ParAMS Workshop 2024

Welcome

- Fedor Goumans, SCM / Chief Customer Officer, goumans@scm.com
- Paul Spiering, SCM / Software Developer, spiering@scm.com

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

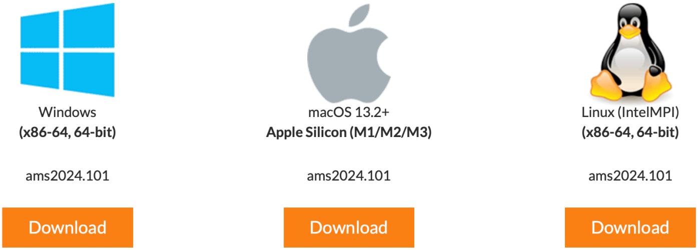
15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

Get ready to run AMS

Download and install AMS2024

https://www.scm.com/support/downloads/



Windows
(x86-64, 64-bit)

ams2024.101

macOS 13.2+
Apple Silicon (M1/M2/M3)

ams2024.101

Linux (IntelMPI)
(x86-64, 64-bit)

ams2024.101

Comments

- We will teach you how to use the software but don't expect to have a production Force Field by the end of the day
- We will leave time for questions at the end of each presentations
- If you think your question is important and should be answered during the presentation you should ask it
- During [Demo], we will demonstrate how to perform various tasks with the software
- During [Demo], you can also try to reproduce some tasks but we recommend you follow the demonstration in priority
- During [Hands-on], we will ask you to perform some tasks with the software
- During [Hands-on], we will demonstrate the solution

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- **General introduction of AMS (30 min)**
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

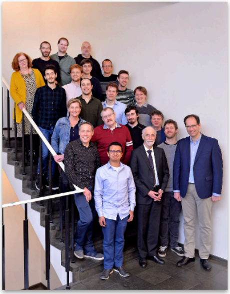
- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

Software for Chemistry & Materials

History

VU VRIJE UNIVERSITEIT AMSTERDAM

Baerends@VU & Ziegler@Calgary



1970s

ADF = first DFT code for chemists
Study transition metal complexes
for catalysis

SCM: Spin-off company
(VU Amsterdam)

1995

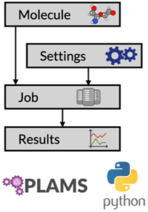
Mitsui, Shell, Akzo, Unilever
Training in Amsterdam to optimize
catalysts

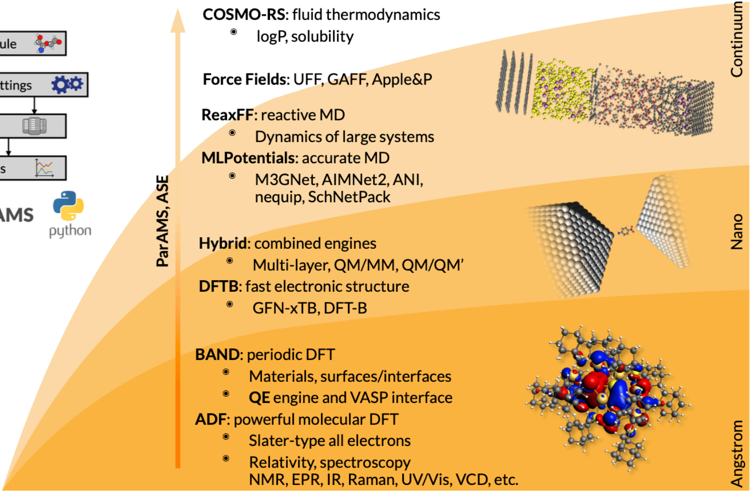
2024

30 peoples (15 senior PhD's)
~ 15 developers
Implement new features, debug, optimize, docs & support
+ Many academic collaborators

The Amsterdam Modeling Suite

Multiple engines to bridge the scales





COSMO-RS: fluid thermodynamics

- logP, solubility

Force Fields: UFF, GAFF, Apple&P

ReaxFF: reactive MD

- Dynamics of large systems

MLPotentials: accurate MD

- M3GNet, AIMNet2, ANI, nequip, SchNetPack

Hybrid: combined engines

- Multi-layer, QM/MM, QM/QM'

DFTB: fast electronic structure

- GFN-xTB, DFT-B

BAND: periodic DFT

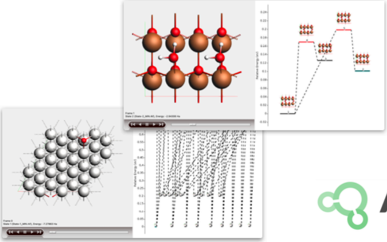
- Materials, surfaces/interfaces
- **QE** engine and VASP interface

ADF: powerful molecular DFT

- Slater-type all electrons
- Relativity, spectroscopy
  NMR, EPR, IR, Raman, UV/Vis, VCD, etc.

The Amsterdam Modeling Suite

A unified driver to explore PES



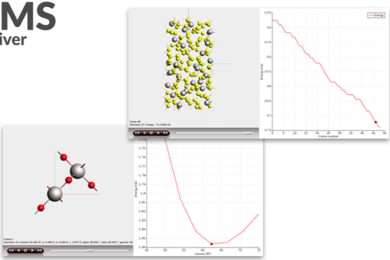
AMS driver

Tasks

- Molecular dynamics
- Frequencies & phonons
- Stress & elastic tensors
- Scan coordinates & constraints
- Monte Carlo & GCMC, etc.

Engines

- ADF
- BAND
- DFTB
- ReaxFF
- MLP

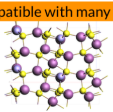


Applications

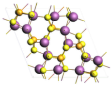
Battery modeling

- Diffusion coefficients
- Diffusion path
- Activation energies

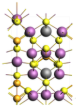
Compatible with many file formats



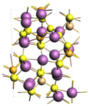
Li_4GeS_4



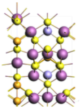
$Li_7P_3S_{11}$



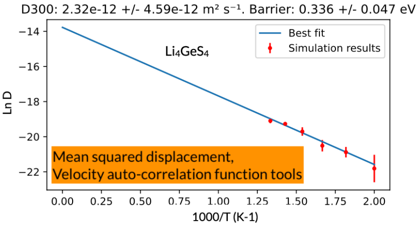
$Li_{10}SnP_2S_{12}$



$Li_{10}GeP_2S_{12}$

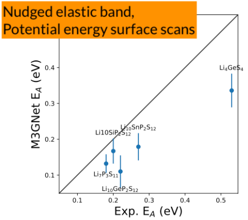


$Li_{10}SiP_2S_{12}$



Mean squared displacement,
Velocity auto-correlation function tools

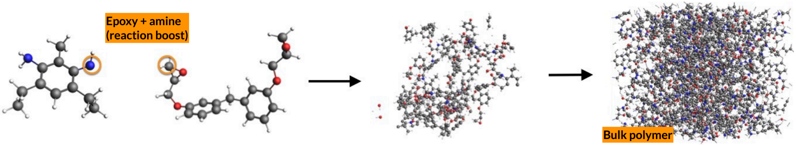
Nudged elastic band,
Potential energy surface scans



Applications

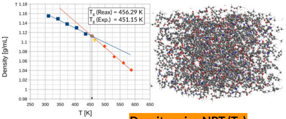
Polymer modeling

- Polymer structure ⇔ properties
- Side reactions
- Mechanical properties

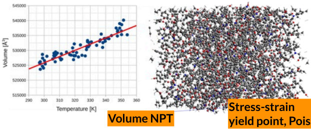


Epoxy + amine (reaction boost)

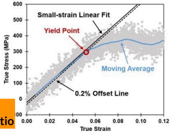
Bulk polymer



Density using NPT (Tg)



Volume NPT

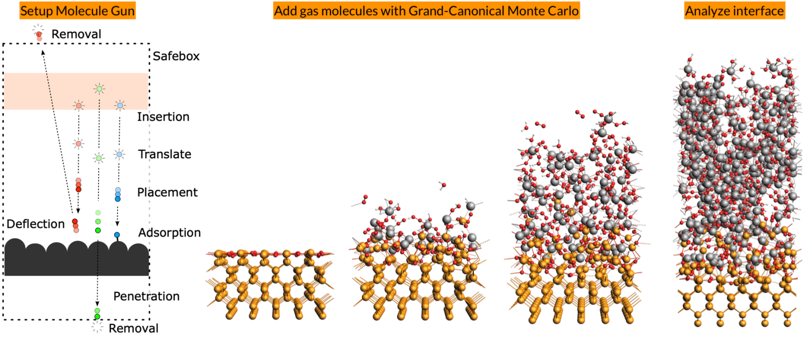


Stress-strain
yield point, Poisson ratio

Applications

Semiconductor modeling

- Understand the formation of defects
- Reactive interfaces
- Diffusion, oxidation

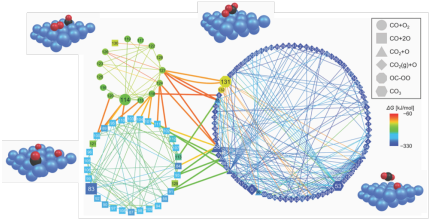


Applications

Catalysis

- Understand heterogeneous catalysis
- Binding sites energies
- Transition states energies and structures

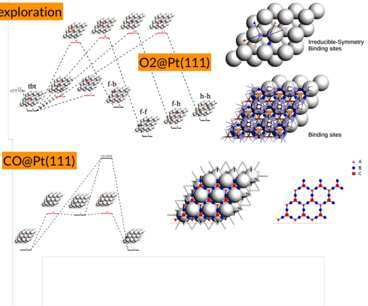
The reaction path network for CO oxidation on Pt(111)



PES exploration

O2@Pt(111)

CO@Pt(111)

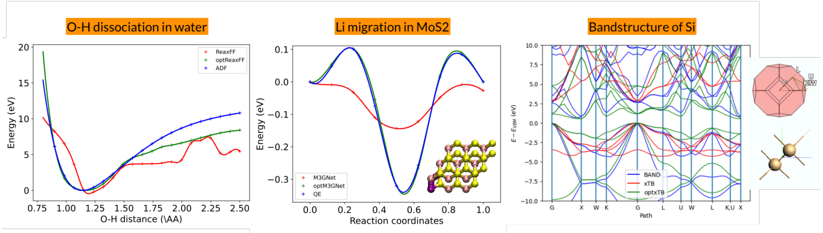


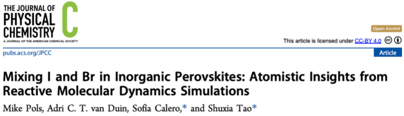
Sugiyama, Kanami, et al. PCCP 21.26 (2019): 14366-14375

Applications

Summary

- All this is great with accurate model potentials..
- Below some examples of model improved via training & active learning with ParAMS

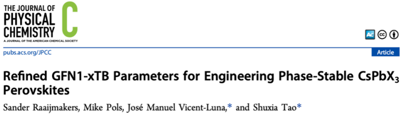



THE JOURNAL OF PHYSICAL CHEMISTRY C
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY
pubs.acs.org/JPCC

Article

Mixing I and Br in Inorganic Perovskites: Atomistic Insights from Reactive Molecular Dynamics Simulations

Mike Pols, Adri C. T. van Duin, Sofía Calero,* and Shuxia Tao*


THE JOURNAL OF PHYSICAL CHEMISTRY C
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY
pubs.acs.org/JPCC

Article

Refined GFN1-xTB Parameters for Engineering Phase-Stable $CsPbX_3$ Perovskites

Sander Raaijmakers, Mike Pols, José Manuel Vicent-Luna,* and Shuxia Tao*

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- **The graphical user interface [Demo] (30 min)**

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

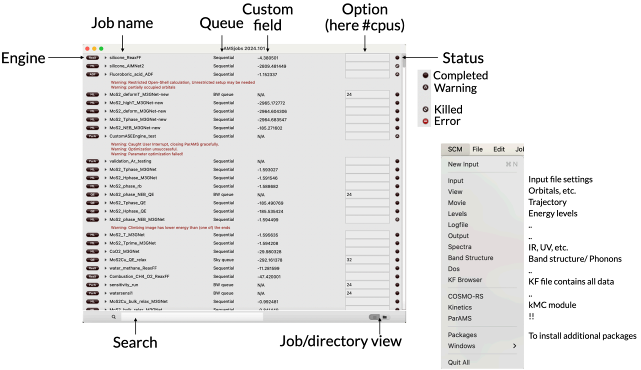
15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

The GUI

AMSjobs

- Manage jobs, set queues, number of processors, monitor job status, etc.



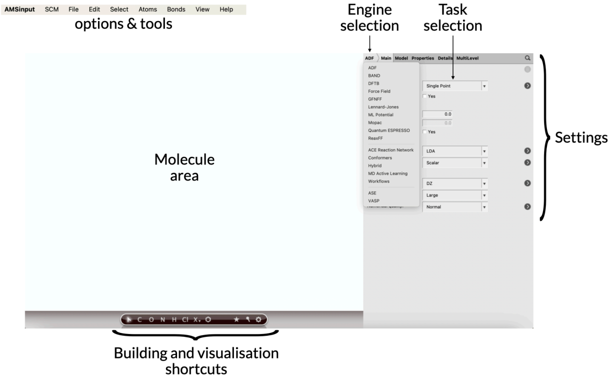
Engine	Job name	Queue	Custom field	Option (here #cpus)	Status
ReaX	silicone_ReaxFF	Sequential	-4.380501		●
ML	silicone_AIMNet2	Sequential	-2809.481449		⊘
ADF	Fluoroboric_acid_ADF	Sequential	-1.152337		⚠
	Warning: Restricted Open-Shell calculation, Unrestricted setup may be needed Warning: partially occupied orbitals				
ML	MoS2_deformT_M3GNet-new	BW queue	N/A	24	●
ML	MoS2_highT_M3GNet-new	Sequential	-2965.172772		●
ML	MoS2_deform_M3GNet-new	Sequential	-2964.604306		●
ML	MoS2_Tphase_M3GNet-new	Sequential	-2964.683547		●
ML	MoS2_NEB_M3GNet-new	Sequential	-185.271602		●
ParA	CustomASEEngine_test	Sequential	N/A		⚠
	Warning: Caught User Interrupt, closing ParAMS gracefully. Warning: Optimization unsuccessful. Warning: Parameter optimization failed!				
ParA	validation_Ar_testing	Sequential	N/A		●
ML	MoS2_Tphase_M3GNet	Sequential	-1.593027		●
ML	MoS2_Hphase_M3GNet	Sequential	-1.591546		●
ML	MoS2_phase_rb	Sequential	-1.588682		●
QE	MoS2_phase_NEB_QE	BW queue	N/A	24	●
QE	MoS2_Tphase_QE	Sequential	-185.490769		●
QE	MoS2_Hphase_QE	Sequential	-185.535424		●
ML	MoS2_phase_NEB_M3GNet	Sequential	-1.594499		⚠
	Warning: Climbing image has lower energy than (one of) the ends				
ML	MoS2_T_M3GNet	Sequential	-1.595635		●
ML	MoS2_Tprime_M3GNet	Sequential	-1.594208		●
ML	CoO2_M3GNet	Sequential	-29.980328		●
QE	MoS2Cu_QE_relax	Sky queue	-292.161378	32	●
ReaX	water_methane_ReaxFF	Sequential	-11.281599		●
ReaX	Combustion_CH4_O2_ReaxFF	Sequential	-47.420001		●
ParA	sensitivity_run	BW queue	N/A	24	●
ParA	watersensi1	BW queue	N/A	24	●
ML	MoS2Cu_bulk_relax_M3GNet	Sequential	-0.992481		●
ML	MoS2_bulk_relax_M3GNet	Sequential	-0.841449		●

Input	Input file settings
View	Orbitals, etc.
Movie	Trajectory
Levels	Energy levels
Logfile	..
Output	..
Spectra	IR, UV, etc.
Band Structure	Band structure/ Phonons
Dos	..
KF Browser	KF file contains all data
COSMO-RS	..
Kinetics	kMC module
ParAMS	!!
Packages	To install additional packages

The GUI

AMSinput

- Build molecule, crystals, select engine, task, settings, etc.



options & tools

Engine selection

Task selection

Settings

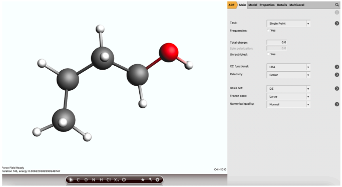
Molecule area

Building and visualisation shortcuts

AMSinput

Build a molecule

- Build 1-butanol, add a ring
- Pre-optimize the molecule
- Rotate molecule, move it, Zoom-in/out
- Orient it within xy, xz, yz plane
- Vary distance, angle, dihedral



Task: Single Point

Frequencies: □ Yes

Total charge: 0.0

Spin polarization: 0.0

Unrestricted: □ Yes

XC functional: LDA

Relativity: Scalar

Basis set: DZ

Frozen core: Large

Numerical quality: Normal

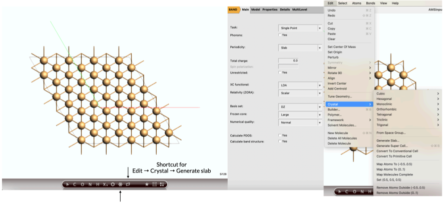
Force Field Ready
Iteration 145, energy 0.006223382893949747

C4 H10 O

AMSinput

Build a crystal

- Search for NaCl, convert to conventional cell, show the unit cell
- Clear the area, Edit → Crystal → Cubic → NaCl
- Clear, Edit → Crystal → Hexagonal → MoS2 → WTe2
- Clear, Edit → Crystal → Cubic → Diamond, Si
- Generate slab, 111, 4 layers, generate supercell 4×4
- Explore periodic views, and more on crystals and slabs



Task: Single Point

Phonons: □ Yes

Periodicity: Slab

Total charge: 0.0

Spin polarization:

Unrestricted: □ Yes

XC functional: LDA

Relativity (ZORA): Scalar

Basis set: DZ

Frozen core: Large

Numerical quality: Normal

Calculate PDOS: □ Yes

Calculate band structure: □ Yes

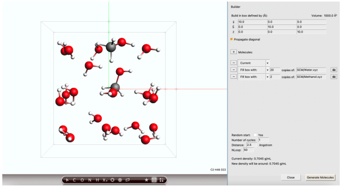
Shortcut for
Edit → Crystal → Generate slab

Shortcut for
Edit → Crystal

AMSinput

Build a liquid mixture

- Edit → Builder..
- Fill box with 20 water, check density
- Add 2 methanol, check density
- Generate molecule..



Builder

Build in box defined by (Å): Volume: 1000.0 Å³

a	10.0	0.0	0.0
b	0.0	10.0	0.0
c	0.0	0.0	10.0

Propagate diagonal

Molecules:

Current

Fill box with: 20 copies of: SCM/Water.xyz

Fill box with: 2 copies of: SCM/Methanol.xyz

Random start: □ Yes

Number of cycles: 1

Distance: 2.5 Angstrom

NLoop: 50

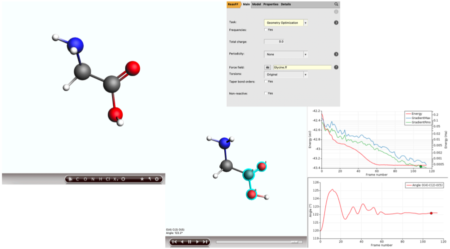
Current density: 0.7045 g/mL

New density will be around: 0.7045 g/mL

AMSinput

Geometry optimization

- Build your favorite molecule with C, H, O, N, Si
- Select Task → Geometry Optimization
- Switch to ReaxFF engine, select the Glycine.ff Force Field
- Save and run the optimization
- SCM → movie to appreciate the relaxation
- Select an angle, Graph → Distance, Angle, Dihedral



Task: Geometry Optimization

Frequencies: □ Yes

Total charge: 0.0

Periodicity: None

Force field: Glycine.ff

Torsions: Original

Taper bond orders: □ Yes

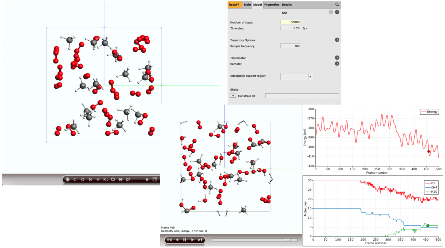
Non-reactive: □ Yes

O(4) C(2) O(5)
Angle: 122.2°

AMSinput

Molecular dynamics

- Build a box of methane and oxygen (15×15×15 Å, 15 CH_4, 30 O_2)
- Switch to Task → Molecular Dynamics
- Switch to ReaxFF, select the CHO.ff Force Field
- Click > next to MD, set 50000 steps, initial temperature 3500 K
- Click > next to Thermostat add NHC at 3500 K @ 100 fs
- Save and run, watch the movie, add graph, plot some molecules, etc.



MD

Number of steps: 50000

Time step: 0.25 fs

Trajectory Options

Sample frequency: 100

Thermostat

Barostat

Adsorption support region:

Shake

Constrain all:

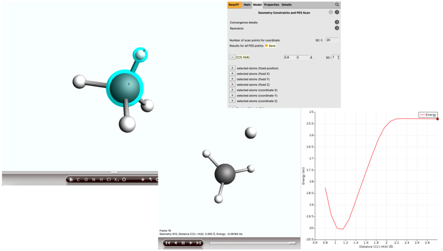
Frame 459

Geometry 460, Energy: -17.01126 Ha

AMSinput

PES scan

- Build a methane molecule, periodicity None
- Switch to Task → PES Scan
- Click > next PES Scan, select C, H atoms, click + next CH distance
- Input 20 points, from 0.8 to 3.0 Å, check for all PES points
- Enable Nuclear gradients from the Properties → Gradients
- Save and run, watch the movie



Geometry Constraints and PES Scan

Convergence details

Restraints

Number of scan points for coordinate SC-1: 20

Results for all PES points: ■ Save

− C(1) H(4) 0.8 3 Å SC-1

+ selected atoms (fixed position)

+ selected atoms (fixed X)

+ selected atoms (fixed Y)

+ selected atoms (fixed Z)

+ selected atoms (coordinate X)

+ selected atoms (coordinate Y)

+ selected atoms (coordinate Z)

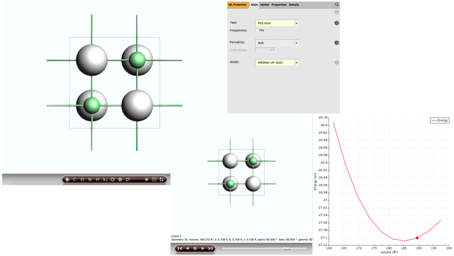
Frame 19

Geometry 913, Distance C(1)--H(4): 3.000 Å, Energy: -0.56184 Ha

AMSinput

Equation of states

- Build NaCl, convert to conventional cell
- Switch to Task → PES Scan
- Switch to MLPotential engine and select M3GNet model
- Click > next PES Scan, and click + Volume and scan from 0.9 to 1.1
- Save and run, watch the movie



Task: PES Scan

Frequencies: □ Yes

Periodicity: Bulk

Total charge: 0.0

Model: M3GNet-UP-2022

Frame 7
Geometry 16, Volume: 189.373 Å³, a: 5.738 Å, b: 5.738 Å, c: 5.738 Å, alpha: 90.000 °, beta: 90.000 °, gamma: 90

15 min break

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- **Theory background (40 min)**
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

Hierarchy of interatomic potentials

Why do we need complex potentials?

- Summary of interatomic potentials.

# parameters, complexity ↓

Empirical potential type	Usage	Feature	Names
Pair-potential	Noble gaz	E $\propto$ Z	LJ, Morse, Buckingham
Pair-functional	Metals	E $\propto \sqrt{Z}$	EAM
Many-body potentials	Opened structures Organic molecules	Angles, dihedral, etc.	OPLS, DREIDING
Reactive potentials	Chemistry, charge transfer/polarization	Many-body +BO, QEq	Tersoff, ReaxFF, REBO, COMB

and Machine Learning Potentials..

ReaxFF: functional form

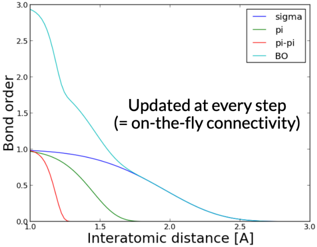
Potential energy, bond order, bond energy

- Total potential energy: ReaxFF parameters documentation

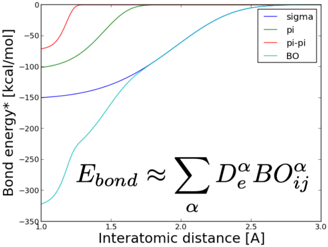
$$E_{pot} = \underbrace{E_{bond} + E_{ang} + E_{tor} + E_{over}}_{f[BO]} + E_{vdW} + E_{coul} + \cdots$$

- Bond order & bond energy

$$BO_{ij} = \exp\left[p_{bo1}\left(\frac{r_{ij}}{r_0^{\sigma}}\right)^{p_{bo2}}\right] + \exp\left[p_{bo3}\left(\frac{r_{ij}}{r_0^{\pi}}\right)^{p_{bo4}}\right] + \exp\left[p_{bo5}\left(\frac{r_{ij}}{r_0^{\pi\pi}}\right)^{p_{bo6}}\right]$$



Updated at every step
(= on-the-fly connectivity)



$$E_{bond} \approx \sum_{\alpha} D_e^{\alpha} BO_{ij}^{\alpha}$$

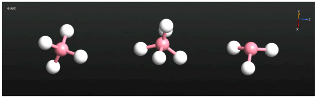
ReaxFF: functional form

Valence angle, over coordination

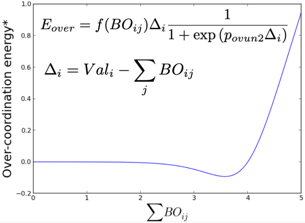
- Valence angle energy

$$E_{ang} \approx \left[1 - \exp\left(-p_{val3} BO_{ij}^{p_{val4}}\right)\right] \left[1 - \exp\left(-p_{val3} BO_{jk}^{p_{val4}}\right)\right]$$
$$\times \left[p_{val1} - p_{val1} \exp\left(-p_{val2} \left(\Theta_{ijk} - \Theta_0\right)^2\right)\right]$$

- Over/under coordination penalties



$\sum_j BO_{ij} = 4$ $\sum_j BO_{ij} = 5$ $\sum_j BO_{ij} = 3$

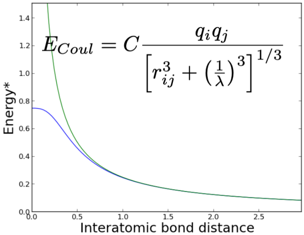


etc.

ReaxFF: functional form

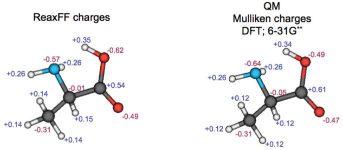
Non-bonded interactions

- vdW ∼ Morse
- Coulomb
  - Shielded Coulomb calculated between all pairs of atoms
  - Charges are calculated at every step with the charge equilibration (QEq, ACKS2)



$$E_{Coul} = C\frac{q_i q_j}{\left[r_{ij}^3 + \left(\frac{1}{\lambda}\right)^3\right]^{1/3}}$$

$$E_{Qeq} = \sum_{i<j} q_i q_j J\left(r_{ij}\right) + \sum_i \left(\chi_i q_i + \frac{1}{2} H_i q_i^2\right)$$



3-parameter per element optimised to match Mulliken charges

ReaxFF: force field

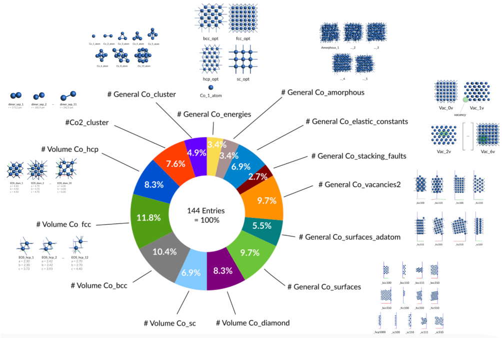
Structure

- General parameters (apply to every types of atom): GEN (in ParAMS)
- Atomic parameters (for each type of atoms): ATM
- Bond parameters (for each pairs/bonds defined): BND
- Off-diagonal parameters (for each pairs/non-bonded defined): OFD
- Angle parameters (for each triplets/angles defined): ANG
- Torsion parameters (for each quadruplets/torsion defined): TOR
- Hydrogen bond parameters (for each pairs/hydrogen bonds defined): HBD

ReaxFF: training set

Recommendations to create a dataset to optimize ReaxFF

- What do we need to include to the training set?
  - Equilibrium, out-of-equilibrium structures
  - PES Scan of dissociation, angle bend, torsion
  - Transition states, equation of states, defects, etc.



Theory of Machine Learning Potentials

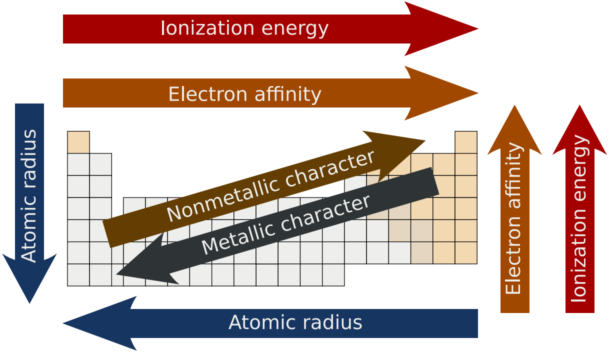
Goal of ML Potentials

- goal is to find $f(\text{atom positions}, \text{atom types}, \text{PBC}, \text{other}) = E_{\text{Pot}}$
- derivatives automatic with ML frameworks $-\frac{\partial E_{\text{Pot}}}{\partial x} = -\frac{\partial f(x,\ldots)}{\partial x} = F$
- Swap with an engine in AMS (e.g. ReaxFF or DFTB) and do MD/GO
- Fit f based on reference data

Theory of Machine Learning Potentials

Embeddings

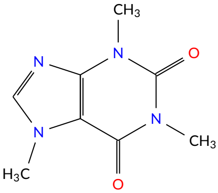
- Atom number to vector description
- Periodic table of elements is human embedding
- Improved scaling w.r.t. number of atom types



Theory of Machine Learning Potentials

Locality

- Predict properties for each atom based on environment

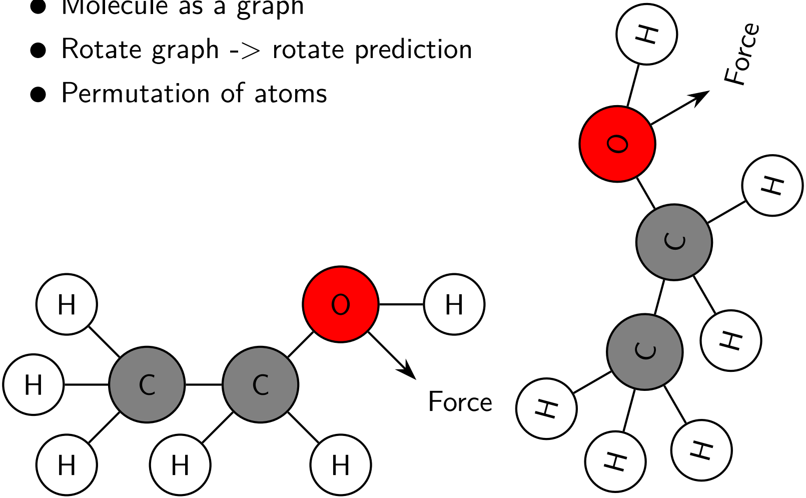


- Sum or average atomic properties to obtain global properties
- Scales well with number of atoms and computational resources
- Physical interpretation
- Short range interactions dominate

Theory of Machine Learning Potentials

Equivariant Graph

- Molecule as a graph
- Rotate graph -> rotate prediction
- Permutation of atoms



Theory of Machine Learning Potentials

Fitting Procedure

- The goal is to fit a potential energy surface (PES) using machine learning models.
- This procedure involves training a model to predict the energy of a system based on atomic positions and types.

Mathematical Formulation

The energy E of a system can be approximated by a function f parameterized by θ, learned from training data:

$$E \approx f(\mathbf{r}; \theta)$$

where **r** represents the positions of atoms.

Theory of Machine Learning Potentials

Fitting Procedure

Objective Function

The fitting involves minimizing the loss function L, typically the mean squared error between predicted and true energies:

$$L(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left[f(\mathbf{r}_i; \theta) - E_i \right]^2$$

where E_i and $\mathbf{r}_i$ are the true energies and atomic positions for each training example.

Optimization by Gradient Descent

Update rules based on the gradient of the loss function:

$$\theta \leftarrow \theta - \eta \nabla_\theta L(\theta)$$

where η is the learning rate.

Theory of Machine Learning Potentials

Transfer Learning

''Transfer learning is a ML method where knowledge from one model is transferred to enhance another model''

- Start with a pre-trained model
- Fine-tune by fitting few parameters
- Saves time and computational resources
- Often achieves superior results
- Particularly effective when new data is scarce

Theory of Machine Learning Potentials

Committee Uncertainty

- Fit multiple models

$$f_i(x)$$

- Prediction is the average

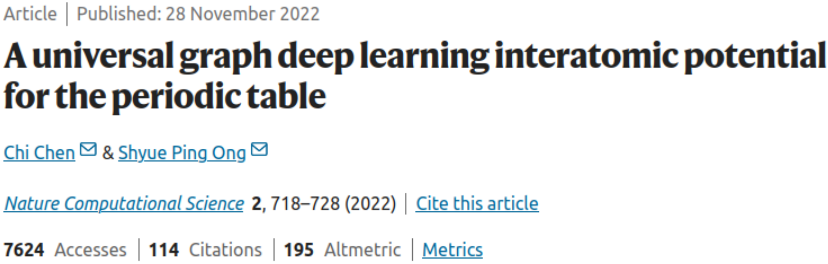
$$\mu(x) = \frac{1}{N}\sum_{i}^{N} f_i(x)$$

- Uncertainty is standard deviation

$$\sigma(x) = \sqrt{\frac{1}{N-1}\sum_{i}^{N}\left(f_i(x) - \mu(x)\right)^2}$$

Theory of Machine Learning Potentials

M3GNet



Article | Published: 28 November 2022

A universal graph deep learning interatomic potential for the periodic table

Chi Chen ✉ & Shyue Ping Ong ✉

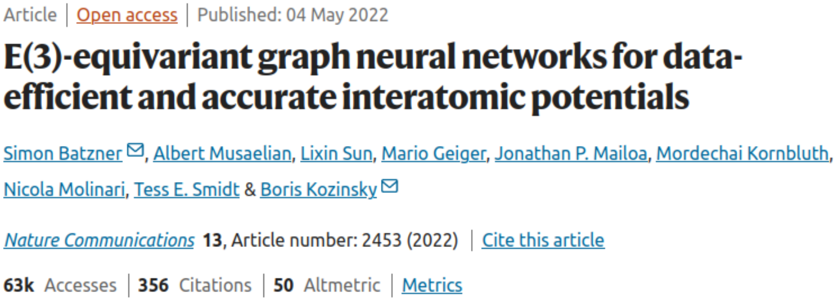
Nature Computational Science **2**, 718–728 (2022) | Cite this article

7624 Accesses | **114** Citations | **195** Altmetric | Metrics

- Universal pre-trained parameters available
- 93 elements
- Transfer learning

Theory of Machine Learning Potentials

NequIP



Article | Open access | Published: 04 May 2022

E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt & Boris Kozinsky

Nature Communications **13**, Article number: 2453 (2022) | Cite this article

63k Accesses | **356** Citations | **50** Altmetric | Metrics

- State-of-the-art accuracy
- Relatively slow

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- **Introduction to ParAMS (40 min)**
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

ParAMS

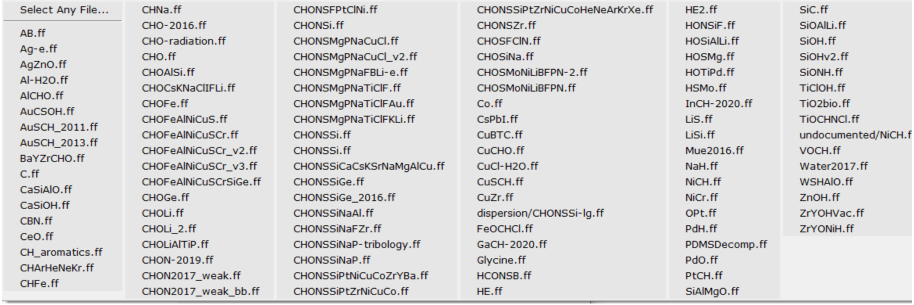
Features

Selected features:

- Import reference data from AMS, VASP, Quantum ESPRESSO, Gaussian, or experiment
- Use a validation set to prevent overfitting
- Submit jobs to remote machines using the GUI
- Results updated on-the-fly in the GUI with many diagrams
- Many properties: reaction energies, forces, bond lengths, angles, cell parameters, stress tensors, charges and user defined.
- Use single points, geometry optimizations, or PES scans during the parametrization
- Set custom weights
- Choose which parameters to optimize, and set allowed ranges

Preparametrized models in AMS

- AMS comes with many published ReaxFF force field and DFTB parameters.



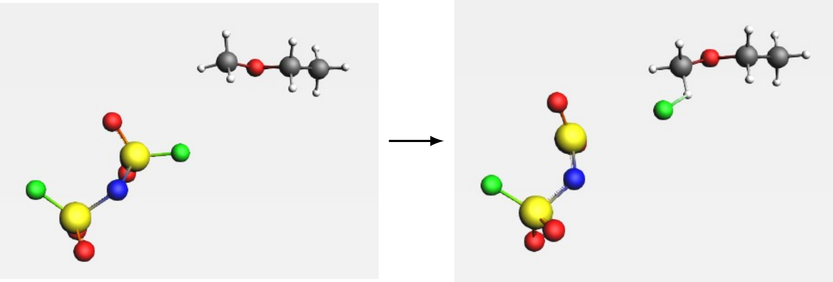
Select Any File...

AB.ff
Ag-e.ff
AgZnO.ff
Al-H2O.ff
AlCHO.ff
AuCSOH.ff
AuSCH_2011.ff
AuSCH_2013.ff
BaYZrCHO.ff
C.ff
CaSiAlO.ff
CaSiOH.ff
CBN.ff
CeO.ff
CH_aromatics.ff
CHArHeNeKr.ff
CHFe.ff
CHNa.ff
CHO-2016.ff
CHO-radiation.ff
CHO.ff
CHOAlSi.ff
CHOCsKNaClIFLi.ff
CHOFe.ff
CHOFeAlNiCuS.ff
CHOFeAlNiCuSCr.ff
CHOFeAlNiCuSCr_v2.ff
CHOFeAlNiCuSCr_v3.ff
CHOFeAlNiCuSCrSiGe.ff
CHOGe.ff
CHOLi.ff
CHOLi_2.ff
CHOLiAlTiP.ff
CHON-2019.ff
CHON2017_weak.ff
CHON2017_weak_bb.ff
CHONSFPtClNi.ff
CHONSi.ff
CHONSMgPNaCuCl.ff
CHONSMgPNaCuCl_v2.ff
CHONSMgPNaFBLi-e.ff
CHONSMgPNaTiClF.ff
CHONSMgPNaTiClFAu.ff
CHONSMgPNaTiClFKLi.ff
CHONSSi.ff
CHONSSi.ff
CHONSSiCaCsKSrNaMgAlCu.ff
CHONSSiGe.ff
CHONSSiGe_2016.ff
CHONSSiNaAl.ff
CHONSSiNaFZr.ff
CHONSSiNaP-tribology.ff
CHONSSiNaP.ff
CHONSSiPtNiCuCoZrYBa.ff
CHONSSiPtZrNiCuCo.ff
CHONSSiPtZrNiCuCoHeNeArKrXe.ff
CHONSZr.ff
CHOSFClN.ff
CHOSiNa.ff
CHOSMoNiLiBFPN-2.ff
CHOSMoNiLiBFPN.ff
Co.ff
CsPbI.ff
CuBTC.ff
CuCHO.ff
CuCl-H2O.ff
CuSCH.ff
CuZr.ff
dispersion/CHONSSi-lg.ff
FeOCHCl.ff
GaCH-2020.ff
Glycine.ff
HCONSB.ff
HE.ff
HE2.ff
HONSiF.ff
HOSiAlLi.ff
HOSMg.ff
HOTiPd.ff
HSMo.ff
InCH-2020.ff
LiS.ff
LiSi.ff
Mue2016.ff
NaH.ff
NiCH.ff
NiCr.ff
OPt.ff
PdH.ff
PDMSDecomp.ff
PdO.ff
PtCH.ff
SiAlMgO.ff
SiC.ff
SiOAlLi.ff
SiOH.ff
SiOHv2.ff
SiONH.ff
TiClOH.ff
TiO2bio.ff
TiOCHNCl.ff
undocumented/NiCH.f
VOCH.ff
Water2017.ff
WSHAlO.ff
ZnOH.ff
ZrYOHVac.ff
ZrYONiH.ff

Industry Example

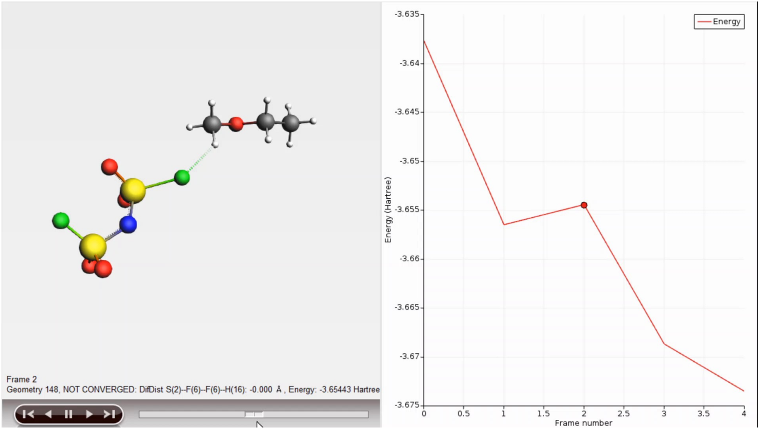
ReaxFF and DFTB Problems

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



Industry Example

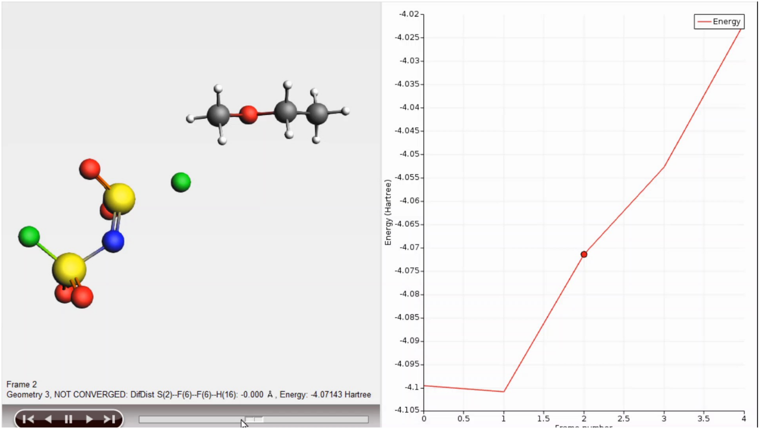
Published ReaxFF predictions



Frame 2
Geometry 148, NOT CONVERGED: DifDist S(2)--F(6)--F(6)--H(16): -0.000 Å , Energy: -3.65443 Hartree

Industry Example

AMS: Easily verify/disprove ReaxFF prediction with DFT



Frame 2
Geometry 3, NOT CONVERGED: DifDist S(2)--F(6)--F(6)--H(16): -0.000 Å , Energy: -4.07143 Hartree

Industry Example

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

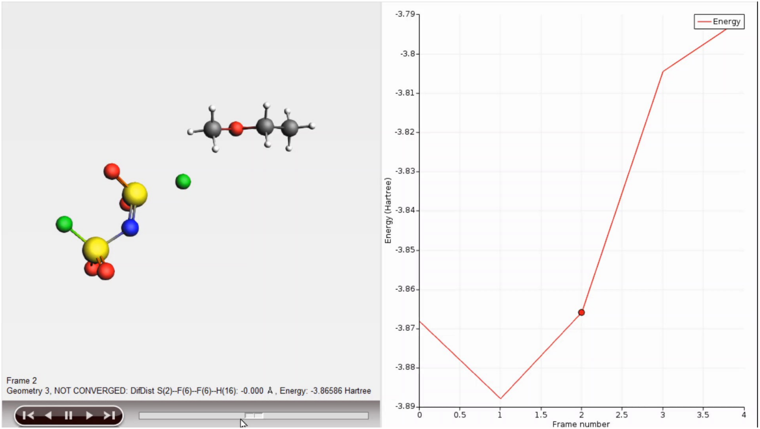
- For this example:

Set up new DFT calculations:	1 hour
Run DFT calculations:	1 hour
Set up ParAMS 20:	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.
- Start from one of the many tutorials and modify as needed.
- *Note: Most parametrization projects will take longer*

Industry Example

Reoptimized ReaxFF predictions

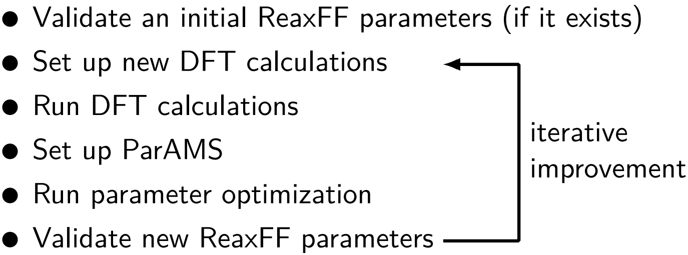


Frame 2

Geometry 3, NOT CONVERGED: DifDist S(2)--F(6)--F(6)--H(16): -0.000 Å , Energy: -3.86586 Hartree

ParAMS conclusions

Typical Workflow



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

iterative improvement

ParAMS Conclusion

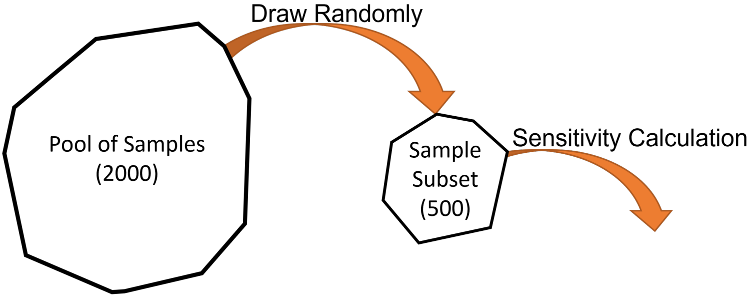
Best practices

- Which parameters to optimize?
    - Choose as **few** as possible
    - Fist try parameters in the **"standard"** category
    - If a parameter value is close to min/max, change the range and continue
- Which optimization algorithm?
    - **CMA-ES** recommended for most optimization problems
    - Details about this algorithm in previous seminar

ParAMS Conclusion

Sensitivity

- **Too few** parameters might make it difficult to find a good fit.
- **Too many** parameters will make the optimization harder, slower, and increase the risk of overfitting.
- Sensitivity is calculated using the Hilbert-Schmidt Independence Criteria (HSIC)



ParAMS Conclusion

Sensitivity

- Obtain estimate of change of loss with change of parameters
- Larger influence on the loss means a parameter is more important
- Sample different parameters and obtain loss function
- Obtain HSIC using (gaussian) kernel
- Easy to interpret: sum is 1 and larger means more important
- Not needed for ML since partial derivatives are similar

Data Model

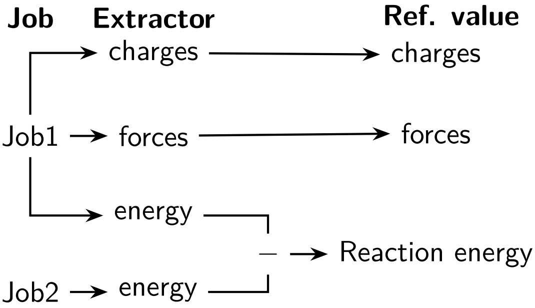
Overview

- Dataset: Reference or inference values
- JobCollection: Input settings required for reference/inference
  - Training Set
  - Validation Set
  - Test Set
- EngineCollection: Engine settings (e.g. DFT functional)

Data Model

Types of Reference Values

- Anything that can be extracted from a job can be 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: Adsorption energy, surface energy, formation energy, ..



Job **Extractor** **Ref. value**

Job1 → charges → charges

Job1 → forces → forces

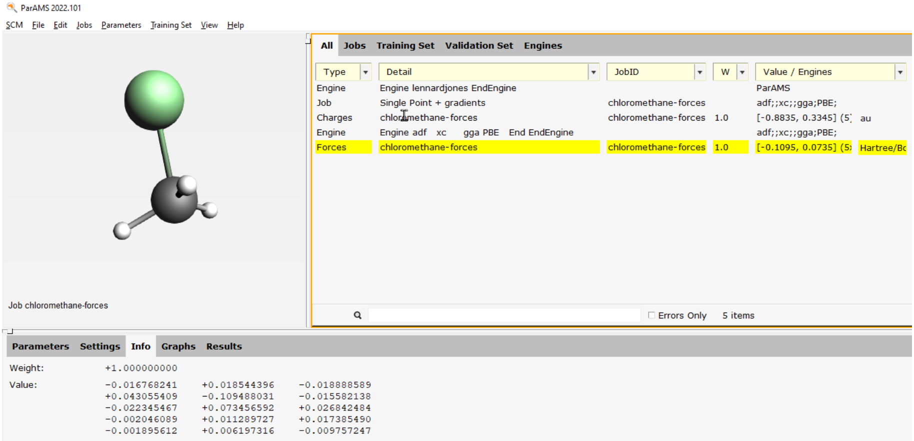
Job1 → energy

Job2 → energy

– → Reaction energy

Data Model

Example reference values: Forces and Charges


ParAMS 2022.101
SCM File Edit Jobs Parameters Training Set View Help
All Jobs Training Set Validation Set Engines
Parameters Settings Info Graphs Results
□ Errors Only
5 items

Job chloromethane-forces

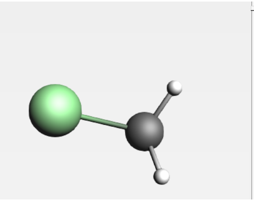
Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	Single Point + gradients	chloromethane-forces		adf;;xc;;gga;PBE;	
Charges	chloromethane-forces	chloromethane-forces	1.0	[-0.8835, 0.3345] (5)	au
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	
Forces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

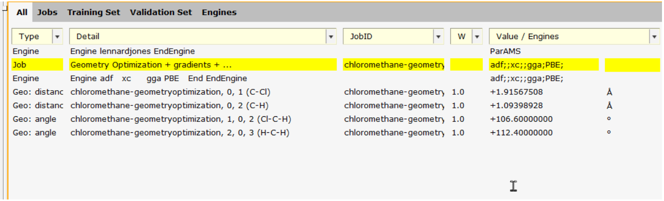
```
Weight:   +1.000000000
Value:    -0.016768241   +0.018544396   -0.018888589
          +0.043055409   -0.109488031   -0.015582138
          -0.022345467   +0.073456592   +0.026842484
          -0.002046089   +0.011289727   +0.017385490
          -0.001895612   +0.006197316   -0.009757247
```

Data Model

Example reference values: Optimized bond lengths and angles

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



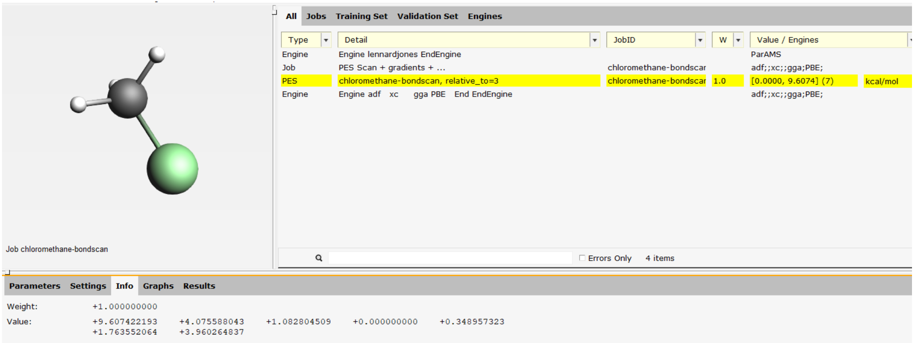


Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	Geometry Optimization + gradients + ...	chloromethane-geometry		adf;;xc;;gga;PBE;	
Engine	Engine adf xc gga PBE End EndEngine			adf;;xc;;gga;PBE;	
Geo: distanc	chloromethane-geometryoptimization, 0, 1 (C-Cl)	chloromethane-geometry	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	°

Data Model

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

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



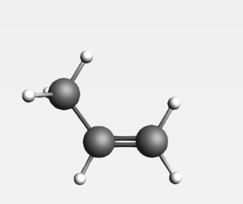
Type	Detail	JobID	W	Value / Engines	
Engine	Engine lennardjones EndEngine			ParAMS	
Job	PES Scan + gradients + ...	chloromethane-bondscar		adf;;xc;;gga;PBE;	
PES	chloromethane-bondscan, relative_to=3	chloromethane-bondscar	1.0	[0.0000, 9.6074] (7)	kcal/mol
Engine	Engine adf xc gga PBE End EndEngine	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

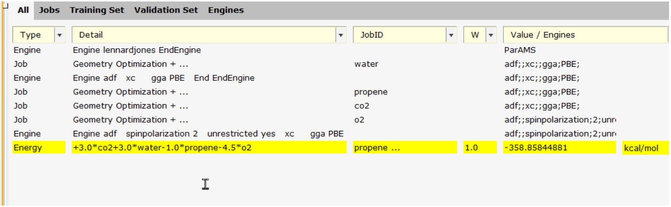
```
Weight:     +1.000000000
Value:      +9.607422193    +4.075588043    +1.082804509    +0.000000000    +0.348957323
            +1.763552064    +3.960264837
```

Data Model

Example reference values: Reaction energy

- Propane Combustion:$C_3H_8(g) + \frac{9}{2}O_2(g) \rightarrow 3CO_2(g) + 3H_2O(g)$
- Automatically balanced stoichiometric coefficients!





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

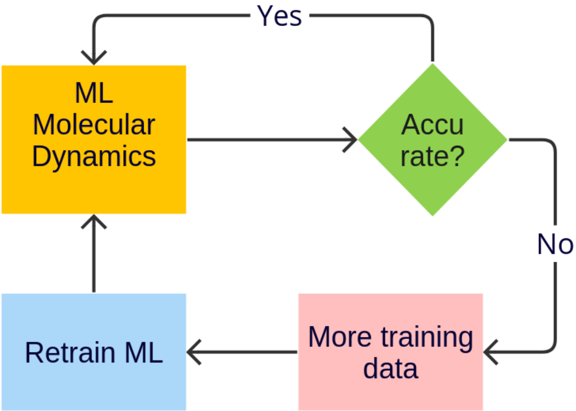
Data Model

Differences for Machine Learning Models

- Machine learning requires partial derivatives
- Can only include direct predictions
  - Single Point jobs
  - Energies
  - Forces
  - Stresses

Active Learning

Automatic Training Workflow



Active Learning

Overview

The workflow:

- **Trains** an initial ML Potential
- **Runs** the MD simulation with ML
- **Pauses** the MD simulation and runs reference calculations
- **Retrains** the ML Potential to the new reference data
- **Rewinds** the MD simulation to the last point where it was known to be accurate
- **Continue** the MD simulation

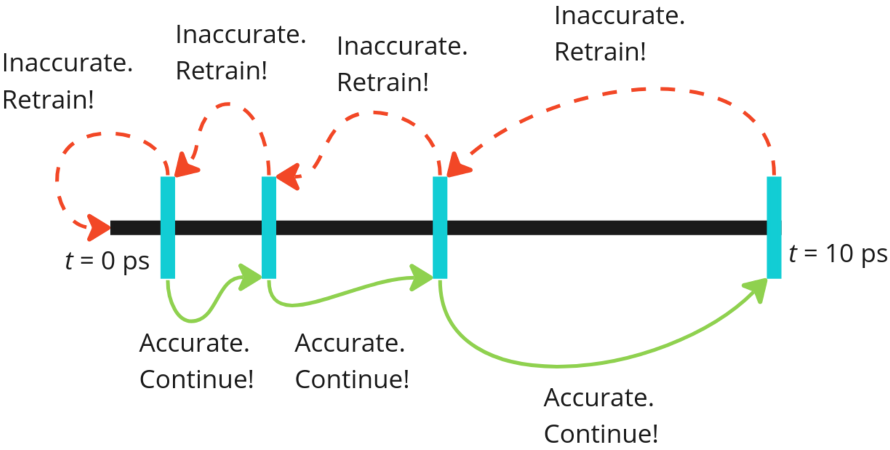
Active Learning

Input

- **Input System.** This is the initial system for the MD simulation. The input is exactly the same as for any other AMS simulation.
- **Molecular dynamics settings.** It can be equilibrium or non-equilibrium MD. The settings/input are exactly the same as for any other AMS simulation.
- **Reference engine settings.** This can be any engine, primarily DFT engines: ADF, BAND, or Quantum ESPRESSO. The settings/input are the same as for any other AMS simulation. This determines the level of theory to which the ML potential is trained.
- **ParAMS ML training settings.** You can train any ML potential that is supported by ParAMS, for example, M3GNet. The settings/input are exactly the same as for running standalone ParAMS with Task MachineLearning.
- **Active learning settings.** These settings determine, for example, how frequently to launch new reference calculation, and how to judge if the ML potential is accurate enough.

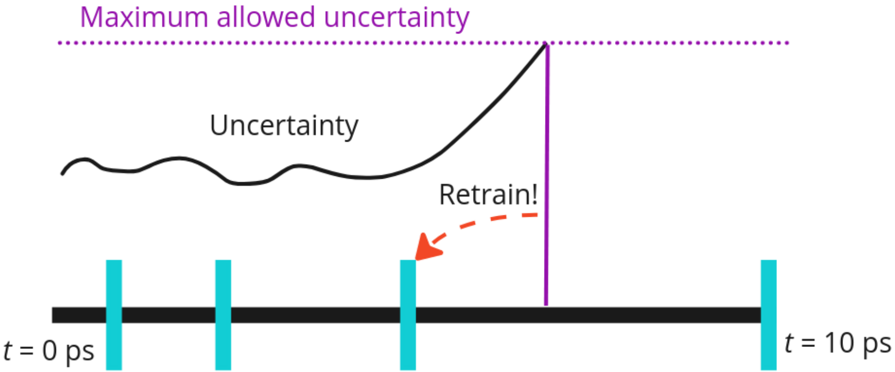
Active Learning

Rewind Molecular Dynamics



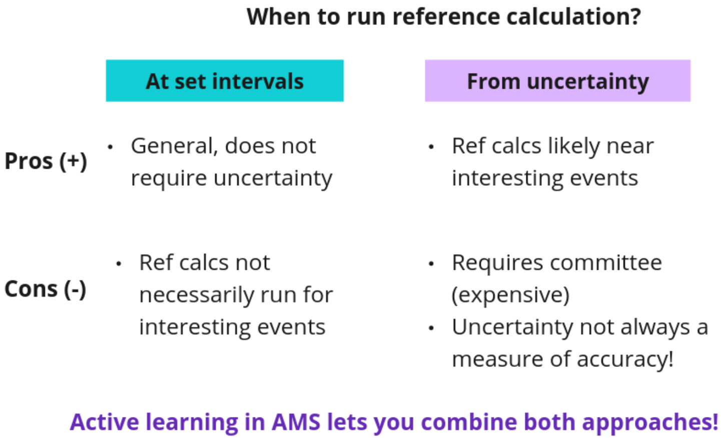
Active Learning

Uncertainty



Active Learning

Advantages vs Disadvantages



When to run reference calculation?

	At set intervals	From uncertainty
Pros (+)	- General, does not require uncertainty	- Ref calcs likely near interesting events
Cons (-)	- Ref calcs not necessarily run for interesting events	- Requires committee (expensive) - Uncertainty not always a measure of accuracy!

Active learning in AMS lets you combine both approaches!

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- **Tour of the ParAMS GUI [Demo] (20 min)**

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

Tour of the ParAMS GUI

Lennard-Jones Gas

- The Lennard-Jones potential is a simple model that describes the interaction between a pair of neutral atoms or molecules.
- It is particularly noted for its ability to capture the balance between attractive and repulsive forces experienced by particles:

  $$V(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{1}$$

  where:
  - ϵ is the depth of the potential well,
  - σ is the finite distance at which the inter-particle potential is zero,
  - r is the distance between particles.
- This potential is used extensively in molecular dynamics simulations of gases, liquids, and solids.

Tour of the ParAMS GUI

Workflows

- **GUI:** Recommended main interface allowing you to easily setup tasks, visualize results, and submit local or remote jobs.
- **Scripting:** Python/PLAMS interface to ParAMS allowing you to integrate it into data workflows and easily setup multiple configurations. Use the $AMSBIN/amspython program to execute python scripts.
- **Command-line:** Console interface for systems without GUI support (e.g. submitting jobs on a cluster). This is not discussed today.

Tour of the ParAMS GUI

Required Files

- **copy** `$AMSHOME/scripting/scm/params/examples/LJ_Ar`
- For this example the reference data has already been computed
- Open with `$AMSBIN/params`

ParAMS Input

Parameter Interface

- Parameters tab
- value and allowed ranges
- active: fit or not
- val%: how close to min/max

ParAMS Input

Job Collection

- periodic view
- jobs tab
- Job-ID
- Detail
- Ref Engine
- ParAMS Engine

ParAMS Input

Training set

- Training tab
- Energy
- Forces
- Detail (clickable)
  - expression
  - sigma
  - weight
  - unit
  - balance
- value
- predicted
- Loss%: contribution to the loss function

ParAMS Input

ParAMS Settings

- Main panel
- Optimizers
- Time limit
- Max optimizers converged

Run ParAMS

Run

- Save as
- Run
- AMSJobs

ParAMS Results

Best Parameter

- parameters tab

ParAMS Results

Correlation Plots

- Graphs Panel
- Loss function
- RMSE
- Scatterplot - select points to view outliers

ParAMS Results

Error Plots

- Stats - Forces
- MAE/RMSE

ParAMS Results

Parameter Plots

- active parameters
- eps
- rmin

ParAMS Results

Editing and saving plots

- double click
- axis
- general
- File: save graph as

ParAMS Results

Predicted values

- Training set panel
- Prediction column
- info panel to see all
- Results Panel
- loss contributions

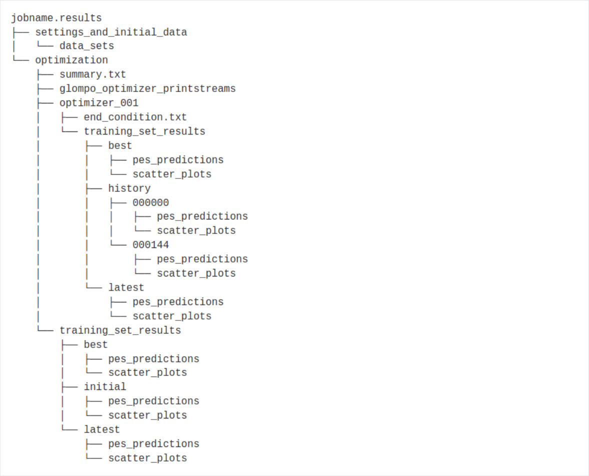
ParAMS Results

Summary statistics

- Results panel
- Training best: stats
- MAE and RMSE
- N
- loss contributions
- find ”bad” parts

ParAMS Results

Output structure



```
jobname.results
├── settings_and_initial_data
│   └── data_sets
└── optimization
    ├── summary.txt
    ├── glompo_optimizer_printstreams
    ├── optimizer_001
    │   ├── end_condition.txt
    │   └── training_set_results
    │       ├── best
    │       │   ├── pes_predictions
    │       │   └── scatter_plots
    │       ├── history
    │       │   ├── 000000
    │       │   │   ├── pes_predictions
    │       │   │   └── scatter_plots
    │       │   └── 000144
    │       │       ├── pes_predictions
    │       │       └── scatter_plots
    │       └── latest
    │           ├── pes_predictions
    │           └── scatter_plots
    └── training_set_results
        ├── best
        │   ├── pes_predictions
        │   └── scatter_plots
        ├── initial
        │   ├── pes_predictions
        │   └── scatter_plots
        └── latest
            ├── pes_predictions
            └── scatter_plots
```

Lunch

Hands-on in the afternoon

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- **Optimize ReaxFF with ParAMS [Hands-on] (1 h)**

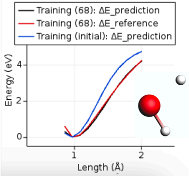
15 minutes break

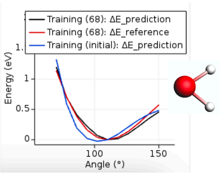
- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- Conclusions

Train ReaxFF with ParAMS

Simple water molecule example

- In this example we will fit a ReaxFF force field
- The goal is to reproduce DFT bond dissociation and angle curves of water in the gas phase
- You will learn how to:
    1. Generate the reference data
    2. Import the data into ParAMS
    3. Evaluate the sensitivity of the parameters [optional]
    4. Optimize the parameters
    5. Analyze the results





Reference data

PES Scans

- Create a water molecule
- Use ADF engine and keep the default settings
- Perform a PES Scan to dissociate H from O from 0.85 to 2.0 Å in 11 steps
- Perform a PES Scan to bend the H-O-H angle from 70 to 150 degrees in 11 steps

Import data into ParAMS

Training set

- Import the PES Scans into ParAMS as Single Job PES Scans

Sensitivity analysis [optional]

Which parameters to optimize?

- Switch to ReaxFF parameters (Parameters → ReaxFF)
- Load the Water2017.ff Force Field (Parameters → Load ReaxFF ForceField..)
- Select all bond and angle parameters corresponding to H.O and H.O.H, respectively
- Switch to task Sensitivity, check Run sampling, and set repeat the calculation to 5 with 500 samples per repeat
- Save the job and run the sensitivity analysis
- From this morning's lesson, try to guess which parameters will be the most sensitive

Sensitivity analysis [optional]

Analysis

- Appreciate the sensitivity of the parameters in the Results tab
- You should be able to understand the most sensitive parameters from the theory background lesson of this morning
- From the Parameters tab, unselect the non-active parameters to display only the active parameters
- From the Sensitivity tab, sort the parameters by decreasing order
- We will only optimize parameters with large sensitivity

Parameters optimization

Let's optimize ReaxFF!

- Select the 4 parameters to optimize: `H.O:D_e^sigma`, `H.O:p_bo1`, `H.O:p_bo2`, `H.O.H:Theta_0,0`
- Setup an optimization with CMAES optimizer
- Set the max loss function calls to 100
- In Options → Optimizer set (σ_o, Popsize, Min sigma) to (0.1, 10, 0.0001)
- Save the job and run the parameters optimization

Parameters optimization

Analysis

- During the optimization you can appreciate the loss function being minimized
- Compare the dissociation and angle bend PES with the initial prediction

Optional (1)

Reaction energy

- Optimize the geometry of H_2O, O_2 (triplet) and H_2 molecules with ADF
- Open a new ParAMS window, File → New to clear it up
- Import the 3 molecules as Single Job, Geometry Optimization, Energy

We can now define the reaction energy corresponding to $H_2O \rightarrow H_2 + \frac{1}{2}O_2$

- Select the H_2O entry and duplicate it with Training Set → Add → Energy
- Double click on the new H_2O entry
- Type `energy("water")-energy("hydrogen")-energy("oxygen")` in the Energy box replacing `water`, `hydrogen`, `oxygen` with the name of your respective jobs (written in the Details column)
- Click the Balance button to balance the equation automatically
- Delete the value in the Values box and click OK

Run a Single Point task to evaluate the reaction energy with ReaxFF

- Load the Water2017.ff Force Field
- Select Single Point task and run with the defaults settings

Optional (2)

Already finished?

- If you were to create a ReaxFF for water, what other interactions are important?
- What can you add to the training set to describe these additional interactions?
- Build the additional systems, perform the DFT calculations, run the sensitivity and the optimization again

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

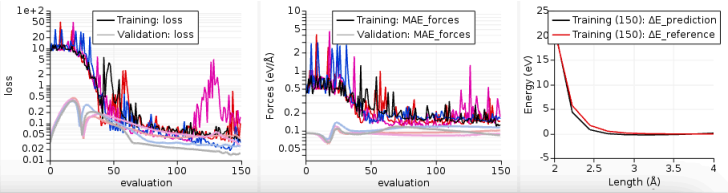
15 minutes break

- **Optimize MLPotentials with ParAMS [Hands-on] (1 h)**
- Active Learning [Hands-on] (1 h)
- Conclusions

Train M3GNet with ParAMS

Simple liquid Ar example

- In this example we will fit a M3GNet force field
- The goal is to optimize M3GNet to describe liquid Ar
- You will learn how to:
  1. Generate some reference data
  2. Import the data into ParAMS
  3. Optimize M3GNet and committees
  4. Analyze the results



Reference data

Load the data

- Start ParAMS, clear it with File → New, if needed
- Load the dataset for Argon using one of this procedure:
  - File → Open and navigate to `$AMSHOME/scripting/scm/params/examples/M3GNet/params.in`
  - or download the file from here and import it with File → Open and navigate to the `params.in` file you just downloaded
  - or from a Terminal type `params -gui $AMSHOME/scripting/scm/params/examples/M3GNet/params.in`
- This loads a few liquid argon structures and sets up some ML training settings
- For training machine learning potentials, you can only train to single-point energy and forces
- Verify from the job collection that the imported dataset will work with MLPotentials training

Reference data

Training and validation set

- Count the number of configurations included in the training and validation sets
- Verify that the energy and forces for a given job belong to the same data set

Reference data

PES scan

We will add the PES corresponding to 2 Ar atoms as a function of separation

- Start AMSinput
- Switch to Force Field engine, keep the defaults settings
- Create 2 Ar atoms separated by 2 Å
- If there is a bond, delete it (select the bond and press delete)
- Switch to PES Scan
- Setup a PES Scan to scan the distance between the 2 Ar from 2 to 4 Å in 11 steps
- Check the box to save the results for all PES points
- Enable the calculation of the nuclear gradients in Properties → Gradients, Stress tensor
- Save and run the PES Scan

Reference data

Training and validation set

- Import the PES Scan to ParAMS as Single Job, PES Scan.
- Do you get one (or two) warning messages from ParAMS? if yes why?

Training

Train M3GNet from scratch

- Go over the settings for the Machine Learning task and be sure you understand them
- Save and run the optimization
- During training you can appreciate the minimization of the loss functions
- Is the training ended before the 150 max epochs set? If yes why?

Note that because ML training can only take single points, the PES Scan added before was not used during training. However, at the end of the training, the PES Scan was evaluated with the final parameters.

- Look in the dropdown menu of the graphs and find the PES Scan prediction and reference calculations
- Does it look accurate?

Training

Train M3GNet from scratch

- from the previous optimization, import the PES Scan to ParAMS as PES Scan SinglePoints and set the Properties to energy and forces
- Change units to eV and eV/Å to be consistent with the dataset already loaded
- Is the imported data added to the training set or validation set?
- Save and run the ParAMS optimization
- Once finished, check the PES Scan. Is it accurate now?

Optional 1

Run MD with the optimized model

- From ParAMS you can easily export the optimized model into AMSinput
- File → Open Optimized Engine in AMSinput → Best validation
- Propose a way to build a box filled with Ar atoms
- Setup an MD simulation with the optimized model for 10000 steps starting with velocities at 300 K
- Run the MD simulation

Optional 2

Train a committee of M3GNet

- Re-optimize the model, this time with a committee size equal to 4
- Appreciate the loss for each committee
- Export the model to AMSinput
- Run an MD simulation with the optimized model
- Appreciate the uncertainties during the MD simulation

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- **Active Learning [Hands-on] (1 h)**
- Conclusions

Simple Active Learning

Input

- **Input System.** This is the initial system for the MD simulation. The input is exactly the same as for any other AMS simulation.
- **Molecular dynamics settings.** It can be equilibrium or non-equilibrium MD. The settings/input are exactly the same as for any other AMS simulation.
- **Reference engine settings.** This can be any engine, primarily DFT engines: ADF, BAND, or Quantum ESPRESSO. The settings/input are the same as for any other AMS simulation. This determines the level of theory to which the ML potential is trained.
- **ParAMS ML training settings.** You can train any ML potential that is supported by ParAMS, for example, M3GNet. The settings/input are exactly the same as for running standalone ParAMS with Task MachineLearning.
- **Active learning settings.** These settings determine, for example, how frequently to launch new reference calculation, and how to judge if the ML potential is accurate enough.

Active Learning

Input System

- Open AMSInput
- Switch ADF to MD Active Learning
- Select File → Import Coordinates... → From SMILES, type OCC=O and click OK
- You can also use another SMILES string for a (small) molecule of interest
- Preoptimize molecule
- Make any modifications you wish

Active Learning

Reference Engine

- Select a reference engine from the drop down menu.
- For workshop select a fast engine like ForceField or MLPotential
- Production work typically uses ADF/Band/QuantumESPRESSO etc
- Click the new engine tab
- Return to the main panel

Active Learning

Molecular Dynamics settings

- Go to the MD panel by clicking next to the Task
- Setup 100000 steps with a time step of 0.5fs
- Setup an NHC thermostat at 300K with a damping constant of 200fs
- Return to the main panel

Active Learning

Machine Learning Settings

- Verify backend is M3GNet (default)
- Go to the Machine Learning panel
- Set up transfer learning
- What does committee size do?
- What does the learning rate do?
- Feel free to pick different settings to compare

Active Learning Settings

- Initial Reference Data
  - Obtain initial reference data using M3GNet Universal Potential
- Step sequence
  - Geometric increasing steps, 5 steps, 10 MD frames for first step
- Success criteria
  - At what accuracy is a step considered successful?
- Reasonable simulation criteria
  - Go to the Model → Reasonable Simulation Criteria panel
  - Why are the maximum temperature and minimum distance important?
  - How to "unlock" energy and gradients uncertainty?
  - Why is uncertainty helpful?
  - What is the downside?

Active Learning

Other Settings

Go to the Details → Active Learning panel

- What to do at the end: retrain and/or rerun simulation. Why?
- What data to save. Why save intermediate steps?
- Determine how many attempts to make per step.
- Determine how many reference calculations to perform per attempt.
- Parallel settings

Active Learning

Run the active learning

- File → Save
- File → Run
- Log file
- MD Trajectory
- Training and Validation results
- Open newly trained model in AMSInput

Active Learning

Log file

- The active learning loop is not deterministic, so your results are likely to look a little different.
- In the **timings**, we see that very little time was spent on reference calculations, which makes sense since we used the UFF force field.
- In the **summary**, some of the attempts are marked as FAILED. This means that the energy/forces of the structure in the MD simulation was not predicted accurately enough in comparison to the reference method. This triggers a retraining of the ML model, and then the step is reattempted.
- The **final engine settings** contains an Engine block that can be used in AMS text input files to use your final trained model in production simulations. Note that the units are always “eV” and “angstrom”, no matter which units you use for the reference data in ParAMS.
- You can copy-paste all the lines Engine MLPotential ... EndEngine directly into AMSinput, in order to set up a new job with your trained model.

Active Learning

MD Trajectory

- In AMSjobs, select the job and choose SCM → Movie
- To open the most recent file, you may need to close AMSmovie and start it again.
- In AMSMovie, you can select File → Related files to see some of the other available trajectory files.
- If you open the trajectory files before the active learning loop finished, you may see strange jumps or discontinuities in the plotted Energy.
- This is because the ML model is retrained on-the-fly!
- If the "Rerun simulation at end" option is enabled (it is enabled by default), then the trajectory in the final_production_simulation directory contains the entire trajectory calculated with only the final parameters.

Active Learning

Training and validation results

- In AMSjobs, select the job and choose SCM → ParAMS
- What can you learn from the loss function graph?
- What can you learn from the scatter plots?
- In ParAMS, select File → Open Optimized Engine in AMSinput → Best validation

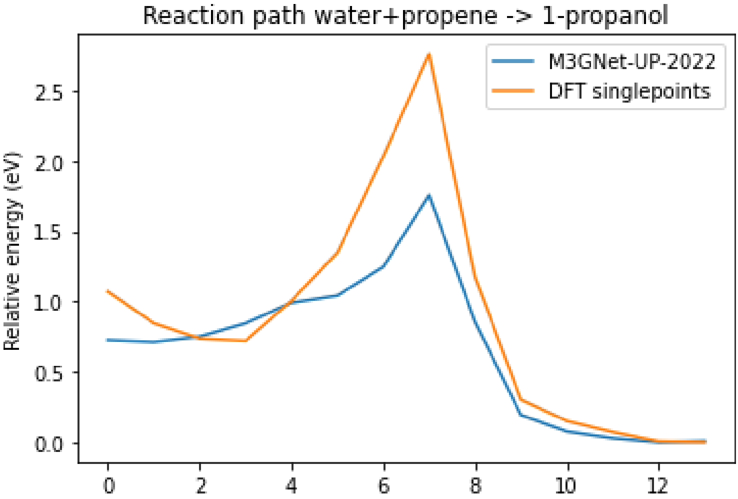
Simple Active Learning

Examples page

- Active learning is useful for many experiments
- We have a page with examples:
- https://www.scm.com/doc/Workflows/SimpleActiveLearning/PythonExa
- Most examples take 1 day or more to run
- Fit ML potential for reactions

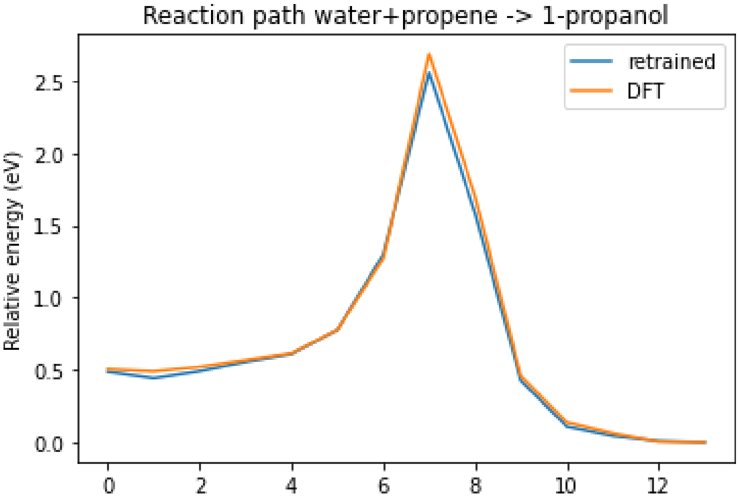
Simple Active Learning

Reaction before fitting



Simple Active Learning

Reaction after fitting



Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 min)
- Introduction to ParAMS (40 min)
- Tour of the ParAMS GUI [Demo] (20 min)

Lunch break (45 min)
Afternoon (3 h)

- Optimize ReaxFF with ParAMS [Hands-on] (1 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active Learning [Hands-on] (1 h)
- **Conclusions**

Overview of the workshop

Use ParAMS to train ReaxFF and machine learning potentials

Morning (3 h)

- General introduction of AMS (30 min)
- The graphical user interface [Demo] (30 min)

15 minutes break

- Theory background (40 h)
- Introduction to ParAMS [Demo] (1 h)

Lunch break (45 min)

Afternoon (4 h)

- Optimize ReaxFF with ParAMS [Hands-on] (2 h)

15 minutes break

- Optimize MLPotentials with ParAMS [Hands-on] (1 h)
- Active learning [Hands-on] (1 h)
- Conclusions

What next?

Learn more about AMS

- Get a trial license www.scm.com/trial
- Explore more ParAMS examples www.scm.com/ParAMS
- Get a license: contact your local reseller

Thank you for your attention!
SCM
Software for Chemistry & Materials
Amsterdam Modeling Suite
Making Computational Chemistry Work For You
Try for free
scm.com/trial
CH3
OH
O
CH3