J Comput Aided Mol Des (2008) 22:571–578
DOI 10.1007/s10822-008-9206-7

Visual exploration of structure–activity relationship using maximum common framework

Sung Jin Cho · Yaxiong Sun

Received: 19 December 2007 / Accepted: 23 February 2008 / Published online: 13 March 2008
© Springer Science+Business Media B.V. 2008

Abstract To help tracking all molecules made in a typical medicinal chemistry project, we have developed an algorithm to generate a maximum common framework (MCF) hierarchy and an interactive tool for its visualization and analysis. By identifying all unique frameworks for a set of molecules and all molecules containing each framework, we were able to simplify the MCF hierarchy build up steps and, as a result, speed up the entire process significantly. By allowing compounds to be assigned to multiple MCFs, users can easily remove bad branching nodes and concentrate on interesting ones. MCF hierarchies provide an effective and intuitive visualization for tracking medicinal chemistry lead optimization projects. We will provide examples to illustrate its usefulness.

Keywords Visual exploration · Structure–activity relationship · Maximum common framework

Introduction

Lead optimization by medicinal chemistry in drug discovery typically involves making large numbers of analogs around a particular chemical scaffold in order to identify a clinical candidate with a desired profile of on-target and off-target activities, pharmacokinetic and pharmacodynamic properties, as well as various pharmaceutical and toxicological properties. Balancing all these often time competing goals makes this a challenging team effort. As a project progresses, the number of compounds involved and activities associated with them can increase significantly. To help organize structural and biological information, we have developed an algorithm that creates a maximum common framework (MCF) hierarchy as well as an interactive visualization tool.

The molecular framework (Fig. 1), defined as one or more ring systems with connecting linkers, was first utilized by Bemis and Murcko [2] in their investigation of common features existing in drug molecules and was found to be a useful structural classification method. This idea was later applied by Wilkens et al. [3] to classify HTS hits through common chemical features and to create a navigable framework hierarchy. Recently, Schuffenhauer et al. [4–6] extended Wilkens et al. approach by introducing prioritization rules during the scaffold formation to remove less characteristic, peripheral rings (including fused rings) first in order to eliminate chemically nonintuitive scaffolds. This allowed a unique hierarchical classification of scaffolds and simplified data analysis by assigning a scaffold to one branch in a hierarchy only. The problem with this approach, however, is that the resulting hierarchy is very sensitive to the prioritization rules and more often than not needs adjustments whenever a mistake in a rule is found or addition is required. We feel that a better approach is to allow a scaffold to be assigned to more than one class as in Wilkens et al. approach but limit the possible framework hierarchies to simplify the analysis. One way to reduce the number of framework hierarchies is to create a MCF hierarchy. Rather than identifying hierarchical relationships of all possible molecular frameworks, hierarchy formation can be limited only to MCFs to avoid the time

S. J. Cho · Y. Sun
Molecular Structure, Amgen, One Amgen Center Drive, Thousand Oaks, CA 91320, USA

S. J. Cho (✉)
CHDI, Inc., 6080 Center Drive, Suite 100, Los Angeles, CA 90045, USA
e-mail: sungjin.cho@chdi-inc.org

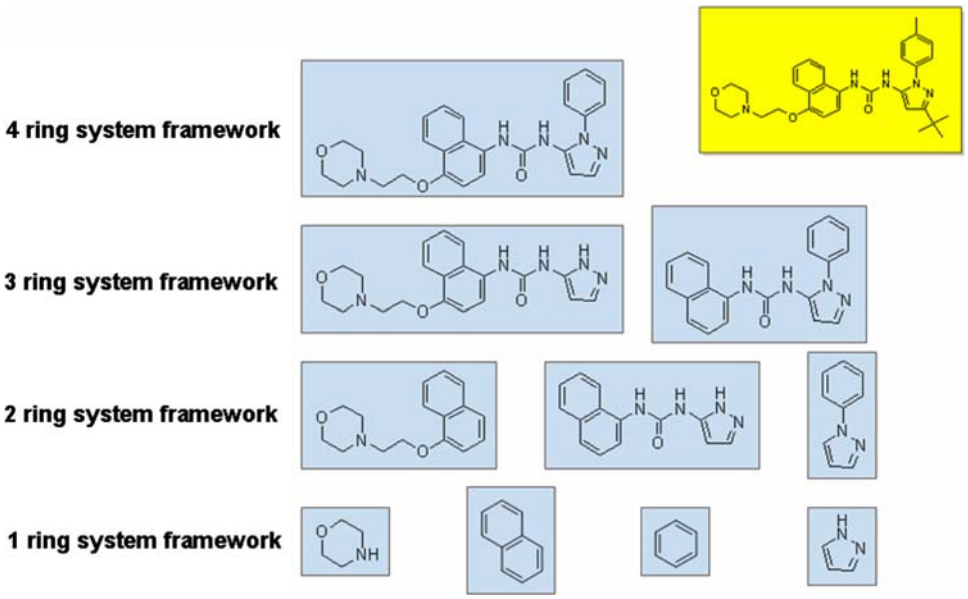

Fig. 1 Examples of 1, 2, 3, and 4 ring system frameworks using p38 MAP kinase inhibitor BIRB 796 [1]

consuming, full superstructure searches. Similar to maximum common substructures, maximum common frameworks are defined as the biggest frameworks found among unique frameworks under consideration. By identifying unique frameworks and finding out which compounds contain each unique framework, we were able to simplify a MCF hierarchy build up steps and speed up the entire process significantly without need for specific rules. For a test case of 2,000 compounds, the entire process of identifying all frameworks, building a MCF hierarchy, and identifying side chain attachment points on MCFs took about 24 and 12 s on 1.7 and 3.0 GHz Xeon processors, respectively (it was reported that a similar process excluding identification of side chain attachment points took 6 min on a 2.5 GHz Xeon processor [3]). An interactive system was also developed to visualize and navigate the results.

Methods

Maximum common framework building algorithm

The recursive algorithm published by Wilkens et al. [3] generates all possible frameworks for a list of molecules. Briefly, the recursive algorithm works by identifying all ring systems (defined as a ring and rings sharing a bond or atom) and generating fragments by systematically deleting each ring system. If the resulting fragments contain more than one ring system, the process is recursively repeated until each final fragment contains only one ring system (Fig. 1). For the entire list, all unique frameworks are identified and for each unique framework, a list of molecules containing it is saved (Fig. 2). Rather than performing superstructure searches against all other frameworks as reported by Wilkens et al. a sorted list of molecules for each framework can be used to generate the MCF hierarchy recursively. Figure 3a illustrates the entire process, and Fig. 3b shows a hierarchy building process in more detail. Briefly, the number of compounds containing each framework is used to rank order frameworks. If there is a tie, the framework with the most number of atoms wins. The first framework which contains the parent node's MCF as its substructure becomes the MCF. The process starts by calculating the rank of jth framework at ith node using the following equation:

$$R_{ij} = S_{ij} + N_j/1,000 \quad \text{if } (N_j > N_{\min} \text{ and } N_j > N_{\text{parent}})$$

$$R_{ij} = 0 \quad \text{if } (N_j \Leftarrow N_{\min} \text{ or } N_j \Leftarrow N_{\text{parent}})$$

where S_{ij} is the number of compounds containing the MCF of ith node and jth framework, N_j is the number of atoms in jth framework, $N_{\min}$ is the minimum number of atoms allowed, and N_{parent} is the number of atoms in the MCF of the parent node. If the current node is the root node and S_{ij} of the highest rank is equal to the number of compounds, jth framework becomes the MCF of the root node. If the current node is not the root node, the highest ranking framework which contains the parent node's MCF as its substructure is identified and becomes the MCF of a new node. The process is recursively repeated using the new non-terminal node as a starting point. The process stops when all compounds are assigned. $N_{\min}$ controls the initial size of a MCF in order to remove a trivial MCF such as a benzene ring, and N_{parent} is used to filter MCF candidates. Substructure searches are performed on high ranking frameworks to ensure the parent MCF is a substructure of a

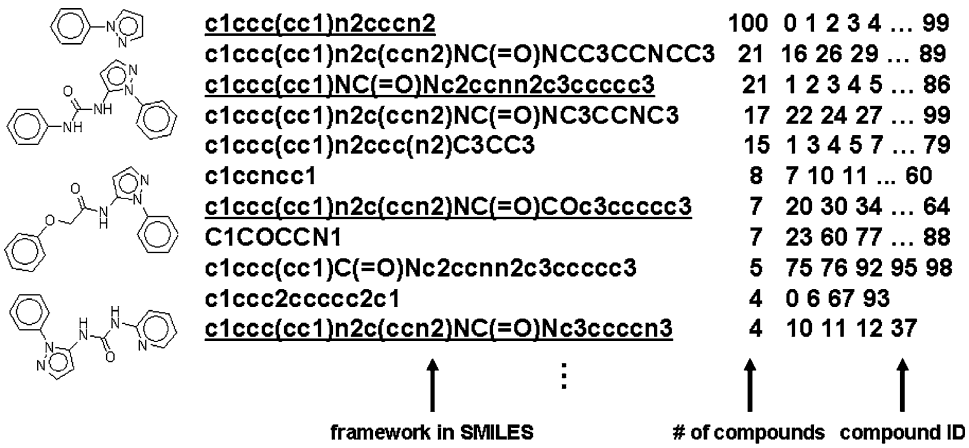

Fig. 2 Sorted frameworks

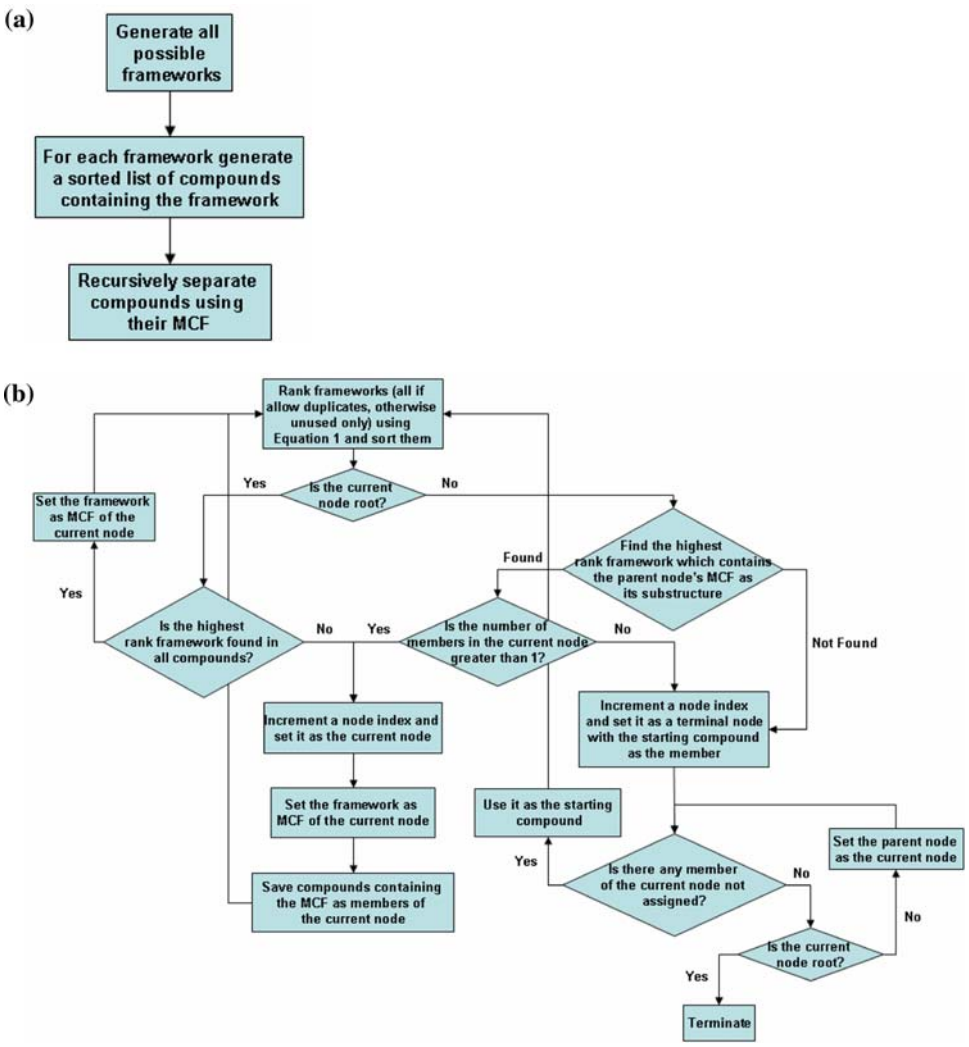

Fig. 3 Flow diagram: (**a**) the overall process; (**b**) MCF hierarchy building process

new MCF. Since the substructure searches need not be performed for all frameworks, the process is extremely fast. Finally, each MCF is compared with its member compounds, and all the attachment points are identified.

MCF hierarchy viewer

In order to fully utilize the MCF hierarchy, we have developed an interactive visualization system. Node class

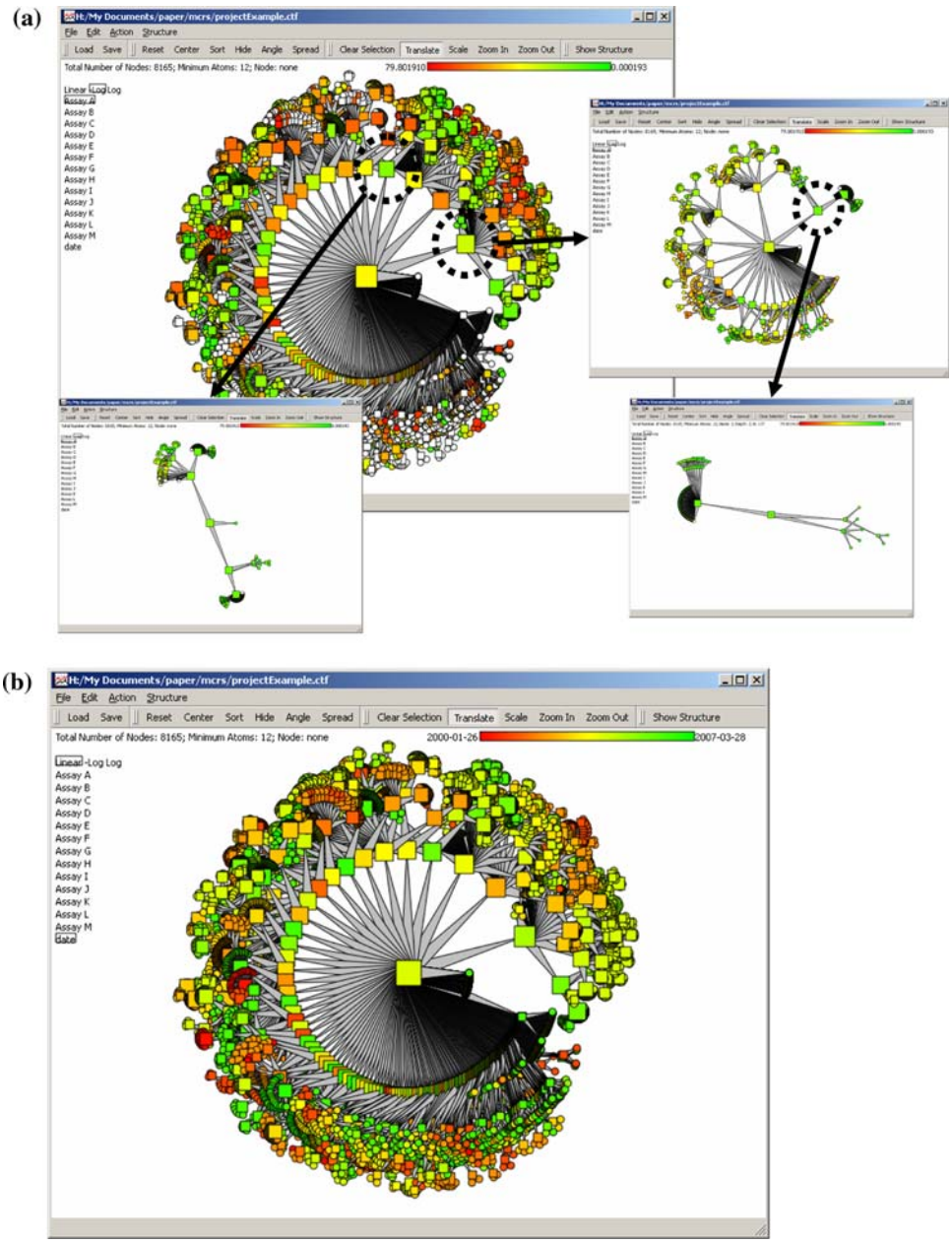

Fig. 4 MCF hierarchy generated using one of Amgen lead optimization project: (**a**) examples of zooming in; (**b**) color coding nodes using registration dates

was created to store all the structure and activity information. The MCF hierarchy can be displayed by recursively drawing each node from the position of the parent node. Color is used to map multiple activities, properties, or any information user wants to associate with MCFs (Fig. 4). A user can zoom in and out of nodes and can translate, rotate, and scale the MCF hierarchy. Circles and rectangles represent terminal (contains an individual structure) and intermediate (contains a MCF) nodes, respectively. Clicking a node reveals a movable structure display box which can be positioned anywhere in the window for optimal viewing. A line is automatically added to indicate which node the display box represents. All structures found in any node can be also displayed in a separate window. A user can hide nodes by specifying node numbers or the minimum number of compounds found. For example, setting 5 to the minimum number of compounds found will only display nodes with the number of compounds greater than 5. A branch spread angle and pattern control the layout of the MCF hierarchy. The branch spread angle sets a branch spread range. The angle value of 360° will allow the branches to spread in all directions, and the default value of 180° will allow the branches to point outward away from the center node. Thee options exist for the branch spread pattern: (1) no weight, (2) use the number of branches, and (3) use the number of compounds. No weight option sets branches to be spaced equally apart. The number of branches or compounds can be used to

weight spacing so that more branches or compounds under a node will lead to bigger branch spacing. The distances between nodes are calculated from the node distances found in the previous layer. For example, let's say the distance between the center node and the nodes found in the first layer is ND_i, which is determined by the size of the display area. The non-terminal node distances found in the next layer, ND_{i+1}, are simply $0.5 \times ND_i$, and the terminal node distances are $0.25 \times ND_i$. Branches can be also sorted by either the size of branches or the similarity of frameworks. Using the size allows the branches with more compounds to be formed first. Using the similarity allows the similar frameworks to be appeared first. This is done by generating the near neighbor list of each framework and simply looking up which framework is the next unused one in the list. A user can use a list of compound IDs to generate the MCF hierarchy, or a table containing the IDs and biological data can be used. Each node can be colored to reflect activity or property values. Clickable menus will be appeared on the left top corner to let a user specify which transformation and data type to be displayed.

Implementation

Framework generator and MCF hierarchy builder have been implemented as C/C++ programs using OpenEye's OEChem toolkit [7]. MCF hierarchy viewer is written in Qt [8] and is a part of ADAAPT system [9].

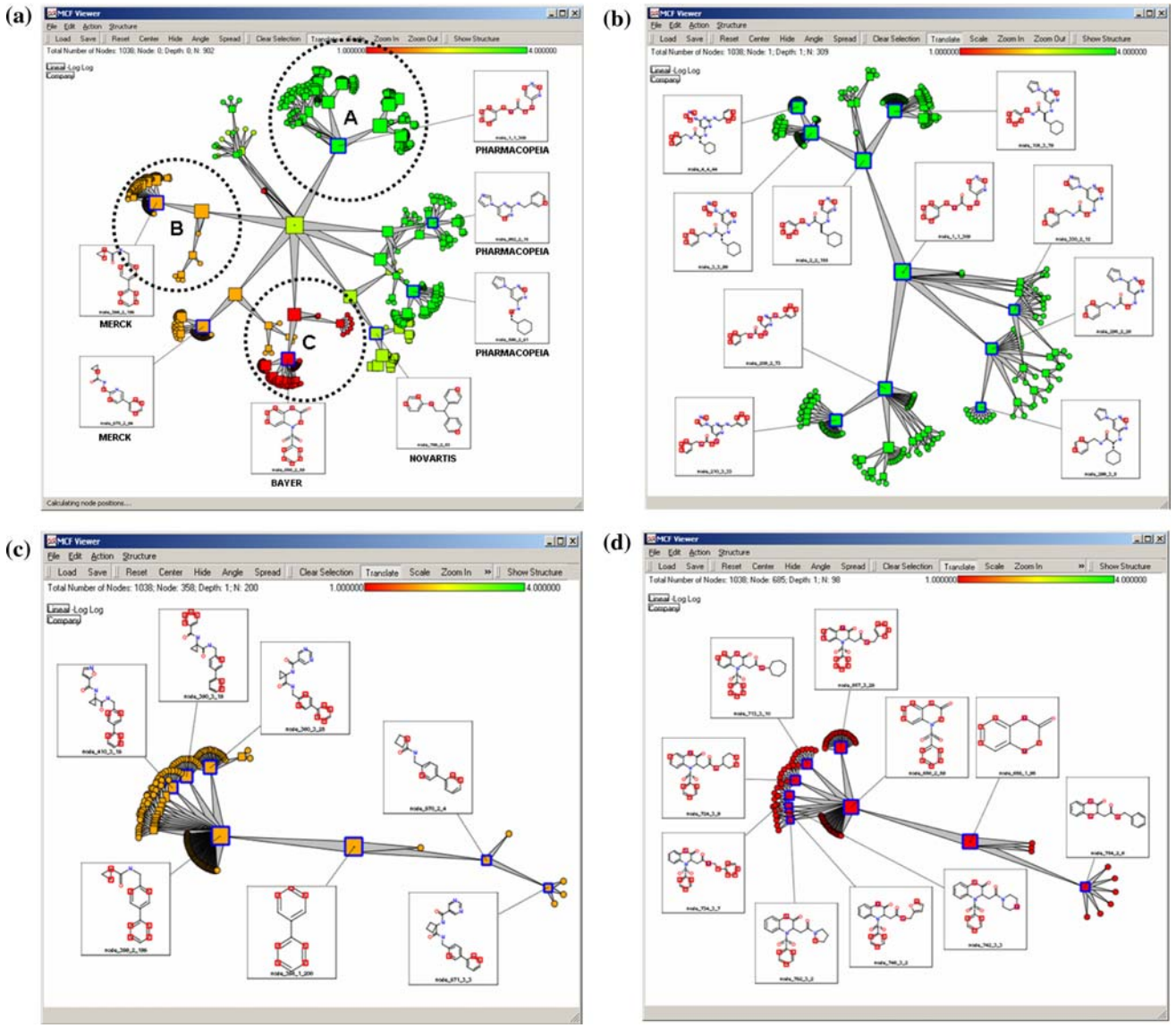

Fig. 5 MCF hierarchy generated using the patent data of B1 antagonists: (**a**) the entire MFC hierarchy; (**b**) zoomed in view of A region; (**c**) zoomed in view of B region; (**d**) zoomed in view of C region

Results and discussion

Figure 4 shows an example of MCF hierarchy of an actual Amgen lead optimization project. Assay description, MCF, and structural information are withheld for a legal reason. Column headers and transformation types appear on the left side. Users can quickly identify which chemotypes were studied most by examining the spread angle and quickly navigate through any chemotypes of interest. When activity data is used to color code nodes as shown in Fig. 4a, users can reveal quickly which nodes contains active, selective, or metabolically stable compounds. Users can then zoom in to a node of interest, and study how a particular chemotype has been explored. When registration dates are used to color code nodes (Fig. 4b), users also recognize which chemotypes are currently under investigation and which are not.

Another example using the patent data of B1 antagonists (patent data is obtained from Target Inhibitor Databases [10]) is shown in Fig. 5. Nodes are color coded by patent issuers. Figure 5b–d show zoomed in view of Pharmacopeia, Merck, and Bayer chemotypes, respectively. Using the MCF hierarchy viewer, chemists can easily deduce which companies are working in this area and which chemotypes are patented. It gives a good overview of the B1 antagonist patent space. Users can also concentrate on a specific chemotype by zooming in and analyze deeper layer. By automatically identifying each attachment point (boxed in red), a user can identify modification points easily. In short how they explored a particular chemotype can be captured visually.

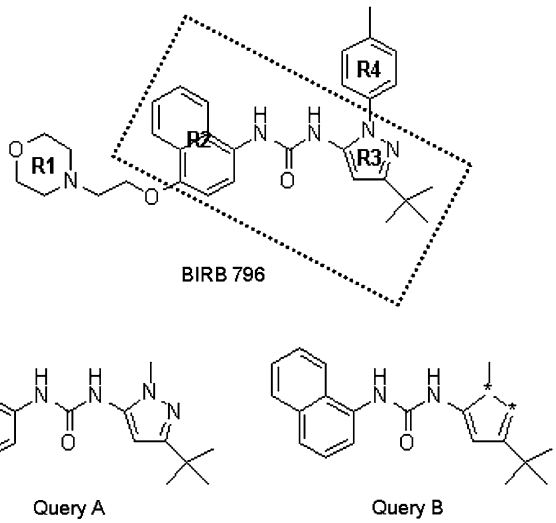

Fig. 6 Two query structures used in substructure searches

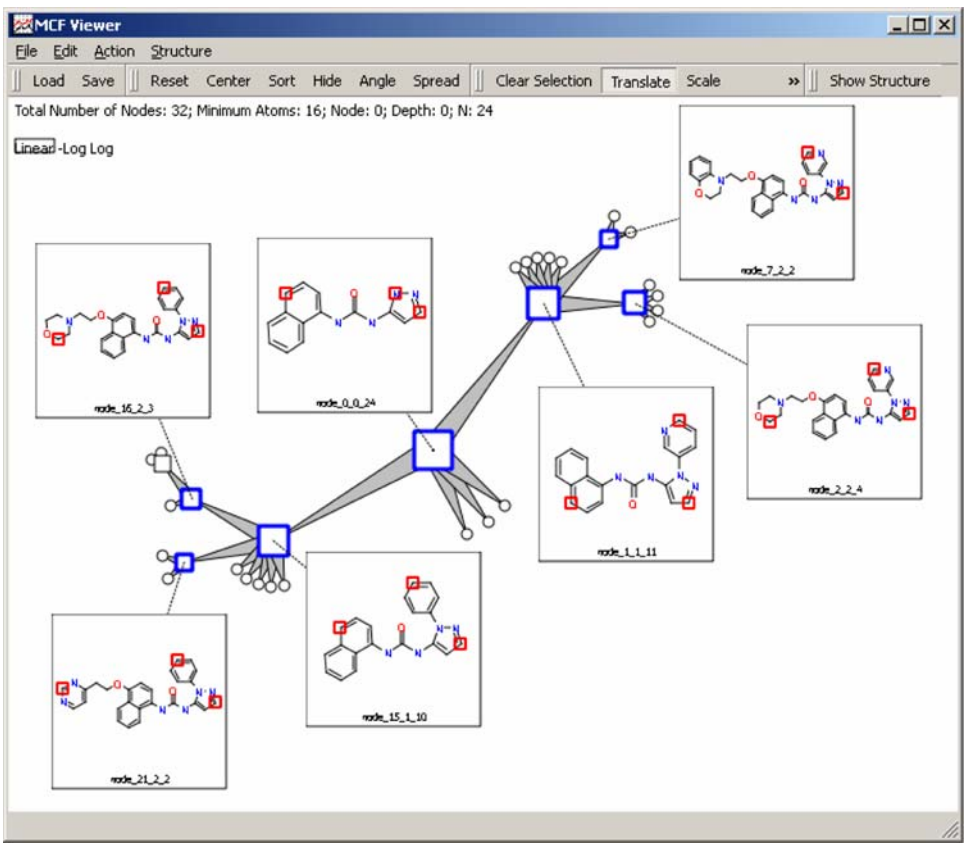

Fig. 7 MCF hierarchy generated using the substructure search result of query A in Fig. 6

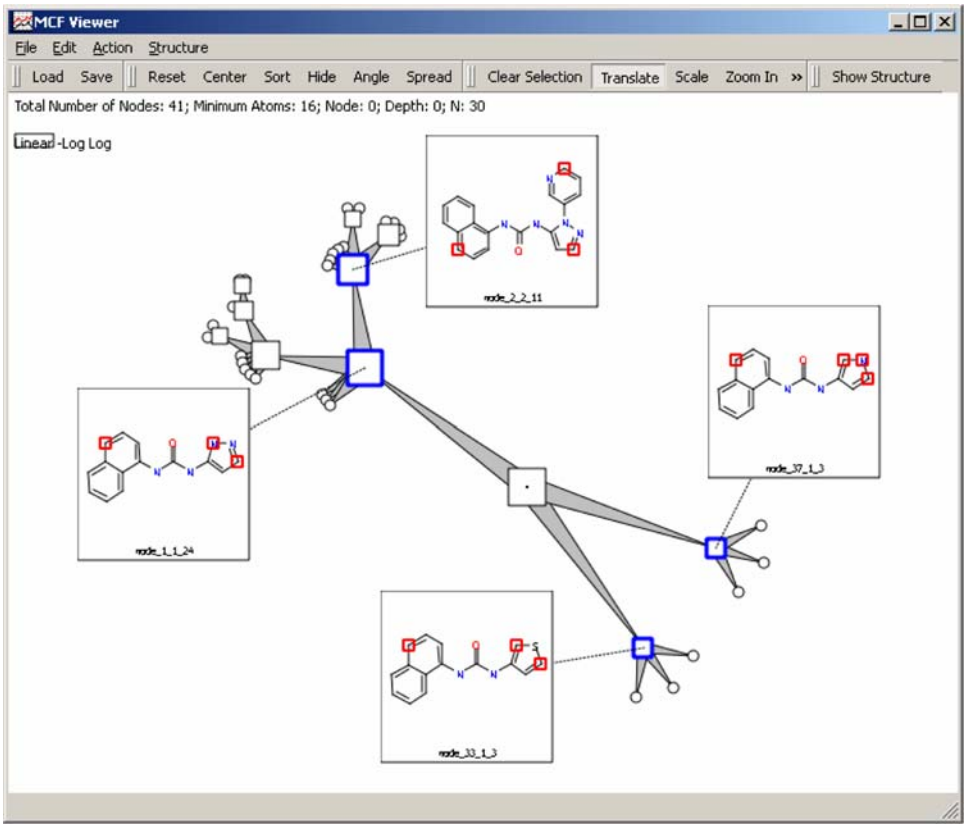

Fig. 8 MCF hierarchy generated using the substructure search result of query B in Fig. 6

While substructure searching is a routinely used in lead optimization, MCF hierarchy provides an alternative way that is more flexible and comprehensive when organizing large number of compounds. To illustrate this, two substructure searches against MDL Drug Data Report (MDDR) [11] database were performed using two query structures generated from compound BIRB 796 (Fig. 6). Figure 6 shows two query structures which were generated by removing 2-(4-morpholinyl) ethanol (R1 and a ethanol connected to R2) and methyl benzene (R4) from BIRB 796. The query B is identical to the query A except for two wildcards in place of two nitrogens in a pyrazole ring (R3). The search results were used to generate two MCF hierarchies (Figs. 7, 8). Two main branches in Fig. 7 are due to two variations, pyridine and benzene, in R4. Additional variation was observed in R1 region and is shown clearly from the figure. When the query B is used to search, the results contain two more variations in R3 region, thiophene and pyrrole, in addition to variations observed in Fig. 7.

Conclusions

We have described an algorithm and a visualization system to generate chemically intuitive hierarchy using MCFs. Examples showing how this system can aid a lead optimization programs have been illustrated. Color coding, movable structure display boxes, and highlighting attachment points were found to be especially useful in navigating the hierarchy for lead optimization, patent summary, or organizing substructure searches. Distinguishing a chemotype from a side chain is still a problem, but by allowing compounds to be assigned to multiple MCFs, users can hide unwanted branching nodes and concentrate on interesting ones. Building a hierarchy using MCFs is an excellent way to keep track of a lead optimization effort.

References

1. Pargellis C, Tong L, Churchill L, Cirillo PF, Gilmore T, Graham AG, Grob PM, Hickey ER, Moss N, Pav S, Regan J (2002) Nature Struct Biol 9:268–272
2. Bemis GW, Murcko MA (1996) J Med Chem 39:2887–2893
3. Wilkens SJ, Janes J, Su AI, Hier S (2005) J Med Chem 48:3182–3193
4. Schuffenhauer A, Ertl P, Roggo S, Wetzel S, Marcus AK, Waldmann H (2007) J Chem Inf Model 47:47–58
5. Schuffenhauer A, Brown N, Ertl P, Jenkins JL, Selzer P, Hamon J (2007) J Chem Inf Model 47:325–336
6. Koch MA, Schuffenhauer A, Scheck M, Wetzel S, Casaulta M, Odermatt A, Ertl P, Waldmann H (2005) PNAS 102:17272–17277

7. The OEChem toolkit is available from OpenEye Scientific Software, Inc. http://www.eyesopen.com
8. The Qt class library and tools are available from Trolltech Inc. http://www.trolltech.com
9. Cho SJ, Sun Y, Harte W (2006) J Comput-Aided Mol Des 20:249–261
10. Target Inhibitor Databases are available from GVK Biosciences Private Limited. http://www.gvkbio.com
11. Molecular Design Drug Data Report, version 2005.2. Molecular Design Ltd., San Leandro, CA