

The centroidal algorithm in molecular similarity and diversity calculations on confidential datasets

Sergey Trepalin^a & Nikolay Osadchiy^{b,*}

^a*Russian Academy of Science, Institute Physiologically Active Compounds, 142432, Chernogolovka, Moscow region, Russia;* ^b*ChemDiv, Inc., 11558 Sorrento Valley Rd., San Diego, CA 92121, USA*

Received 17 May 2005; accepted 13 October 2005
© Springer 2005

Key words: algorithm, centroid, diversity, security, similarity, small molecules

Summary

Chemical structure provides exhaustive description of a compound, but it is often proprietary and thus an impediment in the exchange of information. For example, structure disclosure is often needed for the selection of most similar or dissimilar compounds. Authors propose a centroidal algorithm based on structural fragments (screens) that can be efficiently used for the similarity and diversity selections without disclosing structures from the reference set. For an increased security purposes, authors recommend that such set contains at least some tens of structures. Analysis of reverse engineering feasibility showed that the problem difficulty grows with decrease of the screen's radius. The algorithm is illustrated with concrete calculations on known steroidal, quinoline, and quinazoline drugs. We also investigate a problem of scaffold identification in combinatorial library dataset. The results show that relatively small screens of radius equal to 2 bond lengths perform well in the similarity sorting, while radius 4 screens yield better results in diversity sorting. The software implementation of the algorithm taking SDF file with a reference set generates screens of various radii which are subsequently used for the similarity and diversity sorting of external SDFs. Since the reverse engineering of the reference set molecules from their screens has the same difficulty as the RSA asymmetric encryption algorithm, generated screens can be stored openly without further encryption. This approach ensures an end user transfers only a set of structural fragments and no other data. Like other algorithms of encryption, the centroid algorithm cannot give 100% guarantee of protecting a chemical structure from dataset, but probability of initial structure identification is very small—order of 10^{-40} in typical cases.

Introduction

Molecular similarity and diversity calculations are currently widely used in new small molecule drug design, especially in compounds selection for high-throughput screening campaigns and combinatorial libraries design.

These methods are based on the similar property principle [1], stating that structurally similar molecules should exhibit similar physico-chemical and biological properties.

Structural similarity and nearest neighbors searches are well known for several years and together with substructure searches included to the most of currently commercially available chemical database management systems, such as *MDL ISIS/Host* [2], *Daylight Database Package* [3], *CambridgeSoft ChemFinder* [4], *Oxford*

*To whom correspondence should be addressed. Fax: 858-794-4931; E-mail: no@chemdiv.com

Molecular's RS 3 Discovery [5], *Synopsys Accord* [6] and *Tripes UNITY* [7].

Similarity calculations together with other SAR techniques are used mainly at the lead optimization stage and for the targeted and hit-to-lead libraries design.

Diversity analysis benefits the molecular libraries design for screening and lead generation. Review of commercially available software for this task can be found in work [8].

There are a number of approaches to diversity and similarity analysis different in sets of descriptors used, similarity and diversity measures and compounds selection algorithms. Descriptors sets may include topological indices, physical properties descriptors, 2D and 3D structural fragments, etc.

2D structural fragments are currently the most common descriptors for diversity and similarity analysis. Quite a few papers showed that 2D fragments in most cases outperform other types of descriptors by selectivity between biologically active and inactive compounds [9–11].

There are different types of atom centered, bond centered, and ring centered two-dimensional structural fragments [12]. Structural keys usually work with a predefined fragment dictionary. Examples of structural keys include *CAS ON-LINE Screen Dictionary* [13] and *MDL MACCS keys* [2]. Another type of 2D fragment descriptor is a set of fragments automatically generated for a molecule or a molecular library. This type of descriptors usually includes augmented atoms, fixed length atom sequences, and ring fragments, subsequently coded into fixed length bit-string with a hash-function using pseudo-randomizing algorithms. Examples of these descriptors can be *Daylight fingerprints* [3] and *Tripes UNITY fingerprints* [7]. Often used descriptors also include atom pairs [14], topological torsion [15], and autocorrelation indices [16].

Screens descriptor

An alternative to various fingerprints is the real structural fragments collected specifically from each reference set for the diversity and similarity calculations [2, 13]. This approach is free of some drawbacks inherent to fingerprints (which are also calculated from screens [3]) as well as predefined

keys. For instance, due to a fixed fingerprint's length, one bit in the bit string could correspond to two or more structural fragments, thus being a source of inaccuracies of similarity and diversity calculations. Using predefined keys, due to their finite and limited number, may show no difference between two molecules if their different fragments were not included into the key list. We call a set of automatically generated structural fragments a Screens Descriptor or Screens.

Since screens are structural fragments, their usage becomes clear to chemists. Screens can be stored in an open format, such as SDF file [17]. Open format clarifies any questions about the data leaving company's premises: a chemist can see them and assess possibility of reverse engineering of initial molecules from the set of screens.

As we said, there are several different ways to generate screens, for instance, take all linear fragment of given length [3] or take two-dimensional atom-centered fragments of given topological radius [18], as well as some others. However, the second way has a number of distinctive preferences. Firstly, it adequately describes chemical neighborhood of a central atom, with an accuracy increasing with fragments' radius. Secondly, there is a simple algorithm Figueras [19] available for these screens generation. This algorithm rapidly finds adjacent atoms and bonds. In addition to the bond's type and aromaticity, screens might contain information on bonds' topology, including cycles of 6 and more atoms (optionally). Therefore, a single bond in five-membered ring does not equal to a single bond in six-membered ring. However, single bonds in seven- and eight-membered rings are considered the same. This is consistent with the chemical data showing that properties and reactivity of molecular bonds in large cycles do not change with cycle size increase. One can easily estimate the number of such screens per molecule as:

$$N_{\text{Screens}} = N_{\text{Atoms}} * \text{TOPORAD},$$

where N_{Screens} – number of screens in molecule, N_{Atoms} – number of atoms in molecule, TOPORAD – maximum topological radius of screens. Example of screens of radius 2 generated for 3-methylcyclohexanol is shown at the Figure 1. There are 16 screens total. Generation algorithm and its results are described in the work [20].

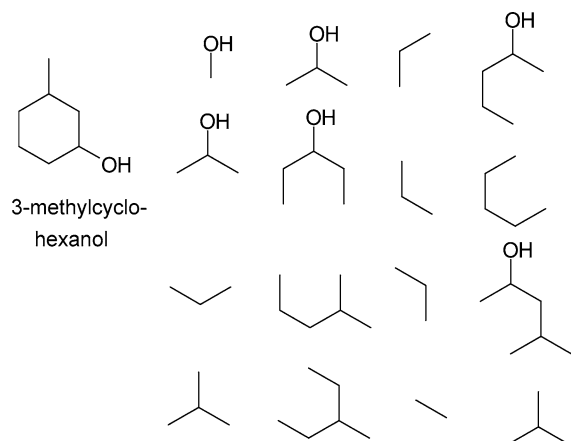


Figure 1. Screens of radius 2 generated for 3-methylcyclohexanol.

If we are to consider the central atom (this is necessary for atomic properties predictions such as ^{13}C shift, charge, etc. [21, 22]) then every screen on the Figure 1 is unique. However, the set of screens constructed this way may contain degenerated, i.e. coinciding screens. Degenerated screens can occur in case of symmetric molecule considering the central atom or for asymmetric molecules omitting atom-centered properties. Diversity or similarity calculations do not use atom-centered properties. Therefore, actual number of unique screens generated for 3-methylcyclohexanol is 11, see Figure 2. To collect a set of unique set of screens for similarity and diversity studies it is sufficient to

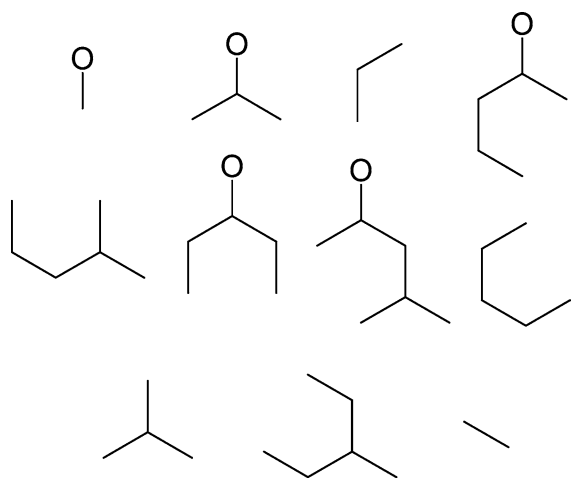


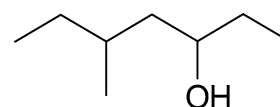
Figure 2. Undegenerated screens for 3-methylcyclohexanol.

compare their structures and eliminate duplicates, as the central atom makes no difference in that case.

Structure comparison uses the structure sensitive indices for preliminary filtration. Index pair sensitive to the molecule topology was determined as reported in [23]. Exact molecular weight is calculated and stored in four-byte *single* type floating point variable. This variable is used as a filter for molecular formula. We have generated all possible combinations in the range of $\text{C}_{0...41}\text{N}_{0...35}\text{O}_{0...31}\text{H}_{...NH_{\max}}$ to get formulas with molecular weight of 500 Dalton or less. Maximal number of hydrogens (NH_{\max}) was calculated from the fragment connectivity assumption. It was found, that there are 238767 different molecular formulas in the range. We found only 58 molecular pairs with different constitution but with the same calculated exact molecular weight. We assume that the same values of molecular weight and 2 topological indices means that the pair has two identical molecules. This criterion was considered sufficient, so no further graph-comparison operations were made. The screens are ordered in the generated database descending by Topological indices and molecular weights, which allows for quick bisection search algorithm.

Reverse engineering of the molecule from its screens

This problem can be deterministic only in the simplest cases. For example, in case of 3-methylcyclohexanol considering bond topology and knowing the 6-membered ring bond, it is possible to reconstruct the initial structure unambiguously from the unique screens showed at Figure 2. However, this problem does not have the single solution if we do not consider bonds' topology. 3-Methylcyclopentanol, and 3-methylcycloheptanol and cycles of larger size satisfy the set of screens as well as non-cyclic analogs comprising fragment



It should be noted that every molecule derived from this fragment by adding CH_2 - group to the terminal carbons or having $\text{CH}_2\text{-CH}(\text{CH}_3)\text{-CH}_2$ and $\text{CH}_2\text{-CH}(\text{OH})\text{-CH}_2$ fragments in the long

chain contain the same set of screens as the initial 3-methylcyclohexanol. Therefore, the reconstruction problem has infinite number of solutions.

The problem also may have an infinite number of solutions considering bond's topology. For instance, in case of 3-butylcyclohexanol, the following 2 screens with aliphatic chain labels will be added to the screens at the Figure 2.



However, the 3-pentylcyclohexanol; 1,5-bis(3-hydroxycyclohexyl)pentane as well as compounds with longer aliphatic chains would satisfy the same set of screens.

Additional conditions can be imposed to restrict the infinite solutions number. Such conditions may include maximum molecular weight constraint; however, there is always a chance to miss a correct result by this constraint.

Screens of larger radius may eliminate the ambiguity of reverse engineering problem. Say, using screens of radius 3, one can unambiguously reconstruct 3-butylcyclohexanol. This is possible because this set of screens will not contain long-chain fragments expected for pentylcyclohexanol and its homologs.

Therefore, in some cases, it possible to reconstruct the initial molecule from its planar 2D screens. The probability of successful reverse engineering increases with the screen's radius increase. At the limit when the screen's radius is set to at least a half of longest chain length, the initial molecule will be present in as one of screens.

Using linear screens one can expect increase of reverse engineering problem difficulty. We can imply that if reverse engineering is possible for the linear screens it is also possible for spherical, however we cannot definitely claim that the opposite is also true.

Similarity and diversity

The most popular measures for similarity and dissimilarity calculations or, more generally, distance between two objects are [24] *Tanimoto Coefficient*, *Cosine Coefficient*, *Dice Coefficient*, *Hamming Distance* and *Euclidean Distance*. The calculation requires knowledge of total number of

the same screens (or fingerprints for hashing) for each pair of the objects considered and total number of screens (fingerprint length) for each object. Dissimilarity and similarity relate to each others as:

$$\text{DISSIMILARITY}(I, J) = 1 - \text{SIMILARITY}(I, J) \quad (1)$$

where $\text{SIMILARITY}(I, J)$ is one of the measures mentioned (Tanimoto, Cosine).

In practice, it is often necessary to calculate similarity (or dissimilarity) not only for a pair of chemical structures, but for some chemical structure and a set of molecules. For instance, term "dissimilarity" is defined for a set of compounds. Majority of diversity measures is based on pairwise molecular dissimilarities. These include following distance based diversity functions: *Minimum Intermolecular Dissimilarity*, *Mean Intermolecular Dissimilarity* and *Average Nearest Neighbor Distance*. *Minimum Weight Spanning Trees* [25] also belongs to this class.

The cosine similarity measure is defined as:

$$\begin{aligned} \text{SIMILARITY}(I, J) \\ = \frac{\sum_{K=1}^F M_I(K) \times M_J(K)}{\sqrt{\sum_{K=1}^F M_I(K)^2 \times \sum_{K=1}^F M_J(K)^2}} \end{aligned} \quad (2)$$

The number of components F of these vectors equals to the total number of unique screens in the set, the vector component $M(K) = 1$ if K -th screen is present in the molecule, and $M(K) = 0$ otherwise. *Mean Intermolecular Similarity* is defined as:

$$\begin{aligned} \text{SIMILARITY}(I, \text{DATASET}) \\ = \frac{\sum_{J=1}^N \text{SIMILARITY}(I, J)}{N} \end{aligned} \quad (3)$$

where N – is the number of compounds in the set.

In case of cosine metric and mean similarity this algorithm can be reduced to the centroid algorithm [26] for similarity calculation between a molecule and a set. Then (3) will be substituted by the dot product of two *centroid vectors*:

$$\text{SIMILARITY}(I, \text{DATASET}) = A_I * A_C / N \quad (4)$$

where A_I – vector with the length equal to the total number of the different screens in the set and molecule I . The component $W(M)$ of this vector is

zero if the M -th screen is not present in the molecule I and:

$$W(M) = \frac{1}{\sqrt{L}} \quad (5)$$

where L – the total number of screens in the molecule I.

A_C – is a centroidal vector for the set, it equals to the sum of centroidal vectors for each molecule in the set:

$$A_C = \sum_{J=1}^N A_J \quad (6)$$

Thus, in the centroidal algorithm, instead of calculation and averaging all pairwise similarity coefficients, we calculate a dot product of two centroid vectors. In the classical approach calculation time for the similarity between a molecule and a set is proportional to $N \cdot F$, where N is the number of compounds in the set and F is the total number of unique screens in the set. With centroidal algorithm [26] calculation time for the same problem is proportional only to the total number of screens F .

It should be noted that screens vector as well as centroidal vector are sparse. This allows storing only non-zero components in RAM and efficient calculation of the dot product for centroid vectors. For a ChemDiv's database (over 650,000 small molecules for bioscreening) the number of screens of radius 2 per molecule is in average 34 (see paper [20]). Hence, molecule to the set similarity takes in average 34 multiplications and 34 additions. In case of screens' vectors the total number of multiplications and additions equals to F – the total number of screens that can reach hundreds of thousand. Paper [20] shows the way to further improve performance in the diversity sorting calculations. The algorithms are implemented in ChemoSoft [20] chemoinformatics software suite developed at ChemDiv.

The most important outcome of the centroidal algorithm, however, is that instead of keeping all screens for each molecule, which is necessary for the classic approach, it is sufficient to store the screens only for the entire set, together with the sum of screens' weights (5), (6), and total number of compounds in the set (3).

Reverse engineering of the reference set molecules from their screens and weights in the centroidal algorithm. theoretical approach

Using screens and their weights for the entire set of molecules imposes additional difficulties to the reverse engineering problem:

- (1) Determine the number of screens (including the same ones) in the set;
- (2) Determine the number of screens in the molecule;
- (3) Determine which screens are present in which molecule.

Each of these problems should be solved in order to move to the next in the list. Problem (1) reflects the fact that centroidal algorithm stores the sum of screens' weights, not their amount in the set. Sum of the weights (5), (6) can be calculated from the total number of screens in each molecule having this screen. The situation aggravates, because weight is not always proportional to occurrence frequency of the screen. While one screen has lesser occurrence frequency than the other, it may have higher weight. It happens when the first screen is present in molecules with relatively small number of screens, while the second is present in large molecules.

One can only estimate a range of the screen's occurrence number. Assuming that the molecule is very small (consists of only one screen), number of screens of each type is equal to this screen type's weight. If we are not to limit molecule's size then minimal occurrence frequency for the screen is 1 (present in the set). Under the constraints to the molecular weight and assuming that all screens have the same molecular weight, occurrence frequency for the screen can be estimated as:

$$K = \text{round} \left(W \times \sqrt{\frac{M_w}{M_s}} \right) \quad (7)$$

where M_s is the screen's molecular weight; M_w – maximal molecular weight of compounds in the set.

Another approach to estimate screen's occurrence frequency is: using formulae (5) and (6) find the set of integers so that the sum of their square root reciprocals would be equal to the screen's

weight. In that case, screen's occurrence frequency would be equal to the number of additives in that sum. Evidently, this problem cannot be solved unambiguously because one screen present once in the molecule of 4 screens total contributes the same weight as the same screen present twice in the molecules of 16 screens each:

$$\frac{1}{\sqrt{4}} = \frac{1}{\sqrt{16}} + \frac{1}{\sqrt{16}} \quad (8)$$

To complicate the reverse engineering process further one can decrease accuracy of weight calculations (5,6). From our experience 5–6 decimal points are quite sufficient for similarity and diversity calculations.

Therefore, screen's occurrence frequency search problem cannot be solved deterministically. Consequently, it is impossible to solve problem (2), i.e. estimate the number of screens in the molecule. One can use an estimation of 34 screens per molecule (average for a large dataset), but this presumes that all molecules in the set are approximately of the same size. It is also possible to estimate minimum and maximum number of screens per molecule. Minimum possible number is 1, valid for a small or symmetric molecule. Assuming there is a limit for a maximum molecular weight, maximum number of screens equals to the ratio of maximum molecular weight to the molecular weight of the smallest screen.

Problem 3 (screens distribution in the molecule set) can be solved only stochastically. Given that the molecule contains k screens and the total number of screens in the set is F , the number of combinations of k different screens equals to C_F^k . For a relatively small set of approximately 20 compounds total number of screens reached a few hundred – for a set of known steroidal drugs (see below) there are 324 screens of radius 2. Using an average number of 34 screens per molecule this gives $C_{324}^{34} = 1.28 \times 10^{46}$ combinations. One of these combinations may be correct provided that we guessed the total number of screens in the molecule right. The reconstruction of the structure from its screens, as shown earlier, can also be ambiguous. It also may be possible to reconstruct the structures from a wrong set of screens, thereby creating a lot of noise around the correct structure. It may be impossible to separate this noise from the signal because the modern software is capable

of working only with $\sim 10^7$ structures, but not with 10^{46} .

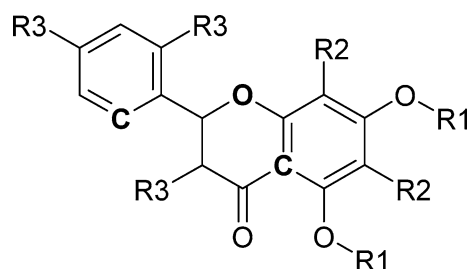
With a number of screens increasing, the number of combinations grows as a factorial; therefore the reverse engineering problem belongs to the NP-complete problems class. However, there are some cases when chemical considerations can provide a shortcut to the reference molecule reconstruction. It happens when the screen's radius exceeds half length of the longest chain in the molecule. In this case, the set of screens can contain the initial molecule, which can be retrieved as the largest fragment in the set.

Reverse engineering of the reference set molecules from their screens and weights in the centroidal algorithm. Scaffold identification in a combinatorial library dataset

We can expect the highest probability of reference structures identification from their screens to happen for a relatively small set of similar molecules. Such sets are pretty common and are known as combinatorial libraries, derived from the same scaffold by various side chains substitution. Obviously, the scaffold of a library will be present in all the molecules throughout the set. Therefore, due to the centroidal algorithm nature, the screens present in the scaffold will have the highest weights. Consequently, one can identify screens pertaining to the scaffold simply by ordering them by their weights and the scaffold can be restored.

A chromanone derivative, pinocembrin (Figure 3) can be used as a scaffold example in attempt to reverse engineer its structure from the set of screens collected from the library of substituted pinocembrin derivatives. Substituents were selected in a way that they correspond to natural pinocembrin's derivatives (Figure 3). It should be noted that the total number of atoms in any substituent does not exceed 15, and it is less than the number of atoms in the scaffold (19).

We generated a library of 128 unique pinocembrin derivatives. The average cosine pairwise similarity between molecules in this library is 0.76. Similarity coefficient was calculated using the screens descriptor and the cosine measure. The library features a very low diversity and also includes the pinocembrin itself ($R1 = R2 = R3 = H$).



R1=H, Me(128)
 R2=H, -CH₂CH=CMe₂(128)
 R3=H, OH(192)

Figure 3. Pinocembrin based scaffold and substituents used for the combinatorial library generation. Three atoms (in bold) of the pinocembrin molecule keep the respective centroid fragments of radius 2 unchanged when a substituent is introduced. Frequency of each non-hydrogen substituent is shown in parenthesis (some molecules contain a number of identical substituents).

We used only radius two screens for the calculations. Screens of radius 4 cannot be used, as they will include the pinocembrin itself. The library of 128 compounds yielded 37 unique screens of radii 1 or 2. The pinocembrin contained 21 different screens. Thus, 127 derivatives add only 16 more unique screens to the screens of the core scaffold.

We compared pinocembrin's screens with the screens gathered from the combinatorial library. There are only three pinocembrin's screens that have the highest weight among screens of the combinatorial library (Figure 4, screens 1–3). This can be attributed to the fact that there are only three atoms in the pinocembrin molecule that keep the respective centroid fragments of radius 2 unchanged when a substituent is introduced. The minimum screen's weight in pinocembrin molecule equals to 12.36 (Figure 3). There are 2 more such screens, which are not shown on the Figure 3 – all of them correspond to the atoms of pinocembrin molecule that have 3 substitution points within topological radius of 2.

Therefore, not every screen of the pinocembrin molecule has the highest weight in the combinatorial library based on this scaffold. The weights are distributed nearly uniformly in a range of 12.36–24.81. This range also has radius 2 screens of some substituents (Figure 4, screen 4). There are 34 screens of the combinatorial library in this range.

The number of possible combinations of 21 screens in 34 screens set equals to $C_{34}^{21} = 9.3 \times 10^8$. This number is 38 orders of magnitude less than the number of combinations in the steroids case. Let's consider what this striking reduction in the number of combinations means practically. It means, that in the worst case, when we have a library of close analogs and large common fragment and provided that we guessed the number of screens of the scaffold right, the probability of selection of a right set of screens is $\sim 10^{-9}$. Furthermore, we should also make a right guess on each screen's occurrence number in the molecule to reconstruct the original pinocembrin. It should be noted, that the result was obtained for the molecules with molecular weight less than 412 (see maximally substituted structure on Figure 3). It seems quite problematic (almost impossible) to reduce the number of molecules further, since all the substituents and the diversity points are represented by the natural bioactive molecules [27].

We have also investigated the possibility of the scaffold identification in a diverse combinatorial library. We selected 127 relatively small substituents: chlorine, vinyl, methoxy-, etc. The largest substituent (*benzoyloxy*) has 9 atoms. In average, a substituent has 4 atoms. Mean pairwise similarity across the substituents is 0.02 (cosine metric). Again, we used pinocembrin scaffold with the same diversity points as above (Figure 3) and generated a library of 128 compounds, including the pinocembrin itself. The resulting library contains 889 substituents. To maximize the diversity

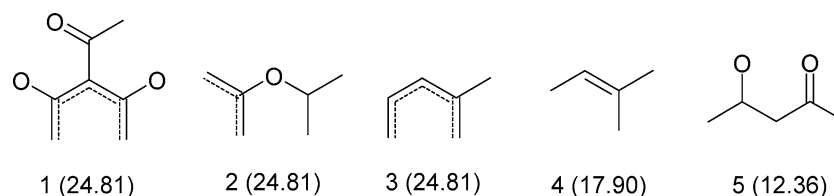


Figure 4. Selected screens and their weights for the centroidal algorithm calculations of pinocembrin derivatives. Screens 1–3, 5 belong to pinocembrin. Screen 4 belongs to a substituent.

of the library, each substituent must be present exactly 7 times. Substituents were selected randomly; however, oxygen of the pinocembrin scaffold could accept only carbon. This condition allowed us to exclude the reactive compounds (e.g. peroxides) from the generated set.

We collected 798 unique screens from this database. The most frequent pinocembrin's screens #1–3 (Figure 4) have the weight of 17.66. This change is due to the increased number of screens per molecule. The least frequent pinocembrin's screen #5 (Figure 4) has the weight of 0.49 and is 430 position by the occurrence frequency. Therefore, knowing the weight of the least frequent screen of the scaffold, one has $C_{430}^{21} = 2.4 \times 10^{35}$ possible combinations of selecting 21 screens of the core scaffold.

Lastly, we want to address a large database search as an approach to reconstruct the reference scaffold. For instance, one can use the most frequent screen as a condition for the substructure search in the ChemNavigator [28] or a similar database. Assuming that the reference scaffold is present in the database, it will be represented in the search results. However, the scaffold remains to be identified from the rest of the results. One case when it can be identified is if the reference scaffold contains small and uncommon fragments, for instance Dewar benzene, [1,1,1]-propellane, uncommon chemical elements (metal complexes), or atoms with unusual valencies (free radicals). However, most of the molecules, that contain mentioned fragments, are never used for bioscreening purposes. These molecules are quite dissimilar to the traditional drug-like molecules, and the problem of diversity optimization of such molecules is non-existing.

Screens radius and performance of calculations

One can legitimately expect better accuracy in molecular similarity and diversity calculations with increasing radius of screens. Larger screens mean more fragments and wider comparison metric. The question is what is a reasonable accuracy level in this type of calculations? Lead optimization or focused libraries design implies the selection of similar molecules, but the order of the compounds in the list is of little value as long as all of them exceed certain similarity threshold.

The similar property principle [1] is only a trend and chances are slim that the most similar compounds in terms of a given measure will exhibit the most similar physical or biological properties. The same is also true for the diversity calculations. The examples below support this thesis.

We chose the MDL Drug Report [29] and run the search for approved drugs with C–C bond. Next we manually excluded polypeptides, porphyrine-complexes, boronic compounds, alkylating agents and non-specific compounds (e.g. disinfectants). The resulting database that we used for the model calculations contained 917 known approved drugs.

Steroids

The first iteration of similarity and diversity studies with the centroidal algorithm has been performed on the set of known steroid drugs from the filtered MDDR database. The initial group of 39 steroids was split into two groups: 20 steroids were used for generation of screens (Figure 5) [16] and the other 19 (Figure 6) [17] together with the rest $917 - 39 = 878$ of the drugs were used for the calculations. Sets of screens of radius 2 (324 screens) and of radius 4 (1238 screens) have been generated for the 20 steroids. The final set of 897 compounds were sorted by similarity using the generated screens (see Table 1 for the results).

For both types of screens all 19 steroids from the test set were placed to the top of the lists. Among them, 8 steroids take the same places in both lists, including the most similar (#1) to the set of 20 steroids and the least similar (#19). Also the most similar non-steroidal compound (#20) is identical for the both types of screens. The similarity gap between two steroids in either list is approximately 0.01–0.02, while the gap between the last steroid and the first non-steroid equals to 0.05. This demonstrates the selective and robust separation between steroids and non-steroid compounds in the similarity sorting even using the smaller radius screens.

We also performed the diversity sorting for the same 2 sets. The diversity curve for 897 compounds relative to 20 training set steroids using screens of radius 2 is shown at the Figure 7. The first steroid is placed at the position #176, on the downward slope of the

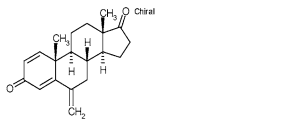
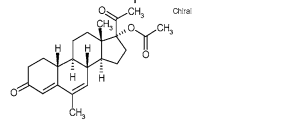
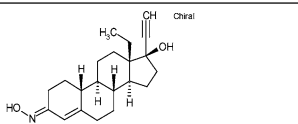
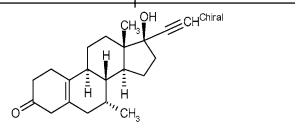
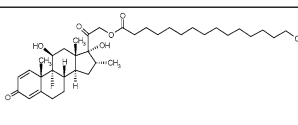
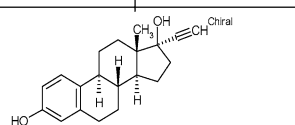
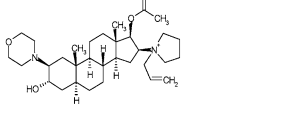
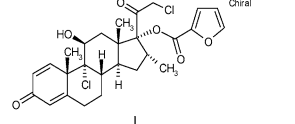
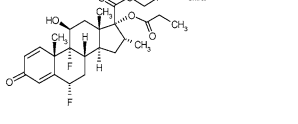
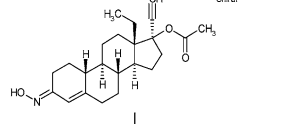
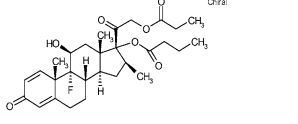
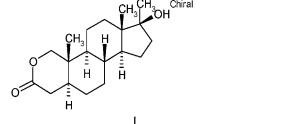
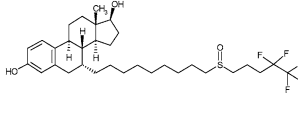
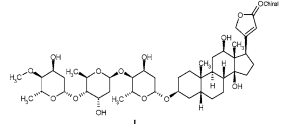
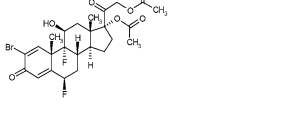
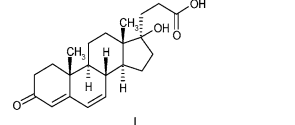
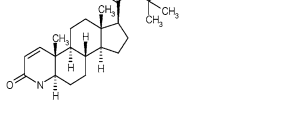
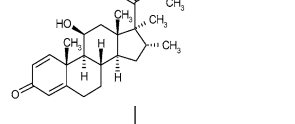
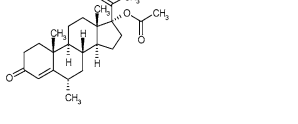
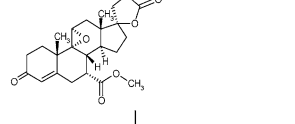
MolRegNo	Structure	MolRegNo	Structure
113828		73240	
134006		73312	
1512		73324	
16876		73342	
26103		73379	
27795		73600	
34090		73643	
4061		73668	
5994		77256	
70281		97494	

Figure 5. Twenty steroids reference set for the screens generation (training set).

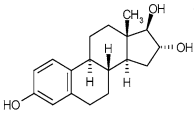
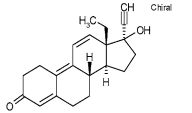
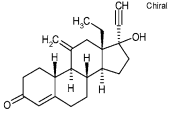
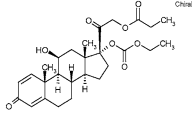
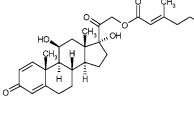
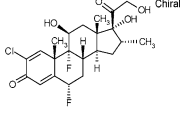
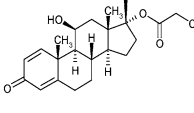
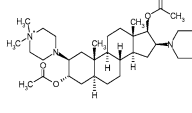
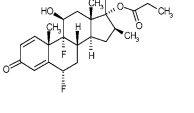
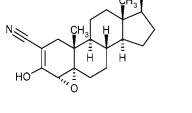
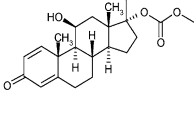
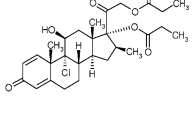
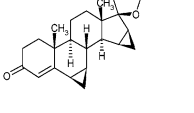
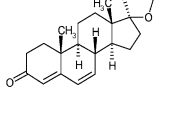
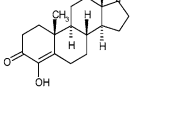
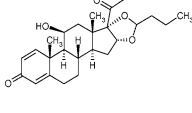
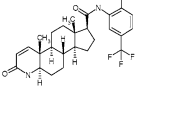
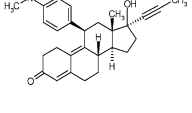
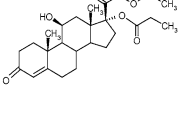
MolRegNo	Structure	MolRegNo	Structure
120675		73260	
135433		73314	
16111		73335	
24256		73364	
26109		73421	
28890		73606	
36433		73667	
46469		73697	
69528		8510	
7159			

Figure 6. Validation set of 19 steroids.

Table 1. Similarity to the reference set sorting results.*

Similarity Rank	Screens' Radius 2		Screens' Radius 4	
	Similarity	MOLREGNOID	Similarity	MOLREGNOID
1	0.521	24256	0.288	24256
2	0.512	7159	0.271	7159
3	0.482	73314	0.269	26109
4	0.480	73606	0.263	73606
5	0.472	16111	0.261	73314
6	0.461	26109	0.239	28890
7	0.453	28890	0.239	16111
8	0.430	135433	0.226	135433
9	0.430	73260	0.225	73335
10	0.428	46469	0.216	73667
11	0.419	73667	0.212	73260
12	0.414	73335	0.210	73697
13	0.406	73697	0.207	46469
14	0.382	8510	0.191	8510
15	0.344	36433	0.169	36433
16	0.338	73364	0.169	69528
17	0.335	69528	0.164	73364
18	0.325	120675	0.162	120675
19	0.320	73421	0.154	73421
20	0.270	72931	0.104	72931

*Compounds ranked 1–19 are steroids. A compound ranked 20 is not a steroid. Similarity sorting of the rest 877 molecules are not shown.

curve. Analogous results have been obtained using the screens with radius 4 (Figure 8). However, larger screens exhibit more specificity to the selection placing the first steroid at the position #523, also at the downward slope.

Quinolines and quinazolines

Two chemically close molecular classes – quinolines and quinazolines were used to test algorithm selectivity. The initial database contained 13 quinolines and 8 quinazolines (Figure 9). Training set included 6 quinolines used for radius 2 screens generation. The rest 911 compounds were sorted by similarity to the training set. We were interested to see whether or not the sorting will place the quinolines to the top of the list.

The results showed quinolines were adequately ranked ##1–5, #8, and #15, however there were also quinazolines at positions ##6, 7, 9, 11, 13, 21, 22, 30 and a naphthalene at the position #10 (Figure 10).

Software implementation

The centroidal algorithm was implemented as a Win32 application for diversity and similarity calculations of any two compounds sets. The first step of the program is generation of screens of a given radius. Generated set of screens is saved as SDF file. In addition to screens, the file contains the components of centroidal vector (a sum of screen's weights) and the total number of compounds in the set. Total number of compounds is stored as additional field for the first screen in the file. SDF file provides an industry standard open format that ensures an end-user that no data except the screens, weights and total number of compounds are used. This distinguishes the SDF file from its binary unreadable analogs that leave an end-user to believe they contain no confidential information and that encryption algorithms are secure enough.

We use two-dimensional screens with bonds' topology. Bond's topology makes screens more

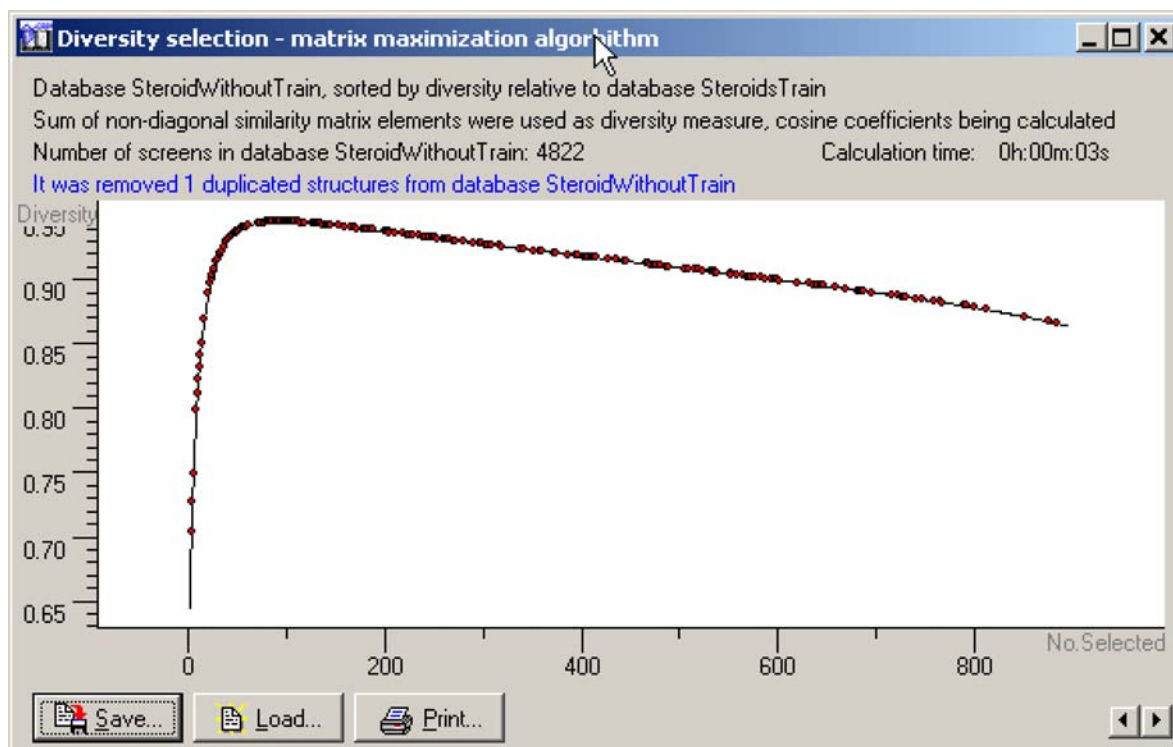


Figure 7. Diversity sorting of the test set against the 20 steroids reference set using radius 2 screens.

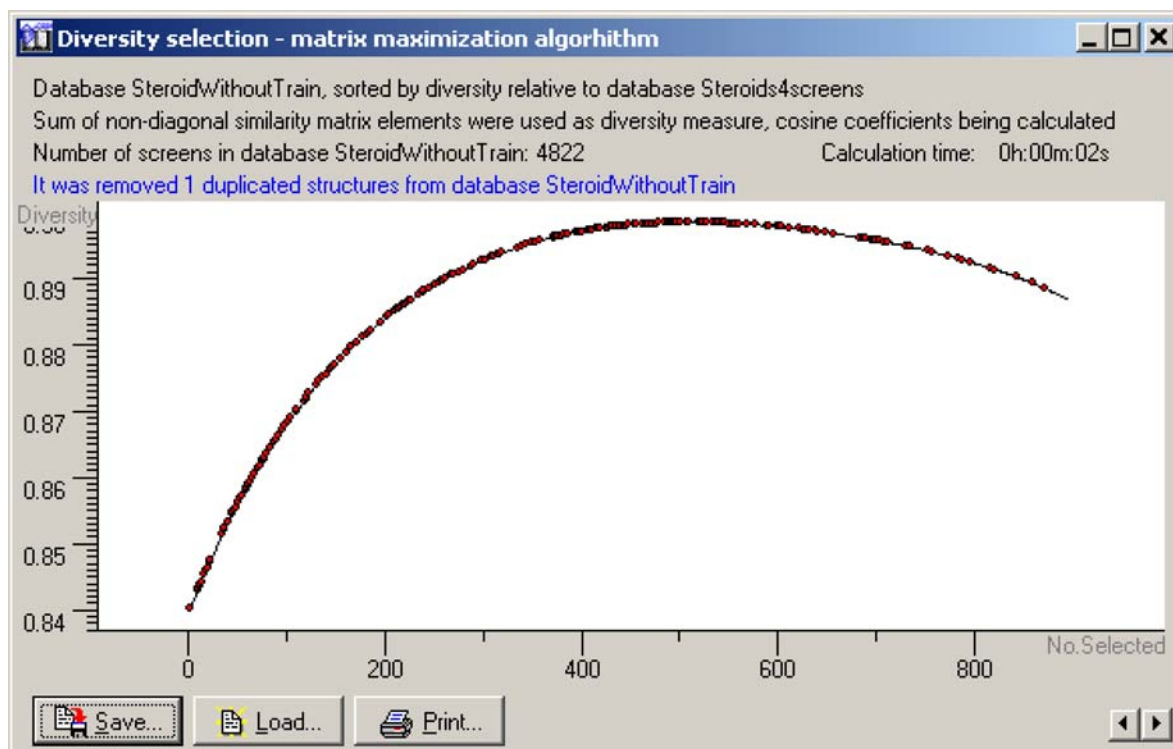


Figure 8. Diversity sorting of the test set against the 20 steroids reference set using radius 4 screens.


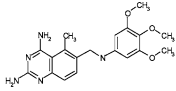
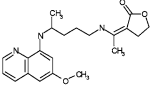
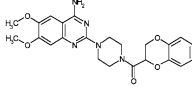
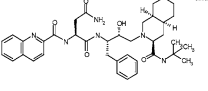
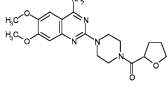
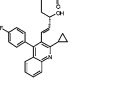
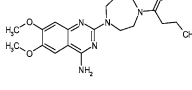
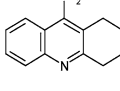
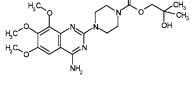
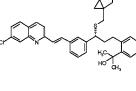
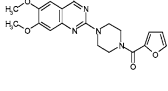
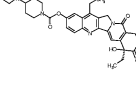
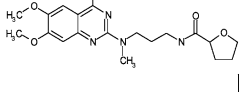
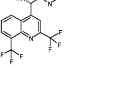
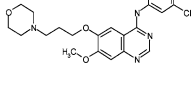
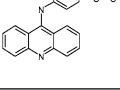
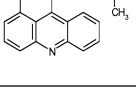
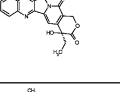
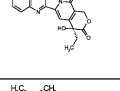
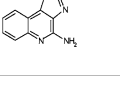
MolRegNo	Structure	Training	MolRegNo	Structure
125483		Yes	58436	
19455		Yes	73265	
20989			73291	
47215			73294	
53966			73301	
62806			73629	
63037		Yes	7691	
73288		Yes	98286	
73377				
73411		Yes		
82562				
8447				
89984		Yes		

Figure 9. Quinolines and Quinazolines Selection. 6 quinolines training set.





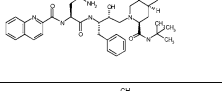
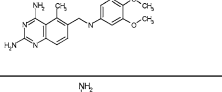
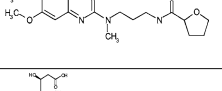

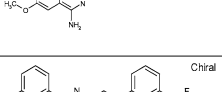
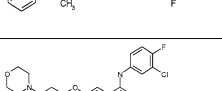
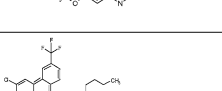
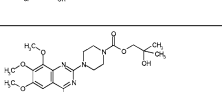
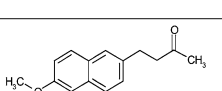
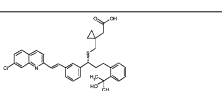
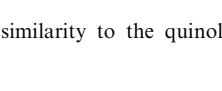
Rank	MolRegNo	Match	Structure
1	82562	0.3204	
2	8447	0.2932	
3	53966	0.2746	
4	73377	0.2696	
5	20989	0.2592	
6	58436	0.2546	
7	7691	0.2434	
8	47215	0.2382	
9	73294	0.2312	
10	124641	0.2273	
11	98286	0.2260	
12	7692	0.2199	
13	73301	0.2180	
14	73220	0.2125	
15	62806	0.2102	

Figure 10. Top 15 molecules by similarity to the quinolines training set.

diverse and selective from the chemical standpoint. We modified the SDF format to accommodate these changes specifically the part containing bonds' topology list. According to [30], the fields 13–15 are not used for the bond's record. We use these fields to keep topological properties of the bond: 0 – non-cyclic bond; 2 – aromatic bond, five-membered ring; 3 – aromatic six-membered ring; 4... n – ($n-1$)-membered rings. If the bond is shared between two or more rings, the bond is assigned the smallest ring's order.

The SDF file in this format can be used for the diversity and similarity calculations. The compounds for diversity and similarity sorting also should be submitted as an SDF file. Similarity sorting results are provided in the graphical interface and can be saved as a *.lst file with the compound's ID and similarity coefficient.

The software alerts a user of potential security holes. The warning of potential reverse engineering possibility is given if during a screen generation process

- the maximum topological diameter of the molecule is found to be less than 3 times of the screen radius, or
- the total number of screens is less than 50, or
- there is a molecule that contains more than 50% of all screens gathered from the set of molecules.

From this standpoint, the pinocembrin based library case does not meet 2 criteria: the total number of screens and molecules with more than 50% of screens present. However, even in this case the probability of reverse engineering is estimated as $\sim 10^{-9}$ or less.

Conclusion

Evidently from our test calculations, screens of radius two and four work equally well in similarity calculations. The diversity sorting, however, benefits more from the larger radius four screens.

Diversity and similarity calculations can be done with the centroidal algorithm given the screens for the whole reference set and their weights. Using smaller radius screens, while delivering adequate results, provides enormous number of possible combinations between them making the reverse engineering of initial molecules virtu-

ally impossible. At the same time the screens can be transferred openly, since modern encryption algorithms such as RSA [31] offer the same security as the centroidal algorithm. In both cases, the code breaking task belongs to the NP-complete problem class. Since data are not encrypted, it allows end user more control to assess their significance as well as security and reliability against reverse engineering attacks. Like other algorithms of encryption, the centroid algorithm cannot give 100% guarantee of protecting a chemical structure from dataset, but probability of initial structure identification is very small – less than 10^{-40} in typical cases.

We should admit three potential security holes of the algorithm:

- (1) Some molecules can be entirely included in a set of screens, especially when the screen radius is large;
- (2) A core structure of a combinatorial library can be potentially identified, especially when the library does not have many unique screens.
- (3) A molecule that contains most of the screens from the set can be identified.

Nonetheless, we suggested an algorithm that can easily detect these potential issues and alert a user.

These cases, though possible, are not encountered often in current industrial setting. For example 10 molecules randomly selected from ChemDiv's small molecule collection have in average 130 screens total, average 34 screens per molecule (radius 2) and there is no molecule with more than 60 unique screens.

Acknowledgements

Authors are thankful to Sergey Tkachenko, Caroline Williams, Alex Khvat, Nikolay Savchuk and Andrey Ivachtchenko for the productive discussions and their support of this work.

References

1. Johnson, M.A. and Maggiora, G.M. (eds.) Concepts and Applications of Molecular Similarity. Wiley, New York, 1990.
2. MDL Information Systems, Inc., <http://www.mdli.com/>.
3. Daylight Chemical Information Systems, Inc., <http://www.daylight.com/>.
4. CambridgeSoft Corporation, <http://www.camsoft.com/>.
5. Oxford Molecular Ltd., <http://www.oxmol.co.uk/>.
6. Synopsys Scientific Systems Ltd., <http://www.synopsys.co.uk/>.
7. Tripos, Inc., <http://www.tripos.com/>.
8. Warr, W.A., *Perspect. Drug Discovery Des.*, 7/8 (1997) 115.
9. Brown, R.D. and Martin, Y.C., *J. Chem. Inf. Comput. Sci.*, 36 (1996) 572.
10. Patterson, D.E., Cramer, R.D., Ferguson, A.M., Clark, R.D. and Weinberger, L.E., *J. Med. Chem.*, 39 (1996) 3049.
11. Matter, H., *J. Med. Chem.*, 40 (1997) 1219.
12. Adamson, G.W., Cowell, J., Lynch, M.F., McLure, A.H.W., Town, W.G. and Yapp, A.M., *J. Chem. Doc.*, 13 (1973) 153.
13. Dittmar, P.G., Farmer, N.A., Fisanick, W., Haines, R.C. and Mockus, J., *J. Chem. Inf. Comput. Sci.*, 23 (1983) 93.
14. Carhart, R.E., Smith, D.H. and Venkataraghavan, R., *J. Chem. Inf. Comput. Sci.*, 25 (1985) 64.
15. Nilakantan, R., Bauman, N., Dixon, J.S. and Venkataraghavan, R., *J. Chem. Inf. Comput. Sci.*, 27 (1987) 82.
16. Moreau, G. and Broto, P., *Nouv. J. Chim.*, 4 (1980) 359.
17. Dalby, A., Hourse, J.G., Hounshell, W.D., Gurchurst, A.K.I., Grier, D.L., Leland, B.A. and Laufer, J., *J. Chem. Inf. Comput. Sci.*, 32 (1992) 244.
18. Bremsler, W., *Anal. Chim. Acta*, 103 (1978) 355.
19. Figueras, J., *J. Chem. Inf. Comput. Sci.*, 36 (1996) 986.
20. Trepalin, S.V., Gerasimenko, V.A., Kozyukov, A.V., Savchuk, N.Ph. and Ivaschenko, A.A., *J. Chem. Inf. Comput. Sci.*, 42 (2002) 249.
21. Trepalin, S.V., Yarkov, A.V., Dolmatova, L.M., Zefirov, N.S. and Finch, S.A.E., *J. Chem. Inf. Comput. Sci.*, 35 (1995) 405.
22. Smith, S.K., Cobleigh, J. and Svetnik, V., *J. Chem. Inf. Comput. Sci.*, 41 (2001) 1463.
23. Hu, C.-Y. and Xu, I., *J. Chem. Inf. Comput. Sci.*, 36 (1996) 82.
24. Willett, P., Barnard, J.M. and Downs, G.M., *J. Chem. Inf. Comput. Sci.*, 38 (1998) 983.
25. Mount, J., Ruppert, J., Welch, W. and Jain, A.N., *J. Med. Chem.*, 42 (1999) 60.
26. Holliday, J.D., Ranade, S.S. and Willett, P., *Quant. Struct.-Act. Relat.*, 14 (1995) 501.
27. <http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=pccompound>. Accessed: October 2005.
28. <http://www.chemnavigator.com>. Accessed: October 2005.
29. MDL drug data report database, 2005.
30. CT file format. MDL report. August 2002, 1–64.
31. Gordon, J., *Electronics Lett.*, 20 (1984) 514.