Ultrafast Algorithm for Designing Focused Combinational Arrays

Dimitris K. Agrafiotis* and Victor S. Lobanov

3-Dimensional Pharmaceuticals, Inc., 665 Stockton Drive, Exton, Pennsylvania 19341

Received April 19, 2000

A novel greedy algorithm for the design of focused combinatorial arrays is presented. The method is applicable when the objective function is decomposable to individual molecular contributions and makes use of a heuristic that allows the independent evaluation and ranking of candidate reagents in each variation site in the combinatorial library. The algorithm is extremely fast and convergent and produces solutions that are comparable to and often better than those derived from the substantially more elaborate and computationally intensive stochastic sampling techniques. Typical examples of design objectives that are amendable to this approach include maximum similarity to a known lead (or set of leads), maximum predicted activity according to some structure−activity or receptor binding model, containment within certain molecular property bounds, and many others.

I. INTRODUCTION

Historically, drug discovery has been based on a serial and systematic modification of chemical structure guided by the "similar property principle",[1] i.e., the assumption that structurally similar compounds tend to exhibit similar physicochemical and biological properties. New therapeutic agents are typically generated by identifying a lead compound and creating variants of that compound in a systematic and directed fashion. The first phase of this process, known as *lead generation*, is carried out by random screening of large compound collections, such as natural product libraries, corporate banks, etc. The second, known as *lead optimization*, represents the rate-limiting step in drug discovery and involves the elaboration of sufficient structure−activity relationship (SAR) around a lead compound and the refinement of its pharmacological profile. Prior to the arrival of combinatorial chemistry, this process involved a simple prioritization of synthetic targets based on preexisting structure−activity data, synthetic feasibility, experience, and intuition.

However, revolutionary advances in synthetic and screening technology have recently enabled the simultaneous synthesis and biological evaluation of large chemical libraries containing hundreds to tens of thousands of compounds.[2] With the expansion of our knowledge base of solid- and solution-phase chemistry and the continuous improvement of the underlying robotic hardware, combinatorial chemistry has moved beyond its traditional role as a source of compounds for mass screening and is now routinely employed in lead optimization and SAR refinement. This has led to the conceptual division of combinatorial libraries into (1) *exploratory* or *universal* libraries which are target-independent and are designed to span a wide range of physicochemical and structural characteristics and (2) *focused* or *directed* libraries which are biased toward a specific target, structural class, or known pharmacophore.[3]

* Corresponding author. Telephone: (610) 458−6045. Fax: (610) 458−8249. E-mail: dimitris@3dp.com.

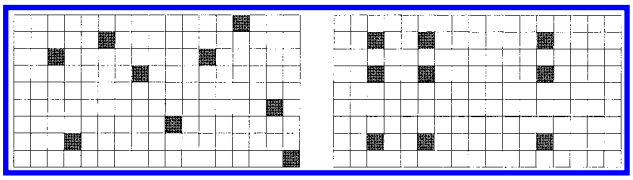

Figure 1. Selection of compounds from a hypothetical two-component combinatorial library in (a, left) singles and (b, right) array format.

Two different methods are known for designing combinatorial experiments.[4,5] The first is called "singles" or "sparse array" and refers to a subset of products that may or may not represent all possible combinations of a given set of reagents. The second is called a "full array" or simply "array" and represents all the products derived by combining a given subset of reagents in all possible combinations as prescribed by the reaction scheme. These two types of designs are illustrated in Figure 1.

The combinatorial nature of the two problems is vastly different. For singles, the number of possibilities that one has to consider (the number of different k-subsets of an n-set) is given by the binomial

$$C_s = n!/(n-k)!k! \tag{1}$$

In contrast, the number of different $k_1 \times k_2 \times ... \times k_R$ arrays (or full arrays) derived from an $n_1 \times n_2 \times ... \times n_R$ R-component combinatorial library is given by

$$C_a = \prod_{i=1}^{R} \frac{n_i!}{(n_i - k_i)!k_i!} \tag{2}$$

For a 10 × 10 two-component combinatorial library, there are 10^{25} different subsets of 25 compounds and only 63 504 different 5 × 5 arrays. For a 100 × 100 library and a 100/10 × 10 selection, those numbers increase to 10^{241} and 10^{26} for singles and arrays, respectively. Note that, in this context, the term "array" is basically equivalent to reagent selection based on the properties of the products and does not

10.1021/ci000038v CCC: $19.00
Published on Web 06/16/2000

necessarily refer to the physical layout and execution of the experiment. Although arrays are generally inferior in terms of meeting the design objectives, they require fewer reagents and are much easier to synthesize in practice.

Here we present a new algorithm for designing focused combinatorial arrays. This algorithm capitalizes on the presence of optimal substructure when the objective function is decomposable to individual molecular contributions and allows the selection of optimal or nearly optimal arrays within a fraction of a second on a modern personal computer. The method can be used to design combinatorial arrays based on maximum similarity to a known lead (or set of leads) or maximum predicted activity according to some QSAR, pharmacophore, or receptor binding hypothesis. It is not particularly useful in situations where the quality of the final design depends on how two molecules in the selected set depend on each other, as is the case with virtually all diversity scores, property distribution scores, etc.

II. METHODS

Selection Criteria. Two types of selection criteria are considered in this work: maximum similarity to a given set of leads and maximum fit to a prescribed set of property constraints. A third criterion that can be used to produce arrays that are predicted to be maximally active and/or selective against a particular biological target is also described, but its application is a straightforward extension of the similarity-based selection and will not be explicitly addressed in Results and Discussion. These criteria may be used individually or in combination and are defined as follows.

A. Similarity. The similarity of a given set of compounds, *C*, to a set of leads is defined as the average distance of a compound to its nearest lead:

$$S(C) = -\frac{1}{N}\sum_{i=1}^{N}\min_{j=1}^{L}(d_{ij}) \tag{3}$$

where N is the cardinality of C, L is the number of leads, and d_{ij} is the distance between the ith compound and the jth lead in some molecular descriptor space. Since a lower similarity score indicates a collection of compounds that are more distant and therefore less similar to the leads, focused libraries are obtained by maximizing S. If the number of leads is large and d_{ij} represent Euclidean distances in some multivariate descriptor space, the innermost loop in eq 3 can be carried out using the $k-d$ tree algorithm presented in ref 6. This algorithm achieves computational efficiency by first organizing the leads in a k-dimensional tree and then performing a nearest neighbor search for each query structure using a branch-and-bound approach. For a relatively small number of dimensions, this algorithm exhibits $N \log N$ time complexity and scales favorably with the number of compounds selected.

B. Confinement. This criterion measures the degree to which the properties of a given set of compounds fit within prescribed limits and is defined as

$$P(C) = \frac{1}{N}\sum_{i}\sum_{j}\max(x_j^{\min} - x_{ij}, x_{ij} - x_j^{\max}, 0) \tag{4}$$

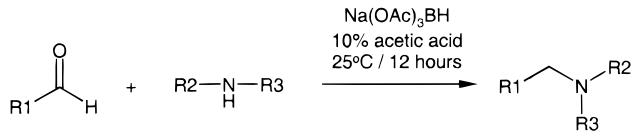

Figure 2. Synthetic sequence for the reductive amination library.

where x_{ij} is the jth property of the ith compound, and $x_j^{\min}$ and $x_j^{\max}$ are the minimum and maximum allowed limits of the jth property, respectively. Since the value of this function increases as more and more compounds fall outside the desired property range, constrained libraries are obtained by minimizing P. When multiple properties are used, they must be normalized to allow meaningful comparisons. In the special case when the properties of interest need to attain a particular target value (i.e. in the case of a degenerate range), eq 4 can be rewritten as

$$P(C) = \frac{1}{N}\sum_{i}\sum_{j}\mathrm{abs}(x_{ij} - x_j^*) \tag{5}$$

where x_j^* represents the target value of the jth property.

C. Activity/Selectivity. A common goal in library design is to produce arrays of compounds that are predicted to be maximally active against a predefined target according to some quantitative structure−activity or receptor binding model. This can be easily accomplished by expressing the predicted activity of a given set of compounds, C, as the average predicted activity of the individual compounds:

$$Q_{\mathrm{A}}(C) = \frac{1}{N}\sum_{i} a_i \tag{6}$$

where a_i is some measure of the predicted activity of the ith compound in C. A similar function can be used to measure the selectivity against a set of biological targets:

$$Q_{\mathrm{S}}(C) = \frac{1}{N}\sum_{i}(a_{ik} - \max_{j\neq k}(a_{ij})) \tag{7}$$

where a_{ij} is the predicted activity of the ith compound against the jth target and k is the target that the molecules should be selective against. Since the value of $Q_{\mathrm{A}}/Q_{\mathrm{S}}$ increases as the compounds become more active/selective, active/selective libraries are obtained by maximizing the respective criterion.

D. Multiobjective Design. The individual criteria outlined above can be combined to produce arrays that satisfy multiple design objectives. This can be encoded by a multiobjective fitness function defined as

$$F(C) = f(F_1(C), F_2(C), \ldots) \tag{8}$$

where $F(C)$ is the overall performance measure and $F_i(C)$ are individual criteria associated with the collection C. The exact form of this function and the coefficients associated with the individual criteria determine the influence of each criterion in the final design.

Selection Algorithm. The selection algorithm described in this paper is used in situations where the objective function is described as a sum of individual molecular contributions, such as eqs 3−7 above. When the compounds are selected as singles, the optimal set can be determined by evaluating the fitness of each compound (e.g. its similarity to a reference

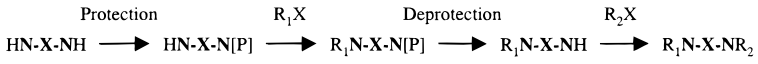

Figure 3. Synthetic sequence for the diamine library.

structure) and selecting the ones with the highest fitness. When the compounds are selected as an array, the following heuristic is employed. Given a combinatorial library of C components (substitution sites), a position $i \in [1,C]$, and a particular choice of reagents for $R_{j\neq i}$, $j = 1, 2, ..., i-1, i+1, ..., C$, the reagents for R_i that maximize the objective function can be determined by constructing and evaluating all possible subarrays derived from the combination of a single reagent from R_i with all the selected reagents for $R_{j\neq i}$ and selecting the ones with the highest fitness. The algorithm starts with a randomly chosen array and optimizes each site in sequence until no further improvement is possible.

Consider, for example, the selection of a $5 \times 5 \times 5$ array from a $10 \times 10 \times 10$ combinatorial library. The algorithm begins by selecting 5 reagents at random from each site and evaluating the fitness of the resulting array. This initial selection is refined in an iterative fashion by processing each reagent list in a strictly alternating sequence. At first, the algorithm constructs 10 $1 \times 5 \times 5$ subarrays derived by combining each reagent from R_1 with the selected reagents from R_2 and R_3 and evaluates their fitness. Every reagent at R_1 is evaluated in turn, and the 5 reagents with the highest score are selected. The process is then repeated for R_2. Each reagent in R_2 is used to construct a $5 \times 1 \times 5$ subarray derived by combining it with the selected reagents from R_1 and R_3, and the 5 R_2 reagents with the highest fitness are selected for that site. R_3 is processed in a similar fashion, and this completes a refinement cycle. Once all the reagent lists have been processed, the selected reagents from each site are combined and the fitness of the resulting full array is evaluated and compared to the fitness of the selection at the end of the previous cycle. If the fitness is improved, the refinement process is repeated starting at R_1. If not, the algorithm terminates.

Let C denote the number of substitution sites in the combinatorial library, and N_i and K_i are the total number of reagents and the size of the requested array at the ith position, respectively. The algorithm proceeds as follows:

(1) Initialize C reagent lists $\{R_{ij} = j; j = 1, 2, ..., N_i; i = 1, 2, ..., C\}$.

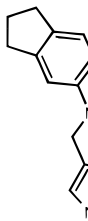

Figure 4. Reference structure used for similarity selections from the reductive amination library.

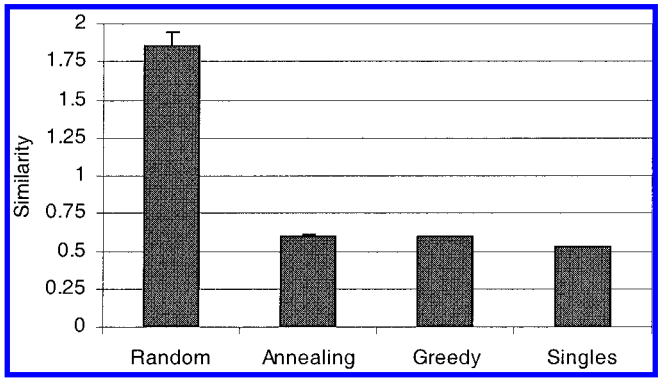

Figure 5. Mean and standard deviations of the similarity scores of 100 compounds selected from the reductive amination library according to maximum similarity to the structure in Figure 4 (10 $\times$ 10 array, 100 singles), collected over 100 optimization runs.

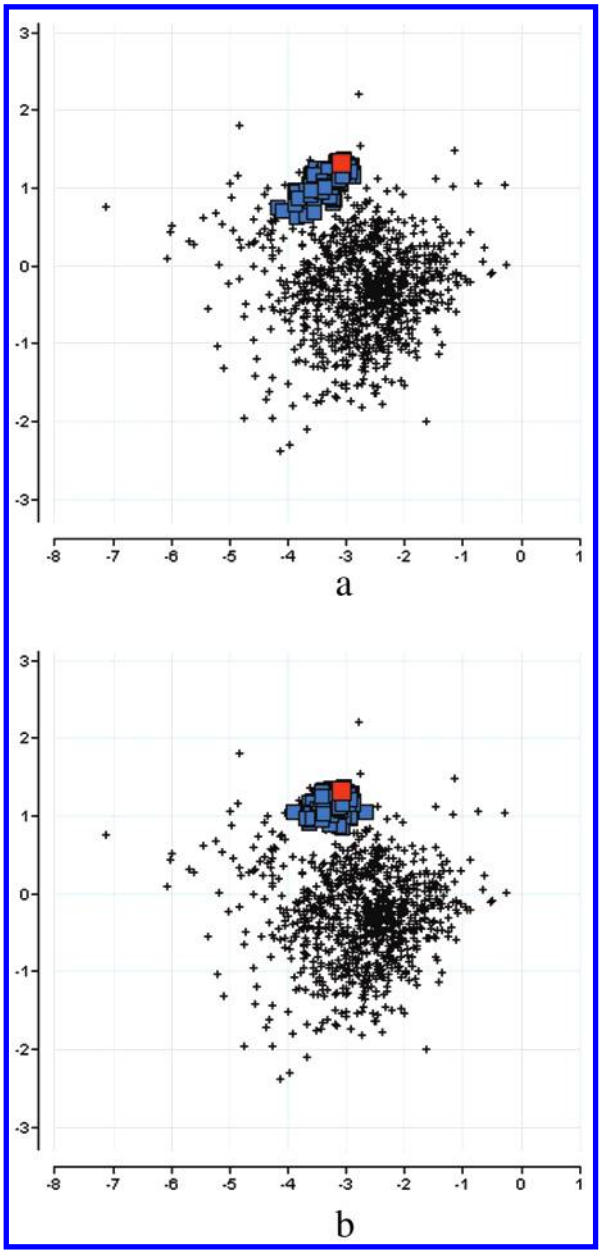

Figure 6. Distribution of compounds selected from the reductive amination library according to maximum similarity to the structure in Figure 4: (a) 10 $\times$ 10 array and (b) 100 singles. The large red square is the reference compound (lead), and the blue squares are the selected compounds, respectively. The plot is a two-dimensional nonlinear projection of the 23 principal components that account for 99% of the total variance in the data, constructed in a way that preserves the proximities (similarities) of the objects as faithfully as possible. The axes are not labeled, since they carry no physical significance.

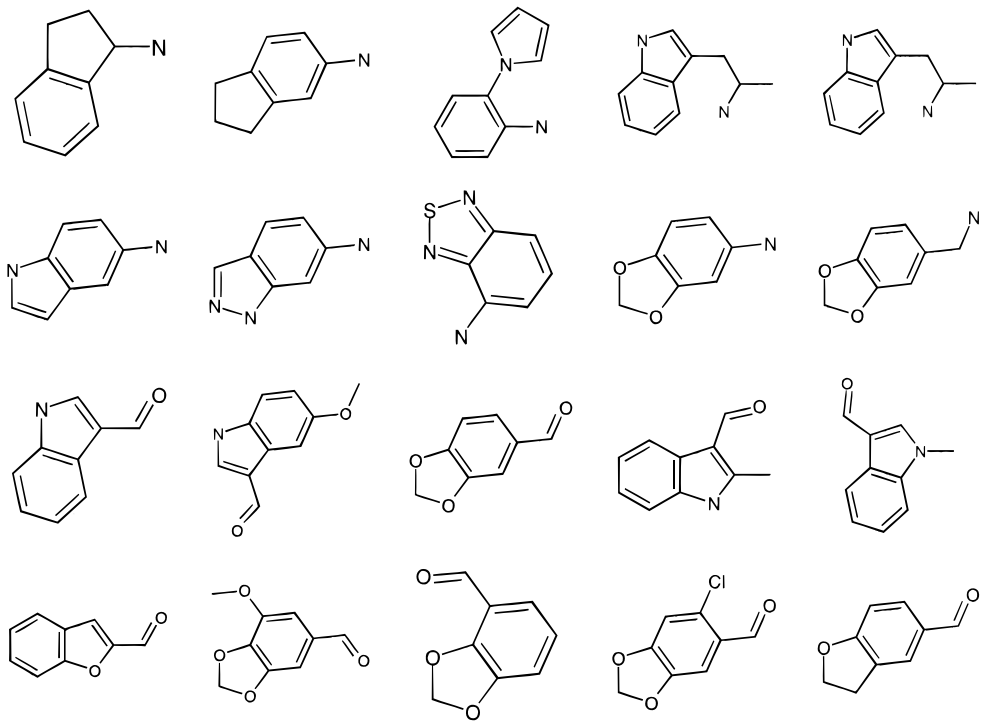

Figure 7. Reagents comprising the 10 × 10 array selected from the reductive amination library according to maximum similarity to the structure in Figure 4.

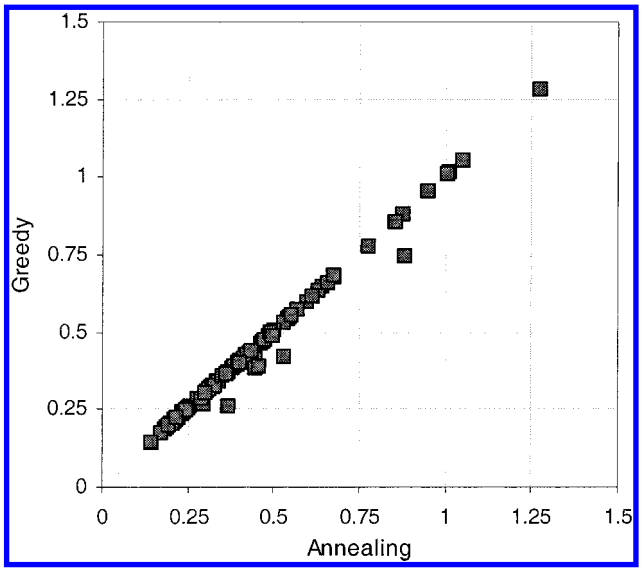

Figure 8. Comparison of the similarity scores of the best 10 × 10 arrays selected from the reductive amination library by the simulated annealing and greedy algorithms for 100 different randomly chosen leads.

(2) Randomize the order of the reagents in each reagent list, R_i.

(3) Evaluate the array $S' = S_1 \times S_2 \times ... \times S_C$; $S_i = \{R_{ij}; j = 1, 2, ..., K_i\}$, and record its fitness, s'.

(4) Perform steps 5−7 for each $i \leq C$.

(5) Perform step 6 for each $j \leq N_i$.

(6) Evaluate the array $S = S_1 \times S_2 \times S_{i-1} \times R_{ij} \times S_{i+1} \times ... \times S_C$; $S_m = \{R_{mj}; j = 1, 2, ..., K_m; m \neq i\}$, and record its fitness, f_{ij}.

(7) Sort the reagents in the ith reagent list, R_i, in descending order of fitness.

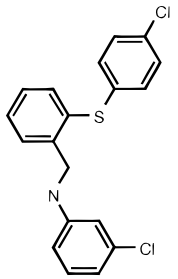

Figure 9. Reference structure used for similarity-based multiobjective selections from the reductive amination library.

(8) Evaluate the array $S = S_1 \times S_2 \times ... \times S_C$; $S_i = \{R_{ij}; j = 1, 2, ..., K_i\}$ and record its fitness, s. If $s > s'$, repeat steps 4−7.

(9) Output S.

To minimize the computational effort required, the algorithm precomputes the individual contribution to the objective function for each compound in the entire library and stores it in a working array. This step is carried out once and is $O(N)$. During the optimization, the fitness of any given array or subarray is evaluated by simply adding the precomputed individual contributions of the molecules that make up the array. Occasionally, this algorithm may get trapped into a local minimum. For this reason, the procedure is repeated a few times starting from a different random seed (i.e. a different, randomly chosen, starting array), and the selection with the best fitness is reported. Note that, with a minor modification, the algorithm can be used to select smaller subarrays from within larger arrays of a given virtual library.

Software. All programs were implemented in the C++ programming language and are part of the DirectedDiversity software suite.[3] The software is based on 3-Dimensional Pharmaceuticals' Mt++ class library[7] and was designed to run on all Posix-compliant Unix and Windows platforms.

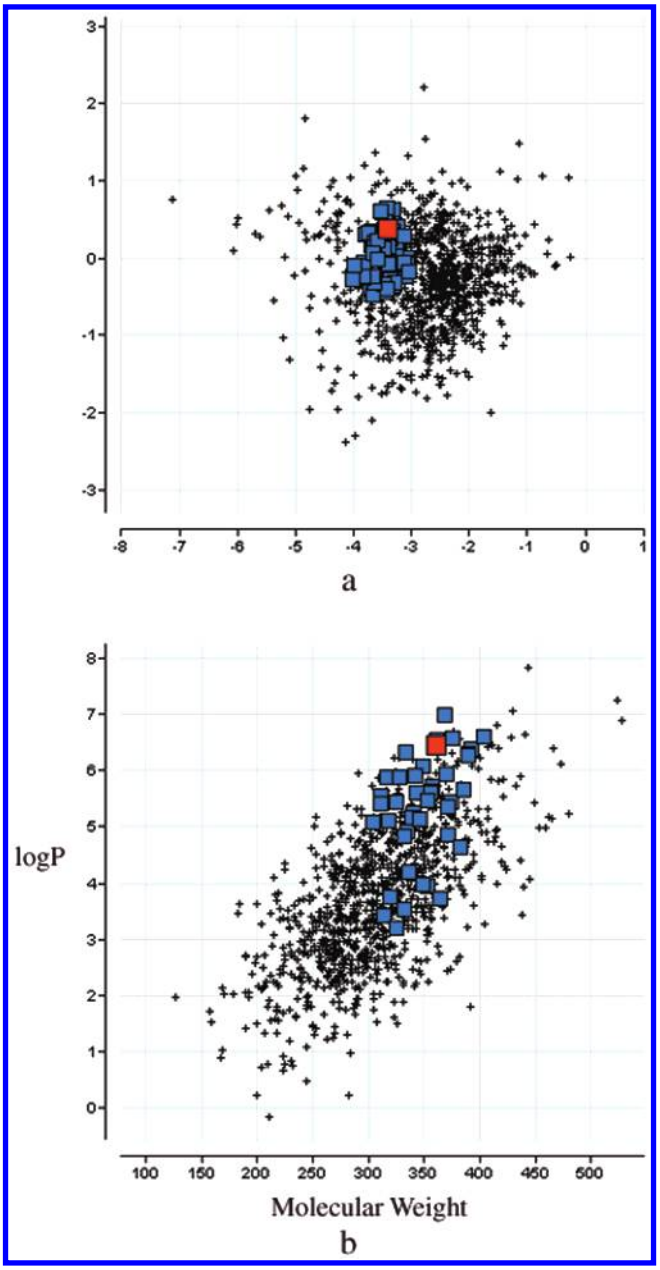

Figure 10. Distribution of compounds in the 10 × 10 array selected from the reductive amination library according to maximum similarity to the structure in Figure 9: (a) diversity space (see Figure 6 for details) and (b) molecular weight vs computed log *P*. The large red square is the reference compound (lead), and the blue squares are the selected compounds, respectively.

Parallel execution on systems with multiple CPU's is supported through the multithreading classes of Mt++. All calculations were carried out on a Gateway 2000 workstation equipped with a 400 MHz Pentium II Intel processor running Windows NT 4.0.

III. DATA SETS

The first data set used in our study is a two-component combinatorial library based on the reductive amination reaction. This library is part of a synthetic strategy that exploits the pivotal imine intermediate and is utilized for the construction of structurally diverse druglike molecules with useful pharmacological properties, particularly in the GPCR superfamily.[8] The synthetic protocol is illustrated in Figure 2. The reaction is carried out by adding a solution of primary or secondary amine to an equimolar ratio of aldehyde in 1,2-dichloroethane/*N*,*N*-dimethylformamide (DMF). Sodium triacetoxyborohydride (2 equiv) in 10% acetic acid/DMF is added to the reaction vial. Stirring of the reaction mixture at 25 °C for 12 h and subsequent addition of methanol followed by concentration yield the product in high purity.

Our analysis follows closely our previous report on the use of the Kolmogorov-Smirnov criterion for enforcing particular property distributions on combinatorial designs.[9] In particular, a set of 300 primary and secondary amines with 300 aldehydes were selected at random from the Available Chemicals Directory[10] and were used to generate a virtual library of 90 000 products using the library enumeration classes of the DirectedDiversity toolkit.[7] These classes take as input lists of reagents supplied in SDF or Smiles format, and a reaction scheme written in a proprietary language that is based on Smarts and an extension of the scripting language Tcl. All chemically feasible transformations are supported, including multiple reactive functionalities, different stoichiometries, cleavage of protecting groups, stereospecificity, and many others. The computational and storage requirements of the algorithm are minimal (even a billion-membered library can be generated in a few seconds on a personal computer) and scale linearly with the number of reagents. Although the products are encoded implicitly, individual structures are accessible at a rate of ~1 000 000 per CPU second.

Each compound in the 90 000-membered library was characterized by a standard set of 117 topological descriptors computed with the DirectedDiversity toolkit.[7] These descriptors include an established set of topological indices with a long, successful history in structure−activity correlation such as molecular connectivity indices, κ shape indices, subgraph counts, information-theoretic indices, Bonchev−Trinajstic indices, and topological state indices.[11,12] We have previously shown that these descriptors exhibit proper "neighborhood behavior"[13] and are thus well-suited for diversity analysis and similarity searching.[14]

These 117 molecular descriptors were subsequently normalized and decorrelated using principal component analysis (PCA). This process resulted in an orthogonal set of 23 latent variables, which accounted for 99% of the total variance in the data. To simplify the analysis and interpretation of results, this 23-dimensional data set was further reduced to two dimensions using a very fast nonlinear mapping algorithm developed by our group.[15,16] The projection was carried out in such a way that the pairwise distances between points in the 23-dimensional principal component space were preserved as much as possible on the two-dimensional map. The resulting map had a Kruskal stress of 0.187 and was used to visualize the selections, which were all carried out in the full 23-dimensional principal component space. The PCA preprocessing step was necessary in order to eliminate duplication and redundancy in the data, which is typical of graph-theoretic descriptors.

Finally, in addition to the 117 topological descriptors, the molecular weight and octanol−water partition coefficient (log *P*) of each compound were computed independently using the Ghose−Crippen approach,[17] as implemented in the DirectedDiversity toolkit,[7] and were used as the target

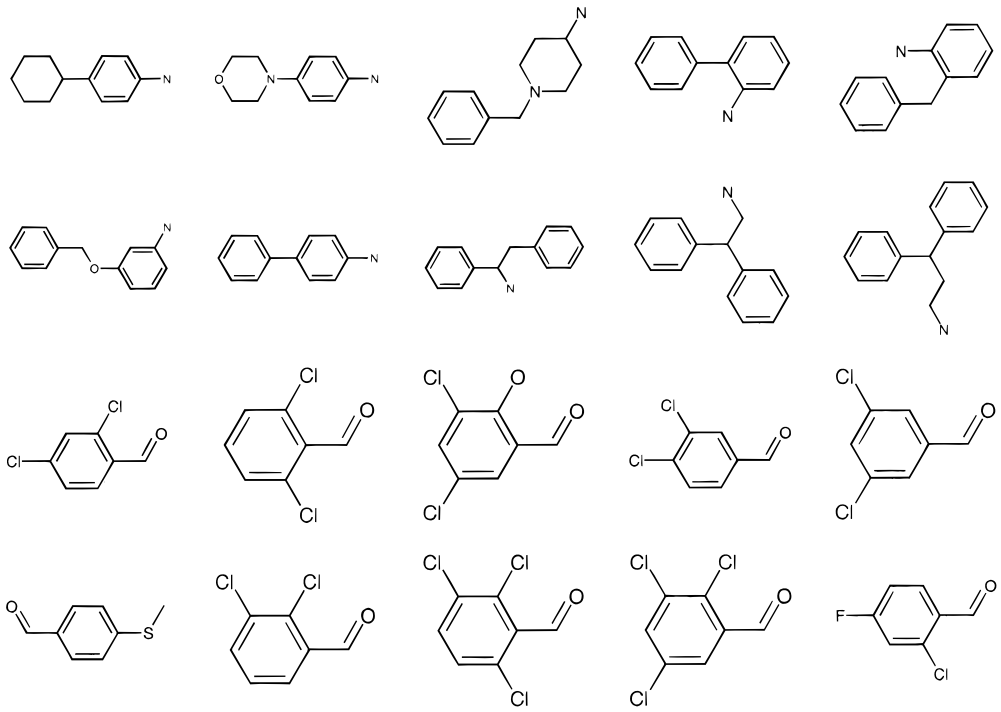

Figure 11. Reagents comprising the 10 × 10 array selected from the reductive amination library according to maximum similarity to the structure in Figure 9.

variables for all constrained designs. These parameters were not included in the descriptor set used for similarity assessment. All selections were carried out as 10 × 10 arrays.

The second example is a three-component library taken from the work of Cramer et al.[18] A diamine molecule containing two primary or secondary amines served as the central scaffold and was derivatized on both sides using an acylating agent, reactive halide, or carbonyl group susceptible to reductive amination. The synthetic sequence required to generate this library involves selective protection of one of the amines and introduction of the first side chain, followed by deprotection and introduction of the second side chain (Figure 3). As the original authors noted, use of commercially available reagents alone (the 1996 Available Chemical Directory[10] contained 1750 reagents of type HNXNH and 26 700 reagents of type RX) would yield over 10^{12} potential products. Since the objective of this work was to validate the array selection algorithms, we generated a small library comprised of 125 000 compounds using 50 commercially available diamines and 50 acid chlorides and alkylating agents. As with the previous data set, each compound was described by 117 topological indices designed to capture the essential features of the molecular graph, which were subsequently decorrelated to 22 principal components, which accounted for 99% of the total variance in the data. These principal components were used as input to our nonlinear dimensionality reduction technique[15,16] to generate a two-dimensional map that reproduced the pairwise distances of the compounds in the principal component space to a Kruskal stress of 0.167. This map was only used to visualize the designs, which were all based on all 22 decorrelated descriptors. Selections were carried out as 5 × 5 × 5 arrays.

IV. RESULTS AND DISCUSSION

To test the effectiveness of the greedy approach, we compared it against a simulated annealing algorithm that we have presented several times in the past[6,9,19,20] and have found to be very effective in the design of combinatorial libraries. This algorithm involves two main components: a search engine and an objective function. The search engine produces a list of *k* compounds from the virtual collection (also referred to as a *state*), which is subsequently evaluated by an objective function to produce a numerical estimate of its quality or fitness. The objective function encodes the design objectives of the experiment, such as the intrinsic diversity of the compounds or their similarity to a predefined set of leads. This fitness value is fed back to the search engine which modifies the state in a controlled manner and produces a new list of compounds, which is, in turn, evaluated against the selection criteria in the manner described above. This process is repeated until no further improvement is possible, or until some predetermined convergence criterion or time limit is met. The major advantage of this approach is that the search algorithm is completely independent of the performance measure,[9,19,20] and can be applied on a wide variety of selection criteria and fitness functions. In the case at hand, the selection was carried out in 30 temperature cycles, using 1000 sampling steps per cycle, a Gaussian cooling schedule, and the Metropolis acceptance criterion ($p = e^{-\Delta E/K_B T}$). In a singles selection, a step represents the substitution of a few compounds comprising the current state, whereas in an array selection a step represents the substitution of a single reagent in the combinatorial array. Boltzmann's "constant", K_B, was adjusted in an adaptive manner, by constantly updating the mean transition energy during the course of the simulation and continuously adjusting the value of K_B so that the acceptance probability for a mean uphill transition at the final temperature was 0.001. Details of this algorithm can be found elsewhere.[9,19,20]

Notice that the combinatorial nature of the problem does not permit an exhaustive enumeration of every possible array in either one of these collections. For the reductive amination

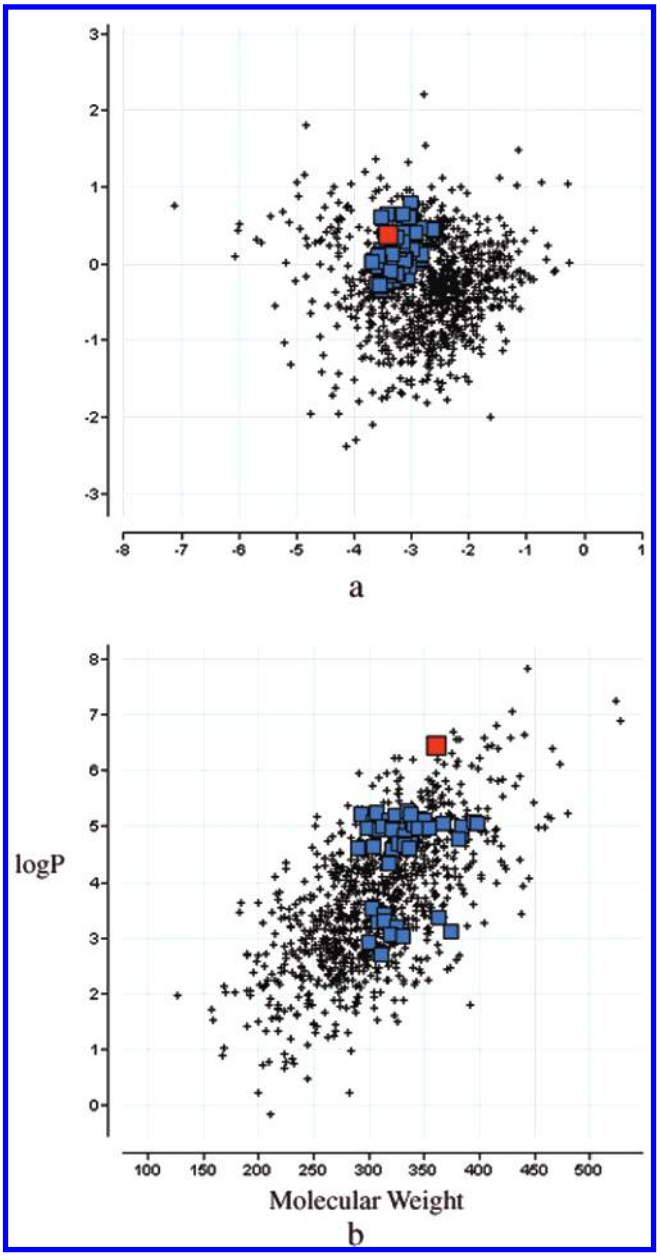

Figure 12. Distribution of compounds in the 10 × 10 array selected from the reductive amination library according to eq 9 and the structure in Figure 9: (a) diversity space and (b) molecular weight vs computed log *P*. The large red square is the reference compound (lead), and the blue squares are the selected compounds, respectively.

library, the number of different 10 × 10 arrays as determined by eq 2 is ~2 × 10^{36}, while for the diamine library the number of different 5 × 5 × 5 arrays is ~2 × 10^{19}. In the absence of a known global minimum, three reference points were used to assess the quality of the solutions produced by the greedy algorithm. The first is the fitness of a singles selection of an equivalent number of compounds and represents the upper bound of the objective function for a given selection size. Since an array is a singles selection with additional constraints, it can never exceed the quality of a pure singles design. The singles are selected in a deterministic manner by evaluating the similarity of each compound to its nearest lead, sorting the compounds in descending order of similarity and selecting the *k* topmost compounds from that list. The other two reference points are the average fitness of a random array and the fitness of the array derived from the annealing optimization algorithm described above.

For the reductive amination library, the similarity selections were limited to 100 compounds (100 singles, 10 × 10 arrays) and were carried out using the similarity fitness function in eq 3 and a randomly chosen member of that library as a "lead" (Figure 4). Since none of the algorithms are completely deterministic, the selections were repeated 100 times, each time starting from a different random initial configuration (random seed). The mean and standard deviation of the resulting distributions of the fitness scores are summarized in Figure 5. The greedy algorithm proved to be superbly convergent: every single run produced the same solution which had a score of 0.594 and was the best among all the arrays discovered in this exercise. In contrast, simulated annealing converged to that solution in only 60 out of 100 trials, although in general its performance was satisfactory with 82 and 98% of the trials leading to scores less than 0.60 and 0.61, respectively, and an overall average of 0.597 ± 0.004. These solutions are considerably better than those obtained by chance (the average score of a random array was 1.855 ± 0.086) and compare very favorably to singles, which had a score of 0.533. The distribution of the selected compounds with respect to the lead is illustrated in Figure 6a, and the amine and aldehyde reagents that comprise the optimum array are shown in Figure 7. As a reference, Figure 6b shows the 100 most similar compounds selected as singles. The reader should be reminded that the selection is based on all 23 principal components and is simply highlighted on this map. There is an inevitable distortion associated with this drastic reduction in dimensionality, and this is manifested by the presence of compounds that appear to be closer to the lead than the ones selected. However, with nonlinear mapping this distortion is distributed across the entire data space, and consequently even the relatively small differences between the singles and the array are clearly evident on the map. The actual selection consists mostly of various heterocyclic analogues of the [6,5] fused ring system found in both side chains of the lead compound (Figure 7).

To determine whether the choice and position of the lead compound has any impact on the ability of the greedy algorithm to detect a good solution, we compared its performance relative to simulated annealing using 100 randomly chosen reference structures from the amination library. The experiment was conducted by choosing 100 compounds at random, selecting for each one the most similar 10 × 10 array using the greedy and annealing algorithms, and comparing the scores of the resulting designs. In 13 out of 100 cases simulated annealing produced an inferior array, while in the remaining 87 cases the two algorithms gave identical results. As shown by the outliers in Figure 8, in five of these cases, the difference between the two techniques was noticeable. Thus, although we cannot be sure that the optimum solution has been identified, it is apparent that the greedy approach compares very favorably with the more elaborate and computationally demanding annealing algorithm.

The previous example illustrates the use of the greedy algorithm with a function involving a single objective (similarity). In fact, this algorithm works equally well with any objective function that can be described as a sum of individual molecular contributions. Consider, for example,

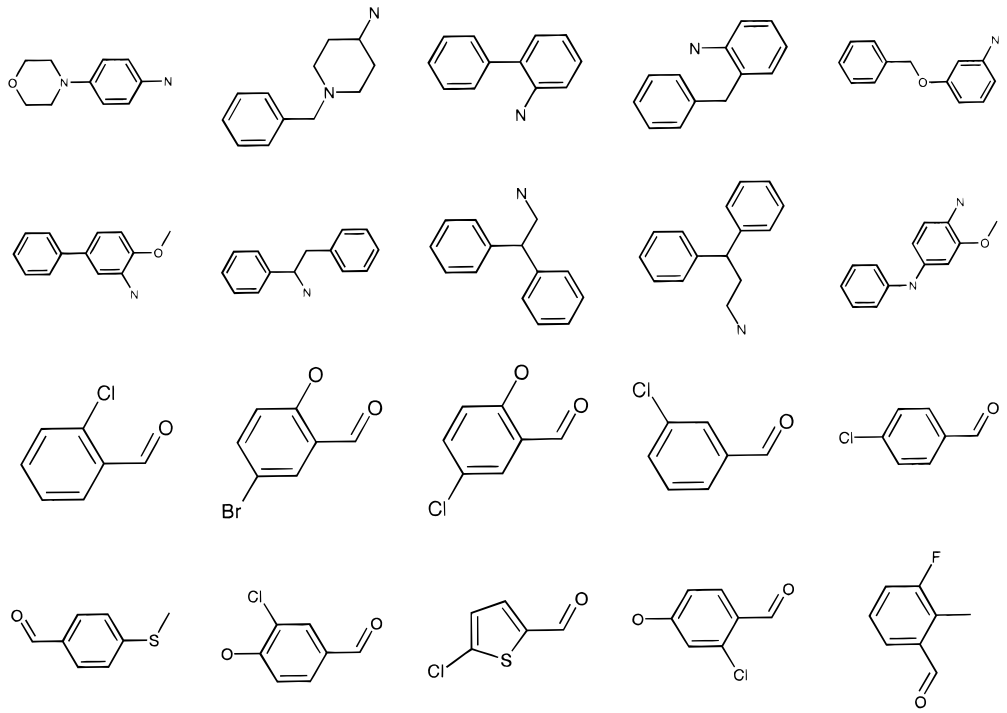

Figure 13. Reagents comprising the 10 × 10 array selected from the reductive amination library according to eq 9 and the structure in Figure 9.

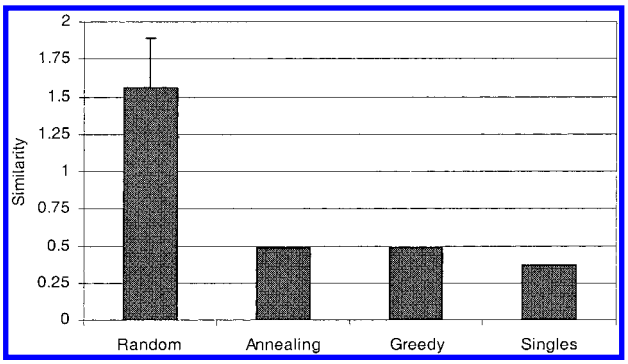

Figure 14. Mean and standard deviations of the similarity scores of 125 compounds selected from the diamine library according to maximum similarity to a randomly chosen structure (5 × 5 × 5 array, 125 singles), collected over 100 optimization runs.

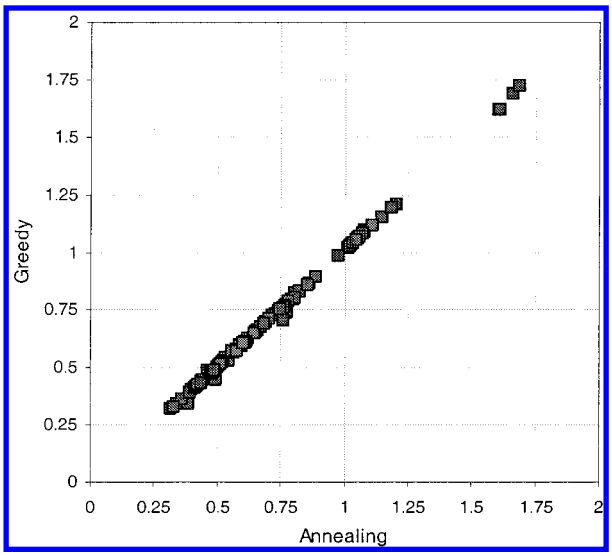

Figure 15. Comparison of the similarity scores of the best 5 × 5 × 5 arrays selected by the simulated annealing and greedy algorithms according to maximum similarity to 100 randomly chosen leads from the diamine library.

the structure in Figure 9. This compound has a molecular weight of 360.3 and a predicted log $P = 6.46$ and falls beyond the boundaries defined by the Lipinski "rule of 5".[21] This rule was derived from an analysis of the World Drug Index and states that for compounds which are not substrates of biological transporters poor absorption and permeation are more likely to occur when there are more than 5 H-bond donors, more than 10 H-bond acceptors, the molecular weight is greater than 500, or the log P is greater than 5. As shown in Figure 10, a 10 × 10 array selected based on maximum similarity to this "lead" consists of compounds with an average log $P = 5.47 \pm 0.85$, with 77 of these compounds having a log $P > 5$. The selected reagents are shown in Figure 11 and consist of halogenated benzaldehydes and hydrophobic amines containing two six-membered rings separated by a small linker. To reduce the hydrophobic character of these compounds while preserving the overall structural similarity, one can combine the molecular similarity criterion in eq 3 with the confinement criterion in eq 4 defined over the boundaries of the Lipinski box (log $P \leq 5$, MW $\leq$ 500). Figure 12 shows the selection of 100 compounds in a 10 × 10 format which minimized the objective function

$$f(C) = S(C) + 2P(C) \qquad (9)$$

where $S(C)$ and $P(C)$ are given by eq 3 and eq 4, respectively. The weights determine the relative influence of each criterion in the final design and were chosen on the basis of a simple scheme described in ref 9. The resulting array consists of compounds with an average log $P = 4.58 \pm 0.72$ and includes the reagents shown in Figure 13. The two selections in Figure 11 and Figure 13 have nine reagents (eight amines and one aldehyde) in common. Two of the hydrophobic

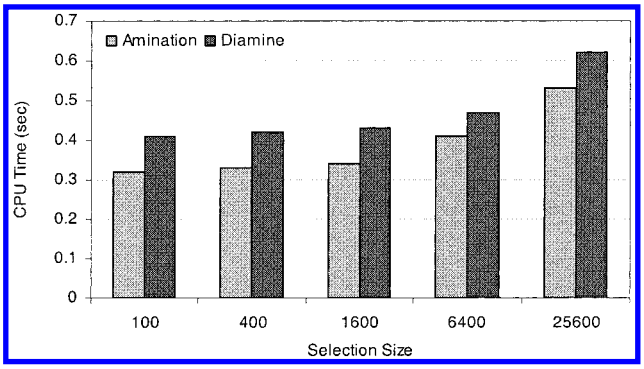

Figure 16. CPU time required for selecting a particular number of compounds using the greedy algorithm as a function of the selection size (i.e. the number of compounds requested).

amines in the original selection are replaced by topologically similar structures bearing a methoxy group, while several of the polyhalogenated benzaldehydes are replaced with reagents having a smaller number of halogen substituents and, in four of these cases, an additional hydroxy group. Thus, with a careful choice of parameters the multiobjective approach is capable of guiding a selection toward more desirable regions of chemical space without destroying the primary objective of the experiment which, in this case, was the design of a series of analogues that are closely related to a known lead.

To ensure that the performance of the algorithm does not degrade with more complex combinatorial libraries, we repeated the same type of analysis for the three-component diamine library described in the beginning of this section. The results are summarized in Figures 14 and 15. Once again, the greedy algorithm was extremely robust, resulting in three distinct but very similar solutions having an average fitness of 0.4833 ± 0.0001; 79 out of 100 trials converged to an array with a score of 0.4832, while the remaining 21 trials converged to two different arrays with an identical score of 0.4835. Simulated annealing showed a slightly higher variability, converging to 8 different solutions with fitness values ranging from 0.4832 (the same array as that identified by the greedy algorithm, visited 21 times) to 0.5032, and an average of 0.4859 ± 0.0043. As with the amination library, this experiment was based on a randomly chosen lead from the diamine collection. The process was then repeated for 100 randomly chosen reference compounds, and the results are summarized in Figure 15. The solutions produced by the greedy algorithm were marginally better in 57 of these trials, marginally worse in 16 trials, and identical in the remaining 27 trials, with differences ranging from -0.059 to $+0.033$.

Of course, the main advantage afforded by this algorithm is the speed at which it executes. Figure 16 shows the CPU times required for the selection of 100, 400, 1600, 6400, and 25 600 compounds from the reductive amination and diamine libraries, respectively, on a 400 MHz Intel Pentium II processor. The algorithm has trivial computational requirements and even the largest selections are completed in subsecond time frames. A significant proportion of this time is spent computing the Euclidean distances of each candidate to the respective lead during the preprocessing step, which scales linearly with the size of the virtual library. The refinement algorithm itself scales linearly with respect to the total number of reagents and produces solutions that are comparable to and often better than those derived from more elaborate stochastic approaches.

V. CONCLUSIONS

We have outlined a new algorithm for the design of focused combinatorial libraries in array format. The algorithm is applicable in situations where the objective function can be described as a sum of individual molecular contributions and makes use of a heuristic that allows the independent evaluation and ranking of candidate reagents in each variation site on the combinatorial library. It is extremely fast and convergent and produces arrays that are comparable to and often better than those derived from the substantially more intensive stochastic sampling techniques. The method involves a preprocessing step which scales linearly with the size of the virtual library and a refinement step which scales linearly with the total number of reagents. This optimization scheme makes possible the selection of large focused combinatorial arrays in subsecond time frames on modern midrange personal computers.

ACKNOWLEDGMENT

We wish to thank Dr. Raymond F. Salemme of 3-Dimensional Pharmaceuticals, Inc., for his insightful comments and support of this work.

REFERENCES AND NOTES

(1) Johnson, M. A.; Maggiora, G. M. *Concepts and Applications of Molecular Similarity*; Wiley: New York, 1990.
(2) Thompson, L. A.; Ellman, J. A. *Chem. Rev.* **1996**, *96*, 555−600.
(3) Agrafiotis, D. K.; Bone, R. F.; Salemme, F. R.; Soll, R. M. United States Patents 5,463,564, **1995**; 5,574,656, **1996**; 5,684,711, **1997**; and 5,901,069, **1999**.
(4) Agrafiotis, D. K. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer, H. F., III, Schreiner, P. R., Eds.; Wiley: New York, 1998; pp 742−761.
(5) Agrafiotis, D. K.; Myslik, J. C.; Salemme, F. R. *Mol. Diversity* **1999**, *4* (1), 1−22. Agrafiotis, D. K.; Myslik, J. C.; Salemme, F. R. In *Annual Reports in Combinatorial Chemistry and Molecular Diversity*; Pavia, M., Moos, W., Eds.; Kluwer: Dordrecht, The Netherlands, 1999; Vol. 2, 71−92.
(6) Agrafiotis, D. K.; Lobanov, V. S. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 51−58.
(7) *The Mt Toolkit: An Object-Oriented C++ Class Library for Molecular Simulations*; 3-Dimensional Pharmaceuticals, Inc.: Exton, PA, 1994−2000.
(8) Dhanoa, D. S.; Gupta, V.; Sapienza, A.; Soll, R. M. Poster 26, American Chemical Society National Meeting, Anaheim, CA, 1999.
(9) Rassokhin, D. N.; Agrafiotis, D. K. *J. Mol. Graphics Modell.*, in press.
(10) *Available Chemicals Directory*; MDL Information Systems, Inc.: San Leandro, CA.
(11) Hall L. H.; Kier, L. B. The Molecular Connectivity Chi Indexes and Kappa Shape Indexes in Structure−Property Relations. In *Reviews of Computational Chemistry*; Boyd, D. B., Lipkowitz, K. B., Eds.; VCH: Weinheim, Germany, 1991; Chapter 9, pp 367−422.
(12) Bonchev, D.; Trinajstic, N. *J. Chem. Phys.* **1977**, *67*, 4517−4533.
(13) Patterson, D. E.; Cramer, R. D.; Ferguson, A. M.; Clark, R. D.; Weinberger, L. E. *J. Med. Chem.* **1996**, *39*, 3049−3059.
(14) Lobanov, V. S.; Agrafiotis, D. K. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 460−470.
(15) Agrafiotis, D. K. *Protein Sci.* **1997**, *6*, 287−293.
(16) Agrafiotis, D. K.; Lobanov, V. S. *J. Chem. Inf. Comput. Sci.*, in press.
(17) Ghose, A. K.; Viswanadhan, V. N.; Wendoloski, J. J. *J. Phys. Chem. A* **1998**, *102*, 3762−3772.
(18) Cramer, R. D.; Patterson, D. E.; Clark, R. D.; Soltanshahi, F.; and Lawless, M. S. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 1010−1023.
(19) Agrafiotis, D. K. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 841−851.
(20) Agrafiotis, D. K. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 576−580.
(21) Lipinski, C. A.; Lombardo, F.; Dominy, B. W.; Feeny, P. J. *Adv. Drug Delivery Rev.* **1997**, *23*, 3−25.

CI000038V