Genetic algorithms in computational materials science and engineering: simulation and design of self-assembling materials

Milan Keser 1, Samuel I. Stupp * 2

Departments of Materials Science and Engineering, Chemistry and Medical School, Northwestern University, Evanston, IL 60208, USA

Received 30 March 1999

Abstract

We introduce here two genetic algorithms that were developed in order to aid in the design of molecules for self-assembling materials. The first constructs molecules from sets of chemical building blocks, searching for candidates that are determined by an ancillary modeling program to assemble into low-energy aggregates. The results of running this Genetic Algorithm (GA) on a set of building blocks are discussed in the context of experimental observations on molecules synthesized from these chemical components. The second genetic algorithm attempts to find the most favorable configuration of four molecules in space, as determined by an empirical molecular mechanics force field. We present the results of the application of this GA to molecules that have been studied experimentally in our laboratory. The two genetic algorithms promise to be of use not only in the context in which they are presented, but also in a wide variety of future applications in molecular design and modeling. © 2000 Elsevier Science S.A. All rights reserved.

1. Motivation

A significant challenge in the fields of materials science and synthetic chemistry is the design of molecules that self-assemble into specific shape-persistent supramolecular nanostructures of diverse shape and function. The covalent stitching of discrete nanoaggregates into nanoscale molecular objects could allow for a synthetic route to a very interesting and useful class of nanostructured materials. Proteins, for example, are synthesized as monodisperse linear chain copolymers, but are encoded with a strong tendency to assume one specific three-dimensional structure. This structure often defines the function of the protein, enabling it to perform as a structural element, catalyst, or receptor. The ability to create nanoscale structures with strategically located chemical functionality and specified shape would provide the ability to conduct coordination chemistry on a new scale, connecting nanostructures rather than ions or small molecules. Controlling the location and relative placement of functional groups on and within the nanostructures, one could design 2D or 3D networks, materials with well-defined nanoporosity, and chemically or topologically well-defined surfaces. Design of these structural features could offer access to materials with novel electrical, optical, transport, catalytic, or adhesive properties.

Recent investigations of the materials formed by self-assembling molecules have raised important questions about the nature of the assembly process, such as the role of preferred molecular conformation [1], the importance of functional group location, factors leading to polar order, and factors enabling formation of discrete nanostructures [2–9]. Due to the difficulty and expense of the laboratory synthesis of the...
large organic molecules currently being explored for their self-assembling behavior, any means of predicting relevant properties of a molecule in advance of its synthesis would be extremely useful. The most basic use of such methods would be to rank proposed synthetic targets in order to eliminate the least promising. For example, computational methods developed for the prediction of preferred molecular conformations and modes of aggregation could be used to predict the likelihood that a given molecule would form a particular desired type of supramolecular cluster. A very interesting and more advanced use of these methods arises when they are combined with an automated method of generating new chemical structures. The combination could provide the ability to completely or partially automate the design of new molecules. A molecular design program could generate and evaluate thousands of potential new molecules in order to provide a list of the most promising structures for any application, provided there exists a sufficient predictive capability.

In the first part of this work we explore an automated method of designing new molecules for self-assembly. A molecule’s ability to self-assemble in a specific arrangement is estimated by placing several copies of the molecule into a tight cluster and then relaxing that cluster by conjugate gradient minimization. The energy of the relaxed cluster is used as a measure of how likely it would be for the molecule to assemble into the desired structure. We note that the energy calculated will almost certainly be a local minimum, determined by the particular arrangement of the molecules in the cluster. This limitation is addressed in the second part of this work, where a method is developed to search for the globally optimal arrangement of four copies of a molecule in space. Both programs were developed to work within the molecular modeling package Sybyl (version 6.1, Tripos Associates, St. Louis, MO), which provides a graphical interface with which to view the molecules as well as the Tripos force field for estimation of molecular energy. The programs are run on a Silicon Graphics Indigo2 XZ workstation. In the development of each of these tools, a method was needed to optimize or search a complex highly dimensional solution hyperspace containing many locally optimal solutions. Where traditional mathematical and stochastic search methods were inadequate, two genetic algorithms were developed and applied.

2. Force field energy evaluation

In all of the work presented below, energy evaluations were performed using an empirical force field [10]. While more accurate methods are available, force field energy evaluations are very fast, and allow the many energy calculations required by the following methods to be performed in a reasonable amount of time. The energy of a particular arrangement of atoms is calculated according to

\[ E = E_{\text{stretch}} + E_{\text{bend}} + E_{\text{torsion}} + E_{\text{nonbonded}}, \]

where the total energy of a molecule is represented as a sum of terms representing bond stretching, bond angle bending, dihedral angle, and non-bonded energies. The bond length term is represented by a harmonic potential of the form

\[ E_{\text{stretch}} = \sum_{\text{bonds}} k_b (l - l_0)^2, \]

where \( k_b, l \) and \( l_0 \) represent the bond spring constant, the bond length, and the ideal bond length, respectively, and will vary for each pair of atoms in the molecule, depending on their type as well as electronic state. The bond angle bending term is also represented by a harmonic potential

\[ E_{\text{bend}} = \sum_{\text{bonds}} k_b (\theta - \theta_0)^2, \]

where \( k_b, \theta \) and \( \theta_0 \) represent the bending constant, bond angle, and ideal bond angle for each bonded combination of three atoms. The non-bonded interaction term contains a van der Waals term and a coulombic term.
where \( \varepsilon \) is the Lennard–Jones well depth, \( r_0 \) the optimal interatomic distance, \( r \) and \( r_{ij} \) represent interatomic distance, \( Q \) the atomic charge, and \( D \) is the dielectric constant. Because the simple van der Waals interactions of (4) do not adequately model the twisting of a molecule about a rotatable bond, a dihedral angle twisting term was explicitly included in most force fields as

\[
E_{\text{twist}} = \sum_{\text{dihedrals}} V_i(1 + s_i \cos n \omega),
\]

where \( V_i \) represents the height of the periodic torsional barrier, \( s_i \) is 0 for staggered and 1 for eclipsed minima, \( n \) is the periodicity of rotation, and \( \omega \) is the dihedral angle. This term, together with the dihedral angle portion of the non-bonded term represents the energy of twisting the rotatable bonds in a molecule.

Three Design of molecules for self-assembly

A number of molecules recently synthesized in our laboratory have been observed to self-assemble into interesting nanostructures. While modeling many of these molecules in the computer, we began to look for ways to explore novel structures computationally, before investing in their synthesis. Since the principle interest was in investigating the packing modes of these molecules, a routine was developed to quickly build and evaluate clusters of several molecules. Given a molecule, the calculation returns an estimate of that molecule's ability to pack in some given mode of aggregation.

Because extended conformations are more likely to effectively pack into a tight, ordered cluster, the cluster evaluation begins with a search for an extended low-energy conformation of the given molecule. In order to arrive at such a conformation from virtually any starting conformation, two gradient minimizations are performed. First, a strong repulsive potential is imposed between the two ends of the molecule, and the conformation is relaxed, resulting in a physically unrealistic stretched linear conformation. Next, the artificial repulsive term is removed from the force field, and the conformation is again relaxed. The result, as shown in Fig. 1, is an extended energy-minimized conformation, which is a local minimum in conformational space. Even though this extended conformation may be a higher energy state for a single molecule than some coiled state, the tight packing of several molecules into a cluster would likely cause the molecules to seek extended conformations. Extended conformations, in general, can pack together more efficiently than coiled conformations can, and are better able to maximize favorable intermolecular van der Waals interactions.

Because of the strong repulsive nature of (4) when molecules are brought into very close proximity, the generation of too tight a cluster will result in an extremely high energy being calculated by the force field. In such a case, gradient minimization tends to have a great deal of trouble finding any reasonable solution. In order to avoid generating too tight a cluster, the following method is used to predict an intermolecular spacing for the cluster that can be easily handled by conjugate gradient minimization. First, a pair of molecules is brought together until there is a slight repulsion between them. This pair is then relaxed, allowing the two molecules to drift apart, and the spacing between them after relaxation is noted. Another pair of molecules is then brought together with a different relative orientation than the first pair in order to
calculate the intermolecular spacing in the cluster along a different direction relative to the molecule. This pair is also relaxed, and the separation between the molecules is noted. The details of this process are shown in Fig. 2. Next, 7 or 13 copies of the molecule are then packed into a tight cluster, with spacings determined by the procedure outlined above, and the cluster is relaxed using conjugate gradient descent [14], as shown in Fig. 3. Clusters of 13 molecules take about three times as long to minimize as clusters of seven. The end product of the procedure is a relaxed but still tightly packed bundle of molecules, and a force field estimate of its energy.

Having developed the above means of estimating any given molecule’s self-assembling ability, the problem of designing molecules for self-assembly can be treated as the minimization of relaxed cluster energy as a function of molecular structure. A genetic algorithm was developed to perform this
minimization. For the purposes of this GA, a molecule is defined as a string of covalently connected chemical building blocks. Each population member is then simply an ordered list of integer building block codes, representing a unique molecule, as shown in Fig. 4. The current library of available building blocks is shown in Fig. 5. A molecule is built by adding any number of internal groups to an end group, and then adding a second end group. To allow for greater control of the molecules designed, the left end and right end groups can be drawn from separate libraries. If the search space is taken as molecules having two end groups, of which there are eight, and five internal groups, of which there are 22, there are then $8^2 \times 22^5 = 165$ million different molecules available. Since the evaluation of a molecule takes several minutes, it is not possible to exhaustively search the solution space in a reasonable amount of time. Genetic algorithms, on the other hand, have been applied successfully in many very difficult search problems, and therefore offered a potential means of performing this minimization.

The GA was developed as follows. In order to allow the user to select which members of the building block set to use for each run of the GA, it was most convenient to use a real-coding for the GA’s population. Under this coding, each population member is a list of integer building block id’s and is interpreted as

$$\{ \text{l-end}, \text{internal}_1, \text{internal}_2, \text{internal}_3, \ldots, \text{internal}_n, \text{r-end} \}$$
To calculate the fitness of a population member, the molecule it represents is constructed in Sybyl and can then be submitted to any desired evaluation subroutine. By changing the definition of fitness, one can tailor the algorithm to design molecules for a wide variety of applications. For example, one could use the algorithm to design rigid molecules that tend to align themselves parallel to each other and have large dipole moments along their long axis, for application in optical or electrical materials. Or one could design molecules that associate with the active site of a protein to form a low energy complex. In the example discussed below, the molecule is submitted to the cluster energy calculation. The member fitness $f_i$ is then calculated from the member's cluster energy $E_i$ according to

$$f_i = \frac{1}{E_i - E_{\text{best}} + 1},$$

where $E_{\text{best}}$ is the cluster energy of the most fit population member.

Originally, a simple linear fitness scaling method was used, which scaled the best member's fitness to $x$, and the average member's fitness to 1. Any fitnesses that scaled to less than zero were set to zero. Under this scheme, however, a large number of strings often received a zero fitness after scaling. In order to preserve population diversity and stave off premature convergence of the GA, a dual linear fitness scaling operator was constructed to scale the population fitness and control the selection pressure of the GA. Members of the population with above average or average fitness are scaled according to

$$f_i = f_{\text{ave}} + f_{\text{ave}} \left( \frac{F_i f_{\text{ave}}}{f_{\text{best}} - f_{\text{ave}}} \right) (x - 1),$$

Fig. 5. The library of available chemical building blocks.
where $f$ is the individual, average, or population best fitness and $x$ is the fitness scaling constant. Population members having a fitness of less than the average fitness have their fitness scaled according to

$$f_i = f_{ave} + f_{ave} \left( \frac{F_f ave}{f_{best} - f_{worst}} \right) (x - 1),$$

where $f$ is the individual, average, or population worst fitness, and $x$ is the fitness scaling constant. The effect of the scaling is that the most fit population member is $x$ times more likely than average to be selected as a parent string for the next generation. The average string's fitness is unchanged, and only the worst member of the population has 0 fitness, and cannot be selected as a parent. All other strings have their fitness linearly scaled in proportion to the difference between their fitness and the average fitness, modified by the difference between the best or worst fitness and the average, depending on whether the string is more fit or less fit than average.

Members are selected from the population using fitness-proportional selection and passed to a multipoint crossover and a mutation operator. The mutation operator can change any building block into any other building block, by changing the block’s id from its current integral value to any other valid integer. For a typical run of this algorithm, one might run 10 generations with a population of 25 members, taking roughly 30 hours on the SGI Indigo² XZ workstation.

There are a number of user selectable parameters when the GA is run. First, the monomers to be used in molecular construction must be selected from three lists, left-end groups, internal groups, and right-end groups. The population size and the number of generations to run the GA must then be entered, and one must choose the number of internal building blocks to be included in each molecule. Finally, one sets the crossover probability, number of crossover points per mating, mutation probability, and fitness scaling constant.

The algorithm was tested by running it with a limited set of building blocks, which were suggested by a molecule that has been observed to be a very good self-assembler [4,5].

Fig. 6 shows the set of monomers used by the GA in this test, which was run for 10 generations with a population of 30 members. After running several test runs and observing convergence behavior while varying the GA parameters, the mutation probability was set to 0.1, the crossover probability 0.7 at two sites, and the fitness scaling constant was set to 2. Fig. 6 also shows the five best molecules created by the GA, and interestingly the fifth one is extremely similar to the molecule that inspired the choice of building blocks. The other four molecules represent interesting possibilities for use in the design of future molecules for self-assembly. Despite being restricted to relatively few generations, and being forced to employ a small population, the GA was able to both find a known molecule of high fitness and to suggest interesting new molecules for investigation. Further investigation of the molecules found by the algorithm is required in order to assess its usefulness for the above application, but it is clear that the search algorithm has no trouble arriving at a wide variety of good solutions to the problem posed.

One limitation of the current algorithm is that only one of the many conceivable modes of packing of the molecule into a cluster is examined. It is possible that a molecule of extremely low fitness could pack very well into some other structure, causing us to neglect a very good self-assembling molecule. Perhaps future versions of this algorithm could examine instead a given molecule in multiple clusters, assigning fitness based on the average or best energy found.

While the genetic algorithm currently uses the predicted energy of a specific arrangement of several identical copies of a molecule to calculate fitness, there are many possible alternative definitions of fitness. One could calculate molecular or cluster estimated dipole moments, cluster volumes, densities, surface areas, surface charge distributions, or a myriad of molecular parameters such as weight, length, aspect ratio, or radius of gyration. The genetic algorithm could be used to generate molecules suited to any purpose that can be defined by a combination of properties such as those mentioned above. One possibility,
Designing Molecules that form Low Energy Clusters

4. Optimal configuration of a small cluster of molecules

After considering the limitations imposed on the previous algorithm by its examination of only one of the possible modes of molecular clustering, we became interested in the feasibility of searching for the optimal spatial arrangement of several molecules. The electrical, optical, and mechanical properties of bulk materials are heavily influenced by the mode in which the constituent molecules are arranged. Materials for piezoelectric, non-linear optical, adhesive, sensor, and coating applications could have properties far surpassing many of today's materials if designers could encode a preference for a desired assembly mode into a molecule's structure. The development of a tool to predict the manner in which copies of a given molecule will assemble would have at least two important implications. First, the explanation of observed properties of a material is made easier by knowledge of the molecular-level structure of that material. When it is not possible to experimentally determine the nature of molecular packing in a sample, a computer prediction can be very useful. Second, if the behavior of a system can be predicted in advance of its synthesis it becomes possible to develop design procedures, such as the one described in the previous section, that would automatically develop molecules specifically suited to their intended application.

The number of possible arrangements of N molecules in space scales as $6^{N-1}$, making the search for optimally arranged clusters of even a small number of molecules rather computationally expensive. A genetic algorithm was designed to search the different possible packing modes of a four-molecule cluster for those having the lowest energy, according to the Tripos force field. A unique molecular cluster is defined using the variables shown in Fig. 7. The four molecules' distance from the origin, z-translation, z-rotation, and orientation are encoded into a binary string using a standard multiparameter, mapped, fixed point coding [15]. Each population member is thus a binary string representing these four variable settings for each of the four molecules in the cluster. The fitness of a population member is calculated by placing the relevant to some of our earlier work [3], would be to define fitness as the energy of two antiparallel molecules minus the energy of two parallel molecules. This definition of fitness would result in a GA that searches for molecules that could be likely to assemble into supramolecular structures with polar order, and therefore having potentially exciting electrical or optical properties.
four molecules in the positions described by the string’s parameters, relaxing the geometry for a user defined number of conjugate gradient minimization steps, and computing the energy with the Tripos force field. The population members’ fitness is calculated from the cluster energy according to the formula [16]

$$f_i = e^{-sf_{best}}$$

where $f$ represents the fitness of the current or best member in the population and $s$ is a scaling factor between zero and one.

Because of the strength of the interatomic repulsive term in the Tripos force field, any cluster where molecules are placed too closely together will have an extremely high energy. A randomly generated starting population will therefore tend to have many members with fitness very close to zero, and a relatively few members that are very highly fit. In order to provide a better starting point for the GA, the initial population can be generated as follows. The user sets the size of the population and a maximum energy threshold for acceptance into the population. Then, random binary strings are created, translated, and evaluated until a full starting population of members having energy less than the threshold energy is collected. Setting the threshold at 100 000 kcal/mol will cause a 75% rejection rate of random strings, while setting a 10 000 kcal/mol threshold causes an 88% rejection rate. Creating a new population when there is a 90% rejection rate takes an amount of time roughly equivalent to running the GA for 10 generations, and is therefore only useful if the GA converges 10 generations sooner, or to a better final solution. Fig. 8 shows the difference between using an elite and a random starting population. Surprisingly, the GA that used the exceedingly unfit random starting population outperforms the GA that was given the “head start” of beginning with a more fit population. Random starting populations were used for all subsequent runs of the GA.

The GA was developed as follows. First, the dual linear fitness scaling procedure described above was implemented. Then, members were chosen as parents using fitness proportional selection with replacement. Test runs showed the best convergence when the fitness scaling was performed such that the most highly fit member of the population was assigned a fitness of three times the population average fitness. The combination of linear fitness scaling and fitness proportional selection allows the user to control the selection pressure in the GA. Tournament selection offers an even finer degree of control over selection pressure, and
could be used as an alternative to the current scheme if desired. String mating is done by standard multi-
point crossover and pointwise mutation. The best convergence in test runs using strings of 64 bits was 
observed with 2-point crossover and a mutation rate of 0.02. A typical run, of 25 generations on a pop-
ulation of 75 members takes roughly 8 hours to complete on our workstation.

There are a number of options the user must specify when running the GA. For the mapping of each 
molecule's distance from the origin and z-displacement, the user can choose the minimum and maximum 
real values, as well as how many bits are used to represent these values. This allows control of the resolution 
of the model, and the length of the binary strings in the population. The z-rotation angle can be free, with a 
user-adjustable number of bits used in its representation, or it can be forced to be a member of one of the 
following sets: \{0° 180°\}, \{0° 120° 240°\}, \{0° 90° 180° 270°\}, \{0° 60° 120° 180° 240° 300°\}, or \{0° 45° 90° 
135° 180° 225° 270° 315°\} if only certain specific rotation states are of interest. Molecular orientation re-
quires 1 bit to specify upward or downward orientation. The user can disable the molecular orientation bit, 
forcing all molecules to lie parallel to each other, and can disable z-translation forcing all molecules to lie in 
registry with each other in the z direction. The user must also specify whether or not to perform several 
steps of conjugate gradient minimization to relax each cluster as it is evaluated. Finally, the user must 
determine the population size, number of generations to run, fitness scaling constant, mutation probability, 
number of crossover sites per mating, and crossover probability.

As the first test of this algorithm, it was used to examine the clustering behavior of a molecule previously 
studied experimentally. The molecule is a phenol terminated pentamer of phenylene vinylene [17].

The above molecule was coupled to a flexible segment composed of an oligobutadiene oligostyrene 
diblock and a thin film was cast from solution. The material's packing was then investigated using electron 
beam diffraction, which revealed that the molecules pack in the mode illustrated on the left side of Fig. 9. 
When relaxed using conjugate gradient descent, the assembly shown has an energy of 19.4 kcal/mol in 
Sybyl. The genetic algorithm was run with disabled z-translation and orientation, for 25 generations on a 
population of 75 members. Gradient relaxation of the clusters was turned off. The mutation probability was 
set to 0.15 and there were three crossover points with a 0.5 crossover probability. Though the GA did not 
arrive at the experimentally observed structure, it found the arrangement shown on the right of Fig. 9 with 
an energy of –25 kcal/mol. Even though the GA did not find the observed structure, the test should be 
considered successful because it found a structure to which the force field assigns a lower energy. This could 
be because this algorithm is only examining four molecules in space, and the experimentally observed 
structure is formed as a result of many-molecule interactions. It is also possible that the force field is in-
correctly calculating the energies for edge to face versus face to face packing of the molecules.
In the second test of the algorithm, the GA was used to compare the packing of two of the rod-like molecular fragments recently used in the construction of rodcoil molecules, when both $2$ and $3$ are attached to identical coil segments and observed using crossed polar optical microscopy, both exhibit very similar birefringent phases over nearly the same temperature range. Both molecules seem to have roughly the same tendency to order. When both are submitted to the genetic algorithm, a cluster of four molecules of $2$ is found with an energy of $-11$ kcal/mol, while a cluster of four molecules $3$ is found with an energy of $-7$ kcal/mol, indicating that the two molecules have roughly the same tendency to order, in agreement with the experimental observations.

The particular mode of placing molecules in a cluster used by this program can generate a wide variety of different arrangements of molecules in space. However, the placement of four molecules at the orthogonal positions $(+x_1, 0), (-x_2, 0), (0, y_1)$, and $(0, -y_2)$, as opposed to another arrangement, such as the triangular positions $(0, d_1), (d_2 \cos 30, -d_2 \sin 30), \text{ and } (-d_3 \cos 30, -d_3 \sin 30)$, could bias the search toward certain arrangements at the expense of others. We are currently investigating the triangular placement in order to compare the nature of the clusters found to those found by the orthogonal placement for the same molecules.

5. Conclusions

The development and application of two new genetic algorithms has been presented. The first algorithm builds molecules from a set of chemical building blocks to satisfy a user-specified criterion. The primary...
fitness criterion used here is the energy of a cluster of aligned copies of the molecule. The algorithm was tested on a set of building blocks that make up a molecule known to be a good self- assembler. Interestingly, one of the best molecules found by the GA in this test run was nearly identical to the molecule we had studied. The GA also identified several other molecules built from the same building blocks, predicting that they should show similar ordering tendencies.

While the genetic algorithm currently uses the predicted energy of a specific arrangement of several identical copies of a molecule to calculate fitness, there are many possible alternative definitions of fitness. One could calculate molecular or cluster estimated dipole moments, cluster volumes, densities, surface areas, surface charge distributions, or a myriad of molecular parameters such as weight, length, aspect ratio, or radius of gyration. There also exist algorithms to predict the toxicity and biodegradability of a molecule based on its structure. Another interesting extension of this work would be to add to the fitness function some measure of synthetic difficulty, or a prediction of the effort required to synthesize the molecule. The algorithm could then be tuned to search for molecules with properties identical to molecules already known to be useful, but which are easier to manufacture. The genetic algorithm could be used to generate molecules suited to any purpose that can be defined by a combination of properties such as those mentioned above.

The second algorithm searches for the best possible arrangement of four molecules in space, as determined by an empirical force field. The method was tested on a phenylene vinylene oligomer and found a cluster structure with a lower energy than the experimentally observed one. When tested on two molecules that displayed similar ordering tendencies, the GA produced clusters of similar energy. When tested against a known structure, the GA found a structure with lower energy than the observed one, according to the force field evaluation. Given sufficient computer time, the genetic algorithm is able to find a very low energy arrangement of a cluster built from copies of a given molecule, though the algorithm is not restricted to work with only one chemical species. With a simple modification it could be made to construct clusters from any four molecules. The algorithm provides useful insight into how several given copies of a molecule might tend to arrange themselves into small clusters, and ultimately into larger structures. These insights can help to explain experimental observations of such properties as nonlinear optical behavior, nonisotropic mechanical behavior, finitely sized nanoaggregates, as well as a host of other observations. The method is therefore useful in development of model packing structures for existing molecules, but could also be used to examine potential new synthetic target molecules to determine the likelihood of their being able to pack into structures at the nanoscale that are commensurate with some set of desired material properties at the macroscopic level. A large set of proposed synthetic targets can be quickly screened for promising candidates, saving considerable research time and synthetic effort. These two genetic algorithms promise to aid in the search for new molecules that self-assemble into designed nanostructures, offering a toolbox for designed materials with novel properties.

Acknowledgements

We would like to thank the National Science Foundation for the support of this work through grant DMR-9312601.

References