Combinatorial Drug Discovery from Activity-Related Substructure Identification

Md. Imbesat Hassan Rizvi, Chandan Raychaudhury and Debnath Pal

Abstract A newly developed drug discovery method composed of graph theoretical approaches for generating structures combinatorially from an activity-related root vertex, prediction of activity using topological distance-based vertex index and a rule-based algorithm and prioritization of putative active compounds using a newly defined Molecular Priority Score (MPS) has been described in this chapter. The rule-based method is also used for identifying suitable activity-related vertices (atoms) present in the active compounds of a data set, and identified vertex is used for combinatorial generation of structures. An algorithm has also been described for identifying suitable training set–test set splits (combinations) for a given data set since getting a suitable training set is of utmost importance for getting acceptable activity prediction. The method has also been used, to our knowledge for the first time, for matching and searching rooted trees and sub-trees in the compounds of a data set to discover novel drug candidates. The performance of different modules of the proposed method has been investigated by considering two different series of bioactive compounds: (1) convulsant and anticonvulsant barbiturates and (2) nucleoside analogues with their activities against HIV and a data set of 3779 potential antitubercular compounds. While activity prediction, compound prioritization and structure generation studies have been carried out for barbiturates and nucleoside analogues, activity-related tree–sub-tree searching in the said data set has been carried for screening potential antitubercular compounds. All the results show a high level of success rate. The possible relation of this work with scaffold hopping and inverse quantitative structure–activity relationship (iQSAR) problem has also been discussed. This newly developed method seems to hold promise for discovering novel therapeutic candidates.

Keywords Graph theory · Vertex index of molecular graph · Root vertex
Combinatorial molecular structure generation · Activity prediction
Compound prioritization and screening · Drug discovery

Md.I. H. Rizvi · C. Raychaudhury · D. Pal (✉)
Department of Computational and Data Sciences, Indian Institute of Science,
Bangalore 560012, India
e-mail: dpal@iisc.ac.in

C. G. Mohan (ed.), *Structural Bioinformatics: Applications in Preclinical Drug Discovery Process*, Challenges and Advances in Computational Chemistry and Physics 27, https://doi.org/10.1007/978-3-030-05282-9_4

Abbreviations

QSAR	Quantitative structure–activity relationship
iQSAR	Inverse quantitative structure–activity relationship
vHTS	Virtual high-throughput screening
MIC	Minimum inhibitory concentration
Mtb	*Mycobacterium tuberculosis*
AAE	Acid alkyl ester
NA	Nucleoside analogue
HIV	Human immunodeficiency virus
MPS	Molecular Priority Score
ARL	Active range length
ARW	Active range weight
ARV	Active range value
MAI	Molecular activity index
IRL	Inactive range length
IRW	Inactive range weight
IRV	Inactive range value
MDI	Molecular de-activity index
SMILES	Simplified molecular-input line-entry system
MOL file	Molecular structural information file

1 Introduction

Exploring chemical space to discover a compound that elicits a desired pharmacologic response without undesired side effect is like searching a needle in a haystack problem. The problem arises because we seek to screen a limited subset that exists among many compounds that elicit a desired pharmacologic response. Different approaches have therefore evolved to make the problem tractable, namely effective use of macromolecular target information, if available, use synthesis tractability of the compounds as guidance, and most importantly, the pharmacological relevance of the compounds selected. While modern advances like targeted library search or chemogenomics have helped in bringing focus to the drug candidate search, the utility of drug candidate search using serendipity-based approaches has not diminished in face of increasing burden of drug resistance and adverse side effects. These problems may possibly be addressed by discovering novel compounds using new drug discovery methods. One of such a new line of thinking has been proposed by Ruddigkeit et al. [1] who have considered all possible compounds having 17 atoms taken from C, N, O, S and halogens to create a database of several billions of compounds. It is tempting to believe that such an effort of discovering novel drug molecules from such a huge collection of compounds can be useful. However, a method that enables searching of potential drug candidates from a relatively smaller set of compounds, quite exhaustive at the same time within given limits, activity linked and rationally guided too may help drug discovery more effectively.

Among the current drug discovery methods, data modelling and quantitative–qualitative prediction of activity [2–4], use of molecular docking methods and scoring functions for virtual high-throughput screening (vHTS) [5] and 3D quantitative structure–activity relationship (QSAR) studies [6] are some of the most used ones. At the same time, combinatorial generation of chemical compounds is also carried out since it increases the possibility of finding novel drug molecules from a large number of chemically diverse compounds generated particularly for the need of making scaffold hopping [7]. It also provides the opportunity to search for compounds having diverse structural characteristics which in turn may help decipher the role of molecular components which may be responsible for the biological activities of new drug molecules, particularly in situations where novel therapeutic candidates are sought for to handle the challenges arising out of drug resistance problem [8].

So far generating molecular structures are concerned, molecular topology-based approaches are in use for generating and designing molecular structures [9, 10] and graph theory [11] and graph theoretical methods [12] have been suitably used for doing that. However, in general these methods are used for generating structures combinatorially [10] with no connection to their biological activities and a separate method has to be used for the prediction of molecular properties and activities. It appears, therefore, that a method that generates a large number of compounds combinatorially and gets linked to their activities at the same time may be more efficient in designing and discovering novel drug molecules. In particular, topological molecular descriptors [2] can be useful in this regard. Moreover, if this is done using a single molecular (structural/substructural) descriptor, the process may also be looked upon from inverse QSAR (iQSAR) point of view [13] since the basic idea of doing iQSAR studies is to get molecular structures back from molecular descriptor which has been used for activity prediction. In this context, it seems reasonable to explore whether a method can be developed that is integrated in such a way that it can be used for generating structures combinatorially that would have molecules of diverse scaffold from a single molecular topological descriptor , can be used for predicting molecular properties/activities and can be used for compound prioritization and screening to help discover potential drug candidates.

So, the first question that may be asked in developing such an integrated method is: Can we have a method such that structures can be generated combinatorially from structural or substructural information that is already related to activity? In this regard, there are two primary aspects in designing potential bioactive compounds from activity-related substructural information—(1) identification of activity-related vertices using a suitable method; (2) a method that can be used for structure generation using topological information associated with such vertices. One of the most useful activity-related substructure identification method was proposed by Klopman [14] where molecular fragments of different length are identified from active and inactive compounds, and the fragments are weighed on the basis of the number of fragments obtained from active and inactive compounds using a suitable measure to assess their usefulness in predicting activities and mathematical–statistical methods are used to do that. However, no structure generation method is used for this work [14].

In this chapter, we have described in detail a graph theory-based method, developed recently by our research group [15], for combinatorial generation of chemical structures from activity-related substructural topological information. This approach [15] has been found to be useful in generating structures of active antitubercular compounds from activity-related vertices of the molecular graphs representing different other active antitubercular compounds. For developing the present method [15], we have leveraged primarily a non-isomorphic rooted tree generation algorithm [16] and a cycle enumeration method [17] to design novel bioactive compounds in the form of reconstructed molecular graph as outlined earlier [18, 19]. In the proposed integrated method, activity-related vertices are first identified by using the rule-based method [18, 19] where topological distance-based vertex indices are used as local molecular descriptors in data sets having the biological activities of interest. Once the activity-related vertices are identified, a suitable vertex is taken for structure generation using the distance distribution associated with the vertex which gives the topological distances of all the vertices in molecular graph from that vertex (say, the root vertex). A large number of rooted trees are thus generated de novo [15]. Subsequently, 2D molecular structures containing cycles of different size are created by joining vertices of the tree graphs. In this way, all the generated structures contain this activity-related substructure, and therefore, there is a possibility that some of generated structures may be classified as active. Furthermore, to get complete 2D structures of the compounds, user-defined parameters are used to add multiplicity of bonds (e.g. double and triple bonds) between pairs of vertices and add chemical nature of the atoms (nitrogen, oxygen, etc.) represented by the vertices. Canonicalization is used to identify unique structures which are further used for screening of potential active compounds.

It may be noted that scaffold hopping [7] is embedded in the method since the generated structures are different from the starting compound and are expected to have diverse topological architecture. Also, since both compound generation and activity prediction are done using the same vertex index (substructural/local descriptor), the method may also be regarded as an attempt to address the inverse quantitative structure–activity relationship (iQSAR) problem [13] in its integrated framework. Furthermore, in order to relax the condition for structure generation from distance distribution as outlined earlier [18, 19] and to make it more flexible, we have developed an algorithm for generating sub-trees by adding or deleting vertices from the tree structures generated on the basis of a given distance distribution associated with an activity-related vertex. To our knowledge, this is the first time that a method [15] has been developed and used for drug discovery through database searching using rooted tree and sub-tree matching algorithms.

The method has already been used to investigate its usefulness for a series of 41 acid alkyl ester (AAE) derivatives and three known antitubercular drugs [15]. In this chapter, we have furnished new results obtained for a series of 19 convulsant and anticonvulsant barbiturates [18], 20 nucleoside analogues (NA) for their activities against HIV [20, 21], and a data set of 3779 compounds (named GTB data set) for which minimum inhibitory concentration (MIC) values have been measured against H37Rv strain of *Mycobacterium tuberculosis* (*Mtb*) [22]. The GTB data set may be obtained from the link [23] given in the reference section. The results described here will therefore substantiate the findings obtained earlier [15]. Regarding activity prediction, results have been reported for NA and barbiturate data sets. For barbiturates data set, we have considered the same training set and test set as used in an earlier study [18]. However, for the NA data set, we have identified a reasonably well-performing training set–test set split and have reported the results for individual compounds present in that split. For prioritization of the generated active compounds that help screen potential active compounds, Molecular Priority Score (MPS) [15] has also been used and the results obtained for NA and barbiturate series of compounds have been given in the tables alongside their activity prediction results. We have carried out combinatorial generation of structures using topological distance-based substructural information associated with identified activity-related vertices (atoms) in some compounds of the data set. We have been able to reconstruct the structures of active NA and barbiturate compounds from the substructural information associated with activity-related vertices of other active NA and barbiturate compounds. Regarding substructure searching exercise, we have reported identified potential active compounds from GTB data set [22, 23] considering activity-related atoms (vertices) in the structures of Isoniazid and Streptomycin, both of which are known antitubercular drugs in use.

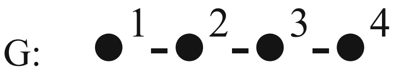
It appears from the outcome of the results that the integrated method would find a place as a useful drug discovery tool for designing and discovering novel bioactive compounds. In particular, the method is believed to be of much help in situations where novel drug candidates having very different structural characteristics/scaffolds are sought for particularly to overcome the drug resistance problem.

2 Methods

In this section, we have described in detail different mathematical approaches/tools which have been used to develop the present integrated drug discovery method and the related computer programs. Examples with tables and figures have been used to illustrate underlying concepts of the methods used. While we have leveraged few existing mathematical aspects for the present purpose, we have introduced some new algorithms as well.

2.1 Computation of Vertex Index

Let G be the carbon skeleton of n-butane and $D(G)$, the corresponding distance matrix is shown in Fig. 1. Computation of D^{-4} indices for the vertices of $D(G)$ has been illustrated below.



D(G):

$$\begin{array}{c|cccc}
 & 1 & 2 & 3 & 4 \\
\hline
1 & 0 & 1 & 2 & 3 \\
2 & 1 & 0 & 1 & 2 \\
3 & 2 & 1 & 0 & 1 \\
4 & 3 & 2 & 1 & 0
\end{array}$$

Fig. 1 Graph G representing vertex labelled carbon skeleton of n-butane and the corresponding topological distance matrix $D(G)$

Therefore, D^{-4} index for the four vertices v_i, $i = 1, 2, \ldots, 4$ of G may be computed as:

$$\begin{aligned}
D^{-4}(v_1) &= 1^{-4} + 2^{-4} + 3^{-4} = 1.0749 \\
D^{-4}(v_2) &= 1^{-4} + 1^{-4} + 2^{-4} = 2.0625 \\
D^{-4}(v_3) &= 1^{-4} + 1^{-4} + 2^{-4} = 2.0625 \\
D^{-4}(v_4) &= 1^{-4} + 2^{-4} + 3^{-4} = 1.0749
\end{aligned}$$

One can, therefore, compute the values of D^{-4} index for all the atoms (vertices) of all the compounds (molecular graphs) in a data set considering the molecular graphs (hydrogen-suppressed or hydrogen-filled) of the compounds. Hydrogen-suppressed graphs may be considered for generating structures from the distance distribution associated with a vertex since structure generation using information about the vertices of hydrogen-filled graphs may pose computational bottlenecks during the process because of a large number of structures that are usually generated in this way. Moreover, if chemical information of the vertices is provided, one can always create the hydrogen-filled graphs from the corresponding hydrogen-suppressed graphs.

2.2 *Rule-Based Activity Prediction*

In order to carry out activity prediction studies using the present method, a data set containing both active and inactive compounds for a biological endpoint of interest is gathered. The data set is then divided suitably into a training set and a test set. The biological activities of the compounds are then predicted for both the training set and the test set using a rule-based system [18, 19]. In order to make the activity prediction, ranges of vertex index values coming from active and inactive compounds are first found out using some rules [18, 19] and the activity is predicted on the basis of the number of vertex index values falling in these ranges as defined in the rule-based system [18, 19]. For the present purpose, the values of vertex index D^{-4} are computed for the vertices of the training set compounds (molecular graphs). Once the indices are computed, they are arranged in an ascending order and ranges of values coming from both active and inactive compounds are found in the ordering and are tagged as "Active" and "Inactive" ranges by applying certain rules [18, 19] given below:

1. Three or more consecutive vertex index values coming exclusively from active compounds and exclusively from inactive compounds are said to form an "active range" and an "inactive range", respectively. However, at least three index values in a range have to be distinct if they come from the same compound and at least two index values in a range have to be distinct if they come from different compounds.
2. Some single vertex index value coming from both active and inactive compounds is not considered to form an "active range" or "an inactive range" by itself or along with other vertex index values unless two-thirds of that single vertex index comes from active compounds or inactive compounds, respectively.

It has been discussed earlier [24] in connection with identifying ranges that the vertices which correspond to the vertex index values forming active ranges may be regarded as topological features responsible for making the compounds active. In other words, they may be regarded as a set of features forming "Topological Biophore " which are responsible for exhibiting a given biological activity of the compound under consideration. From this point of view, it may be said that if the index values of some (or, all) of the vertices of a compound fall in active ranges, then those vertices may be regarded as forming certain topological biophore which make the compound active. Presumably, some of the vertex index values of a compound may fall in inactive ranges as well. Thus, in order to predict activity from the occurrences of the vertex index values in active and inactive ranges, another set of rules [18, 19], given below, are applied:

A compound is predicted "ACTIVE" if all or some of its vertices fall:

1. Only in active ranges or
2. In both active and inactive ranges, the number of index values falling in active ranges is greater than those falling in inactive ranges.

Otherwise, the compound is predicted "inactive".

In order to use this rule-based system for activity prediction, a set of bioactive compounds with known activities (e.g. experimentally determined activities) have to be collected (from the literature or an experimental laboratory). A training set is then formed by picking up compounds from the data set suitably to train the system to learn the structural requirement for a compound to be active. A fewer number of compounds are also kept for testing purposes (test set). Once the training is done, activity predictions for both training set compounds (retrofit studies) and test set compounds are carried out. For predicting the activities of the test set compounds, the D^{-4} index values for the test set compounds are computed. If the system is found to produce high (acceptable) percentage of correct activity predictions for both the training set and the test set compounds along with none or very few (acceptable) wrong activity predictions, it may be regarded as standardized for the prediction of activity of chemical compounds for the biological endpoint for which the system is standardized.

2.3 Training Set–Test Set Split

It is always important that a suitable training set be obtained from a data set of bioactive compounds such that the structural characteristics of the compounds, present in the data set, is reflected in the training set, and the learning of the (expert) system/prediction tool is as adequate as possible for getting useful activity predictions by the method used in this purpose. In general, researchers look for the diversity present in the structures in creating a training set from a given data set. Presumably, some intuition or expertise of the drug designer/medicinal chemist may be required to do that or some mathematical diversity analysis may be carried out in obtaining a suitable training set. However, it appears that generating a large number (e.g. 1000) of training set–test set splits (combinations) and reporting the successful predictions of all or some (e.g. top 20, 25) of the best-predicting splits for a given data set of bioactive compounds would be a very straightforward and useful approach for identifying a suitable training set. Having obtained various top performing splits, one can select a suitable split that gives high percentage of successful predictions for both training set and test set and obtains activity prediction for the compounds present in both the sets. Although such splits have been used [24, 25] for evaluating the performance of vertex indices and a rule-based method for activity prediction [18, 19] considering small and large data sets, no algorithm is available to report the activity predictions for different splits. We have incorporated this algorithm in the program for reporting the outcome of activity predictions for different splits so that one can consider a suitable split for further work such as structure generation. This can be done for both quantitative data and qualitative data (active–inactive type). It may also be noted that the computer program can be used for the identification of training set–test set splits and activity predictions by considering both hydrogen-filled (H-filled) and hydrogen-suppressed (H-suppressed) molecular graphs of the compounds under consideration.

2.4 Compound Prioritization

The present method [15] also contains a section that can be used for prioritization of potentially active compounds. This may be particularly useful for screening few highly active compounds from a big database, e.g. from a set of combinatorially generated compounds (described in the next section). This method is based on some of the characteristics of active and inactive ranges found in the ordering of vertex index values. Therefore, one has to look into some details of such ranges. In doing that, two factors may be given special attention—(1) the number of vertex index values in an active range (active range length: *ARL*); (2) the number of compounds contributing to form the range (active range weight: *ARW*). By applying one's intuition too, it becomes apparent that a joint effect of these two factors may help prioritize predicted active compounds. Therefore, we first propose a measure, active range value (*ARV*), as the algebraic sum of *ARL* and *ARW* values given by:

$$ARV = (ARL + ARW) \tag{1}$$

Clearly, a range larger in length and contributed by more number of compounds in forming the range would have higher *ARV* value. We define such a range of higher *ARV* value a "STRONGER" range compared to those which have lower *ARV* values. Now, let us assume that *M* out of *N* vertices of a molecular graph *G* (representing a chemical compound) have fallen in different active ranges. If the vertices are denoted by $v_1, v_2, \ldots, v_M$, one would get *M* number of *ARV* measures as $ARV(v_1), ARV(v_2), \ldots, ARV(v_M)$. In order to get a measure of the contribution of the vertices falling in different active ranges (i.e. contribution of activity-related vertices), we further propose a molecular activity index (*MAI*) as:

$$MAI(G) = \sum_{i=1}^{M} ARV(v_i) \tag{2}$$

It may also be noted that while considering the length of an active range and the number of compounds contributing to form the range, some single values that come from both active and inactive compounds are taken into account since they are part of the active range according to the second rule of range selection mentioned earlier.

At the same time, there is a possibility that some of the vertex indices of molecular graph *G* may fall in inactive ranges too (the second rule for activity prediction) and that may be considered to pose a negative effect on the activity of the compound. For the prediction purpose, therefore, vertices falling in inactive ranges have to be considered. For doing that, let us assume that M' vertices of *G,* viz. $u_1, u_2, \ldots, u_{M'}$ fall in inactive ranges. We, thus, propose a measure, molecular de-activity index (*MDI*) for *G* and it may be defined as:

$$MDI(G) = \sum_{j=1}^{M'} IRV(u_j) \tag{3}$$

In Eq. 3, *IRV* stands for inactive range value and is the sum of *IRL* (inactive range length) and *IRW* (inactive range weight) which is in line with the definitions used for such measures of active ranges. Computation of *IRV* can be done using Eq. (4) given below:

$$IRV = (IRL + IRW) \tag{4}$$

Therefore, by considering a combined effect of *MAI* and *MDI*, one can prioritize the newly generated active compounds and curate some high-ranking compounds for further studies. Thus, in order to get a measure of combined effect of the vertices falling in active ranges and inactive ranges (if any) and prioritizing (ranking) the compounds according to their activities, we propose a measure, Molecular Priority Score (*MPS*), for *G* and it may be computed using Eq. (5):

$$MPS(G) = MAI(G) - MDI(G) \tag{5}$$

Considering MPS value as a measure for prioritization of active compounds, a compound with higher *MPS* value will occupy a higher position in the ranking. Therefore, a compound may be regarded as more active if it gets higher *MPS* value. This will then help screen some top-ranking compounds. However, ranking of active compounds using *MPS* is not mandatory. One may always wish to consider all the predicted active compounds for further studies particularly if the number of highly ranked compounds (in terms of MPS value) is very small. At the same time, there is no need to prioritize those compounds which are predicted inactive since the idea is to screen potentially highly active compounds for a given biological endpoint.

2.5 Combinatorial Structure Generation from Root Vertex

In developing the structure generation method, we have used an algorithm for generating rooted trees [16] which have been extended to the generation of cyclic compounds and finally a complete 2D structure of chemical compounds. The structure generation exercise starts off as generating all possible canonical trees for any given number of vertices. Subsequently, topological distance restriction on the generated tree structures is used to filter and keep only those trees having a desired distance distribution. Further, for the application of relaxed distance criteria for compound structures having increased or decreased number of vertices (non-hydrogen atoms), the matching criteria of distance distribution have been suitably changed to accommodate the addition, deletion and migration of the vertices over the tree structures with exact distance restriction. The theories and implementation details are described in the following subsections.

2.5.1 Structure for a Given Distance Distribution

A molecular graph represents topological connections between the atoms of the molecules. A spanning tree of the graph can provide the basic skeleton over which additional edges can be inserted to introduce cycles and thereby produce the entire molecular structure. The multiplicity of bonds can be considered as edge weights and can be dealt by assigning weights 1, 2 and 3 for single, double and triple bonds, respectively. Similarly, heterogeneous atoms, with their valency information, can also be introduced as nodes, which are by default considered to be carbon atoms in our discussions.

It is clear from above that the starting point of structure generation for a given number of vertices (atoms) is the generation of rooted trees since the structure generation will be carried out with respect to a particular atom in a molecule in our current approach based on topological distances from a particular vertex. Moreover, to prevent duplicate structures, only non-isomorphic trees should be generated.

For the purpose of illustration, consider the chemical structure and the corresponding graphical and tree representation as shown in Fig. 2.

The numbering of vertices has no structural significance apart from that it is done to obtain the rightmost tree having node 1 as the root and pre-order numbering for the other vertices and is merely for array representation of the tree structure. The tree can be represented by the following parent and level array representations:

$$\text{parent} = [0,\ 1,\ 2,\ 3,\ 1,\ 5,\ 5] \quad \text{level} = [1,\ 2,\ 3,\ 4,\ 2,\ 3,\ 3]$$

where for a given vertex i, $\text{parent}[i] = j$ means vertex j is the parent of vertex i except for root vertex 1 having no parent vertex and is represented by 0 as its parent. Similarly, for a vertex i, $\text{level}[i] = j$ means vertex i is at level j, where root vertex 1 has a level 1 and other vertices have level one greater than the level of its parent vertex. The root vertex can sometimes be considered to have level 0 and the levels of the subsequent vertices follow.

With the illustrated example and the terms introduced in consideration, the different steps in structure generation are explained in the following points:

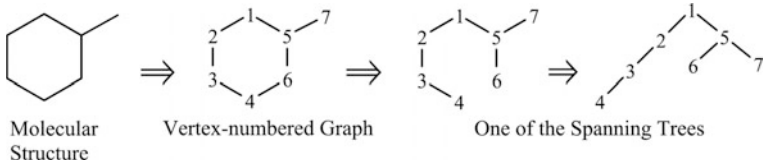


Fig. 2 Graph and tree illustration

(a) **Non-isomorphic canonical tree generation:**

Beyer and Hedetniemi [16] have proposed an iterative algorithm to reverse lexicographically generate non-isomorphic canonical trees for a given number of vertices. The algorithm achieves this transformation through a successor function defined below.

Let $L(T) = [l_1 l_2 \ldots l_n]$ be a level sequence containing an element greater than 2. Let p be the rightmost position of such an element, i.e. $p = \max\{i : l_i > 2\}$. Let q be defined as the rightmost position preceding p such that $l_q = l_p - 1$, i.e. $q = \max\{i : i<p,\ l_i = l_p - 1\}$. Hence, the vertex corresponding to position q is the parent of vertex corresponding to position p. Then the successor of $L(T)$, i.e. $\mathrm{succ}(L(T)) = [s_1 s_2 \ldots s_n]$ is defined such that:

(i) $s_i = l_i\,\mathrm{for}\,1 \leq i<p$
(ii) $s_i = s_{i-(p-q)}\,\mathrm{for}\,p \leq i \leq n.$

The algorithm can be used successively generating all the non-isomorphic canonical level representation of trees from a provided starting level sequence to the last possible reverse lexicographic sequence, i.e. $\left[1, \underbrace{2, 2 \ldots 2}_{n-1\,\mathrm{times}}\right]$. If no starting level sequence can be provided, the algorithm can start with the lexicographically largest sequence $[1, 2, 3 \ldots n]$.

The trees generated by the aforementioned algorithm can in general have any number of children for any parent vertex. In context of chemical structures of carbon atoms, only those trees are being filtered and kept where the root has at most four children and the rest of the vertices have at most three children. This restriction can later be further refined for hetero-atoms in accordance with their valency.

(b) **Cycle introduction by addition of edges:**

The generated rooted trees are graphical models of acyclic compound structures. Cycles can be introduced by adding edges between any two vertices, say i and j, such that:

$$\mathrm{parent}[i] \neq j \quad \mathrm{and} \quad \mathrm{parent}[j] \neq i$$

The size or the number of sides in the cycle so introduced can be obtained by the following relation:

$$\begin{aligned}\mathrm{num\_cycle\_sides} = {} & \mathrm{level}[i] + \mathrm{level}\,[j] \\ & -\ 2 * \mathrm{level}[\mathrm{lowest\_common\_anscester}(i,j)] + 1\end{aligned}$$

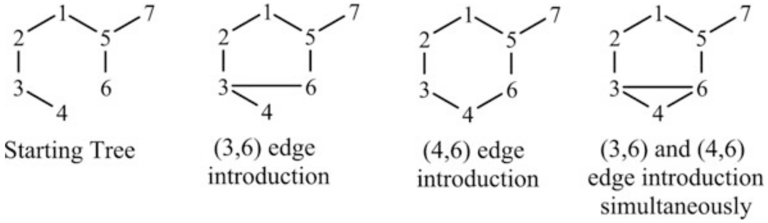
In general, cycles of size 3 onwards will be possible. For more than one cycles to be introduced, a combination of these identified edge introductions can be simultaneously carried out.

However, introduction of multiple edges may lead to fused or bridged cycles and the size of cycle may become different than intended. Consider the case of starting structure generation from the tree in Fig. 2. If it is required to have two cycles which can have size 5, or 6, it can be seen (Fig. 3) that the edge introductions between vertices 3 and 6 and vertices 4 and 6 individually satisfy the size criteria, but in combination, they inadvertently lead to having a 3-sided cycle.

On the other hand, edge introductions between vertices 3 and 7 and vertices 4 and 6 satisfy the size criteria individually as well as in combination (Fig. 4).

Thus, in order to detect and remove cases similar to the first multiple introductions discussed before, it will be required to check the cycle size validity criteria considering all the elementary cycles, e.g. in the case being considered of multiple edge introductions, the elementary cycles present are C_1 (1–2–3–6–5–1), C_2 (1–2–3–4–6–5–1) and C_3 (3–4–6–3), having sizes 5, 6 and 3, respectively, even though the intended cycles were only C_1 and C_2. In graph theoretical terms, C_1 and C_2 are the fundamental sets of cycles while C_3 is a derived cycle. The term elementary cycles here has the standard graph theoretical definition, and from now on, the term cycle is considered to be an elementary cycle unless stated otherwise.

It will thus suffice to identify the fundamental set of cycles corresponding to the smallest sizes. The starting fundamental set of cycles corresponds to the cycles directly resulting from edge introductions. Any cycle enumeration algorithm can then be used to enumerate all the cycles present. We have considered the algorithm by Gibbs [17] which is a cycle vector space method in which the cycles of the fundamental set form the basis of the cycle vector space. With this vector space



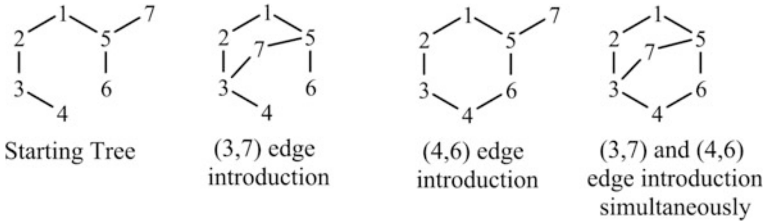
Starting Tree

(3,6) edge introduction

(4,6) edge introduction

(3,6) and (4,6) edge introduction simultaneously

Fig. 3 Multiple cycle introduction example (1)



Starting Tree

(3,7) edge introduction

(4,6) edge introduction

(3,7) and (4,6) edge introduction simultaneously

Fig. 4 Multiple cycle introduction example (2)

construct, one cycle, say C_3, can be obtained from two other cycles, say C_1 and C_2 from the previous example by a symmetric cycle-plus operation $\oplus$ defined below:

Let an edge between vertices i and j be denoted by e_{ij}. Let a cycle be denoted by the set of all such edges present in the cycle. Then for any two cycles A and B, the result of cycle-plus operation is:

$$A \oplus B = \left\{ e_{ij} | e_{ij} \in A \cup B,\ e_{ij} \notin A \cap B \right\} = (A \cup B) \backslash (A \cap B)$$

The same operation can be performed computationally faster when all the edges present in the graph are assigned a unique number and a given cycle is represented by a bit string where bit positions from right are set "on" corresponding to the unique numbered edges in the cycle. The cycle-plus operation is then exactly analogous to the bit-wise XOR (^) operation, i.e. $A \oplus B \Leftrightarrow A^{\wedge}B$.

At this point, it is worthwhile to note that the following property, henceforth called *Property* 1, of the cycle-plus operator holds, which is proved using XOR operation on bit string representation of cycles A and B:

$$\begin{aligned} A \oplus (A \oplus B) &\Leftrightarrow A^{\wedge}(A^{\wedge}B) \\ &\Leftrightarrow (A^{\wedge}A)^{\wedge}B \quad \text{By associative property} \\ &\Leftrightarrow 0^{\wedge}B \Leftrightarrow B \end{aligned}$$

Hence, $A \oplus (A \oplus B) = B \quad \text{Property}(1)$

In terms of cycles, the result of the cycle-plus operation can either be another cycle or a union of cycles having no common edges. Thus, all the cycles present in the graph can be obtained by linear combination of cycles taken two at a time in the fundamental set, supplemented successively by the increasing number of cycles and union of cycles obtained through cycle-plus operation. In the end, the entries that supplemented the fundamental set should only be cycles and the edge disjoint union of cycles should be removed. The final set so obtained will be the set of all cycles, say in the considered example the final set will be $\{C_1, C_2, C_3\}$ starting from the fundamental set $\{C_1, C_2\}$.

It is easy to comprehend and evident from the previous example that the final set may contain cycles smaller in size than those in the starting fundamental set of cycles. Moreover, as the cycles are generated by linear combination over two cycles at any given time using cycle-plus operator and as Property (1) holds, any resultant cycle in combination with a fundamental cycle will yield the other fundamental cycle from which it was produced. This is to say, in previous case, C_2 can be obtained from C_1 and C_3.

Thus, the entire fundamental set can be changed to another fundamental set which contains only the cycles of non-decreasing number of sides starting from the smallest sized cycle, so that all the cycles in the final set can still be generated. Henceforth, the term fundamental set will correspond to this newly constructed set. It can be noted, though, that the cardinality of the fundamental set does not get altered. In the examples considered so far, this will lead to a change of fundamental set from $\{C_1, C_2\}$ to $\{C_1, C_3\}$ while the set of all cycles will still remain $\{C_1, C_2, C_3\}$. This, arguably, is just an instance of change of basis in the cycle vector space.

It will now suffice to check the sizes of the cycles in the fundamental set against the required sizes and keep or discard the generated structure accordingly. This decision made, considering the fundamental set only, is in accordance with the IUPAC convention of the number of rings in polycyclic systems [26] where the number of rings is equal to the minimum number of scissions required to convert the system into an open chain compound or structure. Following this convention of ring count, the example corresponding to Fig. 4 will be a valid structure against the cycle size restriction either being 5 or 6.

(c) **Removal of duplicate cyclic structures using graph canonicalization:**

Although the trees generated by the algorithm given by Beyer et al. [16] are non-isomorphic (hence distinct structures), it is easy to comprehend that introduction of edges may lead to generating more than one chemical structure of same topology. As the entire process starts with tree structure, consider the case of the rightmost tree representation shown in Fig. 2, and two different edge introductions for a given cycle size constraint of 6 and cycle count constraint of 1 as shown in Fig. 5.

Although the presented example is basic in nature, the problem aggravates when the number of nodes is fairly large and such node pairs lie in different branches, sometimes far apart. For example, the molecules with 30 or more non-hydrogen atoms are fairly common in organic compounds developed as pharmaceutical entities. Moreover, even when the graph topology is uniquely fixed, the combinatorial imposition of node colours for imparting heterogeneity by introducing different atoms and the imposition of multiplicity of bonds can again lead to duplicate structures. Hence, any duplicate elimination strategy should consider the complete graph along with heterogeneity and bond multiplicity.

In the above context, molecular graph canonicalization algorithms can be used to identify the duplicate structures and eliminate them during generation. As we intend to store the molecules in SMILES notation format, it has been decided to use the algorithm proposed for generation of unique SMILES by Weininger et al. [27], which tackles the molecular graph canonicalization by extended connectivity through an unambiguous function using product of primes.

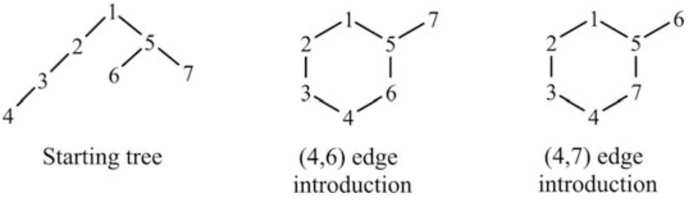


Fig. 5 Duplicate cyclic structures

The algorithmic steps leading to unique SMILES generation is discussed below:

(I) ***Initializing Rank of the Graph Vertices***—The rank initialization of the vertices is achieved using combined invariants which in turns are combinations of several individual atomic invariants. A total of 6 such atomic (node) invariants in the order of their priority are produced below:

(i) Number of connections
(ii) Number of non-hydrogen bonds
(iii) Atomic Number
(iv) Sign of Charge
(v) Absolute Charge
(vi) Number of attached hydrogen atoms.

It may be noted that the number of invariants can be varied based on the desired distinguishing properties [27]. The combined invariant will be the number obtained by successively concatenating the individual invariants such that higher priority invariants are to the left of lower priority invariants in the decimal system. For example, a methyl carbon (CH_3) in a molecule will have the individual invariants 1, 01, 06, 0, 0, 3 listed in the order of their priority while the combined invariant will be 10106003. The distinct combined invariants in the molecule are then sorted and mapped to their position in increasing order, hereafter referred to as consecutive ranks. The mapped position becomes the initial ranks of the atoms. For example, in case of n-Pentane, i.e. (C_1–C_2–C_3–C_4–C_5), where the subscripts denote the vertex labels, the combined invariants are 10106003–20206002–20206002–20206002–10106003 while the initial rank is 1–2–2–2–1.

(II) ***Extended Connectivity through an Unambiguous Function using Product of Primes***—The initial rank will not be able to identify the vertex symmetries. In the case of n-Pentane, vertices 2 and 4 are equivalent in terms of vertex symmetry while vertex 3 is not equivalent to them but is still initially ranked the same. To resolve this, rank of an atom is replaced by the result of an operation of a given function over its neighbours. This result is a representation of extended connectivity. A simple and elegant function is the product of primes corresponding to the rank of the neighbouring atoms. For example, in the n-Pentane case discussed so far, the updated rank of vertex 2 will now be prime number corresponding to rank of vertex 1 multiplied by prime number corresponding to the rank of vertex 3, i.e. $1\text{st}\,\text{prime} \times 2\text{nd}\,\text{prime} = 2 \times 3 = 6$, as ranks of vertices 1 and 3 are 1 and 2, respectively. Similarly, the rank of vertex 3 will be updated to $2\text{nd}\,\text{prime} \times 2\text{nd}\,\text{prime} = 9$. Subsequently, the revised rank will become 3–6–9–6–3 which can be remapped to consecutive ranks 1–2–3–2–1. This procedure of rank update is repeated and is stopped when the updated rank for each atom of the molecule remains same as the previous rank. It may be noted that in the end, the connectivity symmetrical vertices will be ranked the same.

(III) ***Tie Breaking***—The product of corresponding primes will yield same rank for connectivity symmetrical vertices. In such cases, the ties can be broken by arbitrarily choosing a node corresponding to the smallest repeating rank, doubling all the ranks and then reducing only the rank of the chosen vertex by one. The non-consecutive ranks so obtained are then remapped to form consecutive ranks, and the extended connectivity procedure using product of primes is performed to update ranks as described in the previous step. This step of breaking ties followed by rank updates is repeated until all the ties are broken and highest rank becomes equal to the number of vertices in the graph. The completion of this step also marks the completion of canonicalization of the graph.

(IV) ***Initial Vertex Selection and Branching Decisions for Traversal***—With the completion of graph canonicalization, the only steps required for unique SMILES generation is depth-first traversal sequence and identification of ring closures and their order in traversal. To start with, the lowest ranked atom is chosen for traversal. At a branching vertex, the branches are followed in the increasing order of the ranks of the neighbouring vertices; i.e. the branch corresponding to the lowest ranked neighbour is traversed first, then the second lowest ranked neighbour is followed and so on. It may be noted that Weininger et al. [27] also suggest giving branching preference towards the double or triple bonds in a ring even though the rank corresponding to such a vertex may be greater than other neighbouring vertices. However, this further complicates the final traversal sequence in the case of polycyclic compounds while the omission of this preference will save some computation time but will still generate unique SMILES.

V) ***Two-pass Approach***—Although, initially, the ring closures for the compounds are the edges that were introduced by joining vertices in the canonical trees, those edges will not be the ring closures under the depth-first traversal approach of the canonicalized graph and the traversal rule as given in the previous step. Additionally, the rings are to be numbered in the opening order in which they are encountered during traversal. In order to meet these requirements, the graph is traversed two times. During the first pass, the ring closures and their ordering are identified for the canonicalized graph and are stored as auxiliary data. The edges corresponding to these new ring closures will now be treated as if they were the edges introduced to complete the cyclic structure, while the tree obtained by removal of such edges is treated now as the spanning tree. Subsequently, the second pass is undertaken for SMILES string generation using the previously obtained auxiliary data.

2.5.2 Structure for a Relaxed Distance Distribution

The approach taken so far suffers from the drawback that only those compound structures will be generated that have the same number of non-hydrogen atoms as the starting molecule from which the distance distribution was obtained. This subsection tries to tackle this drawback by slightly relaxing the distance distribution matching criteria for the trees with number of vertices deviating from the source or starting distribution. This deviation can either lead to increased or decreased number of vertices.

(a) Non-Isomorphic Canonical Tree Generation with Relaxed Distance Distribution

The first step involves specifying the number of vertices (after factoring in the deviation) and then generating the trees. Positive deviation means required number of vertices is greater than that in the current tree while negative deviation means the required number of vertices is lesser. However, since exact distance distribution matching is not possible in this case, two variants of relaxed distribution matching are considered as explained below:

Strong matching—This situation arises when the distance distribution of the generated tree can be obtained from the starting/source distance distribution by either adding or deleting vertices at any level (named node deviation) although simultaneous insertion or deletion of vertices is not allowed for a given deviation. In essence, the obtained distance distribution corresponds to a pruned tree of the source distance distribution if the node deviation is negative and vice versa if the node deviation is positive.

Thus, to put it mathematically, if trees are to be generated by decreasing or increasing n number of vertices, then only n deletions or insertions are allowed so that:

$$\left| \sum_{i=1}^{e} \left(c_i^s - c_i^p \right) \right| = n$$

where c_i^s is the count of vertices at level i in the source distance distribution; c_i^p is the count of vertices at level i in the present distance distribution under consideration; and e is the maximum of the eccentricity of the source and present distance distribution.

Weak matching—In this case, the distance distribution matching criteria is further relaxed in that one can add and delete vertices simultaneously at any level. This, in effect, executes migration of vertices from one level to another (named node migration). If this is allowed without a cap on the number of node migrations, then all the possible structure generation will be considered a match which will include the linear chain too. Presumably, in order to match the source distance distribution closely using weak matching criterion, number of allowed node migrations should be provided preferably of low value.

For this exercise, if the trees are to be obtained by decreasing or increasing n number of vertices, then n deletions or insertions along with m migrations are allowed that satisfies the following criteria:

$$\left|\sum_{i=1}^{e}\left(c_i^s - c_i^p\right)\right| = n$$

and

$$\min\left(m_p, m_n\right) = m$$

where

$$m_p = \sum_{i=1}^{e} max\left(\left(c_i^s - c_i^p\right), 0\right)$$

$$m_n = \left|\sum_{i=1}^{e} \min\left(\left(c_i^s - c_i^p\right), 0\right)\right|$$

Here c_i^s, c_i^p and e have the same meaning as defined in the case of strong matching while m_p is the sum of vertex surplus and m_n is the sum of vertex deficit in the source distance distribution over the present distance distribution.

The procedure of cycle introduction, canonicalization and unique SMILES notation generation is the same as done before.

Now, once the structures are generated using the methods described above, one can use some user-defined parameters incorporated in the computer program to restrict the number and size of the cycles to be created in the 2D structures. Few other user-defined parameters, available in the program, may also be used to add multiplicity of bonds (double and triple bonds) between pairs of vertices and other hetero-atoms (e.g. nitrogen, oxygen, halogens) in order to get complete 2D structures of the compounds. The output of the generated structures may be saved in SMILES notations and can be viewed using a molecular modelling software that is capable of getting molecular structures from SMILES notation. Subsequently, the activities of the generated structures may be predicted using the rule-based method [18, 19] standardized for a biological endpoint of interest and can be prioritized and screened from their MPS values. In this way, one may be able to screen some potential bioactive compounds from the bigger set of combinatorially generated molecular structures using topological distance information associated with activity-related vertices present in the active compounds of a data set under consideration. It may be worth noting at this point that this newly developed method [15] is essentially a molecular topology-based approach and activity prediction is done using molecular graphs of the compounds where bond multiplicity and atom types are not required. However, since bond multiplicity and atom types can be introduced in the combinatorially generated topological structures using the options available in the program and those structures can be saved in SMILES format, one can always use these generated structures for any 2D and 3D drug design/discovery applications.

3 Results and Discussion

We furnish in this section the results obtained using the method, described in the previous section, that can generate chemical structures combinatorially using activity-related substructural topological information, predict activity for the biological endpoints under consideration, prioritize compounds and screen them to help discover novel therapeutic candidates. The results given here are for a series of 19 convulsant–anticonvulsant barbiturates [18], a series of 20 nucleoside analogues (NA) having anti-HIV activities [20, 21] and a data set of 3779 compounds [22, 23] for which minimum inhibitory concentration (MIC) values have been measured against H37Rv strain of *Mycobacterium tuberculosis* (Mtb).

3.1 Activity Prediction–Compound Prioritization–Molecular Design

We describe in this section the results obtained for combinatorial structure generation from the substructural information of activity-related vertices (atoms), activity prediction using a rule-based system [18, 19] and prioritization and screening of potential drug candidates using a newly defined Molecular Priority Score (MPS) [15]. The application of different algorithms incorporated in the computer program developed using the method, and the results obtained therefrom are given here and discussed accordingly. In particular, the method has been used for activity prediction, compound prioritization using MPS and structure generation considering barbiturates and the NA series of compounds. On the other hand, structure matching algorithm based on distance distribution has been used for searching potential antitubercular compounds from the data set of 3779 compounds mentioned above.

3.1.1 Studies with Barbiturates

The activity prediction for the series of barbiturates [18] considered for the present study is reported here using the rule-based method [18, 19] considering hydrogen-filled (H-filled) graphs of the compounds. Along with activity prediction considering H-suppressed graphs, the method also supports activity prediction using H-filled graphs and that option available in the computer program has been used for the activity prediction studies with the barbiturates. The R-groups of the barbiturates considered here and built on the core structure shown in Fig. 6 are given in Table 1.

Activity prediction for this series of compounds has already been reported [18] by considering information theoretical vertex indices V^d (vertex distance complexity) and V_n^d(normalized V^d), which are also available in this software for use. Although V_n^d has produced very high percentage of correct predictions [18], we

Table 1 A series of 19 barbiturates[a] considered for the present study

	R-group		R-group
1.	$–(CH_2)_3CH_3$	11.	$–(CH_2)_3C_6H_{11}$
2.	$–CH(CH_3)(CH_2)_2CH_3$	12.	$–(CH_2)_2CH{=}C_6H_{10}$
3.	$–(CH_2)_2CH(CH_3)_2$	13.	$–(CH_2)_2CH{=}C_5H_8$
4.	$–CH(CH_3)CH_2CH(CH_3)_2$	14.	$–CH_2C_6H_5$
5.	$–CH{=}CHCH_2CH_3$	15.	$–CH_2CH(CH_3)C_6H_5$
6.	$–C(CH_3){=}CHCH_2CH_3$	16.	$–CH{=}(CH)_2(CH_3)_2$
7.	$–CH_2CH{=}CHCH_3$	17.	$–C(CH_3){=}(CH)_2(CH_3)_2$
8.	$–CH(CH_3)CH{=}CHCH_3$	18.	$–(CH_2)_3C_6H_5$
9.	$–CH_2CH{=}C(CH_3)_2$	19.	$–(CH_2)_2C_6H_5$
10.	$–CH(CH_3)CH{=}C(CH_3)_2$		

[a] The data have been taken from Klopman and Raychaudhury [18]

Table 2 Assigned and predicted activities using D^{-4} index and Molecular Priority Score (MPS) of 19 barbiturates divided into 15 training set and 4 test set compounds

Sr. no.	Compound no.	Activity[a]		MPS[b]
		Assgn.	Pred.	Value
Training set				
1	1	+	+	93
2	9	+	+	10
3	10	+	+	56
4	12	+	+	178
5	13	+	+	168
6	15	+	+	34
7	2	–	–	−132
8	3	–	–	−102
9	4	–	–	−132
10	5	–	–	−113
11	6	–	–	−100
12	7	–	–	−120
13	8	–	–	−74
14	11	–	–	−149
15	14	–	–	−64
Test set				
1	17	+	+	6
2	19	+	+	53
3	16	–	–	−97
4	18	–	–	10

[a] (+) means active and (−) means inactive

[b] Computation of MPS value is described in methods section

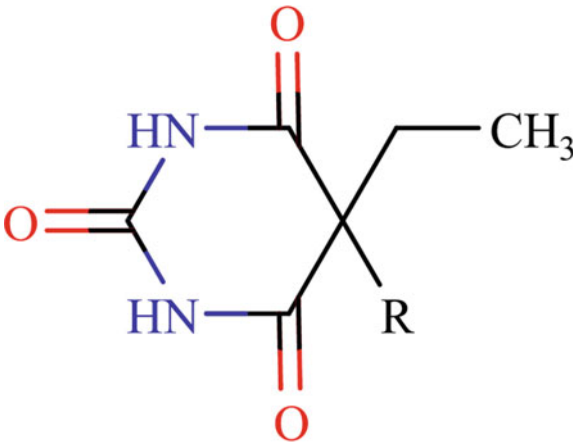


Fig. 6 Barbiturate core structure with R-group (Table 1) attachment point (R)

present here the results obtained using distance exponent index (D^{-4}) to see how this index performs for this series of compounds. The activity prediction results along with MPS values using D^{-4} index, computed for the hydrogen-filled graphs of the compounds, are shown in Table 2. It may, however, be noted that the indices of only non-hydrogen atoms have been considered for ordering of index values, range selection and activity prediction purposes. Thus, the indices computed for the hydrogen atoms in the H-filled graphs have not been used for this purpose.

Activity Prediction and Compound Prioritization for Barbiturates

For the prediction of activity and prioritizing the compounds on the basis of MPS values, we have considered the same set of compounds as well as the same training set and test set for the present study as used earlier [18]. In may be noted that, in this data set, the convulsant barbiturates are tagged active and the anticonvulsant barbiturates as inactive.

It can be observed that accuracy of activity prediction using D^{-4} index in the barbiturate data set is 100% for both training set and test set which equals the prediction obtained using V_n^d index reported earlier [18]. This further substantiates earlier findings [15] using this vertex index, rule-based method and MPS value about the usefulness of the method for activity prediction and compound prioritization. This is believed to help scientists work on the crucial issues related to convulsion and help drug designers find novel therapeutic agents in the area of anticonvulsant drug discovery.

Structure Generation for Barbiturates

The structure generation exercise has been carried out for the barbiturate data set with the same training set and test set split as considered earlier [18]. The index computation for the non-hydrogen atoms (vertices) has been performed considering hydrogen-filled graphs. As described in the method section, the D^{-4} index values computed for the training set compounds are arranged in an ascending order to find active and inactive ranges in order to get a "*strong*" range to identify an

Table 3 Details of the range in which vertices 17 and 18, in the molecular graph of compound no. 13, lie in

Serial no.	D^{-4} index value	Compound no. (Atom no.)	Activity
1	4.40994	13(16)	+
2	4.40994	13(19)	+
3	4.430099	12(16)	+
4	4.430099	12(20)	+
5	4.430937	13(17)	+
6	4.430937	13(18)	+
7	4.440002	1(14)	+
8	4.441781	13(13)	+
9	4.444924	12(13)	+
10	4.449867	12(18)	+
11	4.451095	12(17)	+
12	4.451095	12(19)	+

(+) means active, (−) means inactive

activity-related vertex to start structure generation considering that vertex as the root vertex. It has been observed that the vertices 17 and 18 (the numbers correspond to those in the respective SMI file used to work with the compounds considered) in the molecular graph representing compound no. 13 (Table 1), an active compound, fall in a strong range. Interestingly, when these two vertices are chosen

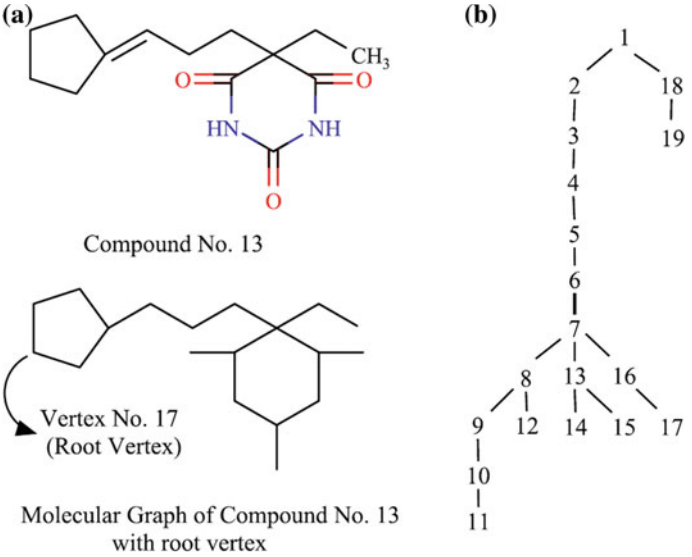


Fig. 7 **a** Compound no. 13 (Table 1), its molecular graph and the root vertex (vertex no. 17). **b** Sample rooted tree structure generated. In the tree, the root vertex is labelled as vertex 1

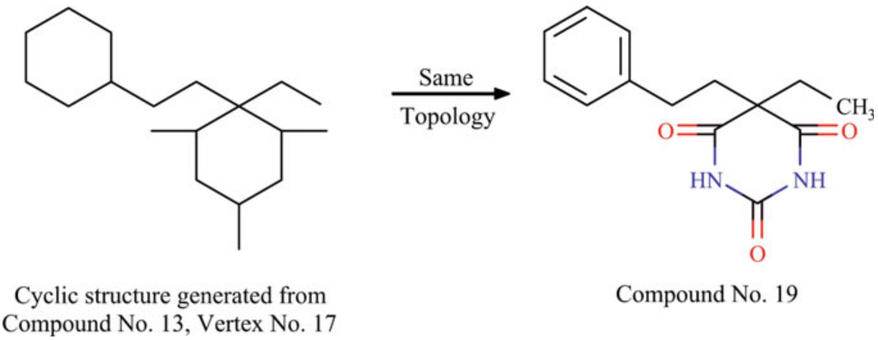

Same
Topology
CH3
O
O
HN
NH
O
Cyclic structure generated from Compound No. 13, Vertex No. 17
Compound No. 19

Fig. 8 One of the structures generated, from compound no. 13, which resembles the topology of compound no. 19 (Table 1)

for structure generation, both of them lead to the generation of a topological structure of another active compound. The details of the strong active range are given in Table 3 and the structure generation details in Fig. 8.

The compound no. 13 along with its molecular graph and the chosen structure generation vertex (root vertex) is given in Fig. 7a. The distance distribution associated with this vertex (Vertex No. 17) starting with distance 0 is (1, 2, 2, 1, 1, 1, 1, 3, 5, 1, 1). A sample rooted tree is shown in Fig. 7b with the corresponding distance distribution.

Considering any rooted tree, cycles can be introduced (described in the methods section) to generate the topology of the structural formula of variety of chemical compound while still maintaining the distance distribution. In the present study, we have chosen to generate structures containing two cycles, having number of sides 5 or 6, to investigate whether we are able to generate any other active compound present in the studied data set. A number of structures are generated in the process, and it has been found that the structures generated from the root vertex of compound no. 13 contain one such structure that matches with that of compound no. 19 (Fig. 8). It is interesting to note that compound no. 19 is an active compound from the test set (Table 2) which shows that the method can generate a structure that it has not seen in the training set. Therefore, one can expect to design novel structures using this method.

3.1.2 Studies with Nucleoside Analogues

For the nucleoside analogues (NA), we have carried out activity prediction and structure generation studies. It may be noted that for this series of compounds, we have investigated the performance of the training set–test set identification tool using the corresponding algorithm incorporated in the computer program. As mentioned earlier, in this way we are able to obtain a suitable training set for the system's learning and predict activities of the compounds on the basis of this

Table 4 A series of 20 nucleoside analogues [a] considered for the present study

	Compound Name		Compound Name
1.	3′-deoxyadenosine	11.	2′-deoxyinosine
2.	2′-deoxycytidine	12.	2′,3′-dideoxythymidine
3.	2′-deoxyadenosine	13.	2′,3′-dideoxyuridine
4.	2′,3′-dideoxyadenosine	14.	2′,3′,5′-trideoxyadenosine
5.	2′,3′-dideoxycytidine	15.	3′-amino-2′,3′-dideoxycytidine
6.	3′-fluoro-2′,3′-dideoxythymidine	16.	3′-amino-2′,3′-dideoxyadenosine
7.	3′-azido-2′,3′-dideoxythymidine	17.	2′-deoxyguanosine
8.	2′,3′-dideoxyinosine	18.	3′-azido-2′,3′-dideoxyadenosine
9.	2′,3′-dideoxyguanosine	19.	3′-azido-2′,3′-dideoxycytidine
10.	5′-iodo-2′-deoxycytidine	20.	3′-azido-3′-deoxyadenosine

[a] Data were taken from Raychaudhury et al. [20, 21]

training. This section, therefore, contains the results of the performance of training set identification and activity prediction. We have also reported here the results of structure generation for some of the NA series compounds in the same way as it has been done for the barbiturate series. For identifying a suitable training set–test set combination for the purpose of identifying a suitable training set that can produce high percentage of successful activity predictions, the program generates 1000 such combinations. The program has the option of getting the output on the basis of best test set predictions (starting from no misprediction) and best training set predictions. It has been observed that there are combinations where no mispredictions are found for the training set although there are 2 or more mispredictions for the test sets. On the other hand, there are combinations where there is one misprediction each for both the training set and the test set and it seems quite reasonable to consider such a balanced combination for activity prediction of newly generated compounds. We have reported here the activity predictions and MPS values of such a balanced outcome in Table 5 for the nucleoside analogues (NA) considered for the present study given in Table 4. The structural information of the compounds has been taken from the corresponding MOL files.

Activity Prediction for Nucleoside Analogues

For carrying out activity prediction and prioritization studies for NA series of compounds, we have used training set–test set split algorithm and the prediction results for split that has given one misprediction each for the training set and the test set are reported here.

It can be seen that for this NA series, activities of 92.86% (13 out of 14) of the training set compounds and 83.33% (5 out of 6) of the test set compounds have been predicted correctly, compound no. 10 of the training set and compound no. 13 of the test set being the lone mispredictions in each case. It is interesting to note that in both the cases the inactive compounds have been predicted to be active which may be regarded as an important factor in situations where a drug designer

Table 5 Assigned and predicted activities using D^{-4} index and Molecular Priority Score (MPS) of 20 nucleoside analogues divided into 14 training set and 6 test set compounds

Sr. no.	Compound no. #	Activity[a]		MPS[b]
		Assigned	Predicted	Value
Training set				
1	4	+	+	65
2	5	+	+	83
3	6	+	+	8
4	7	+	+	103
5	9	+	+	55
6	18	+	+	97
7	19	+	+	98
8	1	–	–	−56
9	2	–	–	−36
10	3	–	–	−48
11	10	–	+	8
12	14	–	–	−13
13	15	–	–	−36
14	16	–	–	−48
Test set				
1	8	+	+	65
2	12	+	+	65
3	20	+	+	50
4	11	–	–	−48
5	13	–	+	83
6	17	–	–	−6

[a] (+) means active, (−) means inactive and (#) means incorrect prediction

[b] The details for the computation of MPS value are described in methods section

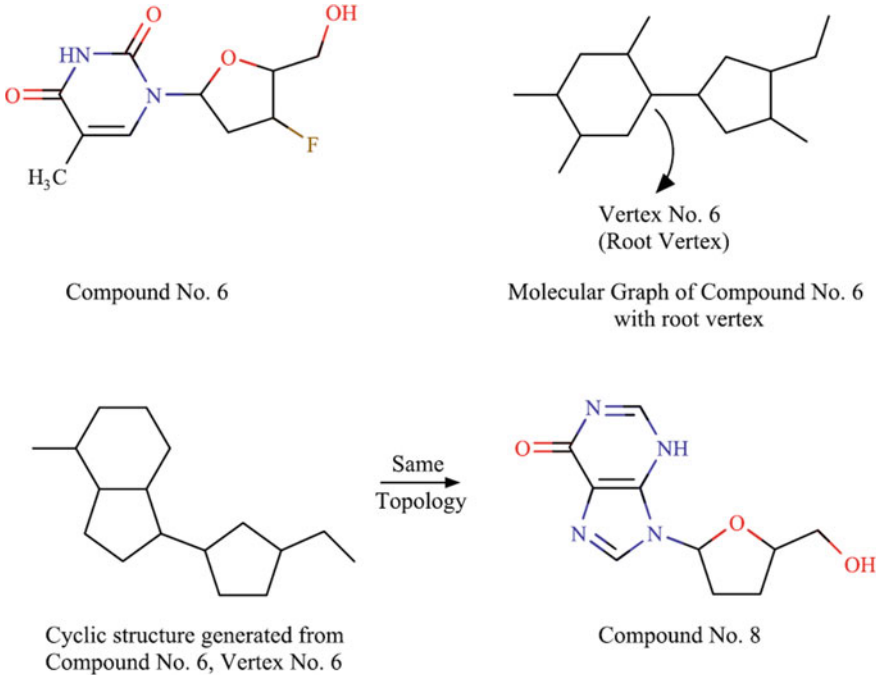
#Compound numbers are correspond to those in Table 4

does not want to lose any potential active compound/drug candidate particularly the one like the mispredicted compound of the test set (compound no. 13) which has a high MPS value (MPS = 83). Clearly, a number of active compounds have got high MPS values including compound no. 8 which represents a potent anti-HIV drug—Didanosine—and is a test set compound (Table 5). The method has also produced high MPS values for a number of training set active compounds too like compound nos. 5, 7, 18, 19 (Table 5). Therefore, picking at least a couple of top scoring (from MPS values) compounds out of them from prioritization point of view may help screen useful drug candidates using the present method. This finding therefore indicates that this method can be used for creating suitable splits in getting a reasonably useful training set from an available data set and help screen putative active compounds for drug discovery.

Structure Generation for Nucleoside Analogous

As done for the barbiturates, structure generation from various starting points, i.e. compound no., atom no., was carried out for the NA series of compounds too. In doing that, activity-related vertices have been picked up from the strong ranges in the ordering of D^{-4} index values for the vertices (atoms) of the H-suppressed graphs of these compounds. It has been found that a few carbon skeletons resembling the structure of other active compounds than the ones from where the activity-related vertices and the corresponding distance distribution values are taken have been generated.

For the purpose of illustration, the structure of the compound no. 6 and the generated structure which corresponds to compound no. 8 are shown in Fig. 9. It can be seen that in this case too, the algorithm is able to generate a structure with significantly different scaffold than the starting compound and has a higher MPS value (MPS = 65) too compared to that (MPS = 8) of the starting structure indicating that this generated structure has the potential of being highly active and therefore may be picked/prioritized for further studies. In fact, compound no. 8 is a potent anti-HIV drug—Didanosine. Therefore, the method may be regarded as a useful tool for generating, prioritizing and discovering potent anti-HIV compounds. Moreover, the generated compound belongs to the test set indicating that the structure of a compound that has not been used for training the system can also be



Compound No. 6

Molecular Graph of Compound No. 6 with root vertex

Cyclic structure generated from Compound No. 6, Vertex No. 6

Compound No. 8

Fig. 9 Compound no. 6, its molecular graph with root vertex and one of the structures generated from compound no. 6 that resembles the topology of compound no. 8

designed by this method which may be believed to carry higher importance for discovering novel therapeutic candidates.

3.2 Rooted Substructure Searching for Drug Discovery

In the previous section, we showed how the exact matching algorithm can help find structures of active compounds which could be obtained from the trees generated from the topological distance distribution information of activity-related vertices obtained from other active compounds. In this section, we describe the use of two other matching algorithms—strong matching and weak matching—along with exact matching algorithm for searching active compounds in a data set in the form of tree and sub-tree matching. As given in the method section, these sub-trees are obtained by means of applying node deviation and node migration in the actual tree obtained from the distance distribution associated with an activity-related vertex. The presence of such trees and sub-trees are then searched for in the compounds present in a data set to identify potential drug candidates. In doing that, we have considered two known TB drugs—Isoniazid and Streptomycin—to describe the usefulness of the present method in finding potential antitubercular compounds from a data set (named GTB data set) of 3779 compounds [22, 23] for which *MIC* values against H37Rv strain of *Mtb* have been measured. The authors have made $MIC = 5.0$ as the cut-off point and the *MIC* value of any compound which is higher than 5.0 give an inactive compound in the data set. It therefore seems reasonable to consider the same cut-off value for the present purpose. We will first furnish the results obtained for Isoniazid which will be followed by those obtained for Streptomycin. It may be noted that the activity-related vertices for both Isoniazid and Streptomycin have been taken from the literature information and not by using rule-based method in the ordering of vertex indices which has been done for the barbiturate and NA series of compounds. In fact, it shows that the method can be used successfully in identifying potential drug candidates by picking activity-related vertices by other means than by the rule-based method.

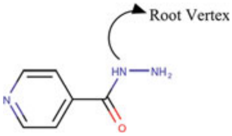
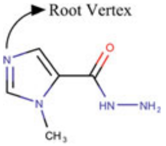
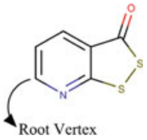
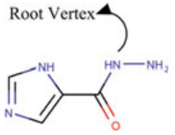
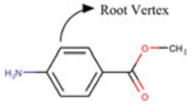
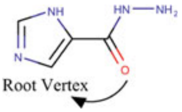
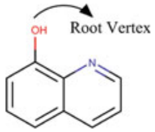
3.2.1 Studies with Isoniazid

Isoniazid is a known first line drug for the treatment of tuberculosis. However, it may become resistant in situations, and therefore, this leads researchers look for novel drug candidates to overcome drug resistance problem for the treatment of tuberculosis . We have described in this subsection how structures generated from activity-related vertex information of Isoniazid using the present method can help search for potential TB drugs from a data set of 3779 compounds [22, 23]. It is known that the chemical/biochemical reaction takes place at the point of the first nitrogen (N) atom (underlined) of the fragment (–N̲H–NH2) in isoniazid molecule to convert this pro-drug into its metabolite that works as the effector molecule. Therefore, this vertex (N atom) may be regarded as an activity-related vertex for Isoniazid. Accordingly, the distance distribution associated with the vertex representing this nitrogen (N) atom has been considered for generating structures. In order to screen out potential antitubercular compounds having high activities, the exact, strong and weak matching algorithms (method section) have been applied on the GTB data set of 3779 compounds considered for the present study. A number of highly active compounds have been obtained in the process and the information for some of them obtained applying different node deviation and node migration on the tree obtained from the distance distribution associated with the root vertex are shown in Table 6 along with the structures of Isoniazid (with root vertex specified) and the screened compounds. As said earlier, in their studies [22], the researchers have considered a compound having *MIC* value less than 5.0 to be active. In this way, data set is composed of almost equal number of active and inactive compounds implying no bias for active or inactive compounds in forming the data set. Accordingly, compound nos. 1–1890 are active compounds and the other compounds are inactive. Considering the same cut-off value, one can see that only compound no. 3296 has *MIC* value higher than 5.0 and the rest of the compounds may be screened out as potential active compounds. In particular, compound no. 180 which is obtained by two types of node deviation and node migration in generating structures from the root vertex has quite low *MIC* value which identifies it as a highly active compound. Therefore, the result clearly shows that the method may be used to successfully screen potentially highly active antitubercular compounds from this data set starting from Isoniazid.

3.2.2 Studies with Streptomycin

Streptomycin is another antitubercular drug in use, an antibiotic. For this compound, the removal of even one of the two guanidino groups present in the structure reduces the activity of the compound. Considering that, we have taken the vertex representing the nitrogen (N) atom in one of the guanidino groups as the root vertex to start generating/designing novel structures. Out of a number of structures designed using the present method, i.e., using exact matching as well as strong matching and weak matching algorithms in relation to node deviation and node migration on the trees obtained from the distance distribution associated with the root vertex, information about some of these compounds are given in Table 7 along with the structures of Streptomycin having root vertex indicated and the matched/searched compounds from GTB data set. It is found from this table that all the compounds shown here are active according to the adopted criterion ($MIC \leq 5.0$ is active) with compound no. 183 being the most active among them. Therefore, it appears from this finding that the method may be used successfully to screen potentially highly active antitubercular compounds from the data set of 3779 compounds starting from Streptomycin.

Table 6 Screened compounds obtained from the matching of trees/sub-trees obtained from the generated structure from the root vertex (indicated) of Isoniazid molecular graph

<table>
<tr><td colspan="2">Source Compound</td><td colspan="2">
Isoniazid</td></tr>
<tr><td colspan="4">Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside</td></tr>
<tr><td>S. no.</td><td>Node deviation</td><td>Node migration</td><td>Matched compound</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>
Compound No 1387</td></tr>
<tr><td>2</td><td>0</td><td>0</td><td>
Compound No. 3296</td></tr>
<tr><td>3</td><td>1</td><td>0</td><td>
Compound No. 180</td></tr>
<tr><td>4</td><td>1</td><td>0</td><td>
Compound No. 1174</td></tr>
<tr><td>5</td><td>1</td><td>1</td><td>
Compound No. 180</td></tr>
<tr><td>6</td><td>1</td><td>1</td><td>
Compound No. 1192</td></tr>
</table>

(continued)

Table 6 (continued)

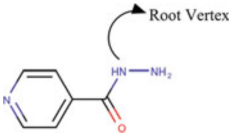
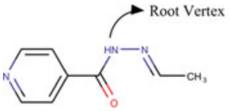
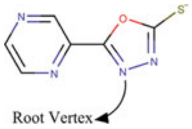
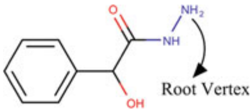
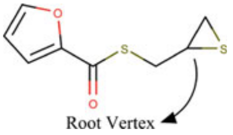
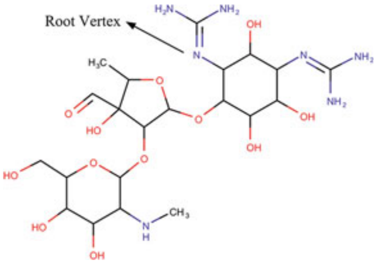
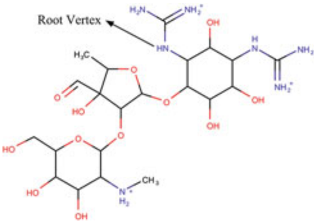
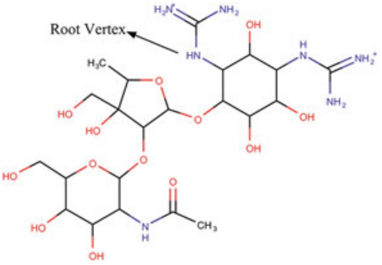
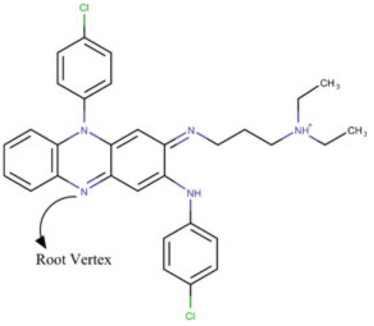
<table>
<tr><td>Source Compound</td><td colspan="3">
Isoniazid</td></tr>
<tr><td colspan="4">Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside</td></tr>
<tr><td>S. no.</td><td>Node deviation</td><td>Node migration</td><td>Matched compound</td></tr>
<tr><td>7</td><td>2</td><td>0</td><td>
Compound No. 524</td></tr>
<tr><td>8</td><td>2</td><td>0</td><td>
Compound No. 928</td></tr>
<tr><td>9</td><td>2</td><td>2</td><td>
Compound No. 661</td></tr>
<tr><td>10</td><td>2</td><td>2</td><td>
Compound No. 1333</td></tr>
</table>

Table 7 Screened compounds obtained from the matching of trees/sub-trees obtained from the generated structure using the root vertex in Streptomycin molecular graph

<table>
<tr><td colspan="2">Source compound</td><td colspan="2">


Streptomycin</td></tr>
<tr><td colspan="4">Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside</td></tr>
<tr><td>S. no.</td><td>Node deviation</td><td>Node migration</td><td>Matched compound</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>


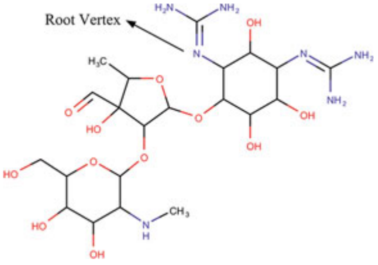
Compound No 183</td></tr>
<tr><td>2</td><td>2</td><td>0</td><td>


Compound No. 1483</td></tr>
<tr><td>3</td><td>2</td><td>1</td><td>


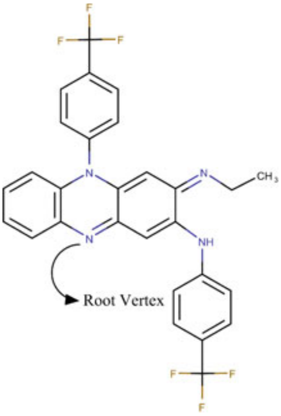
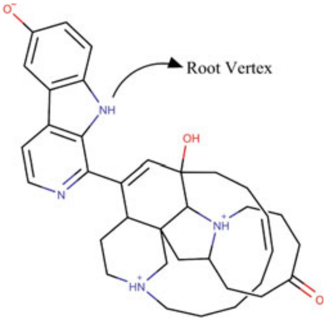
Compound No. 1059</td></tr>
</table>

(continued)

Table 7 (continued)

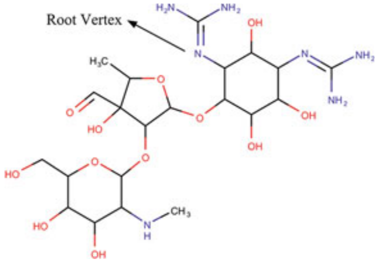
Source compound	 ***Streptomycin***

Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside

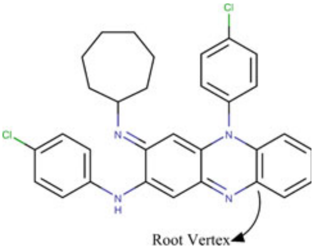
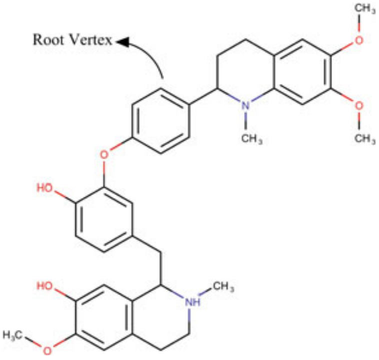
S. no.	Node deviation	Node migration	Matched compound
4	2	2	 ***Compound No. 468***
5	3	1	 ***Compound No. 1006***

(continued)

Table 7 (continued)

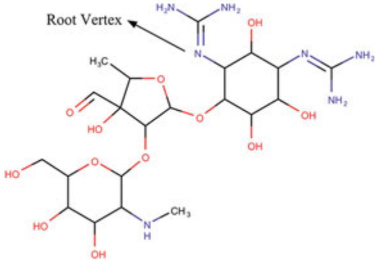
Source compound	
	 Root Vertex H₂N NH₂ N OH N NH₂ NH₂ H₃C O O HO O OH OH HO O O HO N H CH₃ OH ***Streptomycin***

Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside

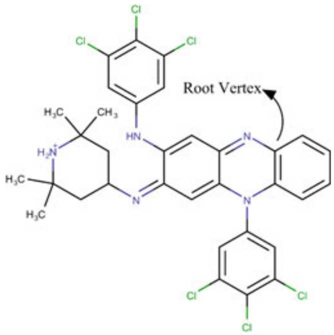
S. no.	Node deviation	Node migration	Matched compound
6	3	2	 ***Compound No. 671***
7	4	1	 Root Vertex CH₃ O N CH₃ O CH₃ O HO HO NH CH₃ H₃C O ***Compound No. 1287***

(continued)

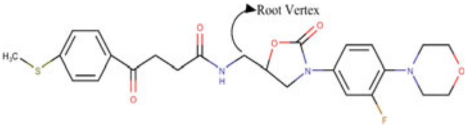
Table 7 (continued)

<table>
<tr><td colspan="2">Source compound</td><td colspan="2">

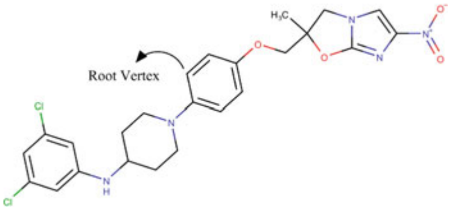
Root Vertex
H₂N
NH₂
OH
N
NH₂
NH₂
H₃C
O
O
HO
O
OH
OH
O
HO
O
N
H
CH₃
HO
OH

Streptomycin</td></tr>
<tr><td colspan="4">Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside</td></tr>
<tr><td>S. no.</td><td>Node deviation</td><td>Node migration</td><td>Matched compound</td></tr>
<tr><td>8</td><td>4</td><td>2</td><td>

Cl
Cl
Cl
Root Vertex
H₃C
CH₃
HN
N
H₂N⁺
H₃C
H₃C
N
N
Cl
Cl
Cl

Compound No. 211</td></tr>
<tr><td>9</td><td>5</td><td>0</td><td>

Root Vertex
H₃C
S
O
O
N
H
O
O
N
N
O
F

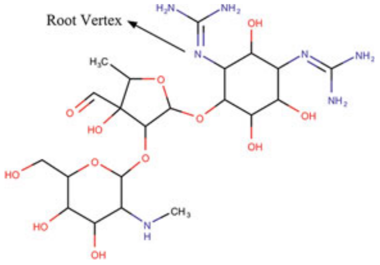
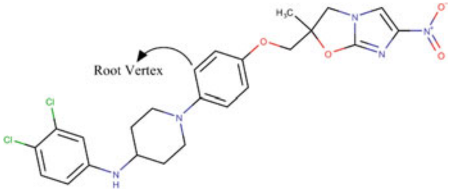
Compound No. 1086</td></tr>
<tr><td>10</td><td>5</td><td>1</td><td>

H₃C
N
O⁻
N⁺
O
O
O
N
Root Vertex
Cl
N
N
H
Cl

Compound No. 335</td></tr>
</table>

(continued)

Table 7 (continued)

Source compound	 Root Vertex H₂N NH₂ N OH NH₂ NH₂ H₃C O O HO O OH OH O HO O HO N H CH₃ OH ***Streptomycin***		
Compounds (in the Global TB data set) whose structures topologically matched with the source compound with the node deviation and node migration mentioned alongside			
S. no.	Node deviation	Node migration	Matched compound
11	5	2	 H₃C N O⁻ N⁺ O Root Vertex O O N Cl Cl N N H ***Compound No. 232***

4 Conclusions and Future Prospect

The results obtained for different series of compounds using recently developed graph theory-based drug design/drug discovery method by our group [15] for combinatorial drug design from substructural topological information have been described in this chapter. Its application and usefulness for different series of antitubercular compounds have already been reported [15]. In this chapter, we have presented some new results for designing active compounds for barbiturates [18, 19] and nucleoside analogues [20, 21]. We have also reported some new results obtained for discovering novel active compounds from a data set using rooted tree/sub-tree searching/matching algorithms. In doing that, a data set (GTB) of 3779 potential antitubercular compounds [22, 23] has been taken for this study and the method has helped search a number of potentially highly active antitubercular compounds from this data set. Thus, to our knowledge, we have introduced here a method that can be used for searching databases to discover novel drug molecules using rooted tree and sub-tree matching algorithms. Furthermore, the usefulness of newly proposed Molecular Priority Score (MPS) for prioritizing and screening highly active compounds has also been described for the studies with a series of convulsant–anticonvulsant barbiturates and a series on nucleoside analogues for their activities against HIV. It is also found that the proposed method is capable of generating structures of known active compound that has scaffold different from that of the starting one. Furthermore, the structure generation starts from a vertex which plays a role in predicting biological activity. These observations seem to address the relationship of the present method [15] with two important aspects of modern-day drug discovery research—scaffold hopping and inverse QSAR (iQSAR) problem. Therefore, it appears that this newly developed method [15] may find useful applications in designing novel therapeutic candidates and may be helpful for working with drug resistance problems where compounds of very different molecular architecture may be sought for.

Our work presents an interesting alternative to "3D" drug discovery, where actual molecular coordinates in Cartesian space is used. Combinatorial design and generation in three-dimensional space would be far more expensive compared to our approach. Interestingly, one can always follow up on "3D" drug discovery based on molecule predictions from our method. This would allow a far tractable approach to drug discovery compared to a seemingly infinite exploration of molecules in actual "3D" Cartesian space.

Regarding future work, it may be worth exploring whether application of any quantitative measure for activity prediction can help screen potential bioactive compounds more effectively. Also, incorporation of new rooted tree-based compound generation and searching algorithms in the existing computer program would be another important aspect to work on. Finally, it would be of special interest to see how incorporation of ADME/Tox and drug-able property filters in the computer program can help discover drug molecules having desired pharmacological and undesired toxicological activities using the present method.

References

1. Ruddigkeit L, Van deursen R, Blum LC, Reymond JL (2012) Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. J Chem Inf Model 52:2864–2875
2. Hansch C, Sammes PG, Taylor JB, Ramsden C (1990) Comprehensive medicinal chemistry: quantitative drug design, vol 4. Pergamon Press
3. Kier LB, Hall LH (1986) Molecular connectivity in structure-activity analysis. Research Studies Press
4. Stuper AJ, Brügger WE, Jurs PC (1979) Computer assisted studies of chemical structure and biological function. Wiley
5. Kitchen DB, Decornez H, Furr JR, Bajorath J (2004) Docking and scoring in virtual screening for drug discovery: methods and applications. Nat Rev Drug Discov 3:935–949
6. Cramer RD (2003) Topomer CoMFA: a design methodology for rapid lead optimization. J Med Chem 46:374–389
7. Sun H, Tawa G, Wallqvist A (2012) Classification of scaffold-hopping approaches. Drug Discovery Today 17:310–324
8. Tanwar J, Das S, Fatima Z, Hameed S (2014) Multidrug resistance: an emerging crisis. Interdiscip Perspect Infect Dis 2014

9. Gálvez J, García-Domenech R (2010) On the contribution of molecular topology to drug design and discovery. Curr Comput Aided Drug Des 6:252–268
10. Gugisch R, Kerber A, Kohnert A, Laue R, Meringer M, Rücker C, Wassermann A (2014) MOLGEN 5.0, a molecular structure generator. In: Advances in mathematical chemistry and applications, vol 1. Bentham Publishers, pp 113–138
11. Harary F (1972) Graph theory. Addison-Wesley
12. Faulon JL, Bender A (2010) Handbook of chemoinformatics algorithms. CRC press
13. Wong WW, Burkowski FJ (2009) A constructive approach for discovering new drug leads: using a kernel methodology for the inverse-QSAR problem. J Cheminform 1:4
14. Klopman G (1994) Artificial intelligence approach to structure-activity studies: computer automated structure evaluation of biological activity of organic molecules. J Am Chem Soc 106:7315–7321
15. Raychaudhury C, Rizvi MIH, Pal D (2018) Combinatorial design of molecule using activity-linked substructural topological information as applied to antitubercular compounds. Curr Comput Aided Drug Des https://doi.org/10.2174/1573409914666180509152711
16. Beyer T, Hedetniemi SM (1980) Constant time generation of rooted trees. SIAM J Comput 9:706–712
17. Gibbs NE (1969) A cycle generation algorithm for finite undirected linear graphs. J ACM 16:564–568
18. Klopman G, Raychaudhury C (1990) Vertex indexes of molecular graphs in structure-activity relationships: a study of the convulsant-anticonvulsant activity of barbiturates and the carcinogenicity of unsubstituted polycyclic aromatic hydrocarbons. J Chem Inf Comput Sci 30:12–19
19. Raychaudhury C, Pal D (2012) Use of vertex index in structure-activity analysis and design of molecules. Curr Comput Aided Drug Des 8:128–134
20. Raychaudhury C, Klopman G (1990) New vertex indices and their applications in evaluating antileukemic activity of 9-anilinoacridines and the activity of 2′, 3′-dideoxy-nuclosides against HIV. Bull Soc Chim Belg 99:255–264
21. Raychaudhury C, Dey I, Bag P, Biswas G, Das B, Roy P, Banerjee A(1993) Use of a rule based graph-theoretical system in evaluating the activity of a class of nucleoside analogues against human immunodeficiency virus. Arzneim Forsch Drug Res 43:1122–1125
22. Prathipati P, Ma NL, Keller TH (2008) Global bayesian models for the prioritization of antitubercular agents. J Chem Inf Model 48:2362–2370
23. GTB data set. http://pallab.cds.iisc.ac.in/gtb_data.mol
24. Kandel DD, Raychaudhury C, Pal D (2014) Two new atom centered fragment descriptors and scoring function enhance classification of antibacterial activity. J Mol Model 20:2164
25. Raychaudhury C, Kandel DD, Pal D (2014) Role of vertex index in substructure identification and activity prediction: a study on antitubercular activity of a series of acid alkyl ester derivatives. Croat Chem Acta 87:39–47
26. Moss G (1999) Extension and revision of the von Baeyer system for naming polycyclic compounds (including bicyclic compounds). Pure Appl Chem 71:513–529
27. Weininger D, Weininger A, Weininger JL (1989) SMILES. 2. Algorithm for generation of unique SMILES notation. J Chem Inf Comput Sci 29:97–101