

## Tutorial: Determining Best Practices for Using Genetic Algorithms in Molecular Discovery

Brianna L. Greenstein,<sup>1</sup> Danielle C. Eley,<sup>1</sup> and Geoffrey R. Hutchison<sup>1, a)</sup>

*Department of Chemistry, University of Pittsburgh, 219 Parkman Avenue, Pittsburgh, Pennsylvania 15260, United States*

(Dated: 20 December 2023)

Genetic algorithms (GAs) are a powerful tool to search large chemical spaces for inverse molecular design. However, GAs have multiple hyperparameters that have not been thoroughly investigated for chemical space searches. In this tutorial, we examine the general effects of a number of hyperparameters, such as population size, elitism rate, selection method, mutation rate, and convergence criteria, on key GA performance metrics. We show that using a self-termination method with a minimum Spearman's rank correlation coefficient of 0.8 between generations maintained for 50 consecutive generations along with a population size of 32, 50% elitism rate, 3-way tournament selection, and a 40% mutation rate provides the best balance of finding the overall champion, maintaining good coverage of elite targets, and improving relative speedup for general use in molecular design GAs.

---

<sup>a)</sup>geoffh@pitt.edu; <https://hutchisonlab.org/>; Department of Chemical and Petroleum Engineering, University of Pittsburgh, Pittsburgh, Pennsylvania, 15260, United States

## I. INTRODUCTION

In recent years, genetic algorithms (GAs) have become an increasingly popular method for inverse molecular design<sup>1-4</sup>. With the near-infinite size of chemical space, computational tools provide the most efficient means to find novel molecular candidates with optimized properties tailored to specific applications. Multiple inverse design methods have been used to traverse chemical space, including generative machine learning<sup>5,6</sup> (ML), genetic algorithms<sup>3,7-11</sup> and heuristic global optimization<sup>12</sup>. If one wishes to find the best molecular candidates for a given property for which we have no prior knowledge or little data, however, ML methods will not be suitable since they require training. Instead, a GA can efficiently sample a wide range of chemical space and examine its potential without any training data.

GAs are based on Darwin's theory of natural selection, in which only the fittest individuals survive and reproduce for the next generation. Each successive generation contains increased fitness that increases their chance of survival, or in chemical terms, better molecular motifs that optimize a chemical property. In a traditional GA, as seen in Figure 1, an initial population of molecules is randomly selected from a subset of chemical space. Using a fitness function, which measures how well the molecule optimizes a target property, each molecule is examined and scored. The top performers, or "elites", are passed down to the next generation unchanged. To finish repopulating the next generation, parent molecules are selected from the current generation to reproduce by undergoing crossover and mutation operations to form a child. These children finish populating the next generation and the fitness of this new generation is evaluated. This cycle continues until the convergence conditions have been met. Throughout this work, a generation is defined as one pass through the cycle, with each generation having the ability for a slightly different population than the generation preceding it.

Although GAs have already been used for a variety of chemical applications<sup>3,8,10,13-25</sup>, to our knowledge their convergence strategies and multiple hyperparameters have not been thoroughly investigated for molecular discovery. For example, is it better to have a small population size and allow the GA to run for many generations or a large population size with fewer generations? How does the rate at which we introduce diversity with the mutation rate affect GA performance? In this work, we examine the effect of systematic convergence criteria, population size, elitism percentage, selection method, and mutation rate on key GA performance metrics, which we define to measure the quality of the top performer found, percentage of top candidates discovered and the

speedup over a brute-force approach. In order to thoroughly explore the effects of these parameters on chemical applications, we use three target molecular properties: polarizability, optical bandgap, and solvation energies, with our polymer-based GA.

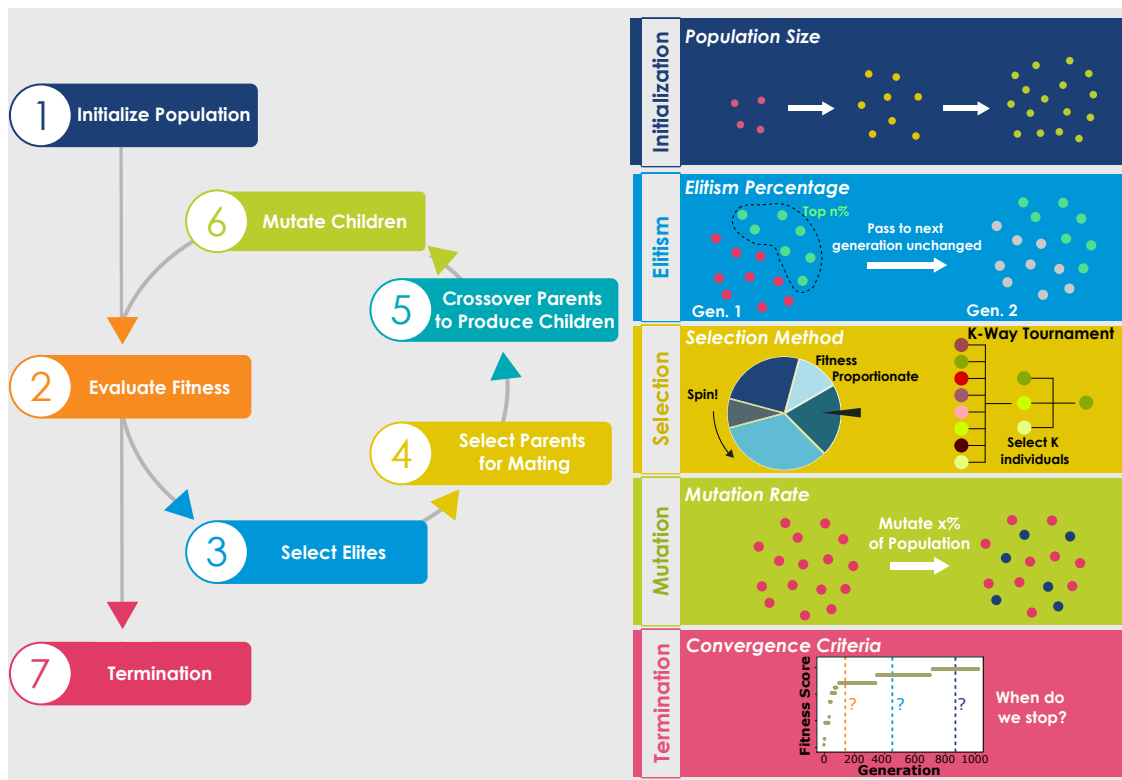


FIG. 1. Schematic of a genetic algorithm. The hyperparameters examined in this work, displayed on the right side, are the population size, elitism percentage, selection method, mutation rate, and convergence criteria.

## A. Initialization

The first step of a GA is the selection of individuals for the initial population. This initial pool of candidates contains a subset of genes that can encode for task-specific traits, such as monomer identity and sequence. Although there are chances for incorporation of traits not selected in this initialization step later on, the chance of finding the global best solution is increased if some of the top traits are already within this population.

There are two main strategies for selecting the initial pool of candidates<sup>19–21</sup>. The first is to seed the population with good solutions<sup>26</sup>. This method allows the global optimum to be found

much faster by optimizing known good solutions. To use this strategy, however, prior knowledge is necessary. Additionally, this may decrease the population diversity and not allow the exploration of non-intuitive candidates. The other and most commonly used option is random selection from the search space. With this method, no prior knowledge of the system is required and no restrictions are imposed. This helps start the GA with a diverse pool of candidates and allows a wide range of chemical space to be covered.

## **B. Fitness Function**

The individuals in each population need to be scored to assess how well they optimize a given property. The fitness function is a model that scores each individual and can range widely in complexity, from relatively simple tasks such as maximizing molecular weight that can be calculated simply with RDKit<sup>27</sup> to more time-intensive tasks such as calculating the dipole moment with quantum-chemical tools like DFT. In most GAs, the fitness function is the time-limiting step, so care must be taken to use cost-effective calculations. One way to do this is to use computationally inexpensive techniques, such as semi-empirical methods like sTD-DFT-xTB for calculating optical bandgap instead of the more costly but accurate TD-DFT. Another common method is to utilize ML as the fitness function. For properties that can be either computationally time-intensive or too difficult to calculate, a pre-trained ML model can drastically speed up fitness evaluation time. One example of using ML as the fitness function in a GA is reported in our previous work<sup>13</sup>, where we developed an ensemble random forest and neural network ML model to predict the power conversion efficiency (PCE) of materials for organic solar cells. This model was then used as the fitness function in a series of GAs to develop and find the best combination of new materials for tandem organic solar cells. Power conversion efficiency is arguably one of the most important metrics when developing solar cells, yet there is no simple computation to calculate it. It is highly dependent on multiple properties and ML is one of the most widely-used methods for computationally predicting it.

## **C. Selection Methods**

To repopulate the subsequent generations, the first step is to select two parents from the previous generation. A known problem of GAs is "premature convergence", or having a population

dominated by one type of solution<sup>7,28</sup>. This can prevent the GA from discovering the global best candidate by getting stuck in a local extremum. Thus, this operation is crucial for maintaining diversity and allowing for rapid convergence with highly fit individuals.

There are multiple types of selection methods, such as a k-way tournament, roulette wheel, stochastic universal sampling (SUS), rank, and random. In a k-way tournament, k random individuals are selected and their fitness scores are compared with each other<sup>20</sup>. The best-scoring individual is selected as a parent and this process is repeated to select the second parent. Another popular method is roulette wheel, where the chances of an individual being selected are proportionate to its score<sup>20</sup>. Just like a roulette wheel, the wedges on the wheel are the size of the individual score divided by the sum of all scores in that generation. A random number is generated, or the "wheel is spun", to pick the location on the wheel for the selection of the parent. This process is repeated for the other parent. Similarly, SUS is another wheel-based method, where instead of spinning the wheel twice, the wheel is spun once and two points on the wheel are chosen for the two parents<sup>29</sup>. Another similar method to roulette wheel is rank selection, where the sizes of the wedges are proportionate to the ranking of the population, not the individual score<sup>30</sup>. This can be beneficial in multiple circumstances, such as negative fitness scores or when the fitness scores are close together (leading to almost equal wedge sizes and no selection pressure).

These aforementioned methods are all classified as fitness proportionate methods, meaning that higher-scored individuals have a higher chance of being selected, and the entire population is accessible for selection. A different and much simpler method is "random" selection, where 2 parents are randomly selected either from the entire population or from some top percentage of individuals<sup>3,8,13,15</sup>. By selecting from only the top candidates, this selection can allow for faster convergence.

While there has not been a thorough comparison between these methods, the most popular ones for chemical applications are roulette wheel<sup>16,31,32</sup> and tournament selection<sup>33,34</sup>. An analysis comparing GA parameters for heterogeneous catalysts found that, out of wheel, rank, threshold, and tournament selection methods, 3-way tournament selection was the best<sup>28</sup>. Another technique implemented by De Sousa et al. randomly selects parents from a pool of the previous generation, plus the top 10 individuals obtained thus far<sup>15</sup>. Previous GA implementations in our group used a random selection method by randomly selecting two parents from the top 50% of candidates<sup>3,8</sup>.

## D. Elitism

A strategy to speed up GAs while balancing exploration and exploitation is the use of elitism. This approach keeps a certain percentage of the top candidates to pass down to the next generation unchanged, ensuring there are always good traits in each population<sup>21</sup>. Without using elitism, the GA will converge much slower and is less likely to reach champion performers. Elitism is almost always used in GAs, although the percentage of candidates denoted as "elites" can vary.

## E. Chemical Representation, Crossover, and Mutation

In biological systems, an individual's physical traits are encoded in its DNA, which is unique to each individual, and contain information from both of its parents. During reproduction, parts of the DNA from both parents crossover to form a new sequence for the child. During this process, there is a chance a gene may undergo mutation and lead to a different trait. Similarly in GAs, each individual solution has unique sets of genes that contain information from parents in the previous generation. Each individual's genes need to be represented computationally in a way that allows for efficient crossover and mutation. The simplest representation is binary strings, where each position codes for a trait. In chemical applications, it is difficult to maintain all relevant molecular information with this strategy. The most common representations for chemical GAs are molecular-based strings such as SMILES<sup>3,8,35</sup> and SELFIES<sup>2,36</sup>, or molecular graphs where the nodes are atoms and the edges are bonds<sup>10,31,37,38</sup>.

SMILES-based representations commonly use a fragment-based approach, using a library of SMILES fragments to mix and match and design new molecules. One drawback to this method is that it places more limits on molecular design and therefore on the search space. Alternatively, performing random mutations to a SELFIES string will always maintain chemical validity and allow for near-infinite possibilities. This representation has already been incorporated into successful GAs, allowing for the insertion, deletion, and replacement of characters in the string. A more recent technique is the highly successful graph-based GA developed by Jensen<sup>10</sup> that has been shown to quickly explore chemical space with the molecules having little resemblance to the initial population. Since each individual is a graph instead of a string, crossover of parents occurs by cutting each parent into fragments and swapping the fragments to make the new molecule, while ensuring the newly formed children are chemically valid. Graphs can have mutations similar to SELFIES,

such as insertions, deletions, and replacing atoms. Additionally, there are opportunities to mutate the bonds, such as changing the order of the bonds or adding a ring bond<sup>10,37</sup>. In this work, we use fragment-based GAs with SMILES representations because of their simplicity and easily defined search space.

The crossover and mutation operators are essential for traversing chemical space. One of the main challenges with genetic algorithms is the balance of exploration and exploitation. In each generation, there is a population of candidates, some of which perform much better than others in an optimization task. Considering that the goal of the GA is to find high-performing solutions, we want to optimize the top candidates in each generation by mixing and matching their genes. This crossover step is crucial for exploiting the nearby chemical space. However, if we were to rely on just the crossover operation, the GA can easily get trapped at a local maximum and fail at finding the global best solution. To resolve this, the mutation operation can be performed on the newly formed child, during which one of the child's genes receives a random change. Mutation ensures diversity and introduces new genes into each population. Since we still want to balance exploitation and exploration, each child has a chance of undergoing a mutation. The probability of a mutation occurring is set by the mutation rate, one of the hyperparameters of the GA. Although a wide variety of mutation rates have been reported in the literature<sup>10,34,39,40</sup>, ranging from 1% to 50%, there have been few reports on optimizing this parameter for chemical applications<sup>41</sup>. In this work, we examine mutation rates ranging from 10% to 90% with 10% increments.

## F. Convergence

Across the wide variety of chemical problems probed by GA searches in recent years, the most common method of ending a run is by simply having the GA terminate after completing a predetermined number of generations.<sup>8,18,42-46</sup> The strategy most often employed is to run the GA for a set number of generations and stop the GA at that point, declaring it "converged" as demonstrated in the schematic in Figure 2. The number of generations is set at a level thought to be substantially greater than the number needed to reach a plateau where the top performer ceases to change for many generations. This approach is common because it generally works and is easy to implement, but it has several drawbacks.

First, this approach requires an initial guess of the approximate number of generations necessary to reach convergence, resulting in initial runs that may either have to be run for hundreds of

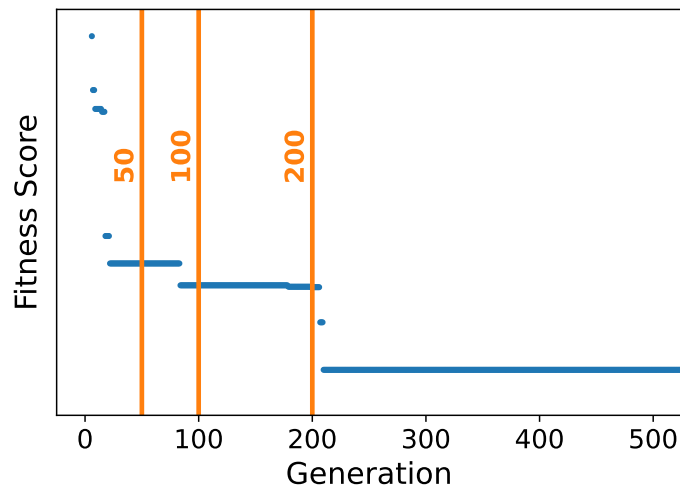


FIG. 2. This schematic, based on an actual chemical GA run, demonstrates the difficulty with terminating a GA after a set number of generations. As shown, selecting a reasonable round number such as 50, 100, or 200 generations as the predetermined endpoint would miss the true champion fitness score. Given the local plateaus around these numbers, however, a researcher may be given a false sense of true convergence.

generations more than necessary or be actively monitored by the researcher to determine when to manually stop the GA. Even once a good estimate of the number of generations necessary to reach convergence has been established, the GA must be set to complete a substantial number of additional generations to ensure that the random variability of each run is taken into account and the GA does not terminate prematurely; this can result in wasted resources as a GA runs for far longer than necessary. Finally, this predetermined endpoint approach is not easily transferable between GA methods for different applications. Whenever the optimization target changes (for example, choosing to optimize a different chemical property or set of properties), or even when the hyperparameters of a GA are changed, the number of generations needed for convergence can be impacted. This means that new initial guesses are needed and it is difficult to compare the endpoints of GAs without identical parameters due to the lack of a standard convergence protocol. Due to these inherent drawbacks of this termination approach, in this work, we examine systematic convergence criteria which can be applied broadly to chemical GAs.

Kwon et al. recently implemented a genetic algorithm with a generalized convergence criterion as part of a broader evolutionary design method.<sup>33</sup> In their algorithm, they defined two convergence conditions: a minimum of 500 total generations and a period of 30 consecutive generations during which the fitness was not enhanced. They terminated the evolutionary cycle only after both

conditions were met. Previous work in our group also explored termination in a molecular GA focused on building new chemical structures from a pool of known monomers.<sup>47</sup> This method calculated Spearman's rank correlation coefficient of the top given percentage of monomers used at intervals over 100 generation GA runs with variably-sized monomer pools. The point at which the average of this coefficient met or exceeded a value of 0.5 was considered convergence and was used to estimate the number of generations needed to reach convergence for a GA with a similar monomer pool size, and therefore search space.

In this work, we go a step further and completely divorce the concept of convergence from completing a set number of generations. Instead, we define two convergence criteria: the Spearman coefficient and the number of consecutive convergence generations. Spearman's rank correlation coefficient (here referred to as "Spearman coefficient" for brevity) is calculated at each generation to measure the correlation between the current and previous generation's ranks of top monomers ordered by frequency, as measured by total usage of each monomer in all oligomers generated over the course of the GA run. This coefficient measures covariance as the nonparametric correlation between the ranks of two variables, in this case, top monomers ranked by frequency. Because it is defined as the Pearson correlation coefficient between the rank variables, it is a normalized measure with values ranging from -1 to 1, where the most negative value indicates perfect negative correlation, the most positive value indicates perfect positive correlation, and zero indicates no correlation.<sup>48</sup> In principle, the Spearman coefficient should asymptotically approach 1.0 with increasing GA generations, indicating that the top candidates have been definitively found and therefore there is no change in subsequent generations. Due to the general lack of convergence strategies among molecular GAs, we are unaware of other similar projects that currently use the Spearman coefficient as a measure of convergence. We believe it is a mathematically appropriate measure of correlation for this application, however, and as noted above previous work in our group successfully used this coefficient to measure GA convergence.<sup>47</sup>

To determine convergence, a minimum Spearman coefficient and a minimum number of consecutive convergence generations are set at the outset of the GA run. When the Spearman coefficient between two generations meets or exceeds the minimum convergence Spearman coefficient, meaning the two generations are meaningfully similar, that generation is considered a convergence generation, and the counter for consecutive convergence generations is advanced by one. This counter continues advancing as long as the following generations continue to meet or exceed the minimum convergence Spearman coefficient; if this does not happen, the counter resets to zero.

Once the counter reaches the minimum number of consecutive convergence generations required, termination is triggered and the GA cycle halts. Using these convergence criteria allows GA runs to self-terminate based on their own individual trajectories in a comparable, systematic way and eliminates the artificial nature of predetermined cut-off generations.

## **G. GA Performance Metrics**

To evaluate the optimal values for each GA hyperparameter, three metrics are examined: champion, coverage, and speedup. The champion is the rank of the best-performing candidate found in the entire search space. A champion of 1 means the global best polymer was discovered. This metric is important since it will give insight into the global optimization performance of the GA. The coverage is the percentage of the top 100 polymers discovered throughout each GA run. This metric helps evaluate how well the GA can explore and jump out of local optima to find multiple good candidates. In some optimization tasks, it may be more important to find a myriad of top performers as opposed to only the global optimum. The speedup is the size of the search space divided by the number of unique individuals examined by the GA until convergence. This metric shows how much quicker using the GA is compared to a brute-force approach.

## **II. COMPUTATIONAL DETAILS**

### **A. Search Space Calculations**

A list of 447 monomer SMILES previously used in our group was used as building blocks to create oligomers of 6 monomers (hexamers). The complete list is available at <https://github.com/hutchisonlab/GA-Best-Practices>. These monomers are all small conjugated organic molecules. The constraints for designing the hexamers were a maximum of 2 monomer types, with the alternating ABABAB sequence. Since end groups were not included, in most cases the sequences ABABAB and BABABA are chemically identical. Additionally, homopolymers were allowed, meaning A can be the same as B. This led to a total search space of 100,128 oligomers.

The three chemical properties examined in this paper are the polarizability, optical bandgap, and solvation energy ratio between water and hexane. Since the goal of this work is to understand the effects of the GA hyperparameters on the search through the chemical space, the chemical space needs to be fully mapped. With this method, the global optimum will be known and one

way the hyperparameters can be evaluated is based on if they found the champion hexamer. To do this, the three chemical properties were exhaustively calculated for all 100,128 oligomers. This database of properties can be accessed within the GA during the fitness evaluation step.

To calculate these properties, the hexamers (built from the monomer SMILES) first underwent force field geometry optimization with MMFF94<sup>49</sup> using OpenBabel<sup>50</sup>, and then further geometry optimization with GFN2-xTB<sup>51-53</sup> (xTB version 6.4.1). The polarizabilities were extracted from the GFN2-xTB calculation. To calculate the optical bandgap, simplified time-dependent-DFT-xTB (sTD-DFT-xTB) was performed<sup>54</sup> up to 5 eV. This was performed with the sTDA-xTB package, version 1.0 for xTB for sTDA<sup>54</sup> and sTDA<sup>55</sup> version 1.6.2. The solvation energies in water and hexane were calculated using GFN2-xTB with the ALPB solvation models. The solvation energy ratio was calculated as: Solvation energy ratio =  $(G_{solv}^{H_2O} - G_{solv}^{C_6H_{14}}) / |G_{solv}^{H_2O}|$

Although this paper is a tutorial on the GA methodology and not the quantum chemical methods, we explored if the molecular conformations had an effect on the polarizability, optical bandgap, and solvation energies in water and hexane. The Conformer-Rotamer Ensemble Sampling Tool (CREST)<sup>56</sup> was used to sample accessible conformers of the champion hexamer discovered for all three chemical properties. The number of stable conformers for the hexamer champions for polarizability, optical bandgap, and solvation energy ratio were 1, 6, and 169 conformers, respectively. The polarizability, optical bandgap, and solvation energies in water and hexane were calculated using GFN2-xTB, sTD-DFT-xTB, and ALPB implicit solvation model in xTB, respectively, for all stable conformations of the 3 hexamers. Analysis of the distribution of polarizabilities found that the conformation has a negligible effect on the polarizability, with the average range in polarizability among conformers to be  $0.18 \text{ \AA}^3$ . Evaluating the optical bandgap conformed bandgap was a little trickier, due to limitations in calculating very small with sTD-DFT-xTB or lack of excitations under 5 eV. Only one hexamer yielded all valid optical bandgap calculations, and this gave a range of 0.085 eV among conformers, again showing the conformation has a minimal effect (ESI Figure S1). The average range in solvation energy in water and hexane was  $0.0049 E_h$  and  $0.0014 E_h$ , respectively. Thus, exhaustive conformer sampling has minimal effect on these chemical properties and is not necessary for the purposes of this GA.

To visualize the search space, t-SNE and UMAP were used. t-SNE was performed with scikit-learn. Principal component analysis (PCA) was first performed to reduce the data to 50 dimensions. This was then fit for t-SNE to reduce it further to 2 dimensions. UMAP<sup>57</sup> was used as well for comparison. The data were reduced to two dimensions, with the number of neighbors as

25 (size of the neighborhood to look at), the minimum distance for clustering as 0.001, and the distance between points is calculated with the Jaccard metric.

## **B. Convergence Criteria**

Cumulative frequency counts are kept of the number of times each monomer in the monomer dataset is found in the GA's population. These frequency counts are updated after each generation and used to rank monomer indexes from most to least frequently used. The Spearman coefficient is calculated at each generation to measure the rank correlation between the ranked monomer index order of the current generation and the previous generation. To determine convergence, a minimum Spearman coefficient and a minimum number of consecutive convergence generations are set at the outset of the GA run. When the Spearman coefficient between two generations meets or exceeds the minimum convergence Spearman coefficient, that generation is considered a convergence generation, and the counter for consecutive convergence generations is advanced by one. This counter continues advancing as long as the following generations continue to meet or exceed the minimum convergence Spearman coefficient; if this does not happen, the counter resets to zero. Once the counter reaches the minimum number of consecutive convergence generations required, termination is triggered and the GA cycle halts.

## **C. GA Hyperparameter Tuning**

The GA code used in this work was developed with Python and is freely available on Github (<https://github.com/hutchisonlab/GA-Best-Practices>). The default GA hyperparameters were a population size of 32, an elitism percentage of 50%, random selection, and a mutation rate of 40%. The population sizes examined are 16, 20, 24, 28, 32, 48, 64, 80, and 96 candidates. The elitism rates examined are 0, 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, and 80%. The selection methods examined are random, random from the top 50%, tournament selections (2-, 3-, and 4-way), roulette wheel, stochastic universal sampling, and rank. The mutation rates examined are 1%, 5%, and 10-90% in increments of 10%.

A detailed schematic of the GA for hexamers in this work is shown in Figure 3. In this schematic, maximizing the polarizability is used as an example fitness function. Starting from a database of manually-generated monomer SMILES, the GA will randomly select monomers to

form the initial population of alternating-sequence hexamers. The number of hexamers in the population is set by the population size. To evaluate the polarizability (i.e. fitness function), each hexamer undergoes geometry optimization and property calculations with GFN2-xTB/D4. The rest of the steps for this generation involve repopulating the next generation's population, either with good solutions from the previous population or with slight modifications through crossover and mutation operations. After the polarizability is calculated for each hexamer in the population, a percentage of them, set by the elitism percentage, is selected to pass to the next generation unchanged. The remainder of the population is replenished with crossover and mutation. For example, if the population size is 32 and the elitism percentage is 50%, then the top 16 hexamers will pass to the next generation unchanged. The process of selecting parents and undergoing crossover and mutation to produce a new hexamer will be repeated 16 times until the new population contains 32 hexamers. In this schematic, 3-way tournament selection is used to select two parents. The parents undergo crossover, by taking one monomer from each parent, to form a new "child" hexamer. This child has a chance, set by the mutation rate, to undergo mutation, which in this example is replacing one monomer with a new monomer from the database. This new population will undergo polarizability calculations and this cycle will continue until some pre-defined convergence conditions have been met.

#### **D. Larger search space**

In the second part of this work, we test the optimized hyperparameters on a larger, not pre-mapped search space. The 447 monomer SMILES list used in the first part of this work was expanded to 1200 monomer SMILES, including additional common repeat units in conjugated materials, as well as common aryl-vinyl and aryl-azo combinations. These monomers are used to create hexamers (oligomers containing 6 monomer units). Contrary to the first part of this work, where only the alternating ABABAB sequence was allowed, all possible 2 monomer sequences are permitted. This yields  $2^6 = 64$  possible sequences. Thus, the total search space for this part of the work is 46,041,600 oligomers.

The chemical properties examined were polarizability, optical bandgap, and solvation energy ratio. These were calculated in the same manner as in the Search Space Calculations section, the only difference being that these calculations were done on the fly rather than being precomputed.

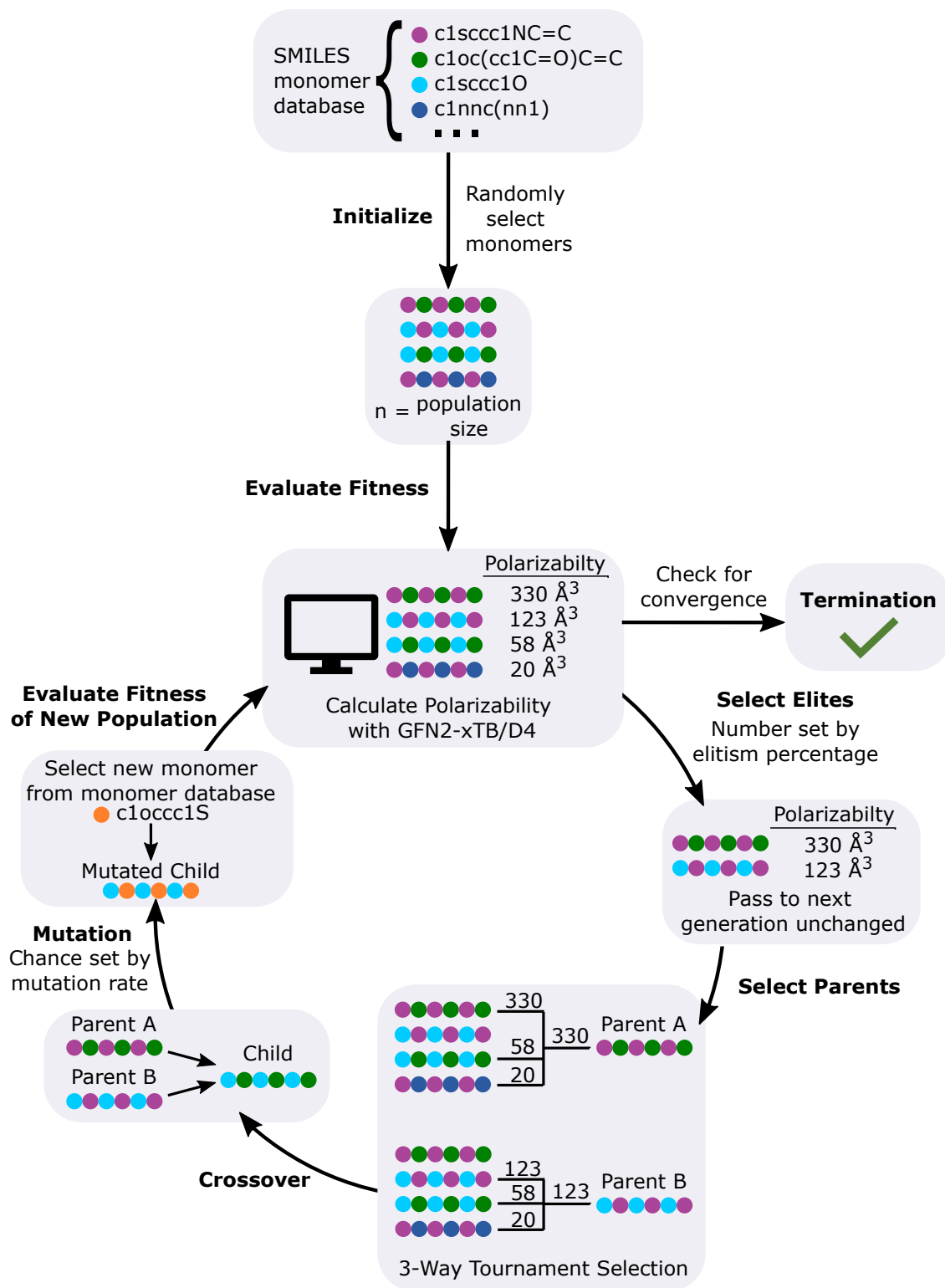


FIG. 3. Schematic of an example GA used in the first part of this work. This GA designs new ABABAB hexamers to maximize polarizability.

### III. RESULTS AND DISCUSSION

#### A. Search Space

Genetic algorithms can be used for a wide range of chemical properties. To ensure the hyperparameters suggested in this work can apply to a variety of chemical applications, three different chemical properties are examined. Optimization tasks include maximizing isotropic static polarizability, minimizing the optical band gap, and minimizing the solvation energy ratio between water and hexane. In addition to different optimization goals, the diversity of the search space is important. Highly clustered search spaces, where the top candidates are localized, have much higher success rates of a GA finding the global best individual and high coverage of the top contenders, compared to more diverse chemical spaces.

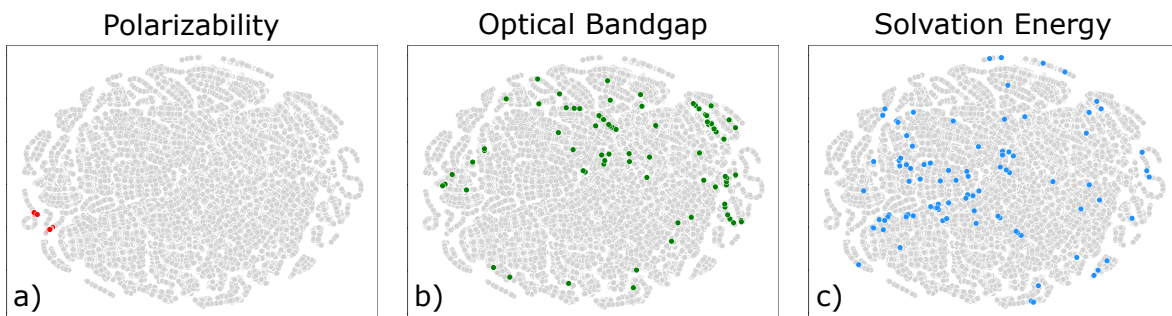


FIG. 4. Visualization of chemical space using t-distributed stochastic neighbor embedding (t-SNE). The input data included 2048-bit ECFP4 fingerprints for each polymer, and monomer descriptors such as HOMO, LUMO, optical bandgap, solvation energy in water, and solvation energy in hexane. The entire search space of 100,128 polymers is represented as grey points, and the colored points represent the top 100 polymers with (a) the highest polarizability, (b) the lowest optical bandgaps, and (c) the lowest solvation energy ratio between water and hexane.

In this work, 447 unique monomers are used to create hexamers with an alternating or homopolymer sequence. This yields a total search space of 100,128 possible polymers. The search space was exhaustively examined to understand how well each GA is traversing the search space. Figure 4 shows a visualization of chemical space computed with t-Distributed Stochastic Neighbor Embedding (t-SNE). This is a dimensionality reduction technique<sup>58</sup> that can visualize high-dimensional data such as molecules. To visualize chemical space, 2048-bit Extended Connectivity Fingerprints (ECFP4) of each polymer were used as input data to reduce to a set of 2-dimensional

coordinates (ESI Figure S2). Additional data for each monomer were also added such as the HOMO, LUMO, optical bandgap, solvation energy in water, and solvation energy in hexane. Figure 4 shows the chemical diversity among the top candidates for each chemical property. The top candidates to maximize polarizability (Figure 4a) are highly clustered, whereas the top candidates for optical bandgap and solvation energy are more spread out. Another dimensionality reduction technique, Uniform Manifold Approximation and Projection<sup>57</sup> (UMAP), was performed but yielded inconclusive results (ESI Figure S3).

To further examine the chemical diversity among the top performers, an analysis using Tanimoto similarity coefficients was performed. The Tanimoto coefficients among the unique monomers that make up the top 100 polymers are calculated iteratively and represented as a heatmap in ESI Figure S4. The heatmap for polarizability (ESI Fig S4a) has more light spots than the other chemical properties, indicating higher chemical similarity among monomers. The distribution of the average Tanimoto coefficients of each monomer with all other monomers is shown in ESI Fig S4d-f. The average Tanimoto coefficients for polarizability range from 0.15-0.23, while for optical bandgap and solvation energy range from 0.06-0.2 and 0.05-0.2, respectively. These coefficients are relatively low, suggesting a high level of chemical diversity among the monomers.

Analysis of variance (ANOVA) was performed in these sets and shows a significant difference overall between the 3 properties ( $p=2.2 \times 10^{-13}$ ). However, this does not tell us what those differences are. Using Tukey's honestly significantly differenced test, which compares each group with each other, we see that polarizability has a significantly different set of Tanimoto coefficients than optical bandgap ( $p \leq 0.001$ ) or solvation energy ratio ( $p \leq 0.001$ ). This indicates that among the three properties, polarizability has a more localized search space of top performers, while the other target properties are more diverse. Examining the GA performance across all three parameters will give generalized hyperparameters that can work on future chemical properties, regardless of chemical space diversity.

## B. Convergence Criteria

In order to efficiently and systematically test the effects of the other GA hyperparameters with a model GA that was capable of self-termination, we first determined the best convergence criteria. Two interlinked criteria are tested: the Spearman coefficient and the number of consecutive convergence generations. All combinations of Spearman coefficients of 0.6, 0.7, 0.8, and consecutive

convergence generations of 0.9, and 5, 10, 25, 50, and 100 are examined.

Figure 5A shows the distributions of the median performance metrics for all 15 runs using each combination of convergence criteria. Looking at the champion metric (ESI Table S1), the median is 1 in most runs using a minimum consecutive convergence generation of 25 or higher, meaning that the global extremum was found in most runs with that criterion. Also of note, the global extremum was usually found in all runs using both a minimum Spearman coefficient of 0.7 or higher and a minimum number of consecutive convergence generations of 50 or higher, as shown by the lack of distribution around the median for those convergence criteria. Looking at these performance metrics separately for each property, we see a similar result (ESI Figure S5). For polarizability, the global champion is discovered for every Spearman coefficient tested after 50 generations, whereas for the optical bandgap and solvation energy ratio, a Spearman coefficient of 0.7 or higher is needed for 50 generations.

Given the broad number of convergence criteria in which the global champion was found consistently, we focused on the coverage and speedup metrics (ESI Tables S2 and S3, respectively) when selecting our generalized convergence method, as shown in Figure 5B (error bars are shown on an alternative version of this plot in ESI Figure S6). Fitting the data to a power curve, we see there is a clear trade-off between these two metrics for our convergence methods. Using the fit to extrapolate out, we recognize that even at 100% coverage, the GA still provides a substantial speedup of approximately 19.6 over a comprehensive search, verifying the benefit of GA-led searches with even relatively resource-inefficient GAs. Future work should examine convergence methods and techniques beyond those suggested here to test whether the apparent Pareto front can be broken.

For the purposes of this work, however, we focus on finding the convergence method that gets closest to a median coverage of 50% while maximizing speedup. We thus chose the method that uses a minimum Spearman coefficient of 0.8 with 50 consecutive convergence generations, which has a median coverage of 42%. Although it performed very similarly to the convergence method using a minimum Spearman coefficient of 0.7 with 50 consecutive convergence generations, we selected the former method as its coverage distribution skewed slightly higher than the latter method.

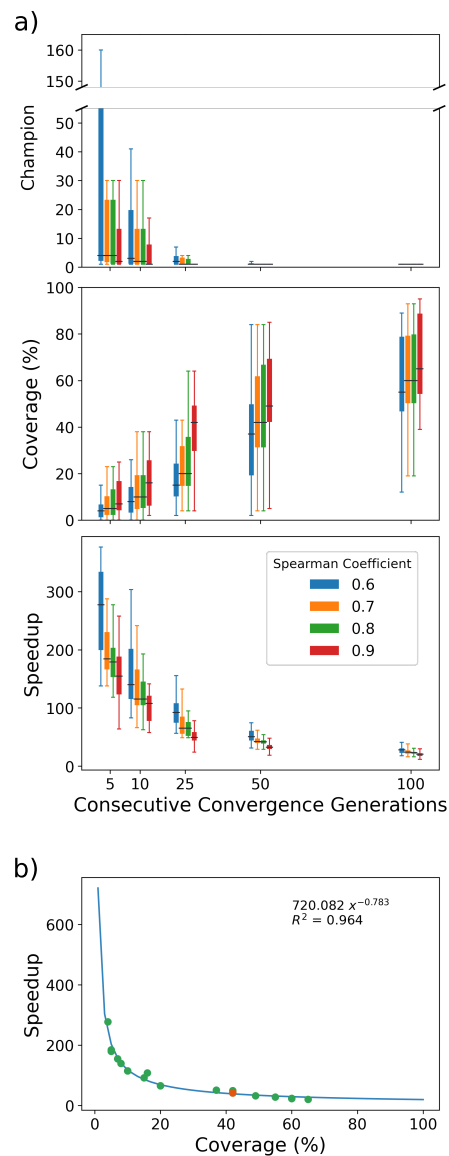


FIG. 5. Median performance metrics from among 15 runs (5 per chemical property) are shown for comparing different combinations of convergence criteria: minimum Spearman correlation coefficient and the number of consecutive convergence generations. a) The champion is the rank of the best polymer found. The coverage is the percentage of the top 100 polymers discovered by the GA. The speedup is the size of the search space / total number of calculations performed. (b) Comparison of coverage vs speedup to determine which set of criteria gives the best balance. The set that was chosen for the purposes of this work is a minimum Spearman coefficient of 0.8 and 50 convergence generations, highlighted in orange.

## C. GA Hyperparameters

To understand how different GA hyperparameters tune the GA's emphasis on exploration versus exploitation, the population size, elitism percentage, selection method, and mutation rate are examined. While testing each parameter, all other parameters remained constant. The default parameters were a population size of 32, an elitism percentage of 50%, random selection, and a mutation rate of 40%. The previously determined convergence method was used with a minimum Spearman coefficient of 0.8 and 50 convergence generations. To measure the performance of the GAs with tuned hyperparameters, the champion, coverage, and speedup performance metrics are calculated in the same manner as described in the convergence criteria section.

### 1. Population Size

Population sizes of 16, 20, 24, 28, 32, 48, 72, and 96 individuals are examined. Figure 6a shows that for all population sizes, the median champion was 1, meaning in a majority of runs it found the global optimum. However, the standard deviation is much larger for a population size of 16. When examining the performance among the chemical properties (ESI Table S4), all population sizes performed well for maximizing polarizability. For optical bandgap and solvation ratio, although some runs were able to find the global best polymer, many runs did not. The champions among the runs that did not find the global optimum had a better rank as the population size increased.

As expected, coverage increases as the population size increases. Having a larger population size allows for more traits in the generation to select from during crossover and allows for efficient exploration. Examining the coverage among each run (ESI Table S5) indicates that the GA has difficulty finding high performers when optimizing the optical bandgap, most likely due to the more diverse search space. Increasing the population size to 72 or 96 allowed the GA to find 99-100% of the top 100 candidates for polarizability.

As the population size increases, more calculations are performed, which decreases speed-up (Figure 6 a). The speedup was similar among the 3 chemical properties (ESI Table S6). Figure 6b shows the balance between coverage and speedup. There is a negative correlation, indicating that the best population size depends on the optimization task. If the goal is to find many high-performing candidates and the computation cost is not important, then a population size of 96 is

recommended. If computational cost is essential, then a population size of 16 is suggested. The best balance of coverage and speedup is a population size of 32, since it has similar speedups to 24 and 28 individual populations but with higher coverage. Examination of this coverage-speedup trade-off for each individual chemical property (ESI Figure S7) is consistent with this result, suggesting a population size of 32 gives the best balance across multiple chemical space topographies.

## 2. *Elitism*

Elitism percentages of 0, 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, and 80% are examined. The percentage of elitism is the percentage of the population that is passed on unchanged to the next generation. Figure 7a shows the performance of each elitism rate. The median champion was 1 for all elitism rates except for 0%. Having no elitism severely limits the exploitation of good candidates already found, making it difficult to pass on good traits to find similarly good candidates. There are very small error bars for 20%, 25%, and 30%, indicating it consistently finds the champion. Investigating the performance for each individual run (ESI Table S7 and Figure S8) reveals that, aside from 0% elitism, all other rates found the global optimum in every run for polarizability. Although optical bandgap found the global optimum in a majority of runs across all elitism rates, the runs that did not find it had very poor champions, with some as high as a champion rank of 74. Solvation energy had one run that never found the global champion across all elitism rates, an inherent issue of the stochastic nature of the GA.

Examining the coverage shows that having even a small amount of elitism can dramatically increase the coverage, from 21% coverage with 0% elitism to 46% coverage with 5% elitism. Although there is some fluctuation, the coverage remains relatively constant among 5-50% elitism and starts to decrease after 60% elitism. Looking at individual runs (ESI Table S8) reveals that the best coverage of 97% was found at 20% elitism to optimize polarizability. There is much higher coverage for polarizability compared to optical bandgap and solvation energy due to the more clustered polarizability search space.

Figure 7a also shows that as the elitism rate increases, so does the speedup (ESI Table S9). This is because as you increase the elitism rate, more of the population remains unchanged and there are fewer new individuals to evaluate per generation. Since the convergence criteria are set to have a Spearman's correlation coefficient above 0.8 between each generation for 50 generations, it is

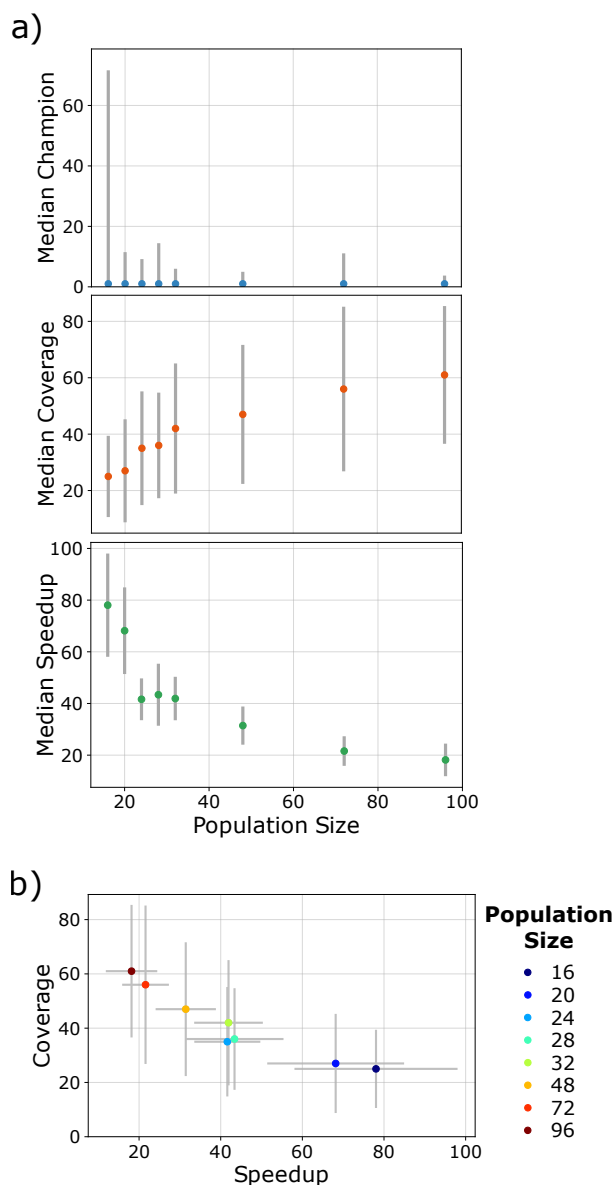


FIG. 6. Performance metrics for evaluating population sizes of 16, 20, 24, 28, 32, 48, 72, and 96 individuals. (a) The champion is defined as the rank of the best polymer found. The coverage is the percentage of the top 100 polymers discovered by the GA. The speedup is the size of the search space / total number of calculations performed. The median from among 15 runs (5 per chemical property) is displayed, with the standard deviation as error bars. (b) Comparison of coverage vs speedup to determine which population size gives the best balance, with the standard deviation as error bars.

much easier for the GA to converge before efficiently exploring chemical space. A GA with no elitism is only slightly better than a brute force approach, with a median speedup of only 4. Since there is no guarantee of good traits in the generation, it is very difficult to converge on a good

population.

Unlike population size, there is no direct relationship between coverage and speedup (Figure 7b). An expected 0% elitism gives the worst performance with poor coverage and poor speed-up. Elitism of 5-50% shows similar good coverage, however, 50% shows the largest speedup. For this reason, 50% elitism is recommended for future GAs.

### 3. *Selection Methods*

Various types of selection methods were examined, such as random, tournament style, and fitness proportionate methods. The fitness proportionate methods that are dependent on the actual fitness score are unable to perform optimization tasks that can have a negative fitness score. Minimizing the solvation ratio allows for negative fitness scores, and thus roulette and SUS could not be used. For the performance evaluation of these two methods, 10 runs were run for polarizability and optical bandgap for a total of 20 runs. The other selection methods were run with the typical 5 runs per property, resulting in 15 total runs per selection method.

Figure 8a shows the distribution of each method for the champion, coverage, and speedup. All methods had a median champion of 1, although random, 3-way and 4-way tournament, and rank selection had more consistent high-performing results. Selecting parents with random selection from the entire population compared to only the top 50% found the champion more frequently and shows a higher median coverage. A possible explanation is restricting selection to the top candidates limits the explorative abilities of the GA. Some monomers found in the poor performers may perform really well when paired with a new monomer, and impeding the GA from selecting these makes it difficult to find all high performers. The selection methods performed similarly for coverage, and all the methods showed comparable outliers. Looking at the individual run performance for champion and coverage (ESI Tables S10 and S11, respectively), all methods performed worse on optimizing the optical bandgap. ESI Figure S9 shows 2-way and 4-way tournament selection led to worse champions for optimizing the optical bandgap, although performed very well on the other 2 properties. The speedup is also similar across all methods (ESI Table S12), with a 3-way tournament selection yielding the highest median speedup. Using ANOVA, comparing methods for the champion, coverage, and speedup show they are statistically indistinguishable ( $p > 0.05$ ).

Studying the balance between speedup and coverage shows that tournament selection is the clear winner. This type of selection method was able to efficiently search through chemical space

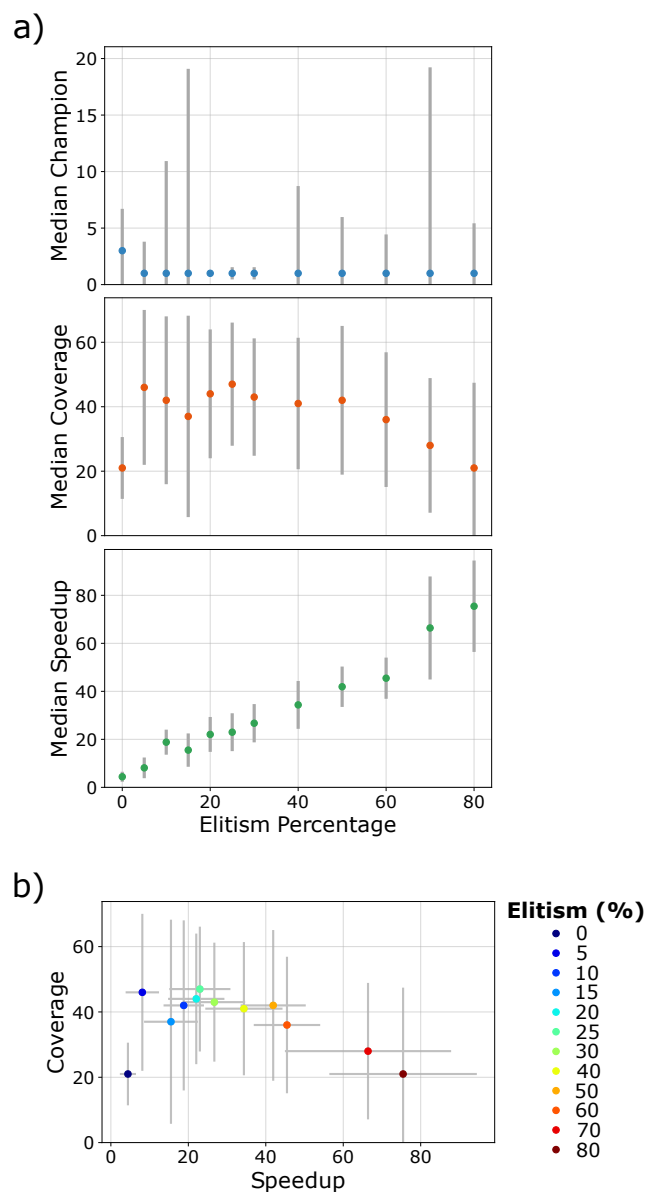


FIG. 7. Performance metrics for evaluating elitism at 0, 5, 19, 15, 20, 25, 30, 40, 50, 60, 70, and 80%. (a) The champion is defined as the rank of the best polymer found. The coverage is the percentage of the top 100 polymers discovered by the GA. The speedup is the size of the search space / total number of calculations performed. The median from among 15 runs (5 per chemical property) is displayed, with the standard deviation as error bars. (b) Comparison of coverage vs speedup to determine which elitism percentage gives the best balance, with the standard deviation as error bars.

to find high-performance materials. 3-way and 4-way tournament selection performed very well, with 3-way showing larger speedup and 4-way showing higher coverage. In ESI Figure S9, 3-

way tournament showed the best balance of coverage and speedup for polarizability and solvation energy, while 4-way tournament appeared slightly better for optical bandgap (although led to worse champions). For future use, a 3-way tournament selection is recommended due to the large speedups and high coverage.

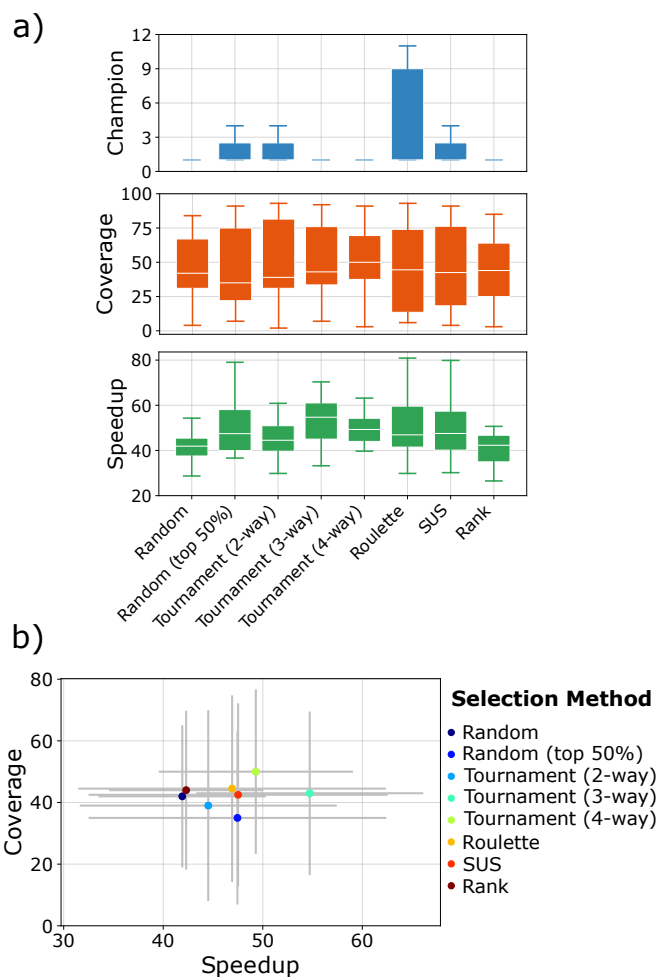


FIG. 8. Performance metrics for evaluating selection methods, such as random, random from the top 50% of candidates, tournament (2-, 3-, and 4-way), roulette, stochastic universal sampling (SUS), and rank selection. (a) The champion is defined as the rank of the best polymer found. The coverage is the percentage of the top 100 polymers discovered by the GA. The speedup is the size of the search space / total number of calculations performed. The median from among 15 runs (5 per chemical property), or 20 runs for SUS and roulette, is displayed, with the standard deviation as error bars. (b) Comparison of coverage vs speedup to determine which selection method gives the best balance, with the standard deviation as error bars.

#### **4. Mutation Rate**

Mutation rates ranging from 1% to 90% are examined to find the best balance between introducing diversity into the generation while keeping enough unchanged to explore nearby space. Figure 9a shows the median champion, coverage, and speedup of all mutation rates in 15 runs. Mutation rates of 10% and above had a median champion of 1, with very low standard deviations. Low mutation rates of 1% and 5% found champions of median rank 14 and 4, respectively, with very large standard deviations. Examining the individual runs (ESI Table S13) revealed that one run found a champion of 1,180 for 1% and 5% mutation rates, with multiple others finding a champion above 100. With mutation rates of 10% and higher, the worst champion dramatically decreases.

The coverage increase as the mutation rate increases, until around 30% mutation where the median coverage remains relatively constant at around 40% coverage. The individual runs (ESI Table S14) reveal a mutation rate of 1% or 5% produced many runs with 0% or 1% coverage. The poor performance of low mutation rates stems from the lack of diversity introduced into the population. The performance of the GA is highly dependent on the initial population, so if there are no good individuals initially, it is extremely difficult to search other areas of chemical space. When the mutation rate is increased to above 30%, the amount of diversity required is most likely saturated. Since elitism is used in these GAs, good traits will always remain in the population so if a new monomer is introduced during mutation and it does not perform well, it will just be discarded, and there are still high performers for the next generation. Surprisingly, the coverage does not decrease as would be expected from less opportunity to locally search after crossover.

Comparing the coverage and speedup shows that a mutation rate of 40% has similar coverage to 30%, 50%, 70%, 80%, and 90% mutation, but much larger speedups (ESI Table S15). Examination of this trade-off for chemical properties individually (ESI Figure S10) showed 40% mutation gave the best balance of coverage and speedup for all 3 properties. For future GAs, 40% mutation rate is recommended.

#### **D. Realistic Trial: Larger Search Space**

As a final phase of testing, we ran a realistic scenario trial in which we used the best GA convergence criteria and hyperparameters found in earlier testing to search for hexamers with the

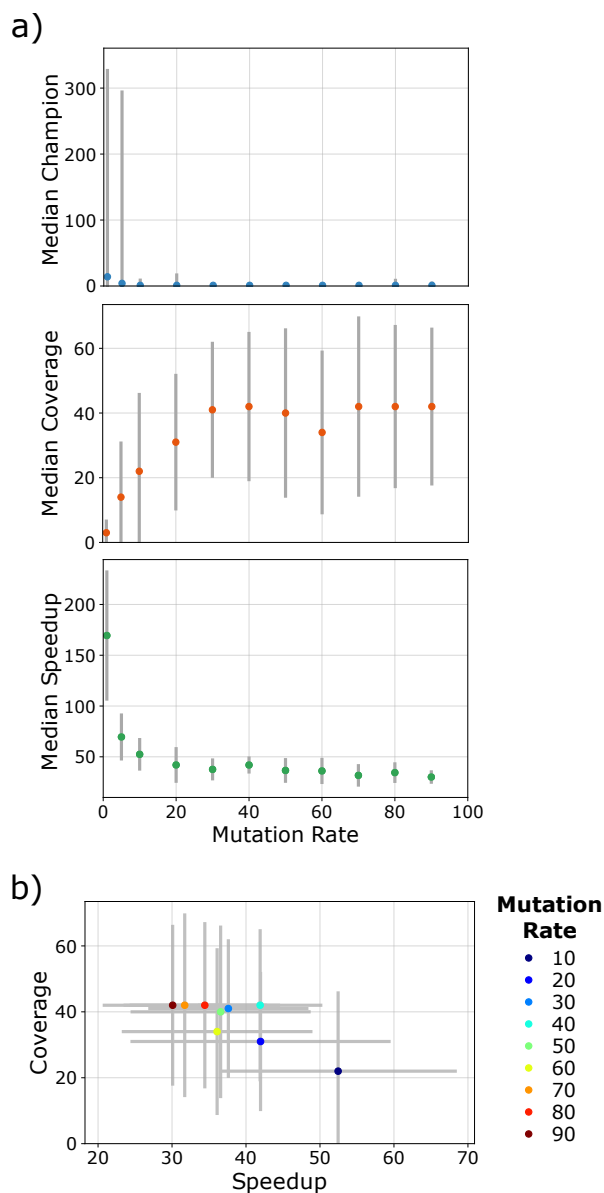


FIG. 9. Performance metrics for evaluating mutation rates of 1, 5, 10, 20, 30, 40, 50, 60, 70, 80, and 90%. (a) The champion is defined as the rank of the best polymer found. The coverage is the percentage of the top 100 polymers discovered by the GA. The speedup is the size of the search space / total number of calculations performed. The median from among 15 runs (5 per chemical property) is displayed, with the standard deviation as error bars. (b) Comparison of coverage vs speedup to determine which mutation rate gives the best balance, with the standard deviation as error bars. 1% and 5% are excluded from the figure due to very poor coverage and speedup.

same three optimized properties, but in a much larger search space. For this phase, our monomer

list is expanded to include SMILES for 1200 unique units, including more common repeat units as well as aryl-vinyl and aryl-azo combinations. Contrary to the work reported thus far, this part of the project allowed any of the 64 possible sequences, instead of limiting it to ABABAB. This yielded a new search space of approximately 46 million possible hexamers, increasing its size by two orders of magnitude in comparison to our originally defined search space. This allows us to see how our recommended best practices performed in a more realistic setting. We again ran five trials with unique starting random states for each of the three properties covered.

Looking at the results for the realistic trial, we saw that all runs met or exceeded the true champion values for the limited search space used in hyperparameter testing, indicating that even in a vastly larger search space the GA was able to efficiently find the top values. The chemical structures of the champions from the previously limited search space, as well as from this realistic trial, can be seen in ESI Figures S11-S13. As shown in Figure 10, individual runs within a given property search did not always converge to the same top performer. The figure also demonstrates that different properties resulted in varying levels of agreement between individual runs. This is likely due to differences in the "roughness" of the search spaces defined for each of the properties, with the polarizability search space being considerably smoother than the solvation energy search space. The polarizability GAs had several runs converge to the same champion while the solvation energy GA's individual runs all converged to different champions. The differences in the convergence between individual runs reinforce the need to perform several trials when possible for any given GA search, especially when a goal is to get as close to the global extremum as possible.

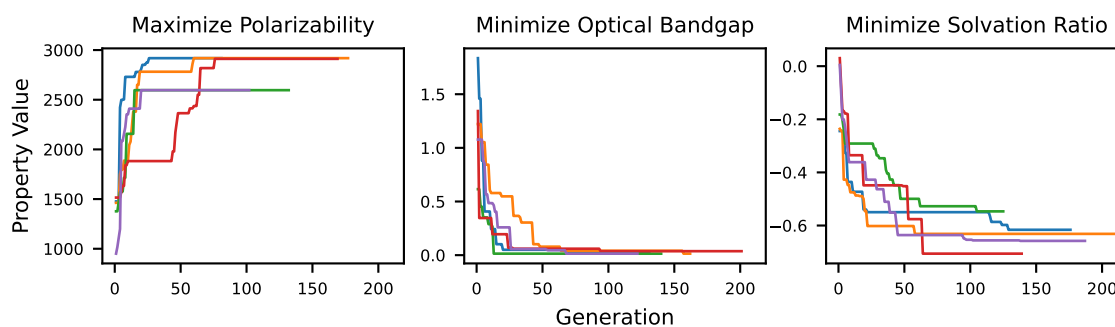


FIG. 10. Convergence of the GA for 5 runs of the three property searches is shown by plotting the top performer in each generation against the generation number.

In order to examine how the GA performed in terms of elite performer coverage, we found the top 10% of values across all 5 runs for each chemical property and then compared the results of

each individual run to this elite pool (ESI Figure S14). We note that no single run captured even a majority of this elite pool and some runs captured very little, even less than one percent. This leads us to recommend that multiple GA runs are highly suggested when practical to allow the best coverage of elite search space.

Finally, we examine the popular chemical motifs found by the GA in each of the property searches. Looking at the most commonly used monomers in the top 10% by fitness across all runs of each property search, we saw some common themes. As shown in Figure 11, the top monomers for polarizability are notably longer, larger molecules than those found in the other two property searches. This makes chemical sense considering that longer conjugated systems allow for greater charge mobility and therefore greater polarizability. While the top monomers for the optical bandgap search do not share especially close motifs, we know that extended conjugation, such as having a vinylene group on unit 273, redshifts the absorption and decreases the bandgap<sup>3,59</sup>. Additionally, recent studies on non-fullerene acceptors show the lone pairs on the nitrogens in unit 539 can delocalize to further reduce the bandgap<sup>60</sup>. The solvation ratio top monomers interestingly tend to contain sulfonyl groups, which are highly polar and hydrophilic. We also determined the most commonly used sequences in the top 10% by fitness for each property search (ESI Figure S15). Although we did not find obvious trends in the top sequences for either optical bandgap or solvation ratio, the top polarizability sequences were all near-homopolymers and occurred with incredibly similar rates of incidence. This supports previous findings in our group that homopolymeric sequences tend to increase overall molecular polarizability (near homopolymer sequences occur often simply because they are statistically more likely to occur and have extremely similar actual polarizabilities compared to true homopolymers).<sup>61</sup> While a more in-depth analysis of the chemical motifs found in our realistic GA trial is beyond the scope of this work, our preliminary findings support the utility of implementing our GA best practices when conducting searches for a range of different optimized chemical properties.

#### IV. CONCLUSION

After examining the general effects of a number of hyperparameters on GA performance for chemical space searches, we suggest several "best practices" for general use in this area.

1. Using the Spearman correlation coefficient in combination with a convergence generation counter was found to be an effective way to terminate GA runs in a consistent, systematic

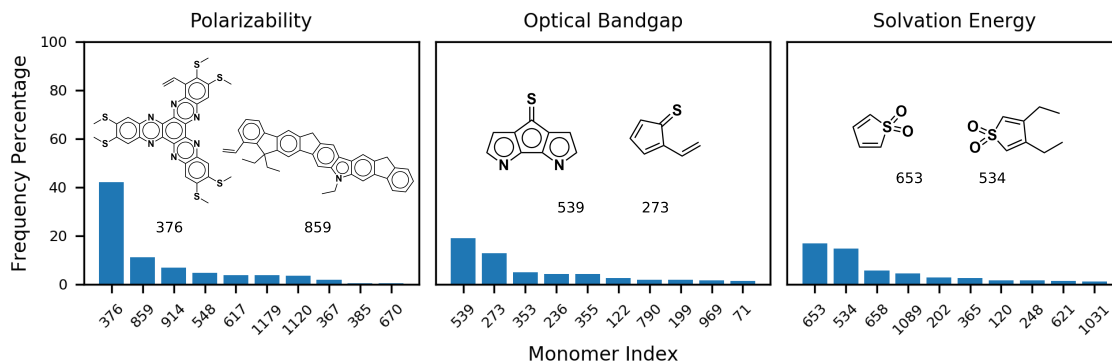


FIG. 11. The monomer indices of the most commonly used monomers in the top 10% by fitness across all runs of each property search are shown, along with the chemical structure of the top two monomers for each property value. The monomers indices refer to the index, or position, in the SMILES monomer database ([https://github.com/hutchisonlab/GA-Best-Practices/blob/main/monomer\\_SMILES.csv](https://github.com/hutchisonlab/GA-Best-Practices/blob/main/monomer_SMILES.csv)).

manner.

- The best metrics found for GA convergence is when a Spearman coefficient threshold of 0.8 is met or exceeded for 50 consecutive generations.
2. The best GA hyperparameters are a population size of 32, elitism rate of 50%, selection of parents with 3-way tournament selection, and mutation rate of 40%.
    - These parameters give the best balance of finding the overall champion, maintaining good elite coverage, and enhancing relative speedup for general use in chemical GAs.
    - If discovering a large number of high-performing candidates is more important than finding the champion, coverage can be greatly improved by increasing the population size and mutation rate. Note that this does require sacrificing some computational efficiency.
  3. We strongly recommend running a GA multiple times to ensure a good group of final candidates.

With these general recommendations, we acknowledge several caveats. While we believe our convergence method to be useful and an important step in automating molecular GA methods, further work is needed to explore the potential of breaking through the perceived Pareto front that currently exists at the tradeoff between elite search space coverage and speedup. Additionally,

such methods could also potentially reduce the need for multiple GA runs if the convergence method is able to better ensure consistent coverage of elite search spaces. Although this work was performed with polymer GAs, we believe the results should transfer to other types of molecular GAs. Using three different chemical properties as optimization targets, we demonstrated that the GA best practices suggested in this work are suited for a wide and varied range of molecular discovery applications.

These best practices are tested in a realistic search scenario, where we found candidate oligomers tailored specifically to each of the three chemical properties explored. After vastly expanding our search space, the GA runs in these trials were able to self-terminate appropriately and found candidates as good as or better than the best candidates in our original limited search space. This indicates that our GA method with self-termination and tuned hyperparameters is proficient in efficiently locating top chemical structures for a variety of different properties and is a recommended starting place for general use.

## **V. DATA AVAILABILITY**

The genetic algorithms, data analysis, and the data that supports the findings of this study are openly available at: <https://github.com/hutchisonlab/GA-Best-Practices>

## **VI. SUPPLEMENTARY MATERIAL**

See supplementary material for visualizations of chemical space using t-SNE and UMAP projections, Tanimoto similarity between top 100 candidates for each property, tables of champions, coverage, and speedup, and comparisons of speedup and coverage.

## **AUTHOR CONTRIBUTIONS**

Brianna Greenstein and Danielle Elsey contributed equally to this work in conceptualization, software, investigation, and analysis. Brianna Greenstein was the primary writer with Danielle Elsey writing all sections dealing with convergence criteria and the realistic trial.

## CONFLICTS OF INTEREST

There are no conflicts to declare.

## ACKNOWLEDGMENTS

We acknowledge support from the Department of Energy-Basic Energy Sciences Computational and Theoretical Chemistry (Award DE-SC0019335) and the University of Pittsburgh Center for Research Computing through the computational resources provided. B.G. thanks the Pittsburgh Quantum Institute Fellowship for partial support.

## REFERENCES

- <sup>1</sup>A. Nigam, R. Pollice, G. Tom, K. Jorner, L. A. Thiede, A. Kundaje, and A. Aspuru-Guzik, “Tartarus: A benchmarking platform for realistic and practical inverse molecular design,” arXiv (2022), 10.48550/ARXIV.2209.12487.
- <sup>2</sup>A. Nigam, R. Pollice, and A. Aspuru-Guzik, “Parallel tempered genetic algorithm guided by deep neural networks for inverse molecular design,” *Digital Discovery*, 10.1039.D2DD00003B (2022).
- <sup>3</sup>B. L. Greenstein, D. C. Hiener, and G. R. Hutchison, “Computational evolution of high-performing unfused non-fullerene acceptors for organic solar cells,” *The Journal of Chemical Physics* **156**, 174107 (2022).
- <sup>4</sup>N. Ree, M. Koerstz, K. V. Mikkelsen, and J. H. Jensen, “Virtual screening of norbornadiene-based molecular solar thermal energy storage systems using a genetic algorithm,” *The Journal of Chemical Physics* **155**, 184105 (2021).
- <sup>5</sup>B. Sanchez-Lengeling and A. Aspuru-Guzik, “Inverse molecular design using machine learning: Generative models for matter engineering,” *Science* **361**, 360–365 (2018).
- <sup>6</sup>B. Sanchez-Lengeling, C. Outeiral, G. L. Guimaraes, and A. Aspuru-Guzik, “Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic),” *ChemRxiv* (2017), 10.26434/chemrxiv.5309668.v3.
- <sup>7</sup>D. Douguet, E. Thoreau, and G. Grassy, “[no title found],” *Journal of Computer-Aided Molecular Design* **14**, 449–466 (2000).

- <sup>8</sup>D. C. Hiener and G. R. Hutchison, “Pareto optimization of oligomer polarizability and dipole moment using a genetic algorithm,” *The Journal of Physical Chemistry A* **126**, 2750–2760 (2022).
- <sup>9</sup>O. D. Abarbanel and G. R. Hutchison, “Using genetic algorithms to discover novel ground-state triplet conjugated polymers,” *Physical Chemistry Chemical Physics* **25**, 11278–11285 (2023).
- <sup>10</sup>J. H. Jensen, “A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space,” *Chemical Science* **10**, 3567–3572 (2019).
- <sup>11</sup>T. C. Le and D. A. Winkler, “Discovery and optimization of materials using evolutionary approaches,” *Chemical Reviews* **116**, 6107–6132 (2016).
- <sup>12</sup>B. C. Rinderspacher, “Heuristic global optimization in chemical compound space,” *The Journal of Physical Chemistry A* **124**, 9044–9060 (2020).
- <sup>13</sup>B. L. Greenstein and G. R. Hutchison, “Screening efficient tandem organic solar cells with machine learning and genetic algorithms,” *The Journal of Physical Chemistry C* **127**, 6179–6191 (2023).
- <sup>14</sup>A. Nigam, R. Pollice, and A. Aspuru-Guzik, “Parallel tempered genetic algorithm guided by deep neural networks for inverse molecular design,” *Digital Discovery* **1**, 390–404 (2022).
- <sup>15</sup>L. E. de Sousa, D. A. Silva Filho, P. de Silva, L. Ribeiro, and P. H. Oliveira Neto, “A genetic algorithm approach to design principles for organic photovoltaic materials,” *Advanced Theory and Simulations* **3**, 2000042 (2020).
- <sup>16</sup>E. S. Henault, M. H. Rasmussen, and J. H. Jensen, “Chemical space exploration: How genetic algorithms find the needle in the haystack,” *ChemRxiv* (2020), 10.26434/chemrxiv.12152661.v1.
- <sup>17</sup>N. M. O’Boyle, C. M. Campbell, and G. R. Hutchison, “Computational design and selection of optimal organic photovoltaic materials,” *The Journal of Physical Chemistry C* **115**, 16200–16210 (2011).
- <sup>18</sup>B. A. Day and C. E. Wilmer, “Genetic algorithm design of MOF-based gas sensor arrays for CO<sub>2</sub>-in-air sensing,” *Sensors* **20**, 924 (2020).
- <sup>19</sup>D. Hibbert, “Genetic algorithms in chemistry,” *Chemometrics and Intelligent Laboratory Systems* **19**, 277–293 (1993).
- <sup>20</sup>R. L. Johnston, “Evolving better nanoparticles: Genetic algorithms for optimising cluster geometries,” *Dalton Transactions* , 4193 (2003).

- <sup>21</sup>R. Leardi, “Genetic algorithms in chemistry,” *Journal of Chromatography A* **1158**, 226–233 (2007).
- <sup>22</sup>F. Curtis, X. Li, T. Rose, A. Vázquez-Mayagoitia, S. Bhattacharya, L. M. Ghiringhelli, and N. Marom, “Gator: A first-principles genetic algorithm for molecular crystal structure prediction,” *Journal of Chemical Theory and Computation* **14**, 2246–2264 (2018).
- <sup>23</sup>Z. Falls, P. Avery, X. Wang, K. P. Hilleke, and E. Zurek, “The xtalopt evolutionary algorithm for crystal structure prediction,” *The Journal of Physical Chemistry C* **125**, 1601–1620 (2021).
- <sup>24</sup>M. P. Lourenço, J. Hostas, L. B. Herrera, P. Calaminici, A. M. Koster, A. Tchagang, and D. R. Salahub, “Gamaterial—a genetic-algorithm software for material design and discovery,” *Journal of Computational Chemistry* **44**, 814–823 (2023).
- <sup>25</sup>M. K. Bisbo and B. Hammer, “Global optimization of atomic structure enhanced by machine learning,” *Physical Review B* **105**, 245404 (2022).
- <sup>26</sup>O. Yañez, A. Vásquez-Espinal, D. Inostroza, L. Ruiz, R. Pino-Rios, and W. Tiznado, “A fukui function-guided genetic algorithm. assessment on structural prediction of si n ( n = 12-20) clusters,” *Journal of Computational Chemistry* **38**, 1668–1677 (2017).
- <sup>27</sup>RDKit, online, “RDKit: Open-source cheminformatics,” <http://www.rdkit.org> (2022), [Online; accessed 11-April-2013].
- <sup>28</sup>S. R. M. Pereira, F. Clerc, D. Farrusseng, J. C. van der Waal, T. Maschmeyer, and C. Mirodatos, “Effect of the genetic algorithm parameters on the optimisation of heterogeneous catalysts,” *QSAR & Combinatorial Science* **24**, 45–57 (2005).
- <sup>29</sup>J. E. Baker *et al.*, “Reducing bias and inefficiency in the selection algorithm,” in *Proceedings of the second international conference on genetic algorithms*, Vol. 206 (1987) pp. 14–21.
- <sup>30</sup>J. E. Baker, “Adaptive selection methods for genetic algorithms,” in *Proceedings of an International Conference on Genetic Algorithms and their applications*, Vol. 1 (Hillsdale, New Jersey, 1985).
- <sup>31</sup>C. Steinmann and J. H. Jensen, “Using a genetic algorithm to find molecules with good docking scores,” *PeerJ Physical Chemistry* **3**, e18 (2021).
- <sup>32</sup>S. Zheng, Y. Wang, W. Liu, W. Chang, G. Liang, Y. Xu, and F. Lin, “In silico prediction of hemolytic toxicity on the human erythrocytes for small molecules by machine-learning and genetic algorithm,” *Journal of Medicinal Chemistry* **63**, 6499–6512 (2020).
- <sup>33</sup>Y. Kwon, S. Kang, Y.-S. Choi, and I. Kim, “Evolutionary design of molecules based on deep learning and a genetic algorithm,” *Scientific Reports* **11**, 17304 (2021).

- <sup>34</sup>B. San, Z. Xiao, and Y. Qiu, “Simultaneous shape and stacking sequence optimization of laminated composite free-form shells using multi-island genetic algorithm,” *Advances in Civil Engineering* **2019**, 1–14 (2019).
- <sup>35</sup>I. Y. Kanal, S. G. Owens, J. S. Bechtel, and G. R. Hutchison, “Efficient computational screening of organic polymer photovoltaics,” *The Journal of Physical Chemistry Letters* **4**, 1613–1623 (2013).
- <sup>36</sup>A. Nigam, P. Friederich, M. Krenn, and A. Aspuru-Guzik, “Augmenting genetic algorithms with deep neural networks for exploring the chemical space,” arXiv:1909.11655 [physics] (2020).
- <sup>37</sup>J. Verhellen and J. Van den Abeele, “Illuminating elite patches of chemical space,” *Chemical Science* **11**, 11485–11491 (2020).
- <sup>38</sup>J. Verhellen, “Graph-based molecular pareto optimisation,” *Chemical Science* **13**, 7526–7535 (2022).
- <sup>39</sup>K. J. Kron, A. Rodriguez-Katakura, P. Regu, M. N. Reed, R. Elhessen, and S. Mallikarjun Sharada, “Organic photoredox catalysts for co 2 reduction: Driving discovery with genetic algorithms,” *The Journal of Chemical Physics* **156**, 184109 (2022).
- <sup>40</sup>W.-W. Zhang, H. Qi, Z.-Q. Yu, M.-J. He, Y.-T. Ren, and Y. Li, “Optimization configuration of selective solar absorber using multi-island genetic algorithm,” *Solar Energy* **224**, 947–955 (2021).
- <sup>41</sup>J. A. Gustafson and C. E. Wilmer, “Intelligent selection of metal–organic framework arrays for methane sensing via genetic algorithms,” *ACS Sensors* **4**, 1586–1593 (2019).
- <sup>42</sup>E. S. Henault, M. H. Rasmussen, and J. H. Jensen, “Chemical space exploration: how genetic algorithms find the needle in the haystack,” *PeerJ Physical Chemistry* (2020), 10.7717/peerj-pchem.11.
- <sup>43</sup>C. Steinmann and J. H. Jensen, “Using a genetic algorithm to find molecules with good docking scores,” *PeerJ Physical Chemistry* **3**, e18 (2021).
- <sup>44</sup>E. Bozkurt, M. A. S. Perez, R. Hovius, N. J. Browning, and U. Rothlisberger, “Genetic algorithm based design and experimental characterization of a highly thermostable metalloprotein,” *J. Am. Chem. Soc.* **140**, 4517–4521 (2018).
- <sup>45</sup>D. J. Kozuch, F. H. Stillinger, and P. G. Debenedetti, “Genetic algorithm approach for the optimization of protein antifreeze activity using molecular simulations,” *J. Chem. Theory Comput.* **16**, 7866–7873 (2020).

- <sup>46</sup>H. Yang and M. W. Wong, “Automatic conformational search of transition states for catalytic reactions using genetic algorithm,” *J. Phys. Chem. A* **123**, 10303–10314 (2019).
- <sup>47</sup>I. Y. Kanal and G. R. Hutchison, “Rapid computational optimization of molecular properties using genetic algorithms: Searching across millions of compounds for organic photovoltaic materials,” (2017).
- <sup>48</sup>J. L. Myers and A. Well, *Research design and statistical analysis* (L. Erlbaum, 2003).
- <sup>49</sup>T. A. Halgren, “Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94,” *Journal of Computational Chemistry* **17**, 490–519 (1996).
- <sup>50</sup>N. M. O’Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, “Open babel: An open chemical toolbox,” *Journal of Cheminformatics* **3**, 33 (2011).
- <sup>51</sup>C. Bannwarth, E. Caldeweyher, S. Ehlert, A. Hansen, P. Pracht, J. Seibert, S. Spicher, and S. Grimme, “Extended tight-binding quantum chemistry methods,” *WIREs Computational Molecular Science* **11** (2021), 10.1002/wcms.1493.
- <sup>52</sup>C. Bannwarth, S. Ehlert, and S. Grimme, “Gfn2-xtb-an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions,” *Journal of Chemical Theory and Computation* **15**, 1652–1671 (2019).
- <sup>53</sup>S. Grimme, C. Bannwarth, and P. Shushkov, “A robust and accurate tight-binding quantum chemical method for structures, vibrational frequencies, and noncovalent interactions of large molecular systems parametrized for all spd-block elements ( $z = 1-86$ ),” *Journal of Chemical Theory and Computation* **13**, 1989–2009 (2017).
- <sup>54</sup>S. Grimme and C. Bannwarth, “Ultra-fast computation of electronic spectra for large systems by tight-binding based simplified tamm-dancoff approximation (stda-xtb),” *The Journal of Chemical Physics* **145**, 054103 (2016).
- <sup>55</sup>S. Grimme, “A simplified tamm-dancoff density functional approach for the electronic excitation spectra of very large molecules,” *The Journal of Chemical Physics* **138**, 244104 (2013).
- <sup>56</sup>P. Pracht, F. Bohle, and S. Grimme, “Automated exploration of the low-energy chemical space with fast quantum chemical methods,” *Physical Chemistry Chemical Physics* **22**, 7169–7192 (2020).
- <sup>57</sup>L. McInnes, J. Healy, and J. Melville, “Umap: Uniform manifold approximation and projection for dimension reduction,” (2018).

- <sup>58</sup>L. van der Maaten and G. E. Hinton, “Visualizing data using t-sne,” *Journal of Machine Learning Research* **9**, 2579–2605 (2008).
- <sup>59</sup>J. Hai, S. Luo, H. Yu, H. Chen, Z. Lu, L. Li, Y. Zou, and H. Yan, “Achieving ultra-narrow bandgap non-halogenated non-fullerene acceptors via vinylene  $\pi$ -bridges for efficient organic solar cells,” *Materials Advances* **2**, 2132–2140 (2021).
- <sup>60</sup>Y. Chen, T. Liu, L.-K. Ma, W. Xue, R. Ma, J. Zhang, C. Ma, H. K. Kim, H. Yu, F. Bai, K. S. Wong, W. Ma, H. Yan, and Y. Zou, “Alkoxy substitution on idt-series and y-series non-fullerene acceptors yielding highly efficient organic solar cells,” *Journal of Materials Chemistry A* **9**, 7481–7490 (2021).
- <sup>61</sup>D. C. Hiener, D. L. Folmsbee, L. A. Langkamp, and G. R. Hutchison, “Evaluating fast methods for static polarizabilities on extended conjugated oligomers,” *Physical Chemistry Chemical Physics* **24**, 23173–23181 (2022).