

Evolutionary Design of Molecules with Desired Properties Using the Genetic Algorithm

Venkat Venkatasubramanian,* King Chan, and James M. Caruthers

Laboratory for Intelligent Process Systems, School of Chemical Engineering, Purdue University,
West Lafayette, Indiana 47907

Received June 21, 1994[®]

Designing new molecules possessing desired properties is an important and difficult problem in the chemical, material, and pharmaceutical industries. The standard approach to this problem consists of an iterative formulation, synthesis, and evaluation cycle that is long, time-consuming, and expensive. Current computer-aided design approaches include heuristic and exhaustive searches, mathematical programming, and knowledge-based systems methods. While all these methods have a certain degree of appeal, they suffer from drawbacks in handling combinatorially large, nonlinear search spaces. Recently, a genetic algorithm-based approach was shown to be quite promising in handling these difficulties. In this paper, we investigate the performance of the basic genetic design framework for larger search spaces. We also present an extension to the basic genetic design framework by incorporating higher-level chemical knowledge to handle constraints such as chemical feasibility, stability, and complexity better. These advances are demonstrated with the aid of a polymer design case study.

INTRODUCTION

The problem of designing new molecules with desired properties is an important and difficult one, encompassing the design of polymers, polymeric composites, blends, paints and varnishes, refrigerants, solvents, drugs, pesticides, and so on. The traditional approaches to this design involve a laborious and expensive trial-and-error procedure. For example, it may take over 1000 design, synthesis, and evaluation cycles before a new drug is designed.¹ Hence, there is considerable incentive in developing computer-assisted approaches toward the automation of molecular design.

In general, computer-aided molecular design requires the solution of two problems: the *forward* problem, which requires the computation of physical, chemical, and biological properties from the molecular structure, and the *inverse* problem, which requires the identification of the appropriate molecular structure given the desired physicochemical properties. While there has been extensive work toward the solution of the forward problem (such as group contribution methods, equation of state approaches, quantitative structure–activity relationships (QSAR), and so on), relatively less attention has been paid for the inverse problem.

Past approaches to the inverse problem can be divided into five categories—random search,¹ heuristic enumeration,^{2–4} mathematical programming,^{5,6} knowledge-based systems,^{7,8} and graphical reconstruction^{9,10} methods. While all these methods have some appeal, they suffer from drawbacks due to combinatorial complexity of the search space, design knowledge acquisition difficulties, nonlinear structure–property correlations, and problems in incorporating higher-level chemical and biological knowledge. The combinatorial complexity associated with CAMD makes random, exhaustive, or heuristic enumeration procedures less effective for large-scale molecular design problems. Mathematical programming methods consider molecular design as an optimization problem where the objective is to minimize the error

between the desired target values and the values attained by the current design. The solutions to mixed integer nonlinear programming (MINLP) formulations are susceptible to local minima traps for problems with nonlinear constraints, as is the case with most structure–property relations. The solutions to these problems are also computationally very expensive, especially for highly nonlinear systems. The knowledge-based systems approach assumes that expert rules exist for manipulating chemical structures to achieve the desired physical properties. However, many nonlinear structure–property relationships cannot be easily expressed as rules, especially when designing for multiple design objectives. Furthermore, extraction of such design expertise from experts on molecular design is not easy, making the knowledge acquisition problem a very difficult one. Lastly, graph reconstruction methods solves the inverse problem of reconstructing the molecular graph(s) of a topological index. Topological indices for structure–property correlations are very common in environmental toxicology studies.^{11,12} Thus, given the desired property value, the topological indices and then molecular graph(s) can be calculated from a structure–property relationship equation. For molecular design using this approach, one must express all structure–property relations in terms of topological indices. This may not be appropriate or feasible for all properties. Furthermore, topological indices are not unique, and therefore currently there does not exist a general graph reconstruction method for all molecular indices. In addition, since one often deals with a number of design criteria to be satisfied and not just one or two properties, this approach may not be feasible in general.

Recently, Venkatasubramanian et al.^{13,14} proposed an evolutionary molecular design approach using genetic algorithms (GA) to address the drawbacks of the other CAMD methods. Genetic algorithms are general purpose, stochastic, evolutionary search, and optimization strategies based on the Darwinian model of natural selection and evolution. The essence of a genetic algorithm lies in allowing a dynamically evolving population of molecules to gradually improve by competing for the best performance. In that study, the

[®] Abstract published in *Advance ACS Abstracts*, December 1, 1994.

mechanics, characteristics, and viability of using genetic algorithm for molecular design were fully elucidated and demonstrated for relatively small molecular design case studies. The study showed that the genetic design (GD) approach was able to locate optimal designs for many target molecules with multiple specifications. It was also able to discover a diverse population of near-optimal designs.

In the present study, we pursue two research themes within the genetic design framework. One is to investigate the efficacy of genetic design for problems with much larger and more complex design spaces. The second theme is to extend the original genetic design framework by incorporating higher-level chemical knowledge to better handle constraints such as chemical feasibility, stability, and molecular complexity. The remainder of the paper is organized in the following manner. First, a brief overview of genetic algorithms and their adaptation to molecular design is presented. Next, results are presented for the large-scale polymer design case study for the basic framework as well as for the knowledge augmented genetic design framework. The paper concludes with a summary and some thoughts on future directions.

CAMD USING GENETIC ALGORITHMS

Pioneered by Holland,¹⁵ genetic algorithms are based on the Darwinian principles of natural selection and evolution. They work with a simulated population which represents potential solutions to the given problem. Each population member is assigned a "fitness" according to how well it satisfies the solution requirements. The highly fit individuals or "parents" are given a greater chance to "reproduce" offsprings. The least fit members are less likely to get selected for reproduction and thus "die" eventually. By stochastically favoring the mating of more fit population members, the most promising areas of the search space are explored at the expense of low performance regions. There are numerous methods for allocating population members for reproduction.^{16,17} The most common form of selection uses a technique known as "fitness proportionate selection". This selection is random but is weighted in proportion to the normalized population fitness values.

As in natural selection, during reproduction good features are spread throughout the population by mixing and exchanging other good characteristics in the population. Two conventional genetic algorithm mechanisms for reproduction are one-point crossover and mutation. These operations are called "recombination operators" or "genetic operators". In one-point crossover, two mating parents are each cut once at randomly selected points, and the sections after the cuts are exchanged. In mutation, each gene of the parent has a small probability of being altered. Thus, crossover facilitates the large-scale exploration of the search space, while mutation explores at a smaller scale. This completes the "life cycle" of the population, which leads to replacement of the current population. The next cycle begins by returning to the fitness evaluation procedure. The termination criteria for a genetic search can be that a solution was found, a fixed number of generations was completed, or if the population fitness converged to stable value.

To implement a genetic algorithm one needs to specify, a suitable representation or code for the problem, a fitness

function which assigns a merit to each coded solution, a parent selection procedure which favors more fit individuals, genetic operators for reproduction, and a stopping criteria. Detailed discussions on genetic algorithms fundamentals and applications can be found in Goldberg,¹⁸ Davis,¹⁹ Rawlins,²⁰ and Michalewicz.²¹ A summary of genetic algorithm applications in chemistry can be found in Hibbert²² and Lucasius.²³ Since the adaptation of traditional genetic algorithms for molecular design has been described in detail by Venkatasubramanian et al.,¹⁴ only a short summary is provided in this paper.

Coding. In the majority of applications, the candidate solutions are encoded as bit strings {0 1}. This choice is due in part to Holland who used this alphabet in his theoretical work on genetic algorithms. In our framework, an alphabet of symbols that represent chemical building blocks called base groups {e.g., >C<, -H, -O-, >C=O, etc.} was developed as it is more appropriate for representing complex molecules. Such a symbolic representation also facilitates the incorporation of higher-level chemical knowledge more readily. The base groups can also be chemical substructures or monomer units.

Fitness Function. To evaluate how well a candidate molecule satisfies the desired target properties and constraints, one needs a fitness function that returns a single numerical "fitness" or "measure of merit". The genetic framework searches for design candidates by evaluating how close (or far) a given candidate molecule's macroscopic properties are from the desired target. Thus, a measure of fitness one develops should take as its input the desired target properties, the candidate's properties, the desired level of tolerance in meeting the target specifications, and some tuning parameters. The output from this fitness function should range smoothly from 0 to 1 such that candidates whose properties are far from the target have a fitness close to 0 (and are thus penalized) and those candidates closer to the target have a fitness closer to 1 (and are thus encouraged). The tuning parameter is needed to adjust how strongly one would like the function to penalize (or encourage) deviations from the target. In addition, one would like this function to be as simple as possible. A Gaussian-like function satisfies these requirements and was used in our framework and is given below:

$$\text{fitness}(\bar{x}) = \exp\left(-\alpha \left[\frac{\sum_{i=1}^n (P_i - \bar{P})^2}{\sum_{i=1}^n (P_{i,\max} - P_{i,\min})^2} \right]\right) \quad (1)$$

where P_i is the i th property value, \bar{P} is the average of the maximum and minimum acceptable property values $P_{i,\max}$ and $P_{i,\min}$, respectively, and α is the fitness decay factor. The parameter, α , controls the relative weight of the best solution compared to that of the average. The fitness function value ranges from 0 to 1.

Constraints. An important problem in molecular design is handling various constraints. The property constraints can be handled directly by the fitness function. However, other constraints, such as maximum molecular length, molecular stability, and complexity are better handled by "penalty" methods in our framework. In this method, a penalty is assigned to the overall fitness for design candidates which

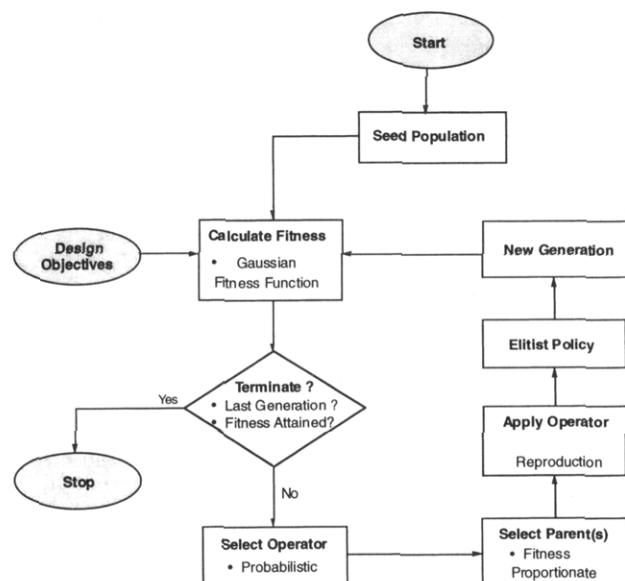


Figure 1. Genetic design flowchart.

violates the defined constraints. This is expressed as

$$\text{fitness}_{\text{total}}(\bar{x}) = \text{fitness}(\bar{x}) + \epsilon \delta \sum_{i=1}^P \phi_i \quad (2)$$

where P is the total number of constraints, δ is an penalty coefficient, ϵ is -1 for maximization and $+1$ for minimization problems, and ϕ_i is a penalty related to the i th constraint. The application of these methods is further elaborated in the design case study.

Operators. The fitness proportionate selection scheme which favors more fit molecular designs was chosen. Having selected the parents, the genetic operators must be defined to facilitate chemical manipulations. In addition to the classical one-point crossover and mutation operations other recombination operators were developed to better express the complex chemistry and interactions between molecular groups. Thus, the operators are as follows.

- *One-Point Crossover* is similar to that of the classical version except that the cutting position is randomly chosen for both parents.

- *Two-Point Crossover* is like one-point crossover, except that two cut points rather than one are selected at random and the groups are swapped between the two points.

- *Blend* produces one offspring from the end-to-end connection of two parents. This essentially combines the attributes from both parents.







- *Mainchain and Side-Chain Mutations* are completely analogous to bit mutations. The mutation operators randomly replaces groups along the main-chain or side-chain positions, respectively.

- *Insertion and Deletion* randomly places or remove groups at main-chain and side-chain positions, respectively.

- *Hop* randomly selects a group in the main chain of the molecule and makes it "hop" to a randomly selected location on the main chain.

Execution. The genetic design execution loop, shown in Figure 1, can be described as steady state reproduction with elitism. The former maintains a constant-sized population, and the latter saves the best designs from the previous generation, respectively. The genetic algorithm is initiated randomly with chemically feasible designs. Such a random

Table 1. Polymer Design Study Base Groups

Mainchain Groups	Sidechain Groups
$>C<$ $-S-$ $-SO_2-$ $-O-$ $-\overset{O}{\parallel}C-$ $-\overset{O}{\parallel}C-$ $-\overset{O}{\parallel}C-O-$ $-\overset{O}{\parallel}C-O-$ $-NH-$ $-\overset{O}{\parallel}C-NH-$ $-\overset{O}{\parallel}C-NH-$ $-NH-\overset{O}{\parallel}C-NH-$     	$-H$ $-CH_3$ $-C_2H_5$ $-nC_3H_7$ $-iC_3H_7$ $-tC_4H_9$ $-F$ $-Cl$ $-Br$ $-OH$ $-OCH_3$ $-\overset{O}{\parallel}C-CH_3$ $-\overset{O}{\parallel}C-OCH_3$  $-CN$

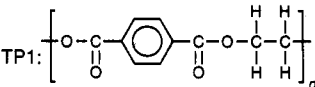
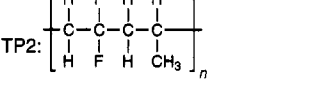
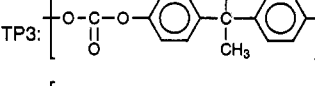
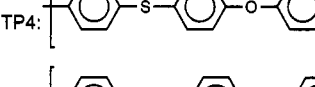
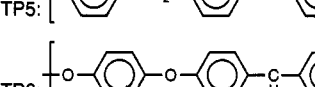
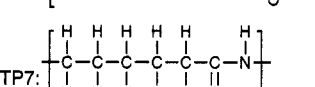
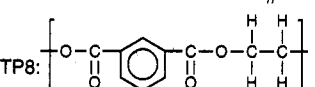
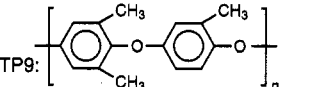
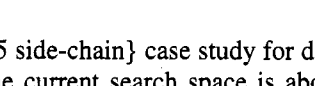
initiation is called "cold starting" the population. Instead, if one has better ideas about the target molecular structure one may start with a population of molecules that reflect such educated guesses. This is referred to as "warm starting" the population. In all the cases examined in this paper, only cold starting was considered as this tests the genetic design approach for the worst case situation. To maintain solution diversity, the elitist policy saves only chemically distinct designs from the previous generation. The genetic algorithm requires values for a number of tunable parameters, such as population size, genetic operator probability rates, and fitness function constants. A list of these parameters and their values is provided in the design case study.

POLYMER DESIGN CASE STUDY

As noted earlier, we investigated the efficiency of genetic design for problems with much larger and more complex design spaces in this study. We also extended the original genetic algorithmic framework by incorporating higher-level chemical knowledge to better handle constraints such as chemical stability and molecular complexity. We pursued these two themes with the aid of a polymer design case study, similar to the one that was considered by Venkatasubramanian et al.¹⁴ The previous investigation examined design problems with four main-chain ($>C<$, $-C_6H_4-$, $-C=O-$, $-O-$) and four side-chain groups ($-H$, $-CH_3$, $-F$, $-Cl$). As a test, the genetic algorithm was required to design molecular structures given the property values of known polymers, namely (i) polyethylene terephthalate (PET), (ii) polyvinylidene propylene copolymer (PVP), and (iii) polycarbonate of bisphenol-A (PC). The genetic design procedure discovered all three target polymers in a fraction of the 200 total generations allowed for all design lengths ($L = 2-7$ and $L = 2-10$) and for all initial population conditions (random main-chain and side-chain groups and $-CH_2-$ only). For example, the average earliest generation for locating the design target and the success rate (in parentheses) for a population initiated with random main-chain and side-chain groups having lengths 2-7 are (i) PET, 11.3 generations (100%), (ii) PVP, 28.2 generations (100%), and (ii) PC, 41.0 generations (100%). The GA was also able to determine many high-fitness alternate structures. However, some of the near-optimal designs contained unstable main-chain structures.

In this work, to investigate the efficacy of the genetic design for much larger and more complex search spaces, the base group choices were increased to 17 main-chain and 15 side-chain groups as shown in Table 1. This increases the total number of design candidates from about 1.4×10^5 candidates for the {four main-chain, four side-chain} case study to about 1.1×10^{13} candidates for the {17 main-chain,

Table 2. Molecular Design Targets and Their Properties

target polymer	density ρ , gm / cm ³	glass transition temperature T_g , K	thermal expansion coeff α , K ⁻¹	specific heat capacity Cp, J / Kg·K	bulk modulus K, n / m ²
TP1: 	1.34	350.75	2.96×10^{-4}	1152.67	5.18×10^9
TP2: 	1.18	225.24	2.81×10^{-4}	1377.82	2.51×10^9
TP3: 	1.21	420.83	2.90×10^{-4}	1135.10	5.40×10^9
TP4: 	1.19	406.83	2.90×10^{-4}	1073.96	5.39×10^9
TP5: 	1.28	472.00	2.89×10^{-4}	995.95	5.31×10^9
TP6: 	1.25	421.12	2.90×10^{-4}	1016.55	6.12×10^9
TP7: 	1.06	322.55	2.98×10^{-4}	1455.90	3.85×10^9
TP8: 	1.27	322.12	2.81×10^{-4}	1152.67	3.42×10^9
TP9: 	1.09	428.73	2.77×10^{-4}	1163.10	4.15×10^9

15 side-chain} case study for design lengths of 2–7. Thus, the current search space is about 100 million times larger than the one in our earlier study. We also increased the number of target polymers from three in the previous study to nine as shown in Table 2.

The number of property constraints are the same as before at five, and they are density, glass transition temperature, thermal expansion coefficient, specific heat capacity, and bulk modulus. Physical property values were calculated by the van Krevelen²⁴ group contribution methods. The search space is further complicated by the increased number of nonlinear group interactions. For example, for polymer design target no. 4 ($-\text{C}_6\text{H}_4\text{SC}_6\text{H}_4\text{OC}_6\text{H}_4\text{C}(\text{CH}_3)_2\text{C}_6\text{H}_4\text{O}-$) the nonlinear van Krevelen group interactions require that every main-chain group, other than the $-\text{O}-$ endgroup, and every side-chain group be in their proper position in order to give the optimal fitness of 1. That is the macroscopic properties depend not only on the group types but also on the *exact* ordering of them in the molecule.

The current study also examines the possibility of incorporating higher-level chemical knowledge to address constraints such as chemical stability and molecular complexity. The incorporation of chemical knowledge within the genetic algorithm framework is discussed in the next two sections. We then present results comparing the performance of the standard or the basic genetic design approach with that of the knowledge-augmented genetic algorithm utilizing higher-level chemical knowledge.

Incorporating Chemical Knowledge. Molecular Stability. In this study, higher-level chemical knowledge is

incorporated to facilitate the genetic design to search toward more chemically realistic and stable polymers. For example, some common group combinations which leads to chemically unstable structures are $-\text{O}-\text{O}-\text{O}-$ and $-\text{OC}=\text{O}-\text{C}=\text{O}-$. These group combinations were often found in many high-fitness polymers in our earlier studies. This was so because in our earlier approach such higher-level knowledge was not included and hence many unrealistic group combinations were allowed by the genetic design algorithm. In the knowledge-augmented GA framework, such unstable main-chain group combinations are given a zero fitness and are not considered further in the design process. The knowledge about the stability of nearest neighbor main-chain groups was obtained from Barton and Ollis.²⁵

Another example of such a constraint is environmental acceptability. Certain molecular groups or group combinations are known to be environmentally toxic or unacceptable. This is a common problem in the design of agrochemicals such as fertilizers and pesticides as well as refrigerants. Yet another example would be the relative ease or difficulty involved in the synthesis and the manufacture of the proposed design candidate. It is important to be able to incorporate such constraints in the design process. In this work, however, only stability and molecular complexity constraints have been considered.

Molecular Complexity. Molecular complexity is encoded as a count of the total number of main-chain and side-chain groups and is given by the following equations

$$\text{fitness}(\bar{x}) = \text{fitness}(\bar{x}) - \beta \cdot \text{sig.complexity} \quad (3)$$

Table 3. Genetic Design Parameters for the Polymer Design Case Study

GA parameters		genetic operator probabilities	
population size	100	crossover	0.20
no. of generations	1000	main-chain mutation	0.20
max. design length	target length + 2	side-chain mutation	0.20
fitness gain	$\alpha = 0.001$	insertion	0.00
complexed penalty	$\gamma = 100$	deletion	0.10
complexity sigmoid gain	$\beta = 0.10$	blending	0.10
total runs per case study	25	hop	0.20
elitist policy keep best	10%		

$$\text{sig} = \frac{2}{(1 + \exp[-\gamma(F - F_{\text{crit}})])} \quad (4)$$

$$\text{complexity} = \frac{\text{MC} + \text{SC}}{\text{MC}_{\text{max}} + \text{SC}_{\text{max}}} \quad (5)$$

where β is a penalty scaling factor, sig is a sigmoidal fitness function (eq 4) which provides a fitness threshold, F_{crit} , for the genetic algorithm to start penalizing complex designs, and γ is a decay scaling parameter. The complexity measure (eq 5) ranges from 0 to 1 and is given by the ratio of the number of main-chain (MC) and side-chain (SC) units in the current design to the maximum allowable main-chain and side-chain units (32 in this case). Thus, the complexity measure is applied only after a design exceeds a fitness threshold.

Genetic Design Parameters and Simulations. The genetic design parameter values used in this study are the same as in our earlier study and are listed in Table 3. The design lengths varied from two base group units to a maximum of two units more than the polymer design target ($L_{\text{max}} = 6-10$). For statistical significance, results were compiled after 25 different genetic design runs of 1000 generations each. The genetic design investigations carried out for the polymer design case study are subdivided as follows: (i) standard or basic genetic design approach (ii) knowledge-augmented genetic design approach which penalizes unstable main-chain combinations, and (iii) knowledge-augmented genetic design approach which penalizes unstable main-chain group combinations and molecular complexity. The fitness function gain, α , in eq 5 is equal to 0.001. The parameters for eqs 3, 4, and 5 which penalize complex solutions are as follows: $F_{\text{crit}} = 0.99$, which results in applying the complexity measure only after near optimal solutions are attained, $\gamma = 100$, which provides a gradual activation of the complexity measure as the fitness approaches the critical value, and $\beta = 0.10$, since a large penalty reduces the overall design fitness to a point where the genetic design algorithm considers the design to be unworthy of further consideration.

RESULTS AND DISCUSSION

The results for the different genetic design cases are presented in Table 4. Part (a) gives the percent success rate (in **bold text**) in achieving the design objective and the number of successful runs in (parentheses). Part (b) gives the average generation when the target was first located (in normal text). Part (c) gives the average number of distinct high-fitness solutions at the end of the genetic design (in *italic next*).

As one might have expected, the genetic design was not as successful as it was in our earlier smaller case study, when

it located the target molecule in every run (i.e., a success rate of 100%). However, the most important observation here is that the genetic design still succeeded in finding the target molecule in eight out of the nine cases, even though the search space had exploded by over a factor of 100 million. As seen from part (a) of Table 4, with the exception of target polymer no. 4, all target polymers were located at least once by one of the genetic design types (i.e., columns 3-7).

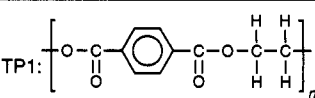
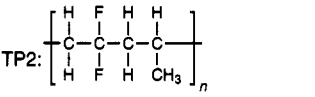
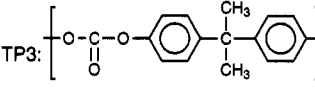
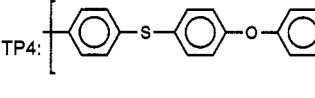
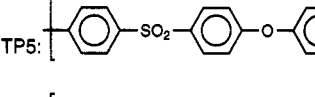
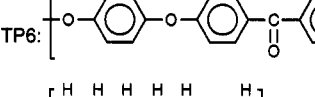
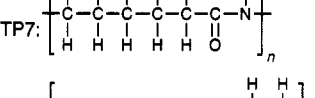
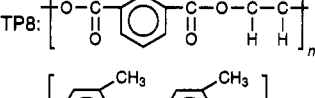
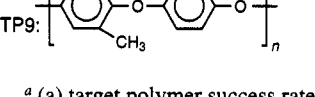
From part (b) in Table 4, we see that some molecules take longer than others to be discovered. For example, the target polymer no. 7 was always found in less than 100 generations. On the other hand, target polymer no. 6 was located with varying degree of success (4-68%) and took more than 400 generations for discovery. Typically, longer molecules which require exact main-chain group ordering and side-chain positioning needed more generations to be discovered. This explains why target polymer no. 7 ($-(\text{CH}_2)_5\text{C}=\text{ONH}-$), which is the only target molecule with no group ordering constraint, was quickly located, while target polymer no. 6, $(-\text{OC}_6\text{H}_4\text{OC}_6\text{H}_4\text{COC}_6\text{H}_4-)$ which requires exact ordering, took much longer to discover. The exact ordering requirement and the long backbone structure are also the reasons why the target polymer no. 4 was never discovered in any of the runs of 1000 generations each.

Table 4, from columns five to seven, presents results for the knowledge-augmented genetic design where the higher-level chemical knowledge about the feasibility and stability of group combinations and molecular complexity were incorporated. From these results one may observe certain general trends. For example, we see that the success rates are higher, in general, with the knowledge-augmented genetic design in comparison with the standard genetic design (part (a) of column 3 vs column 5 and 7), when the initial population consisted of random main-chain and side-chain groups. Thus, adding higher-level chemical knowledge seems to improve the design efficiency. For column 7, since the complexity measure was applied only after the fitness threshold is exceeded, more generations were required to achieve the target. This is also the main reason why the genetic design was unable to locate target polymers number 3, 4, and 9. It appears that by incorporating higher-level chemical knowledge one is not only able to produce candidates that are chemically feasible, stable, and less complex but also able to increase the efficiency of the search by eliminating spurious candidates in the genetic design.

The results also suggest that the initial polymer population complexity plays a role in the success rate of the genetic design. For example, the standard genetic design, in general, gives better results when the initial population side chains were seeded with hydrogen groups (column 3, part (a) vs column 4, part (a)). Large improvements are seen for target polymer no. 1 (12-60%) and for target polymer no. 6 (8-68%). Similar results were obtained for the knowledge-augmented genetic design which penalized unstable main-chain structures (column 5 part (a) vs column 6, part (a)). The best improvements being those for target polymer no. 1 (28-60%) and for target polymer no. 6 (4-32%).

One of the most appealing features of genetic design is that it finds many diverse alternative solutions that are very close to the desired property target values. The number of near-optimal or high-fitness solutions is listed in Table 4, part (c). The high-fitness threshold is 0.99 for all design targets except polymer no. 2, which is 0.985. The genetic

Table 4. GA Search Results for the Polymer Design Case Study^a

target polymer	part	standard GA		feasible MC		
		random MC, SC	random MC, hydrogen SC	random MC, SC	random MC, hydrogen SC	random MC, SC
TP1: 	(a)	12% (3)	60% (15)	28% (7)	60% (15)	64% (16)
	(b)	184	300	233	240	428
	(c)	282	192	281	213	166
TP2: 	(a)	36% (9)	48% (12)	40% (10)	48% (12)	48% (12)
	(b)	411	400	209	522	412
	(c)	6	7	7	6	10
TP3: 	(a)	0% (0)	4% (1)	8% (2)	12% (3)	0% (0)
	(b)		293	640	193	
	(c)	163	91	161	74	109
TP4: 	(a)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)
	(b)					
	(c)	861	564	910	589	570
TP5: 	(a)	32% (8)	56% (14)	48% (12)	48% (12)	92% (23)
	(b)	400	205	317	232	420
	(c)	175	136	197	142	99
TP6: 	(a)	8% (2)	68% (17)	4% (1)	32% (8)	16% (4)
	(b)	548	405	529	632	528
	(c)	199	146	314	168	158
TP7: 	(a)	100% (25)	100% (25)	100% (25)	100% (25)	100% (25)
	(b)	61	61	58	64	85
	(c)	217	188	214	198	163
TP8: 	(a)	68% (17)	68% (17)	76% (19)	88% (22)	96% (24)
	(b)	210	88	147	109	81
	(c)	162	132	158	161	125
TP9: 	(a)	8% (2)	4% (1)	4% (1)	4% (1)	0% (0)
	(b)	382	132	513	868	
	(c)	144	69	174	70	46

^a (a) target polymer success rate "**bold**", times target found out of 25 GA Runs "(parentheses)"; (b) average generation number for locating target polymer "plain text"; (c) number of distinct polymers with fitness ≥ 0.99 (≥ 0.985 for TP2) "*italic text*"; MC = main chain, SC = side chain.

design was unable to find alternate solutions with a fitness value greater than 0.99 for this polymer. While the genetic design did not find the global optima for target polymer no. 4, it did locate more than 500–900 alternative near-optimal solutions.

Table 5 presents two of the numerous nearly-optimal alternatives for target polymer no. 4 for the genetic design cases 1–3. As can be seen from the errors listed, the alternative solutions are very close to the target properties and have fitness values exceeding 0.99. The average absolute error ranged from 0.25% to slightly over 1.0% of the desired property values. For example, the near-optimal solution for case 1 with a fitness of 0.995 has the following values for the various properties:

• Density	1.164 gm / cm ³	(-2.2%)
• Glass Transition Temperature	404.80 K	(-0.5%)
• Coefficient of Thermal Expansion	2.91 x 10 ⁻⁴ K ⁻¹	(0.4%)
• Specific Heat Capacity	1078.26 J / Kg. K	(0.4%)
• Bulk Modulus	5.38 x 10 ⁹ N / m ²	(-0.2%)

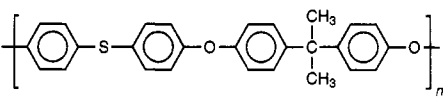
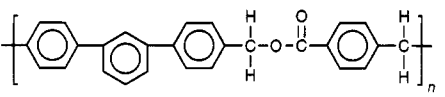
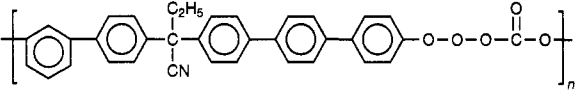
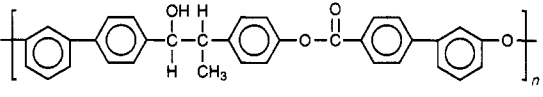
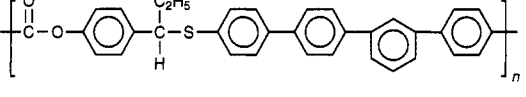
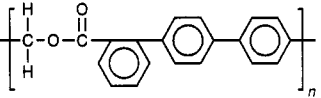
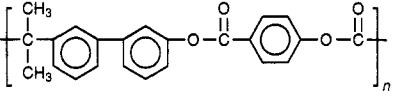
The numbers in parentheses show the relative percent errors. The total absolute error is 0.74%, and the corresponding fitness is 0.995. We see that these properties are very close to those of the target molecule. In many practical situations such a close fit would be considered acceptable, even though one did not locate the exact target.

The solutions varied according to the search type. For example, case 1 (basic genetic design) obtained two infeasible polymers. The first uses a combination of $-O-$ and $>C=O$ groups instead of the single $-OC=O-$ group; and the second contains a $-OOO-$ group combination which is unstable. Using the correct $-OC=O-$ reduces the fitness to 0.976 and increases the average absolute error to 2.04%. Case 2 produces feasible main-chain structures but were generally more complex than those in case 3, which also considers molecular complexity. The number of near-optimal solutions were approximately the same for all genetic design types. Similarly, Table 6 presents near optimal solutions for target polymer no. 3. As in Table 5, all alternative solutions have very high fitness values. These alternative solutions are also seen to be structurally very similar to the target.

CONCLUSIONS

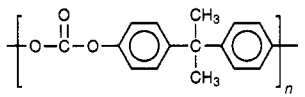
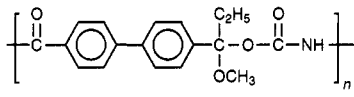
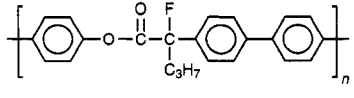
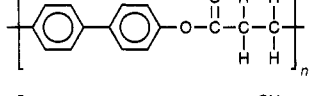
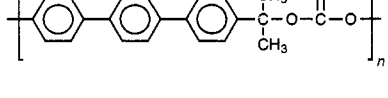
This study investigated the performance of a genetic algorithm based approach for large-scale molecular design. The total candidates of the present case study are about 100 million times larger than that of an earlier study. It was found that, despite the tremendous increase in the search space size and the complex nonlinear group interactions, the genetic design approach was generally able to find the target

Table 5. Near-Optimal Solutions: Target Polymer No. 4

polymer design	% error ^b	fitness
<p>target polymer: TP4</p> 	{0; 0; 0; 0; 0} 0%	1.0
<p>case 1: standard GD^a</p> 	{-2.2; -.5; .4; .4; -.2} 0.74%	0.995
	{1.6; 2.2; -.8; -.2; .9} 1.18%	0.991
<p>case 2: knowledge-augmented GD, stability</p> 	{.04; .09; -.4; .09; .7} 0.25%	0.999
	{-.4; 1.9; .85; .14; -.2.2} 1.10%	0.991
<p>case 3: knowledge-augmented GD, stability & complexity</p> 	{-.1; .6; .1; .08; -.04} 0.21%	0.999
	{.4; -1.0; .02; 1.8; -.9} 0.83%	0.995

^a GD = genetic design. ^b % error is for {density; Tg; thermal expans.; sp. heat; bulk modulus} av absolute error %.

Table 6. Near-Optimal Solutions: Target Polymer No. 3

polymer design	% error ^a	fitness
<p>Target Polymer: TP3</p> 	{0; 0; 0; 0; 0} 0%	1.0
<p>Near-Optimal Solutions</p> 	{.58; .22; .89; -1.3; .09} 0.62%	0.9971
	{-.95; .3; .68; -.4; -1.5} 0.76%	0.9962
	{-.61; .56; 1.2; -.09; 2.1} 0.92%	0.9933
	{-1.9; .34; -.5; -.2; .5} 1.05%	0.9917

^a % Error is for {density; Tg; thermal expans.; sp. heat; bulk modulus} av absolute error %.

molecules, though with a much less success rate and much more slowly compared to our earlier results for smaller search spaces. Furthermore, it was also able to provide a diverse collection of design alternatives which nearly satisfy the property constraints. This would be particularly useful for the design of complex molecules where the forward problem results may not be completely reliable. Hence, one would like a collection of design candidates that are fairly close to

each other to test further with actual synthesis and experimentation in a laboratory.

The original genetic design framework was also extended by incorporating higher-level chemical knowledge so that more realistic, stable, and less complex solutions can be obtained. The knowledge currently incorporated consists of designing for chemically stable and less complex main-chain polymers. Our results indicate that the incorporation of

higher-level knowledge not only eliminates the creation of chemically infeasible structures as expected but also improves the efficiency of the genetic design in general. The incorporation of other higher-level knowledge such as toxicity, ease of synthesis, environmental impact, cost, etc., is an important consideration for future work.

The proposed approach suffers from two main drawbacks. One is the heuristic nature of the search, and that there is no guarantee of finding the target solution. But then, this criticism would be applicable to the other heuristic approaches as well. The other drawback is that the selection of the genetic design parameter values would require some experimentation.

It is evident from the case studies that the genetic design is extremely proficient at rapidly locating favorable regions in the design space. It is, however, less effective at performing very localized searches. This was seen in many design scenarios when a near-optimal design was reached in a few dozen generations, but it took the genetic design several hundred generations more to go from there to the target even though it was quite close. Adapting the genetic design parameters may alleviate this problem. Another alternative is to integrate good local search procedures such as simulated annealing and mathematical programming formulations with genetic design. One main appeal of the genetic design approach is the ability to integrate it with other methods such as math programming, expert systems, etc. to benefit from the advantages of these methods. In conclusion, the problem independent and efficient nature of the approach, the ease with which chemical, biological, design, process knowledge, and constraints can be incorporated make the genetic design framework for CAMD very appealing and worthy of further critical investigation for large-scale molecular design problems.

REFERENCES AND NOTES

- (1) Derringer, G. C.; Markham, R. L. A Computer-Based Methodology for Matching Polymer Structure with Required Properties. *J. Appl. Polymer Sci.* **1985**, *30*, 4609–4617.
- (2) Gani, R.; Brignole, E. A. Molecular Design of Solvents for Liquid Extraction Based on UNIFAC. *Fluid Phase Equil.* **1983**, *13*, 331–340.
- (3) Brignole, E. A.; Bottlini, S.; Gani, R. A Strategy for Design and Selection of Solvents for Separation Processes. *Fluid Phase Equil.* **1986**, *29*, 125–132.
- (4) Joback, K. G.; Stephanopoulos, G. Designing Molecules Possessing Desired Physical Property Values. *Proc. FOACPD '89*. Snowmass, CO, 1989, 363–387.
- (5) Macchietto, S.; Odele, O.; Omatson, O. Design of Optimal Solvents for Liquid-Liquid Extraction and Gas Absorption Processes. *Trans. IChemE*. **1990**, *69*, Part A, 429–433.
- (6) Klein, J. A.; Wu, D. T. Computer-Aided Mixture Design with Specified Property Constraints. In European Symposium on Computer-Aided Process Engineering-ESCAPE-1; Gani, R., Ed.; Elsevier, Denmark, 1992; pp 229–236.
- (7) Nagasaka, K.; Wada, H.; Yoshimitsu, H.; Yasuda, H.; Yamanouchi, T. Expert System for Polymer Design. *AIChE Annual Meeting*, Chicago, IL, November 1990; p 39e.
- (8) Gani, R.; Nielsen, B.; Fredenslund Aa. A Group Contribution Approach to Computer-Aided Molecular Design. *AIChE J.* **1991**, *37*(9), 1318–1332.
- (9) Kier, L. B.; Lowell, H. H.; Frazer, J. F. Design of Molecules from Quantitative Structure-Activity Relationship Models. 1. Information Transfer between Path and Vertex Degree Counts. *J. Chem. Inf. Comput. Sci.* **1993**, *33*(1), 143–147.
- (10) Skvortsova, M. I.; Baskin, I. I.; Slovokhotova, O. L.; Palyulin, V. A.; Zefirov, N. S. Inverse Problem in QSAR/QSPR Studies for the Case of Topological Indices Characterizing Molecular Shape (Kier Indices). *J. Chem. Inf. Comput. Sci.* **1993**, *33*(4), 630–634.
- (11) Keir, L. B.; Hall, L. H. *Molecular Connectivity in Chemistry and Drug Research*. Academic; Academic Press: New York, 1976.
- (12) Keir, L. B.; Hall, L. H. *Molecular Connectivity in Structure-Activity Analysis*; John Wiley: London, 1986.
- (13) Venkatasubramanian, V.; Chan, K.; Caruthers, J. M. Designing Engineering Polymers: A Case Study in Product Design. *AIChE Annual Meeting*. Miami, FL, November 1992; p 140d.
- (14) Venkatasubramanian, V.; Chan, K.; Caruthers, J. M. Computer-Aided Molecular Design Using Genetic Algorithms. *Computers Chem. Engng.* **1994**, *18*(9), 833–844.
- (15) Holland, J. H. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press: Ann Arbor, MI, 1975.
- (16) Baker, J. E. Reducing Bias and Inefficiency in the Selection Algorithm. In Proceedings of the Second International Conference on Genetic Algorithms; Grefenstette, J. J., Ed.; Lawrence Erlbaum Associates: Hillsdale, NJ, 1987; pp 14–21.
- (17) De Jong, K. A. *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*. Ph.D. Thesis, University of Michigan, Ann Arbor, MI, 1975.
- (18) Goldberg, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley: Reading, MA, 1989.
- (19) *Handbook of Genetic Algorithms*; Davis, D., Ed.; Van Nostrand Reinhold: New York, 1991.
- (20) *Foundations of Genetic Algorithms*; Rawlins, G. J. E., Ed.; Kaufmann Publishers: San Mateo, CA, 1991.
- (21) Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*; Springer-Verlag: Berlin, 1992.
- (22) Hibbert, D. B. Genetic Algorithm in Chemistry. *Chemometrics and Intelligent Laboratory Systems*. **1993**, *19*, 277–293.
- (23) Lucasius, C. B.; Kateman, G. Understanding and using Genetic Algorithms part 1. Concepts, Properties and Context. *Chemometrics and Intelligent Laboratory Systems*. **1993**, *19*, 1–33.
- (24) van Krevelen, D. W.; Hoftyzer, P. J. *Properties of Polymers, their Estimation and Correlation with Chemical Structure*, 3rd ed.; Elsevier Scientific: Amsterdam, 1990.
- (25) Barton, D.; Ollis, W. D., Eds. *Comprehensive Organic Chemistry: The Synthesis and Reaction of Organic Compounds*, 1st ed.; Pergamon Press: New York, 1979.

CI940220U