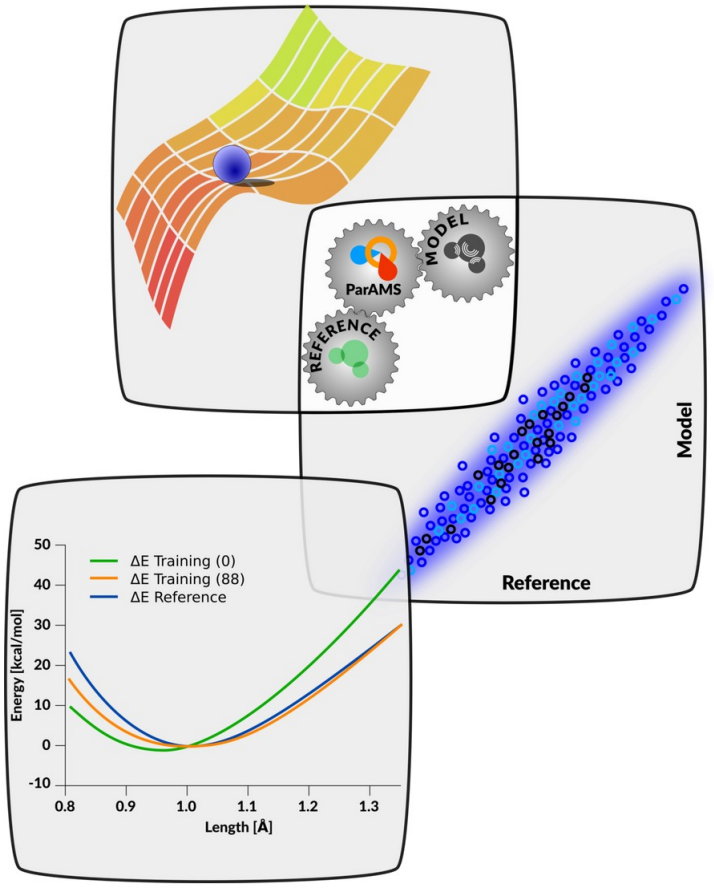
ParAMS: ReaxFF and DFTB parameter optimization

SCM
Software for
Chemistry &
Materials

ParAMS

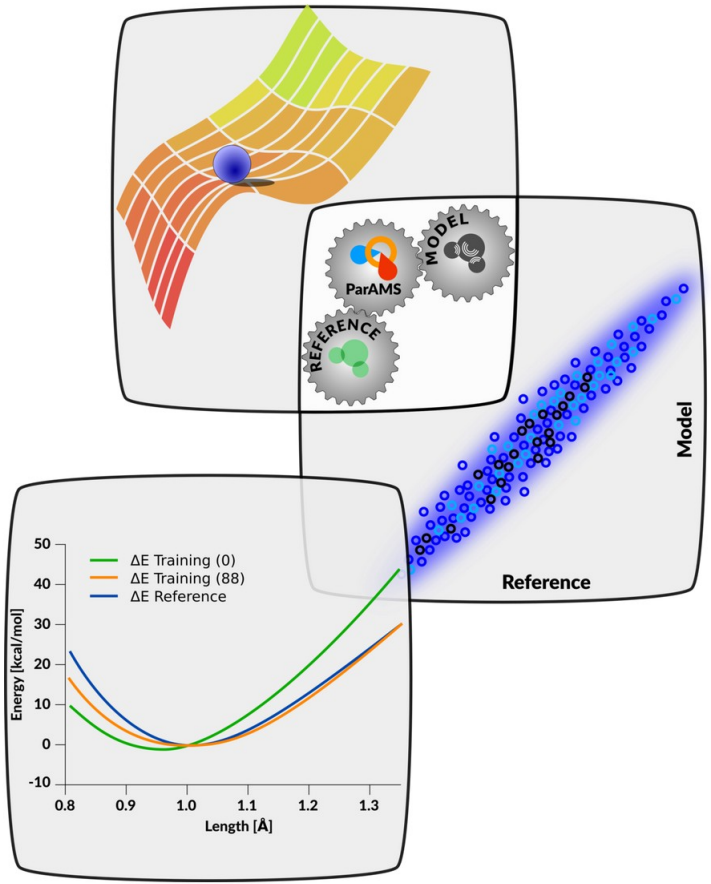


Matti Hellström, Product Manager

hellstrom@scm.com

Outline

- What is ParAMS?
- What can you do with ReaxFF and DFTB?
- Real-life ParAMS ReaxFF example from industry
- How does ParAMS work?
  - Add training data
  - Settings
  - Output
  - Various features
- Summary



What is ParAMS?

- Optimize ReaxFF and DFTB (GFN1-xTB) parameters
- Collaboration with Prof. Toon Verstraelen, Ghent University, Belgium.
- Requires a license for "*Advanced workflows and tools*", a new module in AMS2022
- ParAMS replaces the *AMStrain* module of previous AMS versions

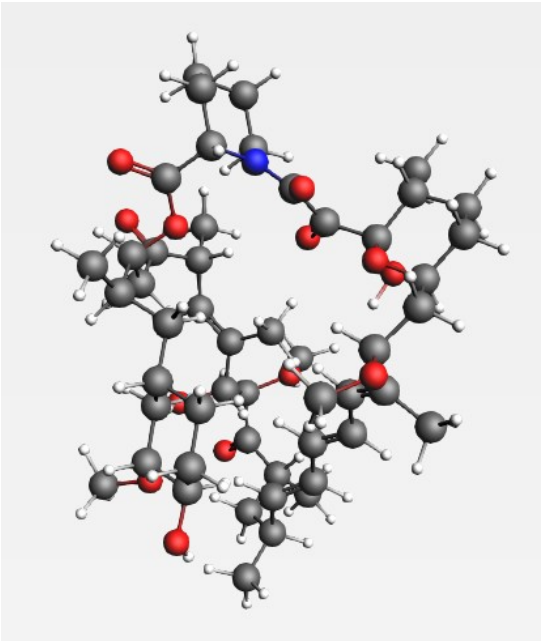
ParAMS: Parameter Optimization for Atomistic and Molecular Simulations

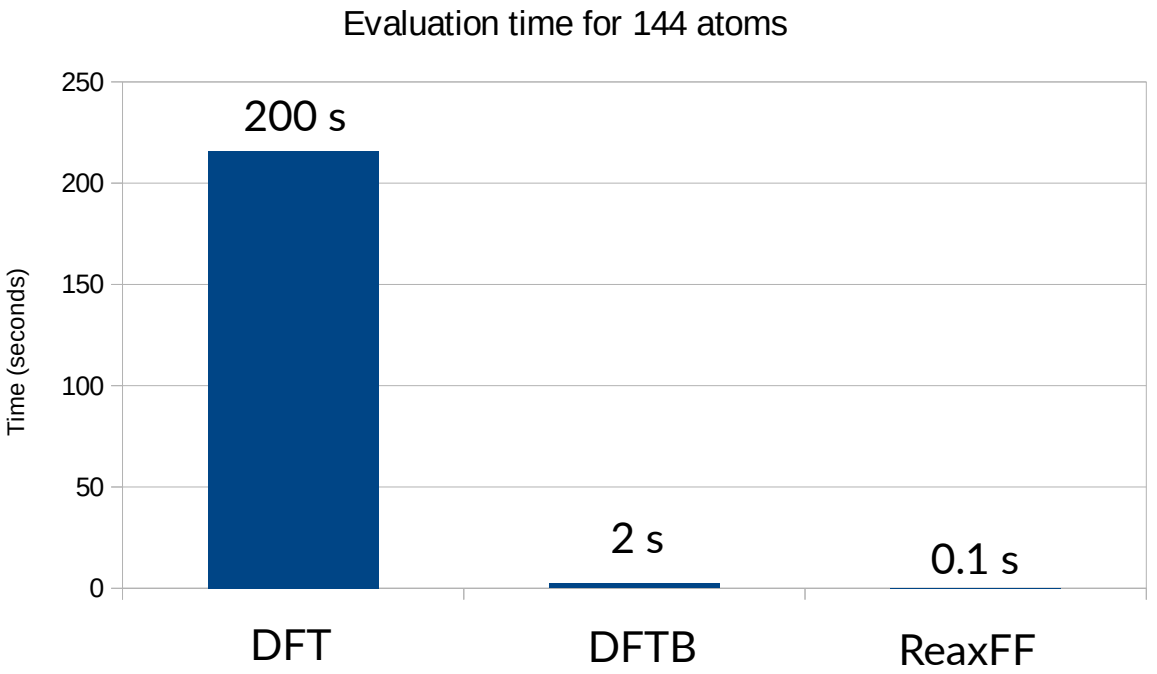
Leonid Komissarov, Robert Rüger, Matti Hellström, and Toon Verstraelen*

Cite this: *J. Chem. Inf. Model.* 2021, 61, 8, 3737−3743
Publication Date: May 13, 2021
https://doi.org/10.1021/acs.jcim.1c00333

Why ReaxFF or DFTB?

- Faster than DFT





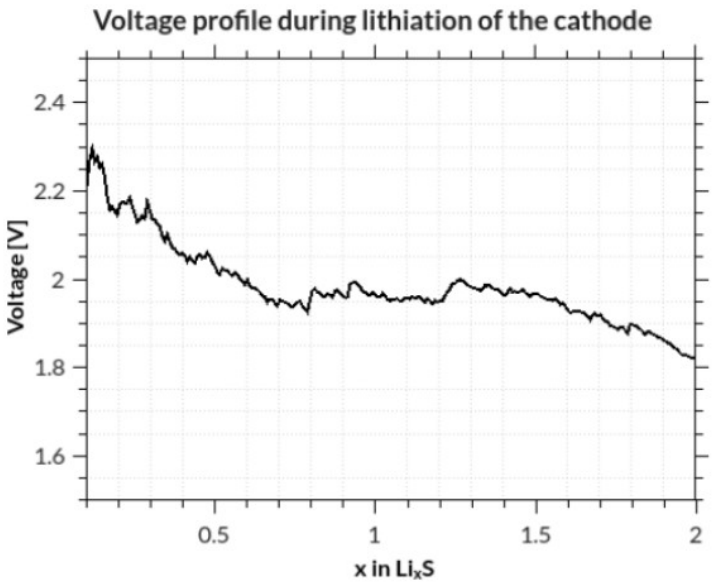
Why are ReaxFF and DFTB faster than DFT?

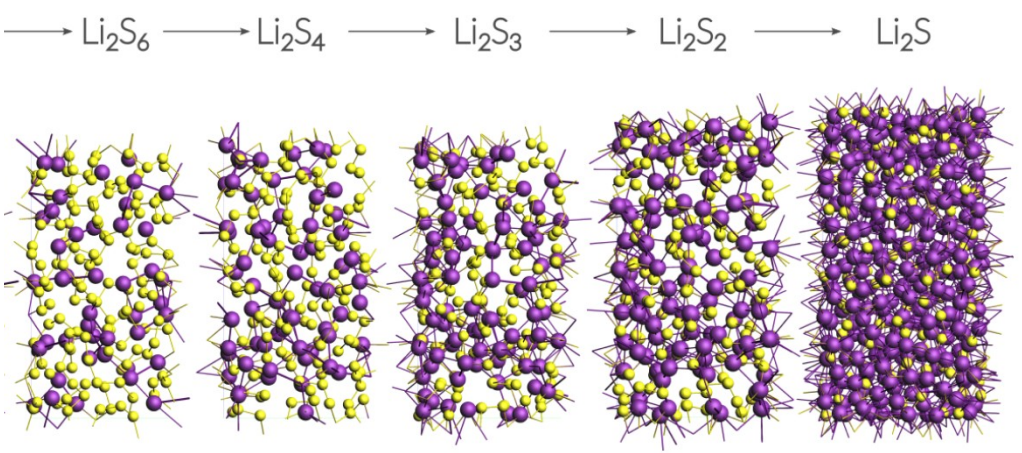
- Parameters
- Example:

  ReaxFF force field file
- ParAMS helps you optimize the parameters to give DFT-quality results

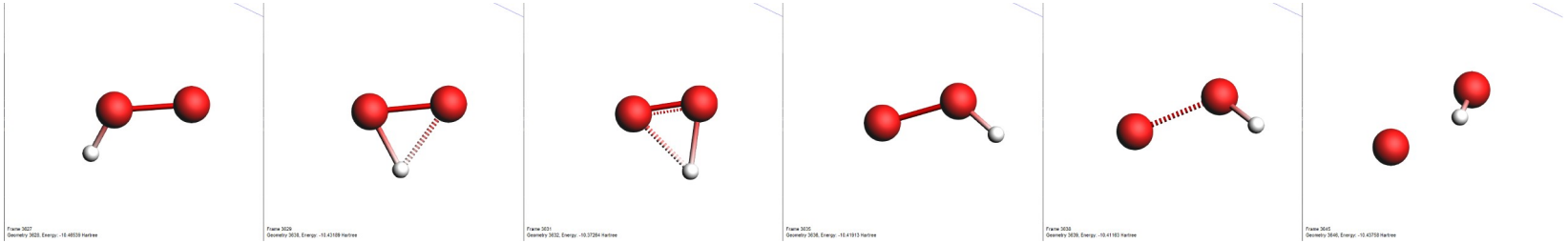
```
   2.6962
   0.0000
   0.0000
 3 ! Number of atoms


S    1.9186   2.0000  32.0600   1.6516   0.4937   0.7530   1.6593   6.0000
     9.0227   4.9055   4.0000  30.0000 112.1416   6.5745   9.0000   2.0000
     1.0000   3.4994  65.0000  12.0000  22.1978  15.3230   0.9745   0.0000
   -15.7363   2.8802   1.0338   6.2998   2.8793   0.0000   0.0000   0.0000
H    0.6646   1.0000   1.0080   1.6030   0.0600   0.7625  -0.1000   1.0000
     9.3951   4.5386   1.0000   0.0000 121.1250   3.8196   9.8832   1.0000
    -0.1000   0.0000  -0.1339   2.5732   2.6456   3.1680   1.0698   0.0000
   -12.9330   3.0626   1.0338   1.0000   2.8793   0.0000   0.0000   0.0000
Zn   1.8862   2.0000  65.3900   1.9200   0.2998   0.4828  -1.6836   2.0000
    11.5134  18.3776   2.0000   0.0078   0.0000   2.0219   5.7915   0.0000
    -1.2000   0.0000 266.4838   5.3430  10.1260   0.7590   0.0000   0.0000
    -3.0614   2.1158   1.0338   6.2998   2.5791   0.0000   0.0000   0.0000
 6 ! Number of bonds

 1  1  60.3691  31.1563   0.0000  -0.7739  -0.4781   1.0000  17.8574   0.3473
        0.6134  -0.1773   8.4125   1.0000  -0.0909   6.1095   1.0000   0.0000
 2  1 108.9303   0.0000   0.0000  -0.2724   0.0000   1.0000   6.0000   0.5333
        4.3945   1.0000   0.0000   1.0000  -0.0984   4.6210   1.0000   0.0000
 2  2 153.3934   0.0000   0.0000  -0.4600   0.0000   1.0000   6.0000   0.7300
        6.2500   1.0000   0.0000   1.0000  -0.0790   6.0552   0.0000   0.0000
 2  3   0.0000   0.0000   0.0000   0.0000  -0.5000   1.0000  50.0000   0.5000
        0.5000  -0.5000  30.0000   1.0000  -0.2000   8.0000   0.0000   0.0000
 3  3  31.5379   0.0000   0.0000  -0.7688  -0.2000   0.0000  16.0000   0.2229
        0.3583  -0.2000  15.0000   0.0000  -0.0983   4.2268   0.0000   0.0000
 1  3 112.1806   0.0000   0.0000  -0.3443  -0.5000   0.0000  35.0000   0.0346
        1.1695  -0.5000  25.0000   0.0000  -0.3381   3.2746   0.0000   0.0000
 0 ! Number of off-diagonal terms
 6 ! Number of angles
 1  1  1  70.3671   5.7180   7.0000   0.0000   0.3683   0.0000   2.4869
```

AMS-ReaxFF: Li battery voltage profiles





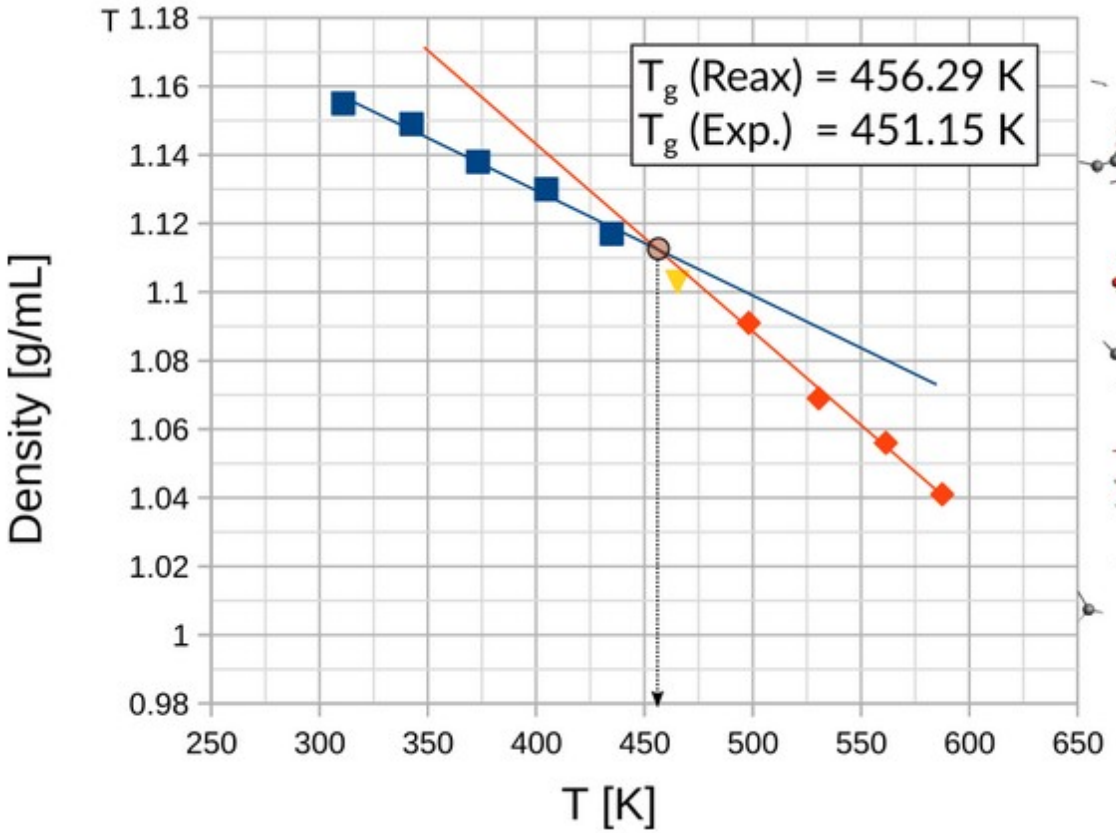
AMS-ReaxFF: analyze reactive trajectories with ChemTraYzer2

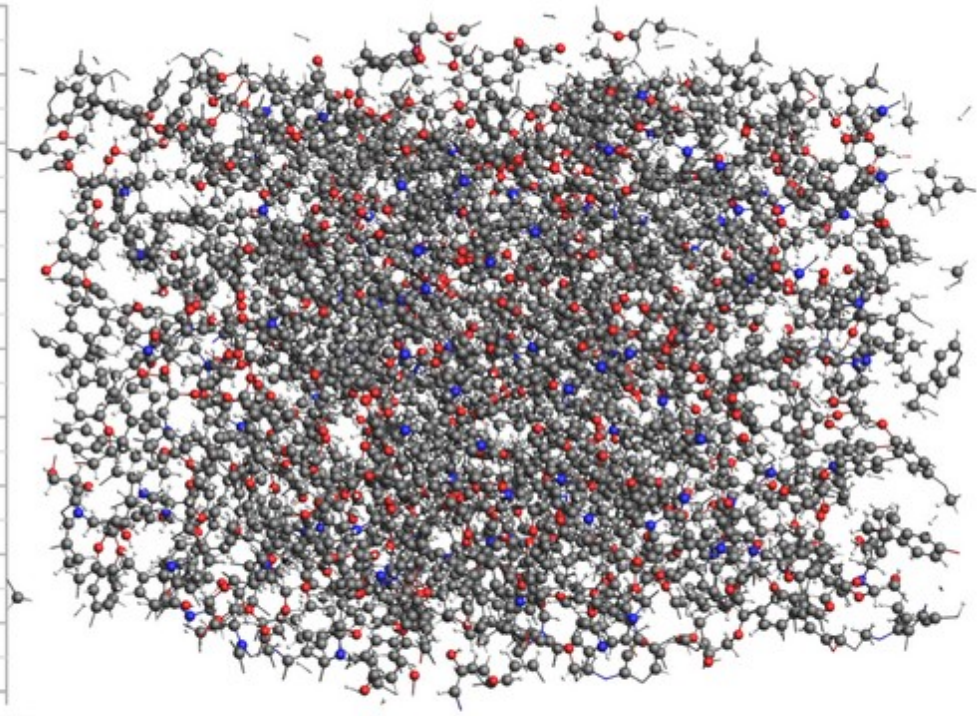


Reaction	Reaction Composition	Order	Rate constant k (cm^3, mol, s)
[O][O] + [H] => [O]O	O2 + H => HO2	2	1.407e+15
[O]O => [O][O] + [H]	HO2 => O2 + H	1	2.587e+12
[H][H] + [OH] => [H] + O	H2 + HO => H + H2O	2	3.881e+13
O[H]O => [OH] + O	H3O2 => HO + H2O	1	4.072e+13
[H] + [OH] => O	H + HO => H2O	2	1.903e+15

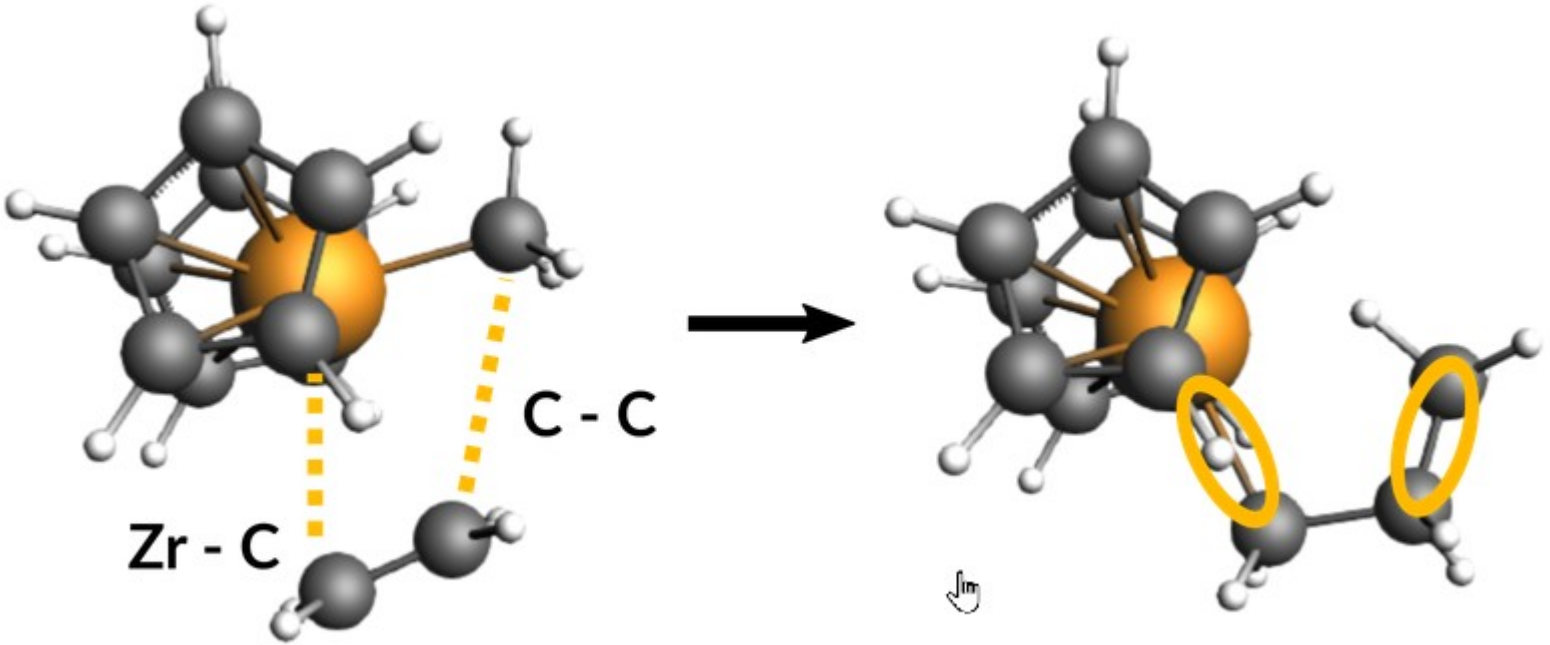
Index	Initial frame	Final frame	Initial time (fs)	Final time (fs)	Reactants	Products
2	819	819	2047.5	2047.5	[O]O	[O][O] + [H]
5	894	894	2235.0	2235.0	[O]O	[O][O] + [H]

AMS-ReaxFF: Polymer cross-linking and glass transition temperature

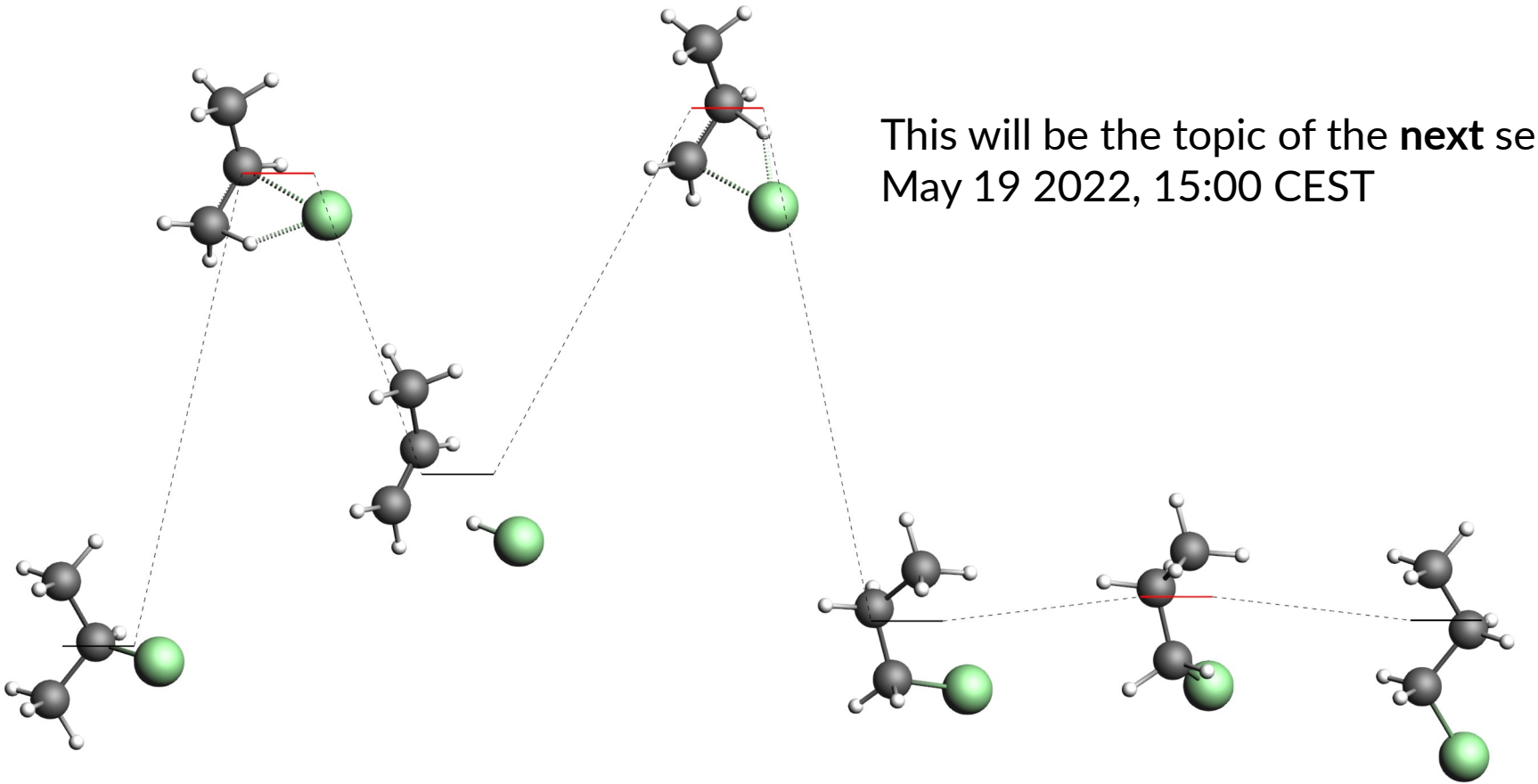




AMS-DFTB: catalysis



AMS-DFTB: automatic reaction pathways



This will be the topic of the **next** seminar!
May 19 2022, 15:00 CEST

AMS-DFTB: High-throughput screening of molecules, conformers, ...

Step 1

- Optimize the structure of all molecules in the database with DFTB and calculate excitation energies and oscillator strengths with TD-DFTB
- Select the molecules with electronic excitations of energies between 1 and 6 eV and non-zero oscillator strenght (since TD-DFTB is less accurate than TD-DFT, we opt for a larger energy range in this step). This should significantly reduce the number of molecules to be considered in the following steps.

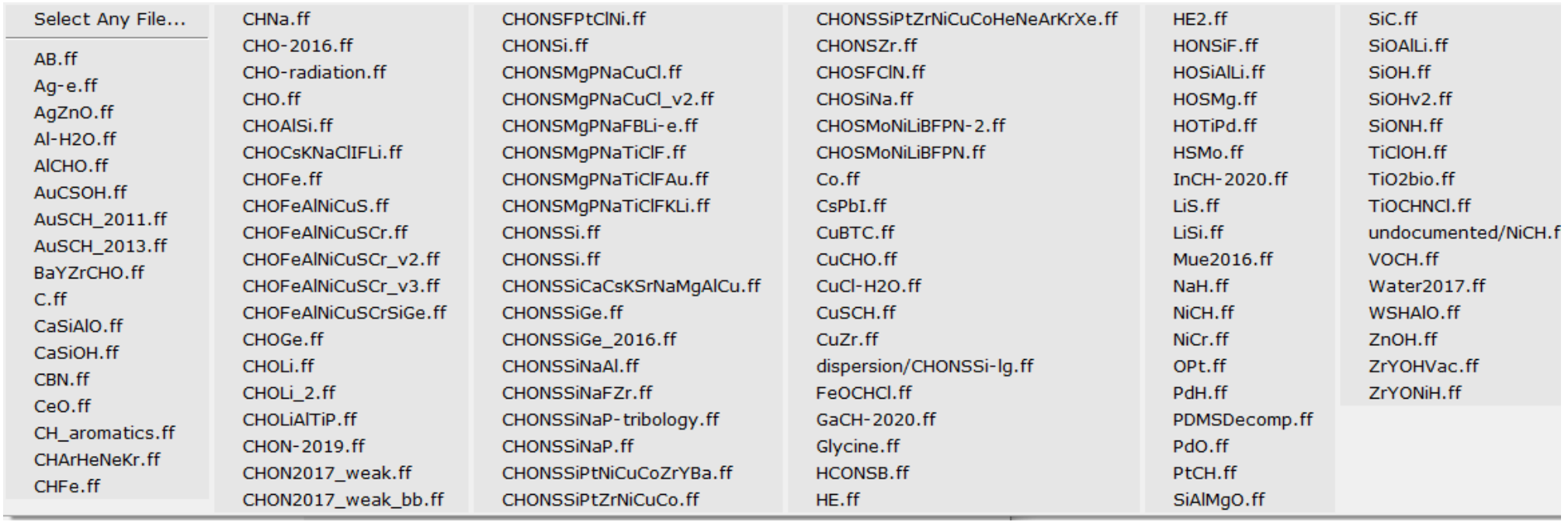
Step 2

- Optimize the structures of the promising molecules with ADF.
- Compute the electronic excitations energies using ADF's TD-DFT and select the molecules with excitations of energies between 2 and 4 eV (and non-zero oscillator strength).

https://www.scm.com/doc/plams/examples/ExcitationsWorkflow.html

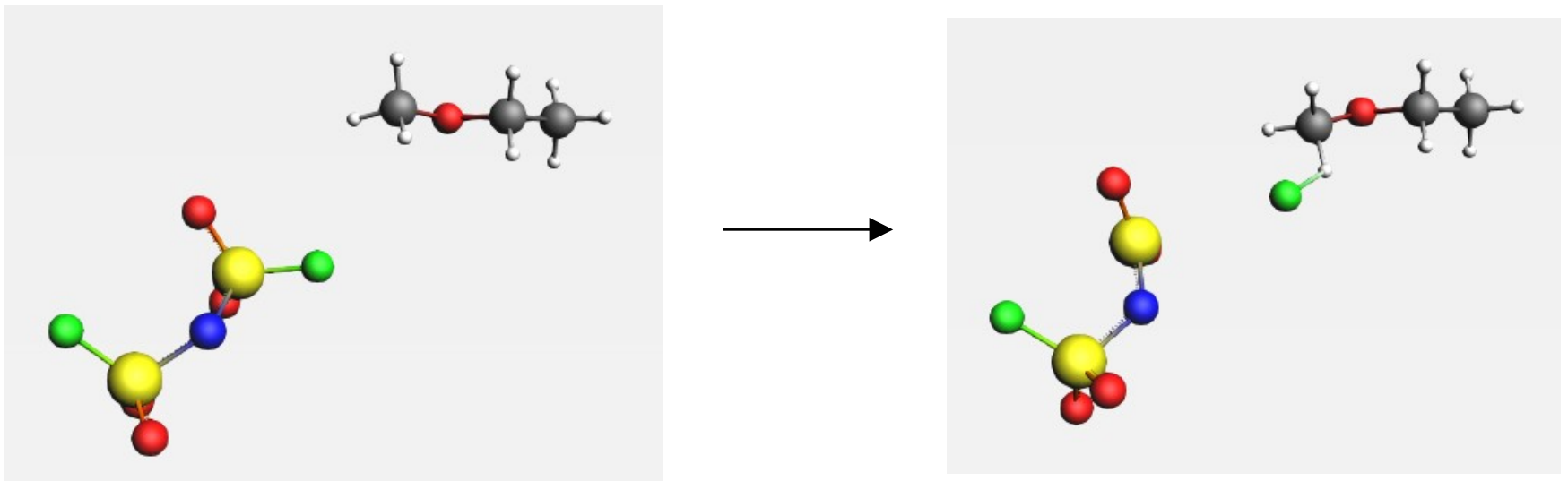
Preparametrized models in AMS

- AMS comes with many published ReaxFF force fields and DFTB parameters

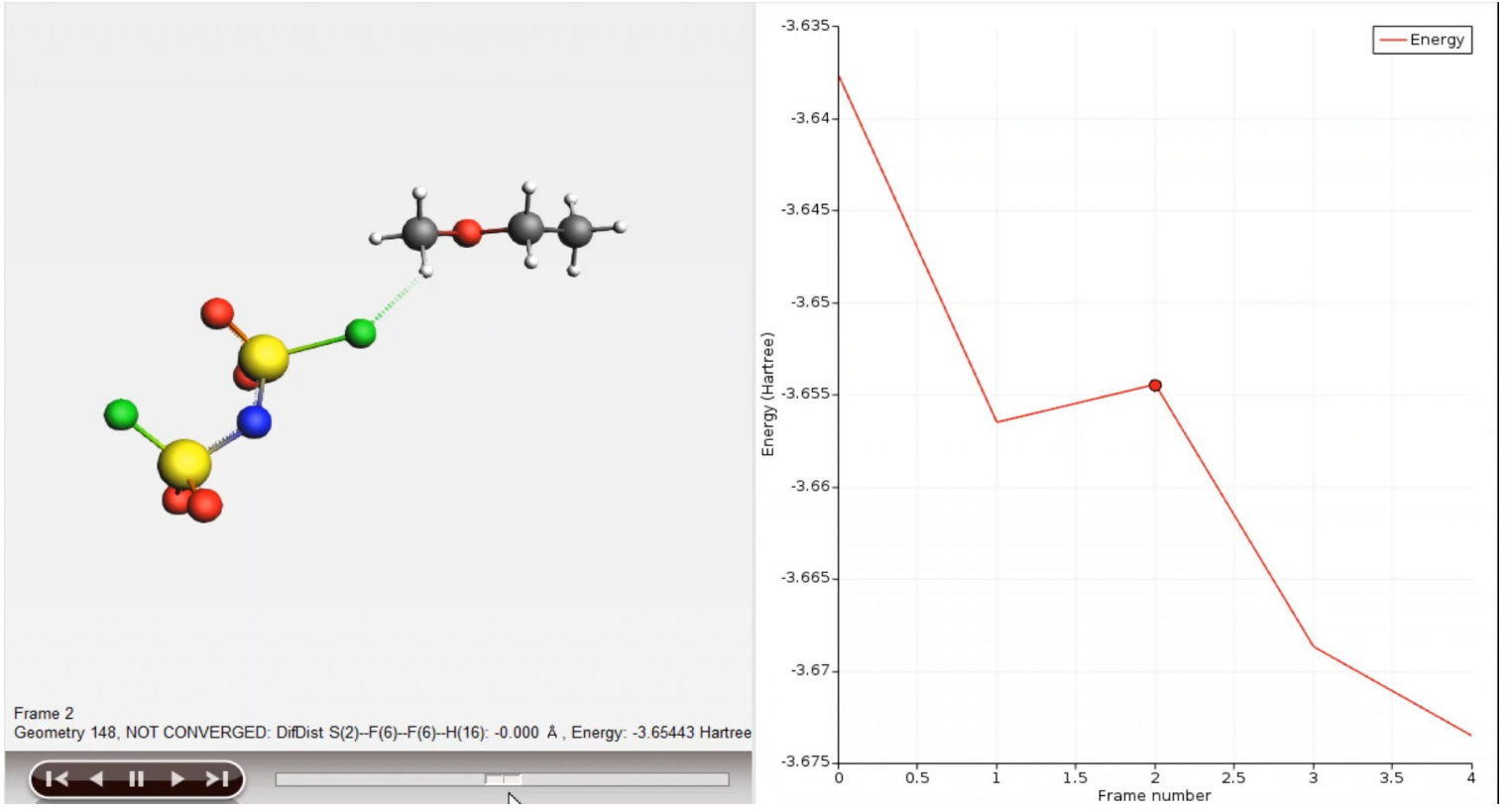


ReaxFF and DFTB problems

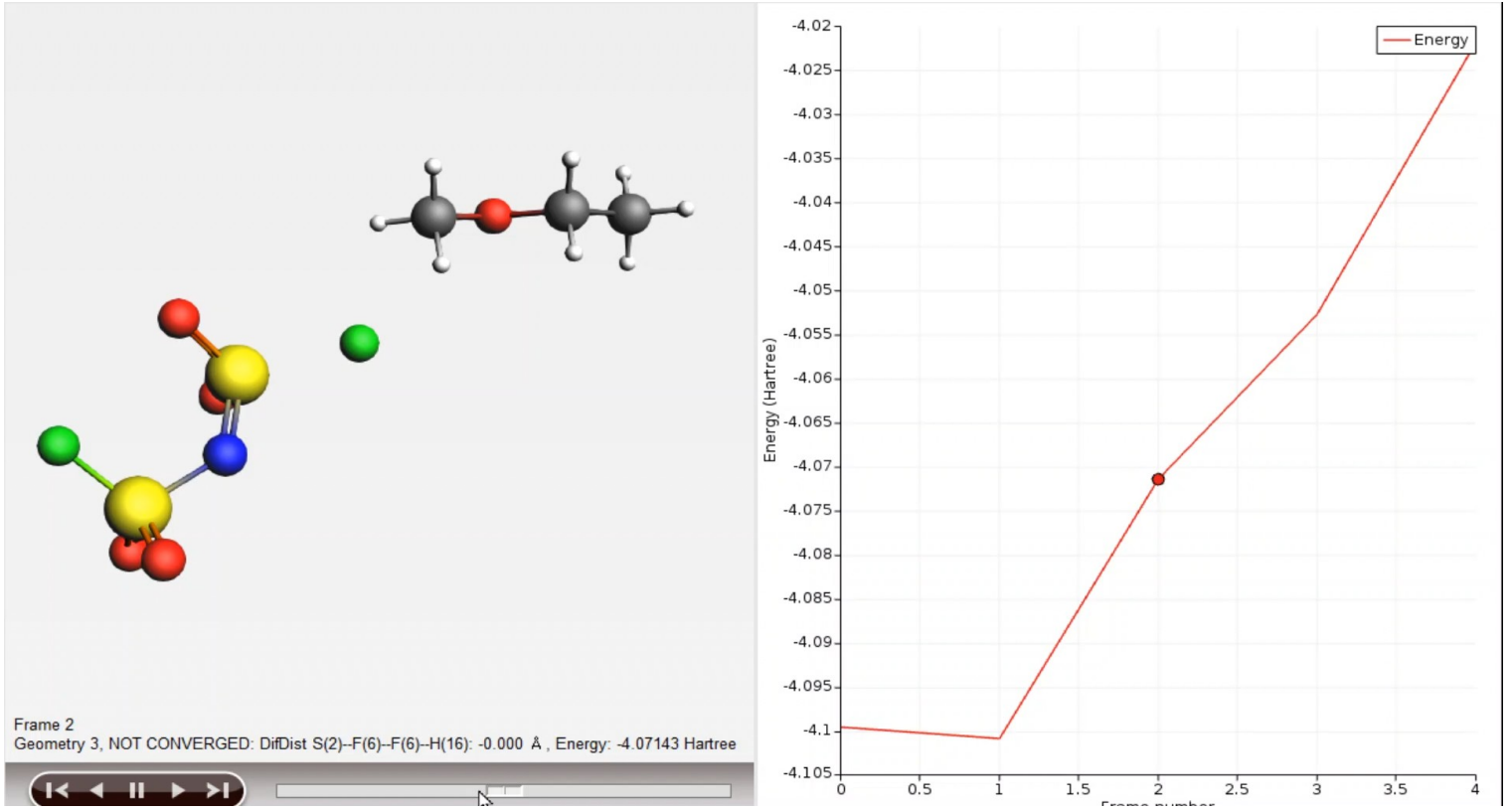
- They are not always accurate enough for “unseen” structures or molecules
- Example from an AMS industry customer: lithium bis(fluorosulfonyl)imide in organic solvent
  - Published ReaxFF force field predicts that a fluorine atom dissociates from the anion



Published ReaxFF force field prediction

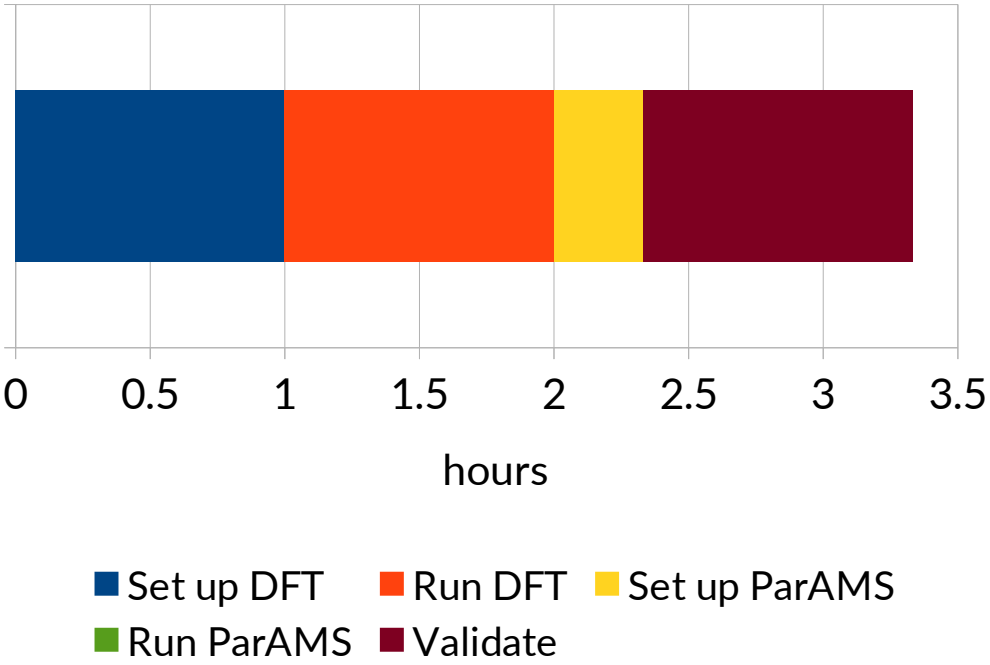


AMS: Easily verify/disprove ReaxFF prediction with DFT



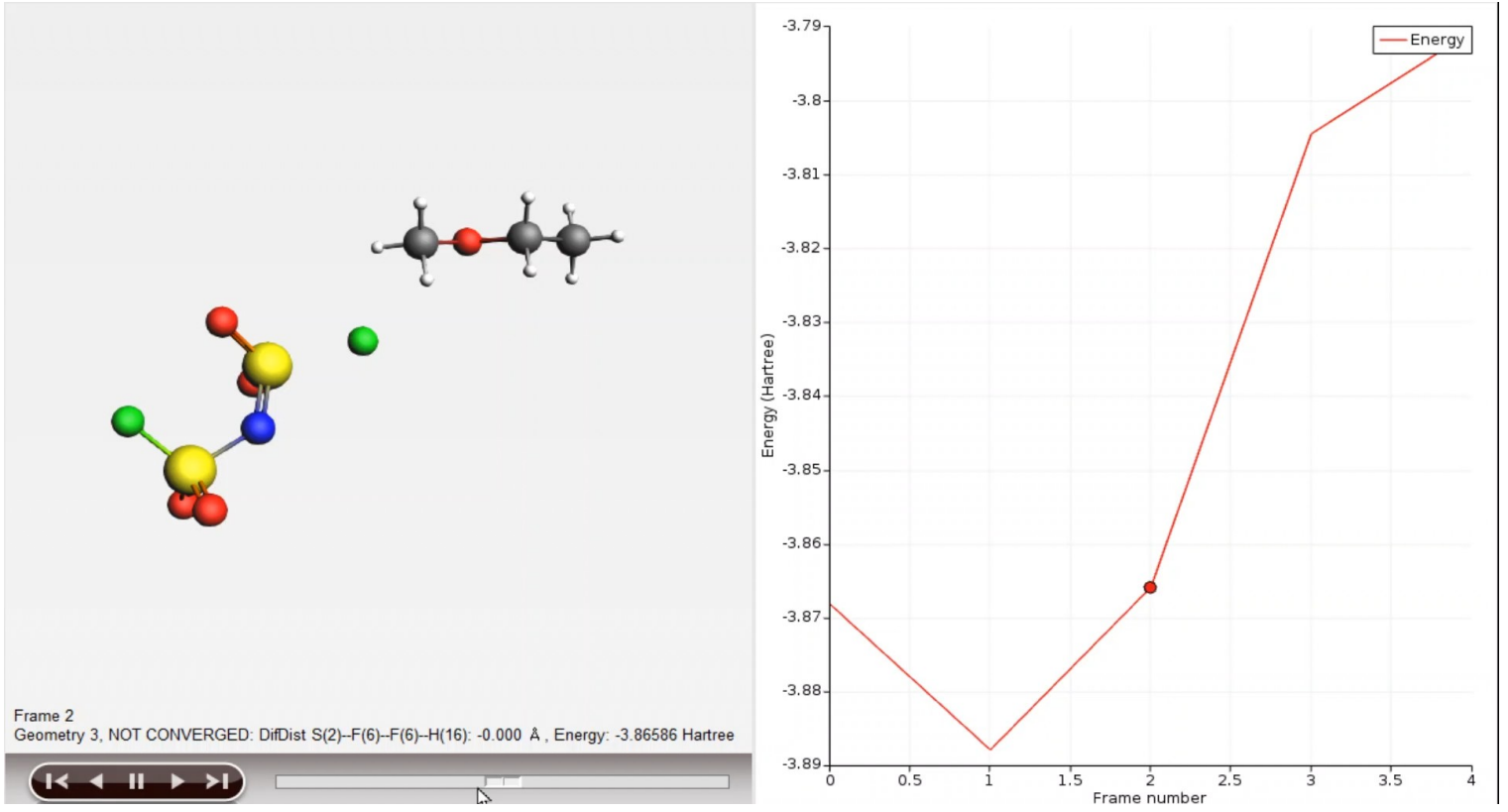
AMS and ParAMS make it *easy* to fix the issue

- For this example:
  - Set up new DFT calculations: 1 hour
  - Run DFT calculations: 1 hour
  - Set up ParAMS: 20 minutes
  - Reoptimize ReaxFF parameters: 10 seconds
  - Validate the new ReaxFF: 1 hour

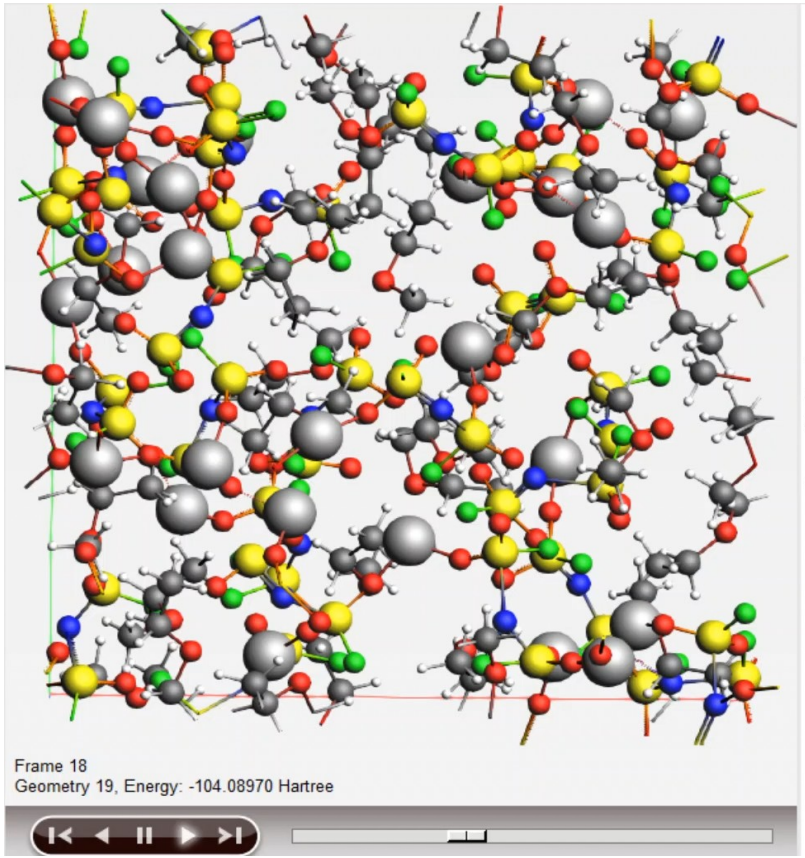


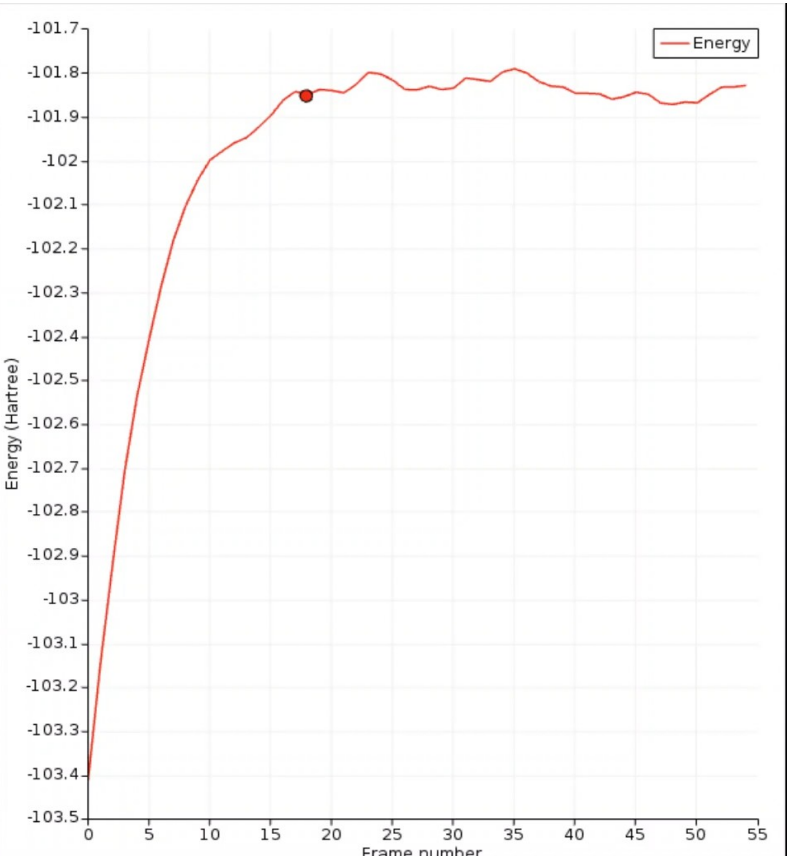
- 0 lines of code! All steps supported by the graphical user interface.
- *Note: Most parametrization projects will take longer*

Reoptimized ReaxFF predictions



Production simulation





General ParAMS workflow

- Validate an initial ReaxFF force field (if it exists)
- Set up new DFT calculations
- Run DFT calculations
- Set up ParAMS
- Run parameter optimization
- Validate new ReaxFF force field

iterative
improvement

ParAMS minimizes the *loss function*

Training set size

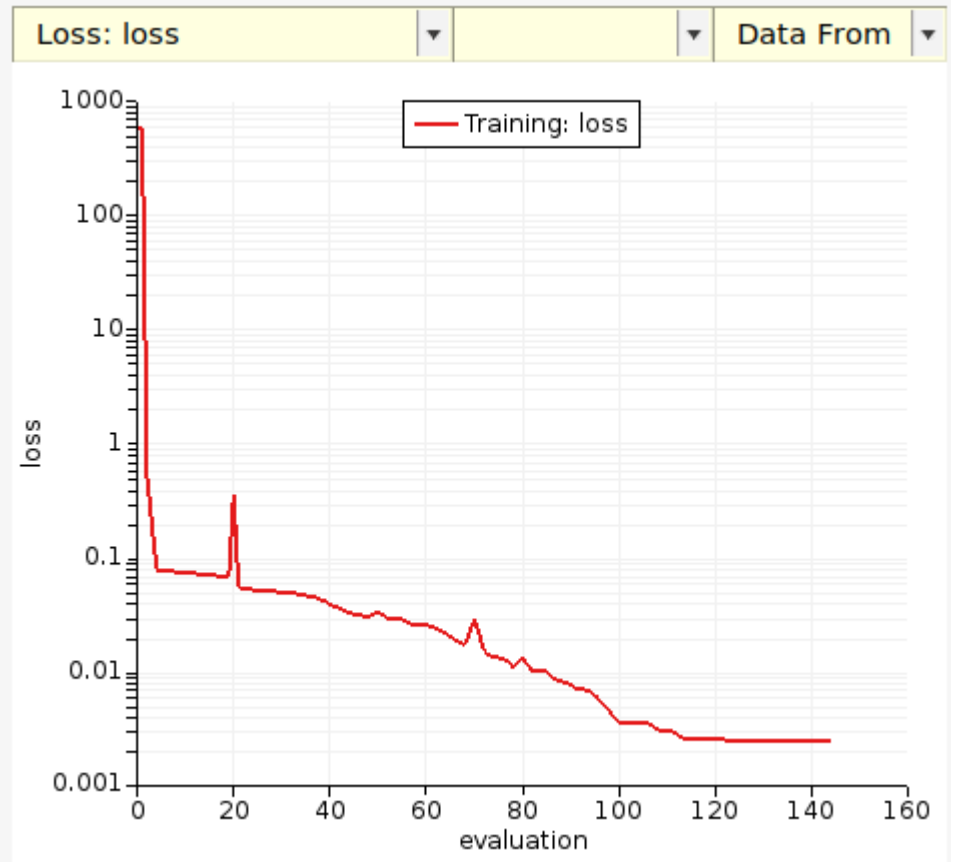
Reference value

Predicted value

$$\sum_{i=1}^{N} w_i \left(\frac{y_i - \hat{y}_i}{\sigma_i} \right)^2$$

Weight

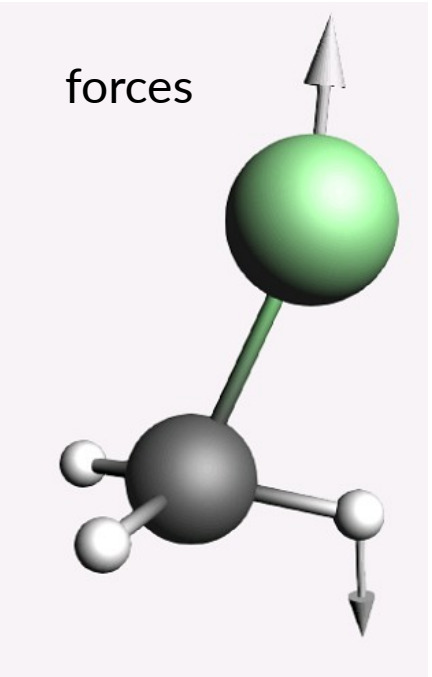
Unit normalization, “accuracy”

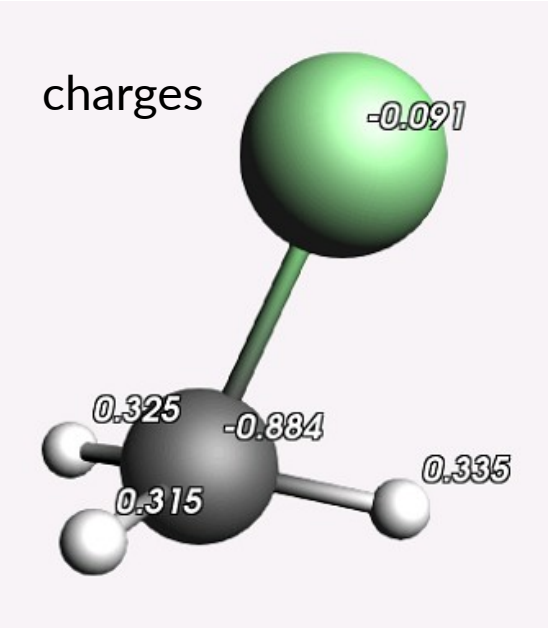


New parameters for every evaluation

ParAMS: Types of reference values

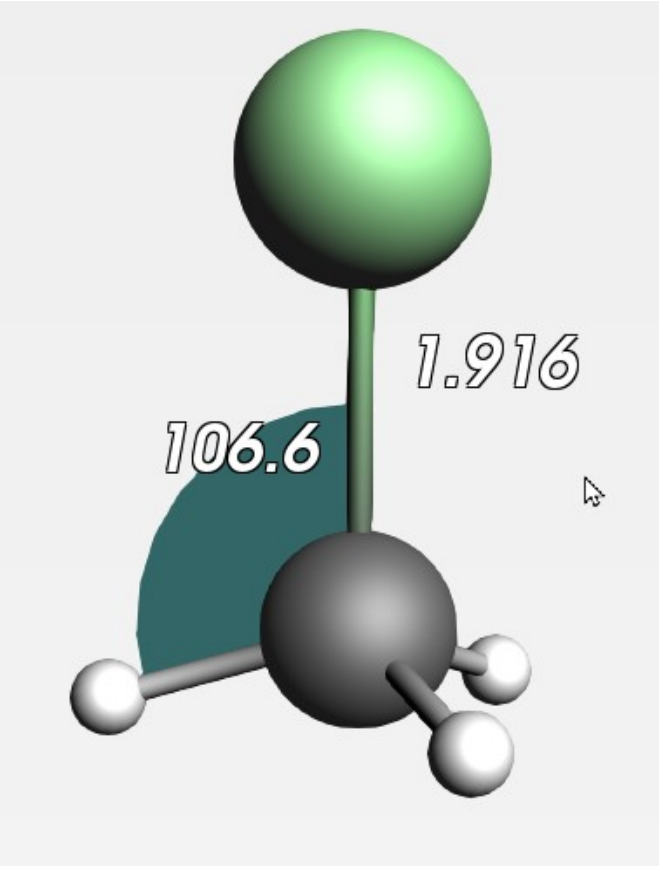
- “Anything” that can be extracted from a **job** can be used as a reference value
  - Forces, atomic charges





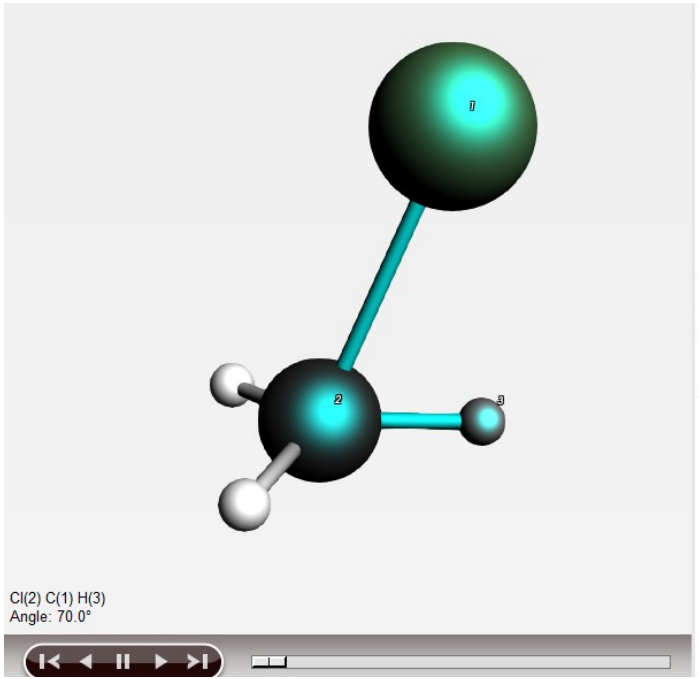
ParAMS: Types of reference values

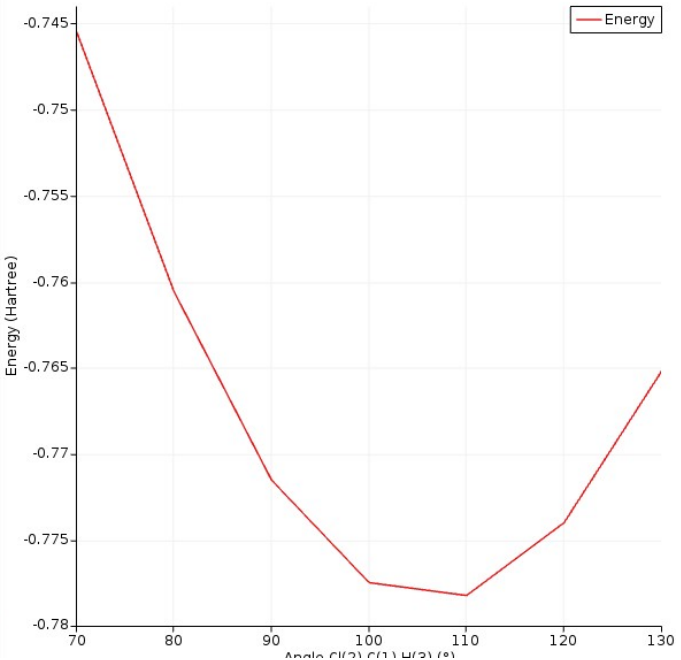
- “Anything” that can be extracted from a **job** can be used as a reference value
  - Optimized bond lengths or angles



ParAMS: Types of reference values

- “Anything” that can be extracted from a **job** can be used as a reference value
  - PES Scans:
    Energy vs.
    bond length,
    angle, or
    cell volume





ParAMS: Types of reference values

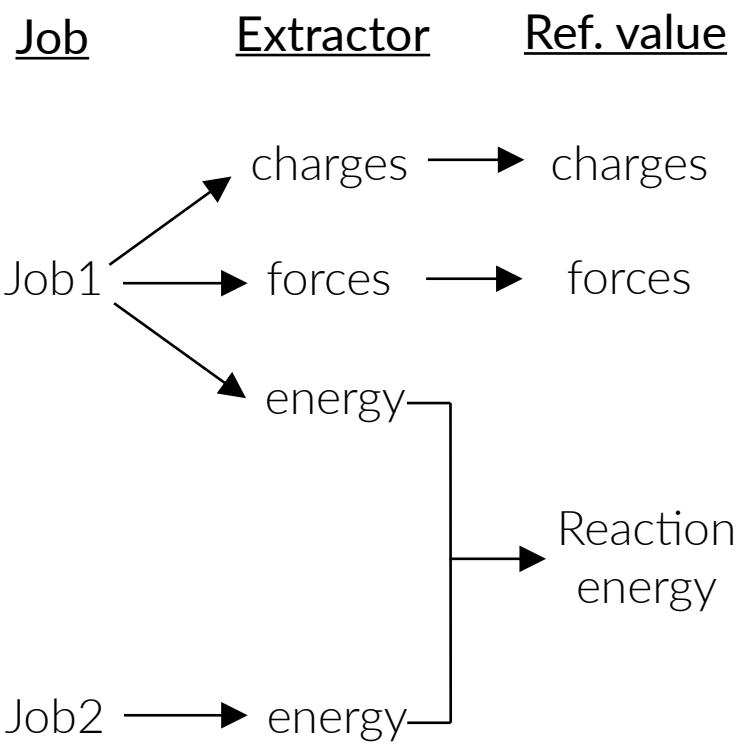
- Values from **multiple jobs**:
  - Reaction energies (adsorption energy, surface energy, formation energy, ...)

$$C_3H_6(g) + (9/2)\ O_2(g) \rightarrow 3\ CO_2(g) + 3\ H_2O(g)$$

$$\Delta H_r^0 = -491.8\ \text{kcal/mol}$$

ParAMS: Types of reference values

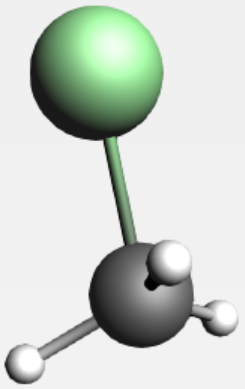
- "Anything" that can be extracted from a **job** can be used as a reference value
  - Forces, atomic charges
  - Optimized bond lengths or angles
  - PES Scans: Energy vs. bond length, angle, or cell volume
- Values from **multiple jobs**:
  - Reaction energies (adsorption energy, surface energy, formation energy, …)



Example reference values: Charges and forces of chloromethane

ParAMS 2022.101

SCM File Edit Jobs Parameters Training Set View Help



Job chloromethane-forces

All | Jobs | Training Set | Validation Set | Engines

Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	Single Point + gradients	chloromethane-forces		adf;;xc;;gga;PBE;	
Charges	chloromethane-forces	chloromethane-forces	1.0	[-0.8835, 0.3345] (5	au
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	
Forces	chloromethane-forces	chloromethane-forces	1.0	[-0.1095, 0.0735] (5	Hartree/Bc

Errors Only 5 items

Parameters | Settings | Info | Graphs | Results

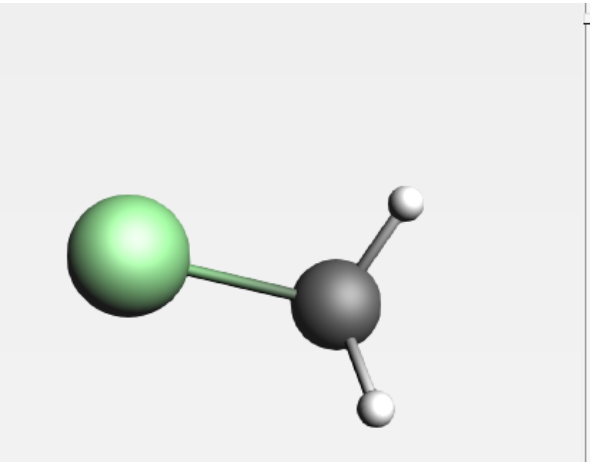
Weight: +1.000000000

Value:

-0.016768241	+0.018544396	-0.018888589
+0.043055409	-0.109488031	-0.015582138
-0.022345467	+0.073456592	+0.026842484
-0.002046089	+0.011289727	+0.017385490
-0.001895612	+0.006197316	-0.009757247

Example reference values: Optimized bond lengths and angles

- For bond lengths and angles, add **geometry optimization** jobs!

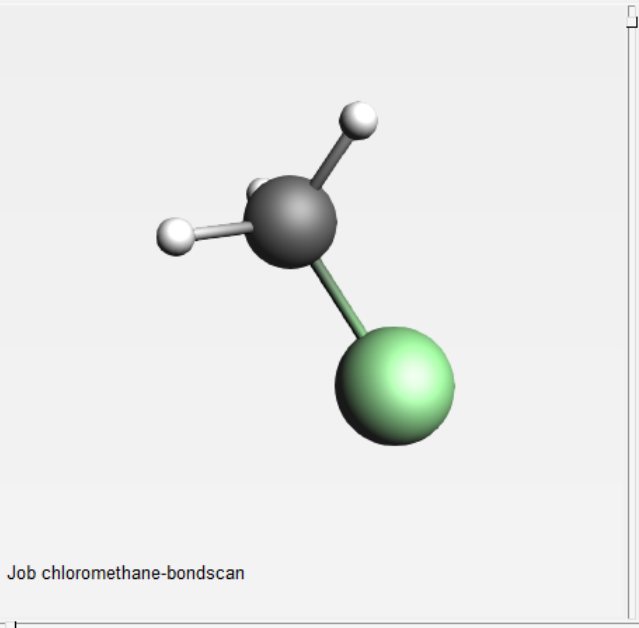


All | Jobs | Training Set | Validation Set | Engines

Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	Geometry Optimization + gradients + ...	chloromethane-geometry		adf;;xc;;gga;PBE;	
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	
Geo: distanc	chloromethane-geometryoptimization, 0, 1 (C-Cl)	chloromethane-geometry	1.0	+1.91567508	Å
Geo: distanc	chloromethane-geometryoptimization, 0, 2 (C-H)	chloromethane-geometry	1.0	+1.09398928	Å
Geo: angle	chloromethane-geometryoptimization, 1, 0, 2 (Cl-C-H)	chloromethane-geometry	1.0	+106.60000000	°
Geo: angle	chloromethane-geometryoptimization, 2, 0, 3 (H-C-H)	chloromethane-geometry	1.0	+112.40000000	°

Example reference values: Bond scan, angle scan, lattice scan

- For PES scans, add **PES Scan** jobs!



Job chloromethane-bondscan

All | Jobs | Training Set | Validation Set | Engines

Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	PES Scan + gradients + ...	chloromethane-bondscar		adf;;xc;;gga;PBE;	
PES	chloromethane-bondscan, relative_to=3	chloromethane-bondscar	1.0	[0.0000, 9.6074] (7)	kcal/mol
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	

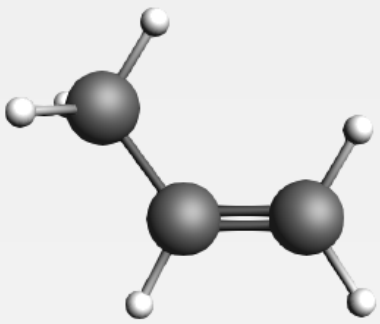
Errors Only 4 items

Parameters | Settings | Info | Graphs | Results

```
Weight:         +1.000000000
Value:          +9.607422193    +4.075588043    +1.082804509    +0.000000000    +0.348957323
                +1.763552064    +3.960264837
```

Example reference values: Reaction energy

- Propene combustion: $C_3H_6(g) + (9/2)\ O_2(g) \rightarrow 3\ CO_2(g) + 3\ H_2O(g)$
- Automatically balanced stoichiometric coefficients!

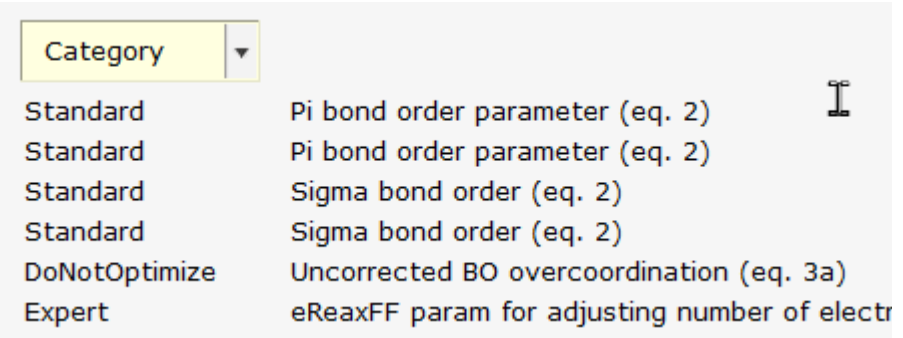


All | Jobs | Training Set | Validation Set | Engines

Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	Geometry Optimization + ...	water		adf;;xc;;gga;PBE;	
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	
Job	Geometry Optimization + ...	propene		adf;;xc;;gga;PBE;	
Job	Geometry Optimization + ...	co2		adf;;xc;;gga;PBE;	
Job	Geometry Optimization + ...	o2		adf;;spinpolarization;2;unr	
Engine	Engine adf spinpolarization 2 unrestricted yes xc gga PBE			adf;;spinpolarization;2;unr	
Energy	+3.0*co2+3.0*water-1.0*propene-4.5*o2	propene ...	1.0	-358.85844881	kcal/mol

ParAMS settings

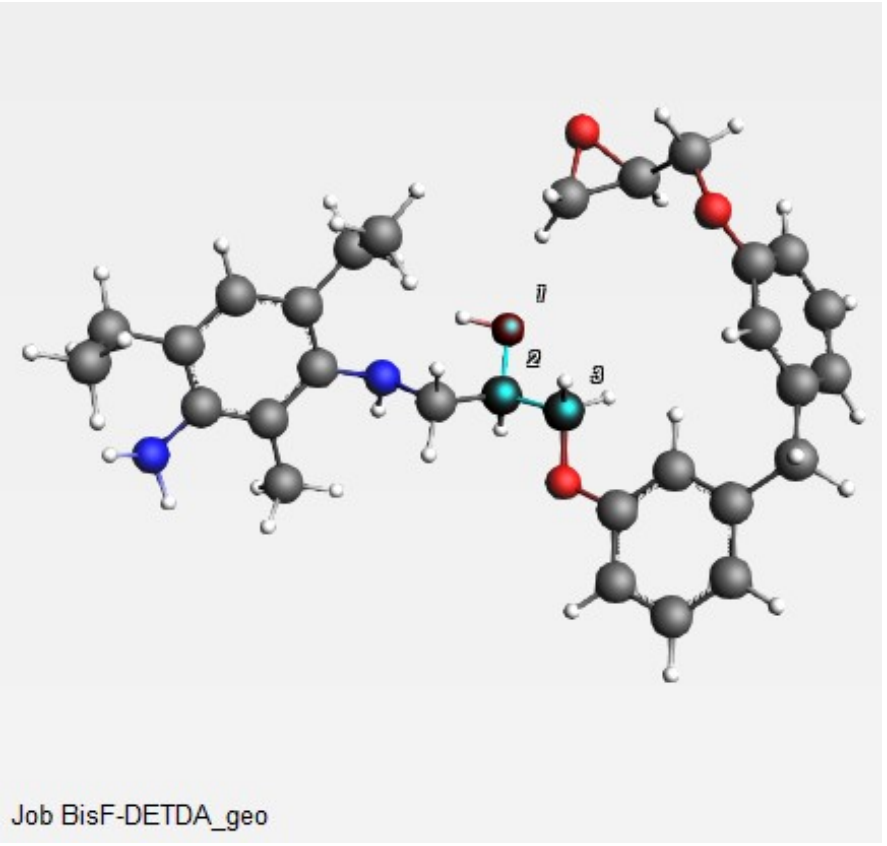
- Which parameters to optimize?
  - Choose as **few** as possible
  - First try parameters in the "**standard**" category
  - If a parameter value is close to min/max, change the range and continue



- Which optimization algorithm?
  - We recommend **CMA-ES** for most optimization problems
  - Details about this algorithm in previous seminar: https://youtu.be/Icv7kWUaoTI

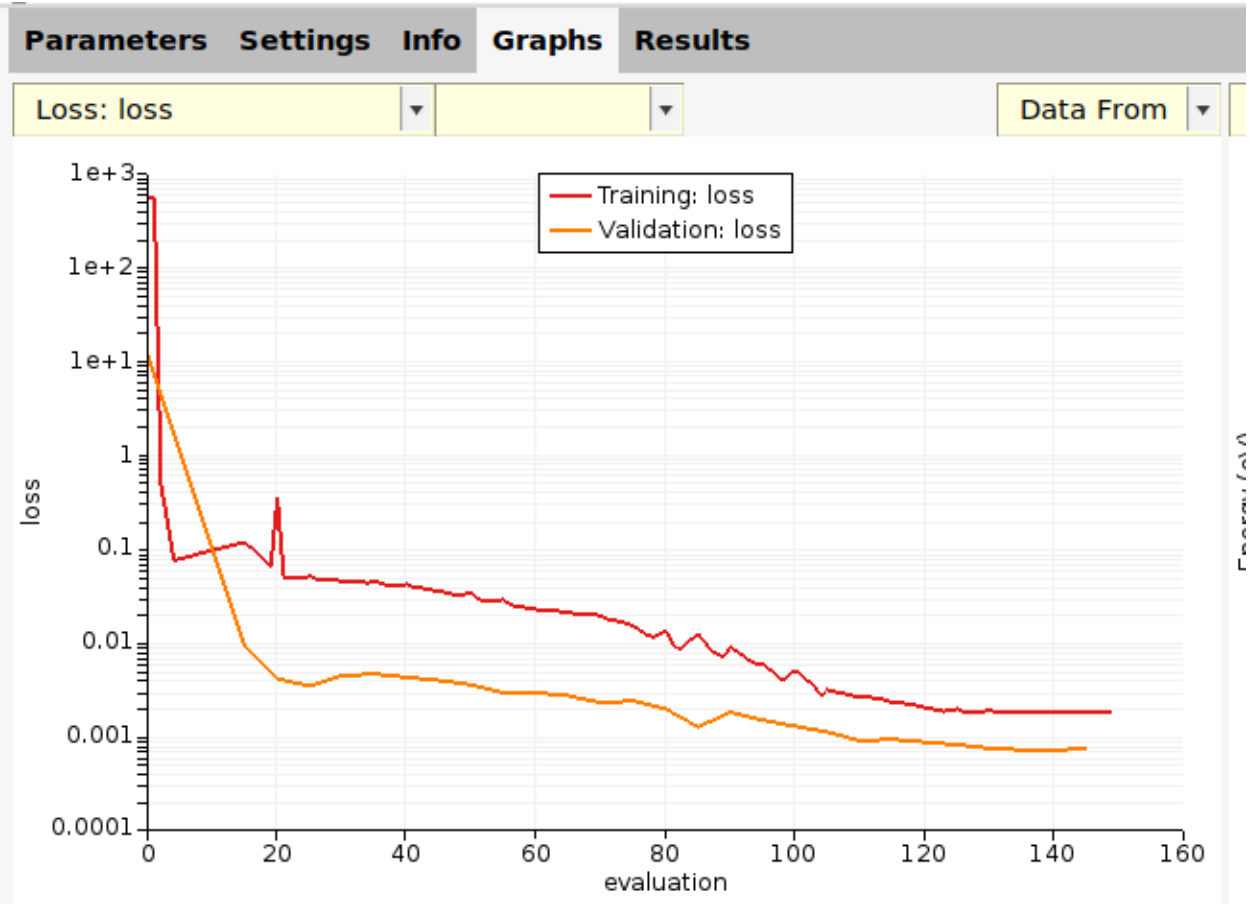
The ParAMS challenge 2022

- Fit and submit a ReaxFF force field
- Hundreds of euros in **prizes**
- Reasonable input has already been prepared:
  - "Open and run", or
  - Try to adjust some settings to get an even better force field
- **Deadline**: 1 July 2022
- https://www.scm.com/news/params-reaxff-parametrization-challenge/

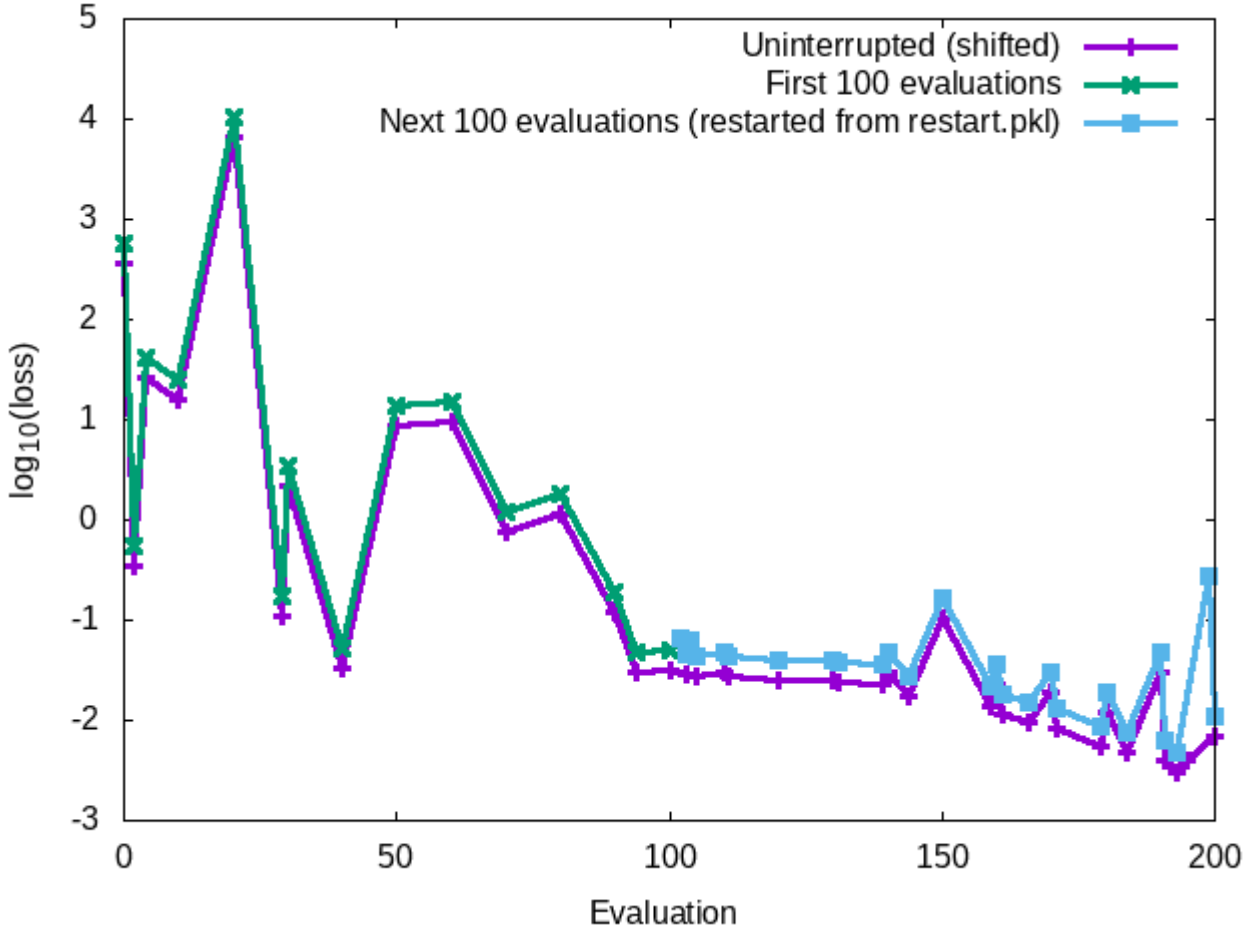


12 bonus features

#1: Validation sets



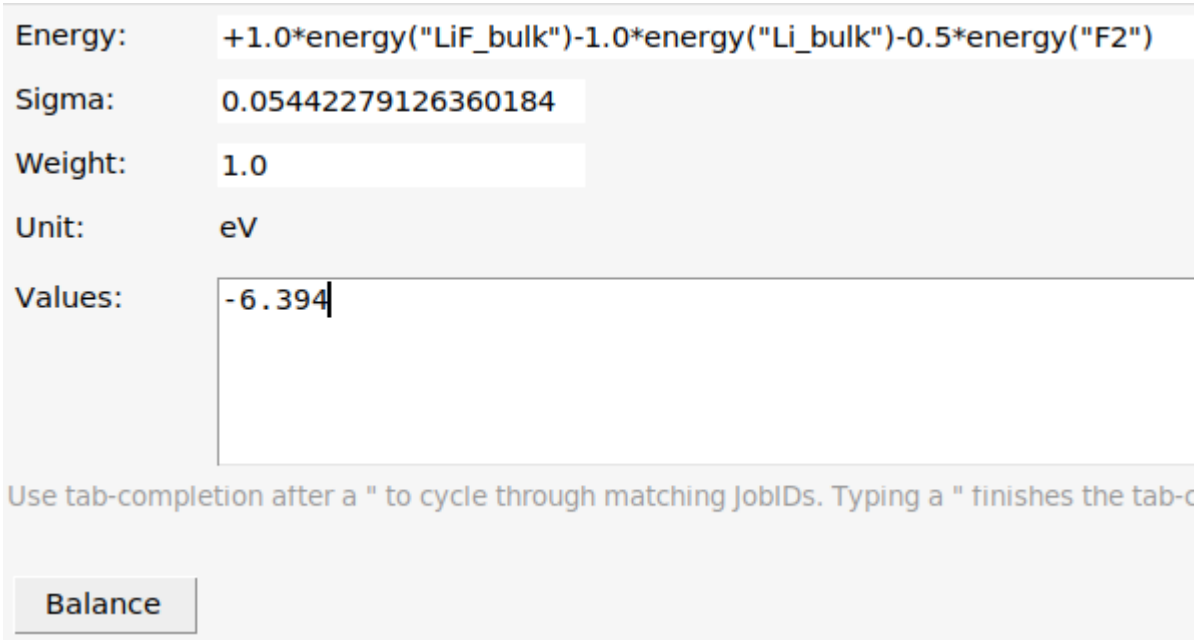
#2: Restart (continue) an optimization



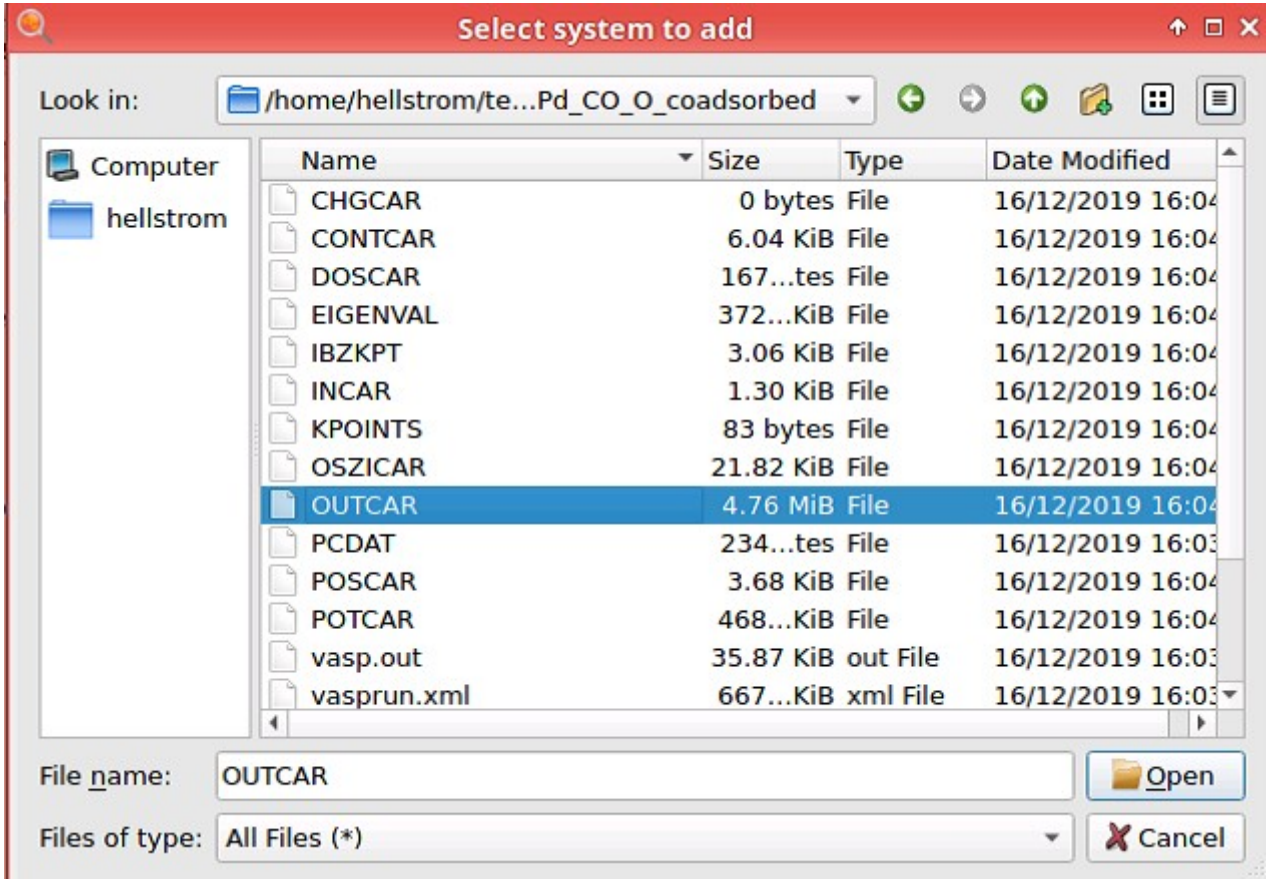
#3: The *params* Python library

```
ri.add_singlejob(water_ams_results_file,         # path to ams.rkf
                 properties={
                     'distance(0,1)': {  # extract O-H bond length
                         'weight': 2.0,
                         'sigma': 0.1,
                      },
                      'angle(1,0,2)': {}, #  H-O-H angle, use default weight and sigma
                 },
                 task='GeometryOptimization',    # set task to GeometryOptimization
                 name='water_geo_opt')           # set a custom name
```

#4: Experimental reference data



#5: Import VASP or Quantum ESPRESSO calculations



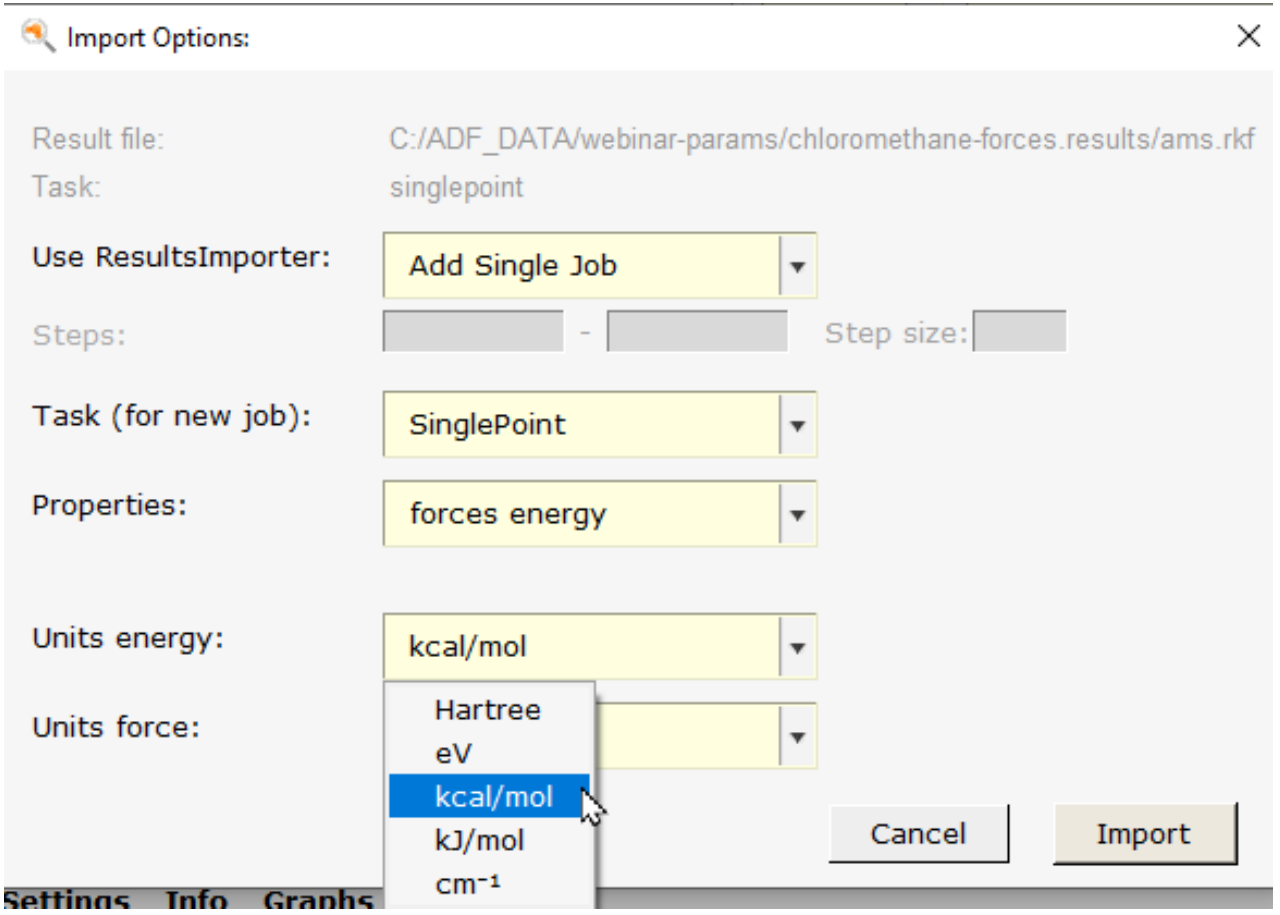
#6: 14 GUI and Python tutorials

- 2. Tutorials
  - 2.1. Introduction to parametrization
  - 2.2. Getting Started: Lennard-Jones Potential for Argon
  - 2.3. Import training data (GUI)
  - 2.4. Import training data (Python)
  - 2.5. ReaxFF: Gaseous H_2O
  - 2.6. ReaxFF: Adsorption on ZnS(110)
  - 2.7. ReaxFF: Convert old training sets to ParAMS format
  - 2.8. ReaxFF: Training set for cobalt
  - 2.9. GFN1-xTB: Lithium fluoride
  - 2.10. Training and validation sets
  - 2.11. Restarting (continuing) an optimization
  - 2.12. Calculate reference values with ParAMS
  - 2.13. The params Python library
  - 2.14. DFTB repulsive potential

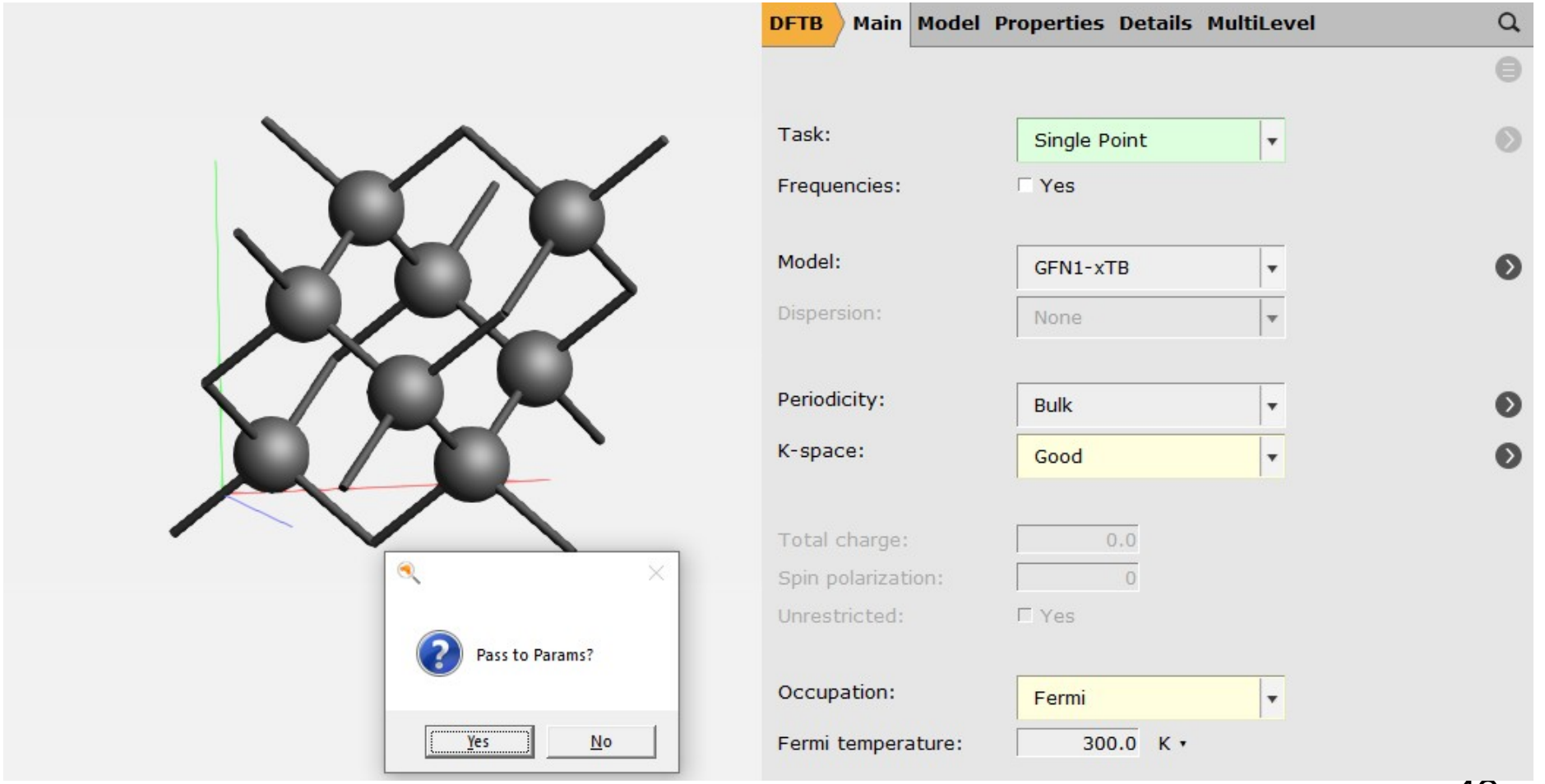
- **2.2. Getting Started: Lennard-Jones Potential for Argon**
  - 2.2.1. Lennard-Jones Parameters, Engine, and Interface
  - 2.2.2. Files
  - 2.2.3. ParAMS input
    - 2.2.3.1. Parameter interface (parameter_interface.yaml)
    - 2.2.3.2. Job Collection (job_collection.yaml)
    - 2.2.3.3. Training Set (training_set.yaml)
    - 2.2.3.4. ParAMS settings (params.conf.py)
  - 2.2.4. Run the example
  - 2.2.5. Parametrization results
    - 2.2.5.1. The best parameter values
    - 2.2.5.2. Correlation plots
    - 2.2.5.3. Error plots
    - 2.2.5.4. Parameter plots
    - 2.2.5.5. Editing and Saving Plots
    - 2.2.5.6. Predicted values
    - 2.2.5.7. Loss contributions
    - 2.2.5.8. Summary statistics
    - 2.2.5.9. All output files

https://www.scm.com/doc/params/examples/examples.html

#7: Work in your preferred units



#8: Modify job settings with AMSinput

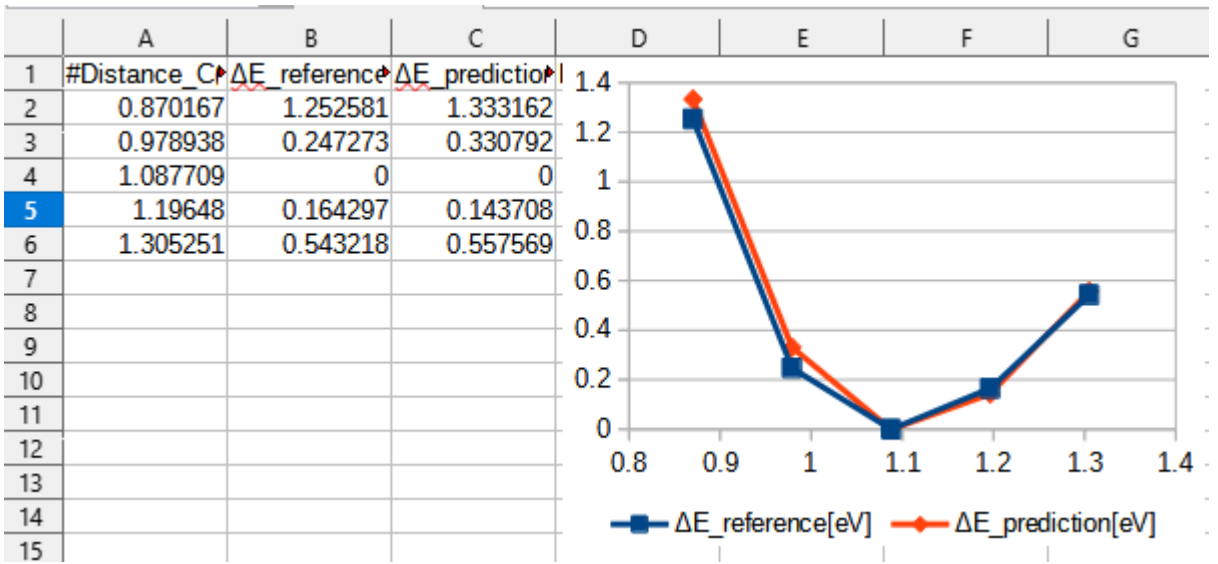


#9: Human-readable input in .yaml format

```
---
dtype: DataSet
version: '2022.101'
---
Expression: charges('water_singlepoint')
Weight: 1.0
Sigma: 0.1
ReferenceValue: |
   array([-0.83399999,  0.417     ,  0.417     ])
Unit: au, 1.0
---
Expression: forces('water_singlepoint')
Weight: 1.0
Sigma: 3.557463138142747
ReferenceValue: |
   array([[ 0.25021751, -0.01895832, -0.        ],
          [-0.03370832, -0.12831032, -0.        ],
          [-0.21650919,  0.14726863, -0.        ]])
Unit: kcal/mol/angstrom, 1185.8210460475823
...
```

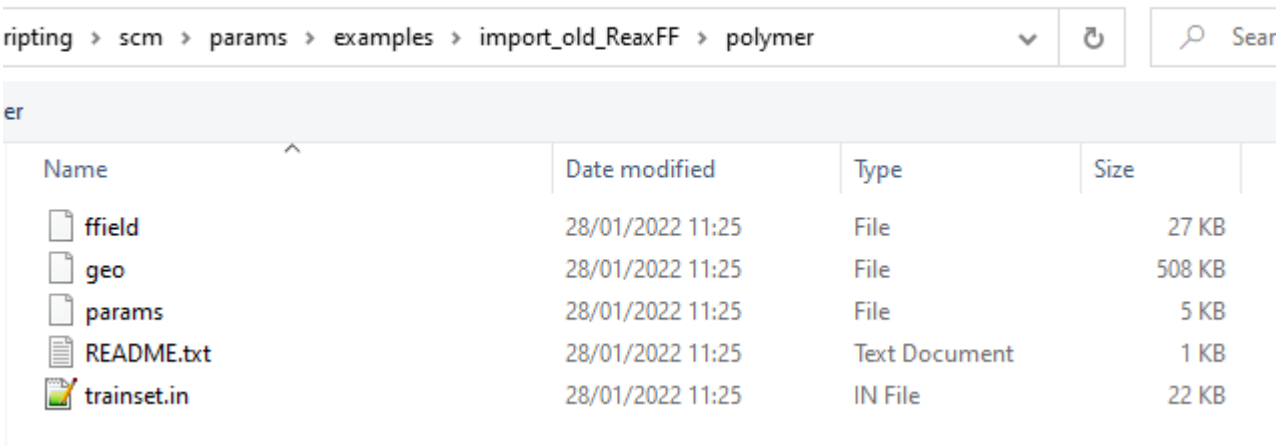
#10: Human-readable output: import directly to Excel, gnuplot, ...

```
#Distance_C(1)--H(5)[angstrom] ΔE_reference[eV] ΔE_prediction[eV] E_reference[eV]
0.870167                       1.252581         1.333162          -53.399365
0.978938                       0.247273         0.330792          -54.404673
1.087709                       0.000000         0.000000          -54.651946
1.196480                       0.164297         0.143708          -54.487649
1.305251                       0.543218         0.557569          -54.108729
```



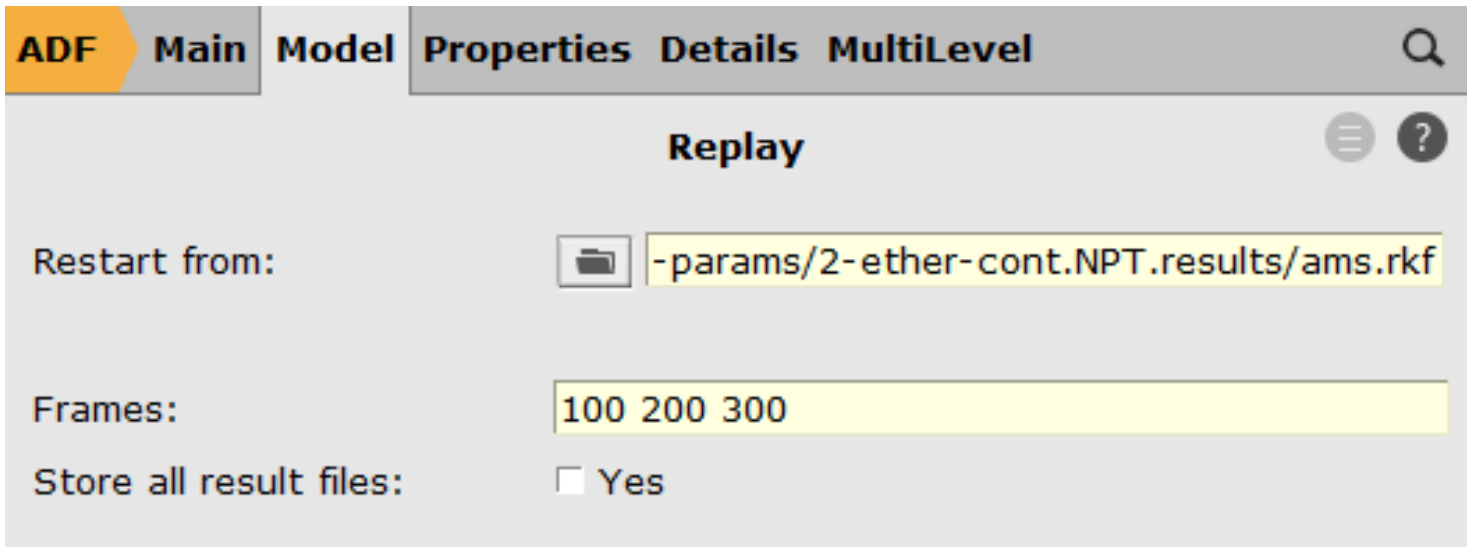
#11: Convert from old ReaxFF training set format (<AMS2022) to ParAMS

Old File	+	(Optional File)	=	New File
`geo`		`control`		`job_collection.yaml`
`trainset.in`				`training_set.yaml`
`ffield`		`params`		`parameter_interface.yaml`



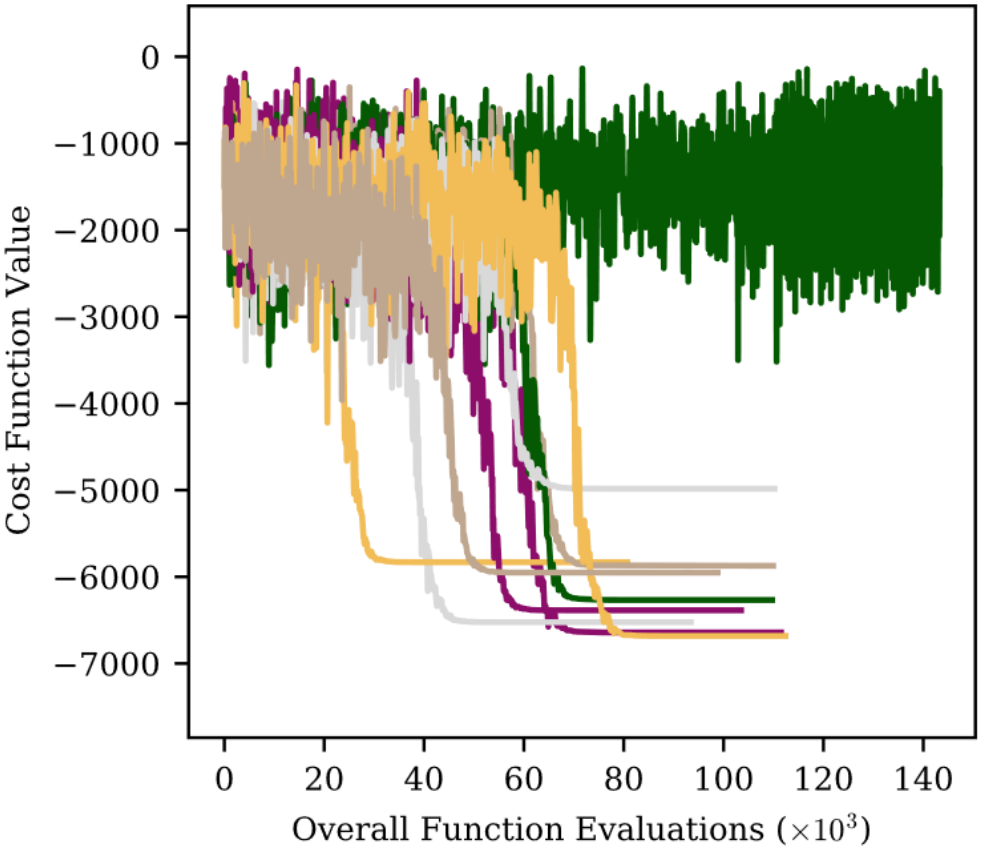
#12: Recalculate a trajectory with a different method

- Recalculate all or some frames from molecular dynamics, PES scans, …
- Compare results or import as reference data into ParAMS



Future developments: AMS2023?

- Improvements to the input
- Improvements to the graphical user interface
- Run multiple optimizations at the same time with GloMPO



Freitas Gustavo and Verstraelen *Journal of Cheminformatics* (2022) 14:7

Summary

- ParAMS and AMS make it *easy* to create and validate
  - ReaxFF force fields
  - DFTB (actually GFN1-xTB) parameters
- Train to reaction energies, forces, charges, bond scans, …
- Results shown in many different graphs and tables
  - Updated on-the-fly, even on a remote machine!
- Use AMS to address your research problems
- Participate in the ParAMS challenge!