Biopolymers in RDKit

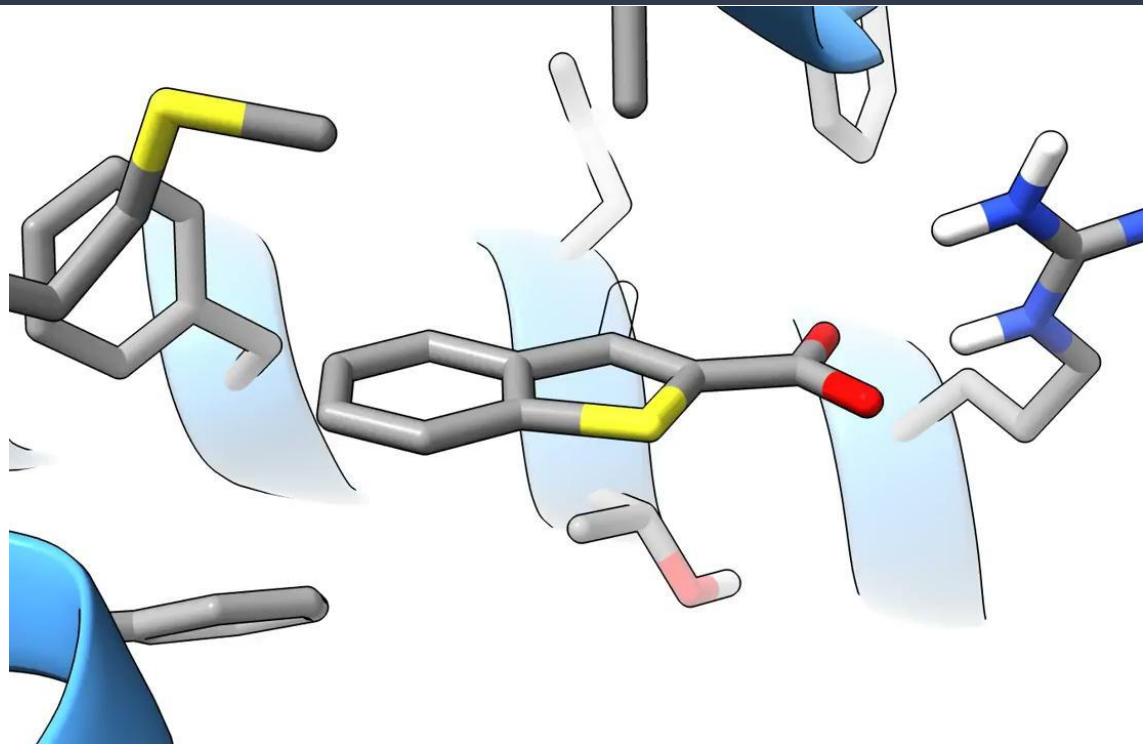
Richard Gowers
OpenFreeEnergy
RDKit UGM 2022

What is Open Free Energy?

Open Free Energy is an open source software project funded by a pre-competitive consortium of industry partners.

Our software helps researchers performing free energy calculations by providing common APIs for defining and executing calculations.

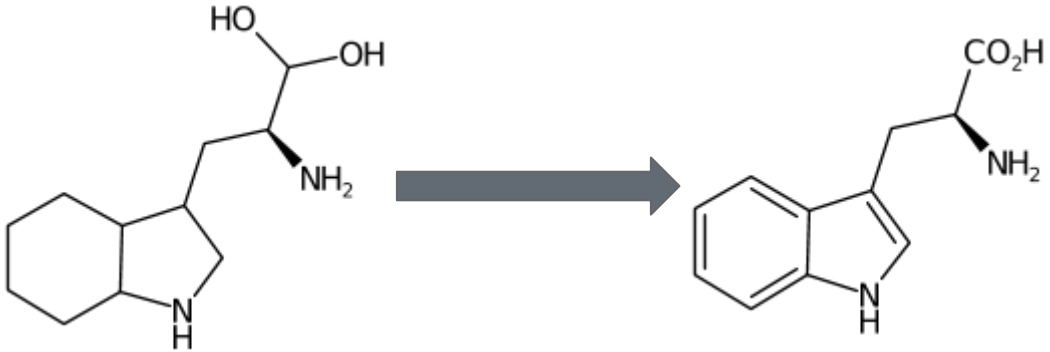
(We're also hiring if you like free energy calculations)



Motivations / Goals

Want to have a chemical model from PDB(x) including bond orders and formal charges.

Historically molecular mechanics didn't care so much about bond orders, this is changing with latest developments in forecfields.



Recap of biopolymer formats (for 3D models)

PDB

- Fixed column format
- Regularly misused and exists in various dialects.

PDBx (because X makes everything sound cooler)

- The new official format, mmcif-style.
- Compared to PDB, is better/more extensible.
- Not very well adopted/supported.

Also MMTF

- A messagepack based format with lots of compression hacks
- Surprisingly good and came with bindings.
- Now officially deprecated / abandoned :(

```
_atom_site.group_PDB
_atom_site.id
_atom_site.type_symbol
_atom_site.label_atom_id
_atom_site.label_alt_id
_atom_site.label_comp_id
_atom_site.label_asym_id
_atom_site.label_entity_id
_atom_site.label_seq_id
_atom_site.pdbx_PDB_ins_code
_atom_site.Cartn_x
_atom_site.Cartn_y
_atom_site.Cartn_z
_atom_site.occupancy
_atom_site.B_iso_or_equiv
_atom_site.pdbx_formal_charge
_atom_site.auth_seq_id
_atom_site.auth_comp_id
_atom_site.auth_asym_id
_atom_site.auth_atom_id
_atom_site.pdbx_PDB_model_num
ATOM   1    N  N   . VAL A 1 1   ? 6.204   16.869  4.854   1.00 49.05 ? 1   VAL A N   1
ATOM   2    C  CA  . VAL A 1 1   ? 6.913   17.759  4.607   1.00 43.14 ? 1   VAL A CA  1
ATOM   3    C  C   . VAL A 1 1   ? 8.504   17.378  4.797   1.00 24.80 ? 1   VAL A C   1
ATOM   4    O  O   . VAL A 1 1   ? 8.805   17.011  5.943   1.00 37.68 ? 1   VAL A O   1
ATOM   5    C  CB  . VAL A 1 1   ? 6.369   19.044  5.810   1.00 72.12 ? 1   VAL A CB  1
ATOM   6    C  CG1 . VAL A 1 1   ? 7.009   20.127  5.418   1.00 61.79 ? 1   VAL A CG1 1
ATOM   7    C  CG2 . VAL A 1 1   ? 5.246   18.533  5.681   1.00 80.12 ? 1   VAL A CG2 1
```

The Chemical Component Dictionary

For PDBx, the new solution to defining bonds is the Chemical Component Dictionary (CCD) which lists all possible residues (fragments).

However

- No (free?) software exists for applying this to PDBx
- This dictionary is likely incomplete as the PDB format is regularly abused anyway

Chemical Component Dictionary

https://www.wwpdb.org/data/ccd

Current state of PDB support in RDKit

- RDKit has Chem.MolFromPDBFile
- Atom connectivity is guessed based upon geometric criteria (& CONECT records)
- Double bonds are identified based upon heuristics on atom and residue names.
- PDBx is not currently supported
    - See RDKit PR: #4812
- Current double bond guessing is limited to standard amino acids

Double bond heuristics:

https://github.com/rdkit/rdkit/blob/master/Code/GraphMol/FileParsers/ProximityBonds.cpp

Alternate solutions for biopolymer support?

OpenMM has more flexible xml representations of residue templates to infer bonding (i.e. not distances)

- This is being extended to include bond orders, see OpenMM PR #3770

OpenBabel has PerceiveBondOrders which uses bond angles, lengths, torsions etc to guess orders.

This is probably heavily inspired by “Cruft to Content”

Finally, gemmi is a good library for giving a DOM view of PDB(x) files, but doesn’t fix my bond problem.

OpenMM Bond inferring & improvements(?):

https://github.com/openmm/openmm/pull/3770

OpenBabel PerceieveBondOrders:

https://github.com/openbabel/openbabel/blob/master/src/mol.cpp#L3192

Cruft to Content:

https://www.daylight.com/meetings/mug01/Sayle/m4xbondage.html

Gemmi:

https://gemmi.readthedocs.io/en/latest/index.html

An alternative way forward for biopolymers?

- Over in OpenFF toolkit, we have something like rdkit's AssignBondOrdersFromTemplate but has multiple templates.
- Currently this does standard amino acids (incl. their tautomers)
- It basically iterates all amino acids and looks for substructure matches
- For handling the CCD (all possible residues) we will want to invert this pattern and instead build a substructure database and search this for each residue.

Existing hacky way to assign bond orders from templates (using rdkit):

https://github.com/openforcefield/openff-toolkit/blob/main/openff/toolkit/utils/rdkit_wrapper.py#L257

Conclusion

I’ve got a hacky solution for applying the CCD to protein molecules in RDKit.

It seems to work for 90% of cases (which yes is 10% of the effort).

Open questions for y’all:

- Is this a stupid approach?
- Do you have a better approach?

I’ll be at the hackathon too.