molecular docking

February 26, 2021

molecular docking

molecular docking predicts how small molecules bind to a protein

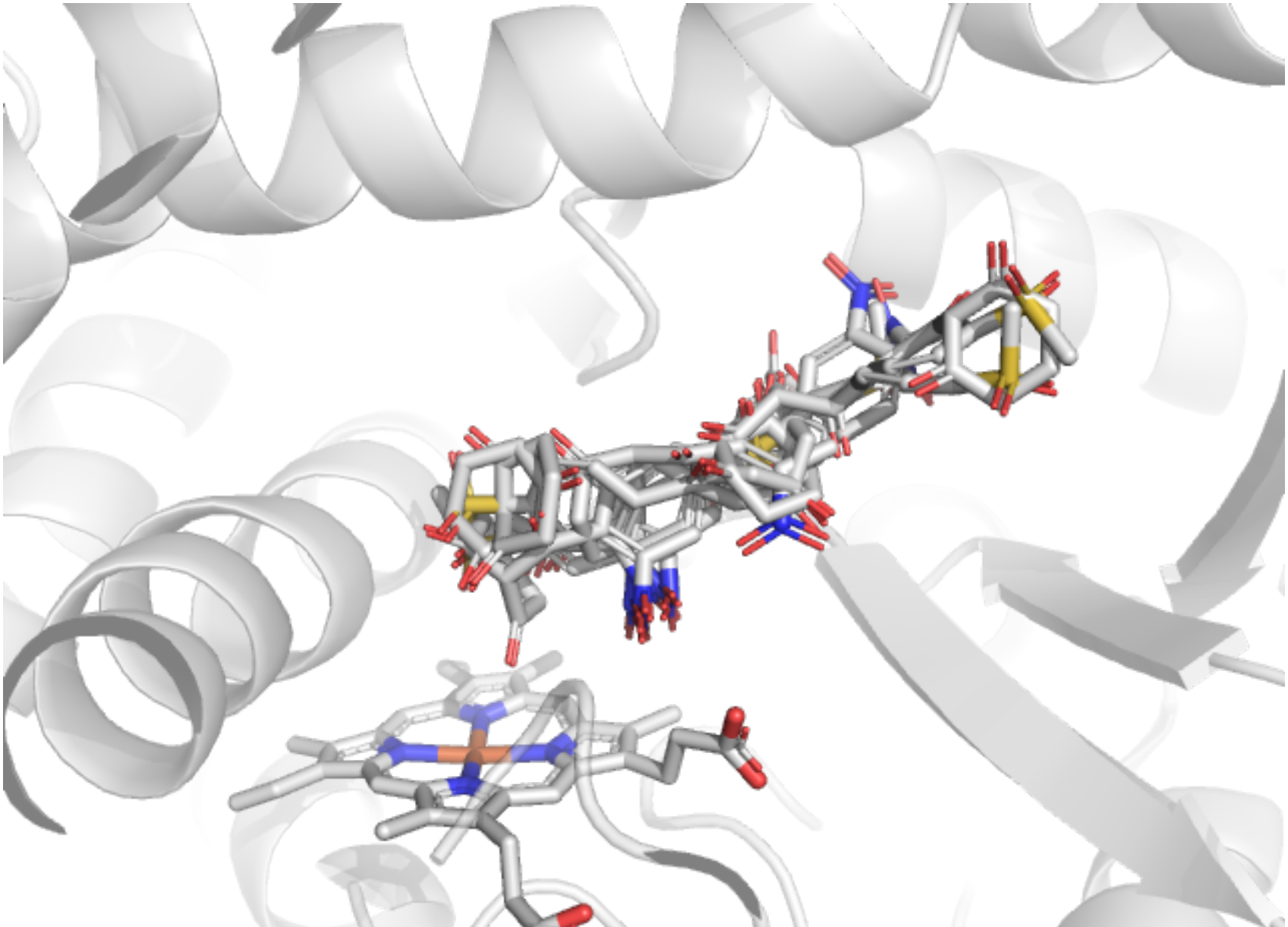
two docking approaches

- shape complimentarity
- simulation

shape complimentarity

simulation

Figure: *poses* generated by simulation-based docking



simulation

- **search conformations**
  - genetic algorithm
  - gradient descent
  - monte carlo search
- **score conformations**
  - energy calculations
  - extra parameters

simulation - scoring conformations

aim: calculate binding energy ΔG of interaction

$$c = \Sigma force_{atom_i atom_j} \times distance_{atom_i atom_j} \quad (1)$$

for all atom pairs *i* and *j* in consideration

sum of molecular interaction energies

simulation - scoring conformations

aim: calculate binding energy ΔG of interaction

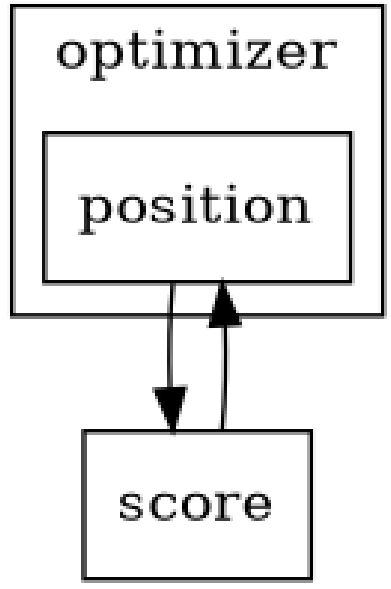
$$c = \Sigma force_{atom_i atom_j} \times distance_{atom_i atom_j} + tuning \qquad (2)$$

tuned on the **PDB-Bind database**

- experimentally measured binding affinities for protein-ligand structures in the protein data bank (PDB)
- 17,000 protein-ligand data points

simulation - search / optimization algorithms

aim: find poses that minimize calculated ΔG of interaction



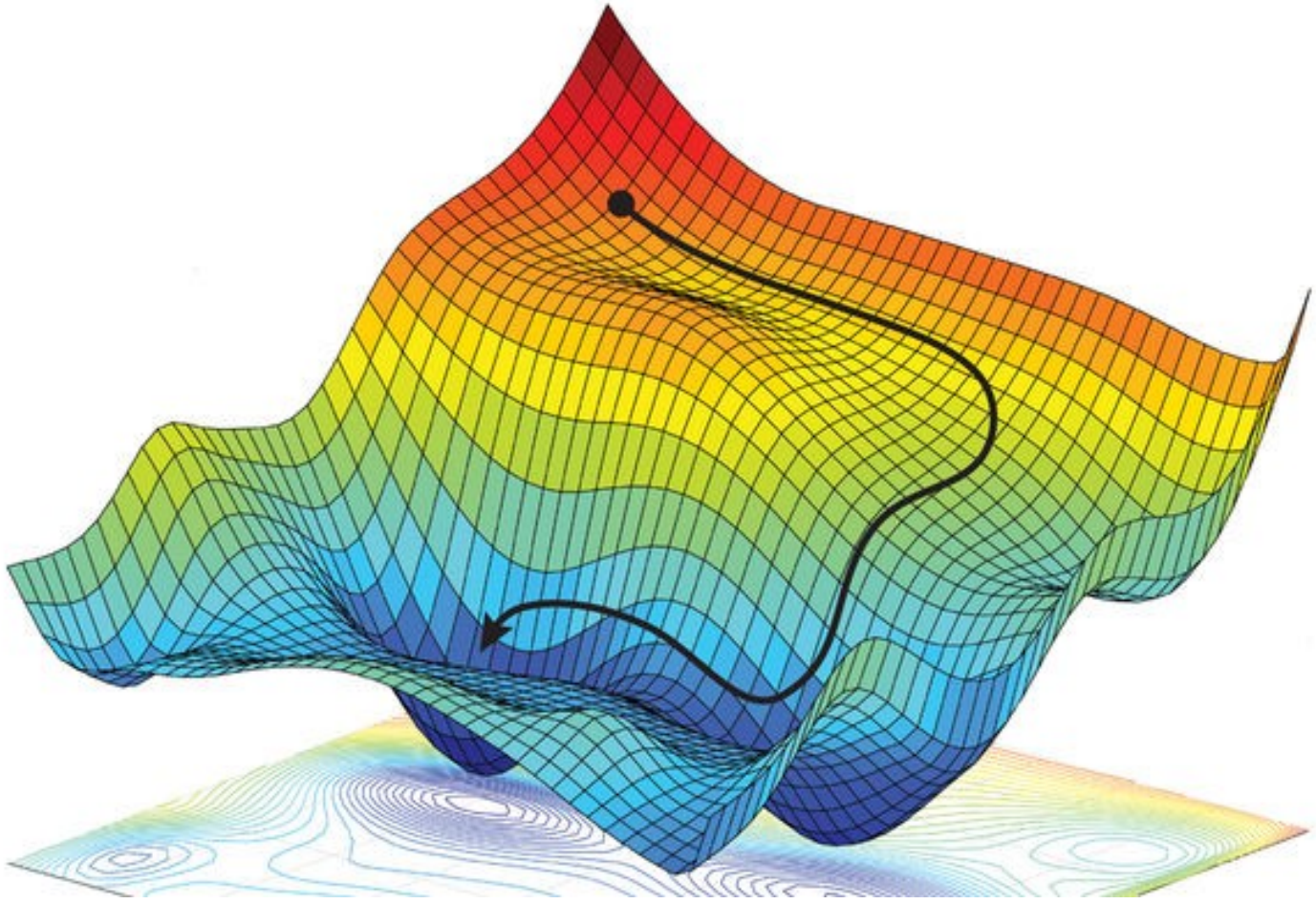
simulation - search / optimization algorithms

aim: find poses that minimize calculated ΔG of interaction

- **gradient descent** - smooth function optimization
- **genetic algorithms** - general function optimizers
- **monte carlo algorithms** - stochastic function optimizers

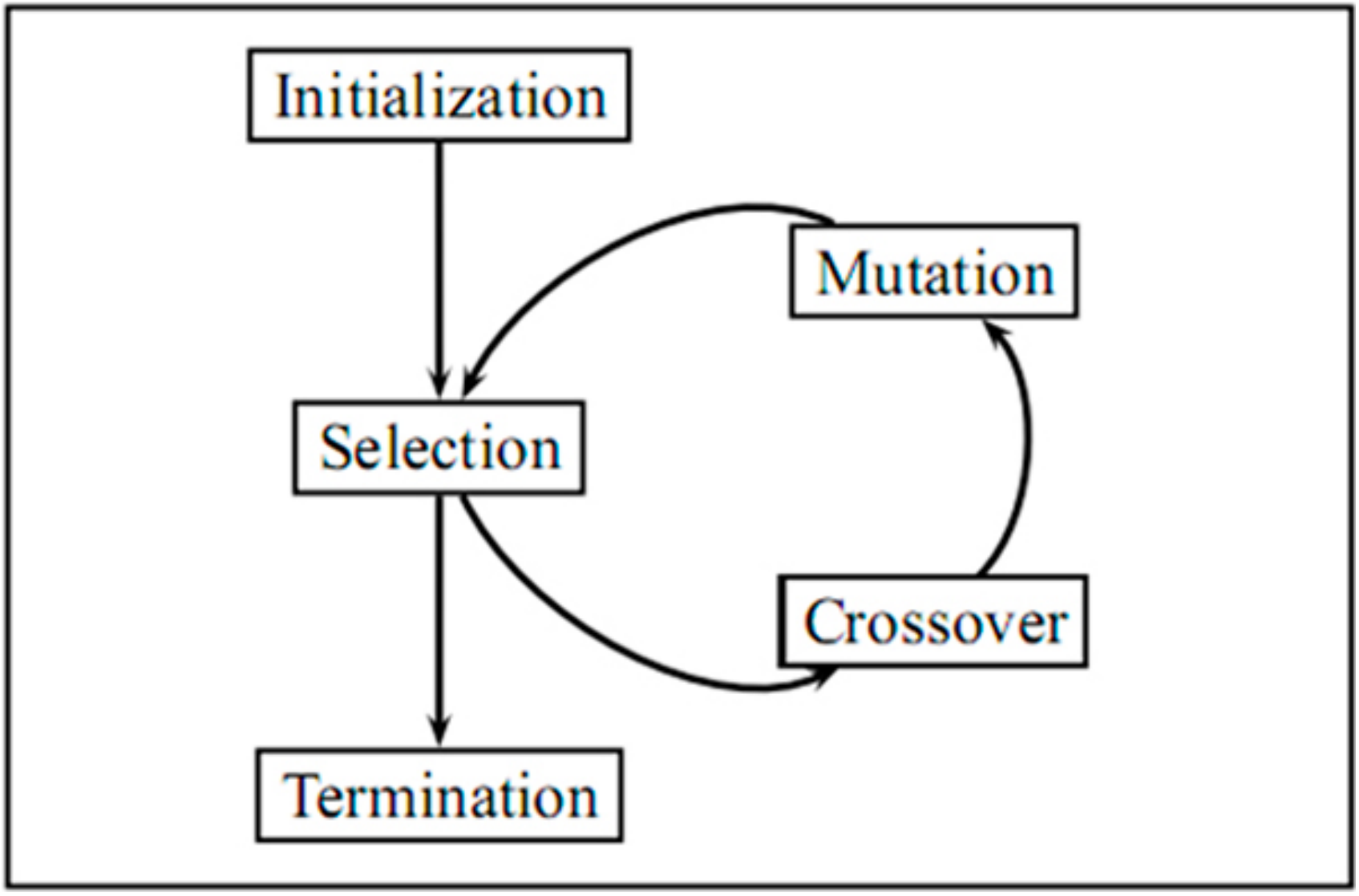
simulation - search / optimization algorithms

gradient descent optimization



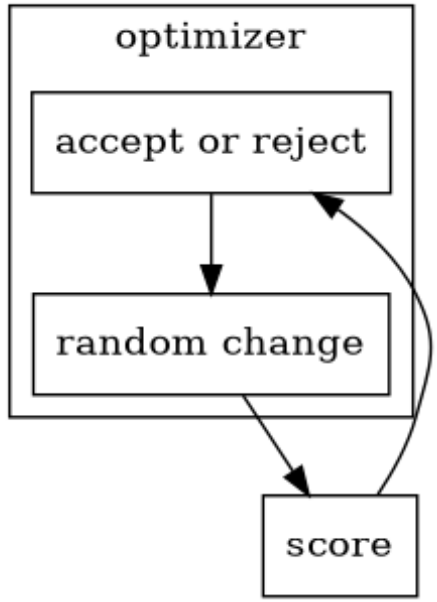
simulation - search / optimization algorithms

genetic algorithm optimization

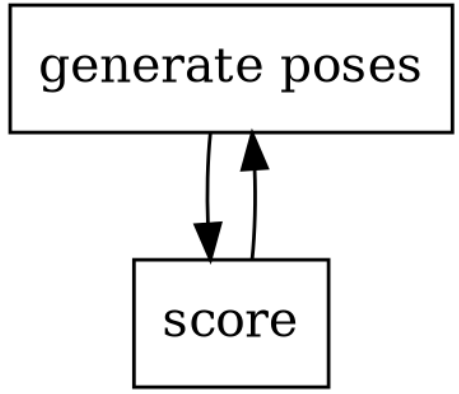


simulation - search / optimization algorithms

monte carlo optimization



summary of docking



caveats and advantages

caveats

- accuracy / computational cost tradeoff
- often don’t account for protein dynamics

advantages

- accurate enough to be useful
- **cheap!**

docking programs

- **autodock vina**
- FlexX

autodock vina

- free
- well known

autodock vina

```
conda install -c bioconda autodock-vina  conda
install -c conda-forge openbabel
```

autodock vina

```
vina --help
Input:
--receptor arg rigid part of the receptor (PDBQT)
--flex arg flexible side chains, if any (PDBQT)
--ligand arg ligand (PDBQT)


Search space (required):
--center_x arg X coordinate of the center
--center_y arg Y coordinate of the center
--center_z arg Z coordinate of the center
--size_x arg size in the X dimension (Angstroms)
--size_y arg size in the Y dimension (Angstroms)
--size_z arg size in the Z dimension (Angstroms)


Output (optional):
--out arg output models (PDBQT), the default is chosen
```

autodock vina

```
vina --receptor XXX.pdbqt --ligand XXX.pdbqt --center_x
0.0 --center_y 0.0 --center_z 0.0 --size_x 10 --size_y
10 --size_z 10
```

autodock vina - pdbqt file conversion with openbabel

```
obabel -i pdb -o pdbqt -O XXXX.pdbqt XXX.pdb
obabel -  CCCCCCCCC=O -o pdbqt -o ligand.pdbqt
```

vina in python with vinadiesel

- vina-diesel is a python wrapper for vina
- file conversion and cleaning is automated
- binding site specified by listing some residue numbers
- access results from objects in `pandas` dataframe.

vina in python with vinadiesel

installation

```
git clone https://github.com/UoMMIB/vina-diesel/

cd vina-diesel

# either:
conda install -c bioconda autodock-vina
conda install -c conda-forge openbabel
conda install -c conda-forge biopandas
pip install nwalign3

# or
conda create -f env.yml
conda activate vdsl

#then
pip install .
```

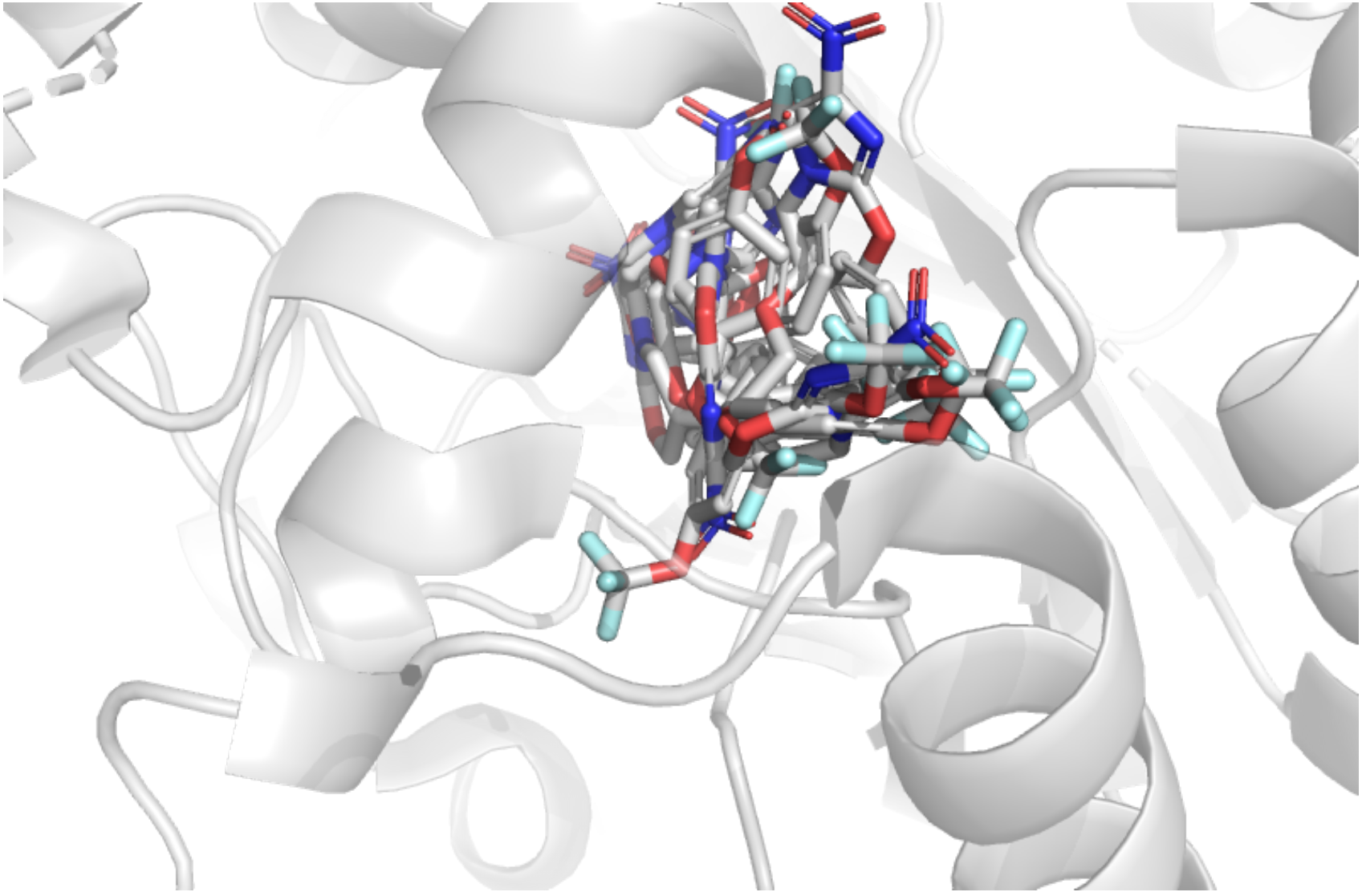
vina in python with vinadiesel

use

```
import vdsl
p = vdsl.protein('3b4y.pdb')
results = p.dock('[O-][N+](=O)c1cn2C[C@@H](COc2n1)OCc3ccc(C
target_sites = [38, 44, 175, 176, 199, 252, 256, 261, 283,
311],
exhaustiveness=1)
results.save('pa824')
```

vina in python with vinadiesel

results



vina in python with vinadiesel

other features

- *enz*-based interface
- easy binding site specification
- auto-cleaning
- access coordinates and energies for `results` and `protein` objects in a `pandas` dataframe for custom scoring