Recore: A Fast and Versatile Method for Scaffold Hopping Based on Small Molecule Crystal Structure Conformations

Patrick Maass,[†] Tanja Schulz-Gasch,[‡] Martin Stahl,*[,‡] and Matthias Rarey*[,†]

Center for Bioinformatics Hamburg, University of Hamburg, Bundesstrasse 43, D-20146 Hamburg, Germany,
and F. Hoffmann-La Roche AG, Pharmaceutical Research, PRBD-CM, CH-4070 Basel, Switzerland

Received March 16, 2006

Replacing central elements of known active structures is a common procedure to enter new compound classes. Different computational methods have already been developed to help with this task, varying in the description of possible replacements, the query input, and the similarity measure used. In this paper, a novel approach for scaffold replacement and a corresponding software tool, called Recore, is introduced. In contrast to prior methods, our main objective was to combine the following three properties in one tool: to avoid structures with strained conformations, to enable the exploration of large search spaces, and to allow interactive use through short response times. We introduce a new technique employing 3D fragments generated by combinatorial enumeration of cuts. It allows focusing on fragments suitable for scaffold replacement while retaining conformational information of the corresponding crystal structures. Based on this idea, we present an algorithm utilizing a geometric rank searching approach. Given a geometric arrangement of two or three exit vectors and additional pharmacophore features, the algorithm finds fragments fulfilling all these constraints ordered by increasing deviation from the query constraints. For the validation of the approach, three different design scenarios have been used. The results obtained show that our approach is able to propose new valid scaffold topologies.

INTRODUCTION

A central task of medicinal chemistry is to identify new classes of compounds with a specified biological profile. One typical and often successful way of solving this task is the modification of known active substances.[1] This process is called scaffold hopping[2] if molecular structures are not peripherally modified but altered more drastically. Many computational methods have been developed to support the process of scaffold hopping, differing both in the search space and in the type of molecular similarity paradigm employed. In particular, 2D design methods based on chemical fragment spaces—sets of molecular fragments with connection rules—offer the advantage of an enormously large search space of druglike compounds.[3−6] A typical task in scaffold hopping is to replace a central element of the molecular scaffold by a new molecular fragment. Databases of 2D molecular fragments like bioisosteres[7] or ring systems[8] have been compiled for this purpose. The likelihood of identifying a truly novel matching fragment is higher if explicit conformations are used. Programs such as SPLICE[9] and BREED[10] "mix and match" sets of overlaid 3D query solutions or crystal structure conformations in a combinatorial fashion. More directed queries are possible with the program CAVEAT:[11] Given a molecular structure and a selection of at least two outgoing bonds, called exit vectors, alternative molecular fragments with a similar geometric arrangement of exit vectors are searched. CAVEAT searches databases of geometric relationships of bond pairs. Through the use of experimentally observed conformations, CAVEAT avoids artifacts generated by conformation generating progams.[12]

Here we present the development and validation of a new fragment replacement tool. It was our goal to combine the virtues of CAVEAT—fast searches and the use of crystal structure conformations—with the advantages of chemical fragment spaces, namely the capability to cope with large search domains and the potential to focus on druglike chemical structures. In addition, we wanted to enable the user to define further pharmacophore-type search constraints.

Since spatial databases based on binning schemes require tolerance ranges for queries, we developed an alternative method of storing molecular fragments for fast database retrieval. The method is based on a generic indexing technique: rotation-invariant representations of the exit vectors and pharmacophore features are stored in a geometric index. Query features are searched in the index via simultaneous k-nearest-neighbor searches, and a voting system is used to rank fragments. By design, the combinatorial search algorithm finds matching fragments ordered by deviation from query features. Due to the construction of the geometric index, it is typically not necessary to scan over all fragments, which makes searches extremely fast and allows us to store a very large number of fragments.

Since it was our intention to focus on good conformations we derived molecular fragments from structures of the Cambridge Crystallographic Structure Database (CSD).[13] To select only druglike fragments, we applied various filters described in more detail later. (Note that the procedure described here is not restricted to crystallographic structures but could be applied to any kind of computationally generated

* Corresponding author phone: +49 40 428387351; fax: +49 40 428387352; e-mail: rarey@zbh.uni-hamburg.de (M.R.) and phone: +41 61 6888421; fax: +41 61 6886459; e-mail: martin.stahl@roche.com (M.S.).

[†] University of Hamburg.

[‡] F. Hoffmann-La Roche AG.

10.1021/ci060094h CCC: $37.00 © 2007 American Chemical Society
Published on Web 02/17/2007

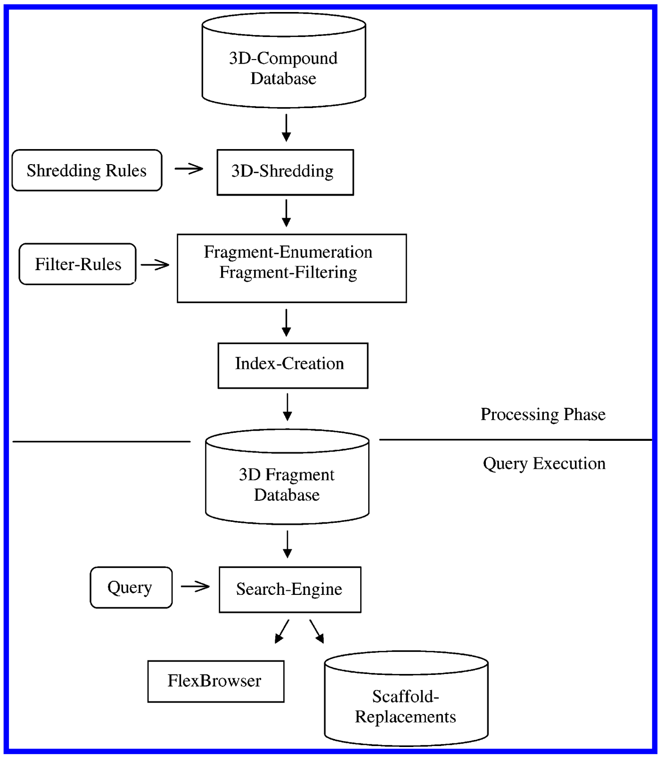

Figure 1. Overall workflow. The procedure consists of two phases, fragment database build-up, and the actual search phase.

conformations as well. The relevance of the solutions identified from computational conformations will of course very much depend on the quality of the conformation generator.) Molecular fragments are identified with a rule-based system by marking specific acyclic single bonds for disconnection. The search space is then defined by the set of all connected fragments that results from realizing all possible subsets of the allowed disconnections within the original compound. In this way, the conformational information in the database is left intact. Fragments are then filtered according to size and substructure criteria to avoid searches in nondruglike chemistry space.

In the following, we present the design of the method in detail, followed by validation experiments using three typical molecular design situations.

MATERIALS AND METHODS

Search Overview. The Recore approach consists of two phases (see Figure 1). During the preprocessing phase, a database of 3D structures is converted into a fragment database. While this index buildup may last several minutes, a typical search is in the range of a few seconds. Based on user-defined disconnection and filter rules, fragments are created, which are filtered and written to the indexed fragment database. After fragment generation, filtering, and indexing, a technique from the field of image recognition, called geometric rank searching,[14] is applied. The query execution phase takes a 3D query as input and searches for alternative scaffolds employing the indexed fragment database.

Query Definition. While our search algorithm works on entities independent of the atom connectivity of molecules, we decided to use an annotated template molecule as query input (Figure 2). A query always consists of two or more exit vectors and an optional set of pharmacophore features. Therefore, at least two bonds of the template molecule have to be marked as exit vectors. Pharmacophore features are defined by selecting atoms of the query molecules relevant for receptor interaction. Interaction directions for hydrogen bonds, etc. are calculated using the FlexX interaction model.[15] The location and, if available, the interaction direction are used as query features.

Fragment Generation. The fragment database is created from a set of 3D-structures. Fragments are generated by applying rules for marking specific acyclic single bonds referred to as cut points. The rules are defined in a way analogous to the RECAP procedure.[16] However, instead of splitting the molecules at the cut points and thus losing the information about their relative arrangements and the torsion angles at the connecting bond, we consider all combinations of cuts for generating fragments. Figure 3 gives an example of a small molecule with 11 cut points and a selection of resulting fragments. Since we do not allow cut points within rings, fragments can be created using a subtree enumeration algorithm.[17]

Fragment Filtering. To avoid considering fragments with unfavorable properties later during the search process, several types of filters are used. Additive filters such as fragment sizes and number of cuts are applied as early as possible during the enumeration process (Figure 3b). This is necessary to avoid exponential behavior especially for ring systems with many substituents. Only nonadditive filters need to be checked after fragment enumeration. The following criteria are used for filtering: *Fragment size* is defined as sum of the number of acyclic heavy atoms plus the sum of the maximum path lengths of all contained rings. The maximum path length of a ring is defined as the number of bonds in the largest of all shortest paths between any two ring-atoms (e.g., the fragment in Figure 2 has a size of 8: 6 acyclic atoms and a ring with a maximum path length of 2). With this definition, the fragment size measure intentionally favors ring systems over acyclic structures. *Distances between cuts* are also defined topologically, namely as the sum of acyclic bonds plus the sum of maximum path lengths of cycles contained on the shortest path between the cuts. Therefore, the substitution pattern of a ring has no influence on the cut distance. The *number of cuts,* which is equal to the maximum number of exit vectors of the fragment, may be limited. Additionally, substructure rules can be used to filter out unwanted fragments.

Index Creation. To enable efficient searching, the arrangement of exit vectors and properties of pharmacophore features (together referred to as *features*) are stored in a geometric index. Currently, hydrogen bond donors and acceptors, phenyl centers, and hydrophobic features are defined similarly to FlexX[15] as pharmacophore features. In contrast to FlexX, we do not use tolerance values but just use the main direction of interactions for directed features and the pure location for undirected features. This is sensible because we search for similar fragments rather than for counter groups, and features are retrieved ordered by increasing deviation from the query. To obtain a rotation- and translation-independent representation of the features in the database, we exploit the fact that each fragment has at least two exit vectors. The idea is to store all features relative to pairs of exit vectors. This can be achieved by using each

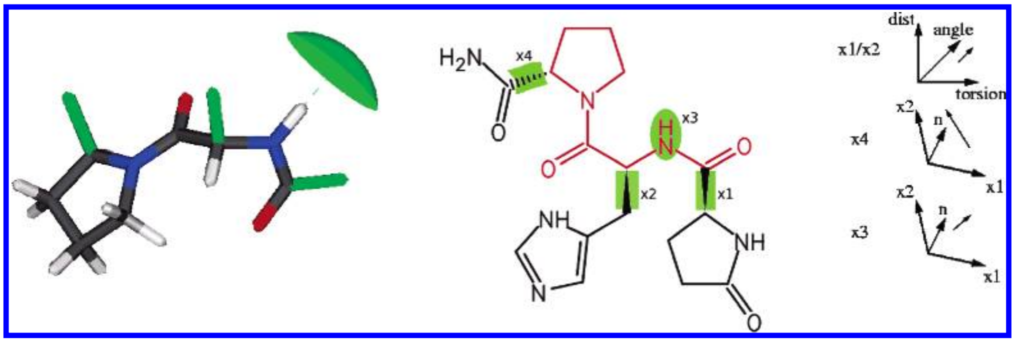

Figure 2. Molecule template fragment and the query (in green). The central fragment in the middle, highlighted in red, is to be replaced. For this set of features, all six ordered pairs of exit vectors are used to create rotation-independent feature vectors for the H-bond donor property (x3) and the respective third exit vector (x4). The illustration on the right shows the generated feature vectors for one ordered pair (for the outer right and outer left exit vector). The pair (x1, x2) itself is represented by a four-dimensional vector with the distance between them, their torsion angle, and the respective angles for both exit vectors as dimensions. The two remaining features, the third exit vector (in the middle) and the main interaction direction of the H-bond donor, are stored with the pair and their normal as basis.

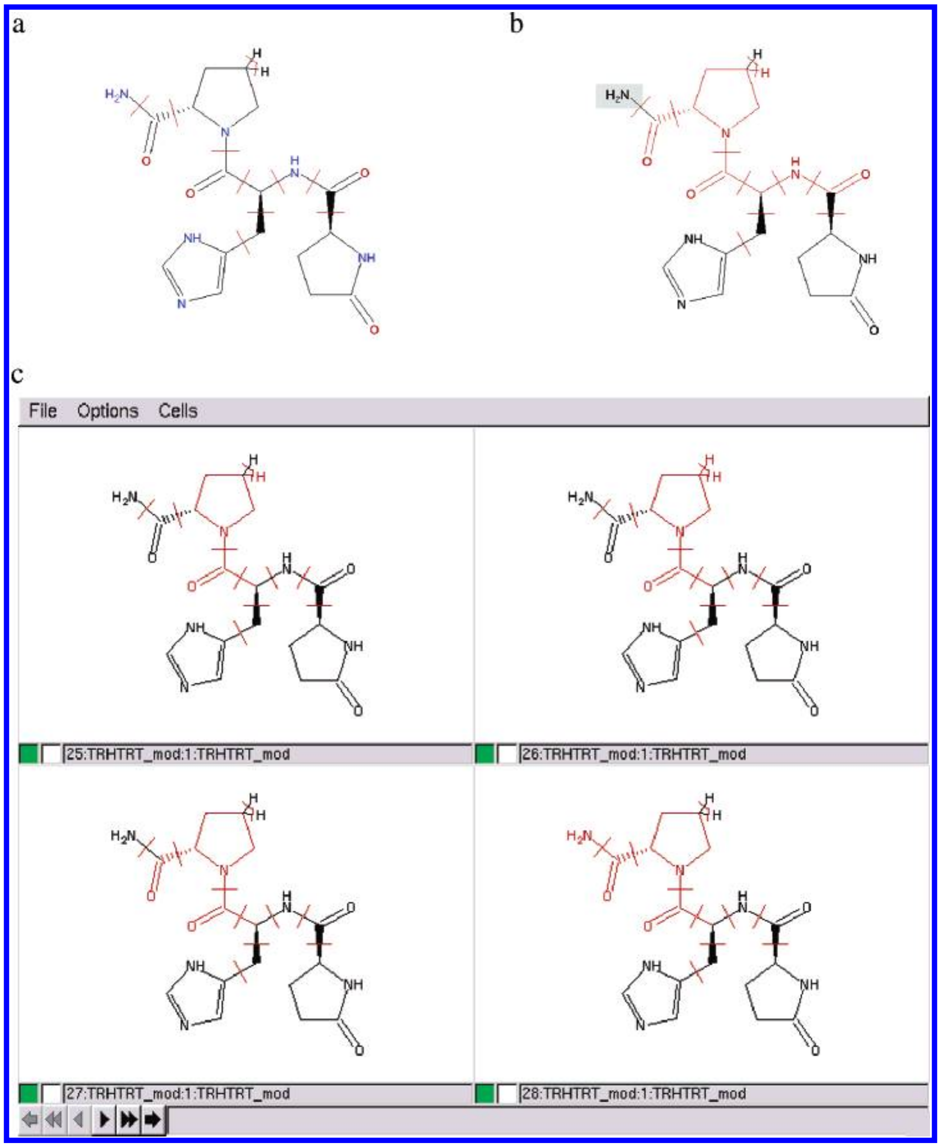
a
b
c
File Options Cells
25:TRHTRT_mod:1:TRHTRT_mod
26:TRHTRT_mod:1:TRHTRT_mod
27:TRHTRT_mod:1:TRHTRT_mod
28:TRHTRT_mod:1:TRHTRT_mod
H2N
O
N
H
NH

Figure 3. (a) Molecule with cut points shown as short lines perpendicular to the cut bonds. In (b), parts of the molecule not considered during the enumeration process are highlighted by gray boxes: As the shown fragment is already too large (size 8 + 2 > 9), the algorithm does not expand it further. In general, this is needed to avoid unnecessary exponential behavior. (c) Highlighted in red are the fragments created by subtree enumeration that are further processed and stored in the fragment database.

ordered pair of exit vectors of a fragment to form an orthonormal basis for the set of fragment features (see Figure 2). If the exit vectors are in a general position, the two exit vectors (with unit lengths) form the first two elements of the three-dimensional system. The third element is the normal on the surface defined by the exit vectors. If they are collinear, one exit vector plus the line between their starting points is considered. If all four points of the two exit vectors

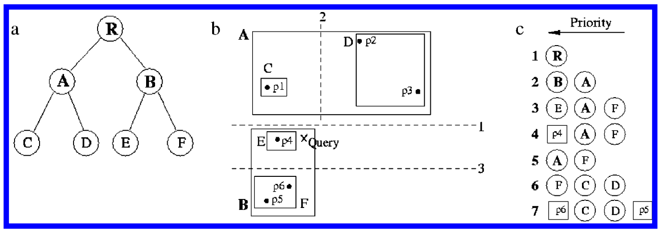

Figure 4. R-tree index and k-nearest-neighbor search. In (b), a partition with six points is shown. The enclosing boxes for the points of R-tree nodes are drawn with solid lines. In (a), the corresponding tree is depicted. The node labels match the labels of the boxes in (b). Subfigure (c) shows the priority queues for the sequence of the first seven steps of the k-nearest-neighbor search for the point labeled “Query” in (b). Increasing Euclidian distances correspond to decreasing priorities.

are collinear, each of the additional fragment features needs to be used to form bases. However, the last case is of no relevance for typical queries, and therefore such bases are normally not added to the index. Feature representations using such a basis as coordinate system are called feature vectors. All pharmacophore features and additional exit vectors are repeatedly stored relative to two exit vectors forming the basis (see Figure 2). Consequentially, feature vectors are rotation invariant: if two (ordered) pairs of exit vectors (e.g., one from the query and one from a fragment in the database) are geometrically equivalent, the corresponding fragments have the same geometric arrangement of features if they have the same feature vectors.

All feature vectors are indexed. For fast retrieval, the feature vectors are geometrically sorted. Basically, we partition the space of feature vectors by recursively applying the divide and conquer principle. For this purpose, we use a set of so-called *optimal R-trees* as underlying structure.[18] An *R-tree* is a tree where all leaves are on the same level (see Figure 4a). A node of an R-tree represents a sector of the space (enclosing a set of feature vectors), e.g. a cube in a 3D R-tree or in general an *n*-dimensional hypercube in an R-tree for *n* dimensions. The hypercube of a node always contains the hypercubes of all descendant leaves and nodes. An R-tree is called *optimal* if none of the hypercubes overlap. We store feature vectors of different types in different R-trees.

To create optimal trees we use a variant of the VAM-SPLIT-method[19] recursively splitting the space in different dimensions (see Figure 4b). In each step, the dimension with the highest variance is selected. A point *near* the median with respect to the split dimension is used as pivot element. The deviation from the median is chosen such that the capacity of inner nodes is fully exploited.

Search Engine. The algorithm searches for a single fragment fulfilling all query constraints. As noted above, a query consists of at least two exit vectors plus additional features, which can be either further exit vectors or pharmacophore features. As for the fragments during index creation, all pairs of exit vectors are in turn considered as a basis; all other features are then stored as feature vectors relative to the respective basis. A k-nearest-neighbor search is performed for each query feature vector. The R-tree is traversed starting with the root using a priority queue (see Figure 4c). The priority is the distance between the query feature vector and the nodes. This distance is defined as follows: If the query vector is within the hypercube represented by the node, then the distance is defined to be zero. Otherwise, the distance is the minimal Euclidian distance between the query vector and the hypercube boundaries. In order to compare a fragment feature vector with the query vector, we use the Euclidian distance between the two. With these definitions, the algorithm can be explained as follows: First, the root node is inserted into the priority queue. As long as there are nodes in the queue, the node with the minimal distance to the query vector is extracted. The distance is then calculated for every child of this respective node, and all children are added to the priority queue. If the node represents a fragment feature vector, then it is reported as next nearest neighbor.

A query may contain one or more features in addition to the required exit vector pair. In this case a so-called multi-k-nearest-neighbor search based on a voting scheme has to be performed: For the representation of the exit vector pair and each of the remaining feature vectors, a priority queue is created. All queues are processed simultaneously. Each time a new match is requested, each of the k-nearest-neighbor queues is checked for the distance of the corresponding query feature vector to the closest fragment feature vector and the basis of the closest vector is selected as a so-called candidate. This candidate gets a vote for this particular query feature vector. If a candidate received votes for every query feature, it is called a hit, and the corresponding fragment is displayed to the user. This procedure is carried out until either a specified number of hits have been generated or all fragments were processed. The voting scheme has two important properties: the last vote for a candidate is always for the worst matching feature vector, and candidates are not reported as hits as long as there are missing votes for single query features. Therefore, the algorithm orders hits by ascending maximal deviation from any query feature vector. That is, a fragment whose worst matching query feature differs to a lesser extent from the query than the worst matching features of other fragments is found earlier. Note that while all hits fulfill all query features, hit fragments are allowed to have additional features not requested in the query. With the method and feature representation as described above, torsion angles at exit vectors are not part of the query. Therefore, a postprocessing step is necessary to filter out fragments with unfavorable torsion angles when combined with the substituents of the query molecule.

While this method is typically fast for queries with a small number of features, the algorithm has to consider more fragments than presented to the user for queries with multiple features. As an extreme example, consider the case where half of the fragments match well to one query feature and the other half to another query feature. In that case half of the fragments will get a vote before the first hit is found.

A critical part of the search procedure is the definition of distances between feature vectors. Currently, we use weighted Euclidian distances for all feature vectors. Three-dimensional vectors represent undirected features (like hydrophobic features) in which the same weights for all dimensions are used. Directed features (like additional exit vectors or H-donors/acceptors) are stored as six-dimensional vectors representing the start and end point of the directed feature of unit length. Here, we use higher weights for the first three dimensions to place more emphasis on the location than on the direction. A pair of exit vectors forming a basis is stored

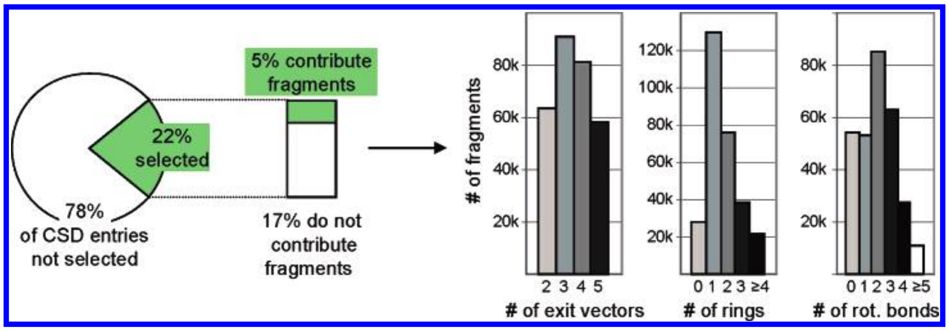

Figure 5. Statistics for 3D fragments derived from the CSD.

as a 4-dimensional vector (similar to CAVEAT) with different weights: for (i) the distance between the start points of the exit vectors, (ii, iii) both angles of the exit vectors relative to the line between the start points, and (iv) the torsion angle of the two exit vectors around the line between the start points. Weights are also used to compare different distance measures for multi-k-nearest-neighbor searches: While the comparison of feature vectors within a tree is well defined as they have the same type and dimensionality, in a multisearch, distance values from different vector spaces are compared. For this scenario, scaling factors are used to weight, for example, the deviation of H-donor features compared to the deviation of exit vectors.

Technical Details of Implementation. Two software tools have been developed: the main program for generating fragments, creating the geometric index and searching, as well as a plug-in for the visualization tool FlexBrowser to check cuts (as in Figure 3) and to search interactively. For the molecule I/O, the contact type model, and basic geometric routines, the Flex* library was used.

For fragment generation, the input is a multicompound file (TRIPOS-MOL2 format) and a set of cut and filter rules. Cleavage rules are specified by SMARTS patterns. The filter rules are ranges for the above-mentioned criteria (like fragment size, number of cuts, etc.) and excluded subgraphs. These subgraphs are also specified as SMARTS patterns and can be specific to fragments of a certain number of cuts. For space efficiency, the fragments are represented by the corresponding set of cuts and are stored in so-called cut-files. (The only time a fragment is explicitly written to disk is if it is reported as a hit). In a second step, fragments are read from cut-files and processed to remove conformational duplicates. A fragment is removed as a duplicate of another fragment if the sum of the pairwise atom distances after superpositioning is less than 1 Å.

The index creation procedure writes a platform independent binary file for the set of R-trees. To save disk space and to avoid reporting several fragments of a molecule that differ only in additional (unused) exit vectors, a combined fragment representation is derived for every pair of exit vectors of a molecule. This allows reporting only the smallest matching fragment for each found combined fragment representation.

The query feature vectors can be created with the same procedure as the fragment feature vectors, except that it is sufficient to choose an arbitrary pair of exit vectors as a basis (as all combinations of exit vector pairs are considered during index creation). In order to save disk space for the fragment database index, an arbitrary ordering for exit vector pairs can be chosen. Consequently, this requires the use of both orderings for the query. Different output modes are implemented: a hit can be reported as the original (CSD) molecule, as a fragment, or as a complete composite solution with the substituents of the query attached to the hit fragment. It is possible to either write a specified number of hits in a TRIPOS-MOL2 file or to view hits on demand with the FlexBrowser plug-in.

The current version of the 3D fragment database was generated by extracting all entries with an associated 3D structure and at least one carbon atom from the 2006 version of the CSD database (ConQuest version 1.8). Furthermore, entries with elements other than H, C, N, O, F, Cl, Br, I, or S were removed. Ions, powder structures, organometallic compounds, and structures with an *R*-factor of less than 10% were omitted from the search. This resulted in approximately 71 500 structures which were exported in sdf format and further processed by Corina[20] with the driver options “wh, no3d, neu, newtypes, rs”. The output format was set to mol2. The final subset of ∼71 200 structures represents 22% of all CSD entries (Figure 5).

RESULTS AND DISCUSSION

In this section, we first outline the rules used for the assembly of the 3D structural fragments from the CSD and then describe a number of search scenarios to illustrate the characteristics of the method. We will assume that a query is always defined by marking at least two acyclic single bonds on a single conformation of an input molecule. These bonds define the fragment that should be replaced.

Choice of Filtering Rules and Database Setup. The foremost goal of this work is the compilation of a set of chemical fragments that are druglike and at the same time represent low-energy conformations not only by themselves but also when incorporated into a larger query scaffold. Thus, we did not mark each acyclic single bond as a potential cleavage point but avoided cuts in sterically congested areas of the molecules (e.g., at quaternary centers) and cuts necessarily leading to substructures undesirable in a drug discovery context (e.g., single bonds to divalent sulfur). A set of 18 cleavage rules was compiled, most of which define C−C, C−O, C−N, and C−H bonds in specific environments (see Table 1). As opposed to the original CAVEAT approach, we consider C−H bonds as substitution points only in conformation-insensitive positions. Thus, hydrogen atoms at aromatic carbons are cleaved if the *ortho* positions are not substituted.

The resulting fragments were filtered according to size and substructures. Fragment size (as defined in the Methods

Table 1. Cleavage Rules Defined by SMARTS Patterns[a]

N	SMARTS expression	R1	R2
C1	[C;X4;!D4]-;!@[C;X4;!D4;!R]	C-R1	C-R1
C2	[C;X4;!D4]-;!@[c;$(c[n&H0,cH,S,O])]	C-R1	c-R2
C3	[c;$(c[n&H0,cH,S,O])]-;!@[c;$(c[n&H0,cH,S,O])]	c-R2	c-R2
C4	[C;X4;!D4]-;!@[C;D3]	C-R1	C-R2
H1	[c;H1;$(c([cH])([cH,n&H0,O]))][H]	c-R2	
H2	[C;H3;$(*Cc*([cH])([cH,n&H0,O]))][H]	C-R1	
H3	[C;H2;$([CH2]([CH2])[CH2])][H]	C-R1	
X1	[c;$(c([cH])([cH,n&H0,O]))][F,Cl,Br,I]	c-R2	
N1	[C;$(C(=))]-;!@[N;!$(N(C=O)(C=O));!$(N[N,O,S])]	C-R3	N-R4
N2	[C;X4;!D4]-;!@[N;X3;!D3]	C-R1	N-R5
N3	[c;$(c([n&H0,cH,S,O]))]-;!@[N;X3;!$(N[N,O,S])]	c-R1	N-R6
N4	[S;X4;$(S(=)(=))]-;!@[N;X3;!$(N[N,O,S])]	S-R7	N-R6
N5	[c;$(c[n&H0,cH,S,O])]-;!@[n]	c-R2	N-R8
N6	[C;X4;!D4]-;!@[n]	C-R1	N-R8
N7	[C;$(C(=))]-;!@[n]	C-R3	N-R8
O1	[C;X4;!D4]-;!@[O;D2;$(O[C,c]);!$(OCO);!$(OC=*)]	C-R1	O-R9
O2	[P,S]-;!@[O;D2;$(O[C,c]);!$(OCO);!$(OC=*)]		O-R9
O3	[C;$(C(=))]-;!@[O;D2;$(O[C,c]);!$(OCO);!$(OC=*)]	C-R3	O-R9

[a] Each pattern describes two atoms in a specific environment. Cut points are marked between matched atom pairs. Columns: N: name for the rule; R1: name for the cutpoint from the first atom; R2: name for the cutpoint from the second atom.

section) was restricted to 2−9. The lower limit eliminates all trivial one-atom linkers but retains potentially interesting bis-substituted 5-membered rings as solutions. The upper limit reduces the number of overly large fragments, which in many cases would be superstructures of smaller fragments due to the overlap of potential fragments generated from a single structure. The number of cuts per fragment was limited to 2−5. The vast majority of queries can be defined with 2 or 3 bonds (exit vectors), such that more cuts would rarely be used simultaneously. Keeping fragments with up to five cuts simply ensures that potentially small solutions can be identified that otherwise would be eliminated by our fixed fragment size limits.

Substructure rules were used to filter out unwanted fragments. Fragments containing undesirable typical protection groups, ring systems with eight and more ring atoms, and long aliphatic chains (≥4 connected carbon atoms) were removed. We further eliminated trivial and undesired fragments abundant in the raw fragment lists, for example simple aromatic fragments like aryl rings with various substitution patterns or short aliphatic chains and also a number of fragments that are highly populated because of their symmetric nature and the resultant high permutation numbers for possible exit vectors (an example is C(N1CCCC1)(N1CCCC1)(N1CCCCC1)(N1CCCCN1)). Conformational duplicates were only removed if the sum of the atom distances after superpositioning was less than 1 Å. This is a very conservative cutoff ensuring that no solution will be missed. Initially we supposed that a conservative cutoff would lead to many almost identical solutions that will be found with a single query. It turned out, however, that many of these almost identical solutions are already filtered out as part of the trivial substructures, in particular of the type C−C−C with all combinations of 2−5 exit vectors (originally these had constituted 1/4 of all fragments). Figure 5 demonstrates the effect of the removal of conformational duplicates filter for the fragment *Cc1cccc(*)c1. Here the fragment frequency is dramatically reduced (from more than 4000 to 69), but all significant orientations of the linker at the exocyclic carbon atom are still present. This is one of the reasons why less than one-quarter of the selected CSD entries (5% of the total CSD, Figure 5) contribute fragments to the database. Many entries only contribute duplicates eliminated in this step (originally ∼1600 CSD entries contributed the fragment *Cc1cccc*c1, pruning reduces this number to 34). Other reasons include the fact that about half of the CSD entries are either large polycyclic or macrocyclic systems without cleavable bonds or lead to overly large fragments not passing the size filter. Another large subset of compounds mainly consists of aliphatic chains that would yield the above-mentioned type of trivial fragments only. Finally, many CSD entries are too small to yield any interesting fragments.

After enumeration, filtering, and removal of conformational duplicates, a total of 300 000 fragments remain with two (63 000), three (91 000), four (81 000), and five (58 000) exit vectors. Most of these are cyclic structures: only 9% contain no ring system, 70% are mono- and bicycles, and 21% contain ≥3 rings (Figure 5). We see this as an encouraging result, as ring systems typically have well-defined preferred conformations thus leading to more predictable bioactive conformations of designed ligands. The final 300 000 fragments are a highly diverse set of almost 90 000 unique SMILES. The 10 unique SMILES that form the largest clusters (clustering is based on identical unique SMILES) are represented by 200−300 conformations (Figure 6b). One of these is for example a simple C1CCCCC1 fragment without any further substitution. We did not put further effort in filtering out these solutions because, in spite of their simplicity, they might provide insight into geometric relationships between exit vectors normally overlooked. On the other hand, almost half of the fragments (43 000) are conformational singletons (6c).

Validation Experiments. Scaffold hopping methods are designed to produce unexpected and novel ideas. For their validation, one has to identify cases where it can be shown retrospectively that the method would have proposed valid structures. Three design scenarios (queries depicted in Figure 7) were chosen to assess the quality of results obtained with the 3D indexing and rank searching procedure. First, replacements for the central diol fragment of HIV protease inhibitors were sought to examine the influence of deviations in exit vector geometry on the output. As a second example, a HIV reverse transcriptase inhibitor in its crystallographically determined binding conformation was chosen as a query structure, and related inhibitors with the same binding mode were compared with the solution structures. Finally, a structure of thyrotropin-releasing hormone was chosen as a query with three exit vectors to arrive at peptide mimetics.

HIV Protease. Two HIV protease structures complexed with symmetric diol inhibitors[21] were chosen from the public domain, one with an open chain backbone (A76928, PDB code 1hvk) and one with a seven-membered ring urea scaffold (DMP450, PDB code 1dmp). Exit vectors for both of these structures were defined as in Figure 7a,b. The CSD contains several crystal structures of the latter class of HIV protease inhibitors. Five of these are included in the CSD subset used in our searches and display the same ring conformation and arrangement of exit vectors as DMP450 in the complex. With DMP450 as query, the CSD entry SEYSAQ is identified among the highest-ranking structures (rank 13), whereas the rank drops to 2009 with A76928 as

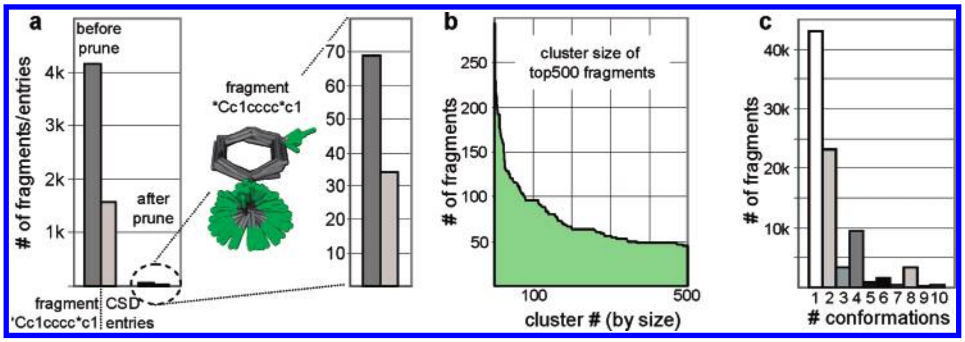

Figure 6. (a) The removal of conformational duplicates is shown on the example of the frequently occurring benzyl fragment. (b) The number of fragments of the 500 largest fragment clusters is shown. (c) The number of clusters with ≤ conformations is shown.

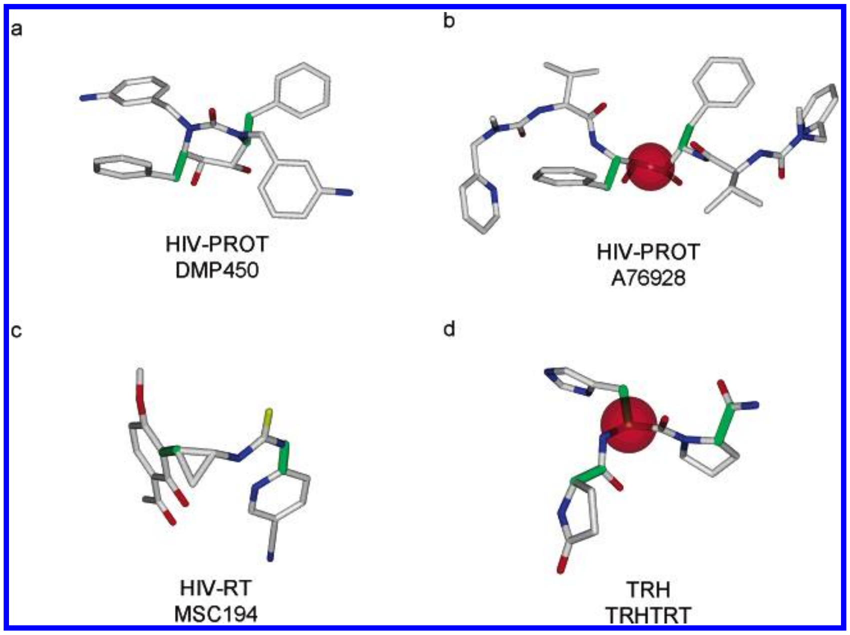

Figure 7. Queries used for validation runs of Recore. Exit vectors are shown in green, ring-constraints for the post-filtering in red: (a and b) the central diol fragments of the HIV protease inhibitors are to be replaced, (c) the linker-ring bonds are chosen as query vectors, and (d) three exit vectors are used for replacing the linker of TRHTRT.

query. What is the explanation for this significant difference? The backbone element to be replaced in A76928 is essentially a butyl fragment with three consecutive *gauche* torsions. The CSD contains many such diamond lattice fragments fitting very well onto the query vectors. The partially unsaturated seven-membered ring of the cyclic urea scaffold displays a slightly larger deviation from the diamond lattice and is therefore ranked much lower. Interestingly, a fully saturated seven-membered ring is found on rank 139 (CSD entry ZEJVAL, Figure 8a). Thus, if the exit vector arrangement of the query corresponds to a widely occurring conformation, one has to analyze many solutions to arrive at a reasonably diverse set or the query itself could be made more specific. To eliminate the high ranking acyclic scaffolds that perfectly match the diamond lattice we added a pharmacophore constraint requiring the presence of a ring atom within a sphere of 1.2 Å centered on the C−C bond carrying the two hydroxyl groups of the query inhibitor (Figure 7b). The fully saturated ring of ZEJVAL is now found on rank 34, and the first of the seven membered-rings that belongs to the class of later HIV protease inhibitors is found on rank 372 (CSD entry SEYSAQ, Figure 8b). Alternatively, a more specific query also leads to more interesting results: In the case of A76928, adding an H-donor pharmacophore feature to the query in the position of one of the hydroxyl groups leads to the identification of the SEYSAQ solution on rank 15. Clearly, there is a tradeoff between specificity of a query and coverage of new chemical space, and it will be necessary to refine search parameters on a case-by-case basis.

HIV Reverse Transcriptase. The inhibitor MSC194 belongs to a class of non-nucleoside RT inhibitors (NNRTIs) adopting a U-shaped conformation in the binding site. A characteristic feature of this compound class is the existence of an aromatic residue at both termini of the molecules, one being positioned in the lipophilic pocket formed by W229 and Y181, the other one facing P236 and Y318. A wide variety of linkers between the aromatic rings is tolerated, slightly shifting the relative orientation of the rings in the pocket and with respect to each other. Thus, the linker-ring bonds in the crystal structure conformation of MSC194 were chosen as query vectors (Figure 7c). Among the top ranking 50 structures in the list of solutions, several linker motifs known from other NRTIs were identified. Figure 9 shows three examples for which X-ray structures of closely related ligands are available. Each hit structure is shown twice. The top row contains overlays between MSC194 and the fully assembled composite structure. Below, the composite structures are overlaid with the closest known analogous inhibitor for which a complex structure has been deposited in the PDB. The solution in Figure 9a (rank 4, MOZJUG) is an analog

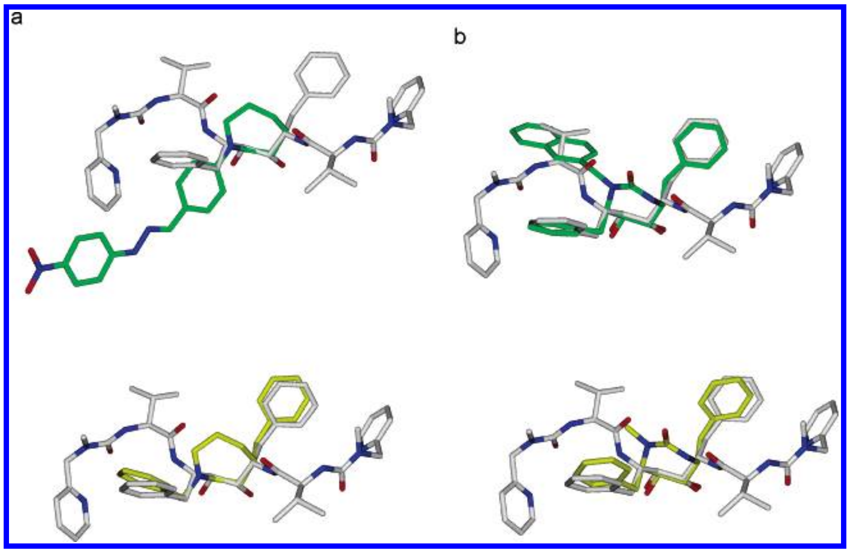

Figure 8. HIV protease test scenario. Original CSD are shown in green, composite structures in yellow: (a) query superimposed with the original CSD entry ZEJVAL (top) and composite solution structure (bottom) and (b) query superimposed with the original CSD entry SEYSAQ (top) and composite solution structure (bottom).

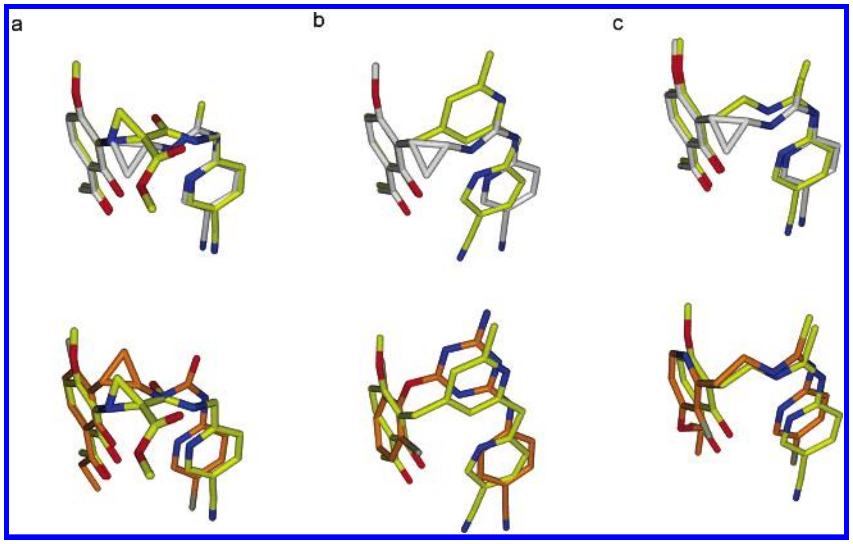

Figure 9. HIV RT test scenario. (Top row) query superimposed with composite solution structures and (bottom row) closely related inhibitors (PDB structures) superimposed with the solution structures (in orange): (a) CSD entry MOZJUG, PDB entry 1eet, (b) CSD entry CABVAD, PDB entry 1s9e, and (c) CSD entry XUCPUG, PDB entry 1ddt.

of the query with a three-membered ring pointing in the opposite direction. In addition, this solution suggests that by exchanging the MSC194 thiourea function by an inverted acid derivative (here an amide), the linker may adopt a new conformation. This is valuable information for a chemistry program aiming at linker replacements, in particular when the suggested conformations have been experimentally observed in a related context. The set of high-ranking solutions contains several *meta*-substituted aromatic linkers (138 in the top 500 solutions). Structure b in Figure 9 (CABVAD) is in fact the top ranking solution. It matches well onto a Janssen inhibitor (PDB code 1s9e). Solution c in Figure 9 is an open chain analog MSC194, identified on rank 125 (XUCPUG). This structure does not suggest a large inventive step, but it closely matches the structure of another known inhibitor, a derivative of trovirdine[22] (PDB code 1dtt).

Thyrotropin Releasing Hormone. On a query with three exit vectors, perfect matches to a new scaffold can generally not be expected. We chose a Thyrotropin Releasing Hormone (TRH), a tripeptide extensively studied during the first half of the 1990s. To match the pharmacophore hypothesis proposed at the time, the crystal structure conformation of TRH (CSD entry TRHTRT) was modified by rotating the imidazole substituent in a *trans* position relative to the pyroglutamate residue. The query was defined as in Figure 7d. Since it is a typical goal of scaffold hopping to substitute flexible scaffolds with conformationally more rigid ring-systems, we filtered the resulting solutions by a pharmacophore constraint (like described for HIV protease a 1.2 Å sphere which has to include any ring atom) shown in Figure 7d.

Olson et al. described the successful design of a constrained cyclic analog of this conformation in which the central amide scaffold is replaced by a 1,3,5-trisubstituted cyclohexane ring.[23] This system is depicted in Figure 10a (orange structure). The CSD contains a structure of a closely related compound (CSD entry ZAQCUP) lacking the imidazole ring. However, in this structure the acetamide substituent adopts a different rotameric position, such that full Olson mimetic cannot be redesigned from it. The closest

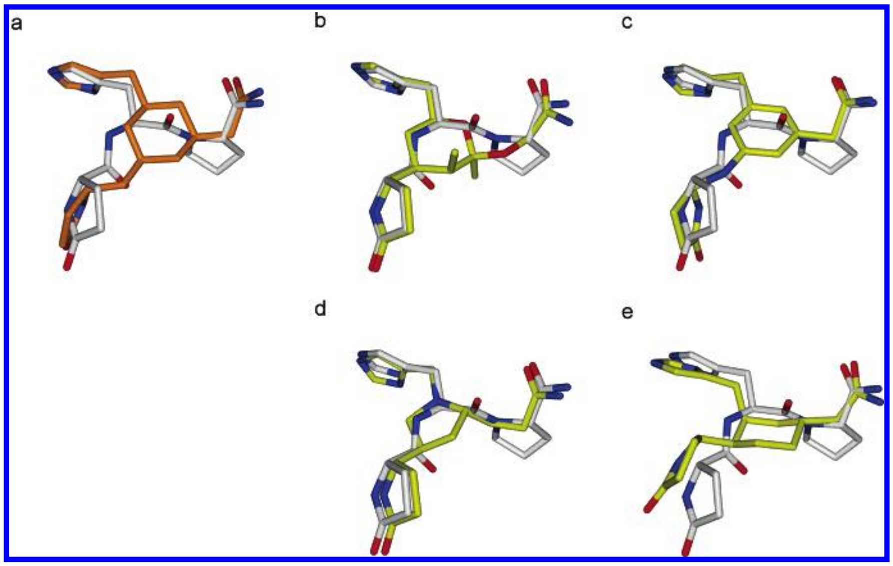

Figure 10. TRH test scenario: (a) peptide mimetic designed by Olson et al.,[23] the remaining structures are solutions generated by Recore, (b) HUWHEM, (c) KIFZAA, (d) SUMVEB, and (e) TCHXET.

analog with a saturated central ring built by Recore is depicted in Figure 10b (CSD entry HUWHEM, rank 12). Compared to the Olson structure, the linker lengths to both the pyroglutamate ring and the amide are varied. The example depicted in Figure 10c (CSD entry KIFZAA, rank 5) indicates that the aromatic ring may be an equally good central scaffold as a saturated ring. The match to the pyroglutamate part of TRH is even slightly better because this substituent can stay in the plane of the aromatic ring. An even better match is achieved with 5-membered rings as central templates. One such solution is depicted in Figure 10c (CSD entry SUMVEB, rank 6). Overall, a high number of interesting cyclic solutions are generated. Figure 10d gives one more cyclic example (CSD entry TCHXET, rank 7), which seems to indicate that a 1,2,4-trisubstituted six-membered ring could also serve as a template for TRH mimetics. We do note that in none of the four solutions in Figure 10b−e the conformations of the generated solutions are fully relaxed. The reasons are (i) that dihedral angles at the newly formed bonds are not checked and (ii) that there may be additional intramolecular clashes upon the connection of a new fragment with the scaffold parts held fixed.

CONCLUSIONS AND OUTLOOK

We have implemented of a new robust method for scaffold hopping based on defined bonds in query conformations.

In contrast to prior methods, Recore is able to search for pharmacophore-type features and exit vectors without fixed tolerance ranges. The main idea for the fragment generation is the enumeration of combinations of cut points to preserve conformational information while still being able to filter out unwanted fragment structures already during database build-up. The k-nearest-neighbor search—together with the voting scheme for queries with multiple features—leads to very short response times.

We have shown that Recore is able to rediscover a variety of known ligand topologies for three different targets. Valid new scaffold topologies were generated even for difficult (multiple vector) queries. Current rank-ordering is based on geometry only. This ordering does not imply an inherent prioritization of solutions. Small deviations from the query exit vectors might be tolerated or even cause improved binding affinity. The selection of appropriate new scaffolds requires thus browsing through a number of top-ranked solutions to get an impression on the diversity of the solutions. One way to reduce this effort (we are showing the top 10 solutions in the Supporting Information in order to demonstrate this) is topology based clustering, which we are currently performing with MOE. This reduces the redundancy of solutions and can yield valuable additional information on the conformational flexibility of a fragment and the reliability of a specific conformation. We are convinced that Recore sets the stage for a better exploitation of the large body of small molecule crystal structure data available in the CSD. Recore produces mostly relaxed conformations of scaffolds that require only a final force field minimization step to yield structures suitable for visual inspection and further interactive modeling. The test scenarios have also demonstrated that it will be necessary to implement additional filters for postprocessing the output. For example, it could be useful to further analyze the validity of the torsion angles at the newly created bonds by comparing them to a torsion angle library. Also, when designing new scaffolds additional knowledge on shape or binding pocket constraints are often available. It could be helpful to be able to add further constraints to the search that could focus the searches even further. In particular, the typical task of replacing an acyclic linker with a ring system could be simplified by directly adding pharmacophore-type ring constraints. We will continue to refine our method in this sense. The software will be made commercially available during summer 2007. For further information see http://www.zbh.uni-hamburg.de/recore.

Supporting Information Available: Hits found by Recore and discussed in the Results section and the first ten ranked hits. This material is available free of charge via the Internet at http://pubs.acs.org.

REFERENCES AND NOTES

(1) Fischer, J.; Breuer, E.; Gaviraghi, G.; Lindberg, P.; Senn-Bilfinger, J.; Proudfoot, J. R.; Wermuth, C. G. *Analogue-based drug discovery*; Wiley-VCH: Weinheim, 2005.

(2) Boehm, H.-J. Scaffold Hopping. *Drug Discovery Today: Technol.* **2004**, *1*, 217−224.

(3) Schneider, G.; Lee, M.-L.; Stahl, M.; Schneider, P. De novo design of molecular architectures by evolutionary assembly of drug-derived building blocks. *J. Comput.-Aided Mol. Des.* **2000**, *14*, 487−493.

(4) Rarey, M.; Stahl, M. Similarity searching in large combinatorial chemistry spaces. *J. Comput.-Aided Mol. Des.* **2001**, *15*, 497−520.

(5) Cramer, R. D.; Jilek, R. J.; Guessregen, S.; Clark, S. J.; Wendt, B.; Clark, R. D. "Lead Hopping". Validation of topomer shape similarity as a superior predictor of similar biological activities. *J. Med. Chem.* **2004**, *47*, 6777−6791.

(6) Lengauer, T.; Lemmen, C.; Rarey, M.; Zimmermann, M. Novel technologies for virtual screening. *Drug Discovery Today* **2004**, *9*, 27−34.

(7) Ertl, P. Cheminformatics analysis of organic substituents: Identification, of the most common substituents, calculation of substituent properties, and automatic identification of drug-like bioisosteric groups. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 374−380.

(8) Lewell, X. Q.; Jones, A. C.; Bruce, C. L.; Harper, G.; Jones, M. M.; McIay, M.; Bradshaw, J. Drug rings database with web interface: A tool for identifying alternative chemical rings in lead discovery programs. *J. Med. Chem.* **2003**, *46*, 3257−3274.

(9) Ho, C. M. W.; Marshall, G. R. SPLICE: A program to assemble partial query solutions from three-dimensional database searches into novel ligands. *J. Comput.-Aided Mol. Des.* **1993**, *7*, 623−647.

(10) Pierce, A. C.; Rao, G.; Bemis, G. W. BREED: Generating, novel inhibitors through hybridization of known ligands. Application to CDK2, P38 and HIV protease. *J. Med. Chem.* **2004**, *47*, 2768−2775.

(11) Lauri, G.; Bartlett, P. A. CAVEAT: A program to facilitate the design of organic molecules. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 51−66.

(12) Taylor, R. Life-science applications of the Cambridge structural database. *Acta Crystallogr., Sect. D: Biol. Crystallogr.* **2002**, *D58*, 879−888.

(13) Allen, F. H.; Davies, J. E.; Galloy, J. J.; Johnson, O.; Kennard, O.; Macrae, C. F.; Mitchell, E. M.; Mitchell, G. F.; Smith, M.; Watson, D. G. The development of versions 3 and 4 of the cambridge structural database system. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 187−204.

(14) Hjaltason, G. R.; Samet, H. In *Ranking in Spatial Databases*; Proceedings of the 4th Symposium on Spatial Databases, 1996; 1996; pp 83−95.

(15) Rarey, M.; Kramer, B.; Lengauer, T.; Klebe, G. A fast flexible docking method using an incremental construction algorithm. *J. Mol. Biol.* **1996**, *261*, 470−489.

(16) Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. RECAP - Retrosynthetic, combinatorial analysis procedure: A poweful new technique for indentifying privileged molecular fragments with useful applications in combinatorial chemistry. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 511−522.

(17) Rarey, M.; Dixon, J. S. Feature trees: A new molecular similarity measure. *J. Comput.-Aided Mol. Des.* **1998**, *12*, 471−490.

(18) Guttman, A. In *R-trees: a dynamic index structure for spatial searching*; SIGMOD '84, 1984; 1984; pp 47−57.

(19) White, D. A.; Jain, R. Similarity Indexing: Algorithms and Performance. In *Storage and Retrieval for Image and Video Databases*; 1996; pp 62−73.

(20) Sadowski, J.; Schwab, C. H.; Gasteiger, J. *Corina, 2.1*; Molecular Networks GmbH Computerchemie: Erlangen, 1998.

(21) Chrusciel, R. A.; Strohbach, J. W. Non-peptide HIV protease inhibitors. *Curr. Top. Med. Chem.* **2004**, *4*, 1097−1114.

(22) Proudfoot, J. R. Non-nucleoside inhibitors of HIV-1 reverse transcriptase. *Expert Opin. Ther. Pat.* **1998**, *8*, 971−982.

(23) Olson, G. L.; Cheung, H.-C.; Chiang, E.; Madison, V. S.; Sepinwall, J.; Vincent, G. P.; Winokur, A.; Gary, K. A. Peptide mimetics of thyrotropin-releasing hormone based on a cyclohexane framework: Design, Synthesis, and Cognition-enhancing properties. *J. Med. Chem.* **1995**, *38*, 2866−2879.

CI060094H