Protein structure informatics and drug discovery

Elective assignment

Task 1: Implement a database of chemical compounds

# 1.Introduction

Developing a new drug from original concept to it passing FDA approval and its launch onto the market is a lengthy and costly process, taking up to 12-15 years and costing in excess of one billion dollars. The use of computers and computational methods permeates all aspects of drug discovery with the aim of delivering new drug candidates more quickly and at a lower cost.Some of the major roles of computation in drug discovery are pre- screening compound libraries, pharmacophore library screening, molecular docking screening and quantitative structure-activity relationship (QSAR).

The early stages of drug discovery aims to reduce the large number of candidate chemicals that are rejected later in the process due to pharmacokinetic-related failures. In silica screening is being used to find” drug-like” hits from the vast compound libraries available. These filers are based on physiochemical properties, one commonly used simple filter is Lipinski’s Rule of Five (Lipinski 1997) in which molecular weight, Log P and hydrogen bond and acceptor and donor sites have to be below a certain limit. There are a huge range of filters that have been developed with varying levels of complexity (Opera 2001). An example is the prediction of ADMET properties such as, lipophilicity, solubility and absorption , these ADMET properties are now modelled with predictive chemistry using machine learning.(Tao et al., 2015)

In this assignment we focus on two aspects of the Computer aided drug design. In the first part we developed a data base to be used to screen libraries of compounds on various filters, each filter being a combination of a set of physiochemical properties. In the second part we developed an algorithm for searching the conformation space a molecule could take. These conformers could form the input for pharmacophore search.

# 2.Methods

The specification for the project is to implement a relational database for the storage and retrieval of a library of chemical compounds with accompanying derived attributes that are relevant to their use in drug discovery. Functionality required of the database was

* To rank the entries in the database on a weighted/summed combination of the attributes to screen the database entries.
* To filter the compounds based on the criteria specified by Lipinski’s rules, lead likeness and bioavailability.

The main considerations throughout the development of this project were

* Scalability in terms of number of molecules to be included,
* Ease of adaption so adding new properties or filters.
* Ease of interaction by the end user.

This task was divided into parts

* Design of the database and creating the tables in MySQL.
* Calculate the required attributes from the SDF records and loading into MySQL tables
* Develop the user interface to aid the user interrogate the data base

## 2.1 Database design

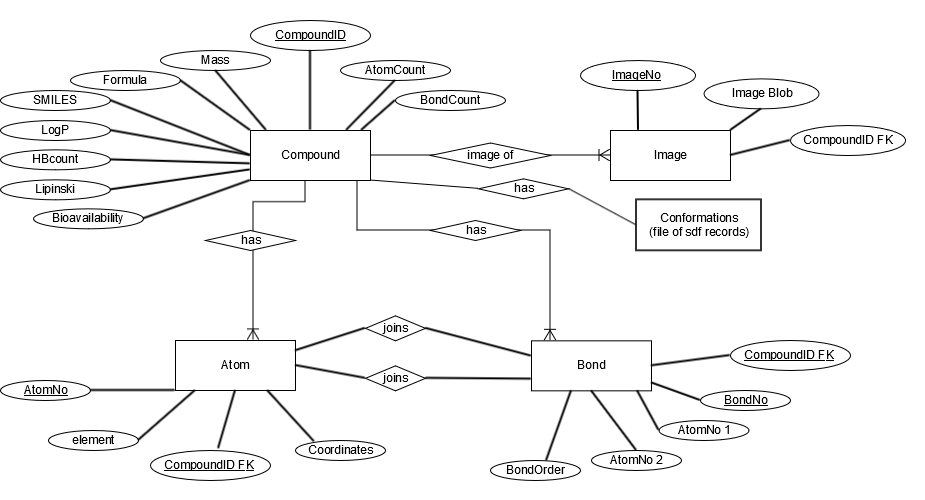
A list of properties required was collated from the various filters that were required for screening. This is a subset of what is available or what may be included in future development.

* Lipinski : All the following conditions met, Mass < 500, LogP < 5, Hydrogen bond donors < 5, Hydrogen bond acceptors <10.
* Oral Bioavailability , 6 out the following 7 conditions met.:

Mass < 500 , log P < 5. hbAcceptor <=10, Hbdonor <=5, rotatable bonds < 10 , Polar surface Area < 200 and aromatic ring count < 5

* Lead-Likeness, all the following conditions met: Mass < 300, log P <3,Hydrogen bond Acceptor < 3, Hydrogen bond Donor < 3 and Rotatable bonds less than 3.

These properties are calculated and stored in the database, as calculating them at the time of query would make the database slow to use especially when a large library of compounds is used.The database currently consists of three tables each representing a different entity, compound, atom and bond, See Figure 2.1

 Figure 2.1 Entity relationship diagram for the ChemDat database.

These properties are all attributes of the compound and as such are stored in the compound table. By using the one table for all

Atom and Bond tables are also produced for use in future development. Including the possibility of generating a subset of viable conformers for a compound.

The Image and Conformation tables have not been implemented.

## 2.2 Database maker.

This was extended from code developed by David Leader, SDBMaker. The ma

Two main adaptions were made:-

Parser

Firstly the method to parse the file was change to be more adaptable and allow for the fact the SDF file format is not a rigid structure but uses keywords to identify the information contained.

The first line containing integers is assumed to contain the “count line” that specifies the number of atoms and bonds to follow.

Keyword specifiers indicate what information is contained in the following line. In this case we were searching for > <CDid> to get the unique compound ID that would be the primary key in our compound table.

$$$$ is always used to indicate the end of the record and was used to start the new loop. For the next molecule until the end of file. The parser is not restricted to files of 30 records.

Property Class

A new Property class was developed to contain all the variables and methods to calculate the derived attribute. Chemaxon (http://www.chemaxon.com) was used via various plugins to calculate all chemical properties required. All the code pertaining to Chemaxon was contained in the property class.

This was done by using a separate method to calculate each attribute. This modularisation makes the code very easy to adapt by adding or removing calls to required properties. At a later date this could be adapted allow the end user to choose what properties they required. For databases containing thousands of molecules this could reduce unnecessary calculations.

The filters were set as Boolean tests of the appropriate properties below the specified range. A weighted scale could have been used in which the value is calculated as the number of conditions met / total number of conditions tested. The Boolean test was chosen for simplicity.

This data was then added to the compound records on saving the output file, property. See Appendix 1 . MySQL Script (Load)

## 2.3. Database Access.

Designed a Graphical User Interface (GUI) to search the database.

The main requirements were:-

* To view a subset of molecules by applying the various filters.
* To choose to see only those properties pertaining to the filter.
* To rank these molecules on one the various properties.

A standard View Model Controller design was used. The controller was in the action performed. The view is in the new routine displayTable and the model is the processSubmit which interacts with the database. This could be improved by keeping the code in separate classes.

Some of David Leader’s code SDBAccess was used but new methods were created to generate the queries and display the results. This takes the information about which radio buttons were chosen and uses this to construct the relevant MySQL queries for the Structures database.

For simplicity and ease of use the user can choose which filter and which set of properties to view via exclusive radio buttons. These are displayed in a permanent window at the top of the screen.

I simplified the way a query was constructed using a dictionary to convert the chosen radio button into the relevant text for the query. This means adding new filters or views is relatively trivial.

A JTable was used to display the results. This is completely adaptable with one method being able to display any set of results returned from the query. It is displayed in a scrollable window. Each time a filter is applied a new window opens enabling the user to make comparisons between results.

The records can be ordered on any column, in ascending or descending order.

## 2.4. Future Improvements

SDB Maker

1. Writing records straight to MySQL data base using INSERT command.
2. Adding a larger range of properties if and when required.
3. Write image compoundID and image BLOB to image table. The table can easily be populated by inserting it into the table within JAVA.

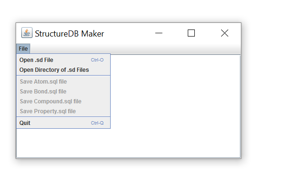
SDB Access

1. Get images output when core chosen.
2. Reinstate DL’s bond search and develop other structural queries. Using a separate panel for choices.
3. Chose a subset of molecules and output to a SDF file (or other suitable format). This could be generated through the SMILES string via ChemAxon ‘s Molconverter.

# 3.Results

## 3.1. SDBmaker

This is used to generate the data for the ChemDat data base from the input sd files.



**Open sd file** – Opens a singlefile , multiple recoreds can be parsed.

**Open Directory of sd files** :- Opens a directory of sd files and parses them all. (Not tested)

**Save … files** allows yo to save the data for the individual tables. Save Property is required for the ChemDat compound table

## 3.2.ChemDat

This is the user interface for the database. The screen shots below of the database interface show the results after the various filters have been selected. The number of compounds returned along with the number of compounds in the library is displayed in the header for the window. The windows are scrollable so with a large data base you can scroll up and down the compounds. Any one of the properties can be selected to order or rank the compounds either ascending or descending, one or two clicks in the column header.

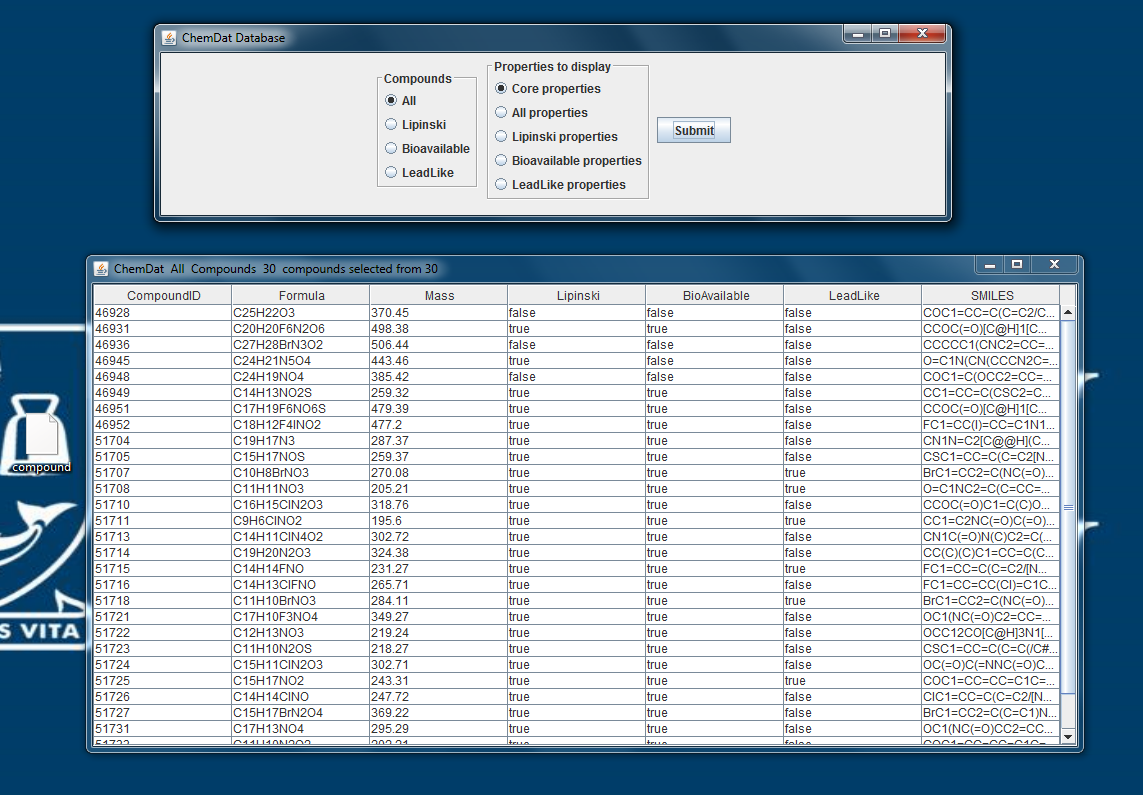


Figure 3.2.1. Screenshot with the Lipinski filter and properties set.

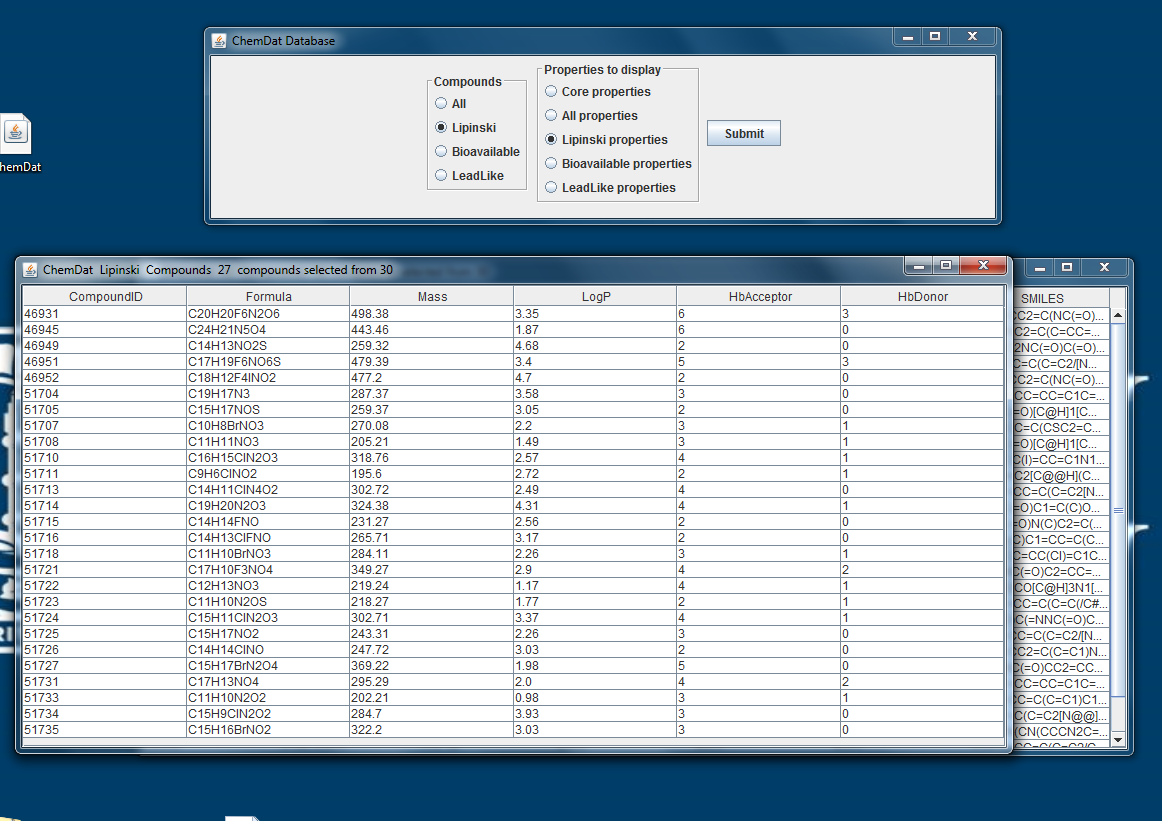


Figure 3.2.2 Screen shot with the bioavailability filter and properties set.

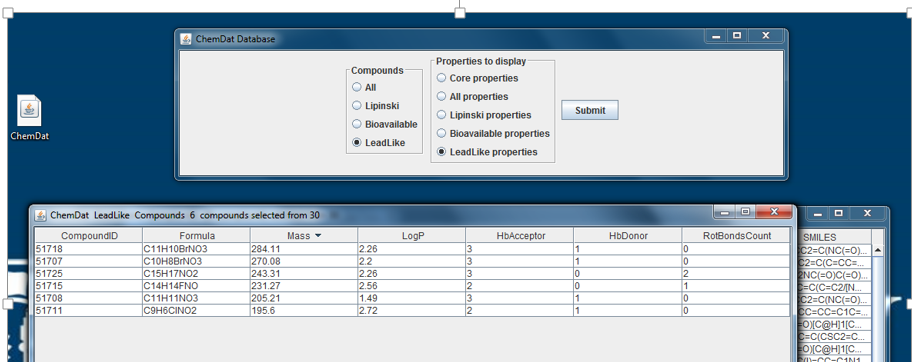


Figure 3.2.3 Screen shot with the Lead like filter set.

# Task 2: Sample Conformational Space

Given a molecule with n rotation bonds sample the possible conformations .

The molecule AW\_00185 has 4 rotational bonds, that is a single bond not in a ring or a bond to terminal atom.( Partial double bonds, as in amide bonds, need to be accounted for). Figure 2 shows how restricted the molecule is and many torsion angles of the bonds will produce invalid conformations where steric clashes occur.

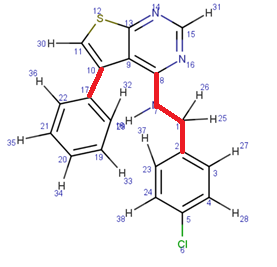
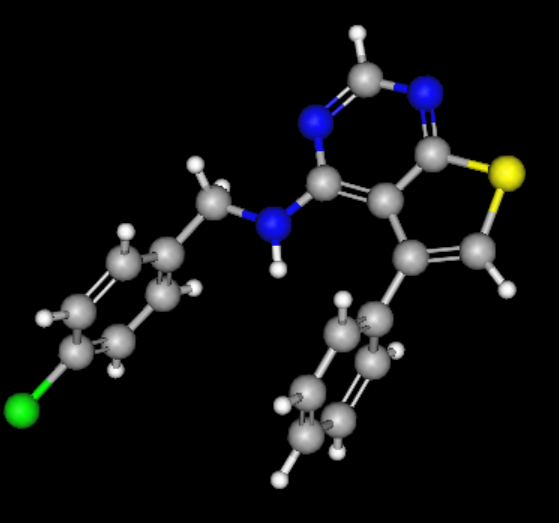


Figure 1a and b. Molecule AW\_00185. Figure b highlighting the rotational bonds.

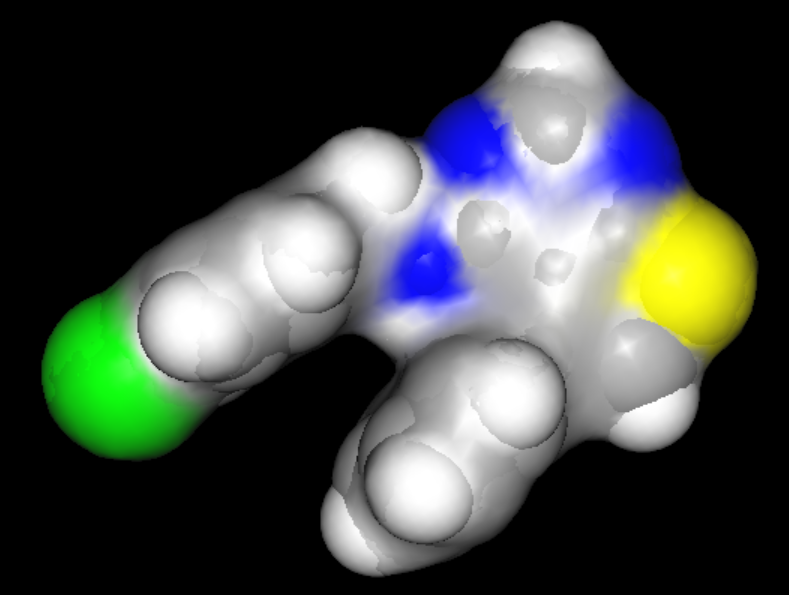


Figure 2. AW\_00185 Vander Waals surface representation.

The first step in the algorithm is to search the molecule to identify the rotational bonds and the groups of atoms that will not move relative to each other. This is done by a recursive call starting from an outside atom and searching all bonds attached to that atom. This will also collect the atoms into groups. Figure3.

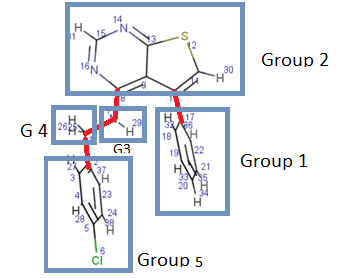


Figure 3 . Molecule divided into atom groups by the rotational bonds

The next step is to generate a multimodal tree. Figure 4. Again this is done recursively by starting with group1 (the root node), generating it’s children (new nodes, copies of parent) by applying rotational transformation matrix to atoms in all the other groups around the vector of bond 1. The number of children will depend on the step, ø, chosen for rotation, Nc = (360/ø)/sym where sym is the symmetry of the group.

The total number of possible conformations is (360/ø)n ,where n is the number of rotatable bonds. With 4 rotatable and ø = 1 there are over 1.6 x 109 possible conformations, for ø = 5 that number is reduced to 2.7 x 107. This will have a major impact on how long the program takes to run.

Each child should be checked to see if it is valid, check for clashes between child and it’s parent, but not atoms in groups that can still move. If it is valid add the child’s atoms to the parent and then may a recursive call to repeat the process for all the children.

Once the last group has been reached store all the nodes generated.The algorithm to perform this search is shown as a flow chart in Figure 5

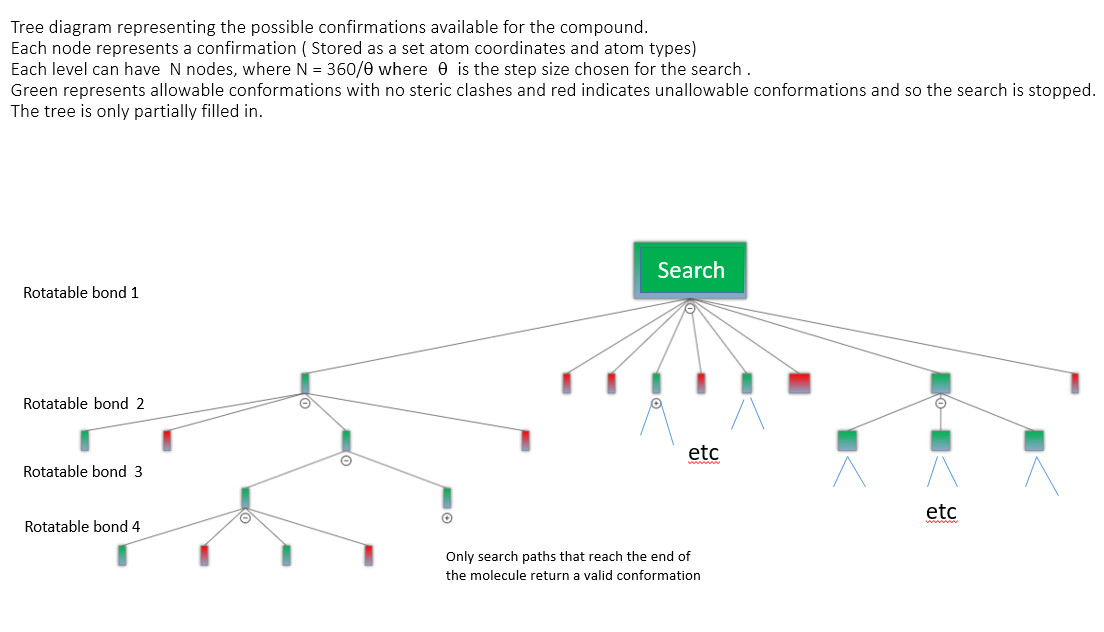
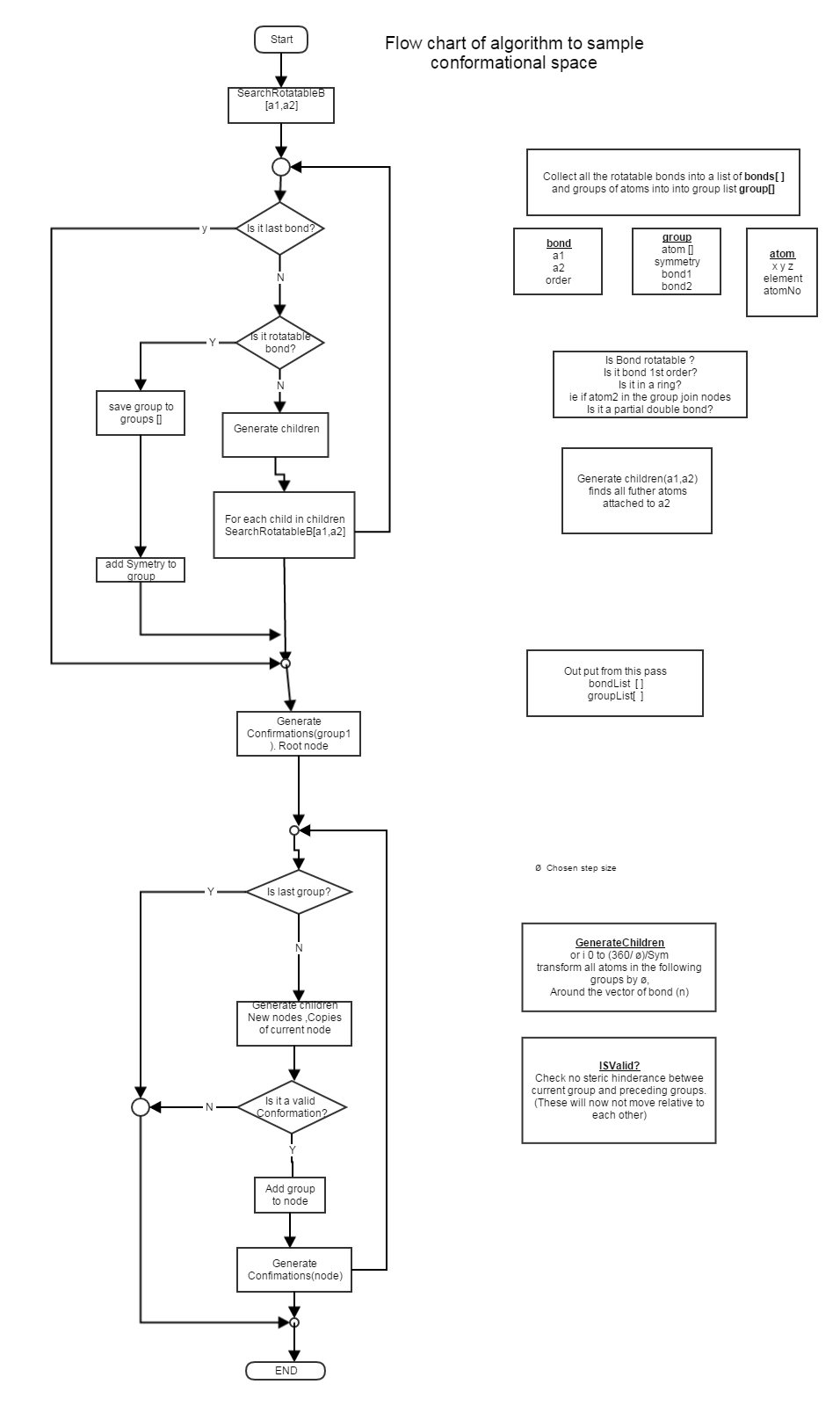


Figure 4. Generating a multi-nodal tree.



# 4. Conclusion.

In this assignment we implemented a database that could be used for screening chemical libraries for required physiochemical properties. Although this database is very simple it demonstrates how easily a series of filters can be used to remove compounds that are highly unlikely to become leads from the search.

The increasing power and size of computers means the areas where computational methods can be applied is ever expanding. More complex methods are continually being developed with increased success , meaning knowledge based drug design is now a reality More complex algorithms are now used as filters including machine learning to predict the ADMET properties that are so crucial in drug design and will further cut the attrition rate further down the pipeline. We also designed an algorithm to generate possible conformers of a compound which could then be used in pharmacophore searching

Computers are now considered vital to streamlining the drug discovery process and the future development of more complex and more accurate methods will only enhance their importance.

# 5.References

Tao, L., Zhang, P., Qin, C., Chen, S.Y., Zhang, C., Chen, Z., Zhu, F., Yang, S.Y., Wei, Y.Q., and Chen, Y.Z. (2015). Recent progresses in the exploration of machine learning methods as in-silico ADME prediction tools. Adv. Drug Deliv. Rev. *86*, 83–100.

C.A. Lipinski, F. Lombardo, B.W. Dominy, P.J Feeney, Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings,

Adv. Drug Deliv. Rev. 23 (1997) 3

Oprea, T.I. et al.; Is There a Difference between Leads and Drugs? A Historical Perspective; J. Chem. Inf. Comput. Sci., 2001, 41, 1308-1315.

## APPENDIX 1

Using SDBMaker and ChemDat

**SDBMaker.**

This is only run once if you are loading a new library of compounds into the database

Run the program. A window appears with a file menu.

You can choose to load 1 file or a directory of files. These should be in SD format.

Once the parser has run you can choose to save data to the relevant files.

Atom for the atom table

Bond for the bond table.

Choose property for the compound table

Once the files have been saved the script in appendix 2 can be used to create the tables and load the files. Ensure you use the correct path.

**ChemDat**

Run the program, a window will appear with the choice of filters and which properties you want to see. Choose one from each and press submit, (defaults are set to All compounds showing the core properties).

A new window will appear displaying the results returned from a query. These can be ordered or ranked by any of the columns by clicking in the column title, once for ascending twice for descending.

## APPENDIX 2

# MySQL creation script saved a tableCreation.sql

#The data from SDBMaker needs to be saved in 3 files on your desktopcompound

# compound, atom and bond

Create Table compound (

CompoundID varchar(30) Not null primary key,

Formula varchar(30) not null ,

Mass double (6,2) not null,

Lipinski varchar (10) not null,

BioAvailab varchar (10) not null,

LeadLike varchar (10) not null,

HbAcceptor int not null,

HbDonor int not null,

LogP double (6,2) not null,

PolarSA double (6,2) not null,

RotBondsCount int not null,

RingCount int not null,

AtomCount int not null,

BondCount int not null,

SMILES varchar (250));

load data local infile 'C:/Users/Fran/Desktop/compound' into table compound;