	machine learning
ODP	ozone depletion potential
RVP	Reid vapor pressure

1. Introduction

Chemicals-based product design refers to products where one or more chemicals define the functions of the product. As Gani and Ng [1] observed, these products can be classified as single species, blends & mixtures, formulations, and devices. In each of these products, selected chemicals are added to provide one or more of the desired functions of the product. For example, as a solvent, the selected chemical must dissolve the solute as well as provide other desired operational properties. In the case of blends or mixtures, for example in a lubricant or liquid fuel, the additive chemical enhances the functions of the product, such as absorbing heat for the former and providing heat for the later. For formulations, which contain several chemicals with specific product related functions, an active ingredient provides the main function of the product, such as the protection of a surface, while a solvent delivers the active ingredient on the surface and vaporizes out. In devices, the added chemical contributes to the function of the device during its operation, such as the refrigerant for cooling or release of aroma as a mosquito repellent.

In all the chemicals-based products (called products from here on), chemicals deliver the desired product functions through their properties. However, for specific products, their desired functions are different and they therefore require different sets of properties [2, 3]. Table 1 gives a selected list of product functions and their related chemical properties. Product design methods therefore need to use different types of property estimation methods for different types of properties in the various steps of the product design work-flow, if measured data are not available.

Note that property estimation could be regarded as a forward computational problem (given the molecule or mixture, estimate its properties), and product design could be regarded as the inverse problem (given a set of target functions defined by properties, find the molecule or mixture that matches the targets). For computer-aided product design, therefore, a collection of measured data as well as property estimation models are needed. A review on computer-aided chemical product design can be found in Adjiman et al. [4] and Zhang et al. [5]. Perspectives on thermodynamic properties prediction for chemical process and product design is given by O'Connell et al. [6], while reviews on group contribution based pure compound properties prediction is given by Gani [7] and on phase equilibrium prediction is given by Gmehling et al. [8].

Table 1: Product functions and their relation to chemical properties

Product			Related chemical properties
Type	Example	Function	
Single species	Solvents (for extraction)	Liquid state	Boiling point, melting point
		Dissolves solute	Solubility parameter, activity coefficients
		Non toxic	LC50, LD50
	Refrigerant	Ability to cool	Boiling point, critical temperature, critical pressure
		Environmental impact	Ozone depletion potential, global warming potential
		Non-flammable	Flash point
Blends	Jet-fuel	Burning ability	Reid vapor pressure
		Engine efficiency	Higher heating value, density
		Environmental impact	Human toxicity (LC50), CO2 emission, emission during combustion
	Lubricant	Ability to lubricate	Kinematic viscosity
		Prevent wear & tear	Friction coefficient
Formulations	Insect repellant	Long lasting	Evaporation rate & solvent
		Water-based	Water miscible solvents
		Stability	Phase stability
	Detergent (emulsion)	Ability to clean	Surface tension, HLB
		Foamability	Micelle concentration, surface tension
		Stability	Cloud point, Krafft temperature, HLD
Device	Inhaler	Deliver drug	Biological activity as a function of LogP

In general, property estimation methods could be broadly classified as those that are predictive and those that are empirical (or semi-empirical) functions, depending on the measured data used to develop the models and their extrapolation features as proposed by Gani [7]. For example, in the case of phase equilibrium related properties, the group-contribution based UNIFAC method [9] is predictive in terms of mixtures handled because the regressed group interaction parameters can be extrapolated to many molecules outside the training set, while the UNIQUAC method [10] is not predictive because the regressed molecular interaction parameters can only be used for the specific molecules present in the chemical system, although they may be extrapolated in terms of temperature, pressure and composition.

In computer-aided product design, the work-flows for property estimation and product design are integrated and the methods could be mainly classified as database search [11], generate and test [12], mathematical programming [13–15], AI-techniques [16, 17]. Review of recent literature shows an increasing trend on the use of machine learning based models for property estimation [18] and AI based techniques to identify potential molecular structures with promising properties [19] as well as integrated mathematical programming solution approaches for product design as proposed by Liu et al. [15]. While good advances have been made in systematic computer-aided product design, it

has been pointed out in [20] that significant improvements (and extensions) of current methods and tools are needed to address the challenges we are facing on earth. For example, the ability to find significantly better products through increased search space; guarantee of product safety; ability to reuse, recycle and/or replace; etc., are challenges where reliability of chemical properties and their related product functions play an important role. Note that questions like "what to make" are related to product design, while questions like "how to make" are related to design of processes to manufacture the designed product. Integrated product-process design involves simultaneous design of the product and design of its sustainable process manufacturing system. In this paper, we concentrate only on selected aspects of product design.

Regarding perspective papers, we take the view that they should have the form of a focused review that provides the reader with an overview on the subject and gives insights into the advances and challenges the future may hold - they are selective in their coverage rather than an in-depth review of an area. Therefore, in this perspective paper, we first give a brief overview of the background concepts, methods and tools related to computer-aided product design. To limit the scope, we will restrict this manuscript to only single species products (molecular design) and blended or formulated products (single phase mixture design). We give a brief overview on the state-of-the-art on group contribution-based properties prediction suitable for product design, and, on computer-aided product design techniques. Next, we highlight some of the challenges and issues related to advancing the state-of-the-art and how they could be tackled through the development and use of integrated methods and tools in a hybrid AI framework. In particular, we highlight the need and use of property estimation methods that are predictive in nature to increase the search space, while fitting in within the work-flow of computer-aided product design. Then, we provide a perspective on future developments and uses of integrated hybrid systems to tackle the challenges. Finally, we make some concluding remarks.

2. Background Concepts

Figure 1 gives an overview of the framework for computer-aided molecular (and mixture) design (single species and mixed products) in terms of three main parts: problem definition, molecular design, and screening and selection. The problem definition is related to the type of product that needs to be designed, where the product needs and functions are converted to property constraints. Molecular design needs tools to generate candidate molecules and to estimate their properties. Screening and

selection verifies the properties of the feasible candidates to make the final selection. In molecular design, the building blocks are usually functional groups or descriptors, whereas, for mixture design, the building blocks are an initial set of molecules from which mixtures are generated. Note that the molecular design problem may also be used to create an initial set of molecules in mixture design. Each of these parts are briefly described below.

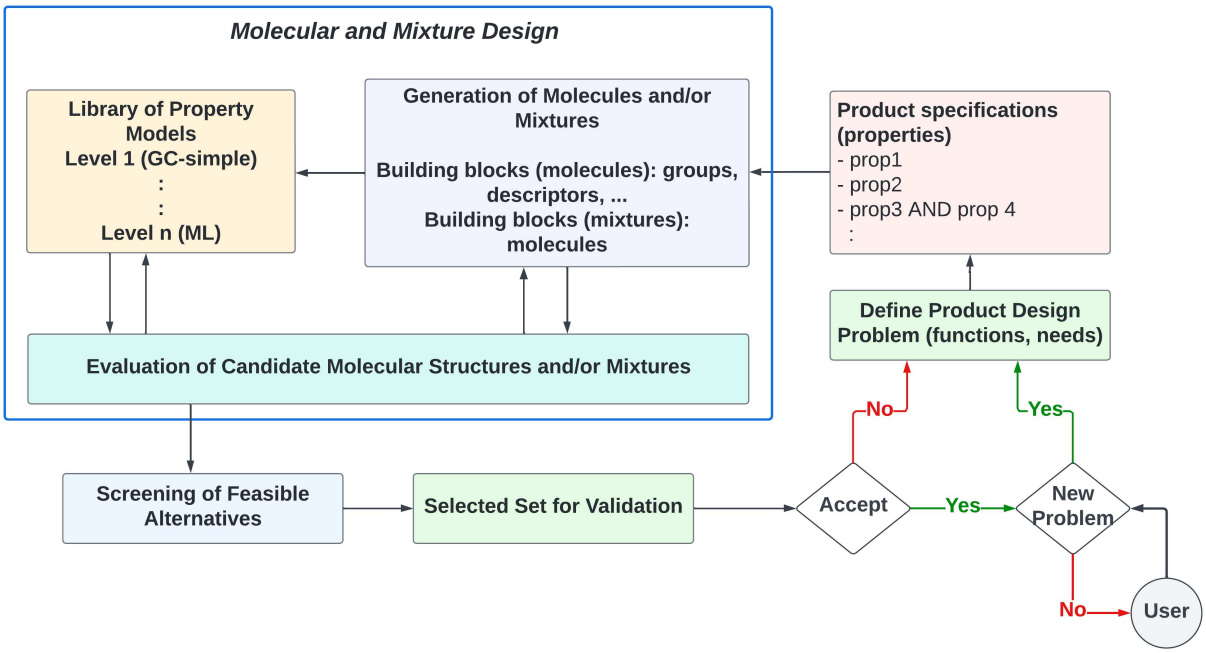


Figure 1: Workflow diagram for computer-aided molecular and mixture design

2.1. Problem definition

Each product type has its unique set of functions and needs. As shown in Figure 1, these product functions are translated to a set of properties and their desired values as the target for design. To help the designer consistently define the design problem, it's useful to have a template based on expert knowledge as shown by Kalakul et al. [11]. Data similar to those given in Tables 1 and 2 help to define the design problem. Knowledge-based intelligent systems could be used to automate the use of templates.

2.2. Molecular structure representation and generation

Molecular structures can be represented in different ways. Ideally, the same representation system should be used for molecular structure generation as well as property prediction. A brief overview of some of the molecular representation systems suitable for group contribution methods is given below.

2.2.1. Use of functional groups

Functional groups are defined as a group of atoms with at least one free bond, with the exception of special (molecular) functional groups that have no free bonds, such as water and methanol. The valency rule [21] is used to make sure that a molecule represented by the functional groups is a feasible chemical compound. These groups are usually characterized into various orders of complexity, where the simple functional groups such as 'CH3-','-OH' are considered first-order groups as shown by Fredenslund [9]. As the first-order groups are unable to distinguish between isomers, conjugation based second-order groups such as 'CH3-CH(CH3)-' are used as additional molecular structural information for GC-based the property prediction models as shown by Gani [7].

Also, to enlarge the application range and prediction accuracy, third-order groups are used for complex molecules [7], such as molecules with fused aromatic rings (for example, biphenyl) and multifunctional molecules (e.g., dipropylene glycol). A useful feature of molecular representation by functional groups is that they are also used as building blocks for the generation of candidate molecules. Similar to functional groups, fragments have also been reported by Li et al. [22].

2.2.2. Use of descriptors

A variation of the functional groups is the representation method employing a combination of atoms, bonds, topological indices, etc., commonly regarded as descriptors. The use of these descriptors can be found in property prediction models classified as QSAR (Quantitative Structural Activity relationships) or QSPR (Quantitative Structrural Property Relationships. For example, Patel and Mannan [23] for flash points of solvents and Abramenko et al. [24] for toxicity of chemicals. Examples of the use of topological indices for property prediction as well as molecular design can be found in Sippl et al. [25] for design of inhibitors and Chemmangattuvalappil and Eden [26] for molecular design for reactive systems including solvent design for ester production. Also, Visco et al. [27] proposed signature molecular descriptors for QSAR based pure compound property prediction.

2.2.3. Properties based descriptors

Another way of representing molecules solely for the purpose of property estimation involves representing molecules as vectors of relevant physicochemical properties that could correlate well with the target property of interest. This often requires expert selection of appropriate descriptors and hence the molecular representation would differ for different target properties. In scenarios where several

properties might be important to be included in the molecular representation, dimensionality reduction techniques such as principal component analysis (PCA) or features selection strategies such as LASSO [28] or genetic algorithms have been shown to be successful by Venkatasubramanian et al. [29]. It is often observed that such physicochemical features (or representations) when combined with other molecular descriptors such as Morgan fingerprints [30], extended-connectivity fingerprint ECFP4 [31], or other custom descriptors [32, 33] could result in better performance. Unlike the functional group representations, the generation of molecular structures with the descriptors is more complex.

2.2.4. Grammar2Vec

Generating dense (continuous-valued) numeric vector representations of molecules, generated using context-preserving neural network frameworks, offer an approach for encoding text-based molecular representations as numeric vectors of any arbitrary size in the latent space. Such methods are inspired by natural language processing methods that utilize the neighboring context of words to learn patterns and 'embed' them in a latent space using continuous-valued vectors. Therefore, generating such embeddings for molecules requires drawing a natural language analogy where the underlying atoms/groups are considered as 'words' in a 'molecular sentence'. Such molecular descriptors are property-agnostic, meaning that each molecule would have a unique representation irrespective of the property being predicted. Approaches such as grammar2vec [34], smiles2vec [35], and mol2vec [36] utilize these ideas to generate molecular descriptors with appropriate regression frameworks. A molecular structure generation routine needs to be developed with the identified set of vectors to help integrate the property modeling and molecular design.

2.2.5. Graph Neural Networks

Molecules could also be represented as graphs or an adjacency matrix (possibly augmented with additional bonds/atoms-specific features) and used in graph neural network framework for property estimation. The graph neural networks involve performing convolution operations on the molecular graph (or adjaceny matrix) to 'embed' the molecule in a latent space which is then used in a neural network framework for property estimation as proposed by Ishida et al. [37]. Owing to the presence of deep neural networks, these methods typically suffer from poor interpretability and black-box nature of the developed models.

2.3. Property prediction for product design

Properties can be classified [7] as primary properties (these are pure compound properties and each chemical has only one value of this property), functional properties (these are also pure compound properties but they may change with temperature and/or pressure), mixture property type-a (these are properties of a homogeneous mixture) and mixture property type-b (these are phase equilibrium properties of a mixture).

The relationship between property prediction and product design can be understood from equation 1

$$\mathbf{p_j} = \mathbf{f}(\mathbf{s_j}, \mathbf{x_j}, \boldsymbol{\beta_i}, \boldsymbol{\Theta_{i,j}}, T, P) \tag{1}$$

where, $\mathbf{p_j}$ is a vector of properties for chemical or system j; $\mathbf{s_j}$ is a vector of molecular structural variables that describe the chemical (for molecular design), or, is a vector of chemical identities (for mixture design) system j; $\mathbf{x_j}$ is a vector of compositions of the chemicals present in the system for which the properties are needed (if all compositions except one are zero, then the corresponding property is a pure compound property), $\boldsymbol{\beta_i}$ is a vector of model parameters for property i, $\boldsymbol{\Theta_{i,j}}$ is a matrix of model parameters related to property i and molecule (or mixture) j; and, T and P are conditions at which the properties are to be estimated. Note that for a specific product design problem, pure compound properties as well as mixture properties may be needed to describe the product functions.

If all the variables on the right-hand side of Eq. 1 are known or specified, then estimation of the unknown property $\mathbf{p_{ij}}$ (property i of chemical or system j) represents the forward property estimation problem. If subsets of variables $\mathbf{s}_j$ and $\mathbf{x}_j$ are unknown while subsets of $\mathbf{p}_j$ are known, it is the reverse problem of property estimation related to molecular design. If subsets of variables $\mathbf{s}_j$ and $\mathbf{x}_j$ are unknown and a subset of $\mathbf{p}_j$ are specified instead, it is the reverse problem of property estimation related to mixture design. More complex reverse problems may also include T and/or P as unknown variables. Note that while for the forward problem, a unique property value is thermodynamically feasible, for the reverse problem, there can be multiple solutions or no solutions, depending on what desired property values are specified. A natural solution approach for the reverse problem is therefore optimization to find the best match among various feasible solutions as shown by Zhang et al. [20].

From the above problem description, it is clear that an integrated solution approach that combines the work-flows and data-flows for property estimation and product design is necessary because the

variables $\mathbf{s}$ and $\mathbf{x}$ are involved in property estimation as well as product design. As stated above, this perspective paper is limited to the concepts, methods and tools for two classes of product design, namely molecular design and mixture design, employing mainly group-contribution based methods for property prediction.

Table 2 gives a list of properties needed for different molecular and mixture design problems. Figure 2 illustrates the different components of GC-model based properties prediction. Note that this is not a comprehensive list. It is given to highlight mainly the relation between properties and product design.

Table 2: Properties needed for different molecular and mixture design problems

Problem	Primary	Functional	Mixture	Phase
Refrigerant	T_c, P_c, ODP, GWP	H_{vap}, C_{p_L}, C_{p_V}		
Refrigerant blend	T_c, P_c, ODP, GWP	H_{vap}, C_{p_L}, C_{p_V}	H_L, H_V, ρ	T_{bub}, T_{dew}
Solvent for extractive distillation	T_b, T_m, $LD50$, δ_P, $LogP$	H_{vap}, C_{p_L}, C_{p_V}		
Solvent for extractive distillation	T_b, T_m, $LD50$	H_{vap}, C_{p_L}, C_{p_V}	H_L, H_V, ρ	T_{bub}, T_{dew}, S_p, $M_s = 1, \gamma_i$
Solvent for liquid-liquid extraction	T_b, T_m, $LD50$	H_{vap}, C_{p_L}, C_{p_V}	H_L, H_V, ρ	T_{bub}, T_{dew}, S_p, $M_s = 2, \gamma_i$
Solvent for crystallization	T_m, $LD50$		H_L, ρ	T_s, S_p, $M_s = 1, \gamma_i$
Solvent for reaction synthesis	T_b, T_m, $LogP$, pKa			T_{bub}, T_{dew}, T_s, S_p, $M_s = 1$ or 2 , γ_i
Lubricant				
Active ingredient	T_b, T_m, $LogP$, pKa, $LC50$, $LD50$, $LogWs$			T_s, S_p, $M_s = 1$ or 2 , γ_i
Diesel blend			RVP, H_V, H_L, ρ	$M_s = 1$
Insect repellent			Evap, ρ, η, solubility	$M_s = 1$

Note: $M_s = 1$ indicates one stable liquid phase; $M_s = 2$ indicates two stable liquid phases
Also, all target properties have not been listed here

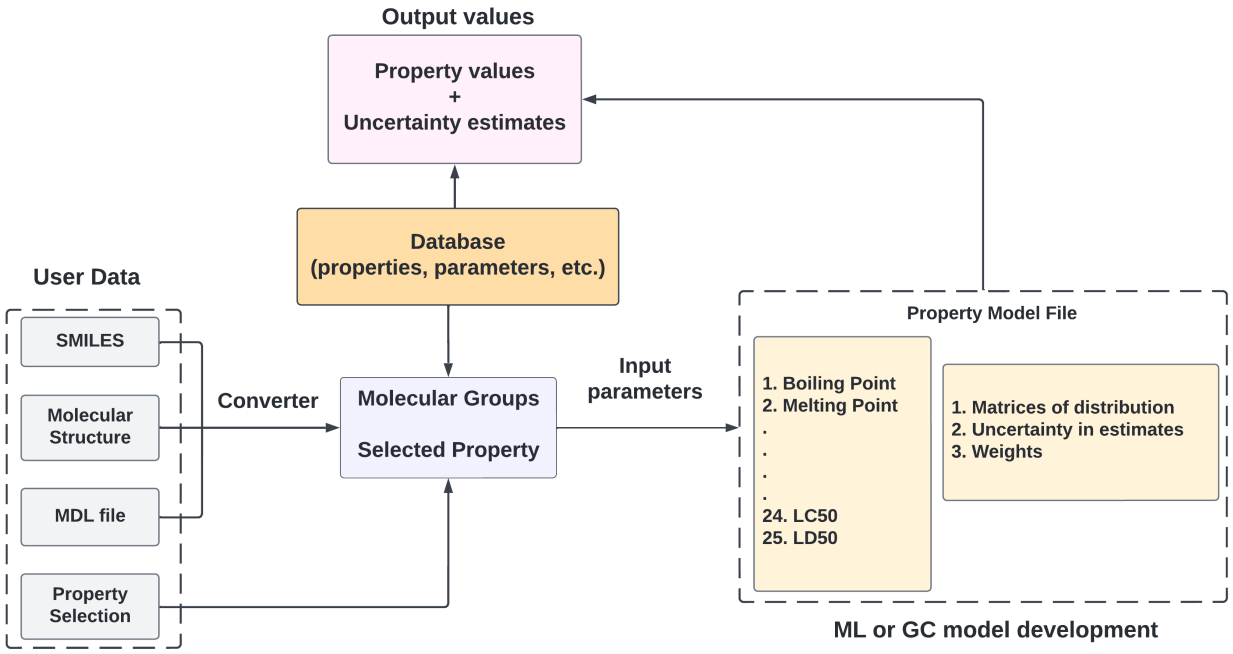


Figure 2: Workflow diagram for GC-based property prediction with starting point as "user data" and end-point as "output values"

2.3.1. GC-based property modelling

For the properties listed in Table 2, GC-based models can be found for most primary properties involving organic chemicals [38]. Although for some functional properties, GC-based models have been developed [39], in most cases, the primary properties are used as input in equations of state to obtain the functional properties. Note that at normal pressure of 1 atm and/or at temperature of 273 K, the corresponding functional property could be regarded as a primary property. For the mixture properties of type-a, if the mixture property has a linear dependence with respect to composition, the primary or functional pure compound properties are used together with linear mixing rules with respect to composition. For non-linear dependence, non-linear mixing rules are often used. For phase equilibrium related properties (type-b), depending on the computational procedure used for phase equilibrium, different sets of primary, functional and mixture properties are used. Here, use of GC-based models for properties such as activity coefficients of molecules present in the liquid phase of a mixture are well-known [9],[10]. Phase equilibrium can also be predicted through GC-based equations of state, such as SAFT [40] and PC-SAFT [41]. Note, however, properties such as, T_{bub}, T_{dew}, M_s are conditional properties because they also need to satisfy the condition of equilibrium. GC-based property models are briefly described in the sub-sections below.

Primary Properties. In GC-based property modeling, we are highlighting those that use the functional groups. For estimation of pure compound properties (primary properties) of chemicals, this information on molecular representation of chemicals is needed together with the property contributions of the groups representing the molecular structure. A simple example for the well-known Marerro-Gani method [42] and its updated versions [38, 43] is highlighted through Eq. 2.

$$f(p_i) = \sum_{nf=1}^{N} F\mathbf{c}_{nf}^{F}\mathbf{n}_{nf}^{F} + \sum_{ns=1}^{N} S\mathbf{c}_{ns}^{S}\mathbf{n}_{ns}^{S} + \sum_{nt=1}^{N} T\mathbf{c}_{nt}^{T}\mathbf{n}_{nt}^{T} \tag{2}$$

In Eq. 2, the summation of the group contributions are given on the right-hand side (RHS) of the property function, where, the superscripts F, S, and T indicate the first-, second- and third-order contributions, respectively, and subscripts nf, ns and nt indicate first-, second- and third-order group identities, respectively; $\mathbf{n}$ is the vector of the number of occurrences of each group and $\mathbf{c}$ is the vector of regressed contributions, which are needed to estimate the property p_i. See also, [44], [45] for other examples of GC-based pure compound primary property modeling.

Functional properties. The pure compound primary property estimation methods can be extended to temperature dependent functional properties (heat of vaporization, specific heats, density, etc.) by making the property group parameter (variable **c** in Eq. 2) also temperature dependent [39, 46, 47]. Another option for estimation of temperature dependent pure compound properties is to use equations of state, which in this case, only need primary properties (such as the critical properties) as input. A simple model to obtain the vapor pressure as a function of temperature is shown through Eq. 3 for the van der Waals cubic equation of state from which the well-known SRK equation of state [48] can be derived.

$$P = RT/(V - b) - a/v^2 \tag{3}$$

where a, b are model parameters, which could be expressed in functional form, as

$$a = f(\mathbf{x}, T, P, T_c, P_c, k_{ij}) \tag{4}$$

$$b = f(\mathbf{x}, T, P, T_c, P_c) \tag{5}$$

As the $k_{i,j}$ in Eq. 4 relate to the binary interactions between molecules, this form of the equation of state are not GC-based, although, GC-based equations of state, such as SAFT [40], PC-SAFT [41] can also be used for pure compound properties; **x** in Equations 4 and 5 are the compositions of the chemicals in the mixture, which is set=1 for the pure compound.

Mixture properties (type-a). For mixture properties of type-a, if the mixture can be assumed to be ideal, then ideal mixing rule is used to compute the mixture property from the known pure compound properties. If the ideal mixing rule cannot be assumed, then specific mixture property models need to be developed. For example, Tochigi et al. [49] proposes ASOG based method for kinematic viscosity and thermal conductivity of liquid mixtures; Liu et al. [50] proposes a GC-based method for surface tension of ionic liquid-water mixtures; and Chen et al. [51] proposes a GC-based model for solubility of selected organic chemicals.

Phase equilibrium properties (type-b). Phase equilibrium properties are typically related to VLE (vapor-liquid equilibrium), LLE (liquid-liquid equilibrium), SLE (solid-liquid equilibrium) plus other combinations. Examples of the phase equilibrium condition that needs to be satisfied for a mixture in

equilibrium (for specific phase equilibrium problems), are given by Equations 6, 7 and 8 for VLE, LLE and SLE, respectively,

$$x_i \gamma_i^I P_i^{sat} = y_i P \tag{6}$$

$$x_i \gamma_i^I = y_i \gamma_i^{II} \tag{7}$$

$$s_i \gamma_i^S = x_i \gamma_i^L exp[\Delta H_i^F / (R T_{mi})((T - T_{mi})/T)] \tag{8}$$

A typical VLE phase equilibrium computation problem with (Eq. 6), involves the iterative determination of, for example, the saturation temperature (T_{bub}) and the corresponding vapor composition (y) in equilibrium with a specified liquid of composition ($\mathbf{x}$) and pressure P. In the above equations, the variables in red indicate properties for which data or models are needed to supply their values. That is, γ_i^I, a mixture property that needs to be predicted through a liquid phase activity coefficient model, while, P_i^{sat} , is a functional pure compound property dependent on temperature that can be estimated in various ways (correlation, equation of state or GC-based), ($\mathbf{s}$) is composition of compound i in solid phase. Therefore, for phase equilibrium computations, primary properties, functional properties as well as phase equilibrium properties are needed. Specific to phase equilibrium properties (for example, activity coefficients, γ_i, in the liquid phase for each chemical), GC-based models can be used, for example, UNIFAC [9], GC-based COSMO-models proposed by Wang et al. [52]. GC-based equations of state can also be used to predict phase equilibrium. The following review articles on PC-SAFT [53], SAFT [54] and cubic EOS models [55] provide useful information.

2.3.2. ML-based property modeling

The significant advances in computational power combined with the availability of data and powerful machine learning frameworks have greatly improved the development of property prediction models as argued by Venkatasubramanian [16]. Several property estimation models have been reported that have either surpassed the accuracy of traditional QSPR and GC-based methods, or have been combined with them to improve their performance under certain scenarios where they were lacking. Such ML-based methods are heavily reliant on the availability of data for learning model parameters by discovering correlations between various molecular descriptors and the target property, thus making the task of representation learning crucial for the success of such models as shown by Mann and Venkatasubramanian [19]. In addition, combining these models with domain knowledge and expert

insights results in the development of hybrid machine learning models that often have better predictive performance, are more explainable, and correlate well with the underlying chemistry as demonstrated by Mann et al. [34] and Alshehri et al. [38]. Other examples of ML-based property prediction models are, prediction of density of deep eutectic solvents proposed by Roosta et al. [56] and prediction of viscosity of bio-chemicals proposed by Hernandez et al. [57].

Typical ML framework for property estimation. The property estimation task is usually formulated as a regression problem where the objective is to build a machine learning-based regression model between the regressors and the target variable as,

$$\hat{p}_{j,i} = \mathbf{f}(\mathbf{s_j}, \boldsymbol{\beta}_i) + e_i \tag{9}$$

where (for the j^{th} chemical) the target variable $\hat{p}_{j,i}$ is the predicted property of interest i; $\mathbf{s_j}$ is the vector representation of the chemical's descriptors and is of dimension $m \times 1$; $\boldsymbol{\beta}_i$ is a vector of regression coefficients (with appropriate dimensions depending on the ML model-form) that is estimated; and e_i is the noise in the prediction that could be attributed to the measurement errors in the training data and/or the modeling inaccuracies.

The two important aspects for estimating the target property $p_{j,i}$ accurately, therefore, are – choosing an appropriate functional transformation $\mathbf{f}(.)$ and using rich molecular descriptors $\mathbf{s_j}$ that correlate well with the property of interest. Note that functional groups are a special type of descriptors. Several works have been reported that either focus on finding an appropriate set of descriptors for a molecule as well as using advanced ML architectures that provided a complex functional form for mapping the descriptors to the target property. While predictive accuracy is central to ML models, the ability to explain and interpret them is equally important but challenging, thus making it an important area of research.

Model architectures for ML. Model architectures play an important role in mapping the various non-linear and complex correlations between molecular descriptors and a target property of interest. This functional mapping could be represented through $\mathbf{f}(.)$ in equation 9, which could be either parametric or non-parametric depending on the underlying ML model architecture. Several ML model architectures that fit in the regression framework could then be used for estimating the target property such as – least squares regression [58], support vector regression [59], Gaussian process regression

[60], ensemble methods [61], or deep neural networks [62].

Since most ML approaches have the inherent drawback of poor explainability, a class of hybrid machine learning frameworks that combine domain knowledge with data-driven methods could play a crucial role towards improving interpretability of ML models. This could be done at several levels such as computing the individual feature importance and contribution towards predictions, correlating observed data-driven trends with underlying chemistry knowledge, performing detailed error analysis for various molecular types and classes, and combining traditional QSPR methods such as GC with machine learning frameworks as shown by Mann et al. [34] and Alshehri et al. [38].

2.4. Methods for Chemical Product Design

2.4.1. GC-based Molecular Design

As pointed out through Eq. 1 and Figure 1, in product design problems, the product functions in the form of properties are specified and the identity of the chemical or the mixture of chemicals matching the properties need to be determined. Much advances have been made in developing efficient methods to solve these problems, for example, as generate and test (Kalakul et al. [11], Sippl et al. [25]), optimization-based techniques (Jonuzaj et al. [14], Liu et al. [15]). Note that for molecular design problems, pure compound, functional properties as well as mixture properties may be selected as target properties. However, the design variables are the building blocks such as functional groups, fragments, or descriptors. A review of computer-aided methods for chemical product design is given by Austin [63].

2.4.2. GC-based Mixture Design

When a large number of target properties need to be matched, it is unlikely that a single molecule would match all the target properties. In this case, a blend or mixture is formulated as a mixture design problem. Therefore, for these problems, the objective is to find mixtures of molecules that satisfy a set of target mixture properties. There are three important decision variables, namely, the number of molecules present in the mixture, their identity and their composition. In addition to the property constraints, the mixture must be a single phase, usually a liquid. A decomposition-based solution strategy where mixtures of fixed number of molecules are generated and tested, first for their miscibility and then for mixture compositions that match the target properties, have been reported for liquid formulated products by Conte et al. [64] and blended fuels by Yunus et al. [65]. Also,

a mathematical programming based design approach to simultaneously obtain values of the decision variables that match the property targets for simpler mixture design problems have been reported by Liu et al. [15].

2.4.3. AI applications in product and molecular-mixture design

Artificial intelligence in our context refers to the broader collection of methods such as pattern matching, search algorithms, knowledge representation, data-driven models-based simulation, and agents learning from their environment (reinforcement learning) that could be used either in conjunction with ML or independently for product design. We treat AI different from traditional ML by limiting the definition of ML only as something used for mapping the inputs molecular descriptors to the target property of interest.

The following sections give a brief overview of the various applications of AI such as high throughput screening, computer aided chemical reaction prediction and synthesis, design and optimization of molecules, modeling interactions between several entities, and so on, that are important for the design of molecular-mixtures.

High throughput screening. High throughput screening is essential for quickly evaluating several candidates with a desired property that would help in screening out the undesired molecules. Such methods offer a virtual experimentation bench that sidesteps the need to perform exhaustive wet-experiments saving a lot of effort, time, and money. Typically, at the heart of such high throughput experimentation tools lies a computational-model that is highly efficient in computation time with fairly high accuracy and performance, as shown by Mayr and Bojanic [66] and Phillips et al. [67].

Chemical reaction prediction and retroysnthesis. There has been a recent surge in the development of accurate reaction modeling strategies that could be used for solving the forward reaction prediction problem (given reactants, predict reaction products) and the restrosynthesis problem (given target product, identify starting reactants). Several different class of methods have been utilized for both of these problems, and they could be categorized into three major classes as identified by Venkatasubramanian and Mann [19] – symbolic AI, purely data-driven AI, and hybrid AI. Each class of method has their own pros and cons but the most promising is the latter class of hybrid AI methods that combines domain knowledge with data-driven methods [34, 68–71].

Multi-step chemical synthesis is often very important since they offer information not only on the

starting reactants but also on the various intermediate steps and transformations that are required to synthesize the target molecule. Similarly, predicting the reaction yield [72], selectivity [73], and the reaction class [74, 75] is equally important and has practical significance for computer aided reaction synthesis.

Molecule design and optimization. Molecule design involves identifying or optimizing the structure of molecules that results in a desired behavior, often in terms of certain physical, chemical, or thermodynamic properties of interest. For practical significance, several properties need to be optimized simultaneously. Many deep learning-based approaches have been proposed in this area that primarily involve learning latent space representation of molecules and then optimizing the properties of interest by sampling (often probabilistically) from that latent space. It is often required to learn a continuous, smooth latent-space representation of molecules using autoencoder frameworks since it has been shown in [76] that a smooth latent space results in better identification of molecules with desired properties.

Agents learning from environment interaction. An important aspect of AI is the problem formulation in the form of agents and environment where the 'agents' interact with an 'environment' and try to maximize their 'reward' (and consequently learn) from sequential 'interactions' with the environment, thus eventually finding the optimal goal state. The definition of the different components – namely, agent, action, reward, and environment – depends on the problem formulation and desired goal. Various applications of reinforcement learning combined with deep learning has been shown in [77–79] to be useful in optimizing chemical reactions, searching for synthesizable novel molecules, and other chemicals-based product design applications.

3. Challenges and Issues

In this section challenges and issues related to representation of the molecular structure, regression to obtain the final form of the model, and integration of different methods and tools within the product design work-flow are highlighted, as they influence the scope and significance of property prediction models as well product design methods and tools. We first discuss some of the related issues through an illustrative and typical single species (molecular) product design problem and then highlight the different associated challenges and issues.

3.1. Single molecule product design problem

The single species (molecular) product design problem is defined as, find a replacement solvent having the following properties:

- Set 1: $90 < M_w < 120$; $350K < T_b < 425K$; $T_m < 250K$; $H_{sp} < 20$ (GC-simple models available)
- Set 2: $LogP > 1.5$; $LogLC50 < 4$ (higher-order GC-models available)
- Set 3: Miscibility with water (GC-based liquid phase miscibility calculation option available)
- Set 4: Check for environmental, health and physical hazards (data available in a separate database)

3.1.1. Problem statement

As the specific target properties with their bounds are given, the translation from needs to properties is not needed here. Also, as an objective function is not given, instead of searching for the optimal replacement solvent, a set of feasible candidates need to be determined. The solution methods that are more appropriate for this problem are database search [11], enumeration-based generation-test [12] and genetic algorithm-based [29] solution approaches. Note that after a feasible set of molecules have been identified, they can be ordered according to any specified selection criteria to identify the most suitable (single species) product. Also, if the application process for the designed molecule is included (as in the case of a solvent-based separation process), then the simultaneous design of the separation process and the selection of the solvent could be formulated as a mathematical optimization problem, which can be solved in different ways [13–15]. Note that if instead of properties and their target values, the functions or needs such as a solvent must be liquid, dissolves selectively a solute, is miscible (or not miscible) with water, and does not have hazardous properties, a knowledge-based system would need to translate these functions to the property sets that will define the product design problem.

3.1.2. Problem solution

The solution of the problem is highlighted through the solution options available in the ProCAPD software [11] as it is available to the authors. The first step is to check the availability of data and/or property models for the target properties in ProCAPD. The following property models and data are available in ProCAPD.

- Set 1: GC-simple models available within the CAMD work-flow and also database engine includes these target properties
- Set 2: Higher-order GC-models available within the CAMD work-flow in a hybrid option; also, a different database includes these properties
- Set 3: GC-based liquid phase miscibility calculation is included in the CAMD work-flow and also qualitative miscibility with water data available in a database search engine
- Set 4: Data available in a separate database that is not part of the CAMD database search engine

Based on the above, different solution schemes could be employed, even though a single solution approach matching all the target properties is not yet available.

Database search. From a practical point of view, this solution option is useful only for search based on single value pure compound properties. Additional levels of search are needed for functional properties and mixture properties. A first search is made for the four properties in set-1, which gives 124 compounds matching the constraints. Figure S1 (see supplementary material) gives details of the search problem specification and Table S1 (see supplementary material) gives the list of feasible candidates. Adding now the search related to miscibility with water (qualitative data available in the same database), compounds that are soluble, or slightly soluble or insoluble in water in addition to the set-1 target properties are identified. This gives 8, 24, and 66 candidate molecules, respectively, for soluble, slightly soluble and insoluble in water, out of the original 124. This is also highlighted in Table S1. From the 124 compounds, the following four compounds representing different molecular types hydrocarbon (n-heptane: 000142-82-5), aldehyde (1-hexanal: 000066-25-1), ketone (diisopropyl ketone: 000565-80-0) and ether (n-butyl-ethyl-ether: 000628-81-9), which are all well-known solvents, are selected for hazards analysis (available in ProCAPD in a separate section of the database library, which is not included within the CAMD work-flow). Not surprisingly only n-heptane is listed as a danger in terms of environmental hazards; all are listed as physical hazards because of flammability, with 1-hexanal being the safer; all have health hazards and again 1-hexanal has warnings as opposed to being listed as danger.

Issues: What this analysis shows is that selection of a chemical product needs to be thoroughly investigated and requires searches in multiple databases, none of whom are currently complete. Another limitation of database search is the issue of mixture properties. For the problem defined above, typically

this solvent would be used for extractive distillation and the specific choice of the solvent would also need to check for solubility of the solute in the solvent as well as creation of a two-phase vapor-liquid equilibrium system. These criteria would need to be checked through phase equilibrium computations, process simulations, as well as experiments.

CAMD based on generate-test method. The same problem solved through database search, can also be solved through the CAMD-option, where the GC-model (simple) based property models are available for different types of properties. This CAMD work-flow is similar to the one highlighted in Figure 1. Unlike the database search, here for every generated candidate molecule, if the GC-model parameters are available, the target properties can be calculated. Therefore, the search space is potentially larger than in database search. Unlike database search, LogP from set-2 and water solubility from set-3 are also included. Set-4 analysis, however, needs to be done separately as only qualitative data for hazardous properties of chemicals are available in a separate database. For a minimum of 2 groups and a maximum of 8 groups (as this kind of solvents are usually not large molecules) and only acyclic hydrocarbons, alcohols, acids, ketones, aldehydes, esters, and ethers (that is, only compounds with carbon, hydrogen and oxygen atoms are considered), the CAMD-option generated 3498 molecules, out of which 82 were found feasible and it took 0.88 seconds on a computer with Intel(R) Core(TM) m7-6Y75 CPU @ 1.20GHz 1.51 GHz processor. Note that this selection of building blocks prevents potential problems with hazardous chemicals by avoiding aromatic compounds or compounds with halogens, for example. In Figure S2 (see supplementary material), the problem definition details are given; and in Tables S2 and S3 (see supplementary material), the list of feasible molecules and the solution statistics are given. It can be noted that a majority of the generated molecules were screened out because some of their GC-model parameters were missing. The solution statistics highlight how many molecular structures are generated and how many were screened out because of different reasons, such as unavailable model parameters or out-of-bound properties. Also, from the list of molecules in Table S2, it can be noted that some generated molecules can be found in the database, while others are not. It is possible that among those that are not found in the database, they may exist in some other database, or are newly generated molecules that would need to be synthesized first before their actual use could be considered. However, the 4 molecules identified through database search are also found in this problem solution and their hazards analysis would again need to be performed separately.

Issues: Although the search space is enlarged compared to database search, there are also limita-

tions. Is the number of generated molecules limited by the the use of combination rules of groups to form molecules, which are employed to avoid a potential combinatorial explosion? Are the inaccuracies of the GC-simple models acceptable? How to predict the properties of the generated candidates if the GC-simple model parameters are not available?

CAMD with hybrid model option. As pointed out by Hukkerikar et al. [43] accuracy of the GC-simple (based on 1st-order groups) is qualitatively acceptable, but a more accurate prediction is obtained through the addition of 2nd- and 3rd-order group contributions. The use of higher-order groups also increases the search space for the design problem as reported by Liu et al. [15]. In this example, use of the higher-order model is used in an outer-loop to validate and screen the results from the GC-simple model in an inner-loop. The number of molecules generated are still the same as the higher-order groups are only used for property prediction. However, the number of feasible molecules has now decreased from 82 to 56 and the computing time has increased from 0.88 seconds to 63.05 seconds. Therefore, increase of search space and accuracy is paid through additional computational time. In Table S2, the 56 screened molecules are highlighted in Italics and in Table S3, the problem solution statistics are given. The target properties for the four selected molecules predicted with GC-simple and GC-ML models (a computationally more expensive model but giving more accurate predictions) are compared with available measured data in Table S4 (see supplementary material). Note that using the hybrid scheme, the additional property in set-2 is included. However, set-4 properties still needs an additional separate analysis of the selected candidates.

Issues: How to incorporate the computationally expensive but more accurate property models within the CAMD work-flow? Is the overall computing time to obtain a more reliable solution worth the additional computational times? Should the decomposition of target properties into different subsets be based on model availability, computational time and/or application range?

Comparison of GC-model and GC-based ML-model. The ProCAPD software [11] also has the option for stand-alone pure compound property estimation tool called Pure. It calculates 46 GC-based primary properties, 11 secondary properties where the primary properties are used as input and 9 temperature dependent functional properties. As reported in [38] the accuracy of prediction for 25 (out of the 46 available in the Pure software), by the GC-ML model is much higher than the GC-simple model (close to 90% of the molecules has less than 1% error with the GC-based ML-model as opposed to around 50%, on average with the GC-based simple model). In Table S5 (see Supplementary

material), the prediction comparison between the GC-simple and GC-ML models are given for the two selected molecules. Note that a detailed list of training data and their correlation error are given as supplementary material by Alshehri et al. [38]. The work-flow for prediction with the GC-based simple and GC-based ML models are illustrated in Figure 2. In Table 3, the storage required, the computation time required for the 25 properties are listed, based on a computer with with Intel(R) Core(TM) m7-6Y75 CPU @ 1.20GHz 1.51 GHz processor.

Table 3: Comparison of GC- and ML-based property prediction models

Comparison metrics	GC-based simple model	GC-based ML-model
Computation time (s)	<1 *	43
Storage (MB)	101 *	3434

Note: * the storage and computing time given for GC-based simple includes all 65 properties in the Pure software tool

Issues: Based on the data in Table 3, clearly, the best way to incorporate ML-based models within the product design work-flow needs to be carefully developed to get the maximum benefit. The accuracy is not an issue here, but the storage and computational time are. However, as the primary property of a molecule needs to be estimated only once, the computational time may not be an issue if the calculated data is added to the database as a pseudo-measured data for future use.

3.2. Other variations of the product design problem

A few variations of the above single species molecular design problem are highlighted briefly.

3.2.1. Refrigerant design - molecular and mixture design

This design problem includes only pure compound target properties such as critical temperature, critical pressure, and functional properties such as boiling point (as a function of pressure), vapor pressure (as a function of temperature), heat of vaporization (function of temperature and pressure), specific enthalpies (function of temperature) and hazardous properties. While GC-based property models have been used (Achenie et al. [21]; Austin et al. [13]), it has been shown recently that the functional groups used in GC-based property models for solvents (for example) are not appropriate for the small molecules representing the refrigerants as shown by Kuprasertwong et al. [80].

Issues: How to define the appropriate descriptors and based on them, develop the associated target property models? Also, as a single chemical is unlikely to satisfy all the target properties, should design of refrigerant blends be considered? How to generate the basic set of chemicals from which the

mixtures could be designed? Should they be binary mixtures or multi-component mixtures? For this class of molecules, the GC-based computationally expensive equations of state could be more suitable. How to incorporate them within the molecular and mixture design work-flow?

3.2.2. Liquid fuel or formulation (mixture) design

These design problems include only mixture target properties but to identify the mixture chemicals and predict the mixture properties, pure compound properties are also needed. As pointed out by Kalakul et al. [11], Conte et al. [64] and Yunus et al. [65], different products need different sets of target properties and therefore, property models. Also, in some cases, as in fuels (for example, gasoline, diesel or jet-fuel), different types of chemicals with different molecular sizes are found in these products.

Issues: To enlarge the application range of the computer-aided methods and tools, potentially a large library of property models and chemicals are needed to cover a wide spectrum of products. Should an apriori set of molecules be developed and stored in a database as potential additives for the mixture design problem?

3.3. Molecular structure representation and generation

As briefly mentioned in Section 2.2 and the above examples, molecular representations play an important role for accurate modeling of correlations between descriptors and the target property. Since they can also be used as building blocks for generating molecular structures, the issue of representation of molecular structures is very important and needs to be tackled to enhance the scope and significance of any product design method. While, in principle, molecular structures for a very large number of chemicals can be generated, the properties for only a fraction of them can be estimated and measured data are available for even a smaller fraction of chemicals as shown by Syeda et al. [81]. Some issues are highlighted as bullet points below:

- How to overcome the well-known limitations of the functional groups? For example, they are not suitable for small molecules, such as refrigerants; and for large complex molecules such active pharmaceutical ingredients.
- Can representation methods like grammar2vec be used to simultaneously identify promising molecular structures and predict their properties?

- Can a system to convert one representation system to another be developed so that the properties for a larger set of molecules can be predicted?

3.4. Property model development

The development of GC-based property estimation models is usually formulated as a regression problem, where, the primary objective is to estimate the model parameters if the model equations are fixed, or, identify the model as well as its parameters if the model is not fixed. This is true irrespective of the underlying modeling approach, which could be either of group-contribution-based, ML-based, or hybrid ML-based approach that combines the two. As described in [34, 38], the model parameters could be obtained for various GC-based property estimation methods, namely, GC-simple and GC-ML as described in the following sections.

3.4.1. Regression of GC-model parameters

Typically, the GC-based formulation involves estimating the model parameters (in Equation 2) using different methods (least squares, maximum liklihood principle, etc.), which involves, for the least squares method, minimizing the difference between the squared error between the true and predicted property values as shown in Equation 10. Such estimation is usually performed at three different levels: y considering only the first-order term; the first and second-order terms; and/or all three terms simultaneously.

$$S(\mathbf{c^F}, \mathbf{c^S}, \mathbf{c^T}) = min_{w_k} \sum_{j=1}^{N} (y_{Mj} - y_{pred,j}) \tag{10}$$

where w_k is a tune-able parameter that controls simultaneous regression of the model parameters. The regression of model parameters using the above approach could either be done simultaneously by regressing all the parameters at once, or in a step-wise manner (SWR regression) where the parameter regression is done sequentially for the first order groups, followed by the second order groups, and finally the third order groups as described by Hukkerikar et al. [43, 82].

On the other hand, for ML-based methods, regression could either be performed for the parameters if the underlying model is parametric (such as artificial neural networks, or support vector regression with linear kernel), or estimated indirectly for non-parametric models such as support vector regression with radial basis function kernels or Gaussian process regression. For a parametric model such as ANN,

the property is estimated as

$$\hat{y} = f_2(\mathbf{W}_2)[nf_1(\mathbf{W}_1) + \mathbf{b}_1] + \mathbf{b}_2 \tag{11}$$

where $\mathbf{W}_1(NP \times NG)$ and $\mathbf{W}_2(1 \times NP)$ are weight matrices, for the hidden and outer layers while $\mathbf{b}_1(NP \times 1)$ and $\mathbf{b}_2(1 \times 1)$ are bias vectors, and NP and NG are the number of neurons and the number of groups, respectively. The parameter matrices are estimated by using the back-propagation method with squared error as a loss function that is minimized over iterations during the model training stage. Similarly, for a non-parametric model such as kernel-support vector regression, the estimation is formulated as an optimization problem with Lagrange and support vectors as:

$$\min_{\lambda} \frac{1}{2} \sum_{j=1}^{N} \sum_{k=1}^{N} (\lambda_j - \lambda_j^*)(\lambda_k - \lambda_k^*) K(x_j, x_k) + \epsilon(\lambda_j + \lambda_j^*) - \sum_{j=1}^{N} y_j(\lambda_j - \lambda_j^*) \tag{12}$$

subject to,

$$\sum_{i=1}^{N} (\lambda_j - \lambda_j^*) = 0, \quad 0 \leq \lambda_j \leq C, \quad 0 \leq \lambda_j^* \leq C$$

where $K(.)$ is the radial basis function (RBF) kernel used for transforming the data to a high dimensional space, given by,

$$K(x_j, x_k) = exp(-\gamma ||x_j - x_k||^2) \tag{13}$$

where the parameter γ controls the width of the kernel. The regressed (predicted) values, $\hat{y}_i$, for an input $\mathbf{x_i}$, is given by

$$\hat{y}_i = \sum_{j=1}^{N} (\lambda_j - \lambda_j^*)\ exp(-\gamma ||\mathbf{x}_j - \mathbf{x}_i||^2) + b \tag{14}$$

Therefore, Equations 12 and 14 characterize a trained non-parametric regression model, and the regressed values for thermodynamic properties of a give molecule could be estimated using Equation 14 for a given molecule represented as $\mathbf{x_i}$. Additional details on regression of model parameters can be found in [34, 38, 83]. Some issues are highlighted as bullet points below:

- How to optimally combine the GC-simple models, which are simple and fast but less accurate with GC-ML models, which are like a black-box model but are very accurate?

- Can the ML-based more accurate property prediction models and other computer-intensive models be used to generate pseudo-measured data for use in regression of the model parameters and thereby develop new models with larger application range?

- What about the extrapolation and interpolation features of the regressed models? How to ensure increased predictive capabilities?

3.5. Integration of databases, property models and design method work-flow

The scope and significance of the molecular and mixture design problems depend on the available data, the application range of the property models together with how the design problems are formulated and solved. As the data in Tables 1 and 2 indicate, the product functions need to be translated to properties with targets or bounds on their values for a typical molecular or mixture design problem. Also, as pointed out through the examples in section 3.1, multiple databases need to be used; different sets of property models are needed for different product design problems; different property models come with their parameter sets, storage requirements, and computational times. From an integration point of view, the following issues are highlighted (not in any order of priority):

- What should be the role of AI in chemical product design problems? Which steps of the product design work-flow would benefit most through integration of AI techniques?

- What should be the role of ML-based property modeling? What would be the best way to integrate ML-based models within the product design work-flow? Should they be used as black-box models in specific applications?

- How to increase the portfolio of product design software tools that can solve much larger numbers of problems, reliably and efficiently?

3.6. Challenges

Based on the above, the following challenges are identified and needs to be tackled with respect to development of product design methods and tools:

- Databases - their extension and use in property models as well as in the product design work-flow

- Property models - Understanding the value of ML-models with respect to property modelling

- Analysis (data, design problem and design solution) - inclusion of AI techniques in more versatile problem definition through translation of needs to property constraints; selection of the best candidates; efficient search and retrieval of data
- Integration of computer-aided methods and tools - define a flexible, reliable and versatile software architecture that meets current needs and allows extensions for future needs.

A few additional specific challenges related to property prediction and their use in chemicals-based product design are highlighted below (not in any order of priority).

- The simple GC-based methods have reached their limits of accuracy (they are simple but further accuracy is probably not possible). How to keep the simplicity and yet improve the accuracy?
- Can ML help with accurate interpolation for generating similar structures and predicting their properties? Can ML help with more accurate extrapolations?
- Can AI techniques be used to capture the important binary interaction parameters for phase equilibrium models such as UNIFAC?
- Can AI techniques help to identify and tackle proximity effects that limit the use of GC-based models?
- With the wide range of data that need to be stored, retrieved and used, is there a need for more advanced knowledge representation systems?
- What is the most efficient way to represent molecules and also use them for property prediction? Do we need different levels of representation systems depending on the product design problem and accuracy?
- Can AI identify inconsistent data in collected data-sets and help to select the data-sets for training and testing?

4. Perspective on Challenges and Opportunities

A discussion on how the challenges listed in section 3 could be addressed is presented here.

4.1. Databases

As shown in Fig 2, databases are used for providing property model parameters during property estimation as well as providing property values during validation of the final selected molecules. As data typically would come from different sources, an intelligent knowledge representation system could check the consistency of the data as well as provide efficient maintenance, search and retrieval of data. As availability of measured data is limited and also measurement of new data is time consuming, expensive and may even be infeasible due to safety concerns, use of ML-based models could be considered to expand the contents of the databases.

4.2. Property models

A central part of computer-aided product design is the estimation of the target properties for a generated molecule or mixture. For the synthesis stage, simple models, such as GC-simple models that are computationally inexpensive could play an important role in product design if they give qualitatively accurate predictions, while, quantitatively accurate but computationally expensive models, such as GC-ML models could be useful for the validation stage. As availability of the model parameters define the application range of the models, they therefore, also affect the range of products that can be designed. Note that as GC-based property models need molecular representation or mixture compositions as input data, inclusion of different options to provide this data would make the property prediction tool more versatile and flexible.

4.2.1. Screening and validation of generated candidates

To identify the molecule or mixture that match a set of target properties, it is essential to evaluate the target properties of generated candidates. Development of a computer-aided tool capable of predicting the target properties (through a versatile property model library) and/or retrieving the needed data (from a large and comprehensive database) for a large percentage of generated candidates could enhance the application range and reliability of these tools for chemical product design.

4.2.2. Understanding the value of ML-based models

Since the GC-based machine learning models are primarily based on modeling correlations between molecular descriptors and the target property of interest, it is imperative that uniformity in the data would lead to better property estimation results. For instance, training a model on data containing molecules of a given type (say hydrocarbons) would perform better than models trained on a diverse

set of molecules with different chemistry (hydrocarbons, alcohols, amides, aromatics, halogens) put together into a single data-set. This is not trivial since segregating the data-set into different molecular categories results in reduction in number of samples which is crucial for the performance of complex ML models such as graph neural networks and deep learning frameworks. However, if the underlying model is simple enough, the trade-off favors better performance due to uniformity as opposed to performance degradation due to smaller data-sets as shown by Mann et al. [34]. Note, however, many molecules of interest could be multi-functional, adding to the complexity of the modelling problem.

Table 4 shows the performance of three models – first, a hybrid ML model that combines GC with ML using Gaussian process regression (GPR) framework that reported better performance than individual approaches [38]; second, a support vector regression (SVR)-based ML model that utilized chemistry-rich dense vector representations of molecules obtained using the grammar2vec framework [34]; and third, the performance of the SVR-grammar2vec method when trained only on hydrocarbons data-set. It is clear that the uniformity of the data-set as a result of considering only hydrocarbons translates into significant performance improvement – almost all the predictions for T_b and T_c are within a 5%-error threshold and the maximum percentage error in prediction error was reduced by nearly a factor of four. However, the application range is also significantly reduced.

Table 4: The numbers indicate the percentage of molecules below the given percentage relative error threshold except the last column that provides values for the maximum percentage relative error observed in the predictions.

		1% error	5% error	10% error	15% error	Max. error
GC-ML approach [38]	T_b	**79.1**	**87.8**	92.7	-	>50%
	T_c	**84.4**	**91.4**	94.2	-	>40%
SVR-grammar2vec [34]	T_b	42.0	82.3	**94.9**	**98.8**	**30.2%**
	T_c	51.8	82.1	**94.9**	**98.9**	**25.3%**
	$T_{b,hydrocarbons}$	65.8	**98.9**	**100**	**100**	**8.2%**
	$T_{c,hydrocarbons}$	72.9	**98.0**	**100**	**100**	**6.3%**

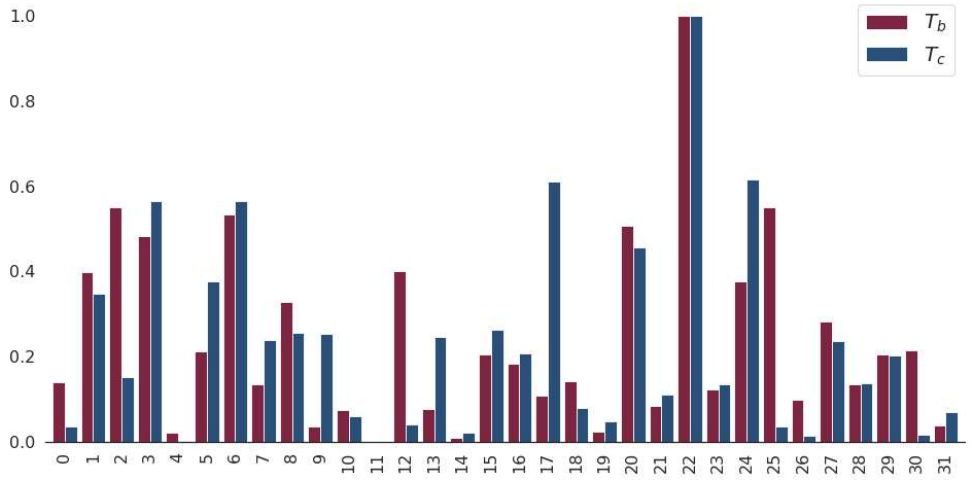
4.2.3. Explainability and interpretability of ML models

As ML is dominated by deep neural networks these days, we focus our discussion on its limitations for certain scientific applications. One of the major limitations is their black-box nature and lack of explainability. However, there are certain indirect approaches that could be used to better understand seemingly black-box models. As an example, we consider the most commonly used approach is that of computing Shapley values, a concept from cooperative game theory used to compute the contribution of each player to the final payout, to understand the feature importance for the property estimation

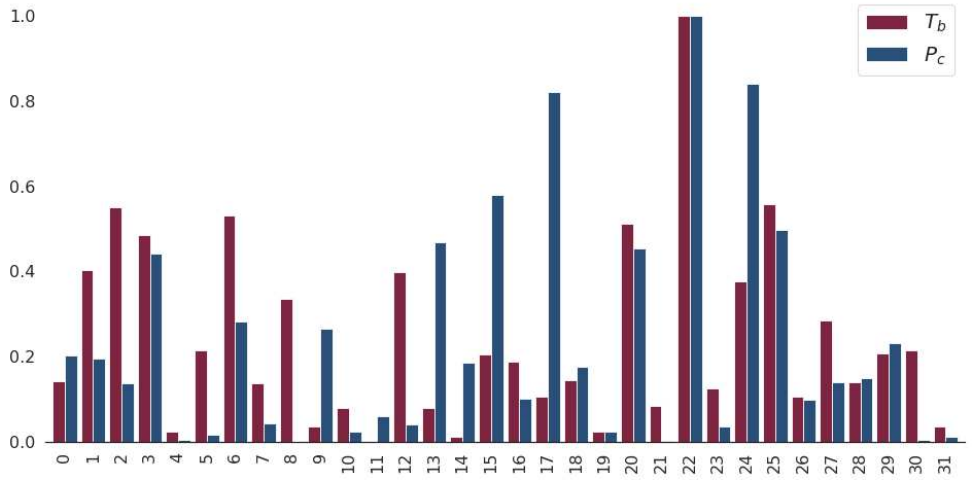
task, as proposed by Lundberg and Lee [84]. Shapley values are a measure of the average marginal contribution of a feature across all possible coalitions (or feature combinations). To quantify the importance of a given feature, different feature coalitions are simulated and the predicted value for the different contributions are averaged and subtracted from the predicted value with the given feature in the coalition. This computation is performed for all possible coalitions, and the Shapley value is the average of all the marginal contributions to all possible coalitions. Formally, the Shapley value for a feature j is defined as,

$$\hat{\phi}_j = \frac{1}{M} \sum_{m=1}^{M} (\hat{f}(x_{+j}^m) - \hat{f}(x_{-j}^m)) \tag{15}$$

where $\hat{f}(x_{+j}^m)$ is the prediction for x with a random number of feature values replaced by feature values from a random data point z except for the respective value of feature j; x_{+j}^m is identical to x_{-j}^m except that the value x_j^m is taken from the random sample z in x_{-j}^m; features on the left of x_j have values from the original observations and those on the right of x_j take their values from a random instance; and M is the number of instances generated. This procedure is repeated M times for all the features and feature importance are computed.



(a) T_b vs T_c



(b) T_b vs P_c

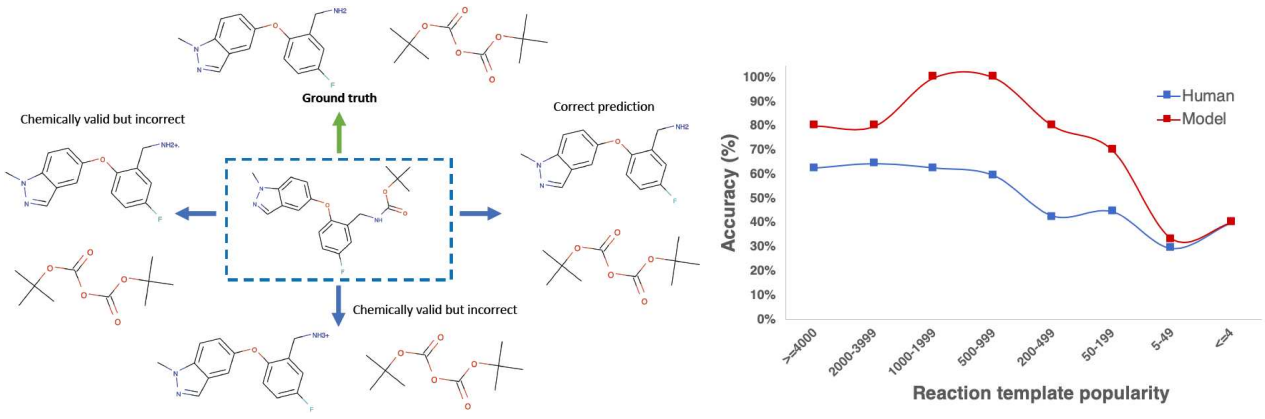
Figure 3: Comparison of feature importance (contributions) for regression models for T_b, T_c, and P_c. Based on the underlying chemistry-based correlations, it is expected that T_b and T_c would have similar feature importance, whereas T_b and P_c would have relatively higher differences in feature importance. This behavior is observed in the above comparison plots between T_b vs T_c and T_b vs P_c. Figure from Mann et al. [34].

The feature importance obtained using this approach has been shown in [34] to correlate with the underlying chemistry and expert intuition. For instance, Figure 3 shows the side-by-side plots of feature importance for T_b, T_c in Figure 3(a) and T_b, P_c in Figure 3(b). An interesting observation based on this is that the feature importance are very similar for T_b and T_c, whereas the correlation between feature importance for T_b and P_c is much weaker. Both the observations agree with underling chemistry and the similarity (and differences) in driving forces responsible for the properties. This points towards a possible correlation between the features and the underlying molecular chemistry (such as intermolecular forces and interactions between various molecular groups) that is captured

by the ML model to some extent. Moreover, as shown by Mann et al. [34], the computed feature importance could be further used to prune or simplify the model by retraining the ML model only using the most important descriptors identified for estimating a given property. Thus, this offers an approach to develop custom-built ML models that are simpler but more accurate, and hence, better suited for chemicals-based product design.

4.3. Analysis of data, design problem and design solution - Use of AI techniques

Computer-aided reaction synthesis is an area where the additional benefits of AI has been realized with several works reporting different approaches for solving both the forward reaction prediction as well as the inverse or the retrosynthesis prediction problem. Moreover, realistic retrosynthetic analysis involving multi-step synthesis reactions with intermediates have been developed and successfully validated against expert-developed synthesis planning. Figure 4(a) shows the reactants predicted by an AI-based chemistry-informed single-step retrosynthesis model for synthesizing a given target molecule. The ground truth reactants are shown in the figure along with the top-3 most likely reactant-sets predicted by the model. It is observed that the most likely prediction matches the ground truth and even the incorrect predictions (second and third) are very close to the ground truth. This means that the AI model has largely learned the correlations characterizing the complex transformations that molecules undergo during chemical reactions and gets the molecular syntax right. Figure 4(b) shows the comparison of the AI-model's accuracy against those of expert chemists on the forward reaction prediction problem across different class of reactions (easier ones are towards the left with increasingly difficult reactions on the right). The model outperforms the chemists across each of the reaction categories except for the last two where the performance is comparable to the human chemists. However, it must be highlighted here that one of the biggest challenges across a majority of the ML methods is to ensure the system respects the underlying physics and chemistry (and biology). Again, explicitly incorporating these right at the beginning using symbolic AI could go a long way in addressing this challenge and ensure wider adoption in chemicals-based product design as argued by Venkatasubramanian and Mann [19].



(a) (b)

Figure 4: Computer aided reaction synthesis (a) Example predictions by retrosynthesis model developed by Mann and Venkatasubramanian [85] (b) Comparison of forward prediction accuracy between model and human chemists shown in Mann and Venkatasubramanian [68]

4.4. Integration of computer-aided methods and tools

Figure 5 presents an example of a general framework for chemical product design together with five of its associated computer-aided tools (software components) that tackles some of the challenges and issues highlighted in section 3. Note that only a few representative paths in each decision tree are shown for purposes of illustration of the concepts. Some of the features in this framework have been implemented and tested in the ProCAPD (computer-aided product design) software tool by Kalakul et al. [11]. As also highlighted in Figure 1, the main components of the framework are an interface (design problem) for communication with the user and the different software components of the framework. In this example, four software components are highlighted, namely, a generator of (product) candidates; a library of databases; a library of property models, and a solver library, in addition to the main (design problem) interface. The communication between all the components is through the chemicals (single molecules or mixtures) or through the list of properties. If a list of chemicals is available from the design problem interface, then it is a product evaluation problem (that is, a properties prediction problem), as highlighted in Figure 2. In this case, the needed properties can be retrieved from the database library, if they are available, or, they may be predicted, if a property model and its associated parameters are available in the property models library. If the design problem specifies a list of target properties with their desired values together with the type of product to be designed, then it is one of many chemical product design problems, which can be solved by following the work-flow highlighted in Figure 1. In this case, the design problem interface selects a solution approach and the corresponding work-flow

in the solver library guides the user through the solution steps involving the candidate generator, the database library and the property model library. For example, in the generate-test paradigm, a sequential approach is employed. That is, first enumerate a list of candidates and then use either the database library or property model library to predict their properties and then use a screening method to find a feasible set. In the mathematical programming approach, the mathematical model of the design problem including the representation of the enumeration procedure together with the models for the target properties are solved with a direct approach (solve all equations simultaneously) or as a decomposition-based approach (solve sub-sets of equations according to a hierarchical order). Note that only the main decision steps are shown in each of the software component diagrams in order to illustrate the main concepts. The SMILES of molecular structures is one option to connect all components in the product design work-flow. In the text below, each of the components are briefly described.

4.4.1. Design problem interface

The user and the other software components communicate through this interface, which is the green decision tree on the top right hand corner. The problem definition decision tree is highlighted for molecular and mixture design. The end-point is a set of properties, which can be a combination of the four types of properties (see section 2.3). Note that while the decision tree is the same for many molecular design problems, the property sets and therefore, the corresponding property models are different and the design problem type specifies the types of chemicals as well as the product type. Consider, for example, design of an API, or a solvent, or a refrigerant. For each of these products, the target property sets, the sections of the database library and the property model library are different. Similarly, the decision tree for liquid formulations and blends are similar but the target property sets for specific products are different. A knowledge-based system is used to convert specific product needs-functions to target property sets. Another knowledge-based system is necessary to select the appropriate property models. Based on the target properties and the type of product to be designed, yet another knowledge-based system is necessary to select the solution strategy together with the property models. As indicated in Figure 5, currently 189 product types with the needs and functions have been identified by Syeda et al. [81]. The output from this tool, which can be used by the other software components, is a set of instructions and the problem definition details. The output for the user is a set of promising candidates that matches the target properties.

4.4.2. Candidate generator

This software component is needed for product design problems only. The decision tree (shown in top left hand corner of Figure 5) provides separate paths for each product type, such as molecular design and mixture design, because the building blocks are different for each product type. From an initial selection of building blocks, candidates are generated that satisfy constraints such as valency rule for single molecule products and normalized compositions for mixtures where the number of compounds may or may not be fixed. In the case of single molecular products, two types of commonly used building blocks, which can generate different types of product candidates (for example, organic chemicals, ionic liquids and polymer repeat units) are highlighted. In the case of mixture design, the decision tree for liquid formulations and blends are highlighted. Note that chemicals involved in each of the building blocks (for example, active ingredients, additives, etc.) are different for different types of products. This software component influences the selection of models in the property models library because the same building blocks need to be used for prediction of at least a sub-set of target properties. The output from this software component is a set of candidates represented by their building blocks and/or SMILES.

4.4.3. Database library

This software component allows the user and the solver library to retrieve collected measured data for specified molecules and mixtures (shown in the bottom left hand corner of Figure 5). The size of the database library needs to increase continuously as more and more measured data are collected and added, after verification. Only the decision tree for organic chemicals is highlighted in Figure 5. At the end of the tree, the available measured data can be viewed or retrieved. Quantitative and qualitative data for a total of 919823 chemicals are available from different sources in the work by Syeda et al. [81]. The end-blocks of the decision tree indicates the numbers of properties available for each property type. The mixture data, which potentially can be very large, is mainly limited to binary mixtures. They could potentially include solid solubility, saturation temperatures, binary azeotrope details, and many more. One source of new data in databases could be pseudo-measured data generated through computationally expensive but accurately predicted property values, increasing thereby, the potential search space and application range of problems that could be solved. Models such as COSMO-based property prediction for specific mixtures, the PC-SAFT equation of state and ML-based property estimation are gaining use in this area. The output from this software component are property values

for a set of target properties of a product candidate represented by, for example, functional groups.

4.4.4. Property model library

As the measured data available in the database library is limited, especially for mixture properties, a suite of models is usually needed in product design. The scope and significance of any product design tool is related to the application range and accuracy of the models in the property model library. The bottom right hand corner of Figure 5 illustrates the decision tree for pure compound and mixture properties. Note that for each property a model is necessary, which can be different for different chemicals. That is, the model for a primary property, such as normal melting point, is different for organic chemicals, ionic liquids and polymers. Even in the case of organic chemicals, different models for different types of chemicals may need to be selected. For example, different GC-based models may be used for critical properties of refrigerants and solvents. This software component needs to be integrated with the software component for candidate generator as both need to use the same building blocks to represent the products. Integration of this software component with a computer-aided modeling tool (not shown in Figure 5) could help to quickly test and implement new models in the property model library. The output from this software component is predicted property values for a set of target properties.

4.4.5. Solver library

The solver library connects the product design interface with the other three software components. Based on the selected product design work-flow, different solution strategies could be used. A collection of product design templates that stores different work-flows, the associated data-flow and the tools needed for each step of the work-flow could guide the user. Note that the work-flow, for example, for liquid formulations is different from the work-flow for blends, even though both belong to the class of mixture design problems. Also, even though the work-flow for different products, for example in blends, are the same, the data-flow, the associated target properties and the corresponding property models (or the model parameters for the same models) are different. An important step in the work-flow is the screening of generated candidates. Here, use of AI techniques could help to make the templates intelligent and smart. Output from this software component is a product design template containing, for example, the required work-flow, a set of models for prediction of target properties and a list of feasible candidates.

4.4.6. Integration aspects

The flexibility and versatility of the framework's ability to access specific options within the needed software components make it more general and widens the scope of the applications. Well-defined interfaces for communication between the software components could be established through training and testing on different product design problems. Use of AI techniques together with appropriate knowledge representation systems could help to make the framework more generic and intelligent. Use of ML-based and other computationally expensive models could help to increase the size of the databases with the database library.

4.5. Hybrid options

With respect to the product design work-flows shown in Figures 1,2, and 5, an important issue is to consider the GC-based simple and XX-based ML (where XX means GC or other forms of molecular representation) models such that the search space for product design problems could be increased and products with more accurate predicted properties could be determined. One option is to use the GC-based simple model in the inner-loop and verify the best solution in the outer-loop with the XX-based ML models. If the GC-based simple and the XX-based ML models use the same molecular representation method, other schemes of combining the different methods could also be considered. More work is needed to establish the convergence criteria for these mixed model design work-flows. Another option would be to use the XX-based ML models to fill out the databases and use these values in the generate-test algorithms, where, testing is done through database comparisons as well as property predictions. If the number of alternatives is not large, the XX-based ML models could also be directly used, if the time to find an innovative and novel design is not a factor.

4.6. Pitfalls in computational product design

This is an important topic because the product designs must not fail when they are used in business-2-business (B2B) or business-2-consumer (B2C) modes. To guarantee correct performance additional precaution needs to be taken with respect to,

- Problem formulation - is the design problem correctly formulated? That is, are the needs for the product correctly and consistently converted to target properties with associated target values?
- Increased use of computer-aided methods and tools for product design - Common functionality among many methods suggests the utility of a software library that could contain the diversity

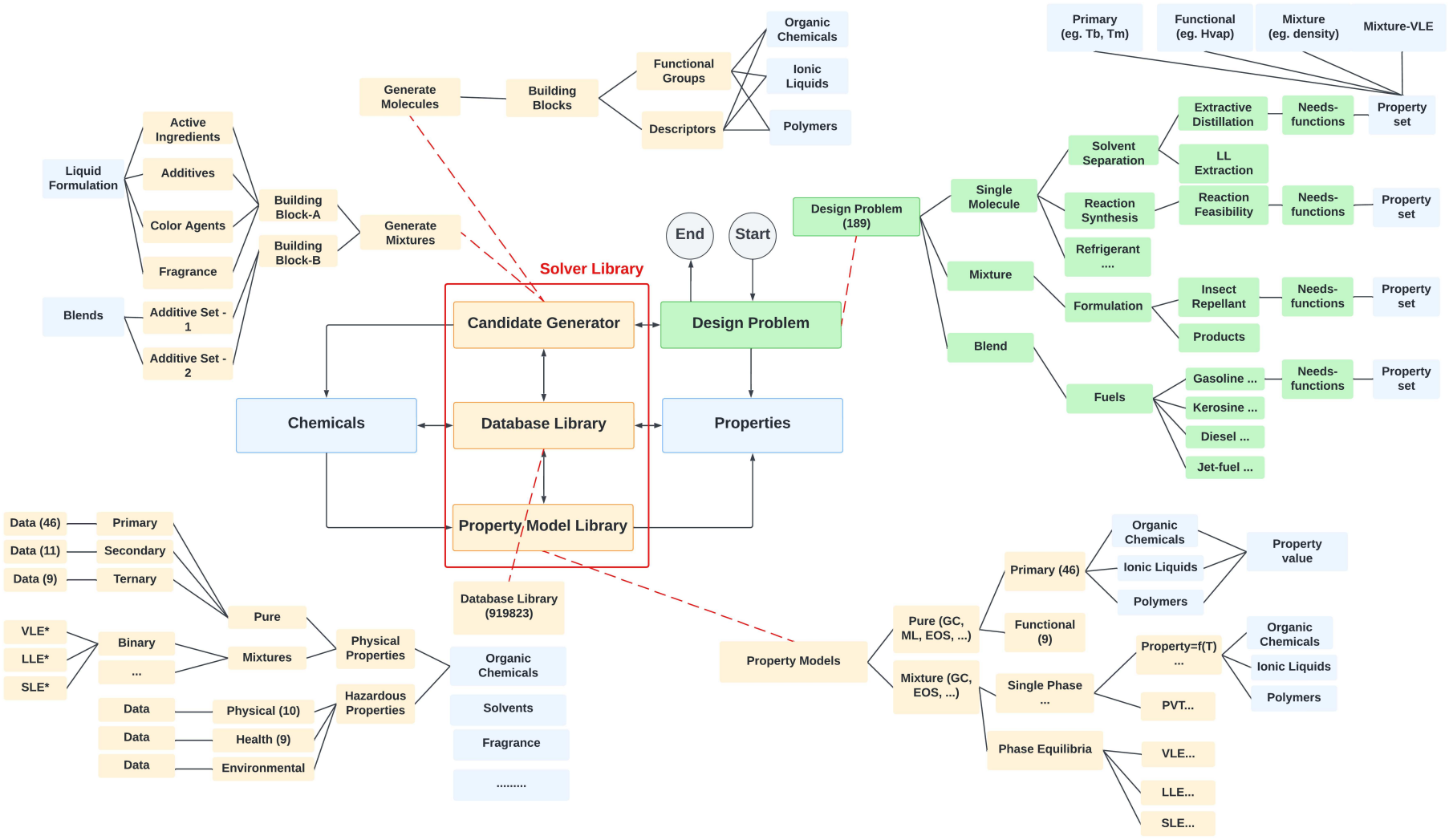


Figure 5: A unified chemicals-based product design framework comprising a combination of sub-modules shown in Figure 1 and 2

of techniques for describing molecules, calculating properties, and exploring the molecular design space. Such a library would lower barriers to using alternative design methods and promote better understanding of the relative capabilities of different design techniques as argued by Austin [63].

- Safety, health, environmental impacts - to avoid proposing products with chemicals that may be banned or dangerous, the safety, health and environmental impacts need to considered. Should it be done at the end or could molecule generation algorithms also incorporate some of these issues?

- Hybridization - the scope and significance of the design methods could be improved through an optimal integration of models of different dimensions and scales, data from different sources and different AI-techniques for decision making.

- Validation - the designed product functions need to be validated with experiments for all model (and data) based design work-flows. Design of experiments could be incorporated to propose the optimal set of experiments to validate the product functions. Also, design algorithms should be compared to enumerated design spaces whenever possible and a common set of enumerated problems would aid in algorithm benchmarking and promote advances in the field. Development of useful validation problems would help to establish the methods and tools as argued by Austin [63].

- Hardware and/or software issues - the advances in computational power, speed and storage, combined with developments in modeling and AI techniques offers the possibility for significant advances in the state of the art in computer-aided chemicals-based product design.

5. Conclusions

The design and synthesis of chemicals-based products has not reached the same status as computer-aided process design because the former has a wider search space, requires the consideration of properties for which measured data may not be available, and the number and type of products is very large. Consequently, many products are still designed using the laborious Edisonian trial-and-error based experimentation. While this approach may not give the best product, it nevertheless is a safe option, because performance of the product is simultaneously verified.

Model and data-based product design work-flow, particularly using AI techniques, present the opportunity to significantly reduce the time to develop and market a product. The model and data, however, need to be validated first. It needs to be understood that to predict properties of a chemical, its molecular structure needs to be known, while, in molecular design, if the desired property is specified, molecular structures that satisfy the desired properties may be found.

The key issue is how to represent the molecular structure. Graph theory-based description of chemicals need the structure as an input. Another challenge is how to tackle the application range of the models and the availability of knowledge of different types of chemicals and products as they influence the scope and significance of the design methods. Also, from a circular economy point of view, recycle, re-use, and recovery of the chemicals in the products also need to be considered.

Furthermore, it seems superfluous to state that the thermodynamic principles cannot be violated in the final design or to stress that all hazardous effects (safety, health, and environmental) need to be considered when screening alternatives to avoid practical problems, but one observes such publications now and then. These mistakes often stem from the incorrect formulation of the design and synthesis problems with all their constraints.

The advances in modeling in terms of machine learning and data-analysis can help address these challenges. However, in all the current excitement about data science, we must not ignore the already developed concepts, theory, models based on first principles that already exist in our domain. It is necessary to use all of them together, like a symbiosis of data, models, and tools to achieve the desired final results. To this end, the development of hybrid artificial intelligence models that combine symbolic knowledge with numeric techniques is the most promising approach [16, 19, 86, 87].

Acknowledgements

This work was supported by the National Science Foundation (NSF) under Grant No. 2132142 and carried out at Columbia University. The authors acknowledge the valuable contribution of Dr. Anjan K Tula with respect to the GC-ML model computations.

References

1. Gani, R. & Ng, K. M. Product design–molecules, devices, functional products, and formulated products. *Computers & Chemical Engineering* **81,** 70–79 (2015).

2. Gani, R. Chemical product design: challenges and opportunities. *Computers & Chemical Engineering* **28,** 2441–2457 (2004).

3. Hill, M. Product and process design for structured products. *AIChE Journal* **50,** 1656–1661 (2004).

4. Adjiman, C. S., Sahinidis, N. V., Vlachos, D. G., Bakshi, B., Maravelias, C. T. & Georgakis, C. Process systems engineering perspective on the design of materials and molecules. *Industrial & Engineering Chemistry Research* **60,** 5194–5206 (2021).

5. Zhang, L., Mao, H., Liu, Q. & Gani, R. Chemical product design–recent advances and perspectives. *Current Opinion in Chemical Engineering* **27,** 22–34 (2020).

6. O'Connell, J. P., Gani, R., Mathias, P. M., Maurer, G., Olson, J. D. & Crafts, P. A. Thermodynamic property modeling for chemical process and product engineering: some perspectives. *Industrial & engineering chemistry research* **48,** 4619–4637 (2009).

7. Gani, R. Group contribution-based property estimation methods: advances and perspectives. *Current Opinion in Chemical Engineering* **23,** 184–196 (2019).

8. Gmehling, J., Constantinescu, D. & Schmid, B. Group contribution methods for phase equilibrium calculations. *Annual review of chemical and biomolecular engineering* **6,** 267–292 (2015).

9. Fredenslund, A. A Group Contribution Method. *Vapor-liquid equilibria using UNIFAC* **380** (1977).

10. Abrams, D. S. & Prausnitz, J. M. Statistical thermodynamics of liquid mixtures: a new expression for the excess Gibbs energy of partly or completely miscible systems. *AIChE Journal* **21,** 116–128 (1975).

11. Kalakul, S., Zhang, L., Fang, Z., Choudhury, H. A., Intikhab, S., Elbashir, N., Eden, M. R. & Gani, R. Computer aided chemical product design–ProCAPD and tailor-made blended products. *Computers & Chemical Engineering* **116,** 37–55 (2018).

12. Harper, P. M., Gani, R., Kolar, P. & Ishikawa, T. Computer-aided molecular design with combined molecular modeling and group contribution. *Fluid Phase Equilibria* **158,** 337–347 (1999).

13. Austin, N. D., Sahinidis, N. V. & Trahan, D. W. Computer-aided molecular design: An introduction and review of tools, applications, and solution techniques. *Chemical Engineering Research and Design* **116,** 2–26 (2016).

14. Jonuzaj, S., Gupta, A. & Adjiman, C. S. The design of optimal mixtures from atom groups using Generalized Disjunctive Programming. *Computers & Chemical Engineering* **116,** 401–421 (2018).

15. Liu, Q., Zhang, L., Liu, L., Du, J., Tula, A. K., Eden, M. & Gani, R. OptCAMD: an optimization-based framework and tool for molecular and mixture product design. *Computers & Chemical Engineering* **124,** 285–301 (2019).

16. Venkatasubramanian, V. The promise of artificial intelligence in chemical engineering: Is it here, finally. *AIChE J* **65,** 466–478 (2019).

17. Goldsmith, B. R., Esterhuizen, J., Liu, J.-X., Bartel, C. J. & Sutton, C. Machine learning for heterogeneous catalyst design and discovery (2018).

18. Jirasek, F. & Hasse, H. Perspective: Machine Learning of Thermophysical Properties. *Fluid Phase Equilibria* **549,** 113206 (2021).

19. Venkatasubramanian, V. & Mann, V. Artificial intelligence in reaction prediction and chemical synthesis. *Current Opinion in Chemical Engineering* **36,** 100749 (2022).

20. Zhang, L., Babi, D. K. & Gani, R. New vistas in chemical product and process design. *Annual review of chemical and biomolecular engineering* **7,** 557–582 (2016).

21. Churi, N. & Achenie, L. E. Novel mathematical programming model for computer aided molecular design. *Industrial & engineering chemistry research* **35,** 3788–3794 (1996).

22. Li, K., Chang, F., Shi, S., Jiang, C., Bai, Y., Dong, H., Meng, X., Wu, J. C. & Zhang, X. A new method of Ionic Fragment Contribution-Gradient Boosting Regressor for predicting the infinite dilution activity coefficient of dichloromethane in ionic liquids. *Fluid Phase Equilibria,* 113622 (2022).

23. Patel, S. J., Ng, D. & Mannan, M. S. QSPR flash point prediction of solvents using topological indices for application in computer aided molecular design. *Industrial & Engineering Chemistry Research* **48,** 7378–7387 (2009).

24. Abramenko, N., Kustov, L., Metelytsia, L., Kovalishyn, V., Tetko, I. & Peijnenburg, W. A review of recent advances towards the development of QSAR models for toxicity assessment of ionic liquids. *Journal of hazardous materials* **384,** 121429 (2020).

25. Sippl, W., Contreras, J.-M., Parrot, I., Rival, Y. M. & Wermuth, C. G. Structure-based 3D QSAR and design of novel acetylcholinesterase inhibitors. *Journal of Computer-Aided Molecular Design* **15,** 395–410 (2001).

26. Chemmangattuvalappil, N. G. & Eden, M. R. A novel methodology for property-based molecular design using multiple topological indices. *Industrial & Engineering Chemistry Research* **52,** 7090–7103 (2013).

27. Visco Jr, D. P., Pophale, R. S., Rintoul, M. D. & Faulon, J.-L. Developing a methodology for an inverse quantitative structure-activity relationship using the signature molecular descriptor. *Journal of Molecular Graphics and Modelling* **20,** 429–438 (2002).

28. Muthukrishnan, R. & Rohini, R. *LASSO: A feature selection technique in predictive modeling for machine learning* in *2016 IEEE international conference on advances in computer applications (ICACA)* (2016), 18–20.

29. Venkatasubramanian, V., Chan, K. & Caruthers, J. M. Computer-aided molecular design using genetic algorithms. *Computers & Chemical Engineering* **18,** 833–844 (1994).

30. Zang, Q., Mansouri, K., Williams, A. J., Judson, R. S., Allen, D. G., Casey, W. M. & Kleinstreuer, N. C. In silico prediction of physicochemical properties of environmental chemicals using molecular fingerprints and machine learning. *Journal of chemical information and modeling* **57,** 36–49 (2017).

31. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *Journal of chemical information and modeling* **50,** 742–754 (2010).

32. Dobbelaere, M. R., Ureel, Y., Vermeire, F. H., Tomme, L., Stevens, C. V. & Van Geem, K. M. Machine Learning for Physicochemical Property Prediction of Complex Hydrocarbon Mixtures. *Industrial & Engineering Chemistry Research* (2022).

33. Moriwaki, H., Tian, Y.-S., Kawashita, N. & Takagi, T. Mordred: a molecular descriptor calculator. *Journal of cheminformatics* **10,** 1–14 (2018).

34. Mann, V., Brito, K., Gani, R. & Venkatasubramanian, V. Hybrid, Interpretable Machine Learning for Thermodynamic Property Estimation using Grammar2vec for Molecular Representation. *Fluid Phase Equilibria* **561,** 113531 (2022).

35. Goh, G. B., Hodas, N. O., Siegel, C. & Vishnu, A. Smiles2vec: An interpretable general-purpose deep neural network for predicting chemical properties. *arXiv preprint arXiv:1712.02034* (2017).

36. Jaeger, S., Fulle, S. & Turk, S. Mol2vec: unsupervised machine learning approach with chemical intuition. *Journal of chemical information and modeling* **58,** 27–35 (2018).

37. Ishida, S., Miyazaki, T., Sugaya, Y. & Omachi, S. Graph neural networks with multiple feature extraction paths for chemical property estimation. *Molecules* **26,** 3125 (2021).

38. Alshehri, A. S., Tula, A. K., You, F. & Gani, R. Next generation pure component property estimation models: With and without machine learning techniques. *AIChE Journal* **68,** e17469 (2022).

39. Ceriani, R., Gani, R. & Meirelles, A. J. Prediction of heat capacities and heats of vaporization of organic liquids by group contribution methods. *Fluid Phase Equilibria* **283,** 49–55 (2009).

40. Peng, Y., Goff, K. D., dos Ramos, M. C. & McCabe, C. Developing a predictive group-contribution-based SAFT-VR equation of state. *Fluid Phase Equilibria* **277,** 131–144 (2009).

41. Jaber, M., Babe, W., Sauer, E., Gross, J., Lugo, R. & De Hemptinne, J. An improved group contribution method for PC-SAFT applied to branched alkanes: Data analysis and parameterization. *Fluid Phase Equilibria* **473,** 183–191 (2018).

42. Marrero, J. & Gani, R. Group-contribution based estimation of pure component properties. *Fluid phase equilibria* **183,** 183–208 (2001).

43. Hukkerikar, A. S., Sarup, B., Ten Kate, A., Abildskov, J., Sin, G. & Gani, R. Group-contribution+ (GC+) based estimation of properties of pure components: improved property estimation and uncertainty analysis. *Fluid Phase Equilibria* **321,** 25–43 (2012).

44. Nannoolal, Y., Rarey, J. & Ramjugernath, D. Estimation of pure component properties: Part 2. Estimation of critical property data by group contribution. *Fluid Phase Equilibria* **252,** 1–27 (2007).

45. Joback, K. G. & Reid, R. C. Estimation of pure-component properties from group-contributions. *Chemical Engineering Communications* **57,** 233–243 (1987).

46. Kolská, Z., Rika, V. & Gani, R. Estimation of the enthalpy of vaporization and the entropy of vaporization for pure organic compounds at 298.15 K and at normal boiling temperature by a group contribution method. *Industrial & engineering chemistry research* **44,** 8436–8454 (2005).

47. Velásquez, J. A., Hernández, J. P., Forero, L. A. & Cardona, L. F. Prediction of phase equilibria, density, speed of sound and viscosity of 2-alkoxyethanols mixtures: A comparison study between SAFT type EoSs and a modified PR EoS. *Fluid Phase Equilibria,* 113570 (2022).

48. Soave, G. Equilibrium constants from a modified Redlich-Kwong equation of state. *Chemical engineering science* **27,** 1197–1203 (1972).

49. Tochigi, K., Matsuda, H. & Kurihara, K. Estimation of kinematic viscosities and thermal conductivities for liquid mixtures using ASOG-VLE, ASOG-VISCO and ASOG-ThermConduct models. *Fluid Phase Equilibria* **565,** 113668 (2023).

50. Fu, Y., Chen, Y., Zhang, C., Lei, Y. & Liu, X. Prediction surface tension of ionic liquid–water mixtures using a hybrid group contribution and artificial neural network method. *Fluid Phase Equilibria,* 113571 (2022).

51. Tun, H., Hao, Y., Haddix, M. & Chen, C.-C. Thermodynamic Solubility Modeling of 2, 2, 4, 4, 6, 6-Hexanitrostilbene (HNS). *Fluid Phase Equilibria,* 113627 (2022).

52. Wang, J., Song, Z., Lakerveld, R. & Zhou, T. Solvent Selection for Chemical Reactions toward Optimal Thermodynamic and Kinetic Performances: Group Contribution and COSMO-based Modeling. *Fluid Phase Equilibria,* 113623 (2022).

53. NguyenHuynh, D. & Nguyen-Thi, T.-X. Modeling the fluid phase behavior of amines, aromatic amines and their mixtures using the modified group-contribution PC-SAFT. *Fluid Phase Equilibria* **551,** 113274 (2022).

54. Shaahmadi, F., Smith, S. A., Schwarz, C. E., Burger, A. J. & Cripwell, J. T. Group-Contribution SAFT Equations of State: A Review. *Fluid Phase Equilibria,* 113674 (2022).

55. Privat, R. & Jaubert, J.-N. The state of the art of cubic equations of state with temperature-dependent binary interaction coefficients: From correlation to prediction. *Fluid Phase Equilibria,* 113697 (2022).

56. Roosta, A., Haghbakhsh, R., Duarte, A. R. C. & Raeissi, S. Machine learning coupled with group contribution for predicting the density of deep eutectic solvents. *Fluid Phase Equilibria* **565,** 113672 (2023).

57. Martinez-Hernandez, E., Zenteno, C., Valencia, D. & Aburto, J. Prediction of viscosity of biomass-based molecules using atom modules and modularity as descriptors in neural network models. *Fluid Phase Equilibria* **565,** 113648 (2023).

58. Watson, G. S. Linear least squares regression. *The Annals of Mathematical Statistics,* 1679–1699 (1967).

59. Vapnik, V. N. The nature of statistical learning. *Theory* (1995).

60. Schulz, E., Speekenbrink, M. & Krause, A. A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology* **85,** 1–16 (2018).

61. Dietterich, T. G. *Ensemble methods in machine learning* in *International workshop on multiple classifier systems* (2000), 1–15.

62. Sze, V., Chen, Y.-H., Yang, T.-J. & Emer, J. S. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE* **105,** 2295–2329 (2017).

63. Austin, N. D. The case for a common software library and a set of enumerated benchmark problems in computer-aided molecular design. *Current Opinion in Chemical Engineering* **35,** 100724 (2022).

64. Conte, E., Gani, R., Cheng, Y. S. & Ng, K. M. Design of formulated products: experimental component. *AIChE journal* **58,** 173–189 (2012).

65. Yunus, N. A., Gernaey, K. V., Woodley, J. M. & Gani, R. A systematic methodology for design of tailor-made blended products. *Computers & chemical engineering* **66,** 201–213 (2014).

66. Mayr, L. M. & Bojanic, D. Novel trends in high-throughput screening. *Current opinion in pharmacology* **9,** 580–588 (2009).

67. Phillips, K. A., Wambaugh, J. F., Grulke, C. M., Dionisio, K. L. & Isaacs, K. K. High-throughput screening of chemicals as functional substitutes using structure-based classification models. *Green Chemistry* **19,** 1063–1074 (2017).

68. Mann, V. & Venkatasubramanian, V. Predicting chemical reaction outcomes: a grammar ontology-based transformer framework. *AIChE Journal* **67,** e17190 (2021).

69. Segler, M. H., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555,** 604–610 (2018).

70. Katare, S., Caruthers, J. M., Delgass, W. N. & Venkatasubramanian, V. An intelligent system for reaction kinetic modeling and catalyst design. *Industrial & engineering chemistry research* **43,** 3484–3512 (2004).

71. Sun, R., Dai, H., Li, L., Kearnes, S. & Dai, B. Energy-based view of retrosynthesis. *arXiv preprint arXiv:2007.13437* (2020).

72. Schwaller, P., Vaucher, A. C., Laino, T. & Reymond, J.-L. Prediction of chemical reaction yields using deep learning. *Machine learning: science and technology* **2,** 015016 (2021).

73. Schwaller, P., Petraglia, R., Zullo, V., Nair, V. H., Haeuselmann, R. A., Pisoni, R., Bekas, C., Iuliano, A. & Laino, T. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chemical science* **11,** 3316–3325 (2020).

74. Mann, V. & Venkatasubramanian, V. AI-driven hypergraph network of organic chemistry: network statistics and applications in reaction classification. *Reaction Chemistry & Engineering* (2023).

75. Baylon, J. L., Cilfone, N. A., Gulcher, J. R. & Chittenden, T. W. Enhancing retrosynthetic reaction prediction with deep learning using multiscale reaction classification. *Journal of chemical information and modeling* **59,** 673–688 (2019).

76. Aldeghi, M., Graff, D. E., Frey, N., Morrone, J. A., Pyzer-Knapp, E. O., Jordan, K. E. & Coley, C. W. Roughness of molecular property landscapes and its impact on modellability. *arXiv preprint arXiv:2207.09250* (2022).

77. Zhou, Z., Li, X. & Zare, R. N. Optimizing chemical reactions with deep reinforcement learning. *ACS central science* **3,** 1337–1344 (2017).

78. Gottipati, S. K., Sattarov, B., Niu, S., Pathak, Y., Wei, H., Liu, S., Blackburn, S., Thomas, K., Coley, C., Tang, J., *et al. Learning to navigate the synthetically accessible chemical space using reinforcement learning* in *International Conference on Machine Learning* (2020), 3668–3679.

79. Popova, M., Isayev, O. & Tropsha, A. Deep reinforcement learning for de novo drug design. *Science advances* **4,** eaap7885 (2018).

80. Kuprasertwong, N., Padungwatanaroj, O., Robin, A., Udomwong, K., Tula, A., Zhu, L., Zhou, L., Wang, B., Wang, S. & Gani, R. Computer-Aided Refrigerant Design: New Developments. **50,** 19–24 (2021).

81. Syeda, S. R., Khan, E. A., Padungwatanaroj, O., Kuprasertwong, N. & Tula, A. K. A perspective on hazardous chemical substitution in consumer products. *Current Opinion in Chemical Engineering* **36,** 100748 (2022).

82. Hukkerikar, A. S., Kalakul, S., Sarup, B., Young, D. M., Sin, G. & Gani, R. Estimation of environment-related properties of chemicals for design of sustainable processes: development of group-contribution+ (GC+) property models and uncertainty analysis. *Journal of chemical information and modeling* **52,** 2823–2839 (2012).

83. Alshehri, A. S., Gani, R. & You, F. Deep learning and knowledge-based methods for computer-aided molecular designtoward a unified approach: State-of-the-art and future directions. *Computers & Chemical Engineering* **141,** 107005 (2020).

84. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. *Advances in neural information processing systems* **30** (2017).

85. Mann, V. & Venkatasubramanian, V. Retrosynthesis prediction using grammar-based neural machine translation: An information-theoretic approach. *Computers & Chemical Engineering* **155,** 107533 (2021).

86. Chakraborty, A., Sivaram, A. & Venkatasubramanian, V. AI-DARWIN: A first principles-based model discovery engine using machine learning. *Computers & Chemical Engineering* **154,** 107470 (2021).

87. Venkatasubramanian, V. Teaching Artificial Intelligence to Chemical Engineers: Experience from a 35-year-old course. *Chemical Engineering Education,* 231–240 (2022).