

Tutorial Session: Deep Dipole

SPEAKER: PINCHEN XIE

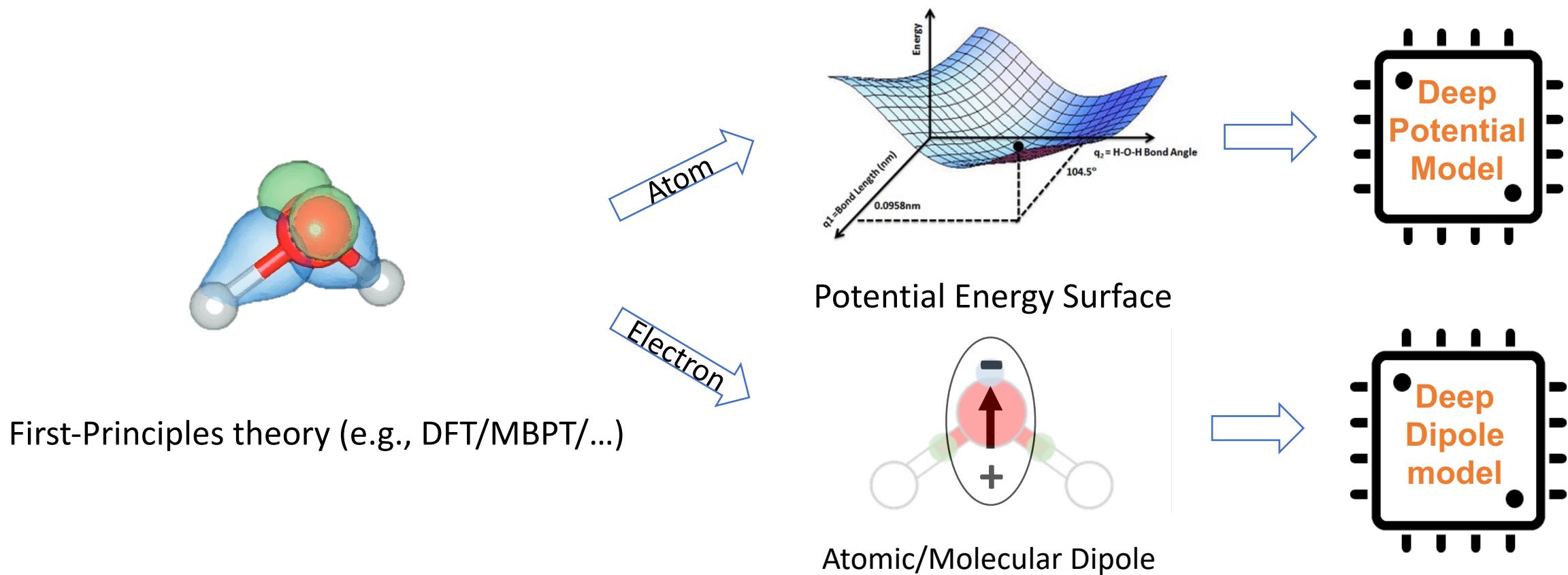
PACM, PRINCETON

JULY, 2023

Slides at:

<https://github.com/CSIPrinceton/workshop-july-2023/blob/main/hands-on-sessions/day-3/7-deep-wannier/csi-workshop-2023-dipole.pdf>

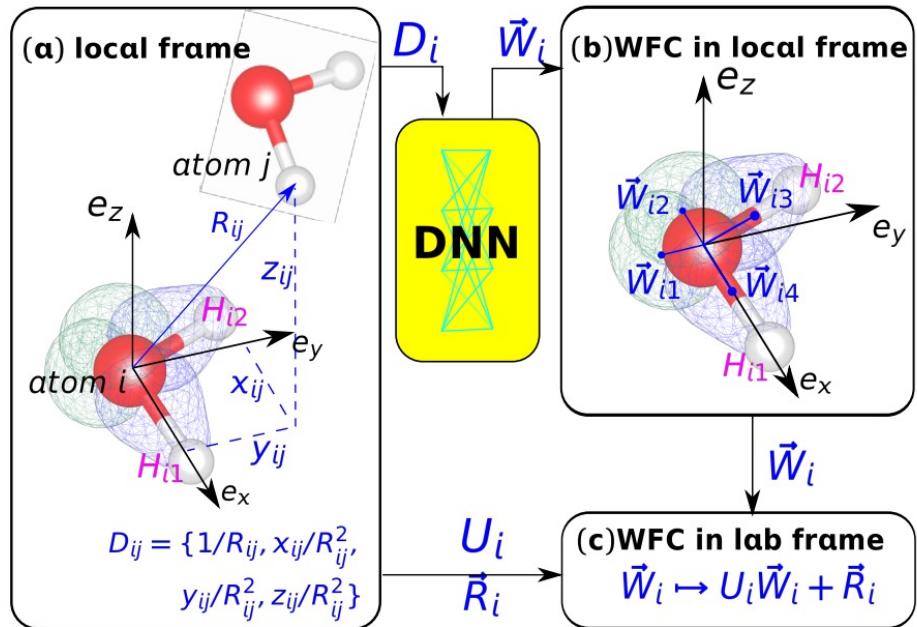
Beyond potential energy surface



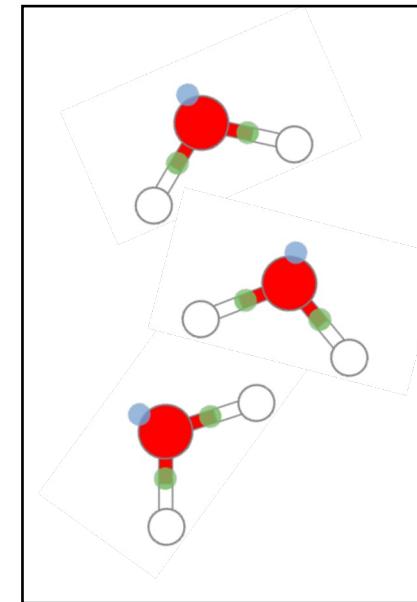
Sketch of water molecule adapted from Fig.2 of PNAS 114.41 (2017): 10846-10851.

A very short introduction to Deep Dipole

The open box



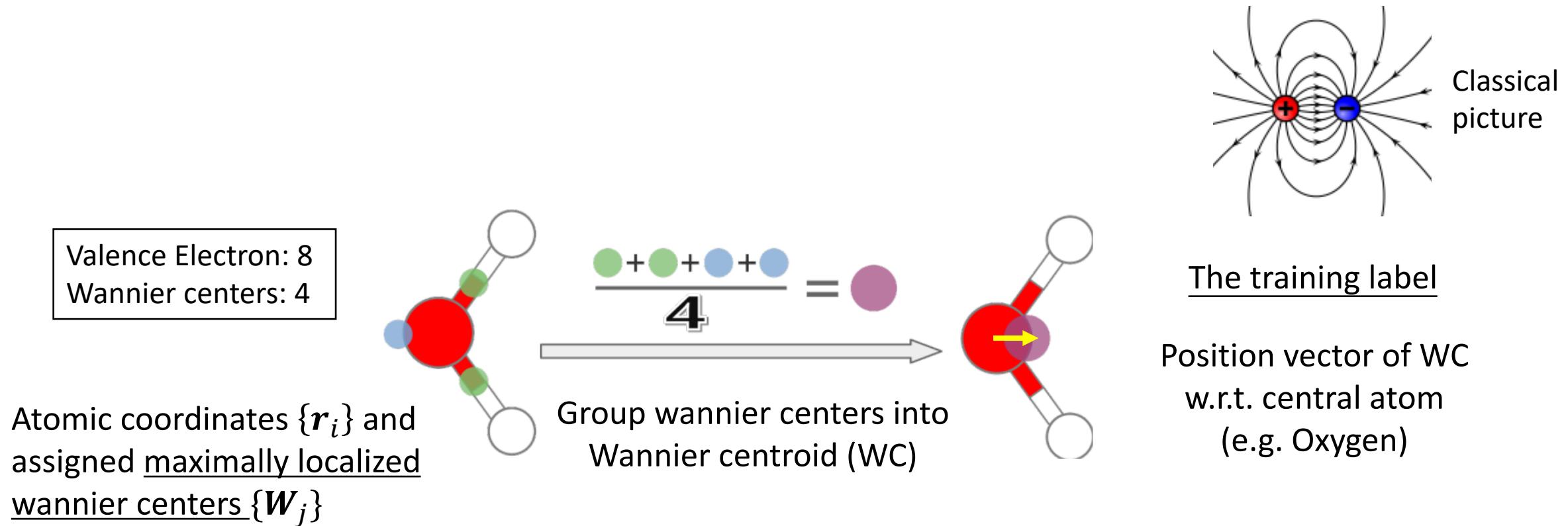
The black box



Zhang, Linfeng, et al. "Deep neural network for Wannier function centers." *arXiv preprint arXiv:1906.11434* (2019).
Zhang, Linfeng, et al. "Deep neural network for the dielectric response of insulators." *Physical Review B* 102.4 (2020): 041121.

A very short introduction to Deep Dipole

Example: the water molecule



A very short introduction to Deep Dipole

Implemented in Wannier90

Maximally localized Wannier functions (MLWF)

The General wannier function: $|\mathbf{R}n\rangle = \frac{V}{(2\pi)^3} \int_{BZ} d\mathbf{k} e^{-i\mathbf{k}\cdot\mathbf{R}} \sum_{m=1}^J U_{mn}^{(\mathbf{k})} |\psi_{m\mathbf{k}}\rangle$

↑
Arbitrary Unitary Rotation

← Bloch orbitals

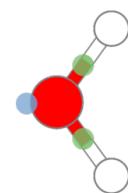
The MLWF: $|\mathbf{R}n\rangle = \frac{V}{(2\pi)^3} \int_{BZ} d\mathbf{k} e^{-i\mathbf{k}\cdot\mathbf{R}} \sum_{m=1}^J U_{mn}^{(\mathbf{k})} |\psi_{m\mathbf{k}}\rangle$

↑

The optimal rotation minimizing the total spreading of wannier functions

Total spreading $\Omega = \sum_n [\langle \mathbf{0}_n | r^2 | \mathbf{0}_n \rangle - \langle \mathbf{0}_n | \mathbf{r} | \mathbf{0}_n \rangle^2]$

$$= \sum_n [\langle r^2 \rangle_n - \bar{r}_n^2]$$



Hands-on Exercise 1 -- 30 minutes

Goal: Compute maximally localized wannier centers of given water configuration with Quantum-Espresso and Wannier90

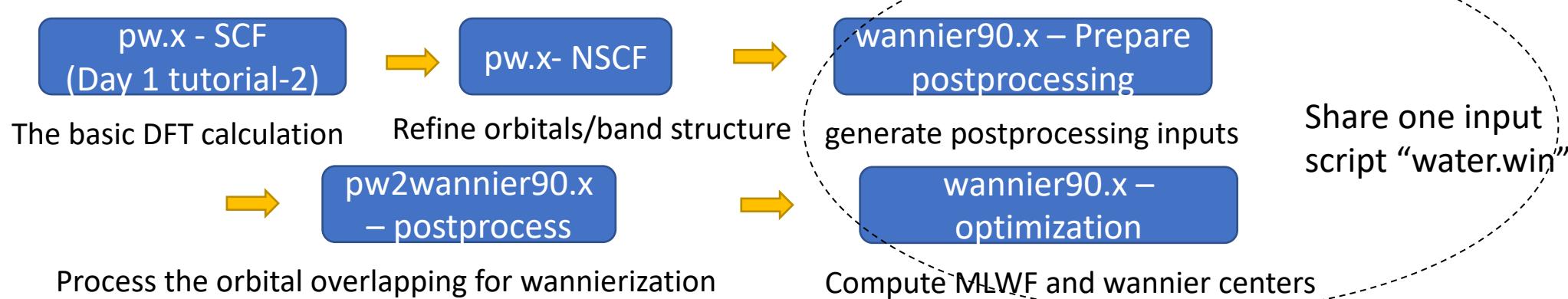
Folder: hands-on-sessions/day-3/7-deep-wannier/1-wannier90_example/ice_mlwf_exercise

How:

Use the newest version of “run.sh” on Github!! “Git pull” or just go to the webpage

- (1) Read and execute “run.sh” (better line by line so you can see the succession of outputs) ~ 10mins
- (2) Meanwhile, get familiar with the inputs&outputs
- (3) Visualize the final output “water_centres.xyz” with OVITO. Share on Slack if you want!

The workflow:



Hands-on Exercise 1 -- 30 minutes

Goal: Compute maximally localized wannier centers of given water configuration with Quantum-Espresso and Wannier90

Folder: hands-on-sessions/day-3/7-deep-wannier/1-wannier90 example/ice mlwf exercise

How:

- (1) Read and execute “run.sh” (better line by line so you can see the succession of outputs) ~ 10mins
- (2) Meanwhile, get familiar with the inputs&outputs
- (3) Visualize the final output “water_centres.xyz” with OVITO

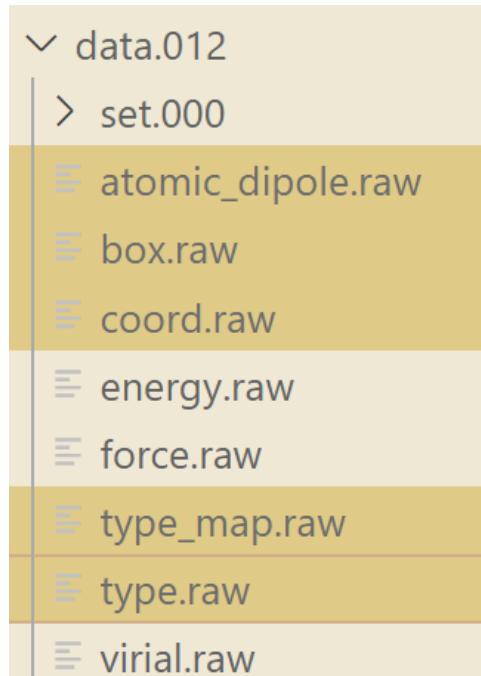
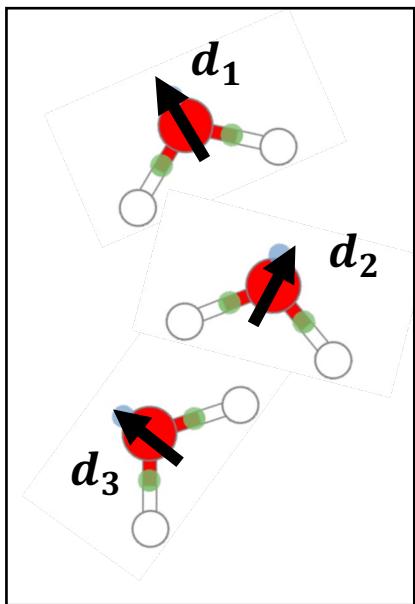
Basic setup of the wannier90 input

```
num_wann = 64    ← Number of wannier centers = #Occupied orbitals
num_iter = 100   ← Iterations for MLWF optimization. 100 is default. If not converging in 100 steps,
                  probably k-grid is too sparse.
write_xyz       = True
```

```
guiding_centres = .true. ← Provide an initial guess for the optimization problem
begin projections ← specify the atomic orbital as initial guess
  O: sp3
end projections
```

Training Deep Dipole: Dataset

- Required data



Optional: `dipole.raw`

- Format

`atomic_dipole.raw:(#frames, #sel_atom×3)`

Reference atoms for dipole moments,
e.g. oxygen for H₂O



Vector concatenated in the
same order as its reference atom
appearing in `coord.raw`

`dipole.raw: (#frames, 3) <- the sum of atomic
dipole`

`atomic_dipole.npy&dipole.npy:`

Same thing but in numpy format.

Training Deep Dipole: parameters

Difference between training Deep Potential & Deep Dipole

	fitting_net	loss
Energy model	<pre>"fitting_net": { "type": "ener", "neuron": [240, 240, 240], },</pre>	<pre>"loss": { "start_pref_e": 0.01, "limit_pref_e": 1, "start_pref_f": 100, "limit_pref_f": 1, "start_pref_v": 0, "limit_pref_v": 0, },</pre>
Dipole model	<pre>"fitting_net": { "type": "dipole", "sel_type": [0], "neuron": [100, 100, 100], },</pre>	<pre>"loss": { "type": "tensor", "pref": 1.0, ← 0.0 if there is no dipole.npy "pref_atomic": 1.0, },</pre>

Types of atom that carries a dipole.

Hands-on Exercise 2 -- 10 minutes

Goal:

- (1) Train a deep dipole with the dataset hands-on-sessions/day-2/7-deep-wannier/data

Folder: hands-on-sessions/day-3/7-deep-wannier/2-train_dipole_model

How:

- (1) Check “input.json”
- (2) Before training, examine dataset and verify that the number of columns in “atomic_dipole.raw” is compatible with "sel_type" and the number of water molecules.
- (3) Run “train.sh”. Check logs.
- (4) Terminate the training due to the lack of time. We provide the trained model in
hands-on-sessions/day-3/7-deep-wannier/dipole_model

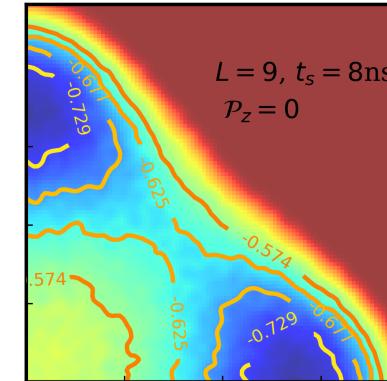
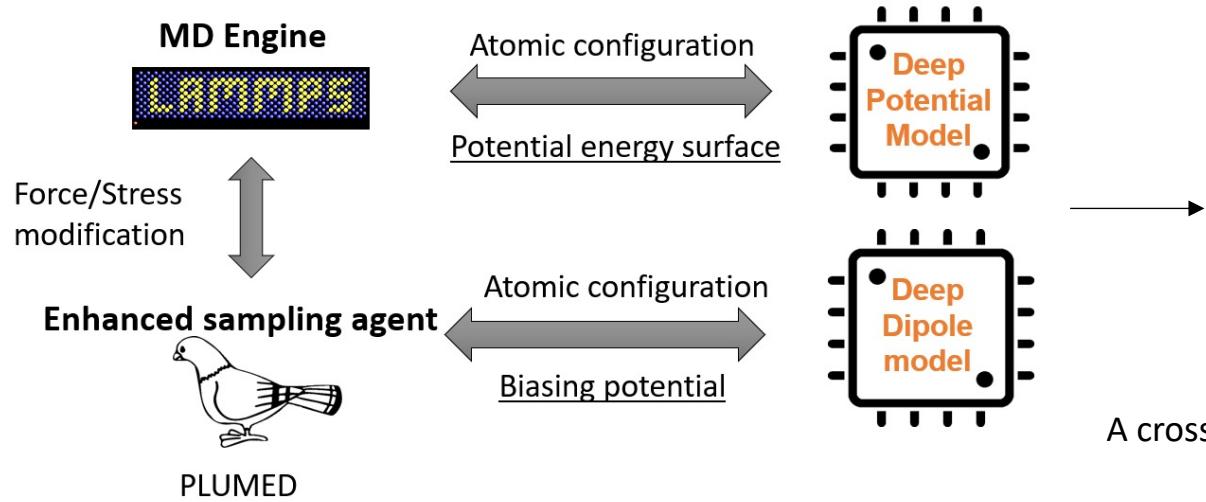
Application of Deep Dipole

- **Postprocessing (Today)**

After a regular DPMD simulation. Process the dumped trajectory with deep dipole model. You can compute dipole distribution / polarization/ dielectric response

- **Enhanced sampling**

Pair Lammmps up with PLUMED and use deep dipole model as a collective variable.



A cross section of the 3D free energy surface for the polarization of PbTiO_3 .

Reference:
arXiv preprint arXiv:2205.11839.
PLUMED plugin:
<https://github.com/y1xiaoc/deepmd-plumed>

Hands-on Exercise 3 -- 20 minutes

Goal:

- (1) Run a DPMD for liquid water with provided Deep Potential model. Then analyzed the dumped trajectory with Deep Dipole model.

Folder: hands-on-sessions/day-3/7-deep-wannier/3-MD_exercise

How:

- (1) Follow “run.sh”
- (2) Compute and plot the molecule dipole moments distribution with Python. Example script:

hands-on-sessions/day-3/7-deep-wannier/dipole_distribution.py

- (3) Experiment with “`dipole_model.eval`”!