Experimental Designs for Selecting Molecules from Large Chemical Databases

Richard E. Higgs,*,[†] Kerry G. Bemis,[†] Ian A. Watson,[‡] and James H. Wikel[‡]

Statistical and Mathematical Sciences and Technology Core Research, Lilly Research Laboratories, Eli Lilly and Company, Indianapolis, Indiana 46285

Received April 9, 1997[⊗]

Recent developments in high-throughput screening and combinatorial chemistry have generated interest in experimental design methods to select subsets of molecules from large chemical databases. In this manuscript three methods for selecting molecules from large databases are described: edge designs, spread designs, and coverage designs. Two algorithms with linear time complexity that approximate spread and coverage designs are described. These algorithms can be threaded for multiprocessor systems, are compatible with any definition of molecular distance, and may be applied to very large chemical databases. For example, ten thousand molecules were selected using the maximum dissimilarity approximation to a spread design from a sixty-dimensional simulated molecular database of one million molecules in approximately 6 h on a UNIX workstation.

INTRODUCTION

High-throughput screening programs are commonly used today for the identification of lead molecules in pharmaceutical discovery programs. The molecules are often selected from large chemical databases maintained by research pharmaceutical corporations. Molecular descriptors such as computed physical properties, molecular fragment bit strings, and molecular shape descriptors can be used to describe each molecule in the database. Using these descriptors, experimental design techniques may be used to identify subsets of molecules from these large databases for biological screening. Recent interest in combinatorial chemistry has led to the application of these methods for selecting diverse reagents for combinatorial synthesis as well as selecting molecules from existing combinatorial libraries.[1−3] Given the size of today's chemical libraries (*e.g.* hundreds of thousands to millions of compounds), the computational complexity of any selection strategy must be considered. In general, methods that require computation of all pairwise molecular similarities are not feasible for these large databases.

Two strategies for selecting diverse molecules that have been previously reported include distance and cell based methods. For example, Cummins *et al*. have described a cell based "binning" method to compare the relative diversity of molecular databases as well as to select diverse subsets of molecules.[4] Hudson *et al*.[5] describe a distance based "sphere exclusion" method to select diverse subsets of molecules with minimal computational complexity. Holliday *et al*. utilized a specific distance metric (cosine coefficient) in order to implement a computationally efficient algorithm for selecting subsets of diverse molecules.[1]

In this paper, three methods for selecting molecules from chemical databases are described: edge designs, spread designs, and coverage designs. Each design method seeks to optimize a different objective. Edge designs identify molecules at the edge of the descriptor space that produce minimum variance estimates for linear model parameters. Spread designs are used to identify a subset of molecules that are maximally dissimilar with respect to each other. Coverage designs identify a subset of molecules that are maximally similar to the candidate set of molecules. The degree of optimization is dependent on the size of the database because of the nonlinear algorithmic complexity of the optimization algorithms. Due to the large size of chemical databases and the frequently marginal improvements achieved with optimization procedures, the methods presented here represent first order approximations to the optimization of a design criterion.

A brief review of edge designs approximated by the D-optimal algorithm, which is available in commercial statistical software packages, is presented. Algorithms that approximate spread and coverage designs are described in detail along with their time complexity. In particular, a linear time order algorithm (max_diss) which approximates spread designs and a linear time order algorithm (k_cov) which approximates coverage designs are described. Both algorithms were inspired by the spread and coverage designs discussed by Tobias[6] which are included in the OPTEX procedure of the SAS system.[7] The max_diss and k_cov algorithms are computationally more efficient than those found in the OPTEX procedure and are also amenable to parallel implementations. Simulation results highlighting the differences between subsets of molecules selected with edge, spread, and coverage designs are reported. The algorithmic time complexity of the max_diss and k_cov algorithms has been confirmed on single-processor and multiprocessor systems using hypothetical chemical databases. The use of these methods is illustrated with a description of how they have been applied to augment a large molecular database with molecules from external sources.

MATERIALS AND METHODS

Finite subset selection methods begin with a database of p molecular descriptors for N molecules of which n_{ps} of the molecules have been selected for screening in previous designs. It is important to note that the design objectives are optimized relative to the N molecules embedded in a p-dimensional descriptor space and are not referenced to the entire p-dimensional space. The n_{ps} previously selected molecules are included in the database so that newly selected

[†] Statistical and Mathematical Sciences.
[‡] Technology Core Research.
[⊗] Abstract published in *Advance ACS Abstracts,* September 1, 1997.

S0095-2338(97)00285-0 CCC: $14.00

molecules will avoid the region of space containing these previously screened molecules (*i.e.* complement the set of previously screened molecules). If no molecules have been screened in a previous design, n_{ps} is set to zero.

The objective is to select n molecules to optimize a specified design criterion. The molecular descriptors can include physical parameters computed from the two-dimensional graph of the molecule, fragment bit strings describing the molecule, parameters computed from a three-dimensional representation of the molecule, or any other numerical property.

Distance Definitions. The first step in selecting subsets of molecules with a spread or coverage design is to define a molecular distance metric for the available descriptors. Distance (or similarity) is defined between pairs of molecules, between a molecule and a set of molecules, and between sets of molecules. These distance definitions are used by spread and coverage design algorithms for selecting the molecules in the subset, evaluating the quality of the design generated by different algorithms, and for the optimization process.

For p real-valued descriptors, the distance between molecules x and y can be given by the l norm of the difference of the vectors representing x and y.

$$d_l(x,y) = ||x - y||_l = (|x_1 - y_1|^l + |x_2 - y_2|^l + ... + |x_p - y_p|^l)^{1/l}$$

For fragment bit strings, the Tanimoto similarity coefficient is one example of a similarity coefficient that is widely used to define intermolecular similarity.[8]

Given a definition of distance between individual molecules, the distance between a molecule and a set of molecules, d_α, may be defined. Examples include the average distance, $d_{\alpha 1}$, and the minimum distance, $d_{\alpha 2}$, between a molecule x and a set of molecules D (where D does not include x):

$$d_{\alpha 1}(x,D) = \frac{\sum_{y \in D} d_l(x,y)}{N_D}$$

$$d_{\alpha 2}(x,D) = \min_{y \in D}\{d_l(x,y)\}$$

where N_D is the number of molecules in set D. Spread designs seek to maximize the average or minimum d_α between each molecule in the selected subset and all other molecules in the subset. Given a definition of distance between a molecule and a set of molecules, the directed distance between two sets of molecules, d_β, may be defined. Examples include the average distance, $d_{\beta 1}$, and the minimum distance, $d_{\beta 2}$, between two sets of molecules C and D:

$$d_{\beta 1}(D,C) = \frac{\sum_{x \in D} d_\alpha(x,C)}{N_D}$$

$$d_{\beta 2}(D,C) = \min_{x \in D}\{d_\alpha(x,C)\}$$

where N_D is the number of molecules in set D. Coverage designs seek to minimize the distance between the selected design subset (D) and the candidate set of molecules (C).

The methods presented here for approximating spread and coverage designs work with any valid definition of distance between molecules or sets of molecules, including the example definitions of distance listed above. Holliday *et al.* have described a linear time order algorithm approximating a spread design when the cosine coefficient is used as the distance between molecules and the sum of molecular distances is used as the distance between sets of molecules.[1] Hudson *et al.* have described a sphere exclusion method for estimating a spread design that has similar algorithmic complexity to the spread algorithms described here.[5] Hudson *et al.* have also described a method to approximate a coverage design, the most descriptive compound (MDC) method.[5] A disadvantage of the MDC method is that its algorithmic complexity is not amenable to large molecular databases. The methods for approximating spread and coverage designs presented here are designed to have linear time complexities and work with any definition of molecular distance. The flexibility to accommodate new definitions of molecular distance as well as new molecular descriptors (*e.g.* three-dimensional descriptors and new distance definitions) is important for continued improvements in these experimental design methods.

Preprocessing. Prior to selecting a set of molecules from a database it is often necessary to preprocess the molecular descriptors to replace missing descriptor values and to scale the descriptors. Although it is possible to develop distance metrics that are tolerant to missing values, we have focused on replacing missing values and using simple distance metrics that assume all descriptor values are present. We have used the following three methods to process missing descriptor values: removal of a molecule from the database, replacement by an average descriptor value, and replacement by a model derived descriptor value. The simplest procedure for eliminating missing values is to remove all molecules containing a missing descriptor value. This method has obvious drawbacks when large numbers of molecules are missing descriptor values. A better option is to replace missing descriptor values by the average value for the descriptor using all molecules with a valid descriptor value. Possibly the best computationally feasible method for replacing missing descriptor values (used for this application) is to use a predicted value from a multiple linear regression model where the descriptor containing missing values is treated as the response and the other descriptors are treated as model inputs. More complex methods of imputing missing descriptor values exist, but the focus here is on simple methods that are practical for very large chemical databases.

Following the replacement of missing values, a scaling procedure is generally required to remove the effects of descriptor scales and their correlations on the distance metric. For the work presented here, we first standardized the descriptors and then computed all standardized principal components using the standardized descriptors. Note that a Euclidean distance on this new scale is equivalent to a Mahalanobis distance on the original scale. Other scaling procedures may be employed depending on the type of descriptors (*e.g.* real values *vs* fragment bit strings) and the type of distance metric.

Edge Designs. Edge designs refer to designs that select molecules on the edge of the descriptor space. In practice this has meant the selection of molecules using D-optimal design. Molecules selected using D-optimal designs populate the edge of descriptor space by first filling in the corners and then moving around the boundary. Edge designs are selected for two different reasons. The first and most appropriate is when one intends to fit a linear regression model where the descriptors are the predictors in the model, for example if one models biological activity as a linear function of the descriptors. This is usually the situation in lead optimization rather than lead generation. The second reason is when one attempts to modify the D-optimal algorithm in a way to achieve a space-filling design such as the spread or coverage design. This is usually achieved by forming new descriptors which are powers (usually quadratic) and cross-products (usually pairwise) of the original descriptors and then computing the D-optimal design. This tends to move points from the edge to the interior of the descriptor space and hence be more “space-filling”. This, however, is a very *ad hoc* way of trying to accomplish what a spread and coverage design can do much better. Martin *et al.*[2] describe this method and claim the resulting design has “maximal overall diversity”, although the resulting diversity is not maximal in the sense of the spread design which we feel is a more appropriate definition of maximal diversity. We will not discuss this use of D-optimal designs any further but will describe in more detail the use of edge designs when one wishes to build a model for lead optimization.

Given a drug lead, it is reasonable to believe that molecules from the candidate set that are near the lead in the descriptor space also have biological activity. It is also reasonable to assume the relationship is linear (a hyperplane in descriptor space) if we are sufficiently near the lead. A linear model could then point directly to untested molecules with predicted high activity. An efficient use of screening material would suggest that from the set of near neighbors to the lead we select a small subset that is optimal for fitting the parameters of our hypothesized model. The D-optimal design is such an optimal design. $\mathbf{X}$ denotes the descriptor matrix with N rows and p columns where N is the number of candidate molecules and p the number of descriptors. The D-optimal design finds the set of molecules that maximizes the determinant of $\mathbf{X}'\mathbf{X}$ where $\mathbf{X}'$ is the transpose of $\mathbf{X}$. The D-optimal design is optimal in the following sense:

(1) It has confidence regions for the parameters with minimal (hyper) volume.

(2) It minimizes the generalized variance of the parameter estimates.

(3) It minimizes the maximum variance of any predicted value (from the model) over the experimental space.

(4) It is invariant to linear changes of scale of the parameters.

There are many algorithms for selecting D-optimal designs. We require an algorithm that allows for preselected molecules and selects each additional molecule only once (in general a D-optimal design allows replication of design points). JMP from SAS institute has such an algorithm.[9]

Edge designs may be selected by algorithms other than D-optimal. There are also A, E, and G optimality which weight the edge points differently.[10] Modifications to D-optimal edge designs can be made to accommodate difficulty or cost of synthesis,[11] or a Bayesian modification to reduce dependency on the assumed model.[12] These methods are beyond the scope of this paper, and the references should be consulted.

Spread Designs. The objective of a spread design is to identify a subset of molecules that is maximally dissimilar given a metric defining the similarity of a set of molecules. One example metric defining the similarity of a set of molecules is to compute the average of the nearest neighbor distances for each molecule in the subset. Given a metric for the overall similarity of a subset, all subsets of size n (including the n_{ps} previously selected molecules) could be tested and the subset producing the smallest overall similarity chosen. This approach suffers from an obvious combinatorial explosion in the number of subsets that need to be tested as the size of the database and the size of the subset increase. For example, examination of all possible subsets of size n from a database of N molecules where n_{ps} molecules have been previously selected would require $(N - n_{ps})!/n!(N - n_{ps} - n)!$ subset evaluations. In practice, the magnitude of $N - n_{ps}$ and n prohibit a full scale optimization, and simple, nonoptimal sequential algorithms are often used to approximate the maximally dissimilar subset. These sequential algorithms were originally described by Kennard and Stone,[13] and one can begin by selecting the first molecule, say the most distant molecule from the centroid, selecting the second molecule farthest from the first, selecting the third molecule that is farthest from the first two (*e.g.* using the $d_{\alpha 2}$ distance between a molecule and a set of molecules), and so on. Optimization can be added to these sequential methods to add and remove molecules (*e.g* Fedorov optimization[7,14]) in order to identify a more dissimilar subset. We have focused on an efficient implementation of Kennard and Stone’s approach applied to large chemical databases and have not implemented design optimization due to the marginal design improvements and greatly increased computational time. To illustrate, the SAS OPTEX procedure (CRITERION=S)[7] was used to select 20 points from the 1400 two-dimensional points shown in Figures 1−4 using a simple sequential selection algorithm and a modified Fedorov optimization algorithm. The OPTEX procedure seeks to maximize the harmonic mean distance from each design point to all other design points for a spread design. Eighty different designs were generated using the simple sequential method and the modified Fedorov optimization method. On average, a modified Fedorov optimization generated a design that was 8.5% better, with respect to the harmonic mean of the minimum distance from each molecule to any other in the set (the design criterion maximized by OPTEX), than the sequential selection method but required eight times more computational time.

We have developed an algorithm (max_diss) to approximate a maximally diverse subset using a sequential selection method that has linear time complexity. The algorithm begins by computing and storing the nearest neighbor distance between each candidate molecule and each previously selected molecule. If all molecules in the database are candidates, then one molecule is selected either by finding the molecule most distant to the centroid (most diverse molecule) or by selecting the molecule closest to the centroid (most representative molecule) as the first molecule in the design, and the nearest neighbor distance between this molecule and all other candidate molecules is computed and stored. The candidate molecule with the largest nearest

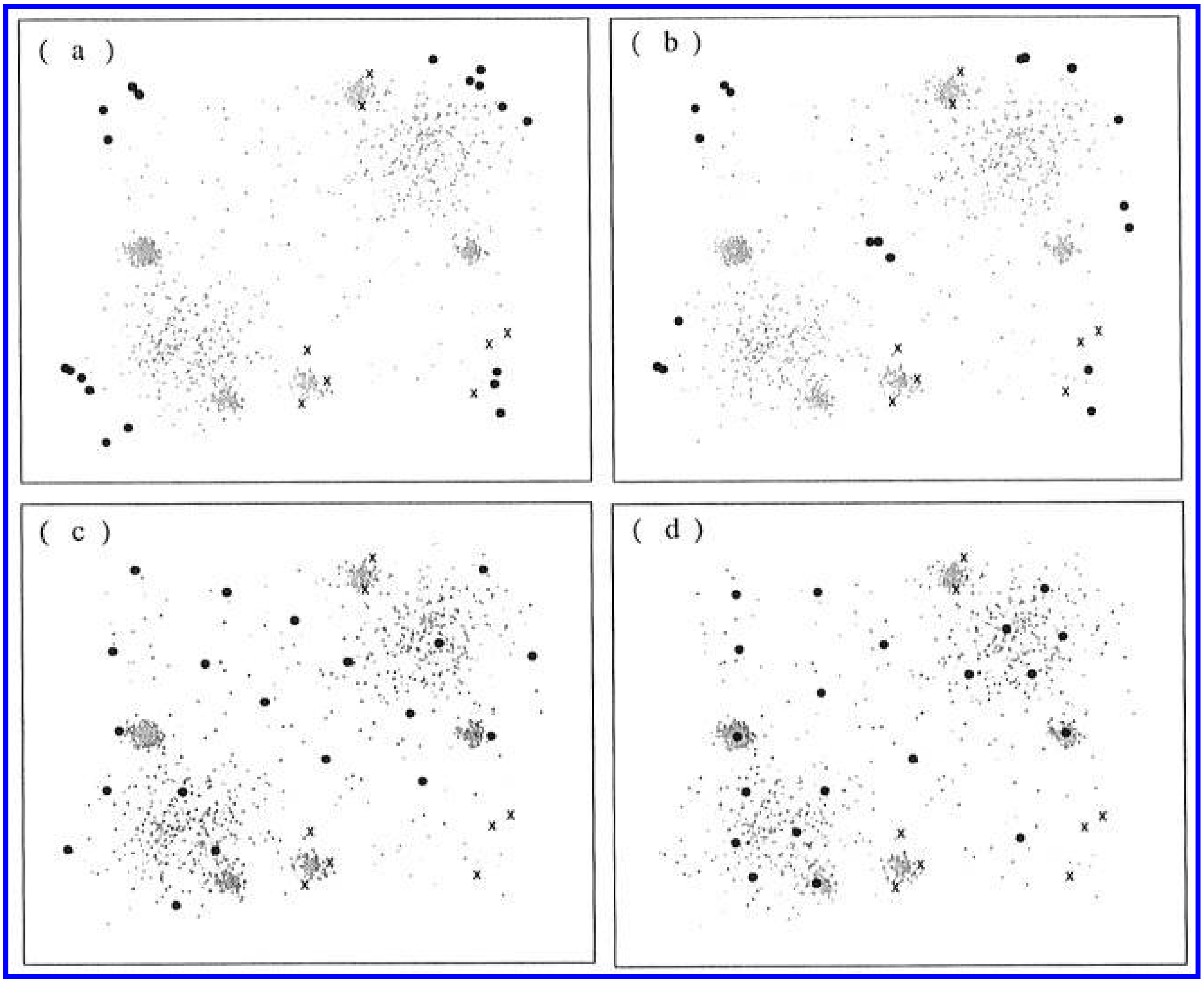

Figure 1. (a) Molecules selected (●) using D-optimal approximation to an edge design with eight previously selected molecules (×). (b) Molecules selected (●) using D-optimal approximation to an edge design with quadratic and interaction terms and eight previously selected molecules (×). (c) Molecules selected (●) using the max_diss approximation to a spread design with eight previously selected molecules (×). (d) Molecules selected (●) using the k_cov approximation to a coverage design with eight previously selected molecules (×).

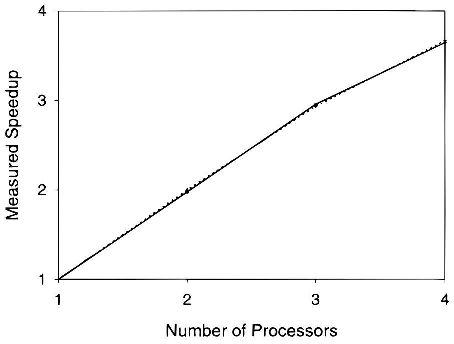

Figure 2. Measured speedup for the max_diss algorithm using a sixty-dimensional data set with N fixed at 160 000, n_{ps} fixed at 10 000, and n fixed at 5000 (dashed line). Measured speedup for the k_cov algorithm using a sixty-dimensional data set with N fixed at 91 000, n_{ps} fixed at 1000, and n fixed at 100 (solid line).

neighbor distance is then placed in the design subset and is removed from the candidate set. Next, the nearest neighbor distance for the remaining candidate molecules is checked to see if the newly selected molecule is the nearest neighbor to any candidate molecule, and, if so, the nearest neighbor distance for the candidate molecule is updated. Again, the candidate molecule having the largest nearest neighbor distance to the set of selected molecules is selected for the design, and the process repeats until n molecules have been selected. There are $(N - n_{ps})n_{ps}$ distance calculations required to select the first molecule. The second selected molecule requires $N - n_{ps} - 1$ distance calculations, the third molecule selected requires $N - n_{ps} - 2$ distance calculations, and so on. The total number ($T_{\text{max\_diss}}$) of distance calculations required for the max_diss algorithm is given by

$$T_{\text{max\_diss}} = (N - n_{ps})n_{ps} + \sum_{i=1}^{n-1}(N - n_{ps} - i)$$

which is equal to

$$T_{\text{max\_diss}} = N(n_{ps} + n - 1) + n\left(\frac{1}{2} - \frac{n}{2} - n_{ps}\right) + n_{ps} - n_{ps}^{2}$$

where n molecules are selected from a database of N molecules with n_{ps} molecules previously selected. Holliday *et al.*[1] have proposed a sequential centroid method that requires $(n - 1)N - n(n - 1)/2$ distance calculations if the cosine coefficient is used as the distance between molecules, and the sum of distances between a molecule and a set of molecules is used as the distance between a molecule and a set of molecules. If previously selected molecules are included, the number of distance calculations (T_{centroid})

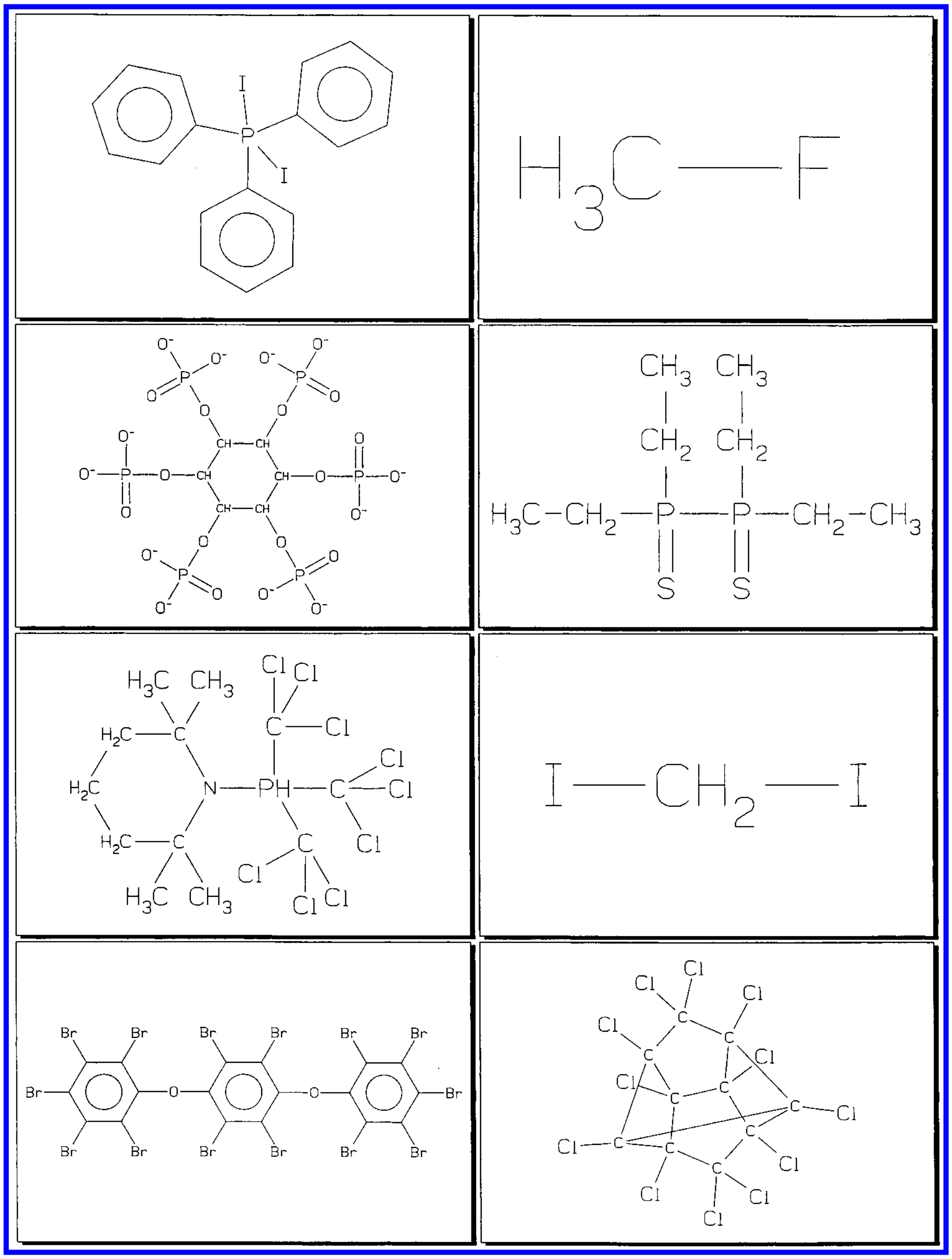

Figure 3. Example structures selected using the max_diss algorithm with no rejection or demerit rules.

required by the centroid algorithm is given by

$$T_{\text{centroid}} = Nn + n\left(\frac{1}{2} - \frac{n}{2} - n_{\text{ps}}\right)$$

Note that when n_{ps} is 1, $T_{\text{max\_diss}}$ is the same as T_{centroid}. When n_{ps} is greater than 1, the max_diss algorithm requires $N(n_{\text{ps}} - 1) + n_{\text{ps}}(1 - n_{\text{ps}})$ more distance calculations than the centroid algorithm. The max_diss method described here allows the use of any metric for determining the distance between molecules and a molecule and set of molecules while maintaining a linear time complexity.

Molecules with erroneous or extreme descriptor values may result in outlier molecules which are selected early in the sequential process described above. In fact, one use of the spread design is to detect pharmaceutically unreasonable molecules by examining the structure of the molecules in the sequential order they were selected. After the undesirable molecules have been identified, the spread design may be re-run to select a diverse set of pharmaceutically reasonable molecules.

Coverage Designs. The objective of a coverage design is to identify a subset of molecules that is maximally similar to the candidate set of molecules. Given a definition of directed similarity between subsets (*e.g.* $d_{\beta 1}$, $d_{\beta 2}$), all subsets of size n could be tested and the subset (including the n_{ps} previously selected molecules) with the highest similarity to the candidate set of molecules would be chosen. Again, this approach suffers from a combinatorial explosion in the number of subsets that need to be tested. Our work to date has focused on efficient implementations of coverage designs that can be applied to large chemical databases and has not considered design optimization due to the marginal design improvements and greatly increased computational time. For example, modified Fedorov optimization (SAS OPTEX procedure, CRITERION=U)[7] resulted, on average, in a design that was 4.6% better than sequential selection (with respect to the sum of the minimum distance from each

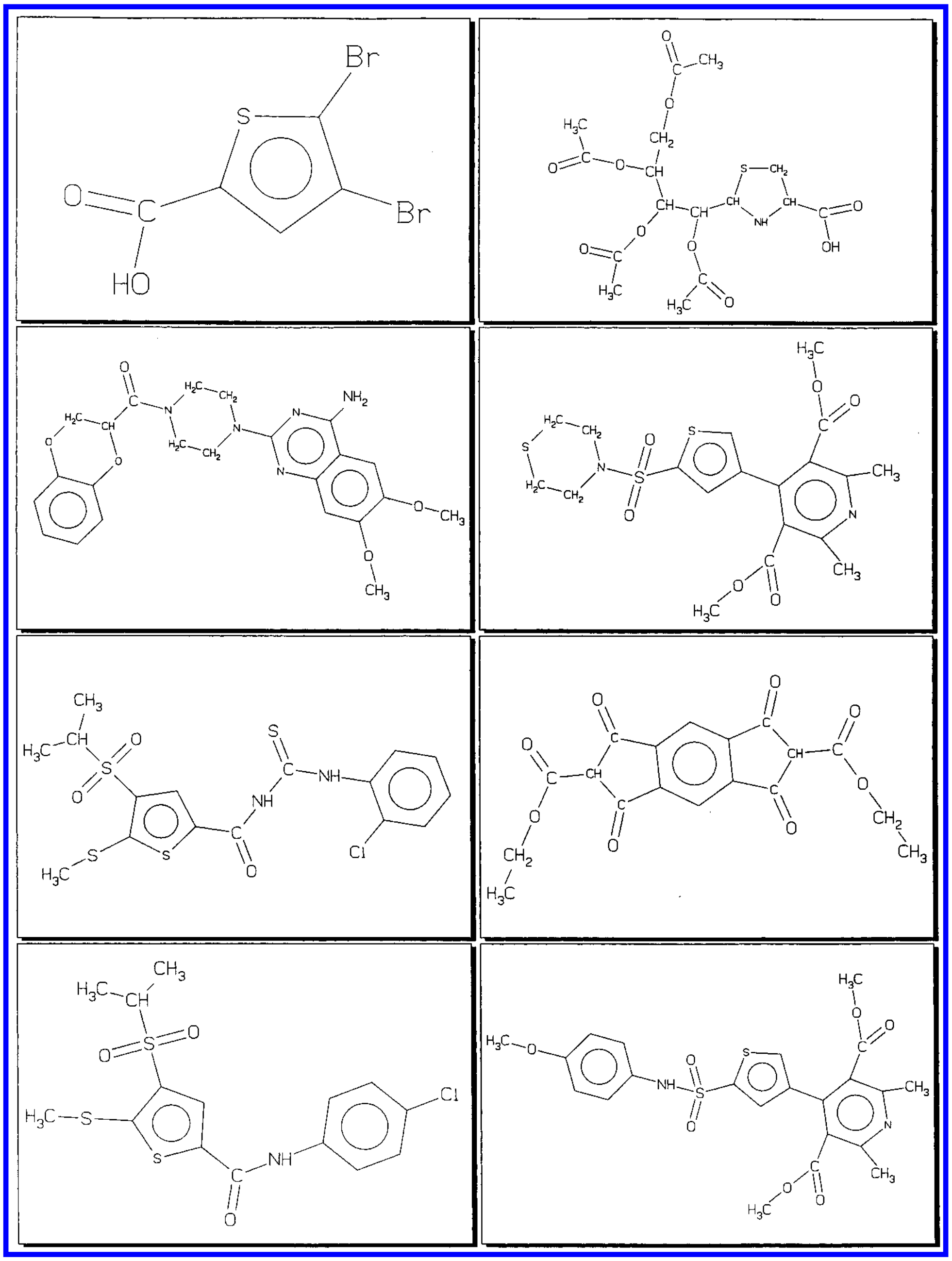

Figure 4. Example structures selected using the max_diss algorithm with rejection and demerit rules.

candidate point to the design), while requiring 21 times the computational time to select 20 points from the 1400 points shown in Figures 1−4.

The method we have used to approximate a coverage design clusters the database of candidate molecules into $n + n_{ps}$ clusters using a modified *k*-means clustering algorithm[15] and selects molecules near the cluster centers for the design subset. The use of cluster-based methods to select design points from a fixed set of possible design points has been discussed by Zemroch.[16] While the k_cov algorithm was developed for real-valued descriptors, the method can be adapted to accommodate binary fragment bit string descriptors as well as alternative clustering methods. There are a large number of options for clustering data reported in the statistical and chemical information literature. For real-valued descriptors, any hierarchical (*e.g.* Ward's) or nonhierarchical (*e.g. k*-means) method can be considered. For binary fragment bit string descriptors, the Jarvis−Patrick method is commonly used to cluster compounds.[17] An extensive review of various clustering methods was done by Downs[18] and Barnard.[19] It should be noted, however, that for very large databases, many clustering algorithms (*e.g* Ward's or Jarvis−Patrick) require computation and storage of the entire distance matrix, an $O(N^2)$ operation, and hence are not computationally feasible for the large collections of molecules of interest here.

The n cluster centers computed from the candidate molecules are treated as dynamic cluster centers and are allowed to migrate normally in the *k*-means algorithm. The n_{ps} previously selected molecules are treated as fixed cluster centers that do not migrate as cluster centers normally do in the *k*-means algorithm. To begin, each molecule in the database is assigned to one of the $n + n_{ps}$ initial cluster centers. Following this initial assignment, the centroid of the dynamic clusters is computed using the coordinates of the cluster members. The n_{ps} fixed cluster centers remain unchanged. Each molecule in the database is then reassigned using the new dynamic cluster centers and the fixed cluster centers. This process repeats until the cluster affiliation of all molecules in the database converges. For each dynamic cluster of molecules identified by the clustering algorithm, the molecule closest to the cluster center is added to the set of selected molecules. Molecules assigned to fixed clusters are not considered as candidates for the design. Cluster based methods require strategies for processing clusters containing a single compound (accepted by k_cov) and deciding which compounds should be selected to represent a cluster (molecule closest to the cluster center by k_cov). These questions do not need to be addressed with the MDC method developed by Hudson.[5]

Molecules with erroneous or extreme descriptor values may result in the outlier molecule as the only member of a cluster. The coverage design may be re-run following the identification and removal of any undesired "singleton" clusters.

The clusters produced by the *k*-means clustering algorithm are dependent on the initial dynamic cluster centers. Milligan showed that, under various conditions, *k*-means clustering performed well if the initial cluster seeds were chosen carefully.[20] Milligan found that the "best" *k*-means clustering was obtained when a hierarchical method (*e.g.* Ward's) was used to generate the initial cluster seeds. This is not possible in our applications since the emphasis here is on large molecular databases for which hierarchical clustering methods are not computationally feasible. In general, the initial dynamic cluster centers are usually selected to be spread over the design space.[20,21] We have used the max_diss approximation of a spread design to identify the initial dynamic cluster centers for the *k*-means clustering algorithm used by k_cov.

The number of distance calculations required by the k_cov approximation to a coverage design is equal to the $T_{\mathrm{max\_diss}}$ computations required to identify the initial dynamic cluster centers plus the number of distance calculations required to cluster the database and select the nearest molecule to each dynamic cluster center. There are $(n + n_{\mathrm{ps}})(N - n - n_{\mathrm{ps}}) + (n_{\mathrm{iter}} - 1)(n + n_{\mathrm{ps}})(N - n_{\mathrm{ps}})$ distance calculations required to cluster the database where n_{iter} is the number of iterations required for the *k*-means algorithm to converge. The number of iterations required for convergence is a function of the spatial arrangement of molecules in the database and the initial dynamic cluster centers. Once clustering is complete, there are $N - n_{\mathrm{ps}}$ distance calculations required to find the molecule closest to the center of each dynamic cluster center. The total number of distance calculations ($T_{\mathrm{k\_cov}}$) required by the k_cov algorithm is given by

$$T_{\mathrm{k\_cov}} = T_{\mathrm{max\_diss}} + n_{\mathrm{iter}}(n + n_{\mathrm{ps}})(N - n_{\mathrm{ps}}) + N - n_{\mathrm{ps}}$$

Implementation. The preprocessing steps to remove missing values and compute standardized principal components have been implemented in the SAS software system.[21] The max_diss algorithm to approximate a spread design and the k_cov algorithm to approximate a coverage design have been implemented as C programs on a symmetric multiprocessor (SMP) Sun Ultrasparc UNIX server. For both algorithms the most expensive loops were threaded to take advantage of multiprocessor systems.

RESULTS

A two-dimensional data set containing 1400 hypothetical molecules was constructed to visually demonstrate the differences between edge, spread, and coverage designs. Again, these designs are defined in terms of their objective functions over a discrete set of *N* molecules, but two-dimensional examples are helpful in understanding what each design is optimizing. The data set was constructed to have five tightly packed clusters (bivariate normal), two loosely packed clusters (bivariate normal), and molecules uniformly distributed over the two-dimensional design space. For illustrative purposes, eight molecules were randomly chosen and labeled as having been selected in a previous design (*i.e.* future selections should complement these eight molecules). The twenty molecules selected from this simulated data set using an edge (D-optimal) design to augment the previously selected molecules are shown in Figure 1a. The effect of including two quadratic terms and one linear interaction term into the D-optimal design is shown in Figure 1b. Figure 1c shows the twenty molecules selected using the max_diss approximation to a spread design to augment the previously selected molecules. Figure 1d shows the twenty molecules selected using the k_cov approximation to a coverage design to augment the previously selected molecules.

Several differences between the D-optimal, spread, and coverage designs are illustrated in Figure 1:

(1) The D-optimal design selects molecules near the corners first and then around the edges of the design space. Regions of space containing previously selected molecules are not avoided to the extent they are with spread or coverage designs.

(2) Adding additional terms (*e.g.* quadratic and interactions) to the D-optimal design selects molecules away from the corners but still does not closely approximate a spread design or avoid previously populated regions.

(3) The spread design selects the most diverse subset of molecules (relative to the other methods presented here), including molecules near the edges as well as throughout the design space.

(4) The coverage design selects molecules near the center of clusters. Molecules near the edges of the design space are naturally avoided since they are unlikely to be near the center of a cluster.

To verify the time complexity analysis of the max_diss approximation of a spread design, simulation studies were conducted using threaded application code on one to four processors. For each simulation, sixty dimensional uniformly random features on (0,1) and a Euclidean distance metric were used. The dimension of the feature vector was selected to provide a sufficiently challenging computation for the simulation. No attempt was made to model the distribution of real molecular databases for these simulations since distributional properties do not affect the computational performance of max_diss. The run times in CPU minutes are reported in Table 1 for simulation cases with *N*, n_{ps}, and *n* varied to demonstrate the linear time complexity of the max_diss algorithm in each parameter. For each simulation case, the linear model predicting CPU time resulted in fits with r^2 values greater than 0.99.

Verification of the time complexity of the k_cov approximation to a coverage design is not as straightforward

Table 1. Run Times (CPU min) with N, n_{ps}, and n Varied To Demonstrate Linear Time Complexity of the *max_diss* Algorithm Using One and Four Processors on a Four Processor SMP UltraSparc

N	n_{ps}	n	1 CPU	4 CPUs	speedup
20 000	10 000	5 000	5.7	1.7	3.4
40 000	10 000	5 000	17.4	5.1	3.4
80 000	10 000	5 000	41.6	11.2	3.7
160 000	10 000	5 000	88.2	24.1	3.7
95 000	5 000	5 000	33.6	9.2	3.7
100 000	10 000	5 000	53.7	15.1	3.6
110 000	20 000	5 000	89.7	24.2	3.7
130 000	40 000	5 000	160.9	42.6	3.8
91 000	1 000	5 000	21.5	5.8	3.7
91 000	1 000	10 000	39.6	10.9	3.6
91 000	1 000	20 000	73.4	20.2	3.6
91 000	1 000	40 000	130.4	36.1	3.6

as the max_diss algorithm. The time required to cluster a database using the *k*-means algorithm is, in general, significantly more than that required for the max_diss algorithm and is dependent on the spatial distribution of the molecules in the database as well as the initial cluster centers. As an example, 19 CPU min were required by the modified *k*-means algorithm to select 480 molecules from a database of 32 000 candidate molecules (a real molecular database, not simulated), described by 78 real-valued molecular descriptors, on a single processor.

Figure 2 shows the speedup obtained when additional processors are applied to the parallel implementation of the max_diss and k_cov algorithms.

DISCUSSION

Eli Lilly and Co. has an active, on-going compound acquisition program. One of the primary objectives of this effort is to enhance the diversity of the molecules tested in high-throughput screening, thereby increasing the probability of finding structurally diverse leads in as many different kinds of screens as possible. Chemical diversity is seen as especially important in assays where there may be little or no useful information on possible lead compounds, and as much of the chemical "haystack" as possible must be searched for the "needle(s)". Our high-throughput screening capacity is, however, limited by the current technology, which means that individual screens can be tested with only a subset of the available chemical "haystack".

Given finite high-throughput screening (HTS) capacity, a finite budget for compound acquisitions, and ever increasing numbers of compounds available from outside suppliers, a rapid, automated means of assessing and ranking the diversity of third party compounds is needed. Of course, in this context, the chemical diversity of a set of molecules is assessed relative to the existing collection, so two different pharmaceutical companies applying the same methods to the same set of new molecules would likely choose very different molecules as being the most diverse.

The molecules tested in HTS must be diverse enough to provide an adequate number of distinct leads in a wide variety of screens. However, they must not be so diverse as to be pharmaceutically unreasonable. Maximum value is derived from an HTS hit when a medicinal chemist decides to use that molecule as the basis for an optimization study. Nor must the molecules be excessively reactive, perhaps giving rise to erroneous results from HTS. The forces of diversity and pharmaceutical reasonableness are in direct competition with each other. Without modification, the spread design methods described here will maximize the diversity of the molecules selected, without regard for pharmaceutical reasonableness. Clearly, other constraining steps must be included in the diversity assessment.

A set of over 100 substructure based rules is used to reject molecules containing undesirable functional groups, either excessively reactive, pharmaceutically unreasonable, or containing groups known to interfere with HTS. Both an upper and a lower molecular weight cutoff are also applied, as well as a cutoff on Computed LogP.[22] Early experiments with the spread design selection method gave strong intuitive support to its effectiveness. After selection, the most diverse molecules were reviewed by a small term of medicinal chemists. Initially, the molecules ranked most diverse were almost all considered pharmaceutically unreasonable, reflecting the fact that our existing collection did not contain such molecules—polyhalogenated molecules were a vivid early example. Once such problems became known, the substructure based rules would be expanded to suppress these classes of molecules and the diversity ranking procedure and review repeated. At first, subsequent runs merely identified other classes of pharmaceutically unreasonable molecules; however, after about four selection/review cycles, the molecules being selected were considered to be "reasonable".

A modification to the spread design procedure allows for the notion of a "demerit" to be assigned to individual molecules. Demerits are scaling factors ranging between 1.0 (no demerits) to 0.0 (rejected). For example, within the team of medicinal chemists, there was not universal agreement concerning the ultimate desirability of molecules containing specific functional groups. Some within the group wished to exclude all molecules containing specific functional groups, whereas others were unwilling to *a priori* reject such a relatively large number of molecules. All agreed to reject molecules containing two or more of these marginally acceptable groups. To this end, a demerit value of 0.5 is applied to each occurrence of such groups in a molecule. Molecules without any occurrence of these groups receive no demerit, molecules with one identified group are demerited by 0.5, and molecules containing two or more are demerited to 0.0, which excludes them from subsequent consideration. Many other substructural features, and computed properties, are assigned demerit values. The demerit values from all sources are combined. Once a molecule's cumulative demerit scale value reaches 0.0, it is excluded from subsequent processing. Molecules having a molecular weight just above the lower molecular weight cutoff are also assigned a diminishing demerit, thereby somewhat smoothing out what would otherwise be a step function for molecular weight filter. Several other properties are treated similarly.

The demerit scale factors multiply computed distances in the max_diss algorithm, which means that molecules with low scale factors (more demerits) will appear to have short distances between themselves and other molecules and therefore will have a decreased probability of being selected. Similarly, if two molecules appear equally diverse, that is, an equal distance to an already selected point, the molecule with the highest scale factor (fewest demerits) will be selected preferentially. Of course different scaling factors can also mean that a less diverse, but without demerits, molecule can be selected before one which is more diverse but which has demerits. This demerit scaling has been found to be very effective in decreasing the number of demerited molecules selected.

Molecule lists for possible acquisition come in a variety of formats, most commonly variants of the MDL MolFile.[23] As the lists are converted to SMILES[24] form, the connection tables are checked for obvious errors, such as a five valent carbon and other unusual valences. As many as 1% of the molecules in a list can be rejected at this step. Unfortunately, connection table errors which nevertheless produce nonerroneous structures cannot be detected. Molecules containing isotopes and molecules containing unusual elements (heavy metals, etc.) are also rejected at this stage. Some degree of chemical standardization is also applied at this stage to transform molecules into forms which can be recognized by subsequent substructure queries.

The rejection/demerit substructure searches are then performed. As many as half of the molecules in a list may be eliminated during this phase. The molecules are then reduced to their largest fragment, and a lookup against the existing inventory is done using the graph lookup capabilities of Thor,[14] which ensures tautomer matches. Duplicates are rejected. A set of 70 topological and other molecular descriptors (Molconn,[25] Clogp,[22] Savol[26]) is then computed for the remaining molecules. Demerits based on computed properties are then determined and combined with existing demerit values, usually resulting in the elimination of many more molecules. For a typical list of 15 000 molecules, the steps above take about an hour. The molecular descriptors are then combined with those for the existing inventory and submitted to a spread design. The selections are then reviewed by medicinal chemists. This final review is used to check the substructure based rejection/demerit rules, as well as to suggest new rules. Molecules may also be rejected at this stage for nonspecific reasons.

To test these rules, and to assess their impact on known drugs, they were applied to the Comprehensive Medicinal Chemistry (CMC) database[23] containing almost 6700 molecules. After summing demerits from all sources, 4664 molecules (70%) remained. Some of the reasons molecules were rejected include the following: (a) molecular weights too low (*e.g.* methyl alcohol, cyclopropane); (b) peroxides (*e.g.* artemisinin); (c) too many halogens (*e.g.* bromoform); (d) reactive alkyl halides (*e.g.* chlorobutanol). Overall, almost 90% of the rules recorded one or more hits against compounds in the CMC database.

As a demonstration of the interaction between the descriptor based maximum diversity selection method, max_diss, and the chemistry based rejection/demerit rules, a simple experiment was performed. A set of 90 865 "small" molecules available from various commercial sources was considered. There was no consideration of the desirability of these molecules as potential pharmaceutical entities. A set of 78 simple descriptors was computed for these molecules and used to select the 500 most diverse molecules from the set.

In the absence of external constraints, the molecules selected by the method are generally not pharmaceutically desirable. Some of the smaller, and therefore easier to depict, molecules are shown in Figure 3. In addition, many much larger molecules were selected; molecules having molecular weights above 1100, having long chains and/or complex ring systems. Many of these most diverse molecules would be rejected by the rejection rules.

In fact, the maximum diversity selection process was a powerful guiding force in developing the rules. When first run relative to the existing inventory of generally drug-like molecules, the method identified large classes of molecules which needed to be suppressed. As these were eliminated, subsequent runs revealed yet other undesirable classes. After several select/review iterations, rule development evolved from eliminating large classes of molecules, to more closely examining detailed structural features of the molecules. Today, more than 100 rules are used for rejecting and demeriting structures.

In order to demonstrate the influence of the set of rules, the set of 90 865 molecules was subjected to the rejection/demerit process, which reduced the set to just 28 896 members. The selection process was repeated, but this time, the resulting molecules are generally more "reasonable"—some examples are shown in Figure 4. It is important to remember that the initial set was not selected with any pretext of pharmaceutical activity, and almost every time this procedure is run on a new set of molecules, either new rules are proposed or refinements to existing rules are made: some kinds of new chemical diversity are of interest; others are not. Final review of compounds for acquisition is always still done by human experts. We are not confident that the current system could be evolved to the point where the human review step could be eliminated. The primary value of the method is to eliminate both undesirable molecules and those which are most similar to molecules currently in the inventory, thereby allowing the human experts to focus on those molecules which are both reasonable and different from what is already owned.

CONCLUSIONS

In this manuscript we have described three methods for selecting finite subsets of molecules from large molecular databases: edge designs, spread designs, and coverage designs. Efficient, linear time order algorithms were described for a simple sequential implementation of a spread and coverage design. The differences between the molecules selected via these three design techniques was illustrated with a simple two-dimensional example. Lastly, we have demonstrated how these selection methods can be used to augment an existing chemical database with molecules purchased from external sources.

ACKNOWLEDGMENT

The authors would like to thank Bill Heath and Steven Kaldor for their help in developing the rejection/demerit rules, Doug Johnson for his many helpful suggestions, and Scott McAhren for his recommendations on the software implementation of the algorithms.

REFERENCES AND NOTES

(1) Holliday, J. D.; Ranade, S. S.; Willett, P. *Quant. Struct.-Act. Relat.* **1995**, *14*, 501−506.

(2) Martin, E. J.; Blaney, J. M.; Siani, M. A.; Spellmeyer, D. C.; Wonk, A. K.; Moos, W. H. *J. Med. Chem.* **1995**, *38*, 1431−1436.

(3) Taylor, R. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 59−67.

(4) Cummins, D. J.; Andrews, C. W.; Bentley, J. A.; Cory, M. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 750−763.

(5) Hudson, B. D.; Hyde, R. M.; Rahr, E.; Wood, J. *Quant. Struct.-Act. Relat.* **1996**, *15*, 285−289.

(6) Tobias, R. *Proc. Symp. Interf. Comput. Sci. Stat.* **1994**, *26*.

(7) *SAS/QC Software, Usage and Reference*, Version 6, 1st ed.; SAS Institute Inc.: Cary, NC, 1995; Vol. 1, pp 657−728.

(8) Salton, G.; McGill, M. J. *Introduction to Modern Information Retrieval*; McGraw-Hill: New York, 1983.

(9) *SAS. JMP*, Version 3.1.6.2; SAS Institute Inc.: Cary, NC, 1996.

(10) Atkinson, A. C.; Fedorov, V. V. Optimum Design of Experiments. *Encyclopedia of Statistical Sciences*, Supplement Volume; Wiley-Interscience: New York, 1985; pp 107−114.

(11) Borth, D. M.; McKay, R. J.; Elliott, J. R. *Technometrics* **1985**, *27*, 25−35.

(12) DuMouchel, W.; Jones, B. *Technometrics* **1994**, *36*, 37−47.

(13) Kennard, R. W.; Stone, L. A. *Technometrics* **1969**, *11*, 137−148.

(14) Cook, R. D.; Nachtsheim, C. J. *Technometrics* **1980**, *22*, 315−324.

(15) Hartigan, J. A. *Clustering Algorithms*; Wiley: New York, 1975.

(16) Zemroch, P. J. *Technometrics* **1986**, *28*, 39−49.

(17) Jarvis, R. A.; Patrick, E. A. *IEEE Trans. Comput.* **1973**, *C-22*, 1025−1034.

(18) Downs, G. M.; Willett, P. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 1094−1102.

(19) Barnard, J. M.; Downs, G. M. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 644−649.

(20) Milligan, G. W. *Psychometrika* **1980**, *45*, 325−342.

(21) *SAS/STAT User's Guide*, Volumes 1−2, Version 6, 4th ed.; SAS Institute Inc.: Cary, NC, 1994; Vols. 1 and 2, pp 823−850.

(22) Daylight Chemical Information Systems, Mission Viejo, CA.

(23) MDL Information Systems, San Leandro, CA.

(24) Weininger, D. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31−36.

(25) Hall and Associates, Quincy, MA.

(26) Dr. Robert Pearlman, University of Texas at Austin, Austin, TX.

CI9702858