# Stochastic Generator of Chemical Structure. 2. Using Simulated Annealing To Search the Space of Constitutional Isomers

Jean-Loup Faulon[†]

Sandia National Laboratories, Albuquerque, New Mexico 87185-0710

Received November 30, 1995[⊗]

While there have been many theoretical and applied studies to explore the space of conformational isomers, little has been reported regarding constitutional (i.e., structural) isomers. Deterministic algorithms have been proposed in the past but are limited to small molecules because constitutional space sizes scale exponentially with the number of atoms. The present paper proposes a stochastic algorithm based on the simulated annealing method that searches constitutional isomers with desired properties. The algorithm is general enough to be used for any class of organic and inorganic compounds, including cyclic and cross-linked structures. Theoretically, the algorithm is shown to be efficient (i.e., polynomial). Practically, the algorithm performs remarkably well even in very large constitutional space sizes (up to $10^{32}$). Applications of this algorithm are suggested in the context of computer-aided molecular design. The practical examples given in the paper include the search for structures having specific topological indices, the search for structures having low log *P* (octanol/water) partition coefficient, and the search for global minimum of energy in constitutional spaces. The proposed algorithm appears to be the first one that searches chemical structures without restricting the search space.

## INTRODUCTION

Constitutional isomers are molecules that have identical molecular formulas but different structural formulas, i.e., different connectivity between the atoms. Conformational isomers are molecules having the same structural formula but different three-dimensional shapes. While there have been many theoretical and applied studies to explore the space of conformational isomers, few have been reported regarding constitutional isomers.

Mathematically, the two spaces are different. The conformational space is a continuous space. A continuous sequence of infinitesimal geometrical transformations can be found to transform one conformer to another. The mathematical tools used to explore this space are continuum calculus tools—classical and quantum mechanics. Techniques such as molecular mechanics, molecular dynamics, semiempirical, and *ab initio* calculations have been applied in almost every field of chemistry.[1] Common applications are folding, docking, kinetics, diffusion, and adsorption.[1]

The space of constitutional isomers is a discrete finite space. The elements comprising the space are structural formulas, or molecular graphs, and there is a finite number of graphs corresponding to a given molecular formula. The tools used to explore this space come from discrete mathematics such as graph theory, group theory, and combinatorics. The techniques proposed to search the constitutional space have been used almost exclusively in chemical information. These techniques include isomer counting and generating[2] and database searching.[3] Because constitutional spaces can be combinatorially large, the above techniques are generally limited to small compounds or restricted spaces (such as databases). To date, the practical application of constitutional isomers searching is mostly limited to database searching. Yet, there are many applications in structural elucidation of natural and synthesized compounds, design of materials with desired properties (solvent, polymers, membranes, catalysts, etc.), and rational drug design if one were able to develop fast algorithms able to handle large combinatorial spaces.

This article is the second of a series,[4] where fast algorithms are proposed to explore the space of constitutional isomers. The first article presented computational techniques based on random sampling for the purpose of structural elucidation. The present article proposes an algorithm based on the simulated annealing method that performs stochastic searches of constitutional spaces. The algorithm is general enough to be used for any class of organic and inorganic compounds. The algorithm can screen the entire constitutional space and can also search structures in specific parts of the space. Practical applications are given in the last section in the context of chemical information and molecular design. The forthcoming paper will discuss other applications and other computational techniques based on the Genetic Algorithm.

## STOCHASTIC VERSUS DETERMINISTIC

Two approaches can be used to explore any given space of combinatorial structures—*deterministic* and *stochastic*.[5] While the deterministic approach systematically explores a specific part or the entire space, the stochastic approach randomizes the exploration of the space.

Deterministic techniques should give the correct answer to any given problem (within the framework of the mathematical tools used), but stochastic techniques do not guarantee that the solution found is indeed the best solution. For example, if one uses the proper force field or the proper *ab initio* basis set, deterministic techniques will always succeed in finding the global minimum of energy of a given compound; however, stochastic techniques may fail. Yet, stochastic techniques are routinely used for conformational searches, because deterministic techniques are computation-

[†] E-mail: jfaulon@sandia.gov, URL: http://fuelan.sandia.gov/jfaulon.

[⊗] Abstract published in *Advance ACS Abstracts,* April 1, 1996.

S0095-2338(95)00179-X CCC: $12.00

ally too expensive.[5] In fact, according to Table 1, the computational time required to explore the entire spaces of conformational or constitutional isomers scales exponentially with the number of atoms.

For conformational isomers, both approaches have been used. Energy minimization and molecular dynamics are examples of deterministic techniques that explore part of the conformational space.[1] Systematic[6] and quasi-exhaustive[8] grid search are deterministic techniques searching the entire conformational space. Random search,[6] Monte-Carlo,[9] simulated-annealing,[10] evolutionary programming,[10] and the genetic algorithm[10,11] are examples of stochastic techniques that have been applied to search conformational isomers.

For constitutional isomers, deterministic techniques have been developed since the 1960s to count and generate isomers[2] and to search structures and substructures in databases.[3] Solving these complex problems have led to the use of Polya theory of counting,[2] to the development of structure generators,[2] and to the elaboration of graph and subgraph isomorphism algorithms.[3] It is interesting to note that the above problems have been the subject of research for more than 45 years and are still under investigation.[12−14] At first, these techniques were integrated into computer-assisted structural elucidation systems. A review of these systems is given in the first article of the series.[4] More recently, new uses for deterministic techniques have been found for computer-aided molecular design.[15−22] A crucial problem of computer-aided molecular design is to retrieve molecular structures that correspond to given physicochemical properties. This problem is referred to in the literature as the inverse imaging problem.[19−22] Inverse imaging solutions are similar to the solutions developed for structural elucidation purposes. Hence, as in the case of structural elucidation, the proposed solutions are limited to small compounds due to the inefficiency of deterministic techniques to deal with large combinatorial spaces.

Stochastic techniques have been used with constitutional isomers only recently, and few solutions have been published. Among them, simulated annealing algorithms have been proposed to evaluate the chemical distance between molecular graphs,[23] and Genetic Algorithms have been developed to match constitutional isomers.[24] All the previous solutions were intended to be used for database searching. It is important to note that database searching is only of limited help in structural elucidation or molecular design. In fact, it is obvious that a database search will give the correct answer only if the solution is already in the database. The author has found only five reports that have attacked the problem of stochastic searching without using a database.[4,15,18,21,22] All have restrictions.

The first published stochastic technique is a generator that constructs random linear polymers.[15] The random construction is repeated until a polymer is found that matches a given set of physical properties. The technique is simple to implement but time consuming, since the algorithm does not try to refine the solution as it progresses.

A paper was published in 1991 proposing a technique that generates random samples of chemical structures from a database of fragments.[18] The intent of the authors was to use their technique for drug discovery. However, sampling techniques are useful to calculate average properties, but sampling techniques are not necessarily appropriate to search for structures that best match physicochemical properties. Hence, as in the previous case, this technique may be time consuming for molecular design problems since the solutions are not improved as the algorithm progresses.

As mentioned before, the first paper of this series proposes a sampling technique in the context of structural elucidation.[4] This technique generates samples of three-dimensional chemical structures that match a given set of structural analytical data. The technique differs from the previous ones because all the structures generated are nonidentical. This insures a better coverage of the constitutional space. Furthermore, the technique also provides an estimate of the constitutional space size. This information is essential to carry out statistical analyses. However, as in the two previous cases, this sampling technique is appropriate for calculating average properties but may be time consuming with molecular design problems.

Recently, two efficient solutions have been proposed in the context of computer-aided molecular design.[21,22] Both solutions are based on the Genetic Algorithm. The first paper[21] presents an application in the context of the design of polymers with desired properties. The second paper[22] proposes a method that suggests combinatorial libraries. While these techniques are the first to be able to efficiently screen large combinatorial space, they both are restricted to non-cross-linked polymeric structures. This restriction is due to the fact that the Genetic Algorithm implies the structures generated to be coded into the form of genomes, i.e., linear series of genes.[25] Both solutions have chosen to associate genes with molecular fragments. Since genomes are linear, cross-linked structures cannot be generated.

To summarize, the stochastic techniques that have been proposed in the past are not fully satisfactory because they are either time consuming or restricted to limited spaces (databases or linear structures). Therefore, there is a need to design fast algorithms that explore spaces of constitutional isomers without restrictions.

## SIMULATED ANNEALING ALGORITHM

Monte-Carlo (MC), and Simulated Annealing (SA) are simple algorithms that were initially designed to provide efficient simulations of collections of particles in condensed matter physics.[26] In the early 1980s, these algorithms were rediscovered, and it was shown that MC and SA are powerful techniques to resolve combinatorial problems.[27] MC and SA algorithms are similar. In each step of these algorithms, a particle is given a small random displacement, and the resulting change, $\Delta e$, in the energy of the system is computed. If $\Delta e \leq 0$, the displacement is accepted, and the new configuration is used as the starting point of the next step. The case $\Delta e \geq 0$ is treated probabilistically: the probability that the configuration is accepted is $\exp(-\Delta e/kT)$. Random numbers uniformly distributed in the interval [0,1] are convenient ways of implementing the random part of the algorithm. By repeating the above basic step many times, one simulates the thermal motion of particles in thermal contact with a heat bath at temperature $T$. With MC the simulations are carried out at equilibrium at a given temperature $T$, while with SA the temperature is decreased according to a predefined cooling program (annealing schedule). Using a cost function in place of the energy and defining configurations by a set of parameters, it is straightforward with the above procedure to generate a population of configurations for a given optimization problem.

As already mentioned, SA techniques have been used in the past to search for global minimum of energy in conformational spaces.[9,10] To the best of my knowledge, SA has not yet been used to explore constitutional spaces. For conformational isomers the random displacement of the SA algorithm consists of slightly modifying the conformation, by either moving atoms or by changing dihedral angles of rotational bonds. According to the definition of constitutional isomers, the SA random displacement for these isomers must consist of changing the connectivity between the atoms. Connectivity can be changed by deleting bonds, creating bonds, or modifying bond order. With the convention that a bond is deleted when its order is set to zero and a bond is created when its order is switched from zero to a positive value, all changes of connectivity can be performed by modifying bond order. Because all constitutional isomers have the same number of bonds, when a bond order is increased (or decreased) another bond order must be decreased (or increased). Hence, changing the connectivity implies the selection of at least two bonds (four atoms) in the molecular graphs. Let $x_1$, $y_1$, $x_2$, and $y_2$ be the four selected atoms. Let $a_{11}$, $a_{12}$, $a_{21}$, and $a_{22}$ be the orders of the bonds [$x_1$,$y_1$], [$x_1$,$y_2$], [$x_2$,$y_1$], and [$x_2$,$y_2$] in the initial molecular graph, and let $b_{11}$, $b_{12}$, $b_{21}$, and $b_{22}$ be the order of the same bonds after the SA random displacement is achieved. Because the valences of the four selected atoms must remain constant, the following equations hold

$$b_{11} + b_{12} = a_{11} + a_{12} \qquad (1)$$

$$b_{11} + b_{21} = a_{11} + a_{21} \qquad (2)$$

$$b_{21} + b_{22} = a_{21} + a_{22} \qquad (3)$$

$$b_{12} + b_{22} = a_{12} + a_{22} \qquad (4)$$

$$b_{\mathrm{ij}} \geq 0 \ \mathrm{i,j} = 1, ..., 3 \qquad (5)$$

$$b_{\mathrm{ij}} \leq 3 \ \mathrm{i,j} = 1, ..., 3 \qquad (6)$$

where it is assumed that the maximum bond order is equal to three (triple bond). The system of eqs 1−6 can be expressed as follows:

$$b_{12} = a_{11} + a_{12} - b_{11} \qquad (7)$$

$$b_{21} = a_{11} + a_{21} - b_{11} \qquad (8)$$

$$b_{22} = a_{22} - a_{11} + b_{11} \qquad (9)$$

$$b_{11} \geq \mathrm{MAX}\ (0, a_{11} - a_{22}, a_{11} + a_{12} - 3, a_{11} + a_{21} - 3) \qquad (10)$$

$$b_{11} \leq \mathrm{MIN}\ (3, a_{11} + a_{12}, a_{11} + a_{21}, a_{11} - a_{22} + 3) \qquad (11)$$

The SA random displacement for constitutional isomers is therefore obtained by choosing at random a value for $b_{11}$ different from $a_{11}$ and verifying eqs 10 and 11. Values for $b_{12}$, $b_{21}$, and $b_{22}$ are obtained using eqs 7, 8, and 9. The SA algorithm is further detailed in Table 2. Examples of SA random displacement are shown in Figure 1.

The initial graph used by the SA algorithm (cf. Table 2) is generated using the equivalent classes deterministic algorithm.[28] This algorithm, which was originally designed to construct in 3D space all the structural isomers corresponding to a set of chemical constraints, is run in the present case until one structure is obtained. The random numbers used by the SA algorithm are generated using an initial seed number. Two runs of the SA algorithm using the same data structures and the same seed will lead to the same results. For the purpose of this paper, the SA algorithm was coupled with the SIGNATURE program.[29] Therefore, the SA algorithm share the same data structures than this program. The molecular modeling products that are currently interfaced with the SIGNATURE program are PCMODEL (Serena Software), Cerius2 3.2, and Insight 2.3.8 (Biosym/MSI). Files can be read and written in the specific formats of these programs. The above commercial products are used in the present paper to optimize the conformations of the chemical structures generated by the SA algorithm.

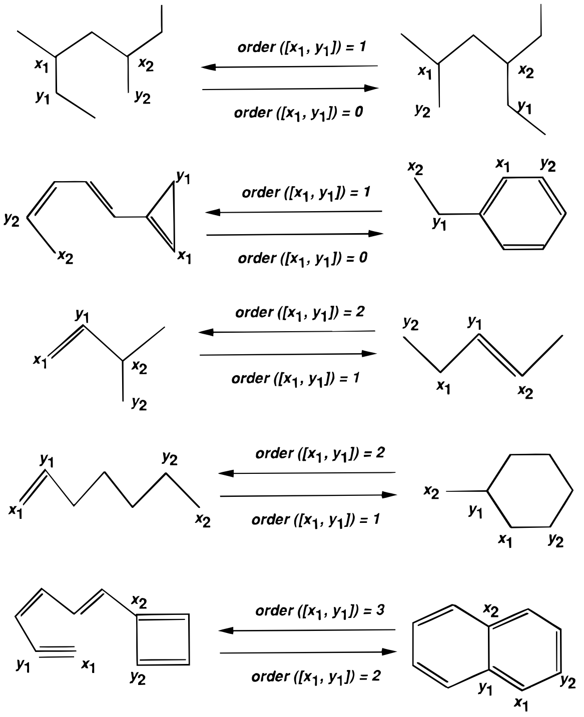

**Figure 1.** Example of SA random displacement. The orders of the bonds [$x_1$,$y_1$], [$x_1$,$y_2$], [$x_2$,$y_1$], and [$x_2$,$y_2$] are changed according to eqs 7−11.

While examining the computational complexity of the algorithm depicted in Table 2, one observes that only two steps depend on the size of the structures constructed. The first step verifies if the chemical constraints are met (step 2.3). The constraints for constitutional isomers consist of checking if the structures are connected and if the valences of the atoms are respected. Both constraints can be verified in a time O($n$), where $n$ is the number of atoms.[30] If the structure generated does not meet the chemical constraints, the steps 2.1−2.3 may be repeated several times. In the worst scenario, all values for $x_1$, $y_1$, $x_2$, and $y_2$ are tested, and therefore the steps are repeated $n^4$ times. Hence, the upper limit for the time complexity of steps 2.1−2.3 is O($n^5$). However, as demonstrated by all the tests carried out in the present paper, the chemical constraints are generally met after repeating the steps 2.1−2.3 no more than twice; consequently, the time complexity is on average O($n$). The second step that depends on the structure size is the computation of the cost function (step 2.4). As shown in Table 1, most of the cost functions used in computational chemistry (group

**Table 1.** Comparison of Several Polynomial and Exponential Time Complexity Functions[a]

| scaling function | problem size $n$[b] | | | | | | problem examples in computational chemistry[c] |
|---|---|---|---|---|---|---|---|
| | 10 | 20 | 30 | 40 | 50 | 60 | |
| $n$ | 10 μs | 20 μs | 30 μs | 40 μs | 50 μs | 60 μs | group contribution calculations<br>force field calculations with cutoff |
| $n^2$ | 100 μs | 400 μs | 900 μs | 0.0016 s | 0.0025 s | 0.0036 s | topological indices calculations<br>force field calculations without cutoff<br>semiempirical calculations |
| $n^3$ | 0.001 s | 0.008 s | 0.027 s | 0.064 s | 0.125 s | 0.216 s | density functional calculations<br>ab initio calculations (Hartree−Fock) |
| $n^5$ | 0.1 s | 3.2 s | 24.3 s | 1.7 min | 5.2 min | 13.0 min | |
| $2^n$ | 0.001 s | 1.0 s | 17.9 min | 12.7 days | 35.7 years | 366.0 centuries | deterministic search in constitutional and conformational spaces[d] |
| $3^n$ | 0.059 s | 58.0 min | 6.5 years | 3855.0 centuries | $2.0 \times 10^8$ centuries | $1.3 \times 10^{13}$ centuries | |
| $n!$ | 3.6 s | 771.4 centuries | $8.4 \times 10^{16}$ centuries | $2.6 \times 10^{32}$ centuries | $9.6 \times 104^8$ centuries | $2.2 \times 10^{66}$ centuries | published algorithms for molecular graph isomorphism[e] |

[a] This table was adapted for computational chemistry problems from Figure 1.2 in ref 5. All times are given for a hypothetical computer processing a problem of size one in a microsecond. [b] Except for isomorphism problems (cf. note e), $n$ is the number of atoms. [c] The examples are those mentioned in the text. See text for references. [d] It is well-known that the number of constitutional isomers is greater than $2^n$, where $n$ is the number of carbon atoms (cf. ref 2). For conformational isomers, deterministic techniques such as grid search algorithms screen conformational spaces by changing the dihedral angles of rotational bonds (cf. ref 6). If $k$ is the number of possible angles for each bonds, and $n$ is the number of rotational bonds, grid search algorithms scale $k^n$. [e] This is the time complexity in a worst case scenario. Although isomorphism algorithms are generally efficient for most molecules; for all published algorithms there exist molecular structures where isomorphism cannot be checked in a polynomial time. For most isomorphism algorithms $n$ is the number of atoms having the same label (i.e., the same extended connectivity), which is much smaller than the total number of atoms (cf. ref 7). Therefore, the time indicated are for cases where there are between 10 and 60 atoms having the same label.

**Table 2.** The SA Algorithm

```
(1) Generate an initial structure (G) using a deterministic technique
(2) For each SA step
      (2.1) choose randomly four distinct atoms $x_1$, $y_1$, $x_2$, $y_2$ in G
      (2.2) G′ = displacement ([$x_1$,$y_1$],[$x_1$,$y_2$],[$x_2$,$y_2$],[$x_2$,$y_1$])
      (2.3) if G′ does not meet the chemical constraints go to (2.1)
      (2.4) compute the cost function $e$(G′)
      (2.5) $\Delta e = e(G') - e(G)$
      (2.6) RN = random number between 0 and 1
      (2.7) compute the coefficient $kT$ according to the
            annealingschedule
      (2.8) if $\Delta e < 0$ or RN < $\exp(-\Delta e/kT)$ then G = G′

displacement ([$x_1$,$y_1$],[$x_1$,$y_2$],[$x_2$,$y_2$][$x_2$,$y_1$])
      (1) $a$11 = order([$x_1$,$y_1$]), $a$12 = order([$x_1$,$y_2$]),
          $a$22 = order([$x_2$,$y_2$]), $a$21 = order([$x_2$,$y_1$])
      (2) choose $b$11 ≠ $a$11 at random in the range
          [MAX(0,$a$11-$a$22,$a$11+$a$12-3,$a$11+$a$21-3),
          MIN(3,$a$11+$a$12,$a$11+$a$21,$a$11-$a$22+3)]
      (3) order([$x_1$,$y_1$]) = $b$11,
          order([$x_1$,$y_2$]) = $a$11+$a$12-$b$11,
          order([$x_2$,$y_2$]) = $a$22-$a$11+$b$11,
          order([$x_2$,$y_1$]) = $a$11+$a$21-$b$11
      (4) return the modified graph
```

contribution, topological indices, energy, ...) can be achieved in a time O($n^k$), where $1 \leq k \leq 5$. Overall, the time complexity of the SA algorithm is O($Nn^k$), where $N$ is the number of structures generated, and $k = 5$ in the worst case. Therefore, the proposed SA algorithm is a polynomial algorithm.

Before applying the SA technique to chemical problems, one needs to verify if all constitutional isomers can be generated using the algorithm described in Table 2. To accomplish this, the algorithm must be set up to maximize the construction of new structures. A new structure is generated by the SA algorithm when the SA random displacement is accepted. Maximizing the construction of new structures can be achieved by imposing the same cost value for all structures. With the notations defined in Table 2, if all structures have the same cost, then $\Delta e = 0$, and RN $< \exp(-\Delta e/kT)$ is always true; therefore, every new structure is accepted without considering its energy. Using the above setup, the SA algorithm was run for 100 steps to generate a sample of 100 structures composed of four carbon atoms and six hydrogen atoms. The results are shown in Figure 2. All nine isomers of $C_4H_6$ were found by the SA algorithm. Figure 2 demonstrates the ability of the SA random displacement to switch from cyclic to noncyclic structures and from structures having only single bonds to structures having double and triple bonds. One notices in Figure 2 that each isomer appears several times. The structures are not unique because the SA algorithm does not check for isomorphic (i.e., redundant) structures. Since the algorithm does not perform an isomorphism check, one may wonder about its effective-

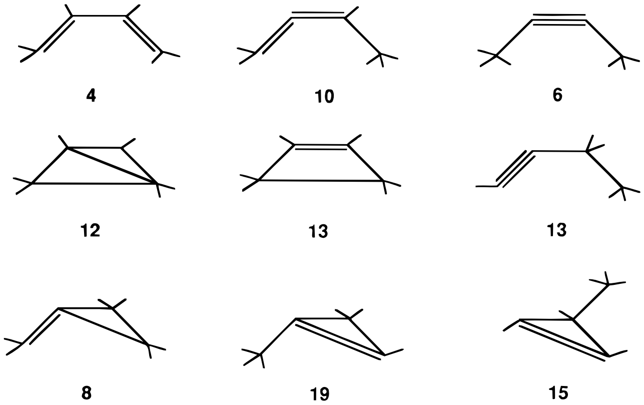

**Figure 2.** SA results for the constitutional space of $C_4H_6$ (sample of 100 structures). Each number indicates the multiplicity of the structure in the sample.

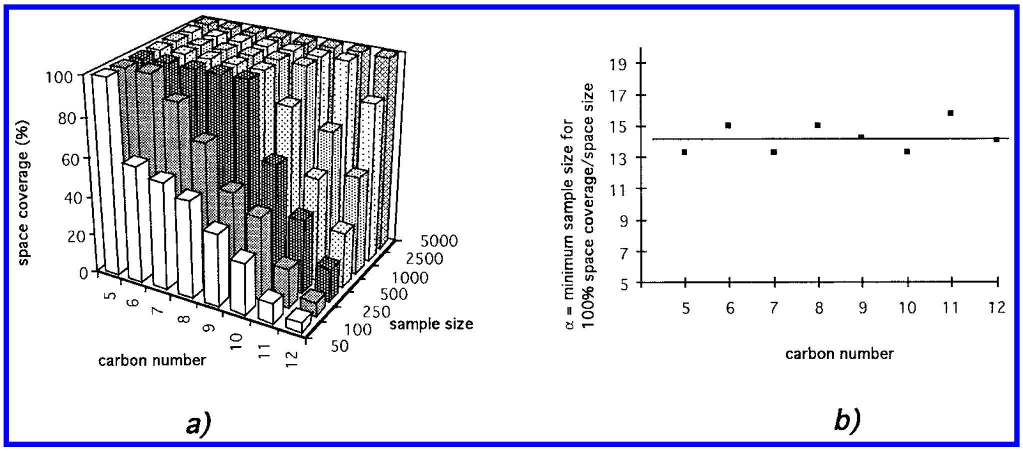

**Figure 3.** Space coverage capability of the SA algorithm. (a) Space coverage versus sample size for all paraffins up to 12 carbon atoms. (b) Ratio between the sample size needed for 100% space coverage and the space size for all paraffins up to 12 carbon atoms.

ness. In other words if too many structures are identical the algorithm may fail in its goal, which is to search chemical structures in large constitutional spaces. It is therefore important to verify what is the minimum sample size needed to cover the entire constitutional space.

In an attempt to find the relationship between sample size and space size, the SA algorithm was asked to generate samples of various sizes for paraffin structures. More precisely, samples containing between 50 and 5000 structures were constructed for paraffins containing from 5 to 12 carbon atoms. The SA algorithm was set up as in the previous run. The results are depicted in Figure 3. From Figure 3a, it can be observed that the sample size needed for 100% space coverage increases rapidly as a function of the number of atoms. This is not surprising considering that the space size (i.e., the number of nonisomorphic structures) increases exponentially with the number of atoms. However, it is interesting to note with Figure 3b that the minimum sample size needed for 100% space coverage is linearly proportional to the space size. Precisely, for the structures tested, the sample size is on average $\alpha = 14.3 \pm 0.9$ times larger than the space size.[31] The fact that the sample size does not increase faster than the space size rules out the need for isomorphism check in the SA algorithm. Futhermore, as it is shown in the following section, when searching for chemical structures having specific properties, the SA algorithm succeeds in finding the solutions without performing isomorphism checks even with very large constitutional spaces. This is due to the power of the SA technique, which can locate global minima or global maxima by covering only small fractions of combinatorial spaces.[27] Considering the complexity of isomorphism algorithms (cf. Table 1), the proposed SA algorithm leads to a substantial saving in computing time compared to any technique that would check for redundant structures.

## RESULTS AND DISCUSSION

A new SA algorithm has been proposed in the last section. This algorithm generates isomers for any given constitutional space. However, as the size of the structure grows, generating all the isomers quickly becomes impossible since the constitutional space size increases exponentially with the number of atoms. The real advantage of the SA technique is to be able to search for structures having specific properties (i.e., specific cost function values) without having to explore the entire constitutional space. The purpose of this section is to test the searching ability of the SA algorithm with large constitutional spaces and different cost functions.

Several cost functions can be used to test the SA algorithm (cf. Table 1). Among them topological indices have the advantage of being simple to compute, and have been extensively used in QSAR and QSPR studies.[32] The Wiener index, or Wiener number[33] was chosen because this index appears to be one of the most cited and utilized.[32] For example, the Wiener index has been used to predict physical properties of organic compounds (boiling points, molar volumes, molar refractions, heats of vaporization, surface tensions, melting points, critical temperatures, and critical pressures).[34] More recently, the Wiener index was utilized to predict the structure/activity relationships of antiviral drugs.[35] Another advantage of the Wiener index is its low degeneracy compared to other indices.[32] This implies that only few structures have the same Wiener number. The Wiener index is thus a challenging test for the SA algorithm.

The SA algorithm was asked to search for the highest Wiener number for constitutional isomers of paraffin structures. It is easy with this search to verify if the SA algorithm succeeded or failed. The Wiener index is a convenient measure of the compactness of a molecule. The smaller the Wiener number is, the higher is the compactness of the molecule. Thus, the paraffin structures having the largest Wiener numbers are always the *n*-paraffin structures. It has been shown[32] that for *n*-paraffins the Wiener number can simply be calculated using the equation

$$n(n^2-1)/6 \tag{12}$$

where *n* is the number of carbon atoms.

A series of runs was carried out for paraffin structures containing from 10 to 32 carbon atoms. While there are only 75 isomers for $C_{10}H_{22}$, the constitutional space size for $C_{32}H_{66}$ is greater than 2.7 × $10^{10}$. The purpose of these runs was to determine the annealing schedule giving the best results (i.e., the fastest convergence on the desired result). A total of 29 schedules were tested where the initial temperature and the rate of temperature decrease were varied. For each run, the difference between the actual highest Wiener number and the number found by the SA algorithm was computed. To test the rate of convergence, all the runs were limited to 100 steps. Results for the isomers of $C_{24}H_{50}$

**Table 3.** Highest Wiener Number Found by the SA Algorithm for the Isomers of $C_{24}H_{50}$ (Actual Value is 2300)

| schedule function[a] | $(kT)_o = 0$ | | $(kT)_o = 10$ | | $(kT)_o = 100$ | | $(kT)_o = 1000$ | | $(kT)_o = 10\,000$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| $f_0$ | 2182 | (5.1)[b] | 2188 | (4.8) | 2069 | (10.0) | 1667 | (27.5) | 1969 | (14.4) |
| $f_1$ | N/A | | 2204 | (4.2) | 2015 | (12.4) | 1735 | (24.5) | 1969 | (14.4) |
| $f_2$ | N/A | | 2100 | (8.7) | 2042 | (11.21) | 1919 | (16.6) | 2026 | (11.9) |
| $f_4$ | N/A | | 2174 | (5.5) | 2171 | (5.6) | 2203 | (4.2) | 2207 | (4.0) |
| $f_8$ | N/A | | 2174 | (5.4) | 2174 | (5.4) | 2188 | (4.9) | 2243 | (2.5) |
| $f_{16}$ | N/A | | 2171 | (5.6) | 2171 | (5.6) | 2034 | (11.5) | 2215 | (3.7) |
| $f_{32}$ | N/A | | 1971 | (14.3) | 1971 | (14.3) | 1971 | (14.3) | 2118 | (7.9) |

[a] The schedule functions are defined as follows, where $(kT)_o$ is the initial $kT$ coefficient, $t$ is the current step number, and $\Delta t$ is the total number of steps (i.e., 100). $f_{0:}$ $(kT)_t = (kT)_o$; $f_{1:}$ $(kT)_t = (kT)_o - (kT)_o/\Delta t$; $f_{2:}$ $(kT)_t = (kT)_o - 2(kT)_o t/\Delta t$: $f_{4:}$ $(kT)_t = (kT)_o - 4(kT)_o t/\Delta t$; $f_{8:}$ $(kT)_t = (kT)_o - 8(kT)_o t/\Delta t$; $f_{16:}$ $(kT)_t = (kT)_o - 16(kT)_o t/\Delta t$; $f_{32:}$ $(kT)_t = (kT)_o - 32(kT)_o t/\Delta t$; for all functions $f$: if $(kT)_t < 0$ then $(kT)_t = 0$. [b] All numbers in parentheses are the relative differences in percentage between the actual highest Wiener number (2300) and the computed Wiener numbers.

**Table 4.** Average Relative Difference between Actual and Computed Wiener Numbers as a Function of the Annealing Schedule[a]

| schedule function[b] | $(kT)_o = 0$ | $(kT)_o = 10$ | $(kT)_o = 100$ | $(kT)_o = 1000$ | $(kT)_o = 10\,000$ |
|---|---|---|---|---|---|
| $f_0$ | 6.7 | 7.9 | 11.7 | 17.3 | 16.6 |
| $f_1$ | N/A | 7.1 | 7.5 | 14.9 | 16.6 |
| $f_2$ | N/A | 8.5 | 8.2 | 12.9 | 11.9 |
| $f_4$ | N/A | 5.7 | 7.4 | 7.11 | 7.8 |
| $f_8$ | N/A | 6.5 | 4.8 | 7.0 | 6.4 |
| $f_{16}$ | N/A | 5.8 | 5.0 | 4.8 | 5.7 |
| $f_{32}$ | N/A | 7.8 | 7.0 | 5.7 | 6.3 |

[a] The differences have been averaged over 12 paraffin structures containing an even number of carbon atoms ranging from 10 to 32. [b] The schedule functions are defined in Table 3.

are presented in Table 3. Table 4 gives the average relative difference between the actual and computed Wiener number for all the tested paraffin structures as a function of the annealing schedule. It is interesting to note in Table 3, that despite the fact that the Wiener number for $C_{24}H_{50}$ varies from 981 (cf. Table 6) to 2300 and that the constitutional space size is equal to 14 490 245, several schedules found values less than 5% lower than the actual highest number with only 100 steps. This was observed to be the case for all the tested paraffins. For example, with $C_{32}H_{66}$, the highest Wiener number found by the algorithm was 5216 while the actual number is 5456. This gives a relative difference of only 4.4%. This result was obtained with the schedule function $f_8$ and an initial $kT$ value equal to 100. According to Table 4, the same schedule ($f_8$, $(kT)_o = 100$) gives the best results overall.

The annealing schedule is not the only parameter that influences the performances of the SA algorithm. The number of steps is also an important variable. To verify the effect of the number of steps, a series of runs was performed where the number of steps was varied from 50 to 1000. The runs were carried out with the same paraffin structures as in the previous runs. Only the four best annealing schedules found in Table 4 were tested. The average relative differences between the actual and computed Wiener numbers are listed in Table 5. Table 5 demonstrates that the performance of the SA algorithm drastically increases as the number of steps becomes larger. Nonetheless, because the results are not significantly better with 1000 steps, the optimum number of steps may be assigned to 500.

**Table 5.** Average Relative Differences between Actual and Computed Wiener Numbers as a Function of the Number of Steps[a]

| no. of | annealing schedule[b] | | | |
|---|---|---|---|---|
| steps | $f_4$, $(kT)_o = 0$ | $f_8$, $(kT)_o = 100$ | $f_{16}$, $(kT)_o = 100$ | $f_{16}$, $(kT)_o = 1000$ |
| 50 | 13.10[b] | 9.16 | 11.70 | 11.20 |
| 100 | 5.72 | 4.80 | 5.01 | 4.81 |
| 250 | 1.28 | 1.21 | 1.17 | 0.58 |
| 500 | 0.25 | 0.15 | 0.13 | 0.07 |
| 1000 | 0.21 | 0.16 | 0.08 | 0.05 |

[a] The differences have been averaged over 12 paraffin structures containing an even number of carbon atoms ranging from 10 to 32. [b] The annealing schedules are defined in Table 3.

The final series of runs performed with the Wiener index consisted of searching for the lowest and highest Wiener numbers for paraffin structures containing from 20 to 88 carbon atoms. Up to 40 carbon atoms each run was composed of four cycles of the annealing schedule ($f_8$, $(kT)_o = 100$) each containing 500 steps. Above 40 carbon atoms the number of cycles was increased to eight. Therefore for each run either 2000 or 4000 structures were generated. The results are presented in Table 6. The evolution of the lowest and the highest Wiener numbers versus the number of steps is shown in Figure 4 for $C_{84}H_{170}$. The SA algorithm performed extremely well, failing in only one case ($C_{88}H_{178}$) to exactly predict the maximum Wiener number. It is remarkable to note that the algorithm was able to find the actual highest Wiener numbers in space sizes up to $10^{32}$, using 4000 or less steps.

Topological indices are not the only ways to establish structure/property or structure/activity relationships. Another reliable technique is the group contribution approach.[36−38] For example, the group contribution approach is known to give good results in estimating boiling points[36] and in computing the logarithm of the partition coefficient between n-octanol and water (log $P$).[37,38] Log $P$ is usually used to characterize molecular lipophilicity, which is an important factor in biological transport processes.[39] As a consequence, log $P$ has been found to be a crucial parameter in biological structure/activity relationships, and therefore, in rational drug design.[39] Compounds having low log $P$ values have higher solubilities and higher transport rates. Among the current log $P$ estimators reviewed by Klopman *et al.*,[38] the group contribution approach has the advantage of being fast, simple, and accurate. With the group contribution approach, log $P$

**Table 6.** Minimum and Maximum Wiener Number Found by the SA Algorithm for Paraffin Structures Containing up to 88 Carbon Atoms

| carbon no. | size of the constitutional space[a] | min. computed Wiener no. | | max. computed Wiener no. | | actual max. Wiener no.[c] |
|---|---|---|---|---|---|---|
| 20 | $3.6 \times 10^{5}$ | 613 | (499)[b] | 1330 | (317) | 1330 |
| 24 | $1.4 \times 10^{7}$ | 981 | (423) | 2300 | (421) | 2300 |
| 28 | $6.2 \times 10^{8}$ | 1411 | (797) | 3654 | (249) | 3654 |
| 32 | $2.8 \times 10^{10}$ | 1953 | (422) | 5453 | (381) | 5453 |
| 36 | $1.3 \times 10^{12}$ | 2589 | (820) | 7770 | (626) | 7770 |
| 40 | $6.2 \times 10^{13}$ | 3321 | (1481) | 10660 | (403) | 10660 |
| 44 | $3.1 \times 10^{15}$ | 4184 | (3364) | 14190 | (1162) | 14190 |
| 48 | $1.5 \times 10^{17}$ | 5123 | (1405) | 18424 | (1087) | 18424 |
| 52 | $8.0 \times 10^{18}$ | 6187 | (1444) | 23426 | (844) | 23426 |
| 56 | $4.2 \times 10^{20}$ | 7361 | (1328) | 29260 | (1441) | 29260 |
| 60 | $2.2 \times 10^{22}$ | 8717 | (2440) | 35990 | (1964) | 35990 |
| 64 | $1.2 \times 10^{24}$ | 10075 | (2481) | 43680 | (2729) | 43680 |
| 68 | $6.4 \times 10^{25}$ | 11741 | (2489) | 52394 | (2529) | 52394 |
| 72 | $3.5 \times 10^{27}$ | 13347 | (2607) | 62196 | (2367) | 62196 |
| 76 | $1.9 \times 10^{29}$ | 15221 | (2563) | 73150 | (2848) | 73150 |
| 80 | $1.1 \times 10^{31}$ | 16970 | (2706) | 85320 | (2885) | 85320 |
| 84 | $5.9 \times 10^{32}$ | 19208 | (3459) | 98770 | (3060) | 98770 |
| 88 | $3.4 \times 10^{34}$ | 21596 | (3128) | 112987 | (3149) | 113564 |

[a] The space size was computed using Yeh algorithm (ref 12). [b] All numbers in parentheses indicate the step numbers when the minimum or maximum Wiener numbers were found for the first time. [c] The actual maximum Wiener number was calculated using eq 12.

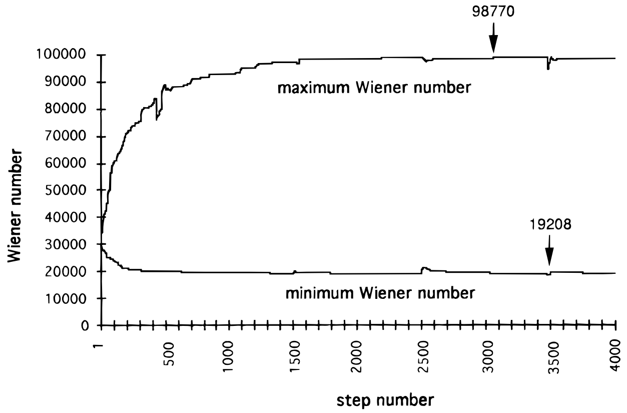

**Figure 4.** Minimum and maximum Wiener numbers for the constitutional isomers of $C_{84}H_{170}$ versus step number.

is predicted using the equation

$$\log P = a + \sum n_i G_i \tag{13}$$

where $a$ is a constant, $G_i$ is the contribution coefficient of the $i$th group, and $n_i$ is the number of occurrences of the $i$th group.

The SA algorithm was run to search for the lowest log $P$ values for aliphatic alcohols containing up to 20 carbon atoms. The group contribution coefficients used for these runs are listed in Table 7. In order to generate only aliphatic alcohols, a new constraint was added to the SA algorithm prohibiting the formation of aliphatic ethers. All SA runs were composed of four cycles of the annealing schedule ($f_8$, $(kT)_o = 100$) each containing 500 steps. The lowest log $P$ alcohols found by the SA algorithm are given in Table 8. Up to five carbon atoms Klopman *et al.*[38] have listed the log $P$ values for all isomeric alcohols. For these compounds, the SA algorithm indeed found the lowest log $P$. For all other structures, the SA algorithm found alcohol structures that have lower log $P$ values than those given by Klopman *et al.*[38] for the same carbon numbers.

**Table 7.** Group Contribution Values for log $P$ Calculations for Aliphatic Alcohols

| group | contribution[a] | remarks |
|---|---|---|
| $-CH_3$ | 0.661 | |
| $-CH_2-$ | 0.415 | |
| $-CH<$ | 0.104 | |
| $>C<$ | −0.107 | |
| $-OH$ | −0.681 | primary alcohol |
| $-OH$ | −0.575 | secondary alcohol |
| $-OH$ | −0.415 | tertiary alcohol |

[a] Contribution values are taken from Klopman *et al.*[36]

The last cost function that was tested is the potential energy. The potential energy is the primary cost function used to explore the space of conformational isomers. In the case of constitutional isomers, searching for the lowest energy structures is of interest in chemical information studies. Potential energy has also practical applications in petroleum chemistry and structural elucidation. For example, it has been shown that there is a strong correlation between the concentrations of specific hydrocarbons in natural petroleum feedstocks and their potential energies.[40] That is, hydrocarbons that have low potential energy are found having high concentrations in natural feedstocks. Within this observation the potential energy can also be used in elucidating natural compounds from MS data. For a given mass, isomers having low potential energies have greater probabilities to occur naturally.

Using the potential energy as a cost function, the SA algorithm was asked to search for the lowest energies of several tetramethyl polycyclic aromatic hydrocarbons ranging from tetramethylbenzene to tetramethyltriphenylene. The constitutional space sizes of these isomers vary from 3 for tetramethylbenzene to 135 for tetramethyltetraphene.[41] Since the space sizes are relatively small, all SA runs were composed of only two cycles of the annealing schedule ($f_8$, $(kT)_o = 100$) each containing 50 steps. The SA random displacement was performed only for aromatic-hydrogen bonds and aromatic-aliphatic bonds. Hence, the formation or deletion of aromatic-aromatic bonds and aliphatic-aliphatic bonds was prohibited. This prevented the modifications of the aromatic cores, and the formations of alkyl substituents different from methyl. For each structure generated, the potential energy was minimized using the DREIDING force field[42] and using a conjugate gradient algorithm for 500 steps or until the root mean square between two successive conformations was lower than 0.1 (kcal/mol)/Å. The lowest energy structures found by the SA algorithm are shown in Figure 5. It is interesting to note that experimental data exist for at least three of the structures listed in Figure 5. In each case, these data agree with the results found by the SA algorithm. For example, it is known that 1,2,4,5-tetramethylbenzene is the most stable isomer among all tetramethylbenzene structures.[43] According to Forster *et al.*,[44] 1,3,6,7-tetramethylnaphthalene is the most abundant isomer in Jurassic, Mioecene, and Ordovician crude oils. Experimentally, 2,4,6,9-tetramethylphenanthrene structures have been found thermodynamically more stable than other tetramethylphenanthrene isomers.[43]

In the first paper of this series, the potential energy distribution of the isomers of $C_8H_{10}$ was calculated using deterministic and sampling techniques.[4] There are 4008 isomers of $C_8H_{10}$. Using a deterministic technique the

**Table 8.** Minimum log *P* Values Found by the SA Algorithm for Aliphatic Alcohols Containing up to 20 Carbon Atoms

| molecular formula | space size[a] | min. log *P* value | corresponding compound name |
|---|---|---|---|
| $CH_4O$ | 1 | -0.72 | methanol |
| $C_2H_6O$ | 1 | -0.31 | ethanol |
| $C_3H_8O$ | 2 | 0.11 | propanol |
| $C_4H_{10}O$ | 4 | 0.46 | isobutyl alcohol |
| $C_5H_{12}O$ | 8 | 0.87 | isopentyl alcohol |
| $C_6H_{14}O$ | 17 | 1.22 | 2,3-dimethylbutan-1-ol |
| $C_8H_{18}O$ | 89 | 2.82 | 3-methyl-2-isopropylbutan-1-ol |
| $C_{10}H_{22}O$ | 507 | 3.58 | 3,4-dimethyl-2-isopropylpropan-1-ol |
| $C_{12}H_{26}O$ | 3057 | 4.35 | 2,3,4,5,6-pentamethylheptan-1-ol |
| $C_{14}H_{30}O$ | 19241 | 5.11 | 4-methyl-3-isopropyl-2-(1,2-dimethylpropyl)pentan-1-ol |
| $C_{16}H_{34}O$ | 124 906 | 5.88 | 2,4,5,6-tetramethyl-3-(1,2-dimethylpropyl)heptan-1-ol |
| $C_{18}H_{38}O$ | 830 219 | 6.64 | 3,4,6,7-tetramethyl-3-isopropyl-2-(1,2-dimethylpropyl)octan-1-ol |
| $C_{20}H_{42}O$ | 5 622 109 | 7.71 | 2,4,5,6,7,8-hexamethyl-3-(1,2-dimethylpropyl)nonan-1-ol |

[a] The space size was computed using Yeh algorithm (ref 12).

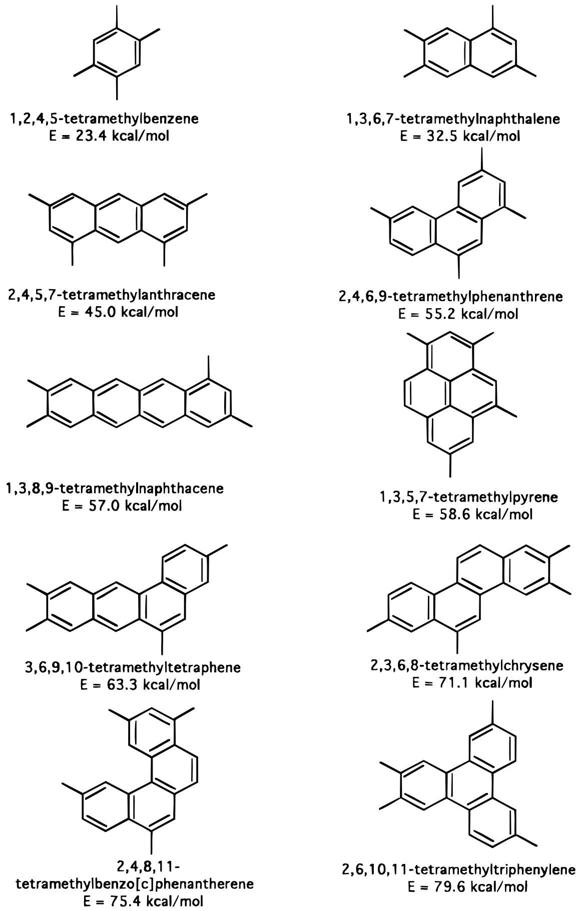
1,2,4,5-tetramethylbenzene
E = 23.4 kcal/mol
1,3,6,7-tetramethylnaphthalene
E = 32.5 kcal/mol
2,4,5,7-tetramethylanthracene
E = 45.0 kcal/mol
2,4,6,9-tetramethylphenanthrene
E = 55.2 kcal/mol
1,3,8,9-tetramethylnaphthacene
E = 57.0 kcal/mol
1,3,5,7-tetramethylpyrene
E = 58.6 kcal/mol
3,6,9,10-tetramethyltetraphene
E = 63.3 kcal/mol
2,3,6,8-tetramethylchrysene
E = 71.1 kcal/mol
2,4,8,11-tetramethylbenzo[c]phenantherene
E = 75.4 kcal/mol
2,6,10,11-tetramethyltriphenylene
E = 79.6 kcal/mol

**Figure 5.** Minimum energy structures found by the SA Algorithm for tetramethyl polycyclic aromatic hydrocarbons. For each generated structure, the potential energy was minimized using the technique described in the text.

potential energy distribution was obtained in 114 285 s CPU time on a SGI Personal Iris Workstation.[4] The lowest energy structure found by the deterministic algorithm was 1-methyl-5-methylidenecyclohexa-1,3-diene with an energy of 0.57 kcal/mol. Samples containing 50 and 500 random structures were also constructed. While the samples gave a fairly good approximation of the potential energy distribution, the lowest energy structure was not present in the samples. For comparison, the SA algorithm was asked to search for the lowest potential energy structure of the isomers of $C_8H_{10}$. The SA run was composed of five cycles of the annealing schedule ($f_8$, $(kT)_o = 100$) each containing 100 steps. The search space was not restricted, and all possible bonds were considered by the SA random displacement. For each structure generated, the potential energy was minimized as previously described. As depicted in Figure 6, the lowest energy structure (i.e., 1-methyl-5-methylidenecyclohexa-1,3-diene) was found several times with the SA algorithm. The run lasted only 1112 s CPU time on a SGI Personal Iris Workstation, which is more than 100 times faster than the deterministic run. Furthermore, the first occurrence of the lowest energy structure appeared after running the SA algorithm for only 45 s CPU time.

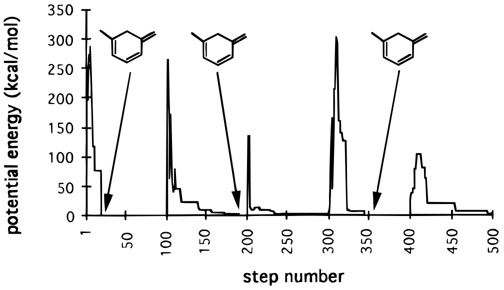

**Figure 6.** Minimum potential energy for the constitutional isomers of $C_8H_{10}$ as calculated by the SA algorithm. The lowest energy structure is 1-methyl-5-methylidenecyclohexa-1,3-diene with a potential energy of 0.57 kcal/mol. This structure occurs for the first time at the step number 20.

## CONCLUSION

The proposed SA algorithm statistically searches chemical structures for any type of compound, including cyclic and cross-linked structures. The algorithm is simple to implement. Both theoretically and practically, the algorithm was proven to run in a polynomial time. More importantly, it was shown that the algorithm performs remarkably well, even in very large constitutional spaces. This is not surprising considering the ability of SA to resolve complex combinatorial problems. For instance, SA has been used in the past to solve traveling salesman problems containing up to 6000 cities.[27] The present paper also demonstrated that the covering of the constitutional space (cf. Figures 2 and 3) or the search for chemical structures having specific properties (cf. Figures 4, 5, and 6 and Tables 6 and 8) can be achieved without performing isomorphism check. Considering the computational complexity of isomorphism, the proposed SA algorithm provides for CAMD inverse problems a computationally efficient solution. However, the stochastic nature of the proposed algorithm does not guarantee to always find the best solution to a given CAMD problem. Hence, when used for CAMD purposes, it is best to run the SA algorithm several times with different random number seeds and to output a ensemble of compounds rather than a unique solution.

The applications presented in this paper are of interest mainly in chemical information. However, there are many applications in structural elucidation and molecular design that could make use of an efficient SA algorithm. For example the SA algorithm may be coupled with MS or NMR spectra simulation codes to elucidate unknown chemical compounds. In such a case, the cost function is the difference between the actual spectrum and the simulated spectrum. Since many relationships have been established between topological indices and physical properties,[32] the results obtained with the Wiener index can be utilized directly to predict molecules having desired properties, such as boiling point. Topological indices and group contribution methods have already been used to predict the relationships between the molecular structures and the pharmaceutical activities (anesthetic, antiviral, hallucinogenic, ...) of organic compounds.[45] Hence, the proposed SA algorithm can be utilized to predict drug structures that maximize or minimize a given activity. Furthermore, the SA algorithm can be coupled with docking programs and be set up to search for drug structures that best fit a given receptor, using for example, a root mean square displacement cost function.

Structural elucidation and molecular design problems are computationally intractable; large problems cannot be solved using deterministic techniques. Heuristics or stochastic techniques are the only alternatives. While heuristics requires expert knowledge and have to be designed case by case, stochastic solutions have the advantage of being simple and general. The algorithm proposed in the present paper is an efficient (i.e., polynomial) solution to the problem of searching for constitutional isomers possessing desired properties or activities. The algorithm appears to be the first to be able to search chemical structures without restricting the search space.

## ACKNOWLEDGMENT

I am pleased to acknowledge the funding provided by the U.S. Department of Energy, Sandia National Laboratories under contract DE-AC04-94AL85000.

## REFERENCES AND NOTES

(1) For a review, see: Lipkowitz, K. B.; Boyd, D. B. *Reviews in Computational Chemistry;* VCH Publishers: New York, 1990; Vol. 1−6.

(2) For a review, see Chapter 11: Trinajstić, N. *Chemical Graph Theory*, 2nd ed.; CRC: Boca Raton, FL, 1992.

(3) For a review, see: Barnard, J. M. Substructure Searching Methods: Old and New. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 532−538.

(4) Faulon, J. L. Stochastic Generator of Chemical Structure. 1. Application to the Structure Elucidation of Large Molecules. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 1204−1218.

(5) Garey, M. R.; Johnson, D. S. *Computers and Intractability;* Freeman: San Fransisco, 1979.

(6) Leach, A. R. A Survey of Methods for Searching the Conformational Space of Small and Medium-Sized Molecules. In *Reviews in Computational Chemistry*, Lipkowitz, K. B., Boyd, D. B., Eds.; VCH Publishers: New York, 1990; Vol. 2, pp 1−56. See also references therein.

(7) Razinger, M.; Balasubramanian, K; Munk, M. E. Graph Automorphism Perception Algorithms in Computer-Enhanced Structure Elucidation. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 197−201.

(8) Smellie, A.; Kahn, S. D.; Teig S. L. Analysis of Conformation Coverage. 1. Validation and Estimation of Coverage. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 285−294.

(9) Saunders, M. Stochastic Exploration of Molecular Mechanics Energy: Hunting for the Global Minimum. *J. Am. Chem. Soc.* **1987**, 109, 3151−3154.

(10) Tufféry, P.; Etchebest C.; Hazout, S.; Lavery R. A Critical Comparison of Search Algorithms Applied to the Optimization of Protein Side-Cain Conformations. *J. Comput. Chem.* **1993**, *14*, 790−798.

(11) Judson, R. S.; Jaeger, E. P.; Treasurywala, A. M.; Peterson, M. L. Conformational Searching Methods for Small Molecules. II. Genetic Algorithm Approach. *J. Comput. Chem.* **1993**, *14*, 1407−1414.

(12) For counting problems, see: Yeh, C. Y. Isomer Enumeration of Alkanes, Labeled Alkanes, and Monosubstituted Alkanes. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 912−913.

(13) For structure generator, see: Conteras, M. L.; Rozas, R.; Valdivia, R.; Agüero, R. Exhaustive Generation of Organic Isomers. 4. Acyclic Stereoisomers with One or More Chiral Carbon Atoms. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 752−758.

(14) For ismorphism and automorphism, see: Balasubramian, K. Graph Theoretical Perception of Molecular Symmetry. *Chem. Phys. Lett.* **1995**, *232*, 415−423.

(15) 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.

(16) Brignole, E. A.; Bottini, S.; Gani, R. A. Strategy for the Design and Selection of Solvents for Separation Processes. *Fluid Phase Equi.* **1986**, *29*, 125−132.

(17) Gani, R.; Nielsen, B.; Fredenslud, A. A Group Contribution Approach to Computer-Aided Molecular Design. *AIChE J.* **1991**, *37*, 1318−1332.

(18) Nilakantan, R.; Bauman, N.; Venkataraghavan, R. A method for automatic generation of novel chemical structures and its potential applications to drug discovery. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 527−530.

(19) Kier, L. B.; Lowel, H. H.; Frazer, J. F. Design of Molecules from Quantitative Structure-Activity Relationship Models. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 143−147.

(20) Skvortosova, M. L.; 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*, 630−634.

(21) Venkatasubramanian, V.; Chan, K.; Caruthers, J. M. Evolutionary Design of Molecules with Desired Properties Using the Genetic Algorithm. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 188−195.

(22) Sheridan, R. P.; Kearsley, S. K. Using the Genetic Algorithm To Suggest Combinatorial Libraries. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 310−320.

(23) Pospichal, J.; Kvasnicka., V. Fast Evaluation of Chemical Distance by Simulated-Annealing Algorithm. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 879−885.

(24) Brown, R. D.; Jones, G.; Willet, P. Matching Two Dimensional Chemical Graphs Using the Genetic Algorithm. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 63−70.

(25) Holland, J. H. *Adapdation in Natural and Artificial Systems*; The University of Michigan Press: Ann Arbor, 1975.

(26) Metropolis, N.; Rosenbluth, A. W. Equation of State Calculation by Fast Computing Machines. *J. Chem. Phys.* **1953**, *21*, 1087−1092.

(27) Kirkpatrick, S.; Gelatt, C. D., Jr.; Vecchi, M. P. Optimization by Simulated Annealing. *Science* **1983**, *220*, 671−680.

(28) Faulon, J. L. On Using Graph-Equivalent Classes for the Structure Elucidation of Large Molecules. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 338−348.

(29) Faulon, J. L. Unraveling complex molecules. *CHEMTECH.* **1995**, *25*, 16−23.

(30) Deo, N. *Graph Theory with application to Engineering and Computer Science;* Prentice Hall: New York, 1974.

(31) The value 14.3 ± 0.9 is only valid for the tested structures (paraffins containing from 5 to 12 carbon atoms). Further calculations need to be carried out to verify if similar relationships exist for other classes of compounds.

(32) For a review, see Chapter 10: Trinajstić, N. *Chemical Graph Theory*, 2nd ed.; CRC: Boca Raton, FL, 1992.

(33) Wiener, H. Correlation of Heat of Isomerization and Difference in Heat of Vaporization of Isomers, Among Paraffin Hydrocarbons. *J. Am. Chem. Soc.* **1947**, *69*, 2636.

(34) Needdham, D. E.; Wei, I. C.; Seybold, P. G. Molecular Modeling of the Physical Properties of the Alkanes. *J. Am. Chem. Soc.* **1988**, *110*, 4186−4194.

(35) Mendiratta, S.; Madan, A. K. Structure-Activity Study on Antiviral 5-Vinylpyrimidine Nucleoside Analogs Using Wiener's Topological Index. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 867−871.

(36) Stein, S. E.; Brown, R. L. Estimation of Normal Boiling Points from Group Contribution. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 581−587.

(37) Leo, A. Calculating log $P_{oct}$ from Structure. *Chem. Rev.* **1993**, *30*, 1283.

(38) Klopman, G.; Li, J. Y.; Wang, S.; Dimayuga, M. Computer Automated log *P* Calculations Based on an Extented Group Contribution Approach. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 752−781.

(39) Watanabe, T.; Matsuhashi, K.; Takayama, S. Placental and Blood-Brain Barrier Transfert Following Prenatal and Postnatal Exposure to Neuroactive Drugs: Relationship with Partition Coefficient and Behavioral Teratogenesis. *Appl. Pharma.* **1990**, *105*, 66−74.

(40) Budzinski, H.; Garrigues, P.; Radke, M.; Connan, J.; Rayez, J. C.; Rayez, M. T. Use of Molecular Modeling as a Tool to Evaluate Thermodynamic Stability of Alkylated Polycyclic Aromatic Hydrocarbons. *Energy & Fuels* **1993**, *7*, 505−511.

(41) Cf. Table 8, Chap. 11 in: Trinajstic, N. *Chemical Graph Theory*, 2nd ed.; CRC: Boca Raton, FL, 1992.

(42) Mayo, S. L.; Olafson, B. D.; Goddard, W. A. DREIDING: A Generic Force Field for Molecular Simulations. *J. Phys. Chem.* **1990**, *94*, 8897−8905.

(43) Cox, J. D.; Pilcher, G. *Thermochemistry of Organic and Organometallic Compounds*; Academic Press: New York, 1970.

(44) Froster, P. G.; Alexander, R.; Kagi, R. I. Identification and Analysis of Tetramethylnaphthalenes in Petroleum; *J. Chromatogr.* **1989**, *483*, 384−389.

(45) For a review, see Chapter 7: Keir, L. B.; Hall, L. H. *Molecular Connectivity in Structure-Activity Analysis*; John Wiley: London, 1986.

CI950179A