Computer Aided Recipe Design: Optimization of Polydisperse Chemical Mixtures using Molecular Descriptors

Anja Massolle[¶], *Jakob Schneider*[¶], *Jan Meyer*[¶], *Christoph Loschen*[¶*]

[¶]Covestro Deutschland AG, Kaiser-Wilhelm-Allee 60

51373 Leverkusen, Deutschland

Email: christoph.loschen@covestro.com

Abstract: A workflow has been developed allowing for the computer aided design and optimization of reactive systems using the concept of molecular descriptor-based similarity. Unlike single-molecule models most often used in polymer informatics, an important feature of this approach is to allow for a more realistic description of reaction mixtures by accounting for polydispersity and individual chain topology.

Starting from a specific set of ingredients, *i.e.,* a chemical recipe or formulation, simulations based on Gillespie's kinetic Monte Carlo scheme are used to generate oligo- and polymeric reaction mixtures. By using the distance / similarity in molecular and topological descriptor space as a metric, the initial recipe is then modified iteratively using a Bayesian optimizer. Target of the optimization procedure is either another chemical recipe with different ingredients or alternatively, a set of desirable descriptors and properties. A key step of the process is the transformation of the graph representing individual polymer species as obtained by the kinetic simulation into atomistic species described as SMILES strings, which enables the computation of a rich set of additional descriptors. This rather general mapping is achieved exploiting similarities between the BNGL and the SMILES graph notation. The workflow is demonstrated on common polyether and polyester oligomeric systems as typically used in chemical industry, but is generally applicable to any other polymer chemistry.

Introduction

Within the last decades, the rational design of materials via computational modeling has become a topic of increasing importance in chemical and pharmaceutical industry. A variety of methods ranging from quantum chemistry and molecular dynamics to solvation thermodynamics and cheminformatics is available for the computational chemist to solve specific challenges in an industrial setting.[1-4]

In the context of materials modelling, the term computer-aided materials design (CAMD) is usually referring to an approach being used for example in the design of drugs[5], polymer materials[6] or solvents.[7-8] In classical CAMD, as for example reviewed by Austin et al.,[9] first a set of structure-based property prediction models is defined and then single or multiple desired target ranges for the specified properties are selected. Subsequently, the inverse design problem is solved by systematically varying the chemical structure according to a mathematical optimization process. Although such an approach has been used for polymers,[6, 10-11] those models usually involve monodisperse molecule models, that assume either a single finite or infinite polymer chain.

For example, group contributions-based approaches developed for single chain molecules have been extended to generate candidate polymers with desired properties accounting for specific constraints.[11]

Another work addresses the design of polymers with optimized macroscopic properties using topological connectivity indices and an optimization framework.[10]

Mavrantzas and co-workers used atomistic simulations and computer aided molecular design techniques based on group contribution models for the design of polymers with desired properties.[6] Similarly, Ng and coworkers presented an approach for a polymer based inverse design problem, where they first identified suitable repeat units via group contribution methods and then carried out molecular dynamics at various molecular weights.[12]

Another study reports a molecular design approach, using a combination of group contribution and quantum chemical descriptors, where a genetic algorithm was used to screen candidate molecules.[13]

The group of Lin developed a method for automated molecular design using the octanol water partition coefficient as target.[14] They were using an activity coefficient model and a genetic algorithm to identify suitable molecular structures having a defined partition coefficient.

Another approach that relates to the systematic design of biochemicals and the identification of respective biomass conversion pathways was developed by Ng *et al*. and applied to the case of biobased fuel production of palm-based biomass.[15] They combined group contribution and topological indices-based methods for property prediction and molecular signature descriptors for solving the structural design problem.

In industrial processes, chemical systems often correspond to mixtures of oligomeric or polymeric species. Therefore, properties and descriptors follow a statistical distribution, such as for the molecular weight in the simplest case. In industrial practice, the modification of this statistical distribution is often one of the few available levers to adjust processes. However, this important aspect is often missing in inverse design approaches.

Some attempts have been made to address this issue for simulations, for example using random branching to take polydispersity into account for molecular dynamics,[16] however, not in an inverse design context and without consideration of detailed kinetics. Marvin *et al.* take into account chemical reactivity by defining a reaction network and solve an optimization problem that yields an optimal product distribution for gasoline blends.[17] Recently, Cravero *et al.* have proposed a QSPR approach demonstrating the relevance of taking descriptor polydispersity into account.[18] To introduce polydispersity, they directly operate on SMILES strings to obtain different polymer chains, *e.g.,* keeping the number or weight averaged molecular weights constant, without using an explicit kinetic model.

In the field of lignin research, a few approaches have been developed which systematically generate lignin oligomer and polymer structures by kinetic Monte Carlo simulation,[19] or other Monte Carlo approaches,[20-21] to match several structural quantities derived from experiments. The idea is to get representative computational models / structure libraries for the rather complicated, source dependent lignin compositions. These methods share some similarities with this study, but they focus on the unique chemistry and materials associated with lignin, which cannot be readily be generalized.

Recently, another approach has been published enabling the generation and sampling of molecular species and respective SMILES using the (generative) BigSMILES notation.[22-23] It samples from a previously specified theoretical distribution (top-down), and not from explicit kinetic equations (bottom-up), and is less well suited for solving inverse design problems of industrial relevance.

In general, with regard to classical polymers, systems are usually assumed to be monodisperse in an inverse design context. This may be a valid approximation for high molecular weight systems but usually leads to problems for smaller molecular weights, where for example the effect of functional end-groups is non-negligible. Detailed kinetics for oligomer or polymer generation are usually out of scope and very simplified molecular topologies are used for descriptor or property prediction.

Therefore, in the following work, a broadly applicable approach is being proposed. Reactive systems are addressed by running a kinetic simulation using Gillespie's algorithm[24] generating discrete species. Polymers are accounted for by using a network free version of this algorithm.[24-25] Those kinds of models have been used in the systems biology community for a long time and can similarly be used for an organic polymer context.[26-27] The finally obtained reactive mixtures containing explicit oligomeric and polymeric species are used to compute descriptors, *e.g.,* either as mean values or, if necessary, as higher statistical moments.

To go beyond descriptors and properties that are derived exclusively from the topological graphs produced by the kinetic Monte Carlo simulations, discrete molecules are generated in the form of SMILES strings[28] for each species. This opens the possibility to use cheminformatics, group contribution-based methods, molecular dynamics, computational chemistry and/or solvation thermodynamics to compute descriptors and properties from the molecular distribution. This is also an important improvement compared to the derivation of descriptors from plain monomers with unreacted functional groups, which are chemically very different when being incorporated into the polymer system.

In the present work, a mapping from a certain chemical recipe, *e.g.,* the input to the kinetic simulation in the form of specific reactive compounds and their stoichiometry, towards a unique fingerprint of different descriptors is obtained.

Coupled with an optimization procedure, the approach allows for variation of the recipe to match either some pre-defined descriptors or an alternative recipe based on different ingredients. This is achieved by minimizing the distance in descriptor space between a reference and the new system to be optimized. Assuming that the descriptors correlate with macroscopic properties, this approach generates different recipes while keeping the product properties constant or generates a recipe for a specific set of properties. As a prerequisite, the kinetics, including the reaction channels and the respective (relative) rate constants have to be known at least approximately. By variation of the initial recipe, instead of the molecular structure, it is possible to address industry relevant processes, such as the *drop-in* replacement of chemicals that have to be substituted due to economic, technical or regulatory reasons. In contrast, proposals of classical CAMD often suggest novel molecular entities *i.e.,* require the synthesis of novel, non-readily available molecules which may be out of scope due to economic or technical reasons. Just modifying a known recipe is a simpler but more pragmatic approach, at least outside the field of drug development and design.

In summary, a workflow is proposed starting with the generation of a discrete molecular weight distribution using a defined recipe and kinetics. A subsequent mapping into discrete molecular species allows for the computation of diverse molecular descriptors. Finally, the recipe is iteratively improved by a global optimization procedure minimizing the distance in descriptor space.

1 Generating Chemical Recipes

The central idea of the approach is to optimize a chemical recipe, as defined by several molecular components, their stoichiometry and their reaction equations including the kinetic parameters. However, instead of optimizing a certain property, the distance (or similarity) in multi-dimensional descriptor space to another recipe is optimized instead. This has several advantages: First, some amount of error cancellation is to be expected, as the descriptors of the target and the new recipe are computed by the same methodology and the same kind of errors are made during their determination. Secondly, in principle, no experimental data is needed for such an optimization, which may be at times difficult or costly to obtain in practice. Of course, optimizing a certain property instead of descriptors is still possible, as often there is no clear separation between a descriptor and a property. However, prerequisite is, that this property or descriptor can be modelled sufficiently accurately from the polymer topology and / or the molecular structure. For the computation of descriptors, the mapping of the species resulting from the kinetic simulation to the atomistic level, in particular SMILES notation, is crucial, as this allows to compute additional relevant properties using other simulation tools.

1.1 General Workflow

The key objective of the general workflow is to substitute raw materials within a polymer recipe, for example, due to regulatory or economic reasons, without compromising on the properties in a subsequent application. The general concept of the workflow is depicted in Figure 1. A concrete example is given in Section 2. Usually, the first step is to select a target or reference recipe which is subject to a modification of its ingredients.

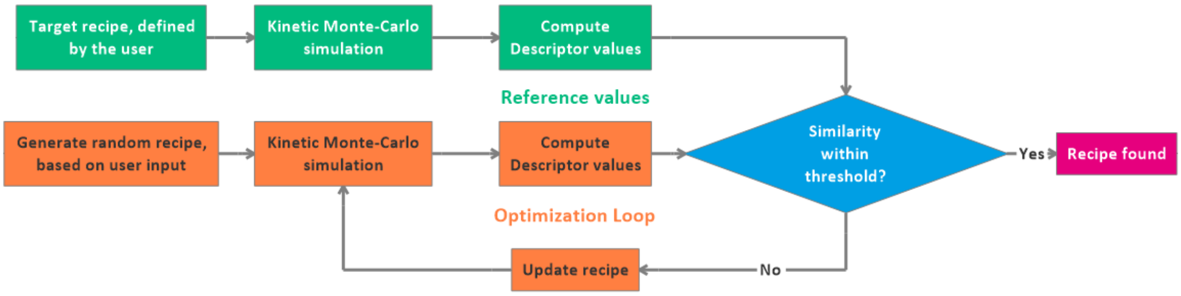

Target recipe, defined by the user
Kinetic Monte-Carlo simulation
Compute Descriptor values
Reference values
Generate random recipe, based on user input
Kinetic Monte-Carlo simulation
Compute Descriptor values
Similarity within threshold?
Yes
Recipe found
Optimization Loop
Update recipe
No

Figure 1: Schematic representation of the workflow, starting from a kinetic simulation and descriptor calculations for a reference recipe and a subsequent optimization process to match the target descriptors with an iteratively modified new recipe.

Then a kinetic Monte Carlo simulation is carried out based on the available data on the reaction mechanism and the (relative) rate constants of the system under investigation. A set of meaningful descriptors $\mathbf{X}^{\mathrm{ref}}$ is then computed for the reference system and stored. The basis for the descriptor calculation is the result from the kinetic simulation (Section 1.3) and in particular the transformation of the polymer graphs into discrete atomistic species (Section 1.5). Details on the available descriptors are given in Section 1.6. Alternatively, a set of reference descriptors can be specified directly, without simulating a reference system.

Once the descriptor set is defined, new recipes are generated systematically by modifying the ingredients and their stoichiometries. A set of alternative ingredient combinations is pre-defined by the user. Also, the underlying chemical reactions could be subject to a change and involve for example different polymerization mechanisms (see Section 2.3), however, in practice most often the chemistry remains unchanged. For the modified recipe, the same set of descriptors $\mathbf{X}^{\mathrm{mod}}$ is now computed and compared with the reference set. As a similarity metric a variant of the relative error of the two descriptor sets is used, the symmetric mean absolute percentage error (SMAPE):

$$SMAPE = \sum_{i=1}^{n} \frac{\left|x_i^{ref} - x_i^{mod}\right|}{\left(\left|x_i^{ref}\right| + \left|x_i^{mod}\right|)/2\right)}.$$

We opted for the SMAPE as the preferred metric over Mean Absolute Percentage Error (MAPE) due to its ability to handle situations with zero values in a more robust and balanced manner. SMAPE's symmetry in evaluating both overestimation and underestimation errors provides a more accurate reflection of prediction accuracy, making it a more suitable choice for our analysis. An update of the new recipe is then carried out iteratively using a global optimizer as explained in Section 1.7. As soon as the two descriptor sets are sufficiently similar, the process is stopped, and a final recipe is proposed. Based on different settings or starting conditions, several diverse recipes can be obtained in this way and subsequently be handed over to experimentalists for further validation and refinement.

1.2 Implementation Details

The overall workflow is implemented as a python library in a modular fashion and a few design choices making the code very versatile are highlighted. A class representing a recipe holds all information about the recipe itself, like the ingredients and the corresponding molar or weight fractions. Furthermore, the kinetic Monte Carlo (kMC) simulation for the recipe can be started from an instantiated object of this class. Second, the result of the kMC simulation is stored in a different class. This class holds the ensemble of molecular species in a Pandas[29] DataFrame, represented by their BNGL string (Section 1.4) and corresponding kMC graph (Section 1.6.3). As a unique identifier of the molecular species the Weisfeiler Lehman graph hash of the kMC graph is used.

It is possible to compute the set of descriptors from an instance of this class / class object. Furthermore, these objects can be used as ingredients for the recipe class, making the simulation of *n*-step synthesis routes or blending possible.

Another advantage of the modular code design is that the optimization problem can easily be adapted. For example, the library was originally designed to optimize recipes regarding their properties from known kinetics. However, it is also possible to optimize relative rate constants towards recipes with experimentally determined properties and known reaction channels.

1.3 Kinetic Monte Carlo Simulation

In contrast to deterministic approaches that address chemical kinetics via a set of coupled ordinary differential equations, kinetic Monte Carlo (kMC) simulations allow for the time evolution of discrete chemical species by using random processes. Both approaches possess their own advantages and disadvantages,[30] but in this context, it is important that the discrete species of the kMC process can be converted to an atomistic representation for descriptor calculations, see also Section 1.4.

Those algorithms were pioneered by Gillespie[24] about half a decade ago and have been applied for stochastic simulations of chemical kinetics in many different fields of chemistry and biology since then. In its original formulation, the algorithm requires the enumeration of all possible reactions and its associated reaction constants, which is, however, not feasible for complex reaction networks as seen in polymers.[25, 31]

Most advancements in this direction have been reported in the field of radical polymerization[31] and systems biology.[25] In this context we use the NFsim software package which implements a network-free variation of Gillespie’s stochastic simulation algorithm (SSA). It is designed for complex biochemical reaction networks, but due to its generic and efficient implementation scheme can also be used for classical polymer chemistry.[26] In principle it's also possible to switch to different (e.g., network based) numerical solvers in the BNGL context.

The applicability of this simulation package for the use cases in this work has been thoroughly validated by comparison with results from experiments, in-house reference implementations and classical analytical methods available for polyaddition and polycondensation reactions.[32-33]

1.4 Chemistry in BioNetGen Language

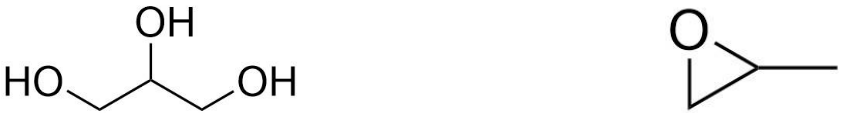
The NFsim software package employs the rule based BioNetGen Language (BNGL).[34-35] In this section, it is described how molecules and functional groups are expressed in the BNGL notation. Then, implementation details for the representation of chemical reactions in the BNGL notation are described in Section 1.4.2.

1.4.1 Representation of Oligomers and Chemical Reactions in BNGL Notation

In Table 1 the BNGL notation terms are mapped to the chemical terms used in this paper. Here, a BNGL molecule represents one repeat unit. As an example, it is shown how a typical chain growth polymerization reaction, a polyether polyol formation via alkoxylation of an alcohol starter with an epoxide, can be represented. Our notation for an exemplary starter, glycerol, and an epoxide monomer, propylene oxide, is shown in Figure 2. The repeat unit name is abbreviated (GLY and PO) and followed by a list of the reactive functional groups as BNGL components.

Table 1: Mapping of BioNetGen Language (BNGL) terms[35] to the chemical terms used in this paper.

BNGL notation	This work
Molecule	Repeat unit or monomer
Bond	A connection between repeat units / monomers
Component	Reactive, functional groups of the repeat unit / monomer
State	Functional group classification or activation
Complex	Oligomer / polymer species



GLY(oh,oh,oh)

PO(epo,oh~0)

Figure 2: Representation of the glycerol (left) and propylene oxide (right) monomer in the BNGL notation.

The KOH catalyzed alkoxylation reaction of the alcohol is schematically shown in Figure 3. During the reaction, the epoxide ring is opened, and a hydroxy group is created. Potential side reactions are neglected for the moment. The hydroxy group can further react with another epoxide. To represent this behavior, the propylene oxide monomer has two components. The first component (epo) represents the unreacted epoxide group which can form a bond to the hydroxy group of an alcohol. The second component (oh) represents the hydroxy group which is formed during the reaction. This hydroxy group carries a state where 0 means that the hydroxy group has not been formed, *i.e.,* the epoxide ring did not react. After the reaction of the epoxide group, the hydroxy component is activated, and the state is changed to 1. The corresponding reaction rules in BNGL notation are shown in Listing 1.

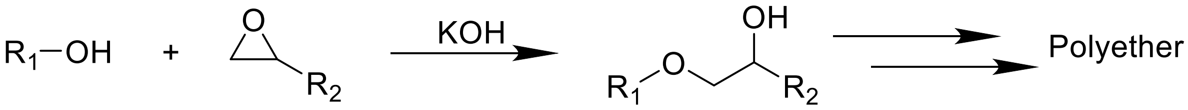


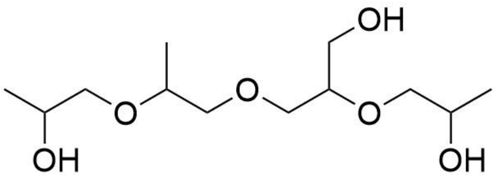
Figure 3: Alkoxylation reaction of an alcohol with an epoxide.

```
begin reaction rules
      GLY(oh) +  PO(epo,oh~0) -> GLY(oh!1).PO(epo!1,oh~1)    1.0
      PO(oh~1) + PO(epo,oh~0) -> PO(oh!1~1).PO(epo!1,oh~1)   1.0
end reaction rules
```

Listing 1: Reaction rules for the alkoxylation of glycerol.

The first reaction rule describes the initial reaction of glycerol with propylene oxide. During this reaction the hydroxy component gets activated. Afterwards, the chain can be elongated by a reaction of the activated hydroxy group with another propylene oxide, which is described by the second reaction. Here, it is assumed for the sake of simplicity that both reactions have the same relative rate constants.

One exemplary polyol species (or complex in the BNGL notation) resulting from this reaction is shown in Figure 4. The bonds between the monomers via the functional groups are indicated with an exclamation mark followed by a bond number (integer), which essentially correspond to an edge list data structure in graph theory. Furthermore, the monomer units within the polyol are separated by dots, this will be revisited in Section 1.5.



GLY(oh!1,oh!3,oh).PO(epo!1,oh~1!2).PO(epo!3,oh~1).PO(epo!2,oh~1)

Figure 4: Representation of a polyol species in the BNGL notation.

1.4.2 Classification of Functional Groups

Until now, we assumed that all three hydroxy groups of the glycerol react with the same relative rate constant. However, glycerol contains two primary and one secondary hydroxy group, as shown in Figure 2. Compared to the primary hydroxy group, the reaction of the secondary hydroxy group with the epoxide is hindered by steric effects. Thus, the relative rate constant has to be smaller for the reaction of the secondary hydroxy group.

In this work, we use states of the BNGL notation to represent the classifications (primary, secondary or tertiary) of the functional groups: e.g., GLY(oh~p,oh~p,oh~s) for glycerin (see also Figure 1) and PO(epo,oh~0) for propylene oxide. Please note that the functional groups are ordered alphabetically during the NFsim workflow. Therefore, the position of identical labeled functional group components in the monomer is arbitrary and those should represent symmetric equivalent functional groups within the molecule. Furthermore, the reaction rules have to be adapted as well. Here, we assume that the secondary hydroxy groups react 2.5 times slower with the epoxide than the primary hydroxy groups. In Listing 2 the reaction rules incorporating different relative rate constants for primary and secondary hydroxy groups are shown.

```
begin reaction rules
      GLY(oh~p) +  PO(epo,oh~0) -> GLY(oh~p!1).PO(epo!1,oh~1)    1.0
      GLY(oh~s) +  PO(epo,oh~0) -> GLY(oh~s!1).PO(epo!1,oh~1)    0.4
      PO(oh~1) + PO(epo,oh~0) -> PO(oh~1!1).PO(epo!1,oh~1)       0.4
end reaction rules
```

Listing 2: Reaction rules for the alkoxylation of glycerol incorporating different relative rate constants for primary and secondary hydroxy groups.

1.4.3 Reaction Templating

Writing out all reaction rules for a system can be quite tedious and error prone since all combinations of reacting monomers have to be built. Furthermore, we also have to write out the reaction rules for all functional group classifications of the monomers. As an example, a system consisting of three different alcohols and two different epoxides would already need nine different reaction rules to be fully described, if all monomers had only one functional group classification. Therefore, reaction templates are used for all implemented reactions which allows for an automatic build of all monomer permutations. For the reaction discussed above, the reaction templates are shown in Listing 3.

```
{alcohol}(oh) + {epo}(epo,oh~0) -> {alcohol}(oh!1).{epo}(epo!1,oh~1)
{epo_1}(epo,oh~0) + {epo_2}(oh~1) -> {epo_2}(oh!1~1).{epo_1}(epo!1,oh~1)
```

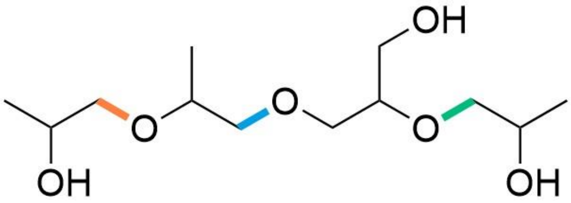
Listing 3: Reaction rule templates.

Within the implemented workflow, every monomer is assigned to one or multiple molecular classes. The template reactions above contain these molecular classes within the curly brackets. When all combinations of the reactive monomers should be built, an internal function creates all permutations of the reactive monomers and inserts them into the template rules at the appropriate places. Afterwards, the subsequently created rules are written out for all combinations of functional group classifications from the reacting monomers.

1.5 Generation of Molecular Representations from Polymer Graphs

This section explains how the molecular structure from the BNGL notation are extra after the kinetic Monte Carlo simulation has been performed and the molecular species have been obtained.

Section 1.4 introduced the representation of monomers, functional groups and oligomers in the BNGL notation. Looking at the BNGL notation of a polyol species, it can be recognized that the structure is basically a graph edge list based on the functional groups, storing additional references to the parent repeat units. The simplified molecular-input line-entry system (SMILES) notation allows for some rather untypical reformulation, using ring labels being in fact very similar to the edge list notation of BNGL, see Figure 5. Instead of being used conventionally for separating compounds, disconnecting dots can be used to separate fragments of a molecule (corresponding to the monomer repeat units) and bonds are explicitly re-introduced by using integer based "(ring) closures" using the numbering from the BNGL notation. This approach for rewriting polymer SMILES was mainly inspired by an algorithm for the arbitrary reordering of atoms in a SMILES string.[36]



BNGL notation	`GLY(oh!1,oh!3,oh).PO(epo!1,oh!2).PO(epo!3,oh).PO(epo!2,oh)`
SMILES	`O1CC(O3)CO.O2C(C)C1.OC(C)C3.OC(C)C2`
Canonical SMILES	`CC(O)COC(C)COCC(CO)OCC(C)O`

Figure 5: Conversion of BNGL to SMILES string. The bonds between the monomers are color coded according to the bond number in the BNGL and SMILES string. For simplification, the states of the functional groups have been omitted in the BNGL string.

Using this reformulation, BNGL and SMILES syntax become very similar and repeat units can be replaced very efficiently using simple regular expressions, which are defined for every implemented monomer. This allows also for a simple way of taking into account unreacted groups. An exemplary regex pattern and the corresponding SMILES string is shown in Listing 4 for the glycerol monomer. The regex pattern captures the bond numbers of the functional groups in the BNGL string and inserts them into the appropriate positions of the SMILES string. After this conversion, we obtain a SMILES string with explicitly labeled connections that can then be used to be fused with neighboring monomer repeating units.

Finally, the resulting string can then be converted to a canonical SMILES or into an InChI string (Figure 5).

```
Regex pattern: GLY\(oh~p!*(\%?\d*),oh~p!*(\%?\d*),oh~s!*(\%?\d*)\)
SMILES: O\1CC(O\3)CO\2
```

Listing 4: Regex pattern to replace the BNGL glycerol string with the corresponding SMILES string.

This transformation presents a very general scheme for SMILES generation that can as well be applied to other polymerization mechanisms, for example polyaddition and polycondensation reactions. This approach has the big advantage that it allows for a very efficient implementation at the string level and no large molecular objects need to be created and manipulated.

1.6 Descriptors

This section gives a brief overview of the types of descriptors used in this work.

1.6.1 Basic Descriptors

First, a class of basic descriptors is described which can directly be computed from the BNGL species string by knowing the molar mass of the monomers. The molar mass of the polymer species can be computed conventionally as the sum over the molar masses of the monomers:

$$M_{\text{Species}} = \sum_{i=1}^{N_{\text{Species}}} M_{\text{Monomer},i} - \sum_{j=1}^{N_{\text{Bonds}}} M_{\text{Condensation},j}\,.$$

For condensation reactions, the sum of the molar masses of the by-product has to be subtracted for all bonds formed by the condensation reaction. As an alternative, the molar mass of the molecular species can directly be obtained from the SMILES string, if the transformation as outlined in Section 1.5 was carried out.

By having the molar masses of the individual molecular species available, the molar mass distribution of the simulated recipe can be investigated as well. This, for example, allows for the simulation of gel permeation chromatography (GPC) spectra, for more detailed information we refer to Ref. [37].

Descriptors based on the molar mass and certain functional groups, like the OH number (OH#), can also be obtained directly from the evaluation of the species string. Additionally, descriptors may also be defined by using the option to define observables in the bngl-syntax.

1.6.2 RDKit Descriptors

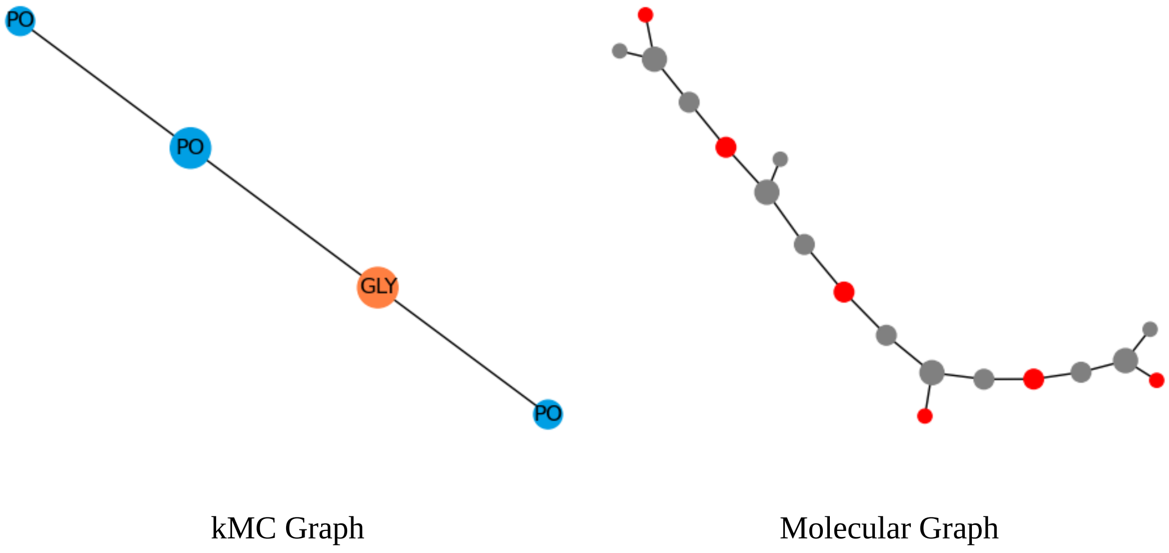
Using SMILES strings (Section 1.5), it is possible to compute various 2D-QSPR descriptors and properties with a cheminformatics toolkit like RDKit.[38] Examples are the octanol-water partition coefficient (logP), the number of rotational bonds or the topological polar surface area (TPSA). Furthermore, substructure patterns within the molecular species can be counted using SMILES arbitrary target specification (SMARTS) patterns.

To compute 3D-QSPR descriptors, it is needed to generate a proper 3D structure for the oligomers. Currently, RDKit's "EmbedMolecule" method is used to generate an initial 3D structure. This 3D geometry can then be optimized using a molecular mechanics force field or an efficient quantum chemical method and descriptors, like the radius of gyration or HOMO / LUMO gaps, can be obtained.

1.6.3 Topological Indices

Using the BNGL string, graphs can be built, where each node represents a monomer and the edges represent the bonds between these monomers, which essentially corresponds to a graph representation via an edge list. Each node holds the name of the monomer and the reactive functional groups including the bond number ("!x") as an attribute. Furthermore, the edges may hold the functional groups forming the respective connection as an attribute. In the following, we will call these graphs "kMC graphs".

Alternatively, SMILES strings can be used to build up graphs where each node represents an atom, and the edges are the covalent bonds between the atoms. These graphs will be called molecular graphs in the following.



kMC Graph

Molecular Graph

Figure 6: Representation of a kMC graph (left) and a molecular graph (right). On the left graph the nodes are colored as: orange: glycerol monomer block, blue: propylene oxide monomer block. On the right graph the nodes are colored as: gray: carbon, red: oxygen.

Using these graphs, it is possible to compute topological indices, like the Wiener index or the Balaban index. For the computation of the Wiener index for acyclic and unicyclic graphs, we implemented the linear time algorithm published in Ref .[39]. Otherwise, the standard algorithm provided by NetworkX[40] is used.

1.6.4 Descriptor Averages

Until now descriptors computed on the level of the individual species are discussed. In addition, descriptor values for the whole ensemble of molecular species by averaging the molecular descriptor values are accessible. Multiple weight functions can be employed here, the most common ones are the number (Dn) and weight average (Dw) of a descriptor D:

$$Dn = \frac{\sum_{i=1}^{N_{\text{Species}}} n_{\text{Species},i} \cdot D_{\text{Species},i}}{\sum_{i=1}^{N_{\text{Species}}} n_{\text{Species},i}},$$

$$Dw = \frac{\sum_{i=1}^{N_{\text{Species}}} n_{\text{Species},i} \cdot M_{\text{Species},i} \cdot D_{\text{Species},i}}{\sum_{i=1}^{N_{\text{Species}}} n_{\text{Species},i} \cdot M_{\text{Species},i}}.$$

Here, N_{Species} is the number of unique molecular species, $n_{\text{Species},i}$ the count, $M_{\text{Species},i}$ the molar mass, and $D_{\text{Species},i}$ the descriptor of species i.

1.6.5 Other descriptors

Since the library serves as an interface to other tools and tool chains, it is possible to incorporate descriptors from various tools providing e.g., quantum chemical calculations or liquid phase thermodynamics computations. Furthermore, it is also possible to generate input for a Molecular Dynamics Simulation, either on an all atomistic or a coarse-grained resolution.

Other descriptors for the ensemble of molecular species can be the standard deviation or Kullback-Leibler divergence to a reference distribution of a descriptor since the distribution of descriptor values over the molecular species is available within the library.

1.6.6 Selection of Descriptors

As discussed in the sections above, a wide range of descriptors have been implemented. In fact, the selection of a suitable set of descriptors for a given use case is challenging and to some degree arbitrary. On the other hand, including descriptors concerning the molecular weight distribution, the topology, the cross-linking characteristics and, for example, the polarity seems very reasonable. However, the set used in a simulation should be selected individually for each use case. For example, for a typical polyol application, the OH number, the functionality should be included as descriptors. Furthermore, some descriptors are highly correlated, e.g., Mn (number average molecular weight), Mw (weight average molecular weight), and Mz (z-average molecular weight), as well as graph indices computed on the kMC graph and molecular graph are highly correlated (note the discussion in the SI section 1.1). Therefore, unsupervised feature / descriptors selection procedures such as inspection of the descriptor covariance matrix may be applied.

1.7 Optimizer

For the optimization procedure mainly a Bayesian optimization approach as implemented in the optuna python library is used.[41] Bayesian optimization is a powerful method for the optimization of an objective (black box) function that is expensive to evaluate. It works by constructing and updating a probabilistic model of the objective function, a so-called surrogate model, during the optimization. New data points are selected using an acquisition function that manages the balance between exploration and exploitation of the parameter search space. The surrogate model and the acquisition function are updated as new data becomes available, allowing the algorithm to focus on the most promising regions of the search space. Bayesian optimization has several advantages, including the ability to handle noisy and non-convex objective functions, and the ability to find global optima with relatively few evaluations without the need for gradients. It is also possible to handle categorical data which is particularly well suited for the variation of chemical ingredients. The optuna library uses the Tree-Structured Parzen Estimator model (TPE), a variant of Bayesian optimization but also other implemented samplers and optimizers have proven to be helpful in some cases. For optimal performance, benchmarking is recommended for a specific optimization problem. We provide such an example for a polyalkoxylation recipe in SI section 1.4.

Furthermore, multi-objective optimization is supported. For details on the algorithms and implementations we refer to the original publication.[41]

2 Examples

2.1 Computational Details

All kMC simulations are performed using NFsim v1.12.1[26] with a simulation length of 0.5 s. During the recipe optimization the total number of start molecules is set to 10,000. For the property prediction of the reference polymers and the proposed recipe 100,000 initial molecules in the kMC simulation were used. The basic descriptors Mn, Mw, Mw/Mn, OH# and their standard deviations (where applicable) are computed based on the molar mass information of the monomers, as described in Section 1.6.1. logP values are obtained by inserting the SMILES string of the molecular species into RDKit 2023.03.2,[38] see also Section 1.6.2. The Wiener index is computed on the kMC graph of the molecular species, which is described in Section 1.6.3. Internally, graphs are built via Networkx 3.1.[40] To optimize alternative recipes, the optuna 3.3.0[41] library is used with the TPE sampler. Here, the ingredient lists are described as a categorical parameter while the concentrations of the ingredients are continuous parameters, as already described above.

2.2 Raw Material Substitution for a Polyester System

For demonstration purposes of the algorithm a hypothetical reference recipe was set up with the following chemicals and the respective mass fraction in parenthesis: trimethylolpropane/TMP (0.5), 1,2-propylene glycol/PG (0.1) and isophthalic acid/ISOPS (0.4). The system is subject to a simple polyesterification reaction, including the generation of 1 molecule water in each reaction step (Figure 7). The kinetic rules for NFsim are defined accordingly (Listing S1). For the sake of simplicity any trans-esterification reaction is omitted in this model. As TMP has been labeled reprotoxic according to REACH, and the polyester produced with the aforementioned recipe contains residual amounts of monomeric TMP, the objective of this toy example is to replace TMP by an alternative alcohol. The change of overall polyester characteristics (as given from the descriptor similarity as described in Section 1.1) should be kept at a minimum. First, the characteristics for the reference system are computed.

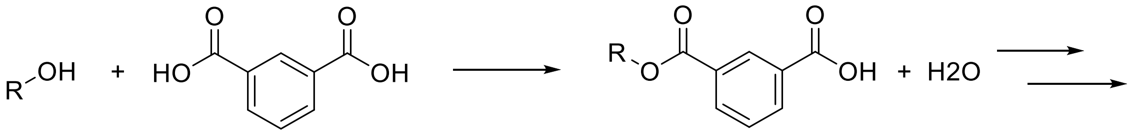


Figure 7: Esterification reaction of an alcohol functional group with isophthalic acid leading to a mixture of oligomeric and polymeric polyester structures.

TMP is defined to have three primary alcohol groups, propylene glycol one primary and one secondary alcohol group, and isophthalic acid to have two carboxylic acid groups. A kinetic Monte Carlo simulation is set up where the primary alcohol groups react with carboxylic acid groups with a relative rate constant of 1, and secondary alcohol groups slower with a relative rate of 0.4. It is assumed that the reaction water is removed by distillation and the esterification equilibrium is shifted completely to the ester side. The simulation is carried out using 100,000 initial seed molecules (monomers in this case) and the graphs of the polymer and molecular species are generated according to the workflow described above.

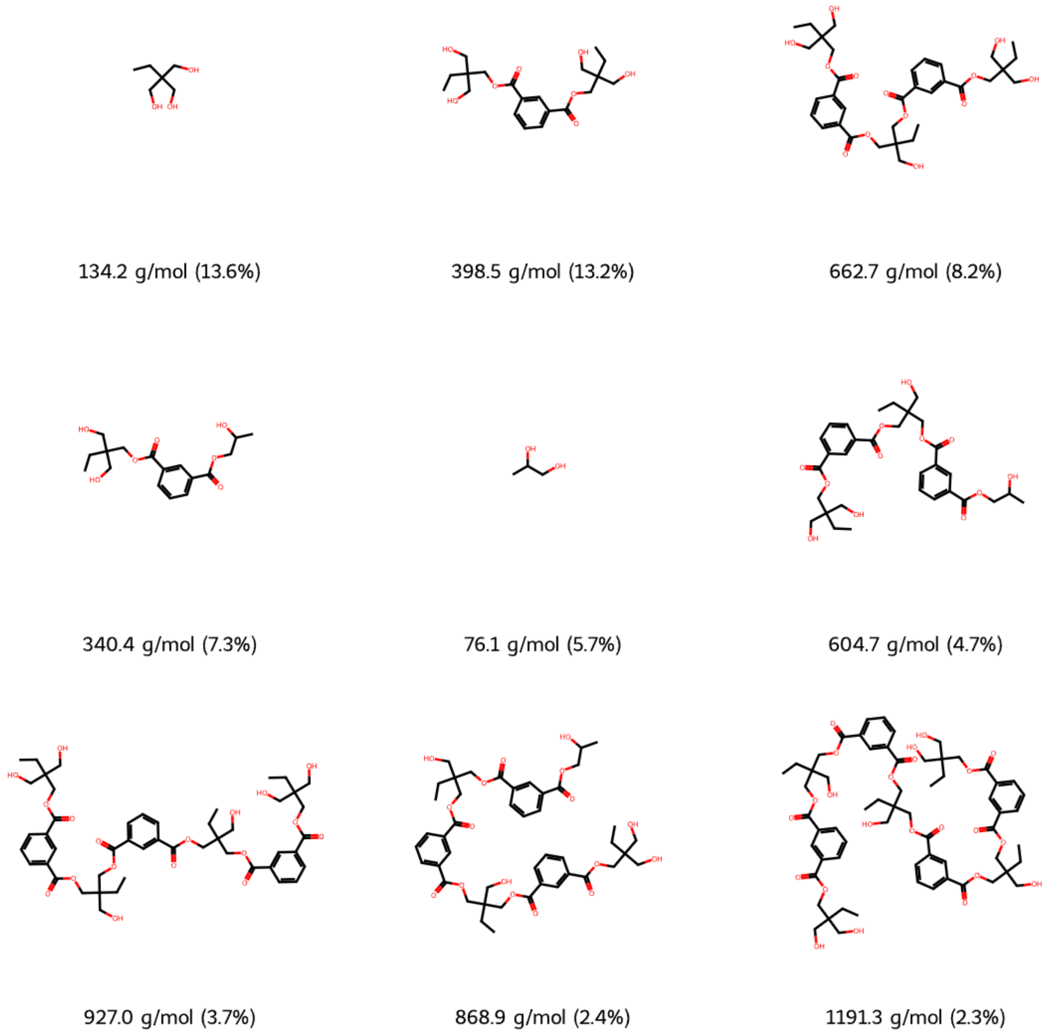


Figure 8: Most frequent oligomeric species (ordered by weight fraction) as created by the kinetic Monte Carlo simulation for the reference system. The molar mass and weight fraction of the individual species are shown below each species. Molecular structures have been obtained by mapping the polymer graphs into molecular graphs *after* the simulation as explained in Section 1.5

Figure 8 shows the most important oligomeric species that have been produced by the simulation. From this result, different descriptors have been computed: The OH number (OH#), as a measure for the hydrogen bond density, the Wiener index representing the overall topology, the average functionality of OH groups (fn(OH)), and the average octanol-water partition coefficient (logP). Note that the average logP is used here merely as a descriptor related to the overall polarity of the system, not as a real physical property that could be determined experimentally. The computed descriptors of the reference system are shown in Table 2.

Table 2: Showing the computed descriptors for the reference and the new proposed recipe. The monomers are abbreviated as trimethylol propane (TMP), propylene glycol (PG), isophthalic acid (ISOPS), hexandiol (HXD) and neopentylglycol (NPG).

System	OH#	Wiener Index	logP	fn(OH)	monomers
Reference system	552	52	0.6	3.4	TMP, PG, ISOPS
Proposed system I	530	57	0.3	3.4	GLY,HXD,ISOPS
Proposed system II	507	62	0.2	3.3	GLY,NPG,ISOPS

Table 3: Computed observables for the reference and the new proposed recipe. The monomers are abbreviated as trimethylol propane (TMP), propylene glycol (PG), isophthalic acid (ISOPS), hexandiol (HXD) and neopentylglycol (NPG).

System	Mn	Mw/Mn	TPSA	#rot. bonds	Monomers
Reference system	346	3.13	117	31.9	TMP, PG, ISOPS
Proposed system I	358	2.59	127	27.8	GLY,HXD,ISOPS
Proposed system II	366	2.63	129	25.1	GLY,NPG,ISOPS

In the next step, a new reactive system is defined, with different starters, and particularly, omitting the TMP molecule: glycerol, 1,2-propylene glycol, hexanediol, neopentylglycol, triethylene glycol and isophthalic acid. The same set of kinetic rules was defined as for the reference system. Starters were grouped in different recipe sets with each set having a maximum of four different alcohols, to make the recipe not too complex.

The Bayesian optimization workflow is then carried out as described in Section 1.1, where the different sets were treated as categorical variables and the concentrations as continuous variables. The same set of descriptors was computed for the test system, and the stoichiometries of the test system were updated by the optimizer, until the distance in the descriptor space between the two reaction systems was minimal. The final set of descriptors for two recipes with the highest similarity is shown in Table 2. In Table 3 some additional descriptors are shown which are reasonably close to the reference but have not been used for optimization, such as the number average molecular weight, the polydispersity index (Mw / Mn) and average topological surface area (TPSA) and the number of rotatable bonds (NRB) as computed from RDKit.

As a result, those recipes most similar to the reference system, have the following chemical composition in mass fractions: (0.32, 0.21, 0.47) for the system GLY, HXD, ISOPS, and (0.30, 0.22, 0.48) for the system GLY, NPG, ISOPS, and serve as a rational starting point for focused lab trials.

2.3 Finding an Alternative One-Pot Synthesis Route

The presented approach is also able to represent multi-step synthesis routes, which is highlighted on another hypothetical use case. Here, the reference recipe is a blend of two polyesters. The synthesis route is depicted in Figure 9. One polyester is obtained through a polyesterification reaction of adipic acid (ADPS) with the alcohols ethylene glycol (EG) and 1,6-hexanediol (HEXD), see Figure 10. To simplify the model, trans-esterification reactions are neglected. For the same reasons, it is assumed that the reaction of primary and secondary OH groups can be described with the same rate constants. The second polyester is based on caprolactone (CPL) where the ring is opened with 1,4-butanediol (BDO) and 1,2-dipropylene glycol (DPG), see Figure 11. Please note the very different underlying reaction mechanisms, *i.e.*, step growth polycondensation and the chain-growth-based ring opening reaction. Blending these two polyesters together yields the final target product.

In this hypothetical example the aim is to replace this polyester polyol with a polyether polyol. Furthermore, a one-pot synthesis route for the polyether polyol is desirable. In addition, the final recipe should have a similar Mn value as well as (mean) OH number. Moreover, a similar OH number *distribution* will result in a similar crosslinking functionality later.

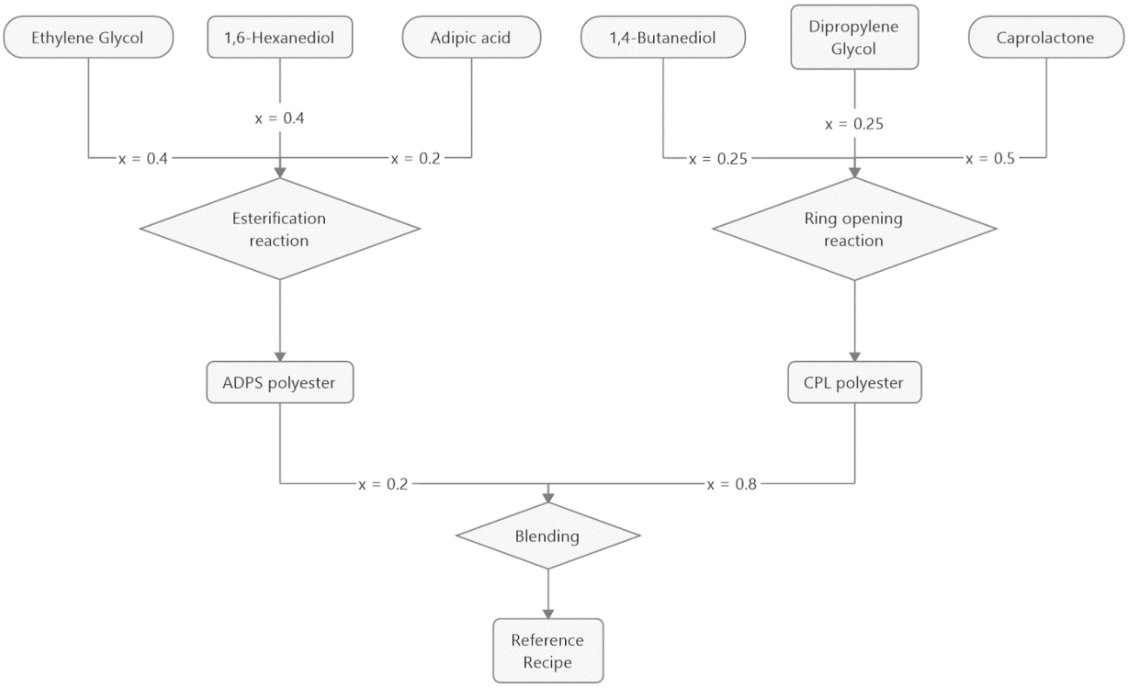


Figure 9: Hypothetical synthesis route of the reference polyester polyol blend, x as mol fractions.

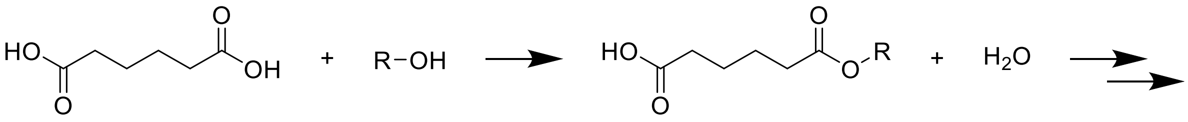


Figure 10: Esterification reaction of adipic acid with an alcohol.

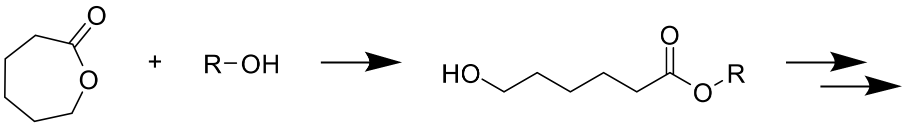


Figure 11: Ring-opening reaction of caprolactone with an alcohol.

2.3.1 Results

The molecular species ensemble of the polyester polyol blend has to be predicted to obtain the target descriptor values for the subsequent optimization. To achieve this, the ensemble of molecular species is computed for every step (reaction and blending) shown in Figure 9. Each reaction step is represented by a kMC simulation while a blending step joins the ensembles of molecular species according to their mole fractions. The nine most frequent molecular species of the final polyester polyol blend are depicted in Figure 12.

Next, the possible monomer combinations for the polyether polyol recipe are created. For this, we set up a list containing four different starter molecules and a list with two different epoxides, see Figure 13. Subsequently, all possible monomer combinations containing two starter monomers and both epoxides are built. This yields 6 different monomer combinations the optimizer can choose from. For the polyol reaction a simple chain growth mechanism is assumed (i.e. epoxide ring opening via OH functional groups) and side reactions during KOH catalysis such as monol formation are neglected, see also **Figure 3**.

In this example, the average and standard deviation of the OH number, Mn and the averaged value of the Wiener Index are used as descriptors. All descriptors are weighted equally during the optimization. As a metric for the loss, the symmetric mean absolute percentage error (sMAPE) is used. By optimizing the recipe towards the average and standard deviation we aim to yield a similar distribution of OH numbers.

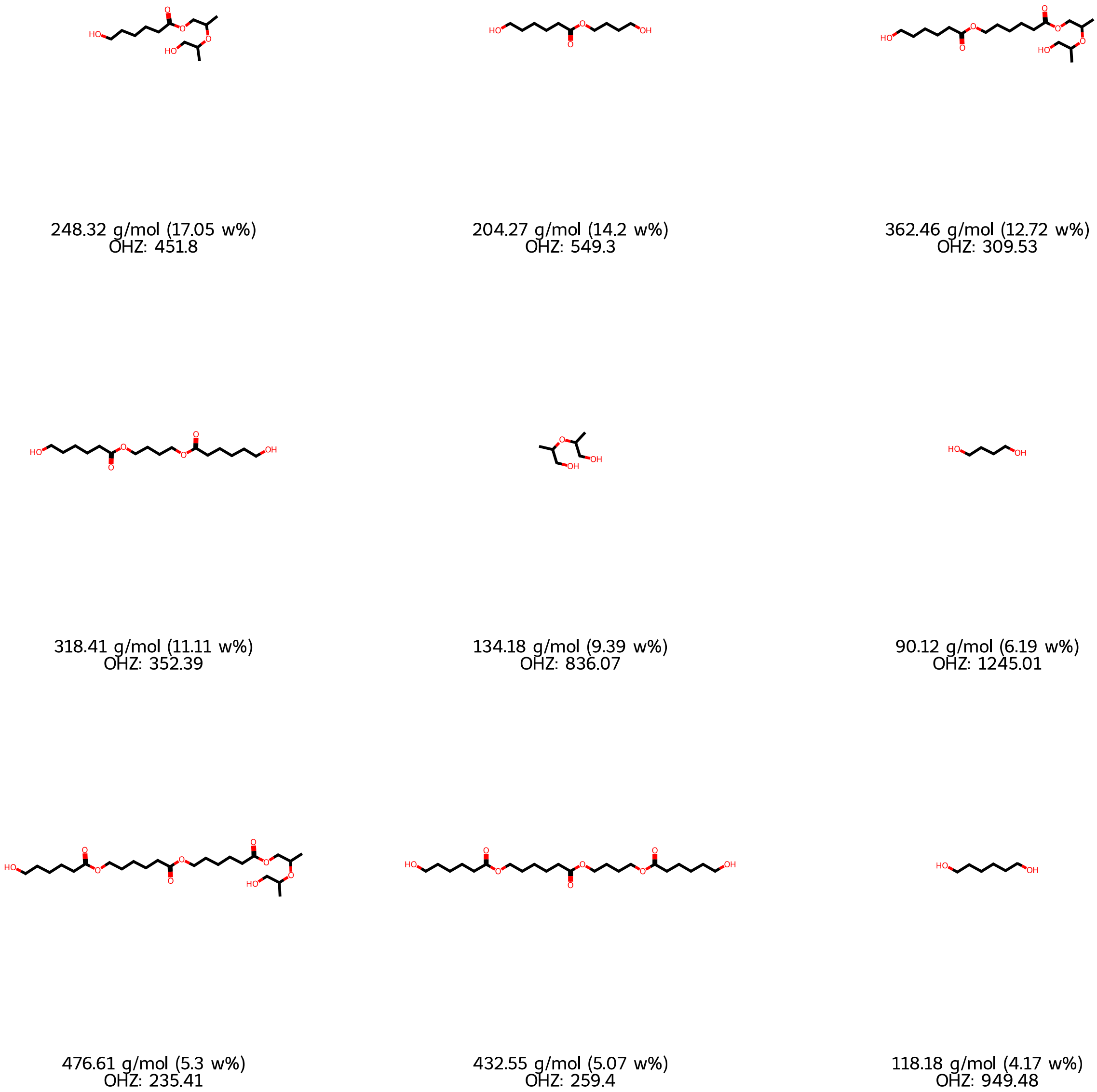


Figure 12: Most frequent oligomeric species (ordered by weight fraction) as created by the kinetic Monte Carlo simulations and subsequent blending for the reference system. The molar mass and weight fraction of the individual species are shown below each species. Furthermore, the hydroxyl number (OHZ) is listed below each species. Molecular structures have been obtained by mapping the polymer graph into molecular graphs after the simulation as explained in Section 1.5.

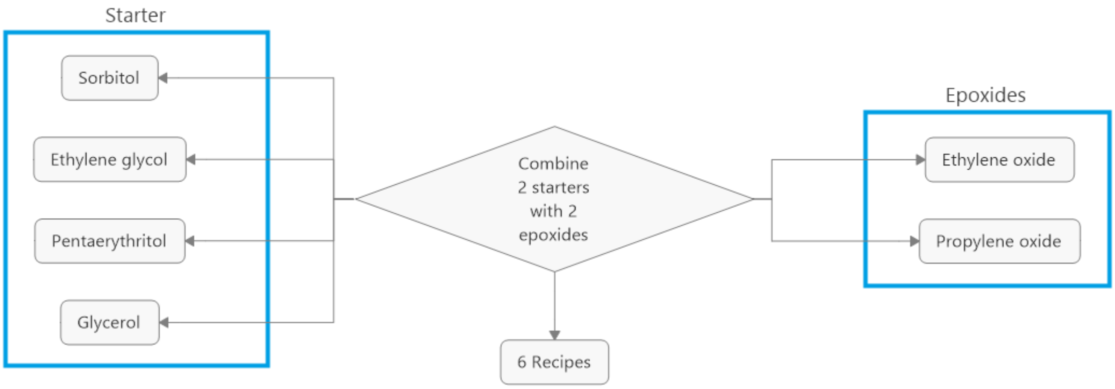


Figure 13: Generation of possible polyether polyol ingredient lists.

The one-pot polyether polyol recipe proposed by the optimizer is depicted in Figure 14. In addition, the nine most frequent oligomeric species of the recipe are depicted in Figure 15. Furthermore, the descriptor values of the reference polyester recipe as well as the proposed polyol recipe are shown in Table 4. It can be observed that, except for the topological index of the kMC graph, the descriptor values show good agreement. Additionally, Table 5 lists computed observables for both recipes, which were no targets during the optimization. Also, for these descriptors a good agreement is observed. Especially the standard deviation of the molar mass and mass-based descriptors (Mw and Mw/Mn) are very similar, indicating that the distribution of the molar mass should be similar between the recipes.

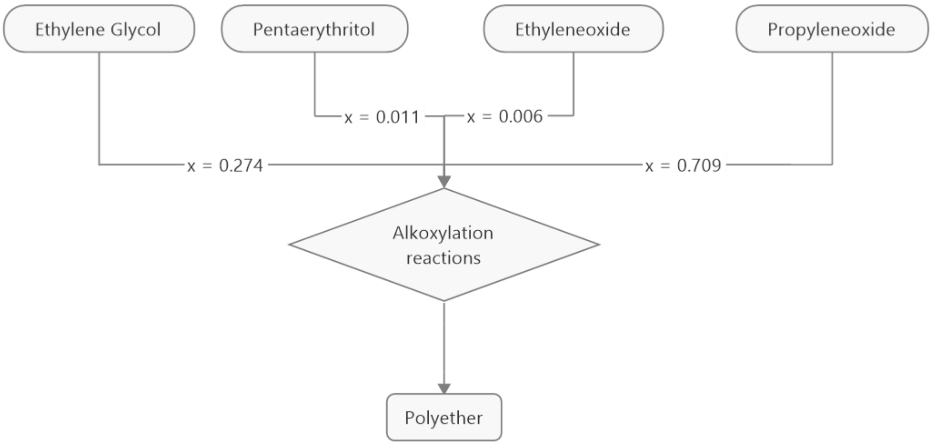
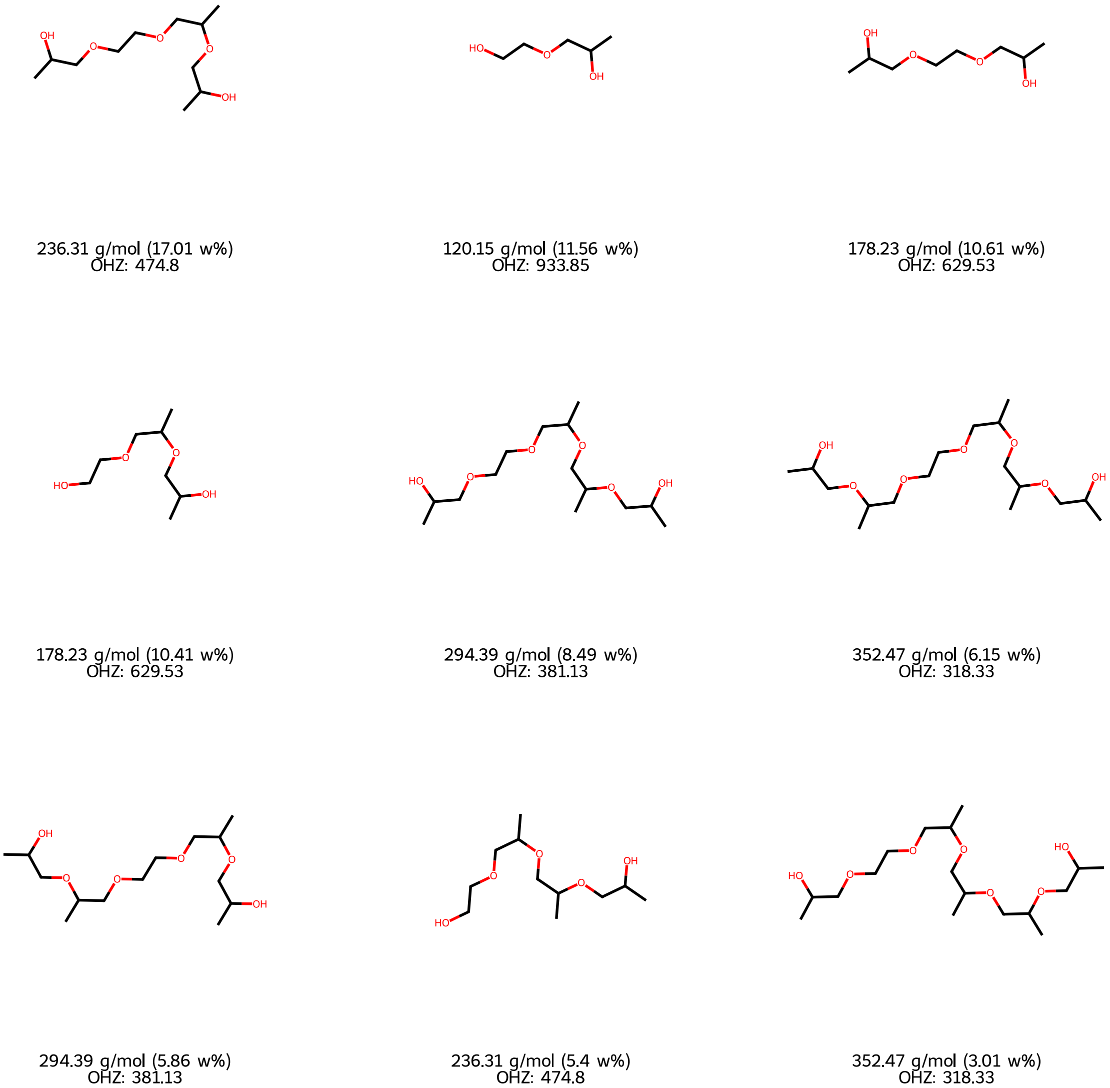


Figure 14: One-pot polyether synthesis route as proposed by the optimizer.



236.31 g/mol (17.01 w%)
OHZ: 474.8

120.15 g/mol (11.56 w%)
OHZ: 933.85

178.23 g/mol (10.61 w%)
OHZ: 629.53

178.23 g/mol (10.41 w%)
OHZ: 629.53

294.39 g/mol (8.49 w%)
OHZ: 381.13

352.47 g/mol (6.15 w%)
OHZ: 318.33

294.39 g/mol (5.86 w%)
OHZ: 381.13

236.31 g/mol (5.4 w%)
OHZ: 474.8

352.47 g/mol (3.01 w%)
OHZ: 318.33

Figure 15: Most frequent oligomeric species (ordered by weight fraction) as created by the kinetic Monte Carlo simulation for the optimized polyether polyol using the TPE sampler. Molar mass and weight fraction of the individual species are shown below each species. Molecular structures have been obtained by mapping the polymer graph into molecular graphs after the simulation as explained in Section 1.5.

Table 4: Computed descriptors for the polyester reference and the optimized polyether recipe proposal. The monomers are abbreviated as caprolactone (CPL), 1,4-butanediol (BDO), 1,2-dipropylene glycol (DPG), adipic acid (ADPS), ethylene glycol (EG), 1,6-hexanediol (HEXD), Pentaerythritol (PERYT), ethylene oxide (EO), propylene oxide (PO).

System	OH#	std(OH#)	Mn	Wiener Index	Monomers
Reference Polyester	528	441	212	0.57	I: CPL, BDO, DPG II: ADPS, EG, HEXD
Proposed Polyether	555	393	210	11.63	EG, PERYT, EO, PO

Table 5: Computed observables for the polyester reference and the optimized polyether recipe proposal. The monomers are abbreviated as caprolactone (CPL), 1,4-butanediol (BDO), 1,2-dipropylene glycol (DPG), adipic acid (ADPS), ethylene glycol (EG), 1,6-hexanediol (HEXD), Pentaerythritol (PERYT), ethylene oxide (EO), propylene oxide (PO).

System	std(M)	Mw	Mw/Mn	logP	std(logP)	Monomers
Reference Polyester	123	284	1.34	0.73	1.17	I: CPL, BDO, DPG II: ADPS, EG, HEXD
Proposed Polyether	100	258	1.23	–0.06	0.64	EG, PERYT, EO, PO

A GPC like spectrum simulated from the molar mass distributions is depicted in Figure 16, following the basic ideas represented in Ref. [37]. Although not being a perfect match, they show similar characteristics, despite originating from very different underlying polymerization mechanisms (chain growth vs step growth!). Similarly, the distribution of OH numbers shows a good agreement between the proposed and reference recipe, as shown in the SI section 1.2.

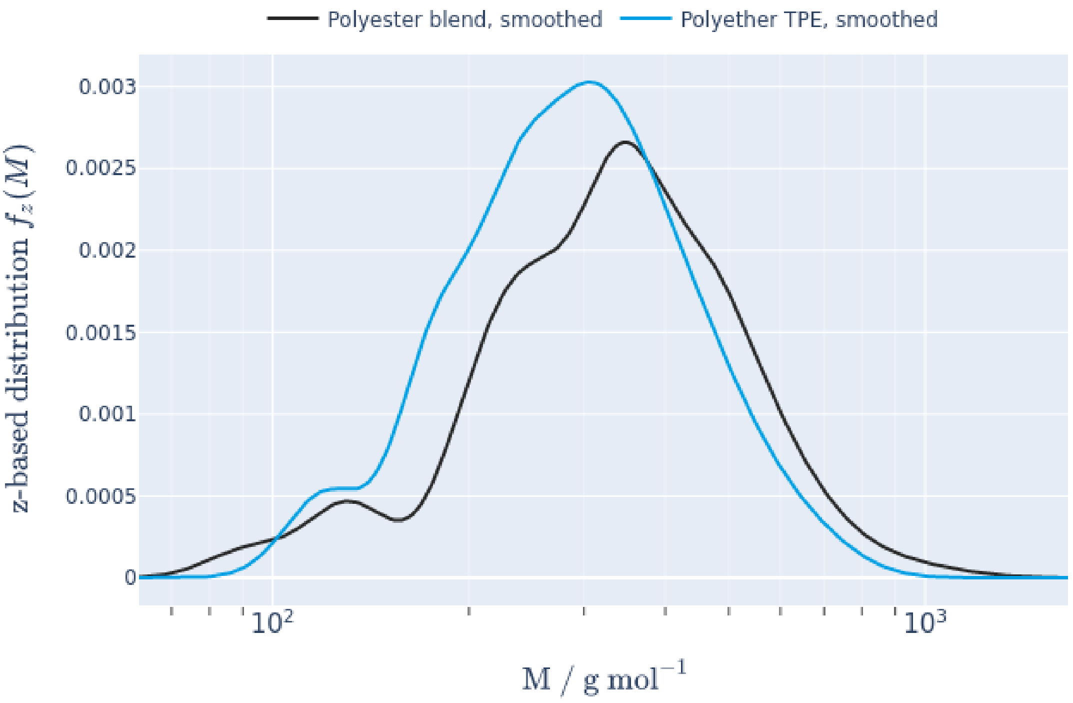


Figure 16: GPC like spectra for the reference polyester polyol (black) and optimized polyether polyol (blue) recipes simulated from the molecular weight distribution. For an improved visual comparison with experimental data, the simulated spectra were broadened, according to eq 16. from Ref. [37] using a Gaussian distribution parameter of $b\sigma_v = 0.06$.

In this example an ingredient list consisting of four different monomers the optimizer can choose from is created. Please note that this only limits the absolute number of ingredients in the final recipe. In the SI (section 1.3), an example is presented in which the molar fractions of two of the ingredients were set to zero during the optimization process, reducing the total number of ingredients and resulting in a significant simplification of the overall composition.

3 Conclusions and Outlook

A workflow for the computer aided chemical recipe design (CARD) is presented which allows for the optimization of existing recipes by interfacing kinetic simulations and cheminformatics tools.

The approach extends computer-aided molecular design (CAMD) approaches, with the focus on relevant uses cases in oligomer and polymer manufacturing. The motivation from an industrial perspective is the consolidation of product portfolios and formulation components, coupled with the in silico testing of (new) biogenic raw materials or those sourced from chemical circular economy (CE) origins. The core idea is to compute descriptor-based fingerprints for a polymer-based reference recipe and propose alternative recipes with similar fingerprints using Bayesian optimization.

In particular, the challenge of dealing with polydispersity, is directly addressed by employing kinetic simulations, with access to realistic chain architectures.

Another key feature of the approach is to use a descriptor-based distance metric (similarity) for optimization. This simplifies issues concerning the replacement of chemicals where the main objective is basically to *not* deteriorate material properties instead of improving properties, for example in a regulatory context.

In addition, a simple but general mapping from the graphs obtained by the kinetic Monte Carlo graphs to atomistic SMILES strings using regular expressions is introduced, exploiting similarities between the BNGL and the SMILES notation. This mapping gives access to a variety of cheminformatics-based descriptors (RDKit) and acts as an interface to simulation models from quantum chemistry or molecular mechanics.

Although, this approach is currently utilized for oligomeric or moderately sized polymeric mixtures, there are no principal restrictions to its potential applications for larger and more complex polymeric systems. Furthermore, processing conditions like temperature dependency or sequence of monomer addition can be addressed.

Ongoing and future work comprises the inclusion of further atomistic simulation tools *e.g.* molecular dynamics or liquid phase thermodynamics as well as incorporation of experimental data already during the optimization procedure.

4 Data And Software Availability

The data that support the findings of this study are available in the supplementary information for this article. The overall workflow is implemented in Python and builds only upon open source packages. It uses an efficient tool for kinetic Monte Carlo simulations (NFsim v1.12.1). Input file for NFsim for the examples of Section 2 can be found in BNGL notation in the supplement. Other packages used are optuna v3.3.0 for the optimization procedure, the RDKit 2023.03.2 for descriptor calculations and networkx v3.1 for graph handling.

5 Supplemental Material

Generated molecular species, graphs, SMILES and descriptor values for the examples (reference and target systems) are available as .csv files in the supplement. Furthermore, NFsim input files are attached in BNGL notation. In the supplementary section there are additional details for the examples such as descriptor distributions and alternative recipe results. Finally, a section on optimizer benchmarks is attached.

6 Acknowledgments

We would like to thank Bill Hlavacek, James Faeder and Ali Sinan Saglam for helpful support concerning the use of the NFsim package and Carsten Möhlenkamp for fruitful discussions related to the use of polymer related kMC simulations. We would like to express our gratitude to Jens Langanke, Head of R&D Simulations & Data Science, Covestro for valuable insights and for backing initiatives associated with this project.

7 Conflict of Interest

There is no conflict of interest to declare.

8 References

1. Weiss, H.; Deglmann, P.; In 't Veld, P. J.; Cetinkaya, M.; Schreiner, E., Multiscale Materials Modeling in an Industrial Environment. *Annu Rev Chem Biomol Eng* **2016,** *7*, 65-86.
2. Abramov, Y. A.; Sun, G.; Zeng, Q., Emerging Landscape of Computational Modeling in Pharmaceutical Development. *Journal of Chemical Information and Modeling* **2022,** *62* (5), 1160-1171.
3. Klamt, A., The COSMO and COSMO-RS solvation models. *WIREs Computational Molecular Science* **2017,** *8* (1).
4. Chen, H.; Kogej, T.; Engkvist, O., Cheminformatics in drug discovery, an industrial perspective. *Molecular Informatics* **2018,** *37* (9-10), 1800041.
5. Sabe, V. T.; Ntombela, T.; Jhamba, L. A.; Maguire, G. E.; Govender, T.; Naicker, T.; Kruger, H. G., Current trends in computer aided drug design and a highlight of drugs discovered via computational techniques: A review. *European Journal of Medicinal Chemistry* **2021,** *224*, 113705.
6. Satyanarayana, K. C.; Abildskov, J.; Gani, R.; Tsolou, G.; Mavrantzas, V. G., Computer aided polymer design using multi-scale modelling. *Brazilian Journal of Chemical Engineering* **2010,** *27*, 369-380.
7. Gertig, C.; Kröger, L.; Fleitmann, L.; Scheffczyk, J.; Bardow, A.; Leonhard, K., Rx-COSMO-CAMD: Computer-Aided Molecular Design of Reaction Solvents Based on Predictive Kinetics from Quantum Chemistry. *Industrial & Engineering Chemistry Research* **2019,** *58* (51), 22835-22846.
8. Zhou, T.; McBride, K.; Linke, S.; Song, Z.; Sundmacher, K., Computer-aided solvent selection and design for efficient chemical processes. *Current Opinion in Chemical Engineering* **2020,** *27*, 35-44.
9. 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* **2016,** *116*, 2-26.
10. Camarda, K. V.; Maranas, C. D., Optimization in Polymer Design Using Connectivity Indices. *Industrial & Engineering Chemistry Research* **1999,** *38* (5), 1884-1892.
11. Vaidyanathan, R.; El-Halwagi, M., Computer-Aided Design of High Performance Polymers. *Journal of Elastomers & Plastics* **1994,** *26* (3), 277-293.
12. Liang, X.; Zhang, X.; Zhang, L.; Liu, L.; Du, J.; Zhu, X.; Ng, K. M., Computer-aided polymer design: Integrating group contribution and molecular dynamics. *Industrial & Engineering Chemistry Research* **2019,** *58* (34), 15542-15552.

13. Lehmann, A.; Maranas, C. D., Molecular Design Using Quantum Chemical Calculations for Property Estimation. *Industrial & Engineering Chemistry Research* **2004,** *43* (13), 3419-3432.

14. Hsu, H.-H.; Huang, C.-H.; Lin, S.-T., Fully Automated Molecular Design with Atomic Resolution for Desired Thermophysical Properties. *Industrial & Engineering Chemistry Research* **2018,** *57* (29), 9683-9692.

15. Ng, L. Y.; Andiappan, V.; Chemmangattuvalappil, N. G.; Ng, D. K. S., Novel Methodology for the Synthesis of Optimal Biochemicals in Integrated Biorefineries via Inverse Design Techniques. *Industrial & Engineering Chemistry Research* **2015,** *54* (21), 5722-5735.

16. Grünewald, F.; Alessandri, R.; Kroon, P. C.; Monticelli, L.; Souza, P. C.; Marrink, S. J., Polyply; a python suite for facilitating simulations of macromolecules and nanomaterials. *Nature communications* **2022,** *13* (1), 1-12.

17. Marvin, W. A.; Rangarajan, S.; Daoutidis, P., Automated generation and optimal selection of biofuel-gasoline blends and their synthesis routes. *Energy & Fuels* **2013,** *27* (6), 3585-3594.

18. Cravero, F.; Schustik, S.; Martínez, M.; Diaz, M. F.; Ponzoni, I., How can polydispersity information be integrated in the QSPR modeling of mechanical properties? *Science and Technology of Advanced Materials: Methods* **2022,** *2* (1), 1-13.

19. Orella, M. J.; Gani, T. Z.; Vermaas, J. V.; Stone, M. L.; Anderson, E. M.; Beckham, G. T.; Brushett, F. R.; Román-Leshkov, Y., Lignin-KMC: A toolkit for simulating lignin biosynthesis. *ACS Sustainable Chemistry & Engineering* **2019,** *7* (22), 18313-18322.

20. Yanez, A. J.; Li, W.; Mabon, R.; Broadbelt, L. J., A stochastic method to generate libraries of structural representations of lignin. *Energy & Fuels* **2016,** *30* (7), 5835-5845.

21. Wang, Y.; Kalscheur, J.; Ebikade, E.; Li, Q.; Vlachos, D. G., LigninGraphs: lignin structure determination with multiscale graph modeling. *Journal of Cheminformatics* **2022,** *14* (1), 43.

22. Lin, T.-S.; Coley, C. W.; Mochigase, H.; Beech, H. K.; Wang, W.; Wang, Z.; Woods, E.; Craig, S. L.; Johnson, J. A.; Kalow, J. A., BigSMILES: a structurally-based line notation for describing macromolecules. *ACS central science* **2019,** *5* (9), 1523-1531.

23. Schneider, L.; Walsh, D.; Olsen, B.; de Pablo, J., Generative BigSMILES: An Extension for Polymer Informatics, Computer Simulations & ML/AI. **2023**.

24. Gillespie, D. T., Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry* **1977,** *81* (25), 2340-2361.

25. Suderman, R.; Mitra, E. D.; Lin, Y. T.; Erickson, K. E.; Feng, S.; Hlavacek, W. S., Generalizing Gillespie's direct method to enable network-free simulations. *Bulletin of mathematical biology* **2019,** *81* (8), 2822-2848.

26. Sneddon, M. W.; Faeder, J. R.; Emonet, T., Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nature Methods* **2011,** *8* (2), 177-183.

27. Liu, B.; Faeder, J. R. In *Parameter estimation of rule-based models using statistical model checking*, 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE: 2016; pp 1453-1459.

28. Weininger, D., SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences* **1988,** *28* (1), 31-36.

29. McKinney, W. In *Data structures for statistical computing in python*, Proceedings of the 9th Python in Science Conference, Austin, TX: 2010; pp 51-56.

30. Brandão, A. L. T.; Soares, J. B. P.; Pinto, J. C.; Alberton, A. L., When Polymer Reaction Engineers Play Dice: Applications of Monte Carlo Models in PRE. *Macromolecular Reaction Engineering* **2015,** *9* (3), 141-185.

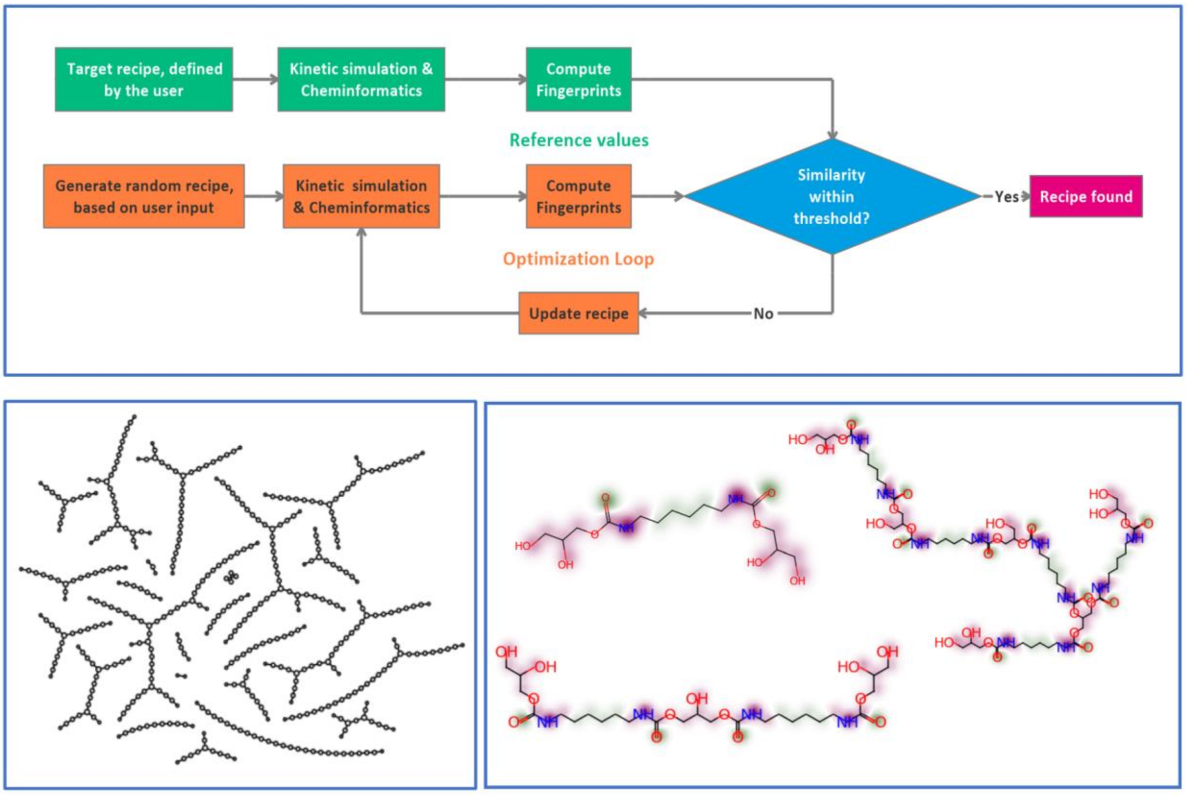
31. Trigilio, A. D.; Marien, Y. W.; Van Steenberge, P. H. M.; D'hooge, D. R., Gillespie-Driven kinetic Monte Carlo Algorithms to Model Events for Bulk or Solution (Bio)Chemical Systems Containing Elemental and Distributed Species. *Industrial & Engineering Chemistry Research* **2020,** *59* (41), 18357-18386.

32. Macosko, C. W.; Miller, D. R., A New Derivation of Average Molecular Weights of Nonlinear Polymers. *Macromolecules* **1976,** *9* (2), 199-206.
33. Bachmann, R.; Klinger, M.; Meyer, J., Random Branching and Cross-linking of Polymer Chains, Analytical Functions for the Bivariate Molecular Weight Distributions. *Macromolecular Theory and Simulations* **2023**, 2200062.
34. Faeder, J. R.; Blinov, M. L.; Goldstein, B.; Hlavacek, W. S., Rule-based modeling of biochemical networks. *Complexity* **2005,** *10* (4), 22-41.
35. Blinov, M.; Faeder, J.; Hlavacek, W., Rule-based modeling of biochemical networks. Google Patents: 2005.
36. Dahlke, A. Reordering SMILES atom order. http://www.dalkescientific.com/writings/diary/archive/2010/12/28/reordering_smiles.html (accessed 6.6.2023).
37. Marien, Y. W.; Edeleva, M.; Figueira, F. L.; Arraez, F. J.; Van Steenberge, P. H.; D'hooge, D. R., Translating simulated chain length and molar mass distributions in chain-growth polymerization for experimental comparison and mechanistic insight. *Macromol. Theory Simul.* **2021,** *30* (3), 2100008.
38. Landrum, G. *RDKit: Open-source cheminformatics.*, 2021.09.2; 2021.
39. Bi, B.; Jamil, M. K.; Muhammad Fahd, K.; Sun, T.-L.; Ahmad, I.; Ding, L., Algorithms for Computing Wiener Indices of Acyclic and Unicyclic Graphs. *Complexity* **2021,** *2021*.
40. Hagberg, A.; Swart, P.; S Chult, D. *Exploring network structure, dynamics, and function using NetworkX*; Los Alamos National Lab.(LANL), Los Alamos, NM (United States): 2008.
41. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M., Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery: Anchorage, AK, USA, 2019; pp 2623–2631.

For Table of Contents Use Only

Computer Aided Recipe Design: Optimization of Polydisperse Chemical Mixtures using Molecular Descriptors

Anja Massolle, Jakob Schneider, Jan Meyer, Christoph Loschen*


Target recipe, defined by the user
Kinetic simulation & Cheminformatics
Compute Fingerprints
Reference values
Generate random recipe, based on user input
Kinetic simulation & Cheminformatics
Compute Fingerprints
Similarity within threshold?
Yes
Recipe found
Optimization Loop
Update recipe
No