A General Method for Exploiting QSAR Models in Lead Optimization

Richard A. Lewis*

Computer-Aided Drug Design, Eli Lilly and Company Limited, Windlesham, Surrey GU20 6PH, United Kingdom

Received September 22, 2004

Computer-aided drug design tools can generate many useful and powerful models that explain structure−activity relationship (SAR) observations in a quantitative manner. These models can use many different descriptors, functional forms, and methods from simple linear equations through to multilayer neural nets. Using a model, a medicinal chemist can compute an activity, given a structure, but it is much harder to work out what changes are needed to make a structure more active. The impact of a model on the design process would be greatly enhanced if the model were more interpretable to the bench chemist. This paper describes a new protocol for performing automated iterative quantitative structure−activity relationship (QSAR) studies and presents the results of experiments on two QSAR sets from the literature. The fundamental goal of this work is to try to assist the chemist in his search for what to make next.

Introduction

The quantitative modeling of structure−activity relationships is a cornerstone of modern medicinal chemistry, in all its various forms, from classical Hansch-type (2D) QSAR through to pharmacophore modeling, 3D-QSAR (for example, CoMFA analysis) and docking. However, anecdotal evidence suggests that many powerful models are not being properly used to drive the design of new compounds. There seems to be a tradeoff between statistically rigorous models that are hard to interpret and more visual models that make weaker predictions but display their results in a chemically intuitive manner. If models are being used by a medicinal chemist, it is often by entering one structure at a time or by constructing a Markush-like library, which in itself is a tedious process. The fundamental question this work tries to address is the plea from the bench chemist, "What do I make next?".

Several approaches toward making 2D-QSARs more interpretable have been developed. In some cases, it has been possible to distill the SAR down into simple rules, the best example of which is the rule-of-5 for bioavailability.[1] The extent to which the rule-of-5 has become entrenched in the psyche of medicinal chemists is a powerful demonstration of how models can and should influence the design process. Inductive logic methods try to employ a similar approach.[2,3] Simple rules can be very powerful tools to give to the bench chemist. The next approach is to restrict the QSAR model to the descriptors that one could reliably interpret, a strategy pioneered by Abraham[4] in his work on physicochemical property prediction. A body of knowledge and experience is built up around the descriptors, based on prior models, allowing the latest model to be interpreted in context. This does, however, limit the number and type of descriptors used. Techniques that color the atoms and bonds according to whether the fragment contributes positively or negatively to the model (holographic QSAR[5]) also have a role to play, but again are restricted to fragment-based descriptors. It would be preferable to have a general method for interpreting QSARs. Automated iterative design takes a QSAR model and a structure and tries to use the model to suggest improved structures. This alone can be helpful to the chemist, as he can see what changes in the structure impact activity. It also helps the chemist to navigate vast regions of SAR space quickly. There is a critical issue in using QSAR models, particularly in a blind, automated fashion: extrapolation. QSAR models will return predictions regardless of whether the prediction is sensible. For example, if the model says that activity is proportional to molecular weight, an automated structure generation program will try to add as many atoms as possible, leading to absurd suggestions.

The paper is laid out as follows: in the Materials and Methods section, an overview of the process of automated iterative design is presented, followed by more detailed descriptions of the individual components and strategies. Two generic QSAR models derived from the literature are used to explore which combination of methods and strategies seem to work well. We then discuss the potential issues with the procedure in more detail, suggesting areas for further research.

Materials and Methods

The process of iterative design can be broken down into simple modules: application of the QSAR model, generation of new ideas, filtering of the ideas, and selection of ideas for further work. The flow of ideas is outlined in Figure 1. This translates into some simple pieces of code that score a compound in a QSAR, perform substructure searching, generate novel structures from a set of seeds, test for extrapolation, and select structures to investigate further. The code has been written using Perl[6] as the glue and readily available third-party programs as the building blocks, making the program easy to implement on many different operating systems.

Application of the QSAR Model. It is assumed that a robust QSAR model for the data set of interest has already been generated and that this model can be used noninteractively. In its simplest form, we require a module that can read a 2D structure and return a prediction or score. The QSAR module can therefore be of any form or complexity, providing

* Current address: WKL-136.3.94, Novartis Pharma AG, CH-4002 Basel, Switzerland. Phone: +41-61-696-2449. Fax: +41-61-696-8676. Richard.lewis@pharma.novartis.com.

10.1021/jm049228d CCC: $30.25
Published on Web 02/11/2005

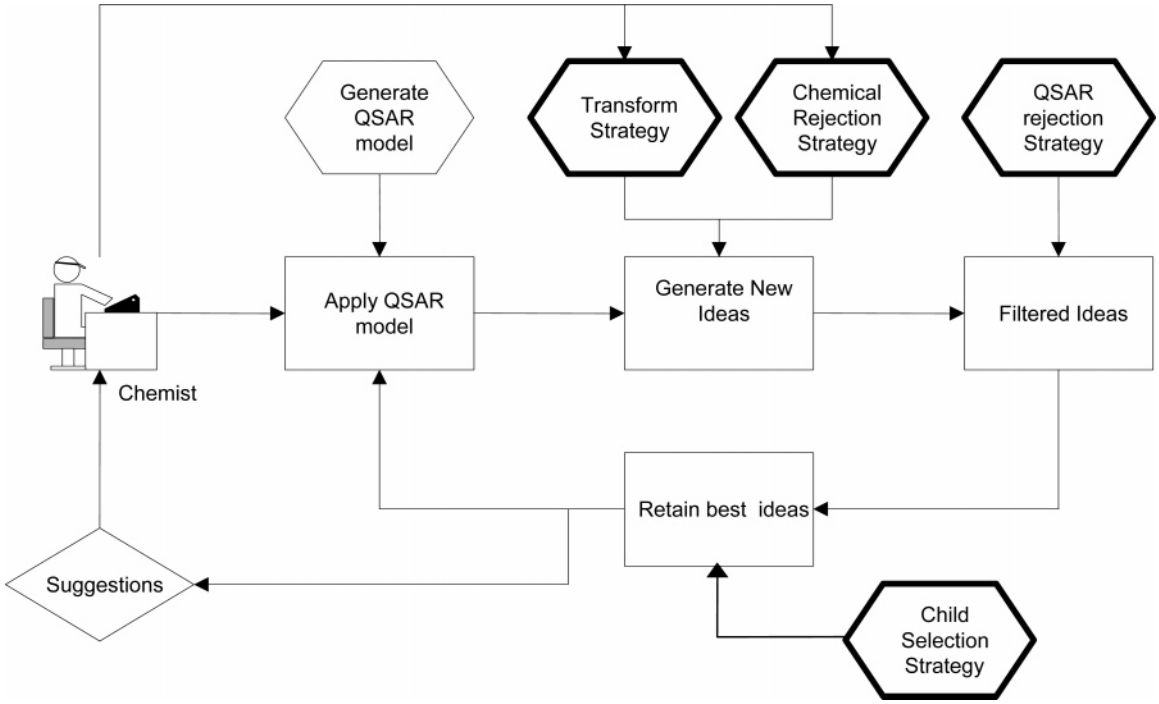


Figure 1. An overview of the automated iterative design process.

this requirement is met. In this work, linear QSAR models will be used, but the method is equally applicable to CoMFA models or docking scores, provided all the information necessary to use the model can be extracted from the 2D representation of the molecule. We have been able to wrap pharmacophore models, docking protocols, and simple CoMFA models in this manner. However, in the case of the more interpretable models such as CoMFA or docking, visual study and design is still the preferred route. For the purposes of this research, QSAR models were generated using the GFA module (part of the Cerius2 modeling suite[7]), with default settings for all parameters. Structures and activity values were taken directly from the literature. Once a model had been generated, a file of the commands required to score a novel structure was created. This file can be replayed to use the model to predict the activity of any new structure.

Generation of New Ideas. The engine for generating new ideas lies at the heart of the process. Two engines have been examined, THINK[8] and RandSmi, an in-house program that randomly permutes structures. The strategy used in THINK is to apply a series of transformations (for example, changing H to F, or Me to Cl). The transformations can also be given weights, which determine the relative probabilities that each transform will be used. The resulting molecules are assessed against substructural filters and QSAR models, before an annealing step is used to select molecules sent back to the user. There is an optional step for looking at similarity to known active and inactive molecules. This was not used here for the sake of generality; we want to be able to employ any structure generator that might be available.

RandSmi randomly permutes a smiles string using libraries of primitive operations, like "add double bond" to change a single bond into a double bond. The types of operation are very similar to those found in the menus of most chemical structure drawing packages: a partial list is given in Table 1. Each operation has an associated probability that governs how frequently it is used. The maximum ring size that can be generated was set to 7, and the program was set to produce 100 new structures from each input structure.

Filtering of Ideas. Chemical Rejection. A fundamental issue in the de novo generation of structures is the retention of chemical sense and feasibility. The chemical rejection strategy was designed to screen out structures that are highly likely to interact nonspecifically with an assay system, for example, acylating agents, nucleophiles, electrophiles, or redox agents. The substructures used in this work were taken from earlier work on analyzing corporate databases, HTS screens, and third-party offerings.[9−12] These rules are fairly mature and can be used with confidence. These have been implemented as a series of substructure filters and were used in all experiments.

Table 1. Operations Used by RandSmi To Permute Smiles Strings

operation	relative probability
add a double bond	0.5
remove an atom	1.0
decrease bond order	0.8
make or break an aliphatic ring	0.4
move fragment from one atom to another	1.0
swap adjacent atoms	1.0
change C to N or O	0.4
change O or N to C	0.6
add a fragment from a library	0.6

The concept of a privileged substructure or scaffold has also been employed. Knowledge of the SAR or intellectual property factors may dictate that a key substructure or scaffold is retained. This can be specified, so that almost all changes, or idea generation, occur outside this scaffold. This has been implemented as a simple postgeneration filter: structures that do not contain the key substructure are penalized by dividing their score by a bias term of 1.05. This will allow structures missing the privileged substructure to become new ideas, but only if they are radically better. It would be more efficient to tie this into the actual generation algorithm, but one would lose the modularity of the current scheme. Although not described here, it would be possible to force the generator to avoid substructures in competitor patents or in the literature (a localized rejection scheme). Synthetic feasibility is always a consideration for structure generation programs, as it is important to generate structures that appeal to the bench chemist. Methods for assessing synthetic feasibility or complexity have been described,[13,14] but the reliability of these models is still questioned by medicinal chemists. Methods for predicting synthetic feasibility suffer from the lack of objective data as to what in practice would be hard to make. These methods have not been implemented here, as the use of the

privileged substructure seems to perform a similar function in the chemists' eyes. The chemist can focus the change made to regions of the structure that he already knows how to manipulate. There is a price to be paid in terms of the reduction in the space that can be searched. The price is worthwhile, as no QSAR is perfect and speeding up the drug discovery cycle by getting experimental data to improve the model is probably more important.

QSAR Filtering. The similarity principle[15] says that similar molecules should have similar biological profiles. The key question then is how to measure similarity. The approach used here is to define the distance between two objects in terms of the descriptors in the QSAR model, on the grounds that these have already been shown to describe the observed activity. The structures used to construct the QSAR model form the training set. Each structure can be represented by the vector of descriptors. Distances can be measured between these vectors, to provide a measure of similarity. Any structure that falls outside the bounds of allowed space can be discarded both as an extrapolation outside the knowledge contained in the QSAR and by invoking the similarity principle. The bounds of space can be set as follows.

Unlimited: this can be used to investigate the effectiveness of the bounded strategies.

Maxmin: the maximum and minimum values plus a tolerance of each descriptor.

Threshold: a maximum distance to the nearest vector in the training set

There is also a choice of distance metrics (Euclidean or Manhattan), the normalization strategy, and whether to use a hard cutoff or a (Gaussian) penalty function. Normalization is performed by descriptor to produce a range of 0 to 1. Using normalized distances, a cutoff of 0.1 (10%) can be used, to reflect the usual level of precision in the QSAR model. A preferred approach is to set the value of the distance cutoff by clustering the training set in the same descriptor space as the QSAR and then visually examining the dendrogram to estimate a fusion distance. The Gaussian function was set to have an acceptance rate of 70% at the cutoff chosen.

Selection of Ideas for Further Examination. The process of selection can be summarized very simply: Start with an idea and then for a given number of iterations, take the next idea off the pile and try to make it better. New ideas that do not look sensible are discarded, and the remainder are added to the pile. This cycle is continued until there are no more ideas or the number of iterations is exceeded. Structure generation is a stochastic process, and there is a tradeoff between the amount of sampling (structures generated) and the time taken. If this tool is to be of any value in an interactive session, cycles must be performed in minutes. It should also be possible to leave the generator running in batch. As a further filter, compounds that are identical to those seen before either from the SAR set or from previous rounds of generation are removed, to prevent recycling. The strategies used to choose the structures for the next iteration are as follows.

Best: only the top compound.

Elite: the *N* best compounds, usually 5 or 10.

Better: everything that shows a predicted improvement over the previously discovered best activity.

MC: Monte Carlo selection using the previously discovered best activity as the reference point.

Manual: the chemist makes the choice.

Chemical Transformations. The chemical transformations govern the changes that can be made to a structure by the program. They do not have to be viable one-step chemical reactions, but they should capture the sort of changes a chemist might make to a molecule during optimization, for example, changing a methyl group to a chloro group. The magnitude of the changes must also be considered. Simple changes such as methyl to chloro are usually conservative and would be considered fine polishing during optimization of a lead. Other changes are more radical, like growing a methyl to a phenyl. The more radical changes will explore chemical space faster, but may miss out on better but more simple changes. Simple changes may explore space too slowly to be useful. There is also a potential conflict between the chemistry-based rejection rules and transforms. Rejection rules can remove the intermediates created by more primitive transforms so that some transforms must be included explicitly to occur. An example is the conversion of carbonyl to oxime. This is a simple chemical reaction, but using transforms, one would have to take a more circuitous route. Two possible routes would be (i) reduce bond order (carbonyl to alcohol), change element type (alcohol to amine), then add atom (amine to hydroxylamine). At this point, route i would fail as structures containing hydroxylamine would be removed by the chemical rejection rules. Route ii would start with a change of element type (carbonyl to imine) and then would fail again due to conflict with the chemical rejection rules. The strategies used in this work are as follows.

Simple:
simple chain length changes, e.g. $CH_2 \rightarrow CH_2CH_2$,
and swapping H, F, Cl, I, OH.
Complex:
simple chain length changes, e.g. $CH_2 \rightarrow CH_2CH_2$;
swapping H, F, Cl, I, OH;
swapping O, N, H;
reduction; oxidation;
ring size changes; and
arylation.

QSAR-Driven Transforms. The similarity principle can be used to tailor the transformation set. This is an extension of the concept of isosterism[16] employed by a medicinal chemist. Each transformation in the *complex* set was scored in the context of the QSAR model by counting the number of times a fragment appears in the QSAR training set. The counts were transformed into relative probabilities by taking a normalized geometric mean of the counts for each pair of fragments in a transformation. Transformations that include H do get higher scores, but this was acceptable. Transformations with very low probabilities can be omitted from the final set used by the algorithm. This transformation set will be referred to as "*weighted*". To enhance this still further, a list of chemical fragments and known isosteres (for example, phenyl and thiophen) was constructed. For the purposes of counts, matches that are part of any privileged substructure are ignored. Exhaustive pairwise combination of the fragments gives 1572 transformations, which were scored as before to give the "*weighted_2*" set. The next approach was to compute the vector of each fragment in QSAR space. Before this can be done, the empty valences of each fragment must be filled; methyl groups were used to fill aliphatic valences and phenyl groups for aromatic valences. Methyl and phenyl groups were used to minimize the effect on the vector of the fragment. The distance between the fragments on either side of a transformation can be computed and merits added for similar fragments. Fragments that had vectors close to the origin were deleted, as they are not well described by the QSAR space. Now fragments that were not part of the original QSAR set, but which are similar, can be introduced. This is the "*QSAR-driven*" transformation set. This restriction may not be appropriate if extra factors, for example, druggability, are important. A change that does not affect activity but which does improve druggability may be very desirable indeed. The final strategy was to classify the transformations into those derived from the QSAR, those from the isosteres, and the rest. Each transformation within a class was given an equal score, and the classes were weighted QSAR:isostere:other in the ratio 7:5:2. This is the *classification* transformation set.

Results

For the first test case, the dataset generated by Scozzafava et al.[17] was chosen, as it had already been shown that a reasonable QSAR model could be built.[18] This set was derived from a combinatorial library of 150 compounds (Figure 2), with −log(activity/M) data on human carbonic anhydrase II ranging from 5.8 to 9.7.

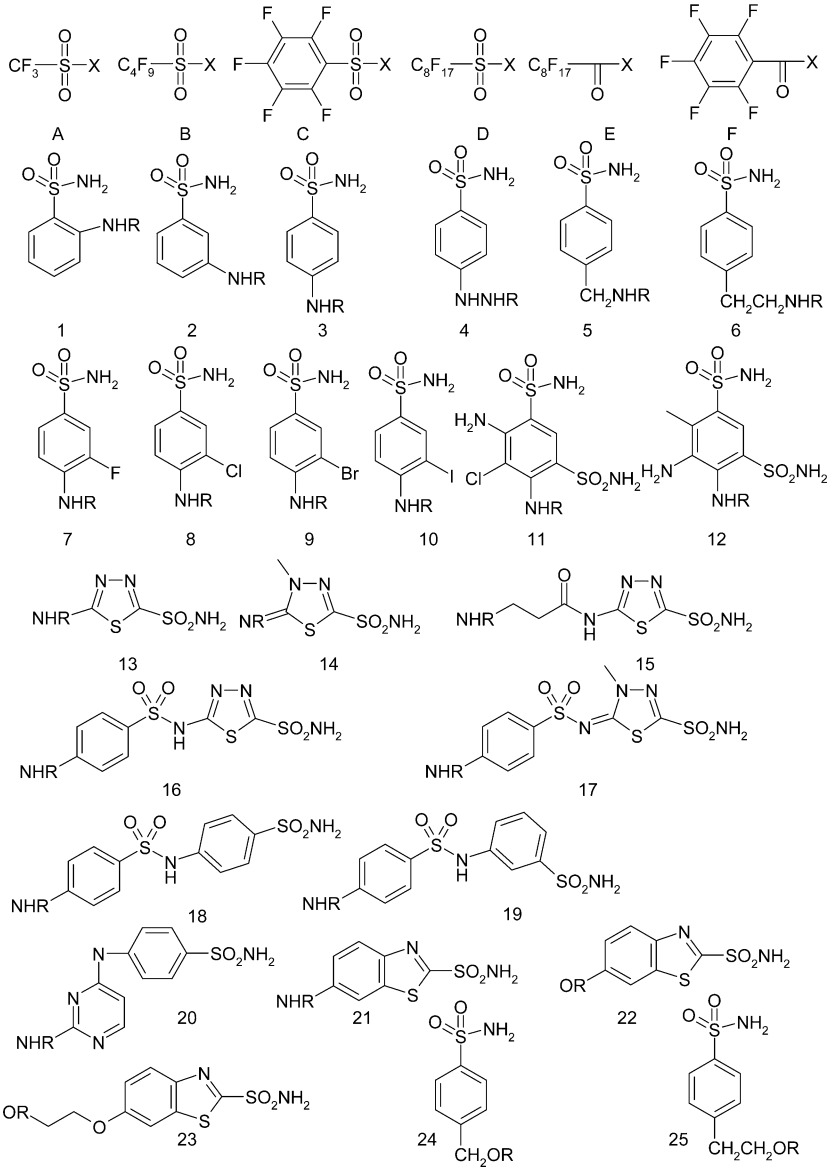


Figure 2. The fragments used to build the combinatorial library used in test case 1.

A linear QSAR model (eq 1) was derived by building the structures in smiles format, converting to 3D using Corina,[19] and then using the GFA module in Cerius2 with default settings and descriptor sets.

$$Y = 7.5 - 0.6\text{PHI} - 5.7\text{Jurs-RPCG} + 0.2\text{S\_dsN} + 1.7\text{N\_aaS} + 0.001\text{Vm} \quad (1)$$

$$R^2 = 0.81,\ F = 127,\ Q^2 = 0.8,\ \text{PRESS} = 33.4$$

The meanings of the descriptors in eq 1 are PHI, molecular flexibility index; Jurs-RPCG, charge of most positive atom divided by the total positive charge; S_dsN, electrotopological state for sp^2 N; N_aaS, electrotopological count for aromatic S; Vm, molecular volume inside the contact surface. The meaning of the descriptors is not discussed further, because the point is not to derive and interpret the model, but to use it as a component in automated iterative design. Todeschini and Consonni have published a comprehensive guide to descriptors[20] and their derivations, if further information is required.

Each compound in the SAR was used as the starting structure, with different combinations of the selection, transform, and rejection strategies. THINK was used to generate the new structures. The goal was to examine the improvement in predicted activity, the number of ideas generated, the running times, and the general similarity of the generated structures to the original idea. Failures (when no improved structures were found) are also recorded. Daylight fingerprints[21] combined with a Tanimoto coefficient were used to give a familiar measure of similarity to aid in the assessment of the results. The results are presented in Table 2. It should be kept in mind that there is no absolute way to assess the results, due to errors in the underlying data, in the QSAR model, from the incomplete sampling, and from extrapolations.

The algorithm was able to propose structures with improved activity (as predicted by the QSAR) for the great majority of the 150 starting structures, and in some cases all of them (runs 5, 10). The trend is that the predicted improvements are greater for ideas with lower starting activity (Figure 3). This is intuitively reasonable, as ideas with high activity will be at the limits of the space described by the training set. It was

Table 2. Results of Different Structure Generation Strategies Using Test Case 1

	strategy			structures with improved activity over starting point					
run	selection	QSAR rejection	transform	best improvement found	av no. found	average improvement	average similarity	no. of failures	av run time/s
1	best	unlimited	basic	1.56	18.61	0.23	0.79	28	519
1a	best	unlimited	basic	1.56	18.66	0.23	0.79	28	492
2	best	unlimited	complex	2.45	43.71	0.82	0.73	47	1503
2a	best	unlimited	complex	2.45	43.09	1.02	0.73	95	884
3	elite	unlimited	basic	1.14	30.61	0.22	0.79	28	905
3a	elite	unlimited	basic	1.45	33.66	0.24	0.79	28	588
4	better	unlimited	basic	1.34	33.21	0.24	0.79	28	1373
4a	better	unlimited	basic	1.56	62.57	0.22	0.79	0	1212
5	better	unlimited	complex	1.99	68.76	0.63	0.79	0	1653
6	MC	unlimited	basic	1.56	387.45	0.33	0.79	66	1680
7	MC	unlimited	complex	2.57	1432.15	0.87	0.65	85	1920
8	elite	maxmin	complex	2.32	268.86	0.87	0.65	18	720
9	elite	threshold	complex	2.60	254.56	0.81	0.67	6	768
10	better	maxmin	complex	2.63	403.44	0.81	0.72	0	1342
11	better	threshold	complex	2.29	314.65	0.73	0.73	1	1407
12	elite	threshold	weighted	1.97	19.24	0.23	0.67	50	888

The predicted improvements are relative to the activity of the structure used to start the run.

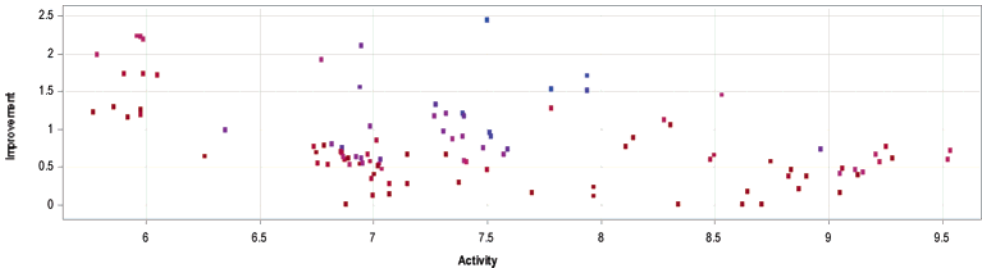


Figure 3. Plot of predicted improvement in activity against starting activity. Activities are plotted on a logarithmic scale. The points are color-coded red to blue depending on how many ideas were generated (range 1−153). Data taken from run 8 (Table 2).

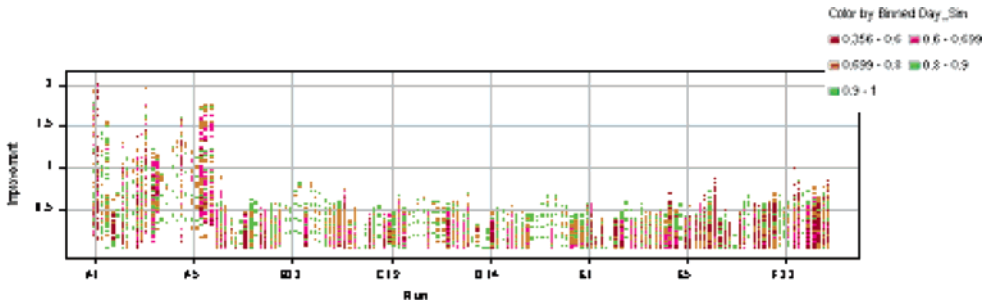


Figure 4. Plot of all improved ideas against initial idea, coded by Daylight similarity. Data taken from run 8 (Table 2). The run code refers to the starting structure formed from the fragments A−F, 1−25 given in Figure 2.

also found that the algorithm explored chemical space well, moving away from the starting idea (as measured by Daylight fingerprint similarity) giving improved structures with both high and low similarity to the original idea (Figure 4)

It was also found that the richer transform sets gave more ideas and better predicted improvements but sometimes with more failures (runs 1/2, 4/5, and 6/7). The use of probabilities gave a poorer performance (runs 9/12) but allows the user to force the structure generation to keep closer, in terms of substructural similarity, to the SAR. A more appropriate form of weighting is presented in test case 2. In terms of the search strategies, using an *elite* of five structures seemed to offer the best compromise between efficiency and coverage. The process is stochastic, so that sometimes the order in which structures are processed can affect the results. "First in, first out" and "Last in, last out" schemes were implemented, but in paired test runs (1/1a, 2/2a, 3/3a, 4/4a in Table 2), there was found to be little difference between the two schemes. Advancing all improved structures using *better* sampling took 50% longer to run as using an *elite*, but without yielding significantly better results, so that *elite* sampling seems to be adequate (runs 9/11). Using *elite* sampling, the best QSAR rejection strategy seems to be *threshold* to the training set (runs 3/8/9), with the distance being calculated in normalized Euclidean space; there are fewer failures, and the average predicted improvement is better. This is probably due to the constraints causing the generator to focus on exploration of the more fruitful areas of chemical space. Use of other functional forms did not seem to add much in this case, but the optimal metric is likely to be dependent on the QSAR model. The best combination of strategies for this QSAR model appeared to be *elite*/*threshold*/*complex*. The CPU times taken (SGI R12K running IRIX6.5) were highly variable (runs between 720 and 1920 s per idea were observed)

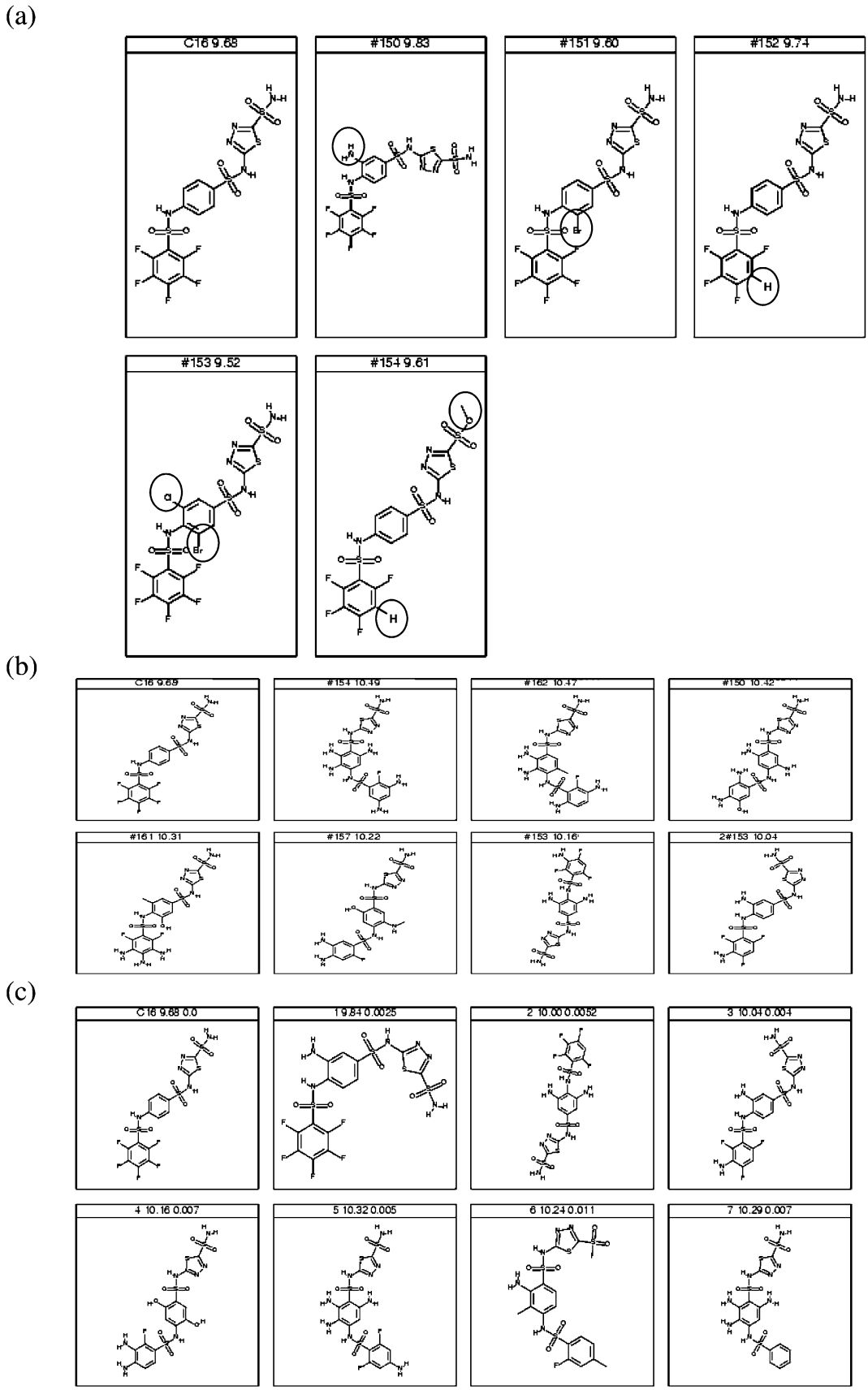


Figure 5. (a) Structures with name and predicted activity. C16 is the initial idea. Changes have been highlighted with circles. Data taken from run 2 (Table 2). (b) A continuation of the run, showing the clear breakdown of the unrestrained optimization which is resulting in oversubstitution in the central ring. (c) Using *threshold* constraints. The iteration in which the structure was generated, its predicted activity, and its distance (1-Daylight similarity) to the training set are given. Data taken from run 11 (Table 2).

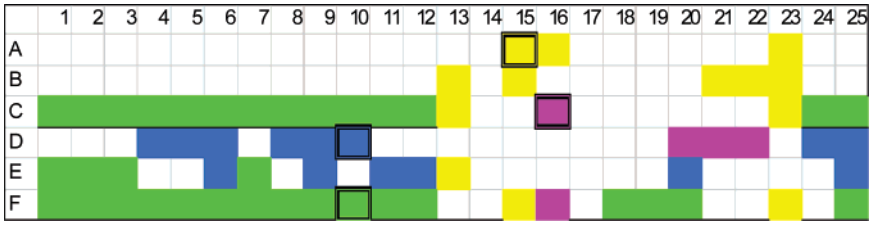
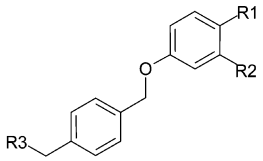


Figure 6. Map of the other members of the combinatorial library regenerated from four starting points, A15 (yellow), C16 (magenta), D10 (blue), and F10 (green).

and depended on the structure of the idea and more often on the speed at which third-party programs could be initiated and executed. A more efficient method would be to run these continuously in parallel, to remove the overhead costs.[22] These times are too long for live interaction with a chemist, and further constraints should be used; these are discussed below.

It is illuminating to look at an example of a single case taken from run 2 in which the structure generator is run in unlimited mode, i.e., allow extrapolation away from the QSAR training set. The structures generated in the first iteration are reasonable (Figure 5a); the next set of structures (Figure 5b) do have much better predicted activities, but the recommended polysubstitution of the central aromatic ring would stretch the bounds of credulity of most chemists. The inclusion of threshold constraints does not prevent the same phenomenon from occurring (Figure 5c), but it takes more iterations (in which many structures are generated). It should also be noted that the structures are similar using the Daylight fingerprint measure, illustrating a known blind spot in this metric.

A good test of the process is to investigate whether one can jump from one starting structure in the SAR to another. By default, a record is kept of all structures that have been generated, and this is initialized with the QSAR training set to stop the program from regenerating what is already known. The next experiment was to turn this off, so that it would be possible to regenerate other members of the training set. This is useful for investigating the transformation rules and sampling issues. It is reasonable to expect that not every structure will be accessible from every other structure: some of the starting and end points may be too dissimilar, or even too similar, or an end point might be by-passed as "better" structures are found. Four of the less active members of the QSAR set were chosen at random as starting points (A15, C16, D10, and F10), and strategies of *elite*/*threshold*/*complex* were used. The combinatorial nature of the set makes it easy to visualize the other structures discovered (Figure 6). As might be expected, the jumps seem to follow along rows and columns, rather than being more spotted. Some rows (A, B) and columns (18, 19) are sparsely populated. Substituents 14 and 17 are not seen, possibly due to an overly strict rule for imine rejection. Fragments B, 18, and 19 would require many modifications to be generated from the other reagents, so they may be sampled more with a more extensive search. To get to products containing fragment B from A or D requires four chain length modifications at least. Using an arbitrary cutoff of 1000 nM, many the inactive compounds contain reagent A (8/25 inactive), so one would not expect this fragment to be regenerated as often during an optimization-driven search.



R1	R2	R3
H, F, Cl, Br, I, CH_3, CF_3, C_2H_5, C_3H_7, $CH(CH_3)_2$, C_4H_9, C_5H_{11}, cyclopentyl, $CH_2C_6H_5$, $CH_2CH_2C_6H_5$, $CH{=}CHC_6H_5$, OCH_3, OC_2H_5, OC_6H_5, $OCH_2C_6H_5$, $C(=O)CH_3$, $C(=O)$-cyclopropyl, $C(=O)$-cyclopentyl, $C(=NOH)$-CH_3, $C(=NOCH_3)$-CH_3, imidazol-1-yl, piperidinomethyl, (phenylamino)methyl	H, F, Cl, CH_3	Piperidin-1-yl, azepan-1-yl, pyrrolidino, $(H_5C_2)_2N$-
$-CH_2CH_2CH_2-$, $-CH_2CH_2CH_2CH_2-$		

Figure 7. A schematic representation of the congeneric series of molecules used in test case 2.

The second test case selected was based on 38 H3 receptor antagonists from Miko et al.,[23] with activities, as $-\log(K_i/\mathrm{M})$, spread between 5.9 and 8.6. The same procedure as above was used to generate the QSAR model given in eq 2:

$$Y = 7.0 - 0.06\text{IAC-Total} - 0.46\text{AlogP98} + 0.6\text{CHI-V-3\_P} - 0.19\text{SC-O} \quad (2)$$

$$R^2 = 0.63,\ F{=}15,\ Q^2{=}0.43,\ \text{PRESS} = 4.6$$

where the descriptors are IAC-Total, the total information content; AlogP98, Ghose−Crippen log *P*; CHI−V-3_P, the third-order valence-modified connectivity index; and SC−O, the number of atoms. This set is made up of congeneric molecules (Figure 7), with a low-quality QSAR, which is a more interesting and more realistic challenge. All runs were done using a *threshold* strategy, based on the results of the previous experiment. THINK was used to generate the new structures.

For this test (Table 3), a privileged substructure was used, so that generation was confined to the phenoxy ring and R3 = piperidin-1-yl. As a control, the experiment was performed without the privileged substructure (run 1, Table 3). Only 13 of the 38 starting points led to structures with predicted improved activity, which dropped to 11 when the privileged substructure was used (run 2, Table 3), even with more extensive sampling strategies. Increasing the diversity of the transformation set makes matters worse (run 3, Table 3). This reflects the much smaller training set, which only covers QSAR space sparsely. Using the *QSAR-driven* transformation set is a marked improvement (run 4, Table 3), given that a lower sampling strategy, *elite*, was used. However, using the *classification* set, in which the different classes of transforms have been weighted,

Table 3. Results of Different Structure Generation Strategies Using Test Case 2

	strategy			structures with improved activity over starting point					
run	selection	QSAR rejection	transform	best improvement found	av no. found	average improvement	average similarity	no. of failures	av run time/s
1	elite	threshold	weighted	0.81	10.08	0.27	0.70	25	177
2	MC	threshold	weighted	0.66	3.55	0.09	0.78	27	125
3	MC	threshold	weighted_2	0.15	1.00	0.04	0.72	33	78
4	elite	threshold	QSAR-driven	0.44	5.22	0.09	0.76	29	100
5	elite	threshold	classified	0.89	8.75	0.29	0.81	18	205
6	elite	threshold	RandSmi	1.08	11.27	0.28	0.67	23	245

[a] The predicted improvements are relative to the activity of the structure used to start the run.

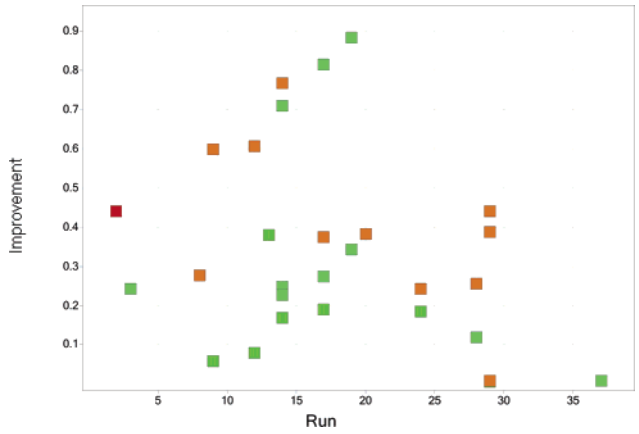


Figure 8. Plot of all improved ideas against initial idea, color-coded by Daylight similarity (red, < 0.7; orange 0.7−0.8; cyan 0.8−0.9; green 0.9−1.0). Data taken from run 5 (Table 3).

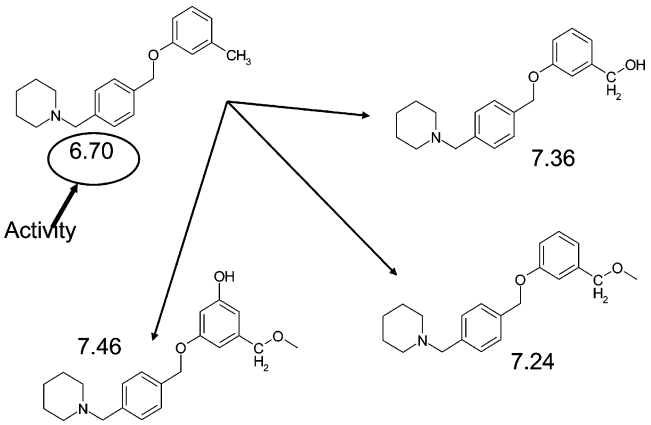


Figure 9. Structures generated from the starting structure in the top left. The activities given are predicted from the QSAR model. Data taken from run 5 (Table 3).

rather than the transformations themselves, gives the best results (run 5, Table 3), even though the use of the privileged substructure has greatly constrained the available chemical space. The ideas that are generated are highly similar (average similarity = 81%) using the Daylight fingerprint as a measure (Figure 8). In the example given, all bar one of the generated structures were more than 80% similar to the starting structure, and the core structure has been maintained, which is comforting to the chemist (Figure 9).

The experiment was repeated using RandSmi instead of THINK as the structure generator (Figure 10). The first point to note is that the chemical and privileged substructure filters removed 90% of the structures generated at the first pass. It was still possible to generate less desirable structures, for example, an enamine, highlighting one of the fundamental issues of de novo structure generation. RandSmi performed well with the starting structures with lower initial activities, as did THINK.

The regeneration experiment was repeated, and the results are illustrated by drawing a line between two numbers if it was possible to regenerate one structure starting from the other (note that the line will not be reversible, as one is always trying to improve activity). The results using the THINK engine are given in Figure 11. When the *QSAR-driven* transformation set was used, the regeneration level was poor. It can be seen that the *classification* strategy gives a much higher level of regeneration of the structures in the QSAR set, giving increased confidence that the other structures generated will also be attractive. The same experiment using the RandSmi engine gives intermediate behavior, implying that the transformations are of the right type, but need to be richer (Figure 12).

Discussion

The fundamental question that this paper tries to address is that we can use a QSAR model to predict activity, given a structure, but can we predict a structure, given an activity model? Several approaches have been tried, the most obvious being brute force. However, this quickly becomes infeasible unless the chemical space can be reduced. Combinatorial libraries offer a way of proceeding, as others have successfully shown.[24−26] One is limited (if that is the right word, given that the size of the virtual library sizes can approach several orders of magnitude) to specific changes at a small number of sites on the molecule, or specific reactions, if the goal is facile synthesis. This is appropriate in the early stages of optimization, but not in the later stages, when a chemist may be looking to handcraft a molecule to deal with metabolism issues, for example. The potential chemical space accessible may be even larger and would be quite tedious to express in library terms. Using a set of generic transformations captures this richness in a general and transferable form. The approaches are complementary, and the choice of strategy will depend on the chemistry of the series under consideration.

It is also possible to use rigorous graph-theoretic procedures to deconvolute from the descriptors to the structures.[27−31] The choice of descriptors that can be used is limited to descriptors derived from the molecular graph, which may in turn limit the quality of the QSAR. In addition, it is possible to hit pathological conditions caused by long-range changes that exceed the neighborhood of the descriptors employed. This seems to have

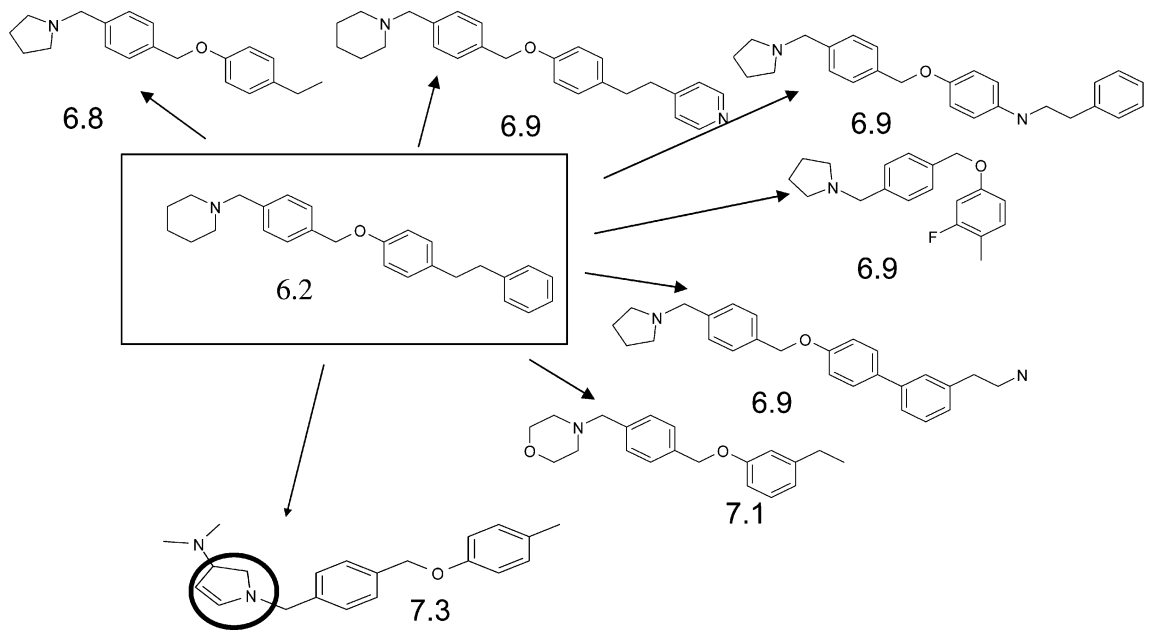


Figure 10. Structures generated from the starting structure in the box, using the RandSmi program. The activities given are predicted from the QSAR model. Data taken from run 6 (Table 3).

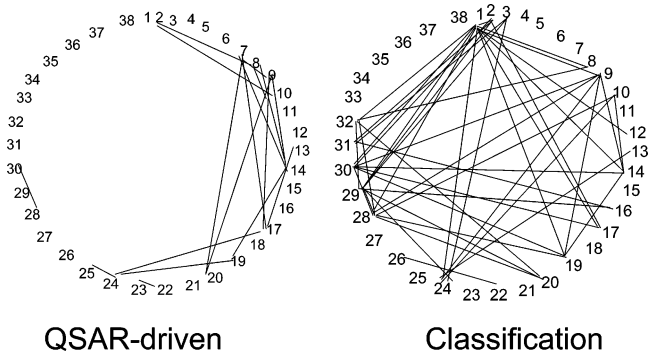


Figure 11. A plot showing which structures could be regenerated from another structure in the QSAR set, using both *QSAR-driven* and *classification* transformation strategies.

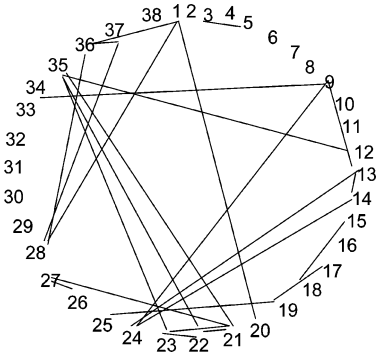


Figure 12. A plot showing which structures could be regenerated from another structure in the QSAR set, using RandSmi.

occurred in the work of Faulon,[32] in which an α-hydroxy acid group is moved from one benzyl group to another benzyl group 4 atoms down the backbone of the molecule, a change that would seem quite radical to most chemists and unlikely to preserve activity. Finally, it should be acknowledged that QSAR models are imperfect: it would seem preferable to generate families of suggestions, which can be assessed, rather than a few mathematically correct structures that may not appeal for reasons outside the scope of the QSAR model. QSAR models do not capture all the implicit understanding of the SAR obtained by a medicinal chemist after intensive study and experience of the chemical series in question. It would be possible to introduce greater or lesser degrees of automation into the process. While most users would be content just to choose the next set of structures to use in further iterations, a more sophisticated user could update the rejection rules and the transformation probabilities after each iteration. On the other hand, maximal common subgraph techniques could be used to derive fragments to be part of transformations or to determine possible privileged substructures in the training set.

Heuristic approaches, like the strategy presented here, have also been tried. An example is the program GROK,[33] which used a genetic algorithm to manipulate a set of input structures according to a user defined fitness function. This program seems to have been very fast and efficient, but also seem to have suffered from issues of extrapolation, pathological conditions being found in many of the test cases examined. The most important difference between this work and previous studies is the explicit provision made for extrapolation and consequent prediction errors in the QSAR model. It has long been known that one of the major pitfalls of QSAR, especially when linked to de novo structure generation, is extrapolation into unreasonable areas of space. Not only will the structures generated become increasingly dissimilar to the original idea, but they will also become distant from the set of structures used to train the QSAR model, so that predictions will become unreliable. A major focus of this work has been on trying to limit the generation of structures to prevent this issue from arising. As Walker et al. put it,[34] "It is crucial that the model domain is known to the user so the user may verify if a given substance can be modeled". The threshold strategy was the most effective at restraining generation to producing similar chemical structures. The choice of the value was subjective, being based on

a manual inspection of a clustering dendrogram. The most reasonable value for the cutoff would reflect the confidence radius of the QSAR model. It is possible to use principal component analysis and the Hotelling statistic to declare 95% confidence limits around descriptors to account for variance, which in turn can be used to set a Mahalanobis distance for constraints. Alternatively, a measure, $P(x)$, of the degree of novelty of an observation x (in this case a structure) can be assessed according to the formula[35]

$$P(x) = \frac{1}{\mathrm{n}(2\pi)^{d/2}\sigma^{\mathrm{d}}} \sum_{q=1}^{n} \exp\left[-\frac{(x-x_q)^2}{2\sigma^2}\right]$$

where σ is a smoothing factor, d is the number of dimensions (descriptors), and n the number of compounds in the training set. This gives information about an individual query, in addition to the validation of the QSAR model provided by hold-out sets. The value of σ should be set to avoid the extremes of large bias or large variance; the recommendation in the original paper[35] is to set σ to the average of the distance between each structure in the training set and its 10 nearest neighbors, averaged over all structures. The effectiveness of this approach toward novelty prediction for controlling structure generation is an area of ongoing research.

This work has only used a single QSAR model to drive the generation process. This was for simplicity of validation rather than any algorithmic limitations. Lead optimization is a multidimensional search for compounds that have the right blend of activity, selectivity, and drugability. Research has already shown that in silico multidimensional optimization is a powerful tool as applied to combinatorial libraries.[36,37] The challenge is to assess the degree of extrapolation when two independent QSAR models are used. For example, the QSAR training sets for a general model of permeability and for a particular chemical series binding to a receptor will have little or no overlap, which will cause the Hotelling approach to break down. The approach favored would be to optimize a desirability function made up of terms for activity and drugability. We have already introduced a bias for retention of a privileged substructure, so that adding other biases would be straightforward.

These measures only form a stop-gap for rigorous model validation and iteration using new experimental data. Staying close to what a chemist knows how to make is also important, as it mirrors the issue of extrapolation. This is another factor pushing toward suggesting structures that should be easy to synthesize, so that the drug discovery cycle is accelerated rather than impeded by the application of QSAR models. The experiments described in this paper have concentrated on making improvements on a starting idea. It might be informative to stand the process on its head, to look for changes that are deleterious to activity, which would delineate areas of SAR to avoid or where the model could be tested quickly with a very small amount of experimental data.

The effect of changing several parameters has been investigated, but the nature and the weighting scheme used in the transformation set seems to have the most impact on the quantity and quality of the structures generated. We have used two programs for performing structure generation: because of the modular nature of the process, other structure generators could be used in this process with minimal changes.[38,39] It is anticipated that many of the issues that affect the structure generation programs used here will be the same for other approaches.

Conclusions

A new protocol for performing inverse QSAR has been described and applied to two literature QSAR sets. It has been shown that novel improved (as predicted by a QSAR model) structures can be generated automatically by starting with chemists' ideas. This has been made practicable by addressing the key issues of avoiding extrapolation of the QSAR model, using constraints to keep the generated structures sensible, and using a definition of similarity based on the QSAR model to construct the set of transformations.

Acknowledgment. The author thanks Keith Davies (Treweren), Benoit Beck, Ian Watson, Mike Bodkin (Lilly), Stephen Johnson (BMS), and the referees for helpful discussions and support.

References

(1) Lipinski, C. A.; Lombardo, F.; Dominy, B. W.; Feeney, P. J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Deliv. Rev.* **1997**, *23*, 3−25.
(2) King, R.; Muggleton, S.; Lewis, R. A.; Sternberg, M. J. E. Drug design by machine learning: The use of inductive logic programming to model the structure−activity relationships of trimethoprim analogues binding to dihydrofolate reductase. *Proc. Nat. Acad. Sci. U.S.A.* **1992**, *89*, 11322−11326.
(3) Marchand-Geneste, N.; Watson, K. A.; Alsberg, B. K.; King, R. D. New Approach to Pharmacophore Mapping and QSAR Analysis Using Inductive Logic Programming. Application to Thermolysin Inhibitors and Glycogen Phosphorylase b Inhibitors. *J. Med. Chem.* **2002**, *45*, 399−409.
(4) Abraham, M. H. Scales of solute hydrogen-bonding: Their construction and application to physicochemical and biochemical processes. *Chem. Soc. Rev.* **1993**, *22*, 73−83.
(5) Tong, W.; Lowis, D. R.; Perkins, R.; Chen, Y.; Welsh, W. J.; Goddette, D. W.; Heritage, T. W.; Sheehan, D. M. Evaluation of quantitative structure−activity relationship methods for large-scale prediction of chemicals binding to the estrogen receptor. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 669−677.
(6) Wall, L.; Christiansen, T.; Orwant, J. *Programming Perl*; O' Reilly: Sebastopol; 2002.
(7) Cerius2, available from Accelrys Inc.
(8) THINK, available from Treweren Consultants; www.treweren.com.
(9) Lewis, R. A.; Mason, J. S.; McLay, I. M. Similarity Measures for Rational Set Selection and Analysis of Combinatorial Libraries: The Diverse Property-Derived (DPD) Approach. *J. Chem. Inf. Comput. Sci*. **1997**, *37*, 599−614.
(10) Hann, M.; Hudson, B.; Lewell, X.; Lifely, R.; Ramsden, N. Strategic Pooling of Compounds for High-Throughput Screening. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 897−902.
(11) Rishton, G. M. Nonleadlikeness and leadlikeness in biochemical screening. *Drug Discovery Today* **2002**, *8*, 86−96.
(12) Higgs, R. E.; Bemis, K. G.; Watson, I. A.; Wikel, J. H. Experimental Designs for Selecting Molecules from Large Chemical Databases. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 861−870.
(13) Vinkers, H. M.; de Jonge, M. R.; Daeyaert, F. F. D.; Heeres, J.; Koymans, L. M. H.; van Lenthe, J. H.; Lewi, P. J.; Timmerman, H.; Van Aken, K.; Janssen, P. A. J. SYNOPSIS: SYNthesize and OPtimize system in silico. *J. Med. Chem.* **2003**, *46*, 2765−2773.
(14) Takaoka, Y.; Endo, Y.; Yamanobe, S.; Kakinuma, H.; Okubo, T.; Shimazaki, Y.; Ota, T.; Sumiya, S.; Yoshikawa, K. Development of a method for evaluating drug-likeness and ease of synthesis using a data set in which compounds are assigned scores based on chemists' intuition. *J. Chem. Inf. Comput. Sci*. **2003**, *43*, 1269−1275.
(15) Johnson, M. A.; Maggiore, G. M. *Concepts and Applications of Molecular Similarity*; J. Wiley: New York, 1990.
(16) Wermuth, C. G. Molecular variations based on isosteric replacements. In *Practice of Medicinal Chemistry*; Wermuth, C. G., Ed.; Academic Press: 1996; pp 202−237.

(17) Scozzafava, A.; Menabuoni, L.; Mincione, F.; Briganti, F.; Mincione, G.; Supuran, C. T. Carbonic Anhydrase Inhibitors: Perfluoroalkyl/Aryl-Substituted Derivatives of Aromatic/Heterocyclic Sulfonamides as Topical Intraocular Pressure-Lowering Agents with Prolonged Duration of Action. *J. Med. Chem.* **2000**, *43*, 4542−4551.
(18) Mattioni, B. E.; Jurs, P. C. Development of Quantitative Structure−Activity Relationship and Classification Models for a Set of Carbonic Anhydrase Inhibitors. *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 94−102.
(19) Corina, available from Molecular Networks GmbH. www.mol-net.de
(20) Todeschini R.; Consonni V. Handbook of Molecular Descriptors. 2000, Wiley-VCH: Weinheim.
(21) Available from Daylight Chemical Information Systems, Inc. www.daylight.com
(22) Lewis, R. A.; Roe, D. C.; Huang, C.; Ferrin, T. E.; Langridge, R.; Kuntz, I. D. Automated Site-directed Drug Design using Molecular Lattices. *J. Mol. Graphics* **1992**, *10*, 66−78.
(23) Miko, T.; Ligneau, X.; Pertz, H. H.; Ganellin, C. R.; Arrang, J.-M.; Schwartz, J.-C.; Schunack, W.; Stark, H. Novel Nonimidazole Histamine H3 Receptor Antagonists: 1-(4-(Phenoxymethyl)-benzyl)piperidines and Related Compounds. *J. Med. Chem.* **2003**, *46*, 1523−1530.
(24) Venkatasubramanian, V.; Chan, K.; Caruthers, J. M. Evolutionary design of molecules with desired properties using the genetic algorithm. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 188−195.
(25) Sheridan, R. P.; Kearsley, S. K. Using a Genetic Algorithm To Suggest Combinatorial Libraries. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 310−320.
(26) Weber, L.; Almstetter, M. Diversity in very large libraries. In *Molecular Diversity in Drug Design*; Dean, P. M., Lewis R. A., Eds.; Kluwer: 1999; pp 93−114.
(27) Skvortsova, M. I.; Baskin, I. I.; Slovokhotova, O. L.; Palyulin, V. A.; Zefirov, N. S. Inverse problem in QSAR/QSPR studies for the case of topological indices characterizing molecular shape (Kier indices). *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 3, 630−63.
(28) Kier, L. B.; Hall, L. H. The generation of molecular structures from a graph-based QSAR equation. *Quant. Struct.-Act. Relat.* **1993**, *12*, 383−388.
(29) Hall, L. H.; Fisk, J. B. Computer generation of vertex degree sets for chemical graphs from a number of vertices and rings. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 5, 1184−1189.
(30) Kvasnicka, V.; Pospichal, J. Simulated annealing construction of molecular graphs with required properties. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 6, 516−526.
(31) Faulon, J.-L.; Visco, D. P., Jr.; Pophale, R. S. The signature molecular descriptor. 1. Using extended valence sequences in QSAR and QSPR studies. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 707−720.
(32) Faulon, J.-L.; Churchwell, C. J.; Visco, D. P., Jr. The signature molecular descriptor. 2. Enumerating molecules from their extended valences sequences. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 721−734.
(33) Weininger, D. Method and Apparatus for Designing Molecules with Desired Properties by Evolving Successive Populations. U.S. Patent No. 5,434,796, 1995.
(34) Walker, J. D.; Carlsen, L.; Jaworska, J. Improving opportunities for regulatory acceptance of QSARs: The importance of model domain, uncertainty, validity and predictability. *Quant. Struct.-Act. Relat.* **2003**, *22*, 346−350.
(35) Bishop, C. M. Novelty detection and neural network validation. *IEEE. Proc.-Vis. Image Signal Process* **1994**, *141*, 217−222.
(36) McKenna, J. M.; Halley, F.; Souness, J. E.; McLay, I. M.; Pickett, S. D.; Collis, A. J.; Page, K.; Ahmed, I. An algorithm-directed two-component library synthesized via solid-phase methodology yielding potent and orally bioavailable p38 MAP kinase inhibitors. *J. Med. Chem.* **2002**, *45*, 2173-2184.
(37) Wright, T.; Gillet, V. J.; Green, D. V. S.; Pickett, S. D. Optimizing the size and configuration of combinatorial libraries. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 381−390.
(38) EA-inventor, available from www.optive.com
(39) Brown, N.; McKay, B.; Gilardoni, F.; Gasteiger, J. A graph-based genetic algorithm and its application to the multiobjective evolution of median molecules. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1079−1087.

JM049228D