Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue



Cite this: *Digital Discovery*, 2023, **2**, 828

# Using physical property surrogate models to perform accelerated multi-fidelity optimization of force field parameters†

Owen C. Madin and Michael R. Shirts*

Accurate representations of van der Waals dispersion–repulsion interactions play an important role in high-quality molecular dynamics simulations. Training the force field parameters used in the Lennard Jones (LJ) potential typically used to represent these interactions is challenging, generally requiring adjustment based on simulations of macroscopic physical properties. The large computational expense of these simulations, especially when many parameters must be trained simultaneously, limits the size of training data set and number of optimization steps that can be taken, often requiring modelers to perform optimizations within a local parameter region. To allow for more global LJ parameter optimization against large training sets, we introduce a multi-fidelity optimization technique which uses Gaussian process surrogate modeling to build inexpensive models of physical properties as a function of LJ parameters. This approach allows for fast evaluation of approximate objective functions, greatly accelerating searches over parameter space and enabling the use of optimization algorithms capable of searching more globally. In this study, we use an iterative framework which performs global optimization with differential evolution at the surrogate level, followed by validation at the simulation level and surrogate refinement. Using this technique on two previously studied training sets, containing up to 195 physical property targets, we refit a subset of the LJ parameters for the OpenFF 1.0.0 (Parsley) force field. We demonstrate that this multi-fidelity technique can find improved parameter sets compared to a purely simulation-based optimization by searching more broadly and escaping local minima. Additionally, this technique often finds significantly different parameter minima that have comparably accurate performance. In most cases, these parameter sets are transferable to other similar molecules in a test set. Our multi-fidelity technique provides a platform for rapid, more global optimization of molecular models against physical properties, as well as a number of opportunities for further refinement of the technique.

Received 10th December 2022
Accepted 28th April 2023

DOI: 10.1039/d2dd00138a

rsc.li/digitaldiscovery



## 1 Introduction

### 1.1 Accurate force fields are important in computational biophysics

Accurate molecular interaction potentials, usually referred to as force fields, are an essential part of modern molecular dynamics workflows. For common applications such as simulations of proteins and computer aided drug design (CADD), the simple fixed-charge force field functional form[1–3] is generally used. This formulation splits the potential energy of molecules into discrete components, with separate energy terms for each component.[4] Broadly, these can be divided into the bonded (or valence) components, which give the energies corresponding to the bond lengths, bond angles, and torsional angles, and the non-bonded components, representing short-range dispersion–repulsion interactions and longer-range coulombic interactions.

This type of force field has been successful in many applications because of its simplicity, interpretability, and computational efficiency. Many studies have used these force fields to probe the mechanisms of protein dynamics,[5–8] and they have become widely adopted in the pharmaceutical industry as a means of screening drug candidate molecules *in silico*.[8–12] While these force fields are quite simple in their functional form, their accuracy is dependent on hundreds to thousands of empirical parameters, which dictate the strength of interactions in different molecular configurations and in different chemical environments. Decades of effort from the computational chemistry community have produced many different parameter sets to cover a wide range of chemistries,[3,13–16] largely by fitting parameters to quantum mechanics (QM) calculations[17,18] and experimental physical properties.[19,20]

*Department of Chemical & Biological Engineering, University of Colorado Boulder, Boulder, CO, USA, 80309. E-mail: michael.shirts@colorado.edu*

† Electronic supplementary information (ESI) available. See DOI: https://doi.org/10.1039/d2dd00138a

### 1.2 Non-bonded training is expensive and difficult

Over the years, fitting of the bonded parameters has perhaps received the most attention, due to their importance in determining the internal structure of molecules and proteins, and the relative ease of generating gas-phase QM data for the molecules of interest. Fitting the atomic partial charges used in the coulombic potential has also received significant attention, but is slightly more difficult, as mapping an continuous electrostatic potential onto a set of discrete atoms is conformation-dependent and involves a loss of fidelity. However, modelers have achieved good results using QM-based methods such as RESP[21,22] and semi-empirical methods such as AM1-BCC.[23,24]

The dispersion–repulsion interactions, usually modeled with the Lennard-Jones (LJ) potential, have received the least attention in fitting, as they are typically trained against experimental physical property data,[19,20,25] since obtaining dispersion–repulsion estimates from QM is difficult.[26] This leads to challenges with curating appropriate sets of experimental physical property data from the literature, as well as the computational cost of simulating sets of physical property data with molecular dynamics. Most physical properties used in training, which include densities,[19] enthalpies of vaporization,[19] enthalpies of mixing,[27] solvation free energies[20] and dielectric constants,[28,29] require equilibrium simulations in one or more phases, and in some cases may require alchemical simulation techniques.[30] In conjunction with the need to train against larger datasets to ensure accuracy and transferability, this makes optimization of LJ parameters a challenging problem. Calculating a single objective function value in order to measure parameter fitness requires a large number of simulations, which can be difficult to coordinate and execute, especially depending on available computational resources. As a result, one can find many instances of LJ parameters in major force fields that have remained unchanged for more than 20 years, despite significant advancements in hardware, simulation software, and methodology in that time.

Recently, as part of the Open Force Field (OpenFF) Initiative, we have examined new methods of LJ parameter optimization. Central to these efforts is the development of the OpenFF Evaluator simulation workflow driver,[31] which provides a standardized set of workflows for automatically building and executing physical property simulations for a given training or test data set. With the automation that this software provides, we can apply optimization techniques to LJ parameters with minimal human intervention. In particular, this software enabled the application of the ForceBalance[32] parameter optimization package to improve LJ parameters. Using regularized least squares optimization with the L-BFGS-B algorithm,[33] we minimized an objective function that captures the ability of a parameter set to reproduce physical property observables. Using this framework, we also studied the benefits of including physical property data of mixtures in training LJ parameters,[27] then applied that training method to a production force field, OpenFF 2.0.0 (also known as "Sage").[34]

While this approach has produced parameter sets with improved performance in predicting experimental physical properties, using simulation-based regularized least-squares optimization has significant limitations. A major drawback is that, regardless of the optimization algorithm used, the number of objective function evaluations possible is limited by the computational cost of simulations. This limits the number of parameter sets that can be considered during optimization, making it difficult to explore high-dimensional and complex parameter spaces. This also necessitates the use of cheaper, fully local optimization methods such as L-BFGS-B with termination after a set number of steps.[27] When coupled with a regularization term in the objective function, included both to ensure the stability of the optimization and to guard against overfitting,[32] local optimization methods have a high probability of remaining in any local minima dictated by its initial values. This means that our ability to explore new areas of parameter space that may provide significant improvement is blunted because of the expense of evaluating the objective and the difficulty of escaping a local minima with a gradient-based optimization method.

### 1.3 Surrogate modeling can accelerate non-bonded training

To facilitate faster evaluation of complex objective functions, modelers often use surrogate models,[35,36] which are meant to approximate an expensive function with a simpler alternative that captures a sufficient amount of the important information of the response function. Surrogate modeling techniques have been developed in response to diverse sets of scientific and engineering challenges, including geological modeling,[37,38] engineering design,[39,40] and chemical process modeling.[41,42] A popular technique is Gaussian process (GP) surrogate modeling, which has seen adoption in many disciplines[43–45] due to its simplicity and efficacy in data-sparse regimes. While these surrogates cannot be perfect imitations of the high-level responses, for sufficiently smooth functions, we can construct surrogates with a reasonable level of accuracy with only a limited number of expensive evaluations. In the context of molecular simulation parameter optimization, Befort *et al.*[46] demonstrated a method of optimizing LJ parameters by building GP surrogates based on physical properties and applied this method to several hydrofluorocarbons as well as ammonium perchlorate.

In this paper, we build on this approach, as well as engineering optimization literature[47] and our OpenFF Evaluator software, to introduce a multi-fidelity optimization framework based on the construction of Gaussian process (GP) surrogate models that approximate the response surface of many physical properties with respect to changes in the LJ parameters. Using the accelerated objective evaluation offered by the surrogates, we implement a global optimization algorithm to search broadly and propose candidate parameter sets. We then validate these parameter sets by evaluating the objective at the simulation level, accepting candidates in good agreement. Iterating between global optimization over the surrogate, and simulation-level validation and surrogate refinement, we can drive the optimizer to explore promising regions of parameter space with a limited number of simulation evaluations.

We test this approach by performing multi-fidelity LJ optimization for 12 commonly exercised LJ parameters from the OpenFF 1.0.0 (Parsley) force field. In training, we use a set of 56 pure compound physical properties curated in a previous paper.[27] We also benchmark the results against test sets curated in the same paper; while newer versions of OpenFF exist, using OpenFF 1.0.0 allows a direct comparison to the results of our previous optimization. With this context, we characterize the optimization method, and discuss reproducibility, seed configurations, and optimization trajectories. We also show that this method can be extended to larger problems by applying it to a larger training set of 195 mixture properties from the same paper.

## 2 Methods

### 2.1 Optimization strategy

Our optimization strategy aims to minimize an objective function $\chi(\theta)$ as in eqn (1), where $\theta$ is a vector of force field parameters, and $\chi$ is some measure of the fitness of those parameters.

$$\min_{\theta} \chi(\theta) \quad (1)$$

In our applications, parameter fitness is described by the ability of a force field containing those parameters to reproduce a specific training set of experimental physical properties, although we note that this strategy could also be applied to a training set containing quantities from QM simulations. The optimization strategy we employ is adapted from the framework proposed by Dennis and Troczon[47] and features two levels of fidelity for estimating the objective function for a parameter set:

- "Simulation level", where the objective function is directly evaluated by using molecular dynamics to simulate the training set with a force field containing the parameter set. This is considered to be the "ground truth", as it is a direct measurement the force field's performance, although there is some level of statistical uncertainty due to the stochasticity in the molecular dynamics simulation.
- "Surrogate level", where the objective function is estimated by a collection of surrogate models that approximate the result of a simulation-level evaluation of the training set. The surrogate-level evaluation of the objective function has systematic uncertainty where the surrogates approximation deviates from the simulation-level estimation of the training set. It may also have statistical uncertainty depending on the type of surrogate used, although the surrogates that we use do not.

Our optimization strategy relies on the cheaper but less accurate surrogate level to perform most of the optimization, using the more accurate and expensive simulation level only to build the surrogates and validates proposed surrogate-level solutions. The optimization framework is illustrated in Fig. 1.

The advantage of this strategy is its use of the properties of both the surrogate and simulation level to drive optimization. While surrogate level evaluation is much faster than simulation-level evaluation, surrogates need simulation points in the region of interest to accurately reproduce the objective function.

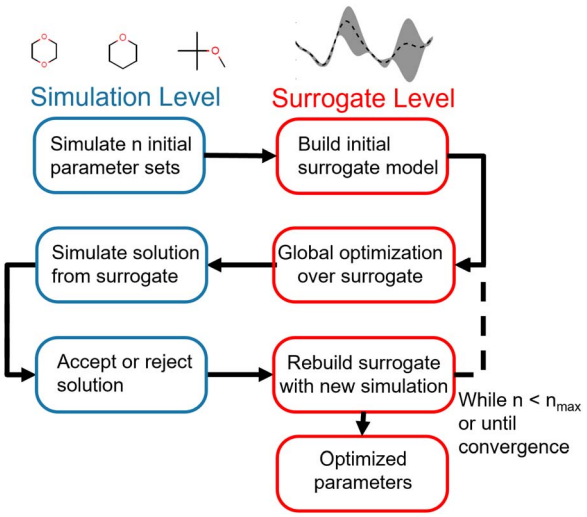


Fig. 1 Flowchart of multi-fidelity optimization strategy. Optimization is initialized by simulating an initial sample of parameter vectors. Surrogate models for each of the physical properties in the training set are then built from this initial sample. Global optimization is then performed at the surrogate level, utilizing the speedup gained with surrogate-level fast objective evaluation. Once this proposes a candidate vector of optimized parameters, the objective function for that parameter vector is evaluated at the simulation level. If the simulation-level objective is lower than the simulation objective for the previous parameter vector, the new parameter vector is accepted as an improved solution; if not, it is rejected. Regardless of acceptance or rejection, the surrogate model is rebuilt with the information from the simulation-level evaluation. This process is then repeated until a maximum number of simulation optimizations is reached, a convergence criteria is met, or the optimizer cannot find an improved solution.

Since parameter spaces are large and the region of interest is not known *a priori*, an exhaustive strategy would require a very large number of simulation-level evaluations to build globally accurate surrogates, negating the speedup gained by using surrogates. We instead build a surrogate from a minimal initial set of evaluations of the objective function and allow the surrogate to suggest new parameter vectors for the simulation level to evaluate. We can therefore iteratively drive the optimization towards the region of interest without incurring too much computational cost, acquiring more information to improve the surrogates along the way. This allows us to pair the global optimization strategies available at the surrogate level with the accuracy of simulation-level validation. While this overall strategy is not strictly a global optimization, since we only use global optimizations at the approximate surrogate level, it does allow for a much wider search of the parameter space than a gradient-based local optimization.

This strategy is sufficiently general to allow for the use of a large variety of objective functions, surrogate-level global optimization techniques, and surrogate modeling strategies. In this particular study we focus on a single combination: a weighted least-squares objective function based on experimental properties, the differential evolution global optimization algorithm, and Gaussian process (GP) surrogate modeling.

**2.1.1 Objective function.** The objective function we use, shown in eqn (2), is adapted from the type used in the ForceBalance optimization software package and used in our previous work.[27] While that objective included a regularization term for stability and to prevent overfitting, we omit that term, allowing our algorithm to search more broadly to find optimized parameter vectors.

$$\chi(\theta) = \sum_{n=1}^{N} \frac{1}{M_n} \sum_{m=1}^{M_n} \left( \frac{y_m^{\text{ref}} - y_m(\theta)}{d_n} \right)^2 \quad (2)$$

In this equation, we consider $N$ types of physical properties, each with some number $M_n$ of measurements for that type. The quantity $y_m$ represents the value of the $m$th measurement for a physical property type, and the denominator $d_n$ is a scaling coefficient for a given property. The values of $y_m(\theta)$ can either be obtained directly from simulation, or from surrogate models based on those simulations. The scaling coefficients are set so that each physical property type contributes equally to the objective function for OpenFF 1.0.0,[18] the starting point of our optimizations.

**2.1.2 Global optimization.** We use the differential evolution[48] global optimization algorithm, as implemented in the SciPy Python package version 1.7.0.[49] Differential evolution is a stochastic direct search algorithm similar to other genetic algorithm strategies.[50] In this strategy, a set of $N$ initial vectors is proposed, and then "mutated" by randomly increasing or decreasing elements of the vector, and recombined, by randomly replacing some elements of the vector with elements of other vectors. The objective function is then evaluated for each of the proposed vectors, and a new set of vectors is proposed based on the lowest objective functions. This process is repeated until convergence, when new vectors no longer outperform the current solutions. We use the default optimization parameters in SciPy, with a population size of 180 vectors, an iteration-dependent mutation constant selected from the range (0.5, 1), and a recombination constant of 0.7. We note that, depending on the problem, a single iteration of this algorithm requires between $10^3$ and $10^4$ objective function evaluations *per iteration*, each of which depends on the value of 50–200 physical properties.

The bounds of the global optimization over the surrogate model are determined by the parameter sets used to build the surrogate. For each parameter $\theta_i$ in the parameter vector $\theta$, the bounds for $\theta_i$ are determined by the minimum ($\min(\theta_i)$) and maximum value ($\max(\theta_i)$) of $\theta_i$ in the set of parameter vectors $\Theta = [\theta^1, \theta^2, \ldots, \theta^N]$ used to build the surrogate. We then apply a small expansion factor $\eta$ to the parameter range, so that the optimization algorithm can search outside of the initial simulation box (described in Section 2.4.2). To expand the box, we multiply $\max(\theta_i)$ by $\eta$, and divide $\min(\theta_i)$ by $\eta$ to form the bounds box [$\text{LB}(\theta_i)$, $\text{UB}(\theta_i)$]. We chose the value of $\eta$ to be 1.1, expanding the box 10% in each direction, to allow the optimizer to search aggressively. This process is repeated for each parameter $\theta_i \in \theta$ to form an $N$-dimensional box, and is described in eqn (3). Due to the nature of the optimization (as described in Fig. 1), the bounds are recomputed at each iteration, after more simulation information has been added to the surrogate. This allows the bounds to change significantly over the optimization as the algorithm explores new areas of parameter space.

$$\text{LB}(\theta_i) = \frac{1}{\eta} \times \min_{\theta \in \Theta}(\theta_i), \quad \text{UB}(\theta_i) = \eta \times \max_{\theta \in \Theta}(\theta_i) \quad (3)$$

### 2.2 Construction of physical property surrogates

GP surrogate models are built with the BoTorch[51] software package, version 0.6.0, which provides a convenient and extensible framework for building a large number of surrogates. Surrogates are constructed individually for each physical property in the test set, from all of the simulation level evaluations available; *e.g.* if there are simulations of 20 physical properties with 10 different LJ parameter vectors, then our process builds 20 individual surrogate models, each using all 10 parameter vectors in their construction. Objective functions are calculated from the surrogates' predictions of their respective physical properties; the surrogate does not predict the objective function directly, such as is done in Bayesian optimization. All surrogates use a constant mean function and RBF (radial basis function) covariance kernel; independent length scales $l$ for each parameter are chosen using automatic relevance determination (ARD).[52] The length scales $l$ in the covariance kernel represent the distances over which points are correlated in each dimension.

If a simulation of a physical property at a given set of parameters finishes with errors, that set is omitted from surrogate building; additionally, any sets that have density measurements lower than 20% of the experimental value are omitted, as this likely indicates that the parameters have induced a phase change. The rationale for this criteria is that we are attempting to build a surrogate model which accurately predicts liquid densities in over parameter space, and parameter sets that predict a gaseous or solid system at temperature and pressure where that system should be liquid will not provide useful information. However, this restriction did not affect the optimization, as parameters violating the density constraint were never produced through our optimizations.

In cases where an optimization iteration over a surrogate model fails to find a lower objective value than the current simulation objective, surrogates are rebuilt with constraints on the length scales $l$ used for the variances of each parameter. This approach was chosen as we found in testing that optimizations may fail because a surrogate was set with a length scale too low during ARD, producing a surrogate with poor quality for a particular physical property. If the optimizer cannot find a better objective value over the surrogate, it is first rebuilt with a length scale constraint such that $l > 10^{-10}$; if this is not successful, a stricter length scale constraint of $l > 10^{-5}$ is imposed. If this is still not successful, the optimization is terminated. The quality of the surrogate model is reduced when length scale constraints are introduced, so constraints are not used unless an optimization fails.

### 2.3 Physical property simulations

Physical property simulations were handled with the OpenFF Evaluator[31] software package, version 0.3.4,[53] using the default workflows[54] for all properties simulated. We performed simulations to estimate pure density ($\rho_L$), mixture density ($\rho_L(x)$), enthalpy of vaporization ($\Delta H_{vap}$) and enthalpy of mixing ($\Delta H_{mix}(x)$). To summarize the procedure, we performed all condensed-phase simulations in the NPT ensemble, with initial simulation boxes of 1000 molecules built using PackMOL.[55] After building the boxes, we perform an energy minimization on the simulation boxes, followed by a 0.2 ns equilibration simulation and a 2 ns production simulation, which was found to be sufficient to converge these simple physical properties.[31] In the calculation of $\Delta H_{vap}$, we use 30 ns single molecular NVT simulations without periodic boundary conditions to estimate the gas phase energies. All simulations use a 2 fs timestep and a Langevin integrator with BAOAB splitting.[56] More complete simulation details are available in our previous work,[27] which uses the same simulation workflows.

### 2.4 Optimization tasks

We focused on two separate optimization tasks, both developed in our previous study.[27] Both tasks optimize the same set of 12 LJ parameters ($R_{min/2}$ and $\varepsilon$ for 6 LJ SMIRKS types), and both use the same small molecules (alkanes, alcohols, ethers, esters and ketones) in the their training sets. The tasks are differentiated by the different types of physical property training data that are used in the evaluation of the objective function:

(1) "Pure only": this task optimizes against a set of 56 pure compound measurements, $\rho_L$ and $\Delta H_{vap}$ for each of 28 compounds in the training set. We used this task to test the optimization strategy, as it represents the typical type of training set used in LJ optimization, and has relatively low computational expense because of the number and types of physical properties that need to be estimated.

(2) "Mixture only": this task optimizes against a larger set of 195 physical properties of binary mixtures ($\Delta H_{mix}(x)$ and $\rho_L(x)$). This task extends the strategy to a significantly larger training set, and represents the type of training set that performed best in our previous study.

While we reported optimized parameter sets for these tasks in our previous work, here we use those sets as a baseline to test our multi-fidelity strategy.

**2.4.1 Parameters to be optimized.** We optimize the LJ $R_{min/2}$ and $\varepsilon$ for 6 LJ types, which are described in Table 1. We also note that several LJ types are exercised by molecules in the training set, but are not optimized, due to either having very specific chemical contexts that are not exercised widely enough to optimize, or, in the case of the [#1:1]-[#8] (hydroxyl hydrogen) parameter, because the $\varepsilon$ has been set to an arbitrary small non-zero value to avoid unphysical effects.[57]

**2.4.2 Initial physical property simulations.** To build an initial surrogate in each optimization, we simulate an initial set of $N$ parameter vectors, one of which is always the parameter vector corresponding to OpenFF 1.0.0. We select these vectors from an initial parameter space, described in Table 2. This space is measured in percentage of the parameter values from OpenFF 1.0.0, and is determined from the results of our previous optimization study, based on how much each parameter was adjusted in that study. From this space, we select the remaining $N - 1$ parameter vectors using Latin hypercube sampling (LHS), as implemented with the Surrogate Modeling Toolbox[58] (SMT) Python library, version 1.1.0. Since the optimization bounds are recomputed after each optimization iteration, solutions are not restricted to this initial space.

**2.4.3 "Pure only" optimization task.** The "pure only" optimization task fits the LJ parameters against a total of 56 physical properties ($\rho_L$ and $\Delta H_{vap}$ for a set of 28 molecules), which are shown in Fig. 2. A list of the molecules in the "pure only" training set are available in the ESI, Section S1.1.†

The measurements here are either sourced from the NIST ThermoML Archive[59,60] ($\rho_L$) or hand-curated from literature ($\Delta H_{vap}$),[61–72] because of the low number of $\Delta H_{vap}$ data points in the ThermoML Archive. All measurements are selected at temperatures and pressures close to ambient (~1 atm, 273.15–318.15 K).

Table 1 All LJ SMIRKS types, both adjusted and not adjusted, in the training of OpenFF 2.0.0, along with descriptions of the chemical contexts they describe. Both LJ $\varepsilon$ and $R_{min/2}$ are adjusted for each of the types under the "refitted parameters" subheading

| Refitted SMIRKS type | Description |
|---|---|
| **Refitted parameters** | |
| [#1:1]-[#6X4] | Hydrogen attached to tetravalent carbon |
| [#6:1] | Generic carbon |
| [#6X4:1] | Tetravalent carbon |
| [#8:1] | Generic oxygen |
| [#8X2H0+0:1] | Divalent oxygen with no hydrogens attached |
| [#8X2H1+0:1] | Divalent oxygen with one hydrogen attached |
| **Parameters exercised but not refitted** | |
| [#1:1]-([#6X4])-[#7, #8, #9, #16, #17, #35] | Hydrogen attached to tetravalent carbon attached to an electronegative atom |
| [#1:1]-[#6X3](~[#7, #8, #9, #16, #17, #35])~[#7, #8, #9, #16, #17, #35] | Hydrogen attached to trivalent carbon attached to 2 electronegative atoms |
| [#1:1]-[#8] | Hydrogen attached to oxygen |

Table 2 Parameter space that initial parameter vectors are sampled from, defined as percentages of OpenFF 1.0.0 values. For a set of $N$ parameter vectors used to initialize the surrogate model, Latin hypercube sampling is used to select $N-1$ parameter vectors from this parameter space, with the final parameter vector being the parameters from OpenFF 1.0.0

| Refit parameters | | |
|---|---|---|
| Refit SMIRKS type | $\varepsilon$ initial parameter range (% of OpenFF 1.0.0) | $R_{min/2}$ initial parameter range (% of OpenFF 1.0.0) |
| [#1:1]-[#6X4] | (50, 150) | (95, 105) |
| [#6:1] | (90, 110) | (95, 105) |
| [#6X4:1] | (90, 110) | (95, 105) |
| [#8:1] | (95, 105) | (95, 105) |
| [#8X2H0+0:1] | (95, 105) | (95, 105) |
| [#8X2H1+0:1] | (95, 105) | (95, 105) |

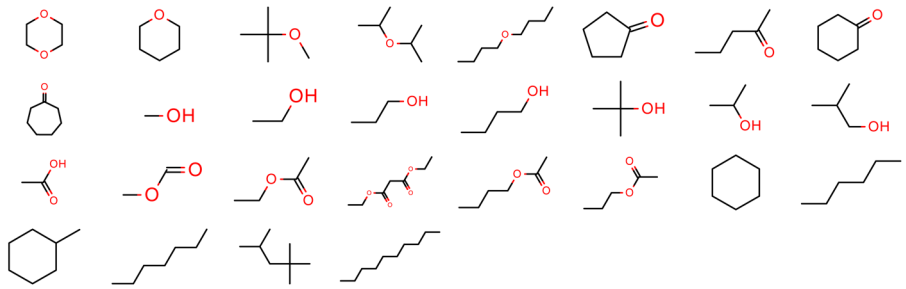

Fig. 2 Molecules in the "pure only" training set. Physical properties in this set include one measurement each of $\rho_L$ and $\Delta H_{vap}$, and are sourced from either the NIST ThermoML Archive ($\rho_L$) or hand-curated from literature ($\Delta H_{vap}$).

For this optimization task, we performed optimizations using $N = 5$ and $N = 10$ initial points, to test the effect of the number of initial simulation points on the performance of the algorithm. We performed 5 replicates for both $N = 5$ and $N = 10$ initial points, in order to assess the consistency of the algorithm. For the $N = 10$ replicates, a different set of 9 LHS initial points is selected each time; for the $N = 5$ replicates, each set of initial points is formed by subsampling 4 LHS points from one of the $N = 10$ replicate initial sets, in order to minimize simulation expense.

**2.4.4 "Mixture only" training set.** The "mixture only" optimization task optimizes the LJ parameters against a set of 195 physical properties ($\rho_L(x)$, $\Delta H_{mix}(x)$) for the set of molecule pairs shown in Fig. 3. These molecule pairs are drawn from the same set of molecules as used in the "pure only" training set. All measurements here are selected from the NIST ThermoML archive, and are selected at temperatures and pressures close to ambient (~1 atm, 273.15–318.15 K). We select measurements at concentrations within 0.05 mole fraction of 3 target concentrations for each mixture, where available: ($x_1 = 0.25$, $x_2 = 0.75$),

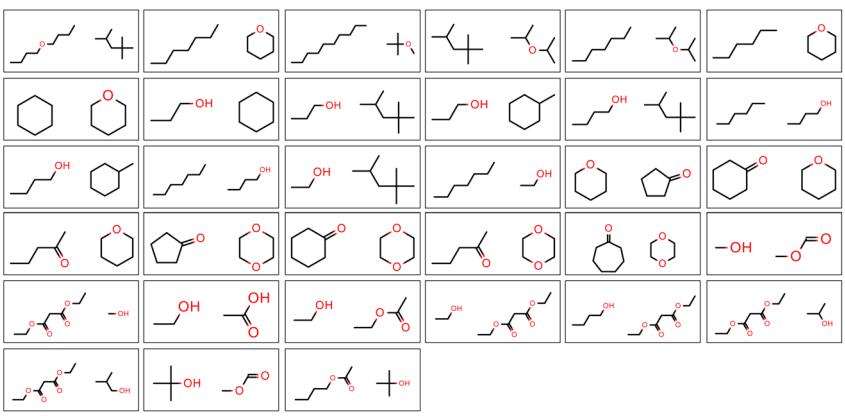

Fig. 3 Molecules in the "mixture only" training set. Physical properties in this set include measurements of $\rho_L(x)$ and $\Delta H_{mix}(x)$ at conditions close to ambient (~1 atm, 273.15–318.15 K), and several concentrations (($x_1 = 0.25$, $x_2 = 0.75$), ($x_1 = 0.5$, $x_2 = 0.5$), ($x_1 = 0.75$, $x_2 = 0.25$)), where available, yielding a total of 195 measurements. All measurements are sourced from the NIST ThermoML Archive.

($x_1 = 0.5$, $x_2 = 0.5$), ($x_1 = 0.75$, $x_2 = 0.25$). If no measurements are available within 0.05 mole fraction of a target concentration, no data point is selected for that target concentration. A list of the mixtures in the "mixture only" training set is available in the ESI, Section S1.2.†

For this second optimization task, we performed an optimization using $N = 20$ initial points, due to the increased complexity of the training set. After this optimization, we performed a second optimization using $N = 10$ initial points, in order to test whether a more data-sparse optimization could be successful. For the $N = 10$ replicate, 9 initial points are subsampled from the 19 LHS points used in the $N = 20$ replicate to minimize simulation expense.

### 2.5 Benchmarking

To assess the quality and transferability of the parameter sets produced by our optimization, we tested them on a set of physical properties (29 $\rho_L$, 318 $\rho_L(x)$, 29 $\Delta H_{vap}$, and 236 $\Delta H_{mix}(x)$) for a new set of molecules and molecule pairs, which serves as the test set. This data set was curated for our previous work, and its selection and composition are discussed there.[27] Physical properties in this set are either hand-curated from literature ($\rho_L$, $\Delta H_{vap}$), or are selected automatically from the NIST ThermoML Archive. Benchmarking simulations are performed using the same OpenFF Evaluator workflows as simulations used in the optimization process.

## 3 Results & discussion

### 3.1 Pure training set

**3.1.1 Optimization.** Optimization was generally successful with both $N = 5$ and $N = 10$ initial parameter vectors, as the process reached significantly lower objective function values than the initial force field in every case. Additionally, when comparing training set RMSE for $\Delta H_{vap}$, all optimization replicates significantly outperform the regularized least squares optimization.

Out of the 10 optimizations run, 4 of them terminated early after the surrogate optimizer could not find an improved solution. This is related to the issues with ARD noted in Section 2.2. This suggests that further refinement is needed to improve the robustness of the surrogate model. Optimizations used between 15–25 total simulations, compared to the 12 used in the simulation-only optimization.

The objective function trajectories and training set RMSEs of the replicates starting from $N = 5$ initial points are shown in Fig. 4, with the training set RMSEs of OpenFF 1.0.0 and the optimized set from our previous work shown for comparison.

We see that in most cases, the optimizer struggles initially, with a high percentage of proposed solutions rejected in the first 8 steps. While these steps do not immediately yield an improved force field, the parameter vectors they propose are added to the pool of parameter vectors used to build surrogates, eventually exploring enough space to find an improved solution, with an average objective of 0.039 *vs.* an initial objective of 0.16, an average $\rho_L$ training set RMSE of 0.016 g mL$^{-1}$ (initial RMSE 0.027 g mL$^{-1}$), and an average $\Delta H_{vap}$ RMSE of 2.75 kJ mol$^{-1}$ (initial RMSE 7.15 kJ mol$^{-1}$).

In order to find these improved solutions, the optimization algorithm searches widely and finds a number of qualitatively distinct minima. The optimization trajectories in parameter space, as well as the trajectory from the simulation-only optimization against the same training set, are shown in Fig. 5.

In comparison to the simulation-only optimization, shown in brown, the replicates of our multi-fidelity optimization search the parameter space much more broadly. Particularly, the values of oxygen and carbon parameters stay within a narrow range in the simulation-only optimization, but vary widely with our multi-fidelity technique.

The training set RMSEs and objective functions for the $N = 10$ runs are shown in Fig. 6.

We note that the optimizations initialized with $N = 10$ initial parameter vectors improve the objective function with fewer iterations that the $N = 5$ optimizations. The $N = 10$ optimizations leverage the additional initial information to build more accurate surrogates, finding improved parameter sets sooner (at the expense of higher inital cost). The effectiveness of the optimization is also slightly improved over the $N = 5$

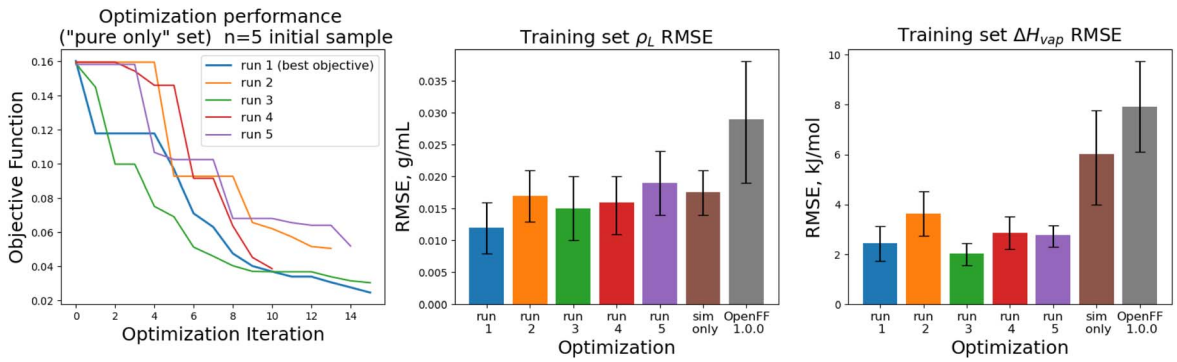


Fig. 4 Performance of the multi-fidelity optimization algorithm on the "pure only" training set, for replicates run with $N = 5$ initial parameter vectors. Left panel shows the objective function at each iteration of the optimization. Right two panels show the training set RMSE for $\rho_L(x)$ and $\Delta H_{mix}(x)$ for each of the optimizations, as well as OpenFF 1.0.0 and the previous simulation-only optimization (labeled "sim only" in the graphs). Error bars represent 95% confidence intervals, computed with bootstrapping over the set of molecules in the training set.

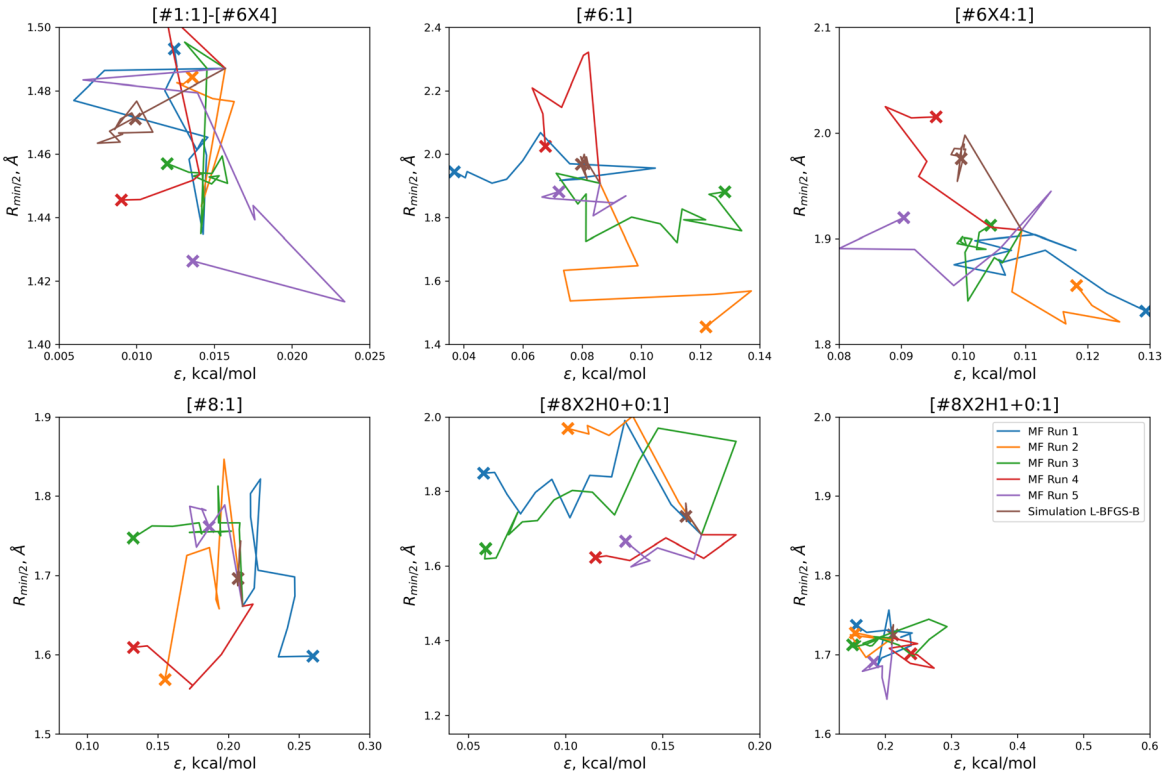


Fig. 5 Parameter-space optimization trajectories for each of the replicates run with 5 initial parameter vectors, as well as the trajectory for the previous simulation-only optimization (brown). Trajectories show that our optimization technique searches widely and finds many distinct solutions for this optimization problem. Plot limits are shared with Fig. 7 for ease of comparison.

optimizations, with an average objective function of 0.031 ($N = 5$: 0.039), an average $\rho_L$ training set RMSE of 0.014 g mL$^{-1}$ ($N = 5$: 0.016), and an average $\Delta H_{vap}$ RMSE of 2.38 kJ mol$^{-1}$ ($N = 5$: 2.75 kJ mol$^{-1}$).

With the exception of two of the runs, the parameter trajectories in the initial $N = 10$ optimizations explore a similar range of parameters to the $N = 5$ optimizations, as shown in Fig. 7. In contrast, runs 3 and 4 make very large changes to some of the parameters, drastically deviating from the initial parameter set.

Particularly in the oxygen parameters for runs 3 (green) and 4 (red), we can see that these optimizations can make some very large parameter changes; run 3 has the lowest overall objective, but more than doubles the hydroxyl oxygen $\varepsilon$. This may suggest that adding some regularization could benefit the transferability of the optimization, but it also reflects that the ratio of targets to inputs (56:12) leads to an optimization where many solutions can be found.

**3.1.2 Parameter interpretation.** Since the optimizations find diverse solutions, and the set of parameters is small enough

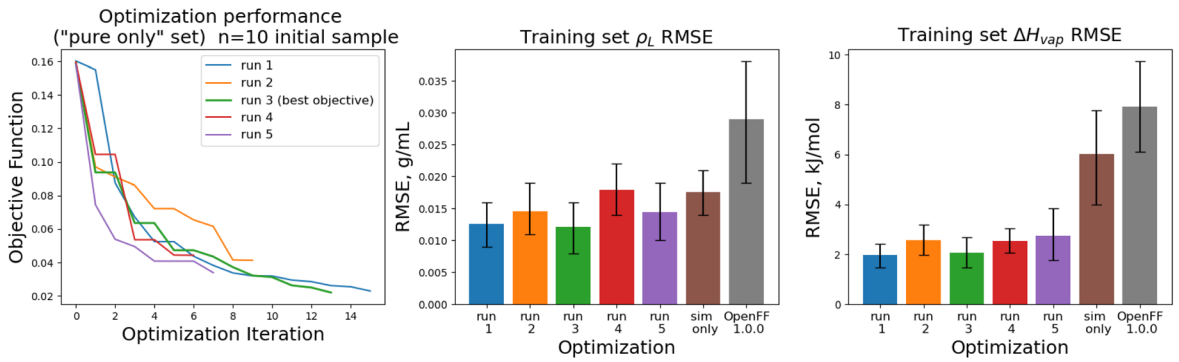


Fig. 6 Performance of the optimization algorithm on the "pure only" training set, for replicates run with $N = 10$ initial parameter vectors. Left panel shows the objective function at each iteration of the optimization. Right two panels show the RMSE for $\Delta H_{mix}(x)$ and $\rho_L(x)$ for the two optimizations, as well as OpenFF 1.0.0 and the previous simulation-only optimization (labeled "sim only" in the graphs). Error bars represent 95% confidence intervals, computed with bootstrapping over the set of molecules in the training set.

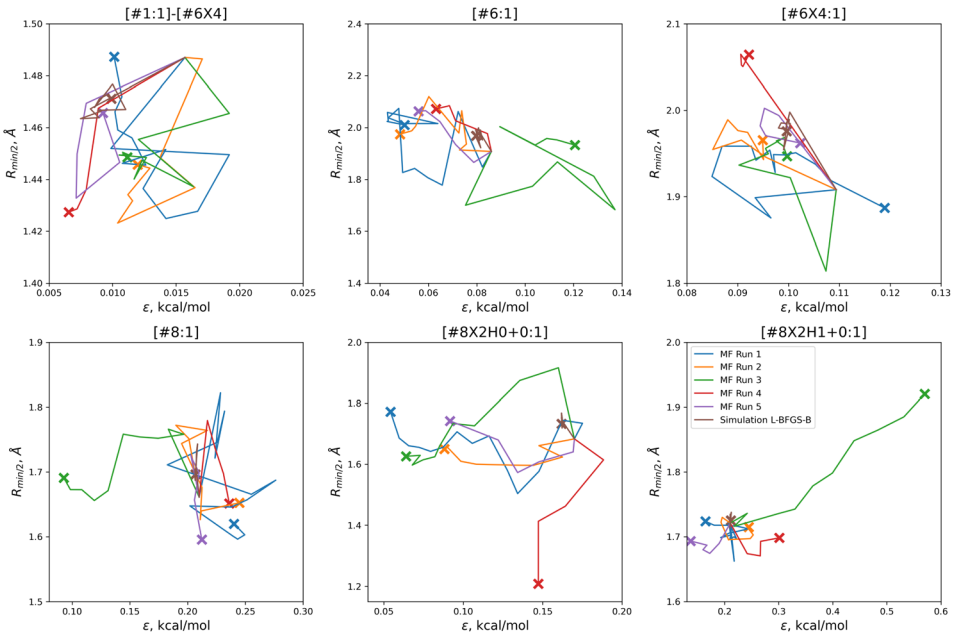


Fig. 7 Optimization trajectories in parameter space for each of the replicates run with $N = 10$ initial parameter vectors. Trajectories show that our $N = 10$ initial parameter vector optimization technique searches more widely than our $N = 5$ technique, but is also more likely to drastically alter a parameter while deeply exploring a potentially promising candidate, as shown for runs 3 (green) and 4 (red). Plot limits are shared with figure for ease of comparison.

to be reasonably interpretable, it is worth examining some of the parameter changes to understand their physical basis and inform future parameter fitting. Here we analyze some of the most notable changes from the $N = 10$ replicates, which had lower objective functions relative to the $N = 5$ replicates. The change in parameters from the original OpenFF 1.0.0 is shown in Fig. 8.To identify what points in the training dataset the parameter changes are affecting, we examine the bias of the physical properties training dataset, as measured by mean signed deviation (MSD) from experiment. To avoid confusion, we use the acronym MSD to refer to this bias, while RMSE refers to the root mean squared error. The bias before and after training for each chemical group is plotted for multi-fidelity run 1, shown in Fig. 9. Run 1 is shown as an example as it has one of the best objective functions and no unusual parameter changes (such as occurred in runs 3 and 4), and MSD values are similar for all optimization runs. Similar plots of RMSE and MSD for all 5 runs are available in the ESI, Section S2.2.†

One consistent trend in most optimizations is the significant overall reduction of $\varepsilon$ for most parameter types. By decreasing

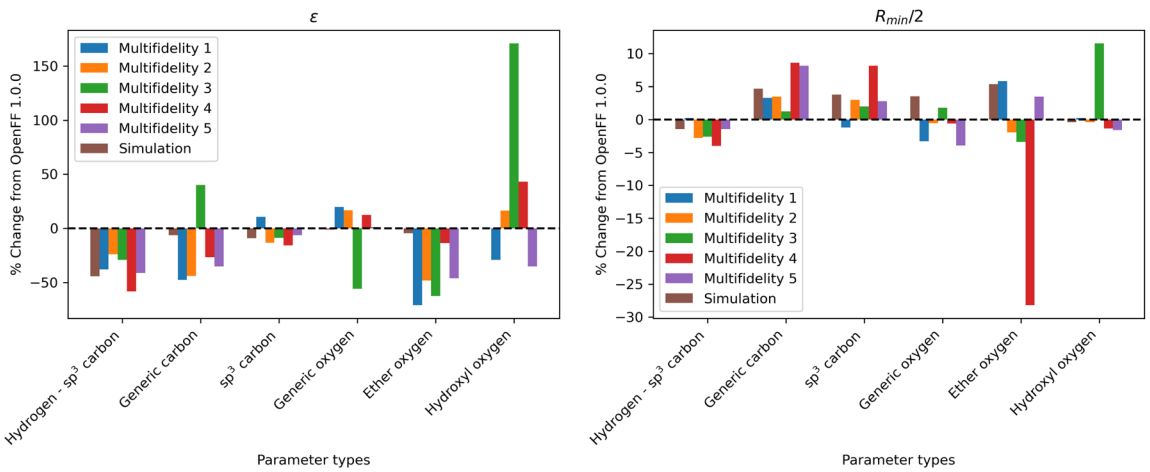


Fig. 8 Changes in parameter values after optimization against the "pure only" data set, relative to OpenFF 1.0.0 (the initial values of the optimization), for the $N = 10$ optimization replicates, as well as the simulation-only solution previously obtained. Parameter changes are typically larger in multi-fidelity optimization compared to the simulation-only optimization, indicating improved exploration of the parameter space; however, this also leads to outliers (hydroxyl oxygen $\varepsilon$, ether oxygen $R_{min}/2$).

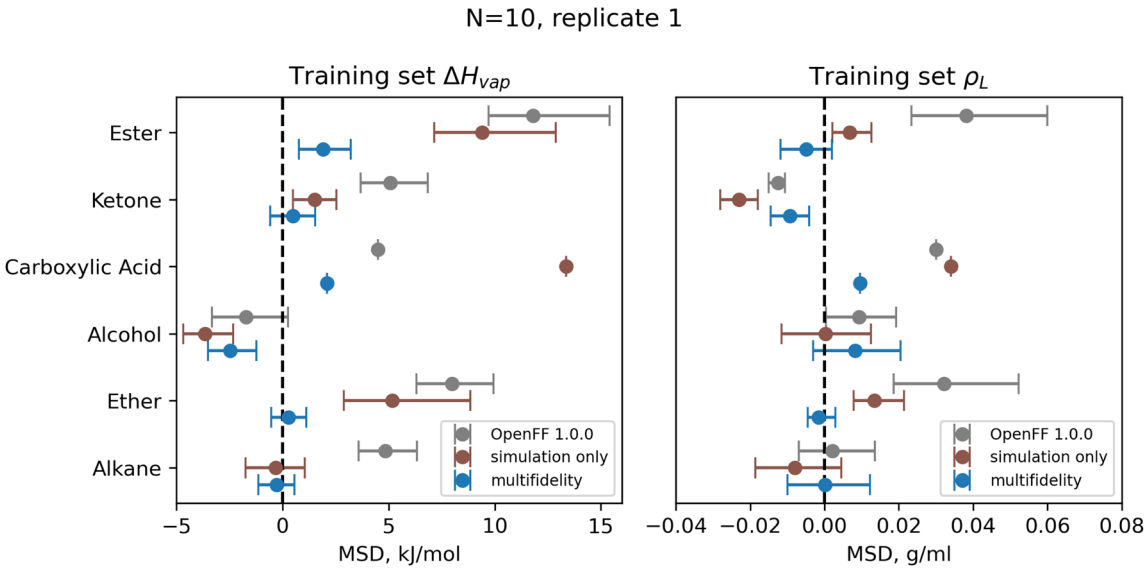


Fig. 9 Bias in training set $\Delta H_{vap}$ and $\rho_L$ by chemical functionality, as measured by the mean signed deviation (MSD), for OpenFF 1.0.0 and retrained parameters from $N = 10$ multi-fidelity run 1. Training with multi-fidelity optimization reduces or eliminates bias in several chemical functionalities, including alkanes, ketones, and ethers, where MSD is near 0 kJ mol$^{-1}$. Reduction in MSD shown for run 1 is typical of all "pure only" $N = 10$ multi-fidelity runs; training set MSDs and RMSEs for all multi-fidelity optimization are available in ESI, Sections 2.1–2.3.† Error bars represent bootstrapped 95% confidence intervals.

the $\varepsilon'$s, cohesive forces in the liquid phase are reduced, lowering the barrier for "liberating" a molecule from the gas phase and thereby lowering the enthalpy of vaporization. In OpenFF 1.0.0, the enthalpy of vaporization measurements in the training set have a positive bias (MSD) of 5.03 kJ mol$^{-1}$, with all moieties except alcohols having a positive deviation from experiment. After multi-fidelity optimization, the training sets have an average MSD (across all multi-fidelity runs) of 0.22 kJ mol$^{-1}$.

The trend of reduced $\varepsilon'$s is strongest for the [#1:1]-[#6X4] (hydrogen attached to tetravalent carbon) and [#8X2H0+0:1] (divalent oxygen with 0 hydrogens attached) atom types. The [#1:1]-[#6X4] type is exercised in all molecules in the training set, so reducing the $\varepsilon$ for this type helps to reduce this overall bias. For the alkanes in the set, [#1:1]-[#6X4] is one of two parameters exercised (along with [#6X4:1], tetravalent carbon), and alkane $\Delta H_{vap}$ training set MSD is reduced from 4.75 kJ mol$^{-1}$ in OpenFF 1.0.0 to an average of −0.01 kJ mol$^{-1}$, virtually eliminating the error. Reducing the $\varepsilon$ of the [#8X2H0+0:1] (ether oxygen) type helps to correct a significant overprediction of ether $\Delta H_{vap}$ in OpenFF 1.0.0, reducing the ether MSD from 7.95 kJ mol$^{-1}$ to an average value of 0.54 kJ mol$^{-1}$ after training. This reduction in error is much larger than the reduction observed after simulation-only local optimization.

The $\varepsilon'$s for [#6:1] (generic carbon) and [#8:1] (generic oxygen) present an interesting case in multidimensional optimization. We see significant changes in the $\varepsilon'$s for both [#6:1] and [#8:1]; however, the presence of more specific types in the training set means that these two types are only exercised together in a C=O double bond (a ketone, ester, or carboxylic acid). In all optimizations but run 3, we see a large reduction in [#6:1] $\varepsilon$ and a slight increase in [#8:1] $\varepsilon$. The adjustment of these parameters, along with an increase in the [#6:1] $R_{min/2}$, corrects an overprediction in the ester and ketone $\Delta H_{vap}$. Notably, simulation-only optimization against the same training set was not able to achieve the same correction for esters.

Interestingly, run 3 takes an opposite approach, increasing $\varepsilon$ for [#6:1] and decreasing $\varepsilon$ for [#8:1] but achieving a similar reduction in bias. This suggests that, for the purpose of this optimization, [#6:1] and [#8:1] are treated as a unit. This is not desirable in a larger context, as these parameters can appear separately in other chemical moieties, such as an alkene for [#6:1]. They are not inherently coupled and will probably lead to statistically significant errors if used in other contexts.

For $R_{min/2}$, the most consistent changes are in [#6:1] and [#6X4:1], which are generally increased. Overall, the effect of increasing $R_{min/2}$ should be to decrease density, as it increases inter-atomic distances and leads to higher molecular volume. This is consistent with the physical properties, as densities are slightly overpredicted in OpenFF 1.0.0, but those overpredictions are concentrated in ethers and esters. The increase in $R_{min/2}$ for [#6:1] in particular helps to reduce a significant overprediction of ester densities.

**3.1.3 Surrogate analysis.** Given that the optimizations produce diverse collection of parameter sets rather than converging on a single set, it is useful to characterize the quality of the surrogate models over the parameter space. Specifically, we compare the global accuracy of the surrogate models and measure the roughness over the surrogate models by running multiple minimizations on the final surrogate models. We performed this analysis for the $N = 10$ optimization runs.

To assess the ability of the surrogate to make accurate predictions outside the region of its minimization, we calculated objective functions with the surrogate produced in each $N$

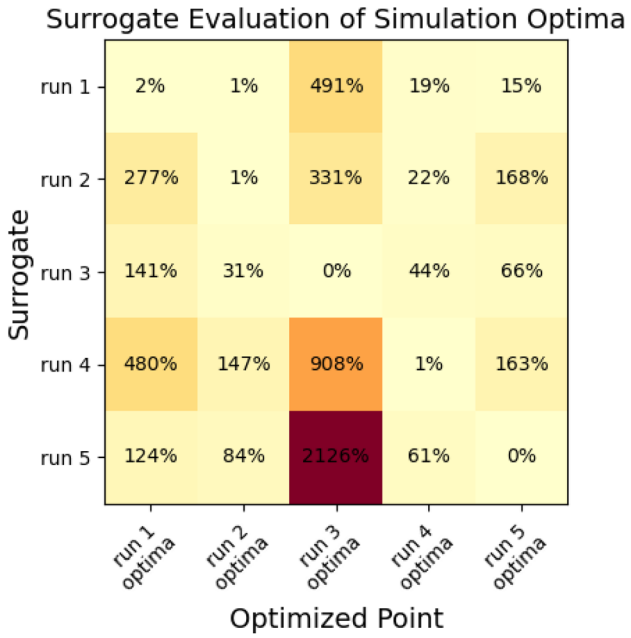


Fig. 10 Cross-validation demonstrates that surrogates produced in the multi-fidelity optimization process are only locally predictive. Figure shows % deviation between surrogate-predicted objective functions and simulation objective functions for each of the five surrogate models and five optimization minima produced for the $N = 10$ "pure only" optimization runs.

$= 10$ optimization for the parameter sets produced from all other $N = 10$ optimizations. If the surrogates were globally predictive, we would observe low prediction error for the other minima; with high error, surrogates are likely only locally predictive. Results are shown in Fig. 10.

This analysis indicates that surrogates produced as a part of a multi-fidelity optimization are usually only locally predictive, as many have very large prediction errors for objective functions at some of the other minima. The surrogate that performs best is the surrogate from run 1; which estimates the objective to within 20% of the simulation value for all cases besides the optima from run 3, which is far away from the region where other surrogates have samples.

We also assessed the robustness of the surrogate by performing repeated L-BFGS-B optimization on the final produced surrogates, starting from random points within the final parameter bounds box used in the optimization. This characterizes the smoothness and multimodality of the produced surrogates, as a smooth, unimodal surrogate would lead to a highly consistent local optimization, whereas a rough and multi-modal surrogate would produce different outcomes.

For the surrogate produced in each $N = 10$ multi-fideity optimization, we ran 100 L-BFGS-B optimizations from random starting points. For each of these optimizations, we calculated the standard deviation of the objective among the 100 minima ($SD_\chi$) and the percentage of optimization within 5% of the best objective ($O_{5\%}$). The results are shown in Table 3, along with the number of simulations used to build each surrogate ($K_{sim}$).

Table 3 Metrics of optimization consistency for 100 L-BFGS-B optimizations starting from random points within the bounds box for the surrogates produced in each of the 5 $N = 10$ "pure-only" optimizations. $SD_\chi$ is standard deviation of resulting minimized objective functions, $O_{5\%}$ indicates the percentage of optimizations within 5% of the best objective for that surrogate, and $N_{sim}$ indicates number of simulations used to build the optimization

| Surrogate | $SD_\chi$ | $O_{5\%}$ | $K_{sim}$ |
|---|---|---|---|
| Optimization run 1 | 0.0002 | 99 | 25 |
| Optimization run 2 | 0.017 | 63 | 20 |
| Optimization run 3 | 0.001 | 72 | 24 |
| Optimization run 4 | 0.005 | 94 | 16 |
| Optimization run 5 | 0.003 | 49 | 17 |

The results vary widely based on the surrogate, indicating that some surrogates are more robust than others. Particularly, the surrogates from optimizations 1 and 3 have the most consistent local optimizations, even though optimization 3 produces a large parameter set outlier, with low standard deviations and ranges. These two surrogates also use the most simulation data and come from optimizations that more deeply explored their local optima, indicating that more sampling in the region of interest leads to a smoother, unimodal surrogate. Conversely, optimizations 2, 4 and 5 spend less time exploring the target region and have rougher surfaces, with L-BFGS-B optimizations less likely to converge. This indicates that exploring the target region in detail builds a more robust surrogate.

**3.1.4 Benchmarking.** We performed benchmarking on the test set described in Section 2.5 for OpenFF 1.0.0, the simulation-only optimization against the "pure only" set, and the 5 multi-fidelity optimization runs with $N = 10$ initial points. The benchmarking set is described in Section 2.5. We focused on the $N = 10$ runs because they generally had better objective function performance compared to the $N = 5$ runs, and they also had larger parameter changes, meaning that they would be more susceptible to overfitting. RMSE statistics for all of these force fields are plotted in Fig. 11.

These results highlight the need to test for transferability, as run 3, which had the lowest objective function over the training set, performs worse than OpenFF 1.0.0 in three of the four physical property data types in the test set. This is likely caused by the very large changes in the [#8X2H1+0:1] (hydroxyl oxygen) parameters. Similarly, run 4 performs poorly on the test set after significant changes to the [#8X2H0+0:1] $R_{min/2}$ parameter.

For optimization runs without these outlier changes to parameters, such as run 1, the results are improved, with a decrease in test set $\Delta H_{vap}$ from an initial value in OpenFF 1.0.0 of 7.52 kJ mol$^{-1}$ (95% CI 6.42, 8.53) to a value after fitting of 3.41 kJ mol$^{-1}$ (95% CI 1.94, 4.68), outperforming the simulation-only optimization value of 5.25 kJ mol$^{-1}$ (95% CI 4.31, 6.16). This improvement in $\Delta H_{vap}$, following the improvement in the training set, suggests that the more aggressive multi-fidelity optimization was able to adjust parameters in a way that results in better prediction of $\Delta H_{vap}$. For the other properties in the test set, run 1 improves over

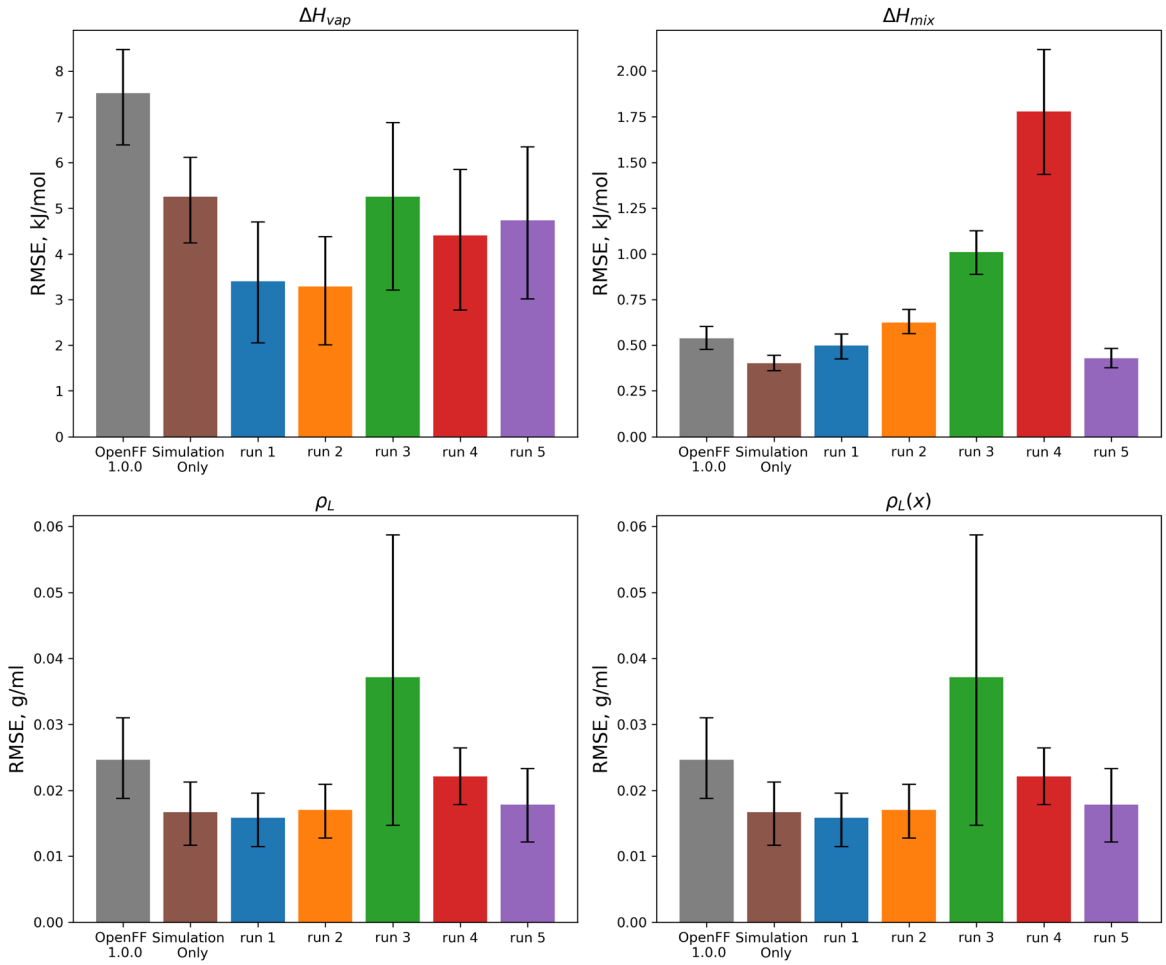


Fig. 11 Test set RMSE for OpenFF 1.0.0, the previous simulation-only optimization, and the 5 $N = 10$ multi-fidelity runs for "pure only" targets. Benchmarking shows that some runs (such as runs 1 and 2) are transferable and outperform the simulation-only optimization, but that other runs (runs 3 and 4) have poor performance on the test set, likely due to overfitting. Error bars represent 95% confidence intervals, bootstrapped over the set of molecules in the test set.

OpenFF 1.0.0 in each case, and is slightly improved over the simulation-only optimization for $\rho_L$ and $\rho_L(x)$. The optimization does perform slightly worse than the simulation-only optimization for $\Delta H_{mix}(x)$, which likely reflects the lack of regularization and overfitting in the surrogate model due to improved prediction of $\Delta H_{vap}$, which depends on vapor phase properties as well as condensed phase properties.

We also see that some of the functional-group specific reductions in the training set RMSE translate to the test set; indeed, most of the reduction in test set RMSE comes from improved treatment of ethers and esters, indicating that the parameter changes that led to these changes, such as significant reduction of ether $\varepsilon$, are transferable. Plots of the test set RMSE for all functional group categories are available in the ESI, Section 3.1.†

These results demonstrate that we can find improved parameter sets using multi-fidelity global optimization, but that care must be taken to avoid overfitting. We performed 5 optimization runs that all significantly improved the objective, but with large variations in the parameter vector solutions. There are a wide range of parameter vectors that are able to satisfy this optimization problem, but their transferability is not guaranteed. This is probably due to the training set which was chosen, as the set contains 56 target points and 12 parameters; additionally, the parameter set is "segmented" in that some parameters and targets are independent from the other parameters/targets. For example, the [#8X2H0+0:1] (ether oxygen) parameters are only dependent on the measurements of $\rho_L$ and $\Delta H_{vap}$ for ethers. This leads to an optimization where overfitting is a significant concern. We could address overfitting by using a regularization scheme, as many others have done, but this many prevent us from escaping local minima in parameter space, as we hoped to do.

More physically, we can also address overfitting by broadening the training set, as more physical property targets will further constrain the optimization. In addition, including mixture data in the training set helps to guard against overfitting, given that the set becomes less segmented, as physical properties of mixtures exercise a wider range of parameters than pure physical properties.

### 3.2 Mixture training set

Implementing multi-fidelity optimization with the mixture training provides us with an opportunity to test whether we can better constrain the training data without implementing regularization, since our set of mixture data is much larger, containing 195 physical property measurements. We ran an optimization with $N = 20$ initial points, as well as one with $N = 10$ initial points, to determine what level of initial information was required to produce a successful optimization.

Optimization against the "mixture-only" training was successful for both the $N = 10$ and $N = 20$ runs set, achieving significant reductions in objective function and training set RMSE, as shown in Fig. 12. The $N = 20$ optimization uses a total of 34 simulation evaluations to find its optimum, whereas the $N = 10$ optimization uses only 17.

In both optimizations we observe a large drop in the objective function, followed by incremental progress until the end of the optimization. The performance is slightly improved compared to the regularized simulation-only optimization; for the $N = 20$ run the training set $\Delta H_{\mathrm{mix}}(x)$ RMSE is 0.19 kJ mol$^{-1}$ (95% CI 0.16, 0.22) *versus* 0.24 kJ mol$^{-1}$ (0.21, 0.29) for the simulation-only optimization. For $\rho_{\mathrm{L}}(x)$, the RMSE is 0.011 g mL$^{-1}$ (0.01, 0.013) *versus* 0.013 (0.011, 0.015). Both optimizations are significantly improved when compared to OpenFF 1.0.0, with $\Delta H_{\mathrm{mix}}(x)$ RMSE of 0.62 kJ mol$^{-1}$ (0.54, 0.69) and $\rho_{\mathrm{L}}(x)$ RMSE of 0.23 g mL$^{-1}$ (0.02, 0.026).

While the performance is slightly improved, the parameter changes are more significant when compared to the simulation-only optimization. The changes in parameter value from OpenFF 1.0.0 are shown in Fig. 13.

Again, the changes in parameters are larger for the multi-fidelity optimizations when compared to the simulation-only optimization, particularly for the values of $\varepsilon$. However, when compared to the multi-fidelity optimization against the "pure only" training set, the changes are smaller and there are not

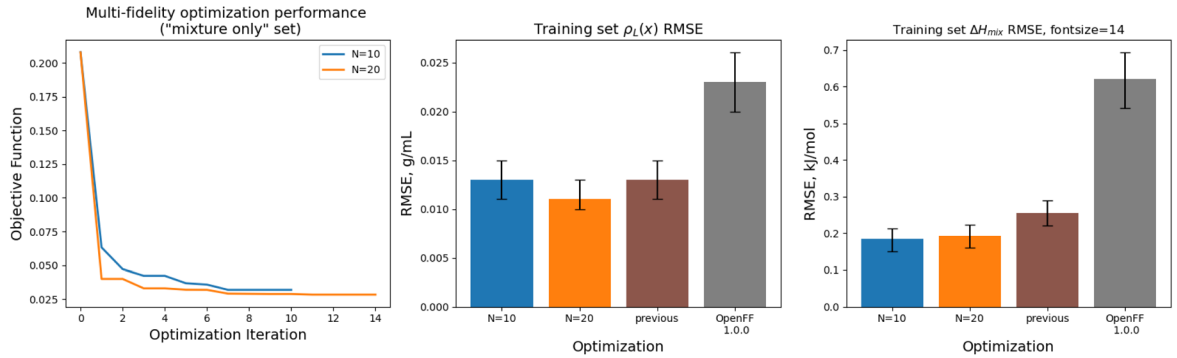


Fig. 12 Performance of the multi-fidelity optimization algorithm on the "mixture only" training set, for runs with $N = 10$ (blue) and $N = 20$ (orange) initial parameter vectors, indicating improved performance on $\Delta H_{\mathrm{mix}}(x)$ targets when compared to simulation-only optimization. Left panel shows the objective function at each iteration of the optimization. Right two panels show the RMSE for $\Delta H_{\mathrm{mix}}(x)$ and $\rho_{\mathrm{L}}(x)$ for the two optimizations, as well as OpenFF 1.0.0 and the previous simulation-only optimization. Error bars represent 95% confidence intervals, computed with bootstrapping over the set of molecules in the training set.

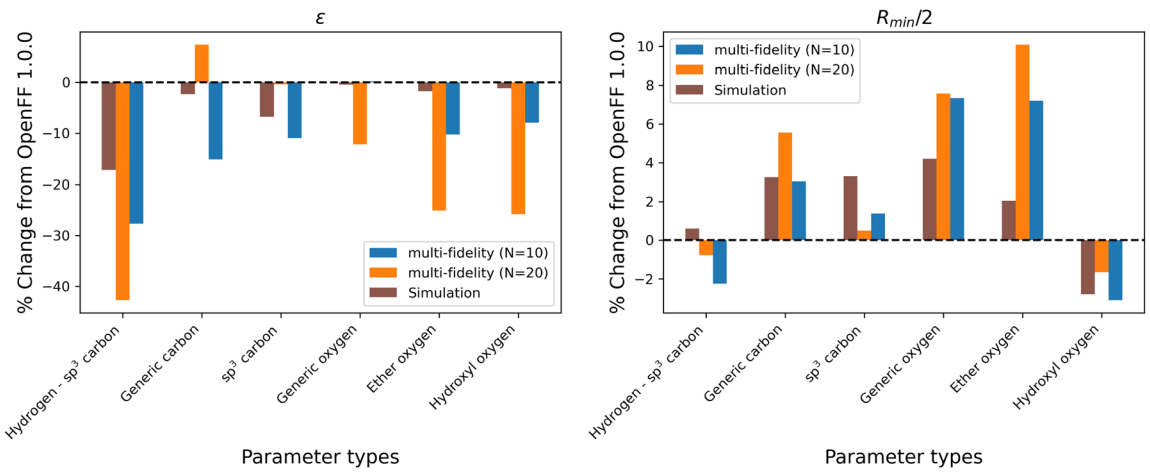


Fig. 13 Changes in parameter values after optimization against the "mixture only" data set, relative to OpenFF 1.0.0 (the initial values), for optimizations with both $N = 10$ and $N = 20$ initial points, as well as the simulation-only solution previously obtained. Multi-fidelity optimizations show larger parameter changes, particularly in $\varepsilon$, when compared to the simulation-only optimization over the same training set, but smaller changes in parameter values when compared to the "pure-only" multi-fidelity optimizations.

significant outliers. We see some of the same parameter trends as in the "pure only" optimizations, like reduced values of $\epsilon$ for the [#1:1]-[#6X4] (hydrogen attached to tetravalent carbon) and [#8X2H0+0:1] (ether oxygen) atom types. A notable difference is the increased $R_{min/2}$ for the [#8:1] (generic oxygen) type, which is likely related to mixture properties better capturing the hydrogen bond donor/acceptor behavior of alcohol/ester mixtures.[27]

**3.2.1 Benchmarking.** We assessed the performance of the refit force fields on the test set. Plots of RMSEs for the four types of physical property data in the test set are shown in Fig. 14.

Between the simulation-only and multi-fidelity optimizations, performance is similar on pure and mixture densities ($\rho_L$ and $\rho_L(x)$), and not significantly improved when compared to OpenFF 1.0.0; densities are already well-predicted in OpenFF 1.0.0. For $\Delta H_{vap}$ and $\Delta H_{mix}(x)$, the multi-fidelity optimizations significantly improve both properties, whereas the simulation-only optimization only improved $\Delta H_{mix}(x)$. The $N = 20$ optimization has a $\Delta H_{mix}$ RMSE of 0.24 kJ mol$^{-1}$ (95% CI 0.22, 0.26), similar to 0.25 (0.23, 0.27) for the simulation-only optimization. For $\Delta H_{vap}$, the $N = 20$ optimization has an RMSE of 4.83 kJ mol$^{-1}$ (3.75, 5.82), significantly improved over the simulation-only optimization value of 7.87 kJ mol$^{-1}$ (6.61, 9.14), which has slightly regressed prediction of $\Delta H_{vap}$ compared to the original force field. It is notable that we are able to achieve significantly improved performance on both types of enthalpy data in the test set, indicating that parameters found with the multi-fidelity optimization process achieve better transferability than a simulation-only optimization against the same data set. Examining the test set results for the $N = 20$ run separated by functional group, as shown in Fig. 15, helps to better understand how parameter changes influence force field performance.

A notable result is that esters and ketones perform better on $\Delta H_{mix}(x)$ and $\Delta H_{vap}$ in this refit force field compared to their performance with the simulation-only refit. This may be due to changes in the parameters (bigger increases in $R_{min/2}$) for the [#8:1], which is exercised only by carbonyl oxygens in both the training and test sets. However, this parameter change has an interesting effect on densities; while ester densities and

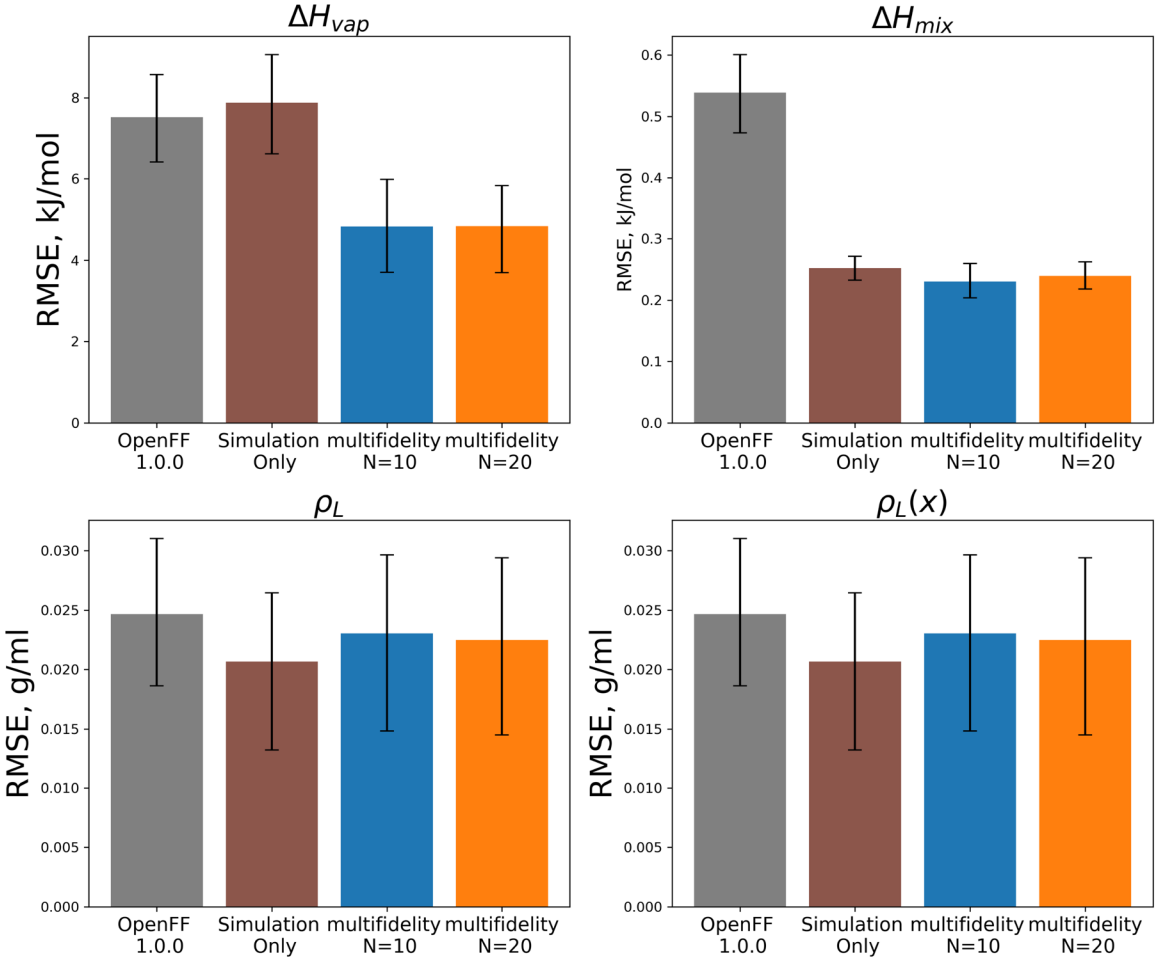


Fig. 14 Test set RMSE for OpenFF 1.0.0, the simulation-only optimization, and both multi-fidelity optimization runs, against the "mixture only" target. Benchmarking on the test set shows the transferability of the optimized parameters; notably RMSE for both $\Delta H_{mix}(x)$ and $\Delta H_{vap}$ are significantly improved compared to OpenFF 1.0.0, despite $\Delta H_{vap}$ not being included in the training, in contrast, simulation-only optimization does not improve $\Delta H_{vap}$. Error bars represent 95% confidence intervals, bootstrapped over the molecules in the test set.

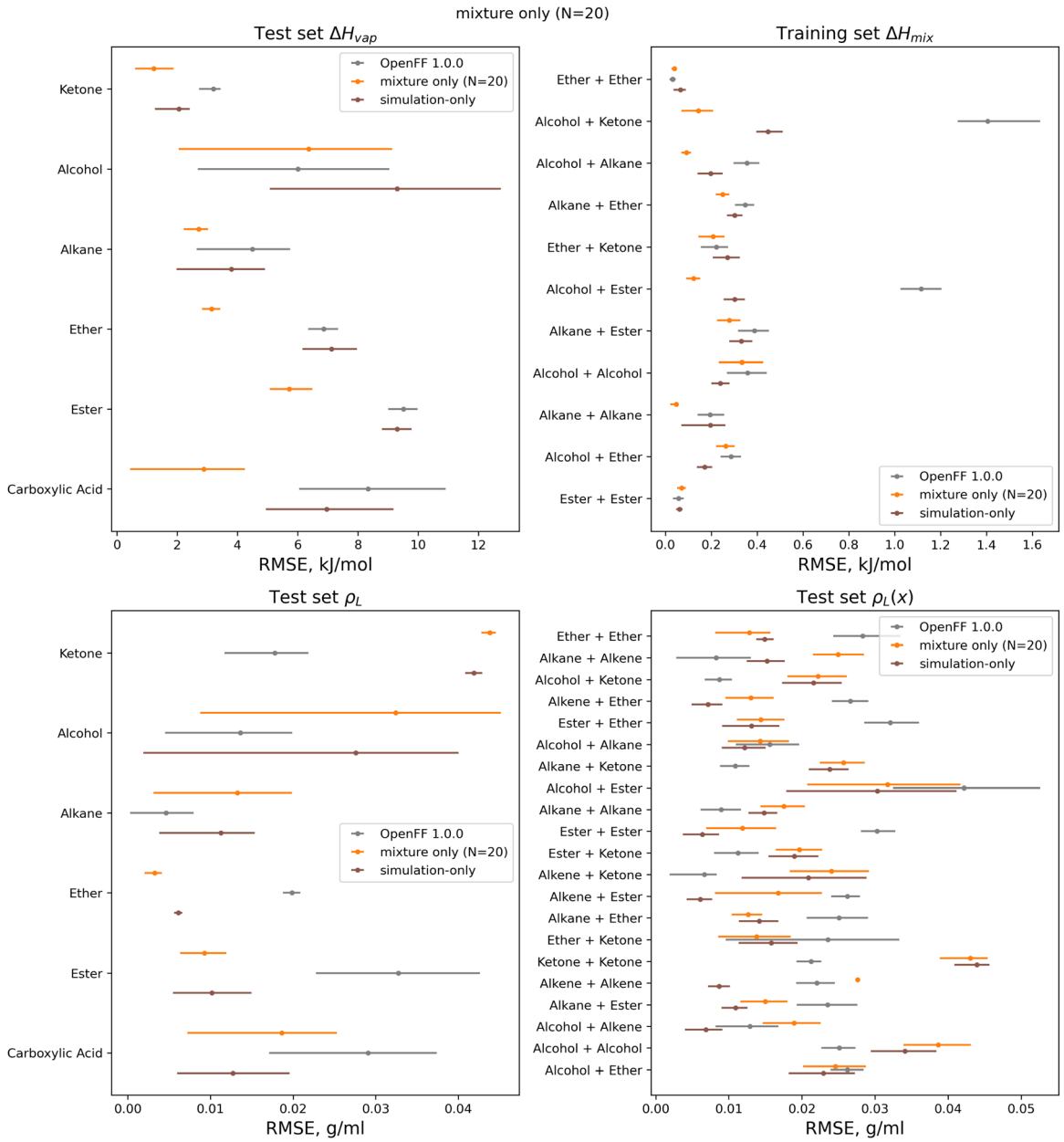


Fig. 15 Test set RMSE for OpenFF 1.0.0, the simulation-only optimization, and both multi-fidelity runs, against the "mixture only" target and separated by function group or functional group pair. Benchmarking highlights important parameter changes and opportunities to tune the atom types in the model, including an apparent improvement of densities of ester mixtures at the expense of ketone mixture densities. Error bars represent 95% confidence intervals, bootstrapped over the molecules in the test set.

densities of ester-containing mixtures are largely improved, the ketone densities regressed significantly, both with multi-fidelity optimization and simulation-only optimization. This, along with parameter gradient evidence from our previous work, suggests that splitting LJ types responsible for ketones and esters may yield improved prediction of densities.

An area where the multi-fidelity optimization struggles is in the prediction of alcohols, where outside of mixtures with the improved esters/ketones, predictions are either slightly improved or degraded. While the simulation-only optimized parameters perform slightly better on alcohols, they see similar regressions in the predictions of alcohol and alcohol-mixture densities. This may due to deficiencies in the AM1-BCC charge model for alcohols,[29] leading to compensation in LJ parameters and reduced transferability.

Another type where splitting may yield improved results is the [#6:1] (generic carbon) type. In the training set; this type is only exercised by carbonyls, but in the test set this type is exercised by both carbonyls and alkenes. While the changes introduced in the multi-fidelity optimization significant improve performance of carbonyl-containing molecules, binary densities of alkene mixtures are significantly degraded,

indicating that the type may no longer be suitable for describing both contexts. This is sensible as carbonyls and alkenes are quite different in their chemistry.

# 4 Conclusions

We present a new approach for large-scale optimization force field parameters against physical property data, based on equilibrium simulations and Gaussian process surrogate modeling. Our multi-fidelity strategy uses an iterative process of global optimization over the cheap surrogate surface and validation performed at the simulation level. We demonstrate that for reasonably sized sets of physical property data, multi-fidelity optimization can find improved parameter sets while exploring more widely than traditional local optimization techniques. Training against binary mixture data, our optimization makes larger parameter changes for ethers and carbonyls, which yield transferable improvements on test set measurements of both $\Delta H_{vap}$ and $\Delta H_{mix}(x)$. Training against the same dataset using only local simulation-based optimization is able to achieve comparable improvements on $\Delta H_{mix}(x)$, but not $\Delta H_{vap}$. Through examination of the training and test data, we are also able to identify targets for parameter type splitting.

While this strategy shows promise, challenges in implementation remain; one of the largest being the stochastic nature of the method. The improved parameters found are highly dependent on the set of initial parameter simulations used to build the surrogate model; the parameter space is rough and high-dimensional, meaning that Latin hypercube sampling struggles to find good starting sets of parameters. Building a better initial surrogate also requires more initial simulations, incurring higher computational expense. Analysis of the surrogates produced in the multi-fidelity process indicates that they are locally predictive models best suited to accelerating optimization, rather than global models accurate across the entire parameter space.

A potential route to improvement for this strategy is to incorporate Bayesian optimization[73] into the parameter search strategy in order to acquire test points more efficiently. Bayesian optimization is generally efficient at solving expensive optimization problems with a limited number of objective function evaluations. Starting with a smaller and more restricted set of initial parameters and allowing Bayesian optimization to acquire samples, could lead to a more efficient and reproducible optimization.

Another target area for improvement is the robustness of the surrogate building process; roughly 50% of the optimization terminate early, as issues with automatic relevance determination (ARD) cause surrogates to sacrifice accuracy to the point where they can no longer find an improved solution. While all optimization runs still led to improved parameters overall, this suggests that parameter quality could be higher with improved surrogates. One surrogate modeling best practice that we did not incorporate into this workflow is parameter and output normalization,[51] which can improve surrogate performance and reduce the risk of failures due to ill-conditioning, potentially yielding a more robust surrogate.

Surrogate quality could also potentially be improved by incorporating additional information from the simulations into the surrogate input. Derivative information can be obtained by reweighing[31] and can be used to better inform the surrogate model.[74,75] More generally, reweighting in parameter space can provide significant additional information about the surrogate model points in the local region of the simulation point.[76,77] Reweighting in local parameter space after each simulation iteration would add relatively low computational overhead to the algorithm, compared to the cost of the simulations and optimization. Additionally, more investigation into the specifics of the optimization algorithm used over the surrogate (either the hyperparameters for the differential evolution algorithm, or other optimization algorithms entirely) could yield more efficient and consistent optimization performance.

The production of some parameter sets with drastic changes that improve training set RMSE, but are not transferable, demonstrates that overfitting is a significant risk when using more effective parameter optimization techniques. Typically, this risk is mitigated with regularization, penalizing solutions that stray too far from the initial solution. In this study, since we are interested in escaping local minima, we did not regularize the optimization; this led to the discover of some significantly improved parameters (lower values of $\varepsilon$ for ether oxygens, higher values of $R_{min/2}$ for carbonyl oxygens) that represented much larger changes that what regularized optimizations produced. Results from multi-fidelity optimizations on mixture properties indicate that a more complex training target can also serve to constrain an optimization and improve transferability, while allowing parameters to vary considerably. To further assess the transferability of the parameters, it would be illuminating to examine their performance on more expensive-to-compute properties, such as solvation free energies or binding free energies, as previous studies have shown that improved mixture properties give rise to improved solvation free energies[27,34] and at least do not hurt binding free energies.[34]

This surrogate-based optimization method can be used with global optimization methods and is able to improve force field LJ parameters by escaping local minima, leading to both chemical insight and improved parameters. The success of the strategy is due to its multi-fidelity approach, using a cheaper surrogate to apply an otherwise prohibitively expensive global optimization algorithm. While already useful in its current form, the flexibility of the framework allows for significant improvement of the strategy in the future. We believe that this technique can help modelers perform better optimizations against physical property data, leading to force fields which more accurately predict the behavior of molecular systems of interest.

# Data and code availability

Software used in this paper, as well as simulated physical property datasets used to build surrogate models, are available at **https://github.com/ocmadin/LJ_surrogates**. To provide feedback on performance of the OpenFF force fields, we highly recommend using the issue tracker at **http://**

github.com/openforcefield/openforcefields. For toolkit feedback, use http://github.com/openforcefield/openforcefield. Alternatively, inquiries may be e-mailed to support@openforcefield.org, though responses to e-mails sent to this address may be delayed and GitHub issues receive higher priority. For information on getting started with OpenFF, please see the documentation linked at http://github.com/openforcefield/openforcefield, and note the availability of several introductory examples. The code for surrogate model creation, optimization, data handling for this study can be found at https://github.com/ocmadin/LJ_surrogates. This repository also includes outputs of the calculations of physical properties from simulation used to build surrogate models. The version of the code employed for this study is this commit version: https://github.com/ocmadin/LJ_surrogates/commit/d7f94153801eb8c0673fe2e62c950e5beed1e999.

## Author contributions

Conceptualization: O. M., M. R. S. Methodology: O. M. Software: O. M. Investigation: O. M. Validation: O. M. Formal analysis: O. M. Data curation: O. M. Writing – original draft: O. M. and M. R. S. Writing – review & editing: O. M. and M. R. S. Visualization: O. M. Supervision: M. R. S. Project administration: M. R. S. Funding acquisition: M. R. S.

## Conflicts of interest

The authors declare the following competing financial interest(s): MRS is an Open Science Fellow with Psivant Sciences and consults for Relay Therapeutics.

## Acknowledgements

Research reported in this preprint was in part supported by the National Institute of General Medical Sciences of the National Institutes of Health under award number R01GM132386. Computational resources used in this research were supported in part supported by the National Institute of General Medical Sciences of the National Institutes of Health under award number P30CA008748, as well as the Sloan Kettering Institute. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. We thank the Open Force Field Consortium for funding in the form of an Open Force Field Fellowship, including our industry partners as listed at the Open Force Field website (http://openforcefield.org). We also appreciate the Molecular Sciences Software Institute (MolSSI) for their support of the Open Force Field Initiative. We thank Simon Boothroyd for assistance with software implementation, including the OpenFF Evaluator Software; we also thank Simon for helpful conversations about the planning and scope of this project. We thank the members of the Open Force Field Initiative for their helpful feedback and discussion on the results of this research, along with the Open Force Field Scientific Advisory Board. We thank John Chodera of the Open Force Field Initiative and Sloan Kettering Institute for the use of their computational resources in this research.

## References

1 T. A. Halgren, Merck Molecular Force Field. I. Basis, Form, Scope, Parameterization, and Performance of MMFF94, *J. Comput. Chem.*, 1996, **17**(5–6), 490–519, DOI: 10.1002/(SICI)1096-987X(199604)17:5/6<490::AID-JCC1>3.0.CO;2-P.

2 K. Vanommeslaeghe, E. Hatcher, C. Acharya, S. Kundu, S. Zhong, J. Shim, E. Darian, O. Guvench, P. Lopes, I. Vorobyov and A. D. Mackerell, CHARMM General Force Field: A Force Field for Drug-like Molecules Compatible with the CHARMM All-Atom Additive Biological Force Fields, *J. Comput. Chem.*, 2010, **31**(4), 671–690, DOI: 10.1002/jcc.21367.

3 J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman and D. A. Case, Development and Testing of a General Amber Force Field, *J. Comput. Chem.*, 2004, **25**(9), 1157–1174, DOI: 10.1002/jcc.20035.

4 S. Riniker, Fixed-Charge Atomistic Force Fields for Molecular Dynamics Simulations in the Condensed Phase: An Overview, *J. Chem. Inf. Model.*, 2018, **58**(3), 565–578, DOI: 10.1021/acs.jcim.8b00042.

5 J. A. McCammon, B. R. Gelin and M. Karplus, Dynamics of Folded Proteins, *Nature*, 1977, **267**(5612), 585–590, DOI: 10.1038/267585a0.

6 G. Zhao, J. R. Perilla, E. L. Yufenyuy, X. Meng, B. Chen, J. Ning, J. Ahn, A. M. Gronenborn, K. Schulten, C. Aiken and P. Zhang, Mature HIV-1 Capsid Structure by Cryo-Electron Microscopy and All-Atom Molecular Dynamics, *Nature*, 2013, **497**(7451), 643–646, DOI: 10.1038/nature12162.

7 S. A. Hollingsworth and R. O. Dror, Molecular Dynamics Simulation for All, *Neuron*, 2018, **99**(6), 1129–1143, DOI: 10.1016/j.neuron.2018.08.011.

8 W. T. Hsu, D. A. Ramirez, T. Sammakia, Z. Tan and M. R. Shirts, Identifying Signatures of Proteolytic Stability and Monomeric Propensity in O-glycosylated Insulin Using Molecular Simulation, *J. Comput.-Aided Mol. Des.*, 2022, **36**(4), 313–328, DOI: 10.1007/s10822-022-00453-6.

9 K. Vanommeslaeghe and A. D. MacKerell, CHARMM Additive and Polarizable Force Fields for Biophysics and Computer-Aided Drug Design, *Biochim. Biophys. Acta, Gen. Subj.*, 2015, **1850**(5), 861–871, DOI: 10.1016/j.bbagen.2014.08.004.

10 W. Yu and A. D. MacKerell. Computer-Aided Drug Design Methods, in *Antibiotics: Methods and Protocols*, ed. P. Sass, Methods in Molecular Biology, Springer, New York, 2017, pp. 85–106, DOI: 10.1007/978-1-4939-6634-9_5.

11 G. L. Warren, C. W. Andrews, A. M. Capelli, B. Clarke, J. LaLonde, M. H. Lambert, M. Lindvall, N. Nevins, S. F. Semus, S. Senger, G. Tedesco, I. D. Wall, J. M. Woolven, C. E. Peishoff and M. S. Head, A Critical Assessment of Docking Programs and Scoring Functions, *J. Med. Chem.*, 2006, **49**(20), 5912–5931, DOI: 10.1021/jm050362n.

12 P. I. O'Daniel, Z. Peng, H. Pi, S. A. Testero, D. Ding, E. Spink, E. Leemans, M. A. Boudreau, T. Yamaguchi, V. A. Schroeder, W. R. Wolter, L. I. Llarrull, W. Song, E. Lastochkin, M. Kumarasiri, N. T. Antunes, M. Espahbodi, K. Lichtenwalter, M. A. Suckow, S. Vakulenko, *et al.*, Discovery of a New Class of Non-$\beta$-lactam Inhibitors of Penicillin-Binding Proteins with Gram-Positive Antibacterial Activity, *J. Am. Chem. Soc.*, 2014, **136**(9), 3664–3672, DOI: 10.1021/ja500053x.

13 C. J. Dickson, L. Rosso, R. M. Betz, R. C. Walker and I. R. Gould, GAFFlipid: A General Amber Force Field for the Accurate Molecular Dynamics Simulation of Phospholipid, *Soft Matter*, 2012, **8**(37), 9617–9627, DOI: 10.1039/C2SM26007G.

14 B. Kurt and H. Temel, Parameterization of Boronates Using VFFDT and Paramfit for Molecular Dynamics Simulation, *Molecules*, 2020, **25**(9), 2196, DOI: 10.3390/molecules25092196.

15 M. M. Ghahremanpour, J. Tirado-Rives and W. L. Jorgensen, Refinement of the Optimized Potentials for Liquid Simulations Force Field for Thermodynamics and Dynamics of Liquid Alkanes, *J. Phys. Chem. B*, 2022, **126**(31), 5896–5907, DOI: 10.1021/acs.jpcb.2c03686.

16 R. W. Pastor and A. D. MacKerell, Development of the CHARMM Force Field for Lipids, *J. Phys. Chem. Lett.*, 2011, **2**(13), 1526–1532, DOI: 10.1021/jz200167q.

17 L. P. Wang, K. A. McKiernan, J. Gomes, K. A. Beauchamp, T. Head-Gordon, J. E. Rice, W. C. Swope, T. J. Martínez and V. S. Pande, Building a More Predictive Protein Force Field: A Systematic and Reproducible Route to AMBER-FB15, *J. Phys. Chem. B*, 2017, **121**(16), 4023–4039, DOI: 10.1021/acs.jpcb.7b02320.

18 Y. Qiu, D. G. A. Smith, S. Boothroyd, H. Jang, D. F. Hahn, J. Wagner, C. C. Bannan, T. Gokey, V. T. Lim, C. D. Stern, A. Rizzi, B. Tjanaka, G. Tresadern, X. Lucas, M. R. Shirts, M. K. Gilson, J. D. Chodera, C. I. Bayly, D. L. Mobley and L. P. Wang, Development and Benchmarking of Open Force Field v1.0.0—The Parsley Small-Molecule Force Field, *J. Chem. Theory Comput.*, 2021, **17**(10), 6262–6280, DOI: 10.1021/acs.jctc.1c00571.

19 W. L. Jorgensen, J. D. Madura and C. J. Swenson, Optimized Intermolecular Potential Functions for Liquid Hydrocarbons, *J. Am. Chem. Soc.*, 1984, **106**(22), 6638–6646, DOI: 10.1021/ja00334a030.

20 B. A. C. Horta, P. T. Merz, P. F. J. Fuchs, J. Dolenc, S. Riniker and P. H. Hünenberger, A GROMOS-Compatible Force Field for Small Organic Molecules in the Condensed Phase: The 2016H66 Parameter Set, *J. Chem. Theory Comput.*, 2016, **12**(8), 3825–3850, DOI: 10.1021/acs.jctc.6b00187.

21 C. I. Bayly, P. Cieplak, W. Cornell and P. A. Kollman, A Well-Behaved Electrostatic Potential Based Method Using Charge Restraints for Deriving Atomic Charges: The RESP Model, *J. Phys. Chem*, 1993, **97**(40), 10269–10280, DOI: 10.1021/j100142a004.

22 M. Schauperl, P. S. Nerenberg, H. Jang, L. P. Wang, C. I. Bayly, D. L. Mobley and M. K. Gilson, Non-Bonded Force Field Model with Advanced Restrained Electrostatic Potential Charges (RESP2), *Commun. Chem.*, 2020, **3**(1), 1–11, DOI: 10.1038/s42004-020-0291-4.

23 A. Jakalian, B. L. Bush, D. B. Jack and C. I. Bayly, Fast, Efficient Generation of High-Quality Atomic Charges. AM1-BCC Model: I. Method, *J. Comput. Chem.*, 2000, **21**(2), 132–146, DOI: 10.1002/(SICI)1096-987X(20000130)21:2<132::AID-JCC5>3.0.CO;2-P.

24 A. Jakalian, D. B. Jack and C. I. Bayly, Fast, Efficient Generation of High-Quality Atomic Charges. AM1-BCC Model: II. Parameterization and Validation, *J. Comput. Chem.*, 2002, **23**(16), 1623–1641, DOI: 10.1002/jcc.10128.

25 W. L. Jorgensen, D. S. Maxwell and J. Tirado-Rives, Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids, *J. Am. Chem. Soc.*, 1996, **118**(45), 11225–11236, DOI: 10.1021/ja9621760.

26 M. Mohebifar, E. R. Johnson and C. N. Rowley, Evaluating Force-Field London Dispersion Coefficients Using the Exchange-Hole Dipole Moment Model, *J. Chem. Theory Comput.*, 2017, **13**(12), 6146–6157, DOI: 10.1021/acs.jctc.7b00522.

27 S. Boothroyd, O. C. Madin, D. L. Mobley, L. P. Wang, J. D. Chodera and M. R. Shirts, Improving Force Field Accuracy by Training against Condensed-Phase Mixture Properties, *J. Chem. Theory Comput.*, 2022, **18**(6), 3577–3592, DOI: 10.1021/acs.jctc.1c01268.

28 M. Schauperl, S. M. Kantonen, L. P. Wang and M. K. Gilson, Data-Driven Analysis of the Number of Lennard–Jones Types Needed in a Force Field, *Commun. Chem.*, 2020, **3**(1), 1–12, DOI: 10.1038/s42004-020-00395-w.

29 C. J. Fennell, K. L. Wymer and D. L. Mobley, A Fixed-Charge Model for Alcohol Polarization in the Condensed Phase, and Its Role in Small Molecule Hydration, *J. Phys. Chem. B*, 2014, **118**(24), 6438–6446, DOI: 10.1021/jp411529h.

30 X. Jia and P. Li, Solvation Free Energy Calculation Using a Fixed-Charge Model: Implicit and Explicit Treatments of the Polarization Effect, *J. Phys. Chem. B*, 2019, **123**(5), 1139–1148, DOI: 10.1021/acs.jpcb.8b10479.

31 S. Boothroyd, L. P. Wang, D. L. Mobley, J. D. Chodera and M. R. Shirts, Open Force Field Evaluator: An Automated, Efficient, and Scalable Framework for the Estimation of Physical Properties from Molecular Simulation, *J. Chem. Theory Comput.*, 2022, **18**(6), 3566–3576, DOI: 10.1021/acs.jctc.1c01111.

32 L. P. Wang, T. J. Martinez and V. S. Pande, Building Force Fields: An Automatic, Systematic, and Reproducible Approach, *J. Phys. Chem. Lett.*, 2014, **5**(11), 1885–1891, DOI: 10.1021/jz500737m.

33 R. H. Byrd, P. Lu, J. Nocedal and C. Zhu, A Limited Memory Algorithm for Bound Constrained Optimization, *SIAM Journal on Scientific Computing*, 1995, **16**(5), 1190–1208, DOI: 10.1137/0916069.

34 S. Boothroyd, P. K. Behara, O. C. Madin, D. Hahn, H. Jang, V. Gapsys, J. Wagner, J. Horton, D. Dotson, M. Thompson, J. Maat, T. Gokey, L. P. Wang, D. Cole, M. K. Gilson, J. D. Chodera, C. I. Bayly, M. R. Shirts and D. L. Mobley, Development and Benchmarking of Open Force Field 2.0.0

— The Sage Small Molecule Force Field, *J. Chem. Theory Comput.*, 2022, DOI: 10.1021/acs.jctc.3c00039.

35 R. Alizadeh, J. K. Allen and F. Mistree, Managing Computational Complexity Using Surrogate Models: A Critical Review, *Res. Eng. Des.*, 2020, **31**(3), 275–298, DOI: 10.1007/s00163-020-00336-7.

36 K. Deb, P. C. Roy and R. Hussein, Surrogate Modeling Approaches for Multiobjective Optimization: Methods, Taxonomy, and Results, *Math. Comput. Appl.*, 2021, **26**(1), 5, DOI: 10.3390/mca26010005.

37 M. A. Oliver and R. Webster, Kriging: A Method of Interpolation for Geographical Information Systems, *Int. J. Geogr. Inform. Syst.*, 1990, **4**(3), 313–332, DOI: 10.1080/02693799008941549.

38 I. Chivatá Cárdenas, On the Use of Bayesian Networks as a Meta-Modelling Approach to Analyse Uncertainties in Slope Stability Analysis, *Georisk*, 2019, **13**(1), 53–65, DOI: 10.1080/17499518.2018.1498524.

39 S. K. Dasari, A. Cheddad and P. Andersson, Random Forest Surrogate Models to Support Design Space Exploration in Aerospace Use-Case, in *Artificial Intelligence Applications and Innovations*, ed. J. MacIntyre, I. Maglogiannis, L. Iliadis and E. Pimenidis, IFIP Advances in Information and Communication Technology, Springer International Publishing, Cham, 2019, pp. 532–544, DOI: 10.1007/978-3-030-19823-7_45.

40 P. Jiang, Q. Zhou and X. Shao, *Surrogate Model-Based Engineering Design and Optimization*, Springer Tracts in Mechanical Engineering, Springer Singapore, Singapore, 2020, DOI: 10.1007/978-981-15-0731-1.

41 C. Nentwich and S. Engell, Application of Surrogate Models for the Optimization and Design of Chemical Processes, in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 1291–1296, DOI: 10.1109/IJCNN.2016.7727346.

42 K. McBride and K. Sundmacher, Overview of Surrogate Modeling in Chemical Process Engineering, *Chem. Ing. Tech.*, 2019, **91**(3), 228–239, DOI: 10.1002/cite.201800091.

43 B. D. Lackey, M. Pürrer, A. Taracchini and S. Marsat, Surrogate Model for an Aligned-Spin Effective-One-Body Waveform Model of Binary Neutron Star Inspirals Using Gaussian Process Regression, *Phys. Rev. D*, 2019, **100**(2), 024002, DOI: 10.1103/PhysRevD.100.024002.

44 G. Tapia, S. Khairallah, M. Matthews, W. E. King and A. Elwany, Gaussian Process-Based Surrogate Modeling Framework for Process Planning in Laser Powder-Bed Fusion Additive Manufacturing of 316L Stainless Steel, *Int. J. Adv. Des. Manuf. Technol.*, 2018, **94**(9–12), 3591–3603, DOI: 10.1007/s00170-017-1045-z.

45 J. Zhong, H. P. Wan, W. Yuan, M. He and W. X. Ren, Risk-Informed Sensitivity Analysis and Optimization of Seismic Mitigation Strategy Using Gaussian Process Surrogate Model, *Soil Dynam. Earthquake Eng.*, 2020, **138**, 106284, DOI: 10.1016/j.soildyn.2020.106284.

46 B. J. Befort, R. S. DeFever, G. M. Tow, A. W. Dowling and E. J. Maginn, Machine Learning Directed Optimization of Classical Molecular Modeling Force Fields, *J. Chem. Inf. Model.*, 2021, **61**(9), 4400–4414, DOI: 10.1021/acs.jcim.1c00448.

47 A. J. Booker, J. E. Dennis, P. D. Frank, D. B. Serafini, V. Torczon and M. W. Trosset, A Rigorous Framework for Optimization of Expensive Functions by Surrogates, *Structural Optimization*, 1999, **17**(1), 1–13, DOI: 10.1007/BF01197708.

48 R. Storn and K. Price, Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *J. Global Optim.*, 1997, **11**(4), 341–359, DOI: 10.1023/A:1008202821328.

49 P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, *et al.*, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, *Nat. Methods*, 2020, **17**(3), 261–272, DOI: 10.1038/s41592-019-0686-2.

50 O. Kramer, Genetic Algorithms, in *Genetic Algorithm Essentials*, ed. O. Kramer, Studies in Computational Intelligence, Springer International Publishing, Cham, 2017, pp. 11–19, DOI: 10.1007/978-3-319-52156-5_2.

51 M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson and E. Bakshy, BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization, arXiv:191006403 [cs, math, stat], 2020, http://arxiv.org/abs/1910.06403.

52 R. M. Neal, *Bayesian Learning for Neural Networks*, ed. P. Bickel, P. Diggle, S. Fienberg, K. Krickeberg, I. Olkin, N. Wermuth and S. Zeger, Vol. 118 of Lecture Notes in Statistics, Springer New York, New York, 1996, DOI: 10.1007/978-1-4612-0745-0.

53 O. Madin, J. Wagner, J. Setiadi, S. Boothroyd, M. Thompson, J. Rodríguez-Guerra and D. Dotson, Openforcefield/Openff-Evaluator: 0.3.4, 2021, DOI: 10.5281/zenodo.4630739.Zenodo.

54 S. Boothroyd, Common Workflows — OpenFF Evaluator Documentation, https://openff-evaluator.readthedocs.io/en/stable/properties/commonworkflows.html#simulation-layer.

55 L. Martínez, R. Andrade, E. G. Birgin and J. M. Martínez, PACKMOL: a package for building initial configurations for molecular dynamics simulations, *J. Comput. Chem.*, 2009, **30**(13), 2157–2164, DOI: 10.1002/jcc.21224.

56 B. Leimkuhler and C. Matthews, Rational Construction of Stochastic Numerical Methods for Molecular Sampling, *Applied Mathematics Research eXpress*, 2013, **2013**(1), 34–56, DOI: 10.1093/amrx/abs010.

57 D. Mobley, C. C. Bannan, A. Rizzi, C. I. Bayly, J. D. Chodera, V. T. Lim, N. M. Lim, K. A. Beauchamp, M. R. Shirts, M. K. Gilson and P. K. Eastman, Open Force Field Consortium: Escaping Atom Types Using Direct Chemical Perception with SMIRNOFF v0.1, *bioRxiv*, 2018, 286542, DOI: 10.1101/286542.

58 M. A. Bouhlel, J. T. Hwang, N. Bartoli, R. Lafage, J. Morlier and J. R. R. A. Martins, A Python Surrogate Modeling

Framework with Derivatives, *Adv. Eng. Software*, 2019, 102662, DOI: 10.1016/j.advengsoft.2019.03.005.

59 M. Frenkel, R. D. Chiroco, V. Diky, Q. Dong, K. N. Marsh, J. H. Dymond, W. A. Wakeham, S. E. Stein, E. Königsberger and A. R. H. Goodwin, XML-based IUPAC Standard for Experimental, Predicted, and Critically Evaluated Thermodynamic Property Data Storage and Capture (ThermoML) (IUPAC Recommendations 2006), *Pure Appl. Chem.*, 2006, **78**(3), 541–612, DOI: 10.1351/pac200678030541.

60 D. Riccardi, A. Bazyleva, E. Paulechka, V. Diky, J. W. Magee, A. F. Kazakov, S. A. Townsend and C. D. Muzny, *ThermoML/Data Archive*, National Institute of Standards and Technology, 2021, DOI: 10.18434/MDS2-2422.

61 J. Cihlář, V. Hynek, V. Svoboda and R. Holub, Heats of Vaporization of Alkyl Esters of Formic Acid, *Collect. Czech. Chem. Commun.*, 1976, **41**(1), 1–6, DOI: 10.1135/cccc19760001.

62 V. Majer, Z. Wagner, V. Svoboda and V. Čadek, Enthalpies of Vaporization and Cohesive Energies for a Group of Aliphatic Ethers, *J. Chem. Thermodyn.*, 1980, **12**(4), 387–391, DOI: 10.1016/0021-9614(80)90152-4.

63 V. Majer, V. Svoboda, S. Hála and J. Pick, Temperature Dependence of Heats of Vaporization of Saturated Hydrocarbons C5-C8; Experimental Data and an Estimation Method, *Collect. Czech. Chem. Commun.*, 1979, **44**(3), 637–651, DOI: 10.1135/cccc19790637.

64 A. Snelson and H. A. Skinner, Heats of Combustion: Sec-Propanol, 1,4-Dioxan, 1,3-Dioxan and Tetrahydropyran, *Trans. Faraday Soc.*, 1961, **57**, 2125–2131, DOI: 10.1039/TF9615702125.

65 V. Svoboda, V. Uchytilová, V. Majer and J. Pick, Heats of Vaporization of Alkyl Esters of Formic, Acetic and Propionic Acids, *Collect. Czech. Chem. Commun.*, 1980, **45**(12), 3233–3240, DOI: 10.1135/cccc19803233.

66 V. Majer, V. Svoboda, V. Uchytilová and M. Finke, Enthalpies of Vaporization of Aliphatic C5 and C6 Alcohols, *Fluid Phase Equilib.*, 1985, **20**, 111–118, DOI: 10.1016/0378-3812(85)90026-3.

67 V. Uchytilová, V. Majer, V. Svoboda and V. Hynek, Enthalpies of Vaporization and Cohesive Energies for Seven Aliphatic Ketones, *J. Chem. Thermodyn.*, 1983, **15**(9), 853–858, DOI: 10.1016/0021-9614(83)90091-5.

68 K. Byström and M. Månsson, Enthalpies of Formation of Some Cyclic 1,3- and 1,4-Di- and Poly-Ethers: Thermochemical Strain in the –O–C–O– and –O–C–C–O– Groups, *J. Chem. Soc., Perkin Trans. 2*, 1982, (5), 565–569, DOI: 10.1039/P29820000565.

69 G. Wolf, Thermochemische Untersuchungen an Cyclischen Ketonen, *Helv. Chim. Acta*, 1972, **55**(5), 1446–1459, DOI: 10.1002/hlca.19720550510.

70 I. Wadsö, M. L. Murto, G. Bergson, L. Ehrenberg, J. Brunvoll, E. Bunnenberg, C. Djerassi and R. Records, A Heat of Vaporization Calorimeter for Work at 25 Degrees C and for Small Amounts of Substances, *Acta Chem. Scand.*, 1966, **20**, 536–543, DOI: 10.3891/acta.chem.scand.20-0536.

71 J. Konicek, I. Wadsö, J. Munch-Petersen, R. Ohlson and A. Shimizu, Enthalpies of Vaporization of Organic Compounds. VII. Some Carboxylic Acids, *Acta Chem. Scand.*, 1970, **24**, 2612–2616, DOI: 10.3891/acta.chem.scand.24-2612.

72 S. V. Lipp, E. L. Krasnykh and S. P. Verevkin, Vapor Pressures and Enthalpies of Vaporization of a Series of the Symmetric Linear N-Alkyl Esters of Dicarboxylic Acids, *J. Chem. Eng. Data*, 2011, **56**(4), 800–810, DOI: 10.1021/je100231g.

73 P. I. Frazier, A Tutorial on Bayesian Optimization, arXiv:180702811 [cs, math, stat], 2018, http://arxiv.org/abs/1807.02811.

74 S. Ulaganathan, I. Couckuyt, F. Ferranti, E. Laermans and T. Dhaene, Performance Study of Multi-Fidelity Gradient Enhanced Kriging, *Structural and Multidisciplinary Optimization*, 2015, **51**(5), 1017–1033, DOI: 10.1007/s00158-014-1192-x.

75 J. Wu, M. Poloczek, A. G. Wilson and P. Frazier, Bayesian Optimization with Gradients, in *Advances in Neural Information Processing Systems*, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett, Curran Associates, Inc., 2017, vol. 30, https://proceedings.neurips.cc/paper/2017/file/64a08e5f1e6c39faeb90108c430eb120-Paper.pdf.

76 R. A. Messerly, S. M. Razavi and M. R. Shirts, Configuration-Sampling-Based Surrogate Models for Rapid Parameterization of Non-Bonded Interactions, *J. Chem. Theory Comput.*, 2018, **14**(6), 3144–3162, DOI: 10.1021/acs.jctc.8b00223.

77 H. Paliwal and M. R. Shirts, Multistate reweighting and configuration mapping together accelerate the efficiency of thermodynamic calculations as a function of molecular geometry by orders of magnitude, *J. Chem. Phys.*, 2013, **138**(15), 154108.