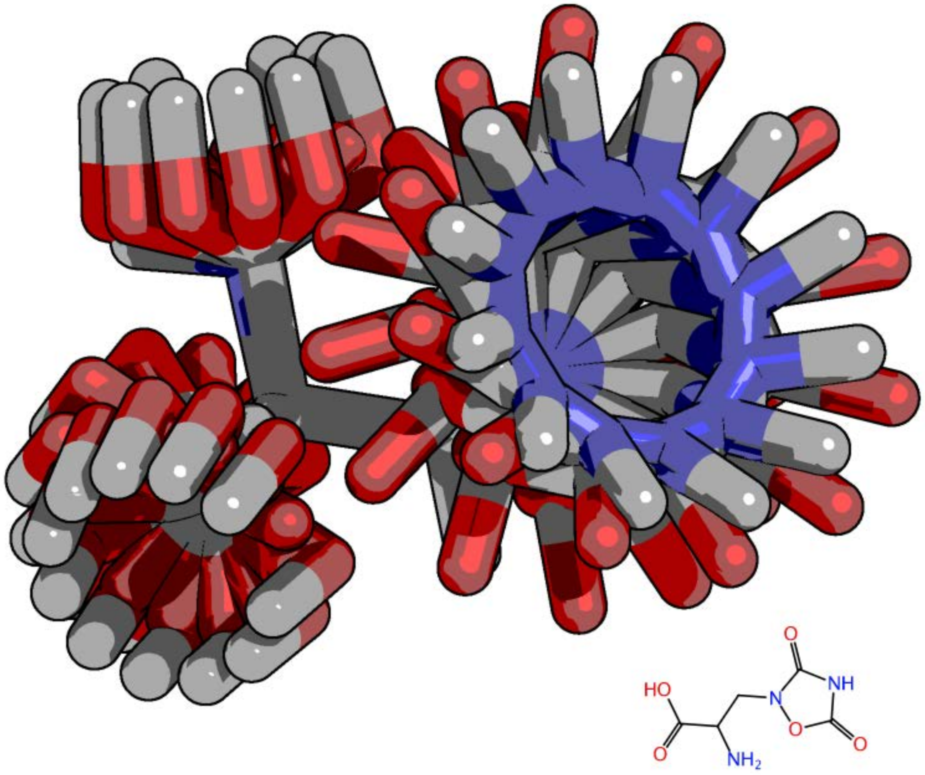
Conformer Generation using RDKit

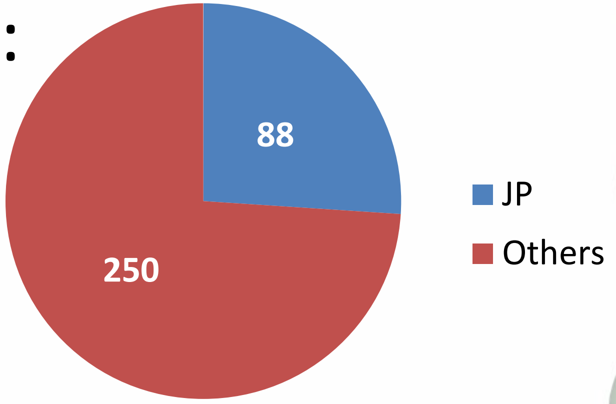


Jean-Paul Ebejer

1st RDKit User General Meeting – London, 2012

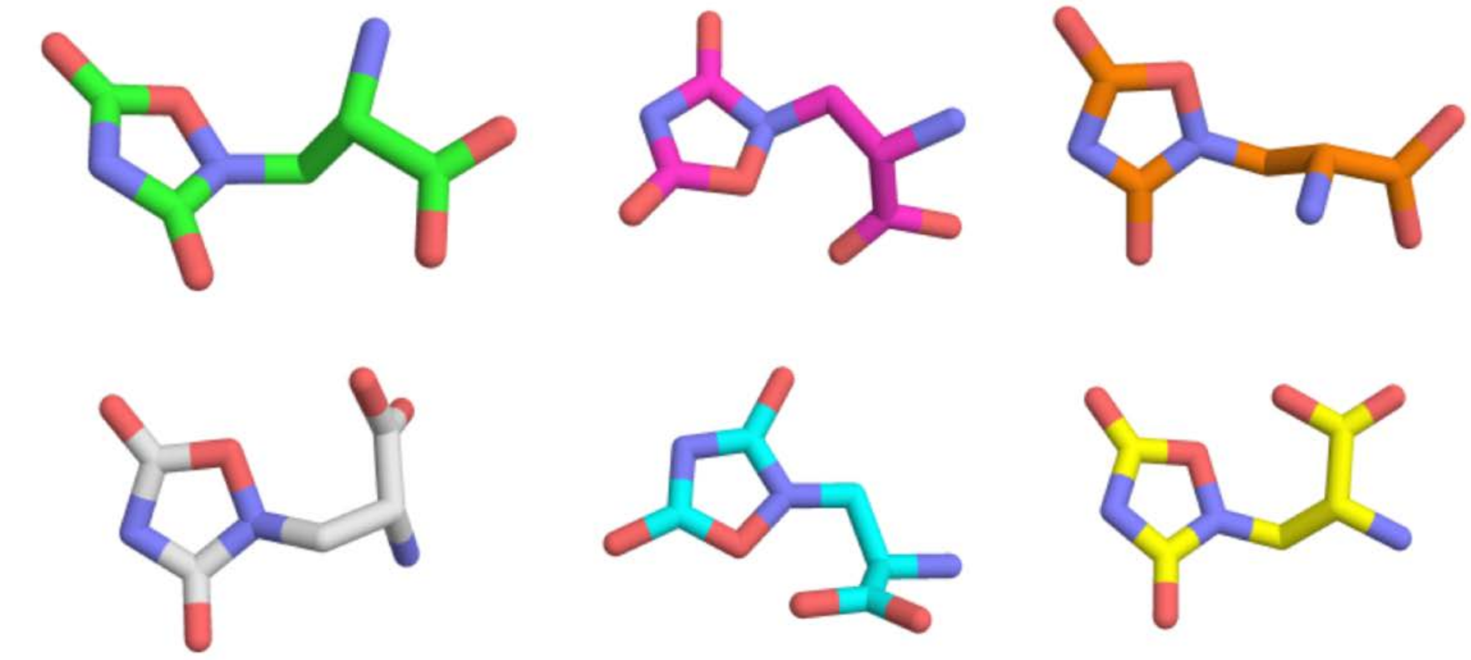
Who am I ?

- 3rd year D.Phil. student at the University of Oxford
  - Marie Curie EU research project on Malaria between industry (InhibOx Limited) and academia (Oxford Protein Informatics Group)
- Background in computer science and bioinformatics
- **RDKit** mailing list subscriber since 8th October 2010 (some bug reports filed, one patch)
- Topics started in mailing list:



What is a conformer ?

- A molecule can take on many different shapes
- Conformational space may be very large, and is a function of number of rotatable bonds

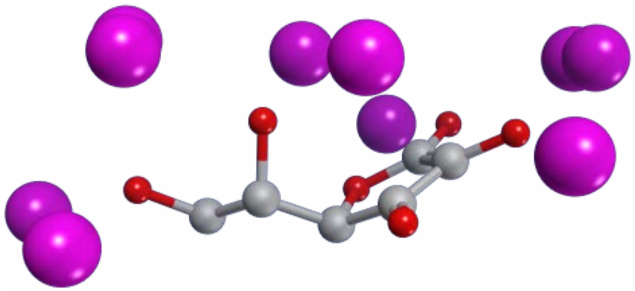


Chemical Component Identifier- QUS

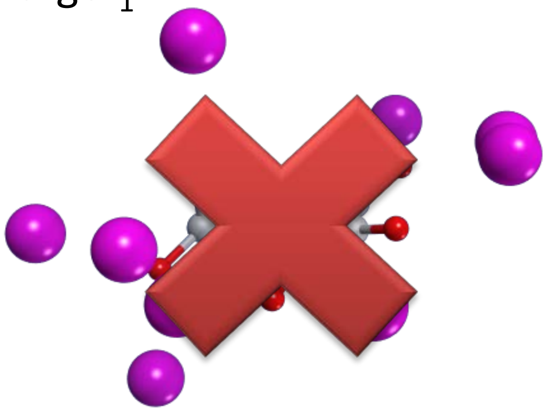
Why is this important ? (i)

- In 3D ligand-based virtual screening

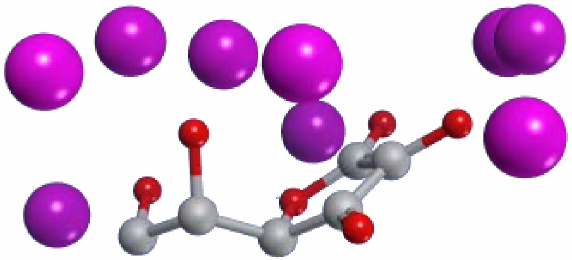
Query Molecule



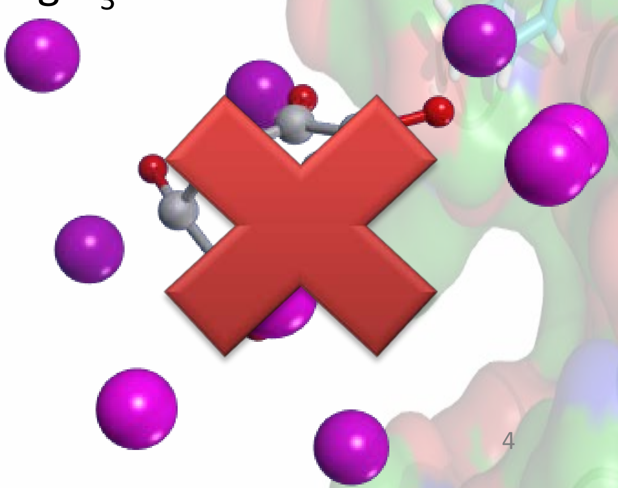
Target$_1$



Target$_2$

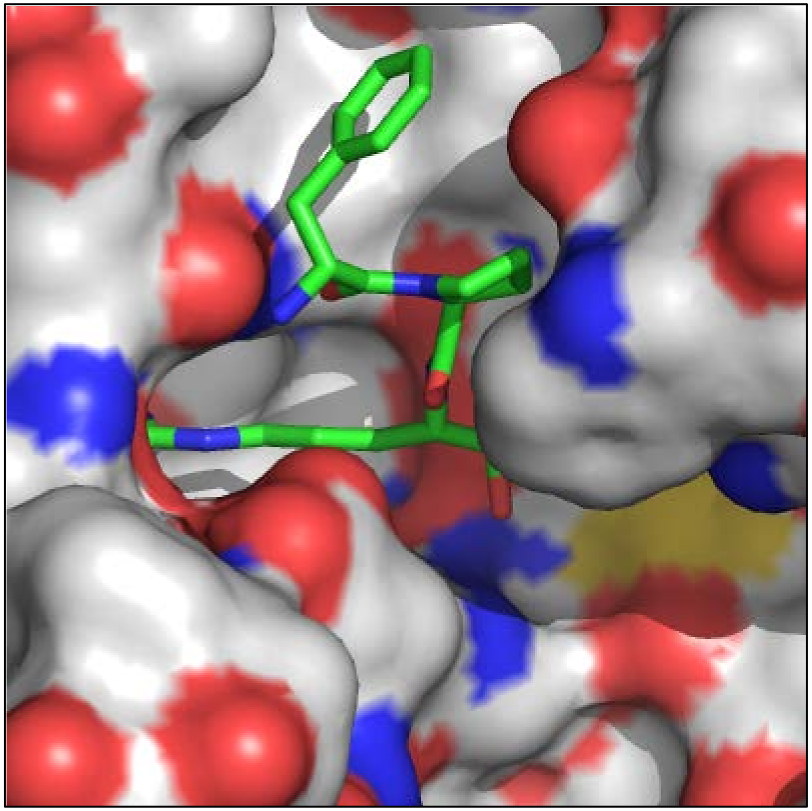


Target$_3$

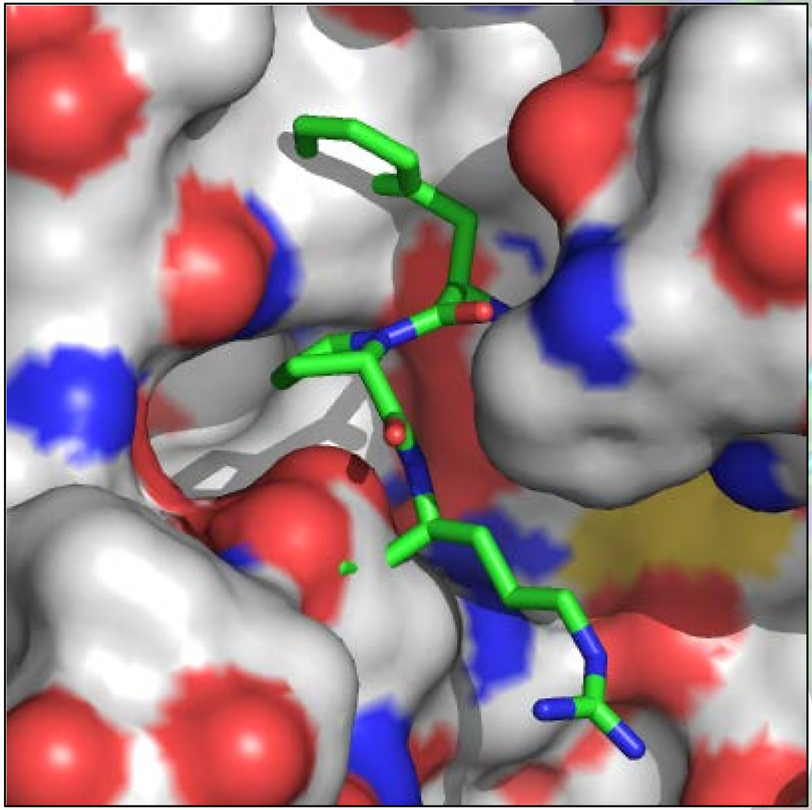


Why is this important ? (ii)

- In structure-based virtual screening (internally in docking programs)



Thrombin (1PPB), with binding inhibitor 0G6

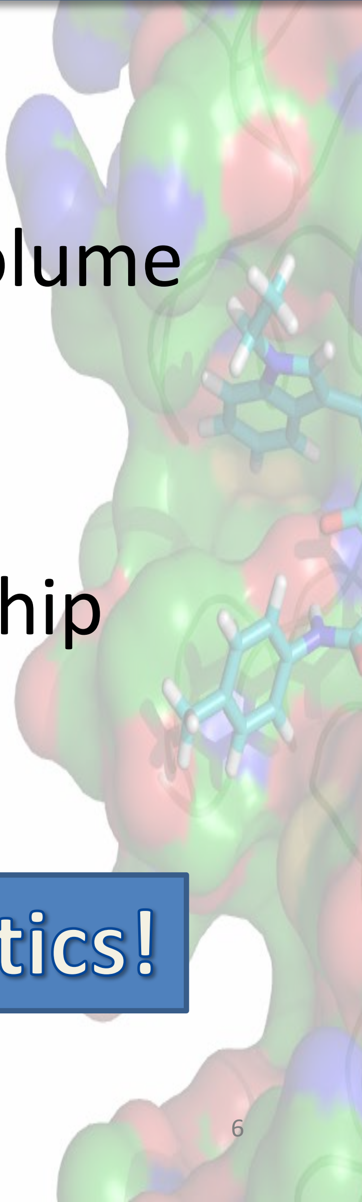


Same receptor, with different conformer of same inhibitor (0G6)

What is conformer generation used for?

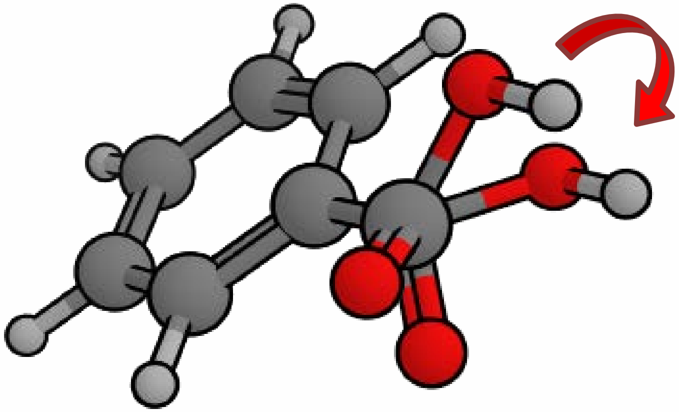
- Virtual screening
- Shape-based similarity searches (*e.g.* volume overlap)
- Pharmacophore modelling
- Quantitative structure-activity relationship (3D QSAR)

Ubiquitous process in cheminformatics!



In general, how does it work ? (i)

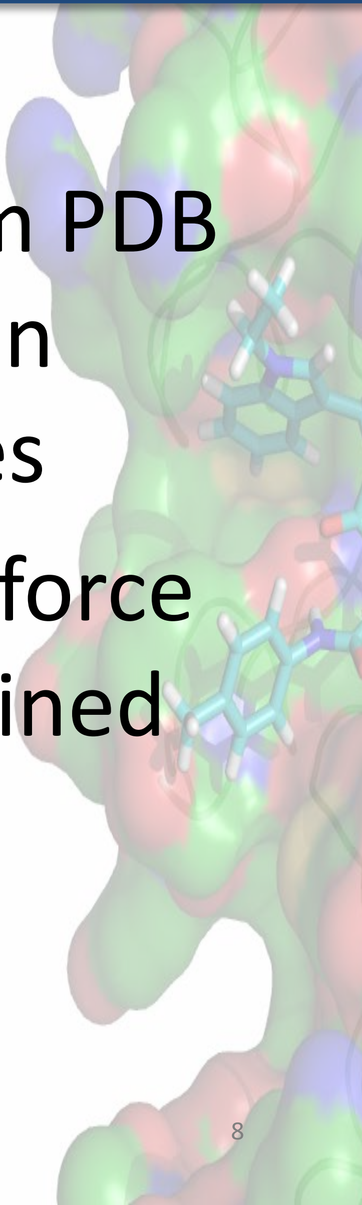
- **Systematic approach**: Change torsion angles of all rotatable bonds by a small amount
  - But for large molecules this is infeasible (generates too many states)



- **Stochastic approach**: Use of random algorithms such as distance geometry, Monte Carlo simulation and genetic algorithms to permute torsion angles

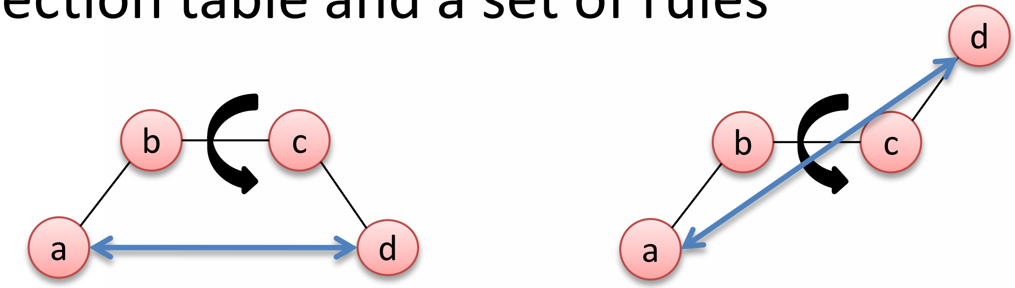
In general, how does it work ? (ii)

- Use of statistically derived data from PDB and CSD to determine most common angles between different atom types
- Typically clean up structures with a force field to avoid steric clashes and strained structures
  - Usually time consuming step



How does RDKit generate conformers?

- Uses distance geometry
- Main ideas:
  - Molecule’s distance bounds matrix is calculated based on the connection table and a set of rules



min. distance

max. distance

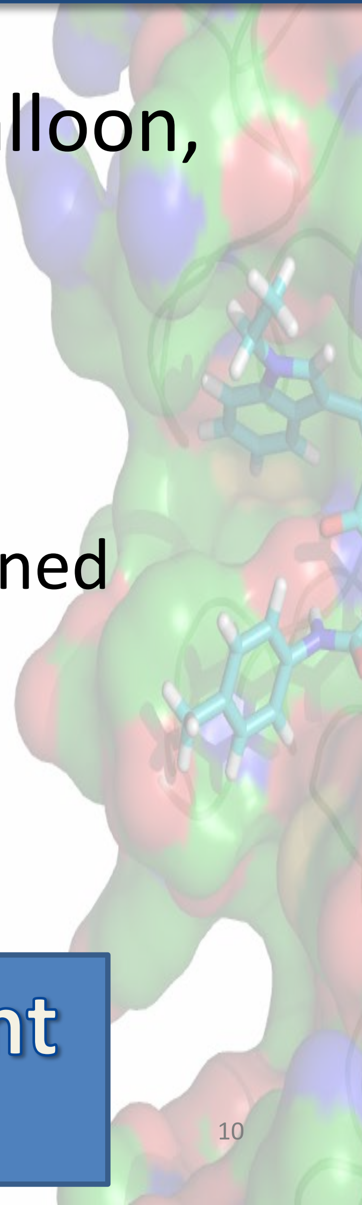
  - Generate a random distance matrix which satisfies these bounds
  - 3D Coordinates produced from these distances (embedding)
  - Different random distance matrices give rise to different conformers

For more details: Greg Landrum: ***RDKit Manual: Getting Started with the RDKit in Python***.Section 1.3.5.

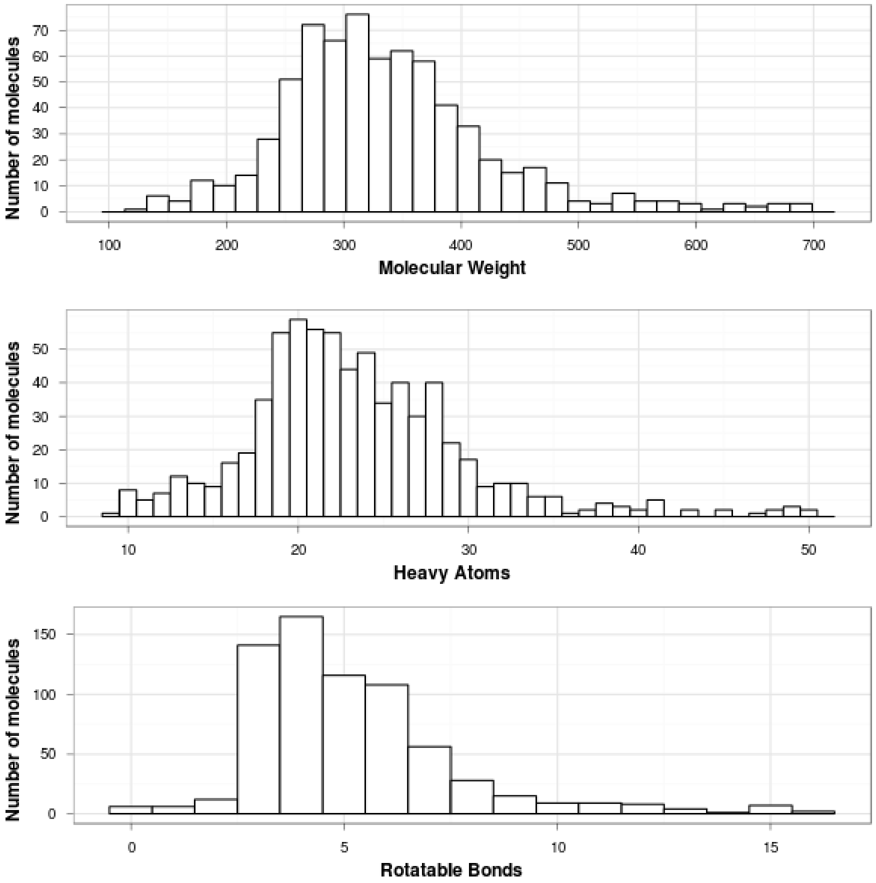
What did we do?

- Reviewed four freely available tools: Balloon, Confab, Frog2 and RDKit together with commercially available MOE
- We are interested in three measures:
  - Ability to generate experimentally determined structure
  - Diversity of conformers generated
  - Speed of conformer generation

Each of these aspects has important repercussions in drug discovery!



We need a molecule test set



- Test set of 708 molecules, made from a previous validation study of another commercially available product (OMEGA) and small molecules from the Astex Diverse Set
- These molecules are high resolution structures (< 2Å) taken from the PDB and CSD
- Drug-like distributions for molecular weight, heavy atoms and rotatable bonds¥

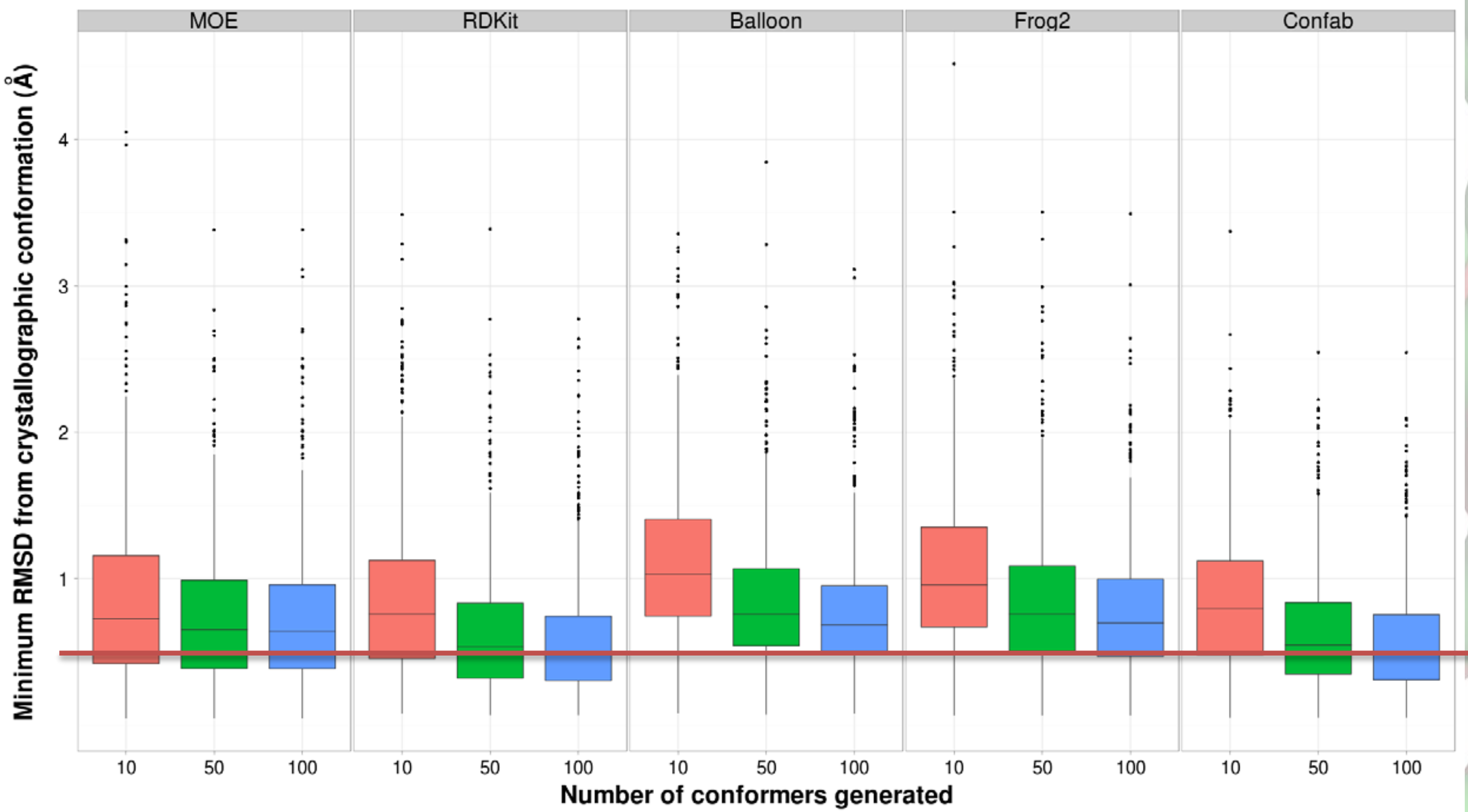
¥ Oprea, T. I. ***Property distribution of drug-related chemical databases***. Journal of Computer-Aided Molecular Design 2000, 14, 251–264, 10.1023/A:1008130001697.

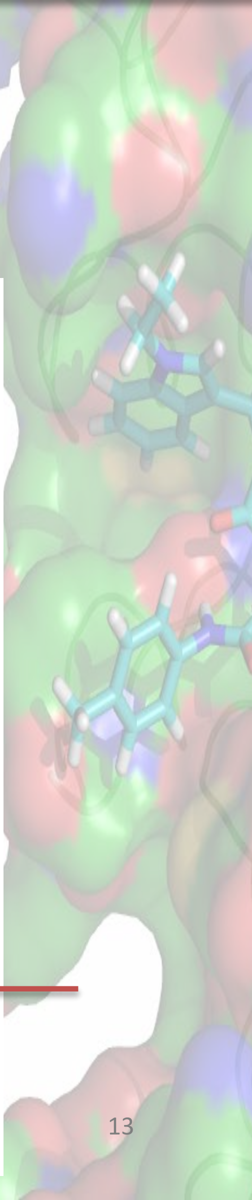
Methods

- Generated SMILES with stereochemistry for 708 molecules to be used as input
  - Confab does not accept SMILES input, so generated initial structure with openbabel (–gen3d option)
  - We do not want initial 3D geometry of reference structure to bias any of the conformer generation tools
- Confab does not generate a specific number of conformers
  - Generates >10,000 conformers for 11 molecules; median number of conformers is 92.5
  - Picked 10, 50 and 100 random conformers where the conformational models exceeded these numbers

Ability to re-generate the crystallographic structure (i)

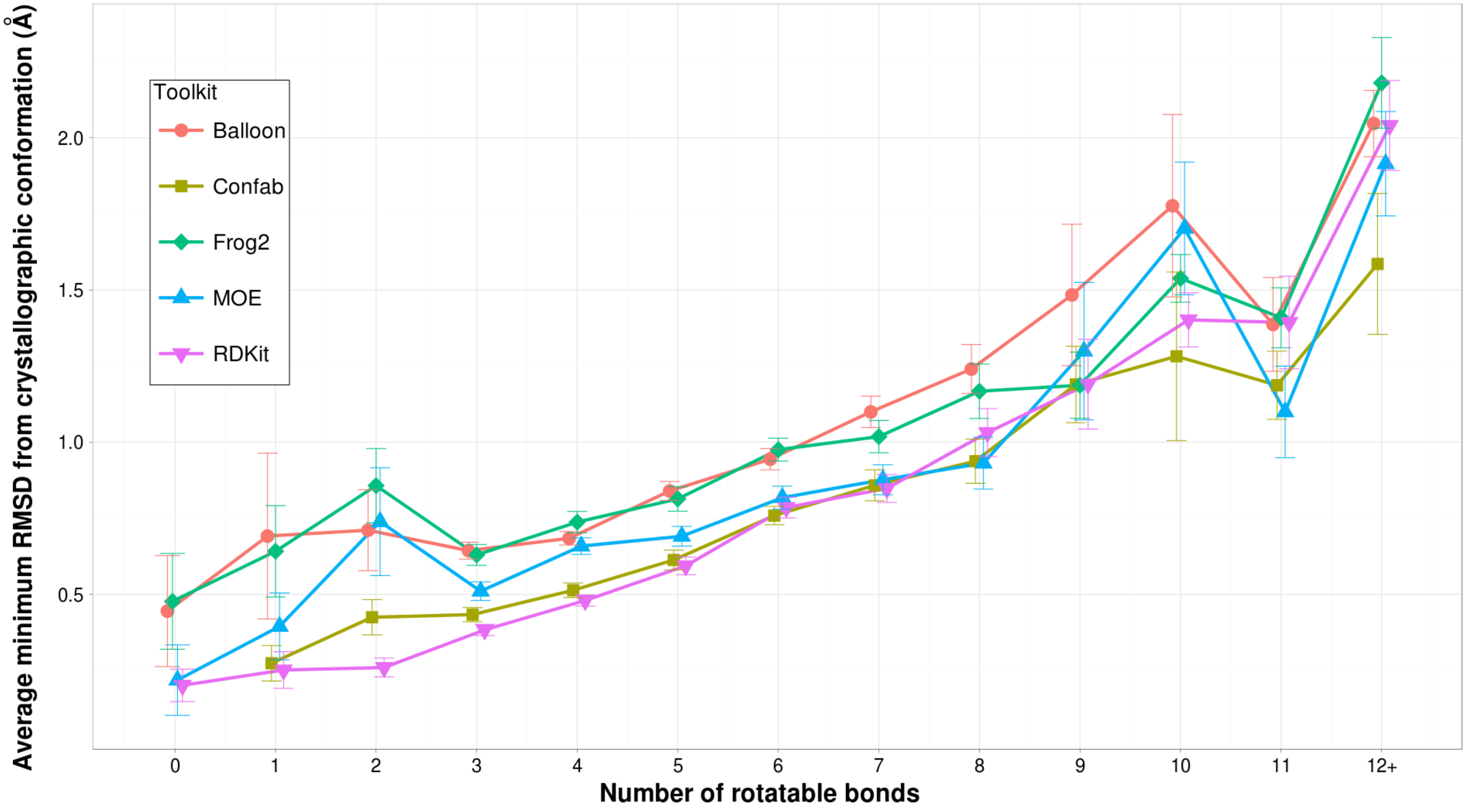
- Generated 10, 50 and 100 conformers for every tool
- For every molecule in dataset found minimum RMSD to experimental structure



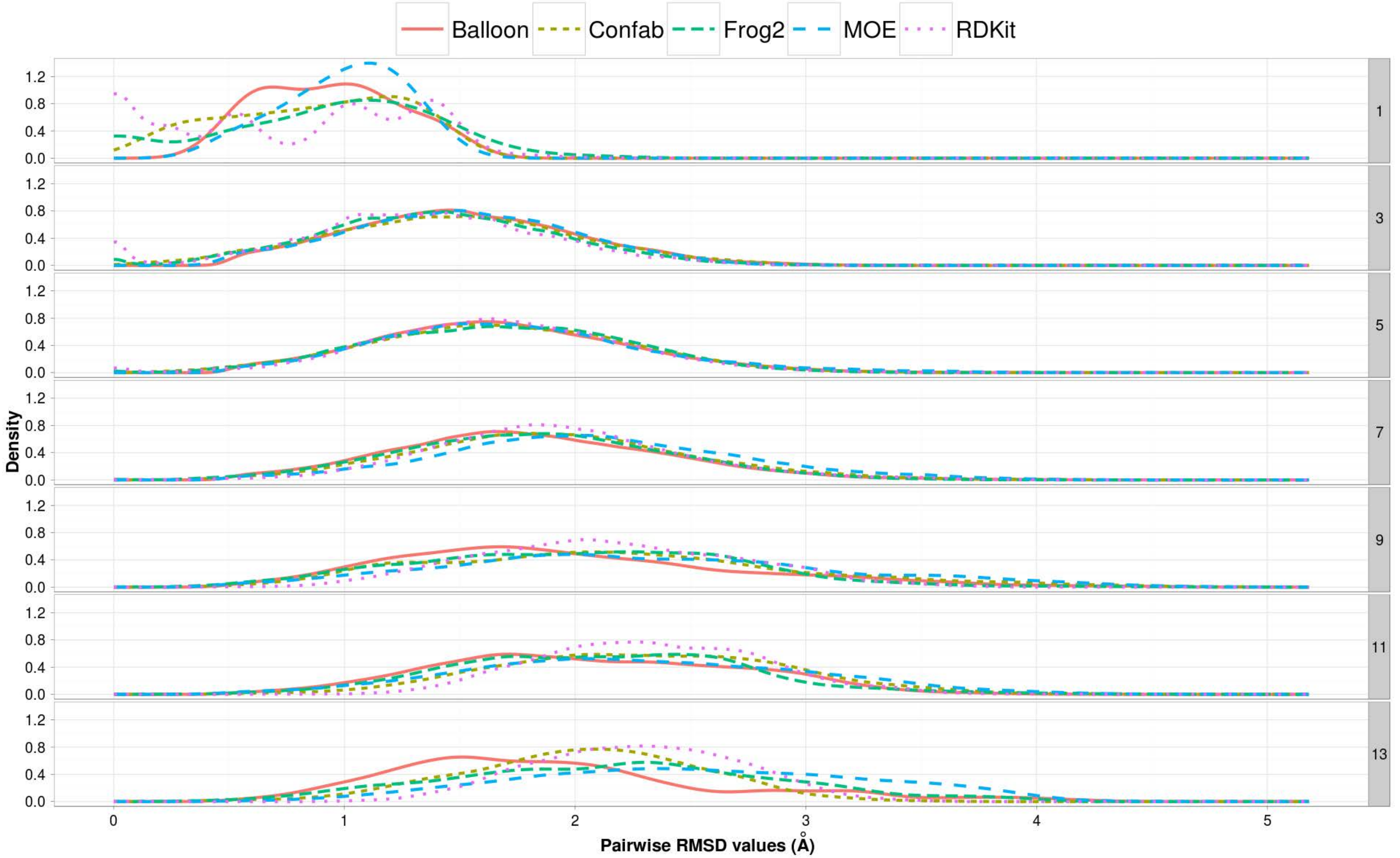


Ability to re-generate the crystallographic structure (ii)

- As expected, it is harder to reproduce the experimental conformation of the more flexible molecules

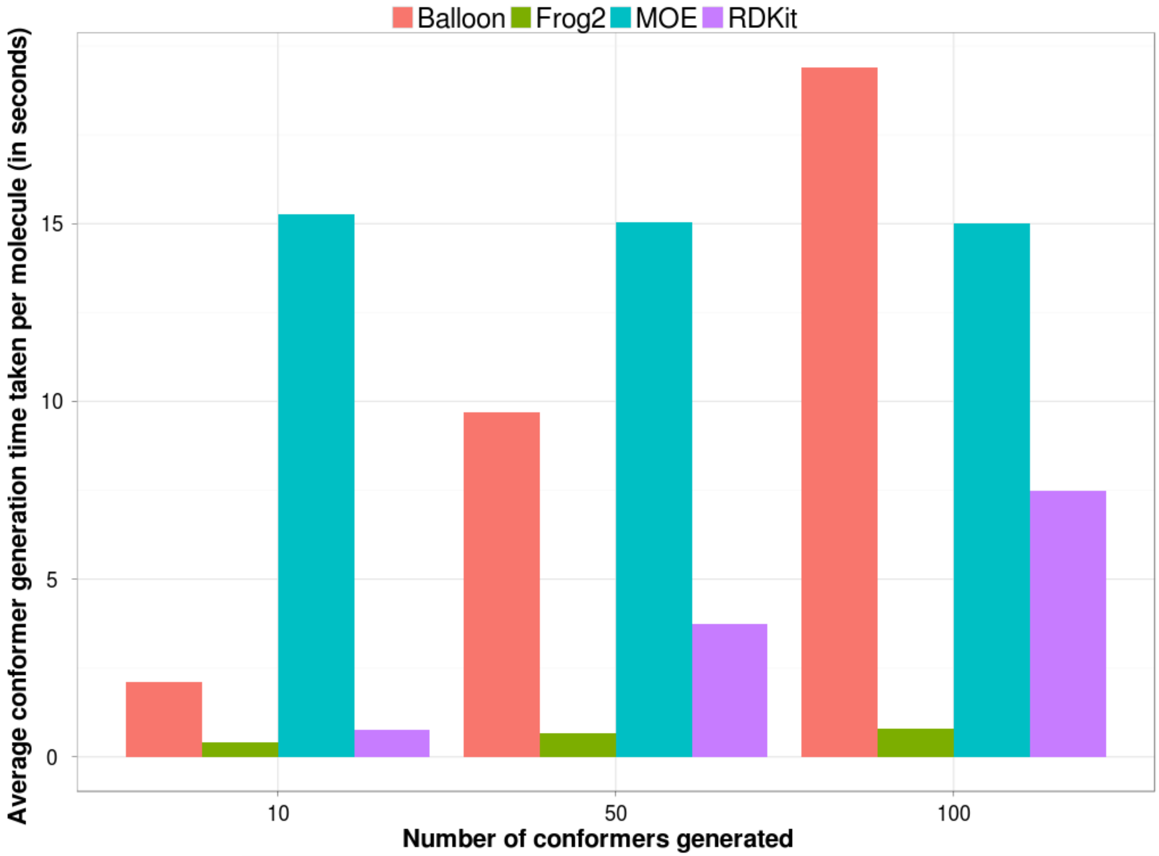


Diversity of conformers generated

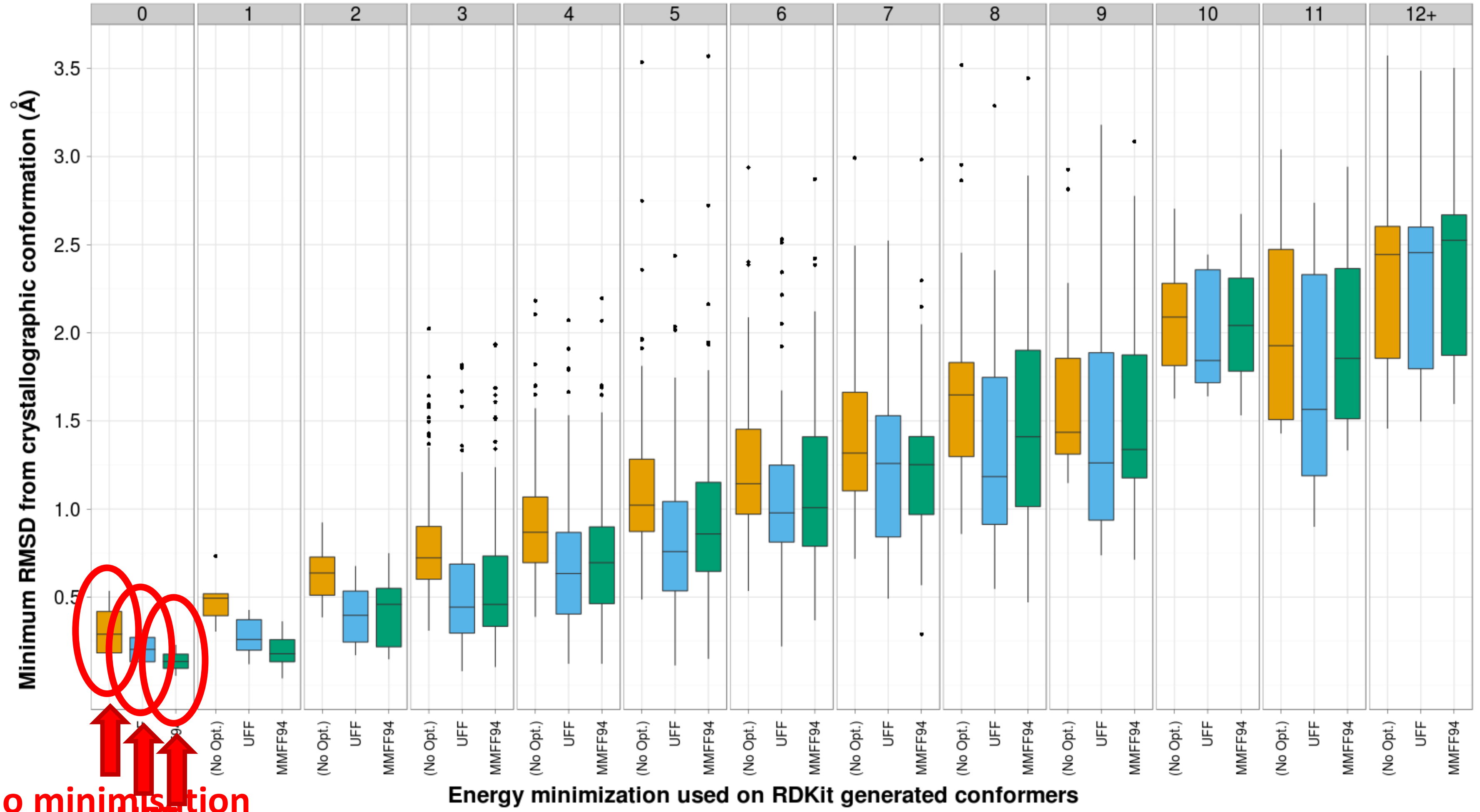


The need for speed

- Frog2 fastest tool by an order of magnitude, followed by RDKit when generating up to 100 conformers
- MOE faster when generating 300 conformers (not shown)



In RDKit, energy minimisation is needed to improve results!

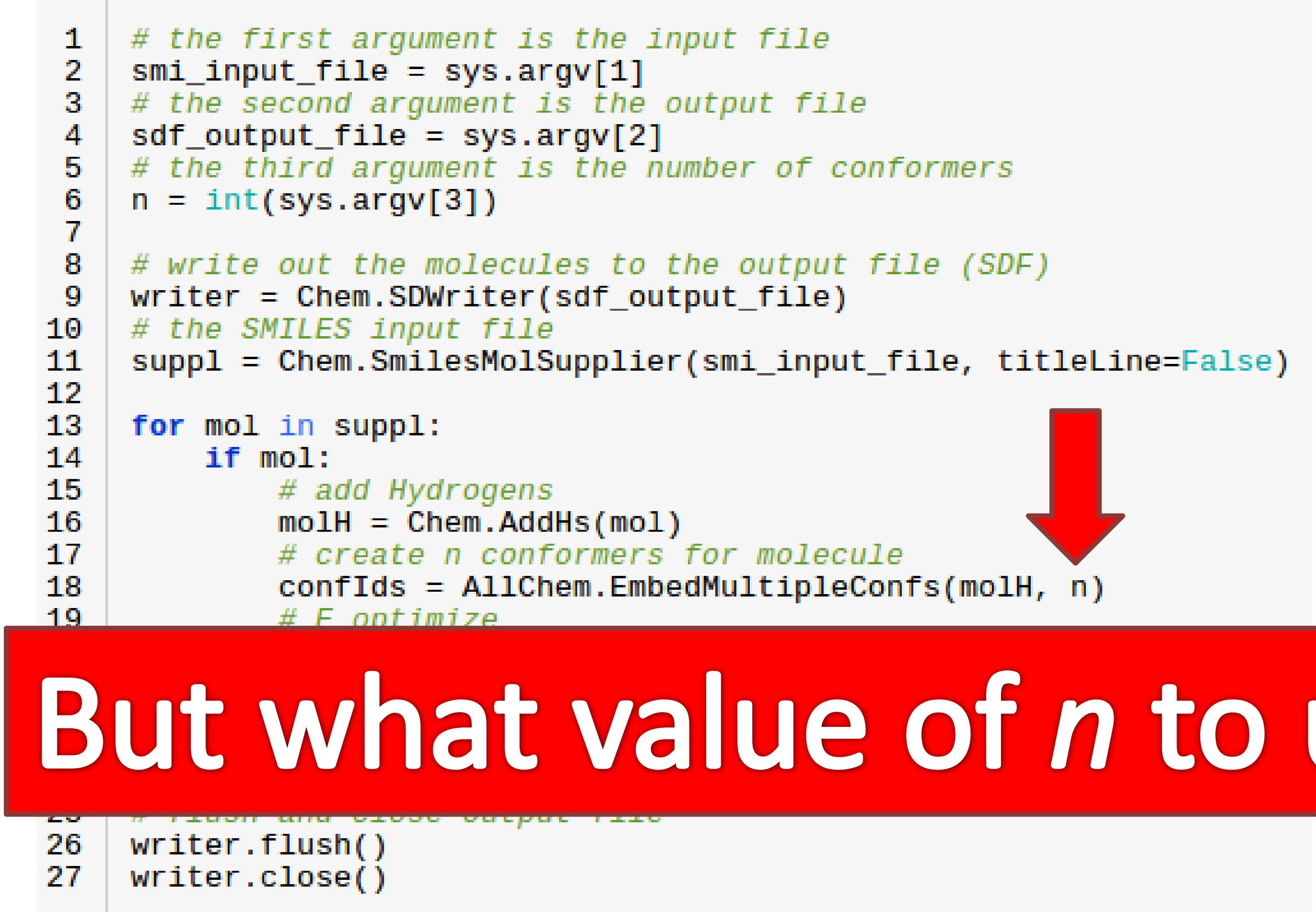


No minimisation

UFF

MMFF94

Conformer generation in 14 lines



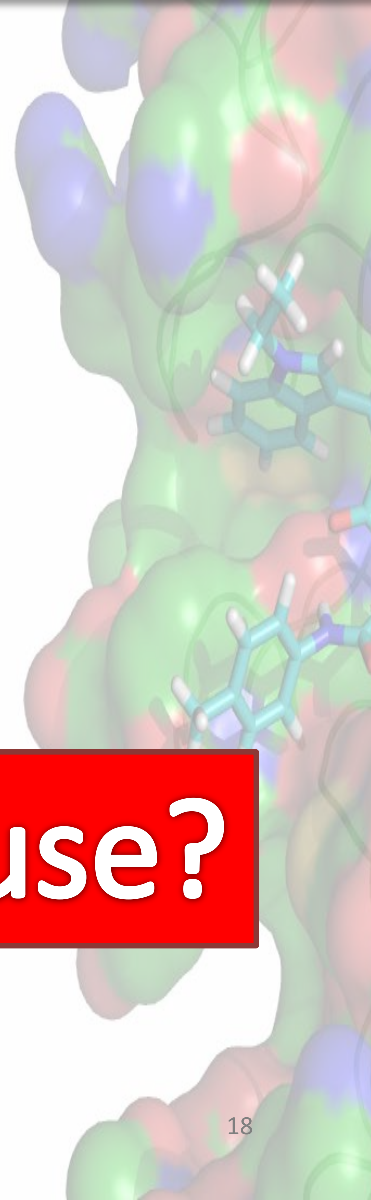
```python
# the first argument is the input file
smi_input_file = sys.argv[1]
# the second argument is the output file
sdf_output_file = sys.argv[2]
# the third argument is the number of conformers
n = int(sys.argv[3])

# write out the molecules to the output file (SDF)
writer = Chem.SDWriter(sdf_output_file)
# the SMILES input file
suppl = Chem.SmilesMolSupplier(smi_input_file, titleLine=False)

for mol in suppl:
    if mol:
        # add Hydrogens
        molH = Chem.AddHs(mol)
        # create n conformers for molecule
        confIds = AllChem.EmbedMultipleConfs(molH, n)
        # E optimize
```

But what value of *n* to use?

```python
# flush and close output file
writer.flush()
writer.close()
```

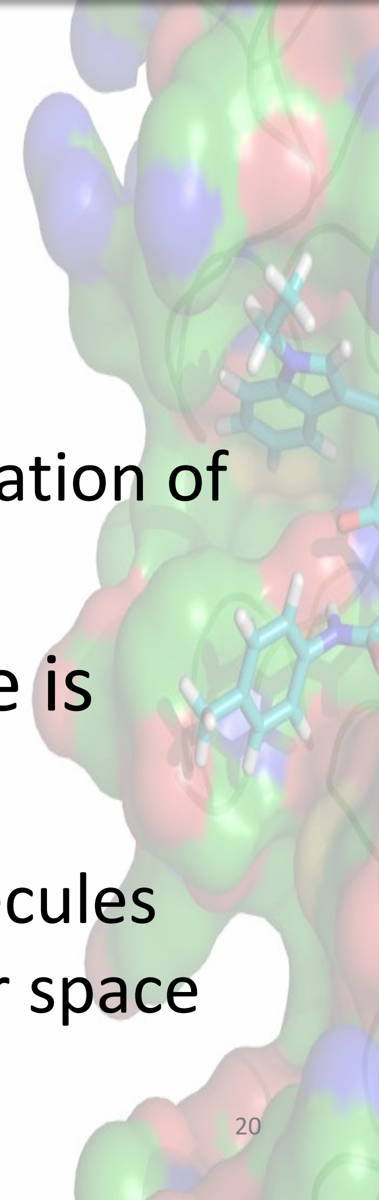


Deciding for a value of *n*

- Function of rotatable bonds of molecules
- Ran an experiment generating 10, 50, 100, 200, 300, 400, 500, 1000 conformers for our test set and partitioned results by rotatable bonds
- Trade-off between RMSD accuracy to crystal structure and number of conformers to generate

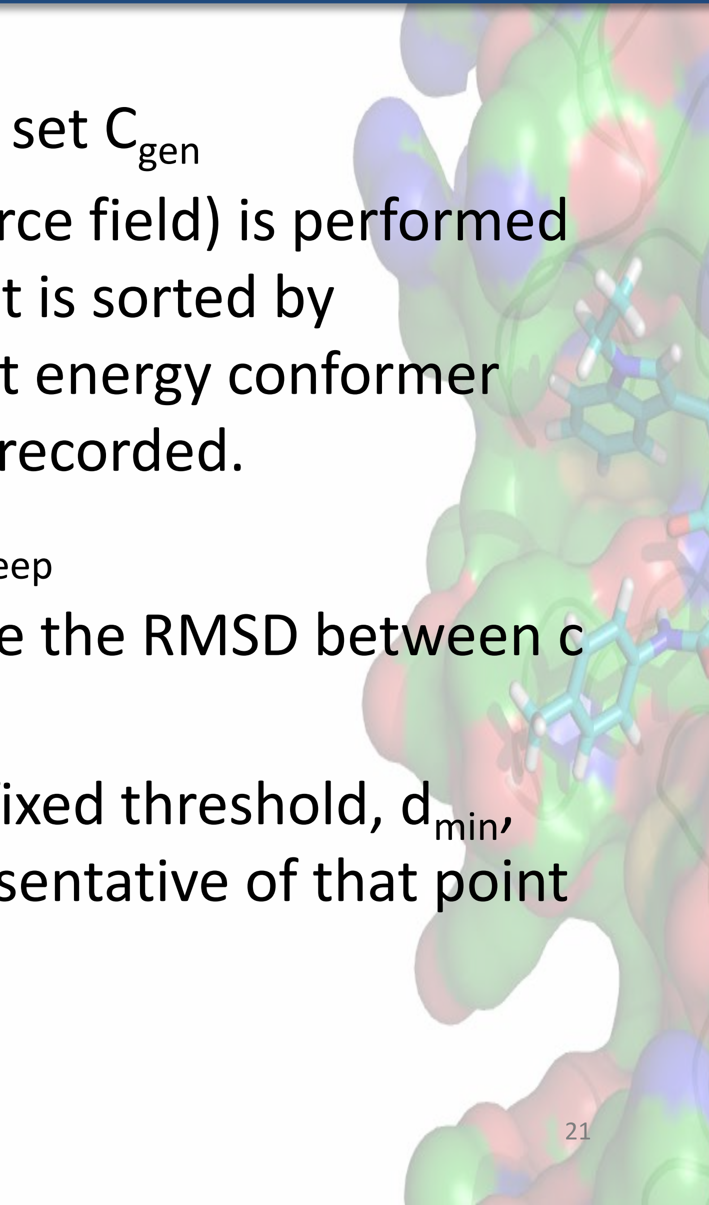
$$n = \begin{cases} 50 & \text{if } n_{rot} \leq 7 \\ 200 & \text{if } n_{rot} \geq 8 \text{ and } n_{rot} \leq 12 \\ 300 & \text{otherwise} \end{cases}$$

Things to watch out for (i)



- RDKit allows to remove very similar conformers from the ensemble (`pruneRmsThresh`)
  - But this only works **before** energy optimization of the molecule!
- Clustering needed again after ensemble is generated
  - After UFF energy minimization some molecules will fall in the same locations in conformer space

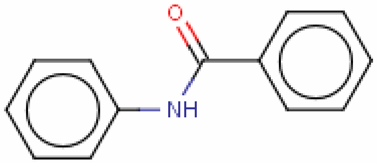
Clustering Algorithm

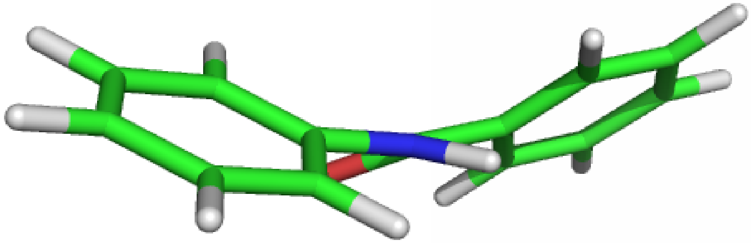


1. Using RDKit, generate *n* conformers in set C_{gen}
2. Energy minimization (using the UFF force field) is performed on every conformer. The conformer list is sorted by increasing energy value and the lowest energy conformer (the first conformer in the list), c_{low}, is recorded.
3. Remove c_{low} from C_{gen} and add it to C_{keep}
4. For each conformer, c, in C_{gen}, compute the RMSD between c and each conformer in C_{keep}
   - If any RMSD value is smaller than a fixed threshold, d_{min}, discard c as we already have a representative of that point in conformational space.
   - Otherwise add c to C_{keep}

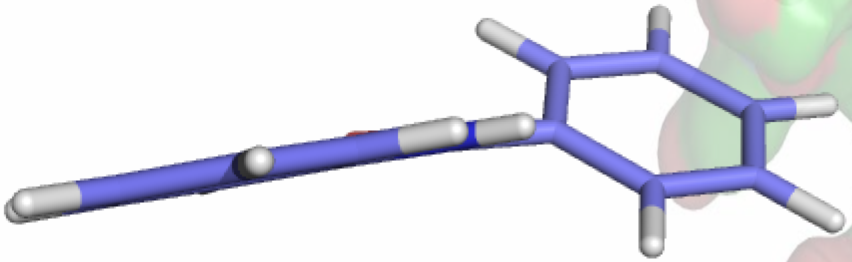
Things to watch out for (ii)

- Planarity of secondary and tertiary amides
- More of a RDKit UFF optimization implementation defect
- Possible solution is to add force field distance constraints





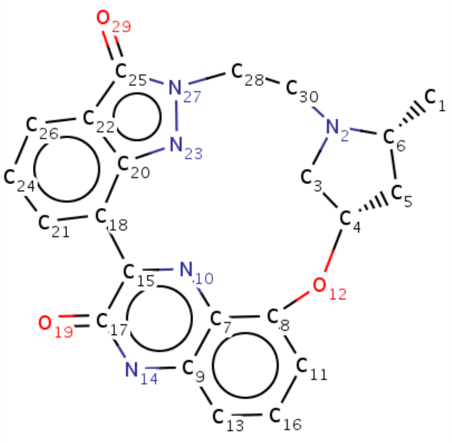
Out-of-plane (poor)



Planar secondary amide (good)

Things to watch out for (iii)

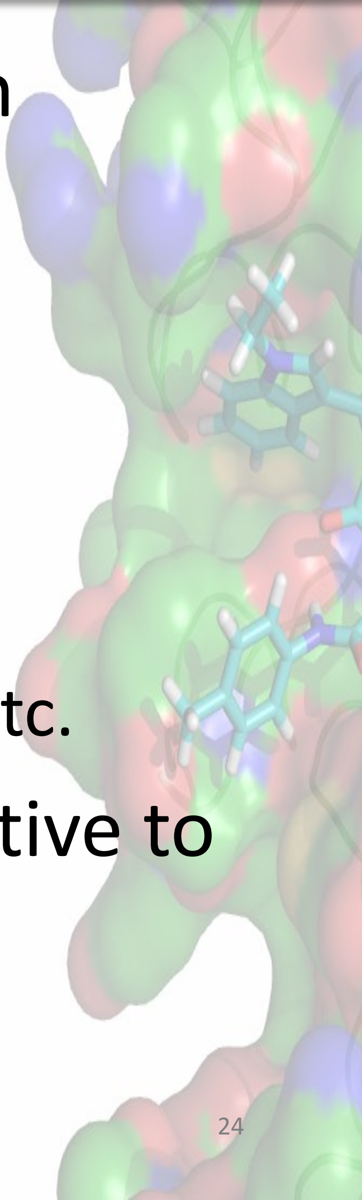
- Very rarely, conformer generation fails, e.g.



- Sometimes trying a different random seed fixes this
- Bug report filed
  - Fixed in revision 1952, use `ignoreSmoothingFailures=True` when this happens

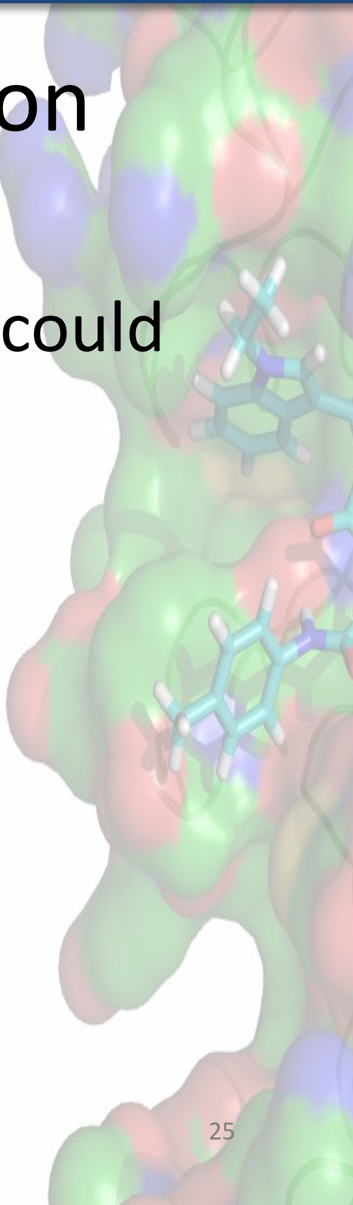
Conclusions

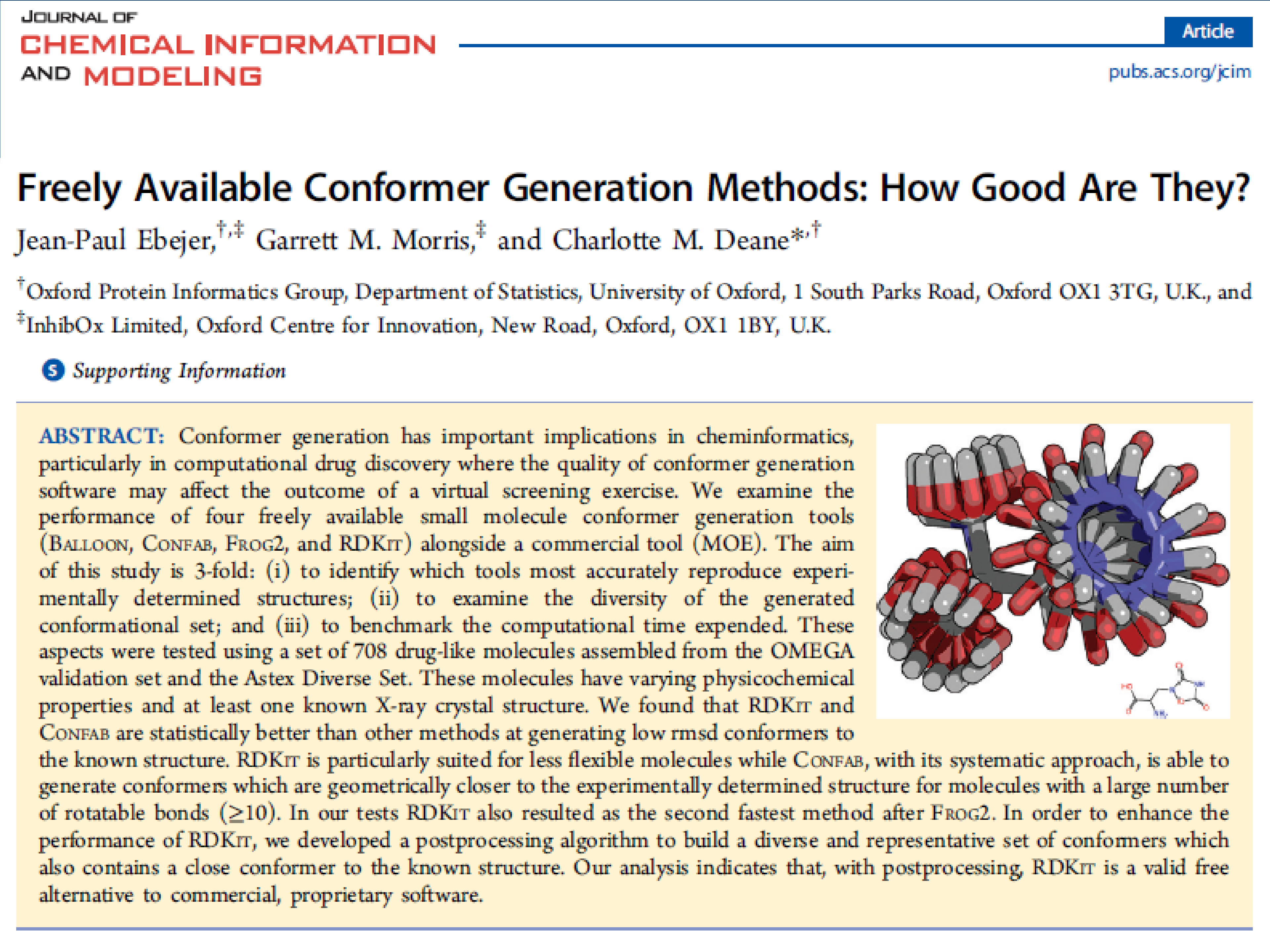
- Used RDKit to generate over 200 million conformers
  - It works!
- Selecting a conformer generation tool depends on other factors as well
  - *e.g.* ability to explore energetic landscape, integration in cheminformatics workflow etc.
- Open source tools offer a viable alternative to commercial, closed source, proprietary software



Future work

- Multi-threaded conformer generation
  - Most time spent in energy minimization
  - Each conformer independent of the other, could be minimized in separate thread
- Move towards a knowledge based approach
  - Use common fragments in the CSD/PDB





Journal of Chemical Information and Modeling

Article

pubs.acs.org/jcim

Freely Available Conformer Generation Methods: How Good Are They?

Jean-Paul Ebejer,[†,‡] Garrett M. Morris,[‡] and Charlotte M. Deane*[,†]

[†] Oxford Protein Informatics Group, Department of Statistics, University of Oxford, 1 South Parks Road, Oxford OX1 3TG, U.K., and [‡] InhibOx Limited, Oxford Centre for Innovation, New Road, Oxford, OX1 1BY, U.K.

S *Supporting Information*

ABSTRACT: Conformer generation has important implications in cheminformatics, particularly in computational drug discovery where the quality of conformer generation software may affect the outcome of a virtual screening exercise. We examine the performance of four freely available small molecule conformer generation tools (Balloon, Confab, Frog2, and RDKit) alongside a commercial tool (MOE). The aim of this study is 3-fold: (i) to identify which tools most accurately reproduce experimentally determined structures; (ii) to examine the diversity of the generated conformational set; and (iii) to benchmark the computational time expended. These aspects were tested using a set of 708 drug-like molecules assembled from the OMEGA validation set and the Astex Diverse Set. These molecules have varying physicochemical properties and at least one known X-ray crystal structure. We found that RDKit and Confab are statistically better than other methods at generating low rmsd conformers to the known structure. RDKit is particularly suited for less flexible molecules while Confab, with its systematic approach, is able to generate conformers which are geometrically closer to the experimentally determined structure for molecules with a large number of rotatable bonds (≥10). In our tests RDKit also resulted as the second fastest method after Frog2. In order to enhance the performance of RDKit, we developed a postprocessing algorithm to build a diverse and representative set of conformers which also contains a close conformer to the known structure. Our analysis indicates that, with postprocessing, RDKit is a valid free alternative to commercial, proprietary software.

Acknowledgements

- **RDKit**
  - Greg Landrum, Nathan Brown
- **InhibOx Limited**
  - Paul Finn, Garrett Morris
- **Oxford Protein Informatics Group, University of Oxford**
  - Charlotte Deane
- **Financial Support**
  - Marie Curie Fellowship