Molecular Dynamics with DeePMD [DeepModeling Tutorial!]

Required Libraries : numpy, pandas, deepmd,dpdata, ase, matplotlib

This is a **DeePMD** tutorial, a Machine Learning Interatomic Potential (MLP) model for interatomic potential energy and force field to perform molecular dynamics (MD).

Example: **1,3-Butadiene cyclization.**

In practice, the construction of MLPs typically involves three main steps:

1. Data generation
2. Training
3. Molecular Dynamics

For data generation, we will use the VASP package.

Here, I will not go into the details for generating the initial configuration. However, there are several ways to do that. A straightforward approach involves performing a classical MD simulation to obtain random configurations. Subsequently, *ab-initio* calculations are performed to compute the potential energy and corresponding forces.

The output configurations are stored in the folder inside the *00.data* directory.

Note: Energies and forces for these configurations are obtained using PBE functional.

In [80]:
```
import os
import glob
import numpy as np
import pandas as pd
import dpdata
import ase
from ase.io import read, write
```

In [85]:
```
import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 150  # Set DPI for all figures
```

In [3]:
```
prefix_path = os.getcwd()
print(prefix_path)
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeePMD/Butadiene/from_traj/Tutorial
```

1. Data Generation

In [4]:
```
os.chdir(
    os.path.join(prefix_path, '00.data')
```

```
)
!pwd
!ls
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeeP
MD/Butadiene/from_traj/Tutorial/00.data
POSCAR              conf.lmp            training_data
btd_trj-nqp-alg.npz outcar_files        validation_data
```

We will use the dpdata package to extract the data from output files

In [28]:
```
# save LAMMPS input ('conf.lmp') from initial POSCAR file
d_poscar = dpdata.System("POSCAR", fmt="vasp/poscar")
d_poscar.to("lammps/lmp", "conf.lmp", frame_idx=0)
```

In [4]:
```
vasp_multi_system = dpdata.MultiSystems.from_dir(
    dir_name="./outcar_files", file_name="OUTCAR.*", fmt="vasp/outcar"
)
```

In [5]:
```
vasp_multi_system.systems['C4H6'].data['energies']
```

Out[5]:
```
array([-56.84108364, -56.84454623, -56.69026115, ..., -56.48688374,
       -56.35107127, -56.58241751])
```

In [6]:
```
print(vasp_multi_system.systems['C4H6'])

data = vasp_multi_system.systems['C4H6']
```

```
Data Summary
Labeled System
--------------------
Frame Numbers      : 4400
Atom Numbers       : 10
Including Virials  : Yes
Element List       :
--------------------
C  H
4  6
```

In [7]:
```
def vasp_dpmd_extract(dataset):

    # get total number of frames in dataset
    n_frames = dataset.get_nframes()
    print("# the data contains %d frames" % n_frames)
    # split training and validation data in 80% and 20%
    trr = int(n_frames*0.8)    # len -> training data
    val = n_frames - trr       # len -> validation data

    # random index for validation choice
    # rng = np.random.default_rng()
    # index_validation = rng.choice(n_frames, size=val, replace=False)
    index_validation = np.random.choice(n_frames, size=val, replace=False)

    # other indexes are training_data
    index_training = list(set(range(n_frames)) - set(index_validation))
    # extracting training and validation data
    data_training   = dataset.sub_system(index_training)
    data_validation = dataset.sub_system(index_validation)

    # all training data put into directory:"training_data"
    data_training.to_deepmd_npy("training_data")

    # all validation data put into directory:"validation_data"
    data_validation.to_deepmd_npy("validation_data")
```

```
    print("# the training data contains %d frames" % len(data_training))
    print("# the validation data contains %d frames" % len(data_validation))
```

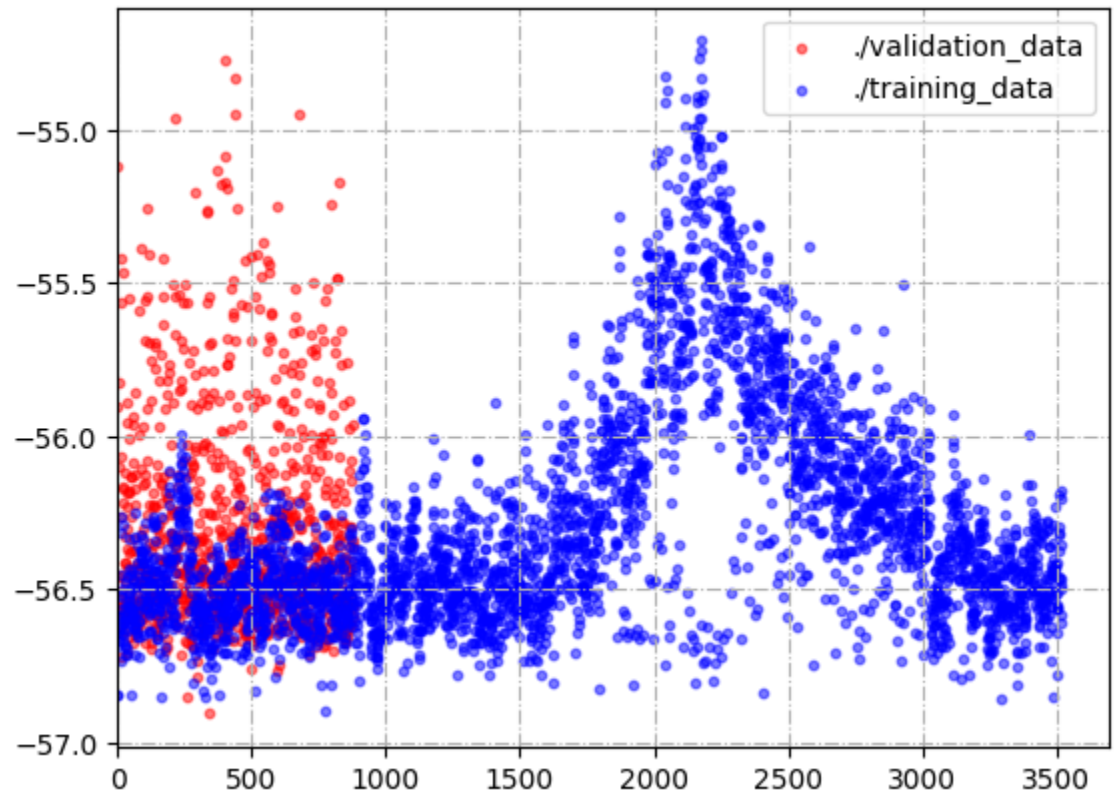
In [8]:
```
vasp_dpmd_extract(dataset=data)
```

```
# the data contains 4400 frames
# the training data contains 3520 frames
# the validation data contains 880 frames
```

In [5]:
```
def plot_data(dir_name):

    en_tr = np.load(f'{dir_name}/training_data/set.000/energy.npy')
    en_vl = np.load(f'{dir_name}/validation_data/set.000/energy.npy')

    print(en_tr.size,en_vl.size)

    plt.scatter(range(len(en_vl)), en_vl, s=10, color='red', alpha=0.5, label=f'{dir_nam
    plt.scatter(range(len(en_tr)), en_tr, s=10, color='blue', alpha=0.5, label=f'{dir_na

    plt.xlim(0,None)

    plt.legend()
    plt.grid(ls='-.')
    plt.show()
```

In [6]:
```
plot_data('.')
```

```
3520 880
```



2. Training (DeePMD)

#! Using the DFT energies and forces to train a model for potential energy surface (PES).

In [7]:
```
os.chdir(
```

```
    os.path.join(prefix_path, '01.train/')
)
!pwd
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeeP
MD/Butadiene/from_traj/Tutorial/01.train
```

Example of `input script` for the training:

- `model defintion`

```
{
  "_comment": " model parameters",
  "model": {
  "type_map": ["C", "H"],
  "descriptor" :{
      "type":     "se_e2_a",
      "sel":      "auto",
      "rcut_smth":    0.50,
      "rcut":     6.00,
      "neuron":       [20, 40, 80],
      "resnet_dt":    false,
      "axis_neuron":  16,
      "seed":     13245,
      "_comment":     " that's all"
  },
  "fitting_net" : {
      "neuron":       [100, 100, 100],
      "resnet_dt":    true,
      "seed":     13245,
      "_comment":     " that's all"
  },
  "_comment": " that's all"
  },
```

- `optimization process and loss function`

```
"learning_rate" :{
    "type":     "exp",
    "decay_steps":  5000,
    "start_lr": 0.001,
    "stop_lr":  3.51e-8,
    "_comment": "that's all"
    },

    "loss" :{
    "type":     "ener",
    "start_pref_e": 0.02,
    "limit_pref_e": 1,
    "start_pref_f": 1000,
    "limit_pref_f": 1,
    "start_pref_v": 0,
    "limit_pref_v": 0,
```

```
    "_comment": " that's all"
    },
```

- `training and validation data`

```
  "training" : {
    "training_data": {
        "systems":      ["../00.data/training_data"],
        "batch_size":   "auto",
        "_comment":     "that's all"
    },
    "validation_data":{
        "systems":      ["../00.data/validation_data"],
        "batch_size":   "auto",
        "numb_btch":    1,
        "_comment":     "that's all"
    },
    "numb_steps":   1000000,
    "seed":     13245,
    "disp_file":    "lcurve.out",
    "disp_freq":    1000,
    "save_freq":    10000,
    "_comment": "that's all"
    },

    "_comment":     "that's all"
  }
```

In [13]:
```
! dp train input.json
```

```
WARNING:tensorflow:From /Users/rverma7/anaconda3/envs/deepmd/lib/python3.11/site-package
s/tensorflow/python/compat/v2_compat.py:108: disable_resource_variables (from tensorflo
w.python.ops.variable_scope) is deprecated and will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
WARNING:root:To get the best performance, it is recommended to adjust the number of thre
ads by setting the environment variables OMP_NUM_THREADS, TF_INTRA_OP_PARALLELISM_THREAD
S, and TF_INTER_OP_PARALLELISM_THREADS. See https://deepmd.rtfd.io/parallelism/ for more
information.
DEEPMD INFO    Calculate neighbor statistics... (add --skip-neighbor-stat to skip this s
tep)
DEEPMD INFO    training data with min nbor dist: 0.9888341620312275
DEEPMD INFO    training data with max nbor size: [4 6]
DEEPMD INFO     _____               _____   __  __  _____           _     _  _
DEEPMD INFO    |  __ \             |  __ \ |  \/  ||  __ \         | |   (_)| |
DEEPMD INFO    | |  | |  ___   ___ | |__) || \  / || |  | | ______ | | __ _ | |_
DEEPMD INFO    | |  | | / _ \ / _ \|  ___/ | |\/| || |  | ||______|| |/ /| || __|
DEEPMD INFO    | |__| ||  __/|  __/| |     | |  | || |__| |        |   < | || |_
DEEPMD INFO    |_____/  \___| \___||_|     |_|  |_||_____/         |_|\_\|_| \__|
DEEPMD INFO    Please read and cite:
DEEPMD INFO    Wang, Zhang, Han and E, Comput.Phys.Comm. 228, 178-184 (2018)
DEEPMD INFO    Zeng et al, J. Chem. Phys., 159, 054801 (2023)
DEEPMD INFO    See https://deepmd.rtfd.io/credits/ for details.
DEEPMD INFO    installed to:         /usr/local
DEEPMD INFO    source :
DEEPMD INFO    source brach:
DEEPMD INFO    source commit:
DEEPMD INFO    source commit at:
DEEPMD INFO    build float prec:     double
DEEPMD INFO    build variant:        cpu
```

```
DEEPMD INFO    build with tf inc:    /Users/rverma7/anaconda3/envs/deepmd/lib/python3.1
1/site-packages/tensorflow/include;/Users/rverma7/anaconda3/envs/deepmd/lib/python3.11/s
ite-packages/tensorflow/../../../../include
DEEPMD INFO    build with tf lib:
DEEPMD INFO    ---Summary of the training---------------------------------------
DEEPMD INFO    running on:           CBE-LT-107
DEEPMD INFO    computing device:     cpu:0
DEEPMD INFO    Count of visible GPU: 0
DEEPMD INFO    num_intra_threads:    0
DEEPMD INFO    num_inter_threads:    0
DEEPMD INFO    -----------------------------------------------------------------
DEEPMD INFO    training without frame parameter
DEEPMD INFO    data stating... (this step may take long time)
DEEPMD INFO    built lr
DEEPMD INFO    built network
DEEPMD INFO    built training
WARNING:root:To get the best performance, it is recommended to adjust the number of thre
ads by setting the environment variables OMP_NUM_THREADS, TF_INTRA_OP_PARALLELISM_THREAD
S, and TF_INTER_OP_PARALLELISM_THREADS. See https://deepmd.rtfd.io/parallelism/ for more
information.
DEEPMD INFO    initialize model from scratch
DEEPMD INFO    start training at lr 1.00e-03 (== 1.00e-03), decay_step 5000, decay_rate
0.005925, final lr will be 3.51e-08
WARNING:tensorflow:From /Users/rverma7/anaconda3/envs/deepmd/lib/python3.11/site-package
s/deepmd/train/trainer.py:1198: py_func (from tensorflow.python.ops.script_ops) is depre
cated and will be removed in a future version.
Instructions for updating:
tf.py_func is deprecated in TF V2. Instead, there are two
    options available in V2.
    - tf.py_function takes a python function which manipulates tf eager
    tensors instead of numpy arrays. It's easy to convert a tf eager tensor to
    an ndarray (just call tensor.numpy()) but having access to eager tensors
    means `tf.py_function`s can use accelerators such as GPUs as well as
    being differentiable using a gradient tape.
    - tf.numpy_function maintains the semantics of the deprecated tf.py_func
    (it is not differentiable, and manipulates numpy arrays). It drops the
    stateful argument making all functions stateful.

WARNING:tensorflow:From /Users/rverma7/anaconda3/envs/deepmd/lib/python3.11/site-package
s/deepmd/train/trainer.py:1198: py_func (from tensorflow.python.ops.script_ops) is depre
cated and will be removed in a future version.
Instructions for updating:
tf.py_func is deprecated in TF V2. Instead, there are two
    options available in V2.
    - tf.py_function takes a python function which manipulates tf eager
    tensors instead of numpy arrays. It's easy to convert a tf eager tensor to
    an ndarray (just call tensor.numpy()) but having access to eager tensors
    means `tf.py_function`s can use accelerators such as GPUs as well as
    being differentiable using a gradient tape.
    - tf.numpy_function maintains the semantics of the deprecated tf.py_func
    (it is not differentiable, and manipulates numpy arrays). It drops the
    stateful argument making all functions stateful.

DEEPMD INFO    batch     500 training time 2.90 s, testing time 0.00 s, total wall time
3.10 s
DEEPMD INFO    batch    1000 training time 2.81 s, testing time 0.00 s, total wall time
2.86 s
DEEPMD INFO    batch    1500 training time 2.79 s, testing time 0.00 s, total wall time
2.84 s
DEEPMD INFO    batch    2000 training time 2.80 s, testing time 0.00 s, total wall time
2.85 s
DEEPMD INFO    batch    2500 training time 2.78 s, testing time 0.00 s, total wall time
2.83 s
DEEPMD INFO    batch    3000 training time 2.81 s, testing time 0.00 s, total wall time
2.86 s
DEEPMD INFO    batch    3500 training time 2.87 s, testing time 0.00 s, total wall time
```

```
2.92 s
DEEPMD INFO    batch    4000 training time 2.85 s, testing time 0.00 s, total wall time
2.90 s
DEEPMD INFO    batch    4500 training time 2.80 s, testing time 0.00 s, total wall time
2.85 s
DEEPMD INFO    batch    5000 training time 2.86 s, testing time 0.00 s, total wall time
2.91 s
DEEPMD INFO    batch    5500 training time 2.84 s, testing time 0.00 s, total wall time
2.89 s
DEEPMD INFO    batch    6000 training time 2.86 s, testing time 0.00 s, total wall time
2.91 s
DEEPMD INFO    batch    6500 training time 2.92 s, testing time 0.00 s, total wall time
2.97 s
DEEPMD INFO    batch    7000 training time 2.90 s, testing time 0.00 s, total wall time
2.95 s
DEEPMD INFO    batch    7500 training time 2.92 s, testing time 0.00 s, total wall time
2.97 s
DEEPMD INFO    batch    8000 training time 2.99 s, testing time 0.00 s, total wall time
3.04 s
DEEPMD INFO    batch    8500 training time 3.05 s, testing time 0.00 s, total wall time
3.11 s
DEEPMD INFO    batch    9000 training time 2.93 s, testing time 0.00 s, total wall time
2.98 s
DEEPMD INFO    batch    9500 training time 2.87 s, testing time 0.00 s, total wall time
2.92 s
DEEPMD INFO    batch   10000 training time 2.90 s, testing time 0.00 s, total wall time
2.95 s
DEEPMD INFO    saved checkpoint model.ckpt
DEEPMD INFO    average training time: 0.0057 s/batch (exclude first 500 batches)
DEEPMD INFO    finished training
DEEPMD INFO    wall time: 58.838 s
```

monitor the training process in the `lcurve.out` file

In [14]:
```
! head -n 4 lcurve.out
! tail -n 4 lcurve.out
```

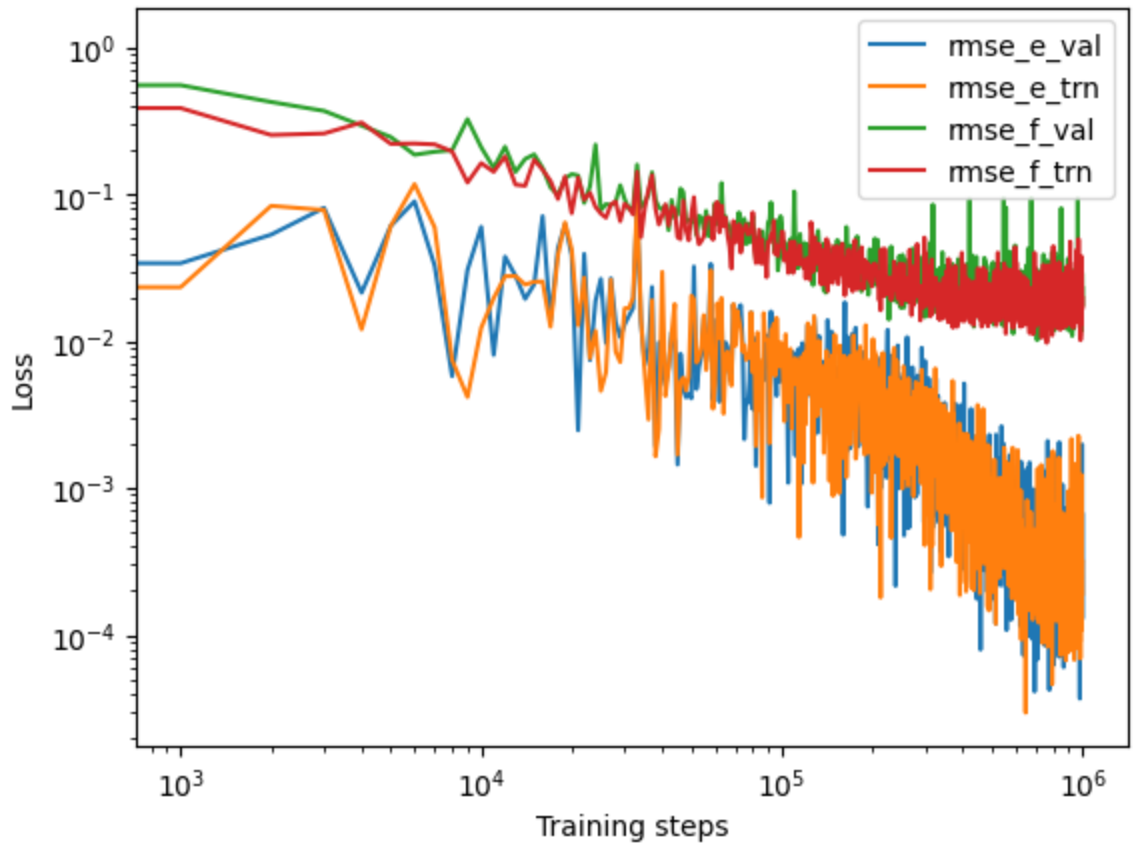
```
#  step      rmse_val    rmse_trn    rmse_e_val  rmse_e_trn    rmse_f_val  rmse_f_trn
      lr
# If there is no available reference data, rmse_*_{val,trn} will print nan
      0      2.90e+01    3.54e+01      8.36e-01    8.23e-01      9.16e-01    1.12e+00
1.0e-03
   1000      1.76e+01    1.23e+01      3.41e-02    2.34e-02      5.57e-01    3.89e-01
1.0e-03
 997000      3.04e-02    2.25e-02      1.99e-03    3.10e-04      2.92e-02    2.21e-02
3.7e-08
 998000      2.60e-02    1.87e-02      7.42e-04    3.25e-04      2.54e-02    1.83e-02
3.7e-08
 999000      1.77e-02    1.71e-02      1.29e-04    2.31e-04      1.74e-02    1.67e-02
3.7e-08
1000000      2.35e-02    2.01e-02      6.59e-04    1.92e-04      2.30e-02    1.98e-02
3.5e-08
```

We can track the progress of the training by plotting the number of steps against various metrics: total RMS error (validation and training), RMS error in energy, RMS error in forces, and learning rate.

In [18]:
```
def plot_lcurve(lcurve):
    with open(lcurve) as f:
        headers = f.readline().split()[1:]
    lcurve = pd.DataFrame(np.loadtxt(lcurve), columns=headers)
    legends = ["rmse_e_val", "rmse_e_trn", "rmse_f_val", "rmse_f_trn"]
    for legend in legends:
```

```
        plt.loglog(lcurve["step"], lcurve[legend], label=legend)
    plt.legend()
    plt.xlabel("Training steps")
    plt.ylabel("Loss")
    plt.show()
```

In [19]:
```
plot_lcurve(lcurve = "lcurve.out")
```



After completing the training, we can proceed to freeze the model using `dp freeze`. This will generate a deep potential file named `frozen_model.pb`. Now compress the model using `dp compress -t input.json -i frozen_model.pb -o frozen_model_1_compressed.pb`. This results in a compressed model (`frozen_model_1_compressed.pb`)

In [ ]:
```
! dp compress
```

In [16]:
```
! dp test -m frozen_model_1_compressed.pb -s ../00.data/validation_data/
```

```
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
tensorflow/python/compat/v2_compat.py:107: disable_resource_variables (from tensorflow.p
ython.ops.variable_scope) is deprecated and will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
WARNING:root:To get the best performance, it is recommended to adjust the number of thre
ads by setting the environment variables OMP_NUM_THREADS, TF_INTRA_OP_PARALLELISM_THREAD
S, and TF_INTER_OP_PARALLELISM_THREADS. See https://deepmd.rtfd.io/parallelism/ for more
information.
2024-05-22 18:55:16.971965: E tensorflow/core/framework/node_def_util.cc:630] NodeDef me
ntions attribute grad_y which is not in the op definition: Op<name=BatchMatMulV2; signat
ure=x:T, y:T -> output:T; attr=T:type,allowed=[DT_BFLOAT16, DT_HALF, DT_FLOAT, DT_DOUBL
E, DT_INT16, DT_INT32, DT_INT64, DT_COMPLEX64, DT_COMPLEX128]; attr=adj_x:bool,default=f
alse; attr=adj_y:bool,default=false> This may be expected if your graph generating binar
y is newer  than this binary. Unknown attributes will be ignored. NodeDef: {{node load/f
ilter_type_0/MatMul}}
```

```
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
deepmd/utils/batch_size.py:62: is_gpu_available (from tensorflow.python.framework.test_u
til) is deprecated and will be removed in a future version.
Instructions for updating:
Use `tf.config.list_physical_devices('GPU')` instead.
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
deepmd/utils/batch_size.py:62: is_gpu_available (from tensorflow.python.framework.test_u
til) is deprecated and will be removed in a future version.
Instructions for updating:
Use `tf.config.list_physical_devices('GPU')` instead.
DEEPMD WARNING You can use the environment variable DP_INFER_BATCH_SIZE tocontrol the in
ference batch size (nframes * natoms). The default value is 1024.
DEEPMD INFO    # ---------------output of dp test---------------
DEEPMD INFO    # testing system : ../00.data/validation_data
DEEPMD INFO    # number of test data : 880
DEEPMD INFO    Energy MAE         : 2.701914e-03 eV
DEEPMD INFO    Energy RMSE        : 5.314903e-03 eV
DEEPMD INFO    Energy MAE/Natoms  : 2.701914e-04 eV
DEEPMD INFO    Energy RMSE/Natoms : 5.314903e-04 eV
DEEPMD INFO    Force  MAE         : 1.458955e-02 eV/A
DEEPMD INFO    Force  RMSE        : 2.174797e-02 eV/A
DEEPMD INFO    Virial MAE         : 5.371615e-02 eV
DEEPMD INFO    Virial RMSE        : 7.890693e-02 eV
DEEPMD INFO    Virial MAE/Natoms  : 5.371615e-03 eV
DEEPMD INFO    Virial RMSE/Natoms : 7.890693e-03 eV
DEEPMD INFO    # -----------------------------------------------
```

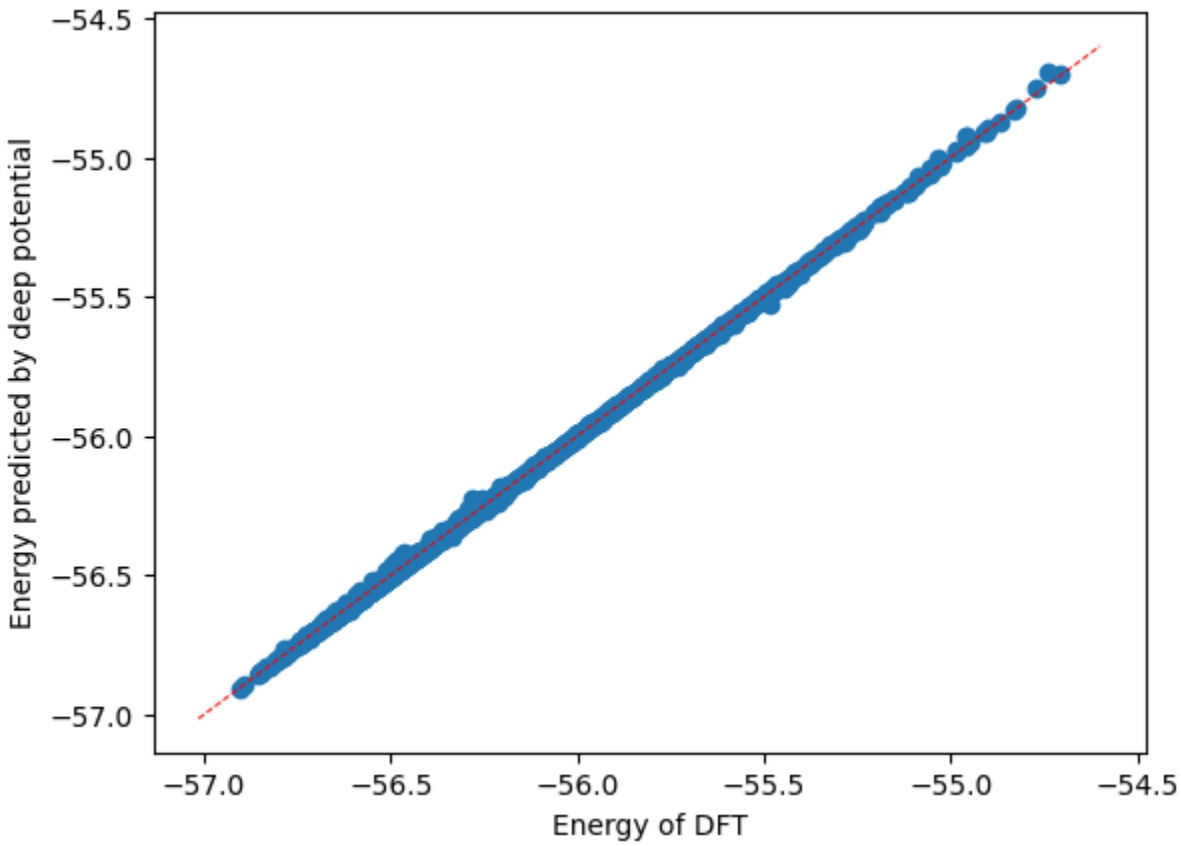
In [18]:
```
training_systems = dpdata.LabeledSystem("../00.data/training_data/", fmt="deepmd/npy", t
predict = training_systems.predict(dp = "frozen_model_1_compressed.pb")
```

```
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
tensorflow/python/compat/v2_compat.py:107: disable_resource_variables (from tensorflow.p
ython.ops.variable_scope) is deprecated and will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
WARNING:root:To get the best performance, it is recommended to adjust the number of thre
ads by setting the environment variables OMP_NUM_THREADS, TF_INTRA_OP_PARALLELISM_THREAD
S, and TF_INTER_OP_PARALLELISM_THREADS. See https://deepmd.rtfd.io/parallelism/ for more
information.
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
deepmd/utils/batch_size.py:62: is_gpu_available (from tensorflow.python.framework.test_u
til) is deprecated and will be removed in a future version.
Instructions for updating:
Use `tf.config.list_physical_devices('GPU')` instead.
2024-05-22 18:58:46.545002: E tensorflow/core/framework/node_def_util.cc:630] NodeDef me
ntions attribute grad_x which is not in the op definition: Op<name=BatchMatMulV2; signat
ure=x:T, y:T -> output:T; attr=T:type,allowed=[DT_BFLOAT16, DT_HALF, DT_FLOAT, DT_DOUBL
E, DT_INT16, DT_INT32, DT_INT64, DT_COMPLEX64, DT_COMPLEX128]; attr=adj_x:bool,default=f
alse; attr=adj_y:bool,default=false> This may be expected if your graph generating binar
y is newer  than this binary. Unknown attributes will be ignored. NodeDef: {{node load/f
ilter_type_0/MatMul}}
2024-05-22 18:58:46.650248: I tensorflow/core/platform/cpu_feature_guard.cc:193] This Te
nsorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the
following CPU instructions in performance-critical operations:  SSE4.1 SSE4.2 AVX AVX2 F
MA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler fla
gs.
2024-05-22 18:58:46.667034: I tensorflow/core/common_runtime/process_util.cc:146] Creati
ng new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_thr
eads for best performance.
2024-05-22 18:58:46.677082: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:3
54] MLIR V1 optimization pass is not enabled
WARNING:tensorflow:From /home/vrahul/anaconda3/envs/deepmd/lib/python3.10/site-packages/
deepmd/utils/batch_size.py:62: is_gpu_available (from tensorflow.python.framework.test_u
til) is deprecated and will be removed in a future version.
```

```
Instructions for updating:
Use `tf.config.list_physical_devices('GPU')` instead.
WARNING:deepmd.utils.batch_size:You can use the environment variable DP_INFER_BATCH_SIZE
tocontrol the inference batch size (nframes * natoms). The default value is 1024.
```

Parity plot

In [20]:
```
plt.scatter(training_systems["energies"], predict["energies"])
x_range = np.linspace(plt.xlim()[0], plt.xlim()[1])

plt.plot(x_range, x_range, "r--", linewidth=0.7)
plt.xlabel("Energy of DFT")
plt.ylabel("Energy predicted by deep potential")
plt.plot();
```



2. Run LAMMS molecular dynamics simulations

In [22]:
```
os.chdir(
    os.path.join(prefix_path, '02.lmp')
)
!pwd
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeeP
MD/Butadiene/from_traj/Tutorial/02.lmp
```

LAMMPS script to run dynamics:

```
# https://www.youtube.com/watch?
v=qsZQ2WPnScI&list=PLqT1fmMucKi712NWarqD4MLbDLH8hrmNj&index=3
# https://rb.gy/0vxw2q
# (1) Printing thermodynamic information
# (2) Printing coordinate information
# (3) Visualization of coordinates and some options in OVITO
(https://www.ovito.org)
```

```
# (4) Any line beginning with a # is a comment line
# (5) Any line ending with & means the command continues in the next line
# ========================================================================
# 1,3-Butadiene
units           metal
boundary        p p p
atom_style      atomic
# ========================================================================
neighbor        2.0 bin
neigh_modify    delay 0 every 1 check yes
# ========================================================================
read_data       conf.lmp
mass            1 12.0
mass            2 2
# ========================================================================
variable        sysvol          equal vol
variable        sysmass         equal mass(all)/6.0221367e+23
variable        sysdensity      equal v_sysmass/v_sysvol/1.0e-24
variable    out_freq    equal 100
# ========================================================================
pair_style      deepmd frozen_model_1_compressed.pb
frozen_model_2_compressed.pb frozen_model_3_compressed.pb
frozen_model_3_compressed.pb out_file md.out out_freq 10 atomic relative 1.0
pair_coeff  * *
# ========================================================================
#
velocity        all create 300.0 2345678 rot no dist gaussian
fix             1 all nvt temp 300.0 300.0 0.05
# ========================================================================
timestep        0.0005
thermo_style    custom step pe ke etotal v_sysdensity temp press vol
thermo          10
# ========================================================================
dump            10 all custom 100 btd.dump id type x y z
dump        101 all xyz 100 btd.xyz
dump_modify 101 element C H
# ========================================================================
fix     extra all print 20 "$(step), $(ke), $(pe), $(etotal), $(temp),
$(press), $(vol)" file logfile screen no
# ========================================================================
restart         100000 dump.restart

run             1000000
# ========================================================================
```

input code block for DeePMD model:

```
pair_style  deepmd frozen_model_1_compressed.pb frozen_model_2_compressed.pb
frozen_model_3_compressed.pb frozen_model_3_compressed.pb
pair_coeff  * *
```

Note: `frozen_model_#_compressed.pb` are four models trained on the same data and with different initial random seeds.

> conf.lmp

```bash

10 atoms 2 atom types 0.0000000000 15.0000000000 xlo xhi 0.0000000000 15.0000000000 ylo yhi 0.0000000000 15.0000000000 zlo zhi

Atoms # atomic

```
     1   1   8.0910000000   7.6280000000   5.7850000000
     2   1   7.2030000000   7.5070000000   6.8330000000
     3   1   7.8050000000   7.4730000000   8.1490000000
     4   1   6.9260000000   7.4360000000   9.2110000000
     5   2   7.7120000000   7.6050000000   4.7210000000
     6   2   9.1340000000   7.6360000000   6.0390000000
     7   2   6.0970000000   7.4940000000   6.6630000000
     8   2   8.9140000000   7.4840000000   8.3510000000
     9   2   5.8210000000   7.3900000000   9.0180000000
    10   2   7.2960000000   7.3470000000   10.2290000000
```

```

In [32]:
```
! lmp_mpi -in input.lammps
```

```
LAMMPS (2 Aug 2023)
OMP_NUM_THREADS environment is not set. Defaulting to 1 thread. (src/comm.cpp:98)
  using 1 OpenMP thread(s) per MPI task
Loaded 1 plugins from /Users/rverma7/anaconda3/envs/deepmd/lib/deepmd_lmp
Reading data file ...
  orthogonal box = (0 0 0) to (15 15 15)
  1 by 1 by 1 MPI processor grid
  reading atoms ...
  10 atoms
  read_data CPU = 0.002 seconds
DeePMD-kit WARNING: Environmental variable TF_INTRA_OP_PARALLELISM_THREADS is not set. T
une TF_INTRA_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
Summary of lammps deepmd module ...
  >>> Info of deepmd-kit:
  installed to:       /Users/rverma7/anaconda3/envs/deepmdDeePMD-kit WARNING: Environmen
tal variable TF_INTRA_OP_PARALLELISM_THREADS is not set. Tune TF_INTRA_OP_PARALLELISM_TH
READS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informa
tion.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
2024-06-28 20:20:05.272002: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:3
88] MLIR V1 optimization pass is not enabled
INVALID_ARGUMENT: Tensor spin_attr/ntypes_spin:0, specified in either feed_devices or fe
tch_devices was not found in the Graph
DeePMD-kit WARNING: Environmental variable TF_INTRA_OP_PARALLELISM_THREADS is not set. T
une TF_INTRA_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
```

```
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
INVALID_ARGUMENT: Tensor spin_attr/ntypes_spin:0, specified in either feed_devices or fe
tch_devices was not found in the Graph
DeePMD-kit WARNING: Environmental variable TF_INTRA_OP_PARALLELISM_THREADS is not set. T
une TF_INTRA_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
INVALID_ARGUMENT: Tensor spin_attr/ntypes_spin:0, specified in either feed_devices or fe
tch_devices was not found in the Graph
DeePMD-kit WARNING: Environmental variable TF_INTRA_OP_PARALLELISM_THREADS is not set. T
une TF_INTRA_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
INVALID_ARGUMENT: Tensor spin_attr/ntypes_spin:0, specified in either feed_devices or fe
tch_devices was not found in the Graph
DeePMD-kit WARNING: Environmental variable TF_INTRA_OP_PARALLELISM_THREADS is not set. T
une TF_INTRA_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable TF_INTER_OP_PARALLELISM_THREADS is not set. T
une TF_INTER_OP_PARALLELISM_THREADS for the best performance. See https://deepmd.rtfd.i
o/parallelism/ for more information.
DeePMD-kit WARNING: Environmental variable OMP_NUM_THREADS is not set. Tune OMP_NUM_THRE
ADS for the best performance. See https://deepmd.rtfd.io/parallelism/ for more informati
on.
INVALID_ARGUMENT: Tensor spin_attr/ntypes_spin:0, specified in either feed_devices or fe
tch_devices was not found in the Graph
  >>> Info of model(s):
  using   4 model(s): frozen_model_1_compressed.pb frozen_model_2_compressed.pb frozen_m
odel_3_compressed.pb frozen_model_3_compressed.pb
  rcut in model:      6
  ntypes in model:    2

CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE

Your simulation uses code contributions which should be cited:
- USER-DEEPMD package:
The log file lists these citations in BibTeX format.

CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE-CITE

Generated 0 of 1 mixed pair_coeff terms from geometric mixing rule
Neighbor list info ...
  update: every = 1 steps, delay = 0 steps, check = yes
  max neighbors/atom: 2000, page size: 100000
  master list distance cutoff = 8
  ghost atom cutoff = 8
  binsize = 4, bins = 4 4 4
  1 neighbor lists, perpetual/occasional/extra = 1 0 0
  (1) pair deepmd, perpetual
      attributes: full, newton on
      pair build: full/bin/atomonly
      stencil: full/bin/3d
      bin: standard
```

```
Setting up Verlet run ...
  Unit style    : metal
  Current step  : 0
  Time step     : 0.0005
2024-06-28 20:20:05.704789: E tensorflow/core/framework/node_def_util.cc:676] NodeDef me
ntions attribute grad_x which is not in the op definition: Op<name=BatchMatMulV2; signat
ure=x:T, y:T -> output:T; attr=T:type,allowed=[DT_BFLOAT16, DT_HALF, DT_FLOAT, DT_DOUBL
E, DT_INT16, DT_INT32, DT_INT64, DT_UINT8, DT_UINT16, DT_UINT32, DT_UINT64, DT_COMPLEX6
4, DT_COMPLEX128]; attr=adj_x:bool,default=false; attr=adj_y:bool,default=false> This ma
y be expected if your graph generating binary is newer  than this binary. Unknown attrib
utes will be ignored. NodeDef: {{node filter_type_1/MatMul}}
Per MPI rank memory allocation (min/avg/max) = 3.06 | 3.06 | 3.06 Mbytes
   Step         PotEng         KinEng         TotEng      v_sysdensity       Temp
   Press          Volume
         0  -56.646382      0.34900239    -56.29738       0.029520714    300           -
950.35539      3375
        10  -56.692944      0.3946739     -56.29827       0.029520714    339.25891
1194.4948      3375
        20  -56.573745      0.27659251    -56.297153      0.029520714    237.75697
2322.5319      3375
        30  -56.807479      0.50908415    -56.298395      0.029520714    437.60516     -
34.258286      3375
        40  -56.721491      0.41971897    -56.301772      0.029520714    360.78748     -
1315.5061      3375
        50  -56.671259      0.36522447    -56.306034      0.029520714    313.94438     -
496.45288      3375
        60  -56.645606      0.33461014    -56.310996      0.029520714    287.62852
36.640762      3375
        70  -56.847242      0.52701327    -56.320229      0.029520714    453.0169      -
222.65262      3375
        80  -56.499776      0.17309985    -56.326676      0.029520714    148.79541     -
322.09278      3375
        90  -56.722877      0.38987966    -56.332997      0.029520714    335.13782     -
186.80254      3375
       100  -56.719406      0.3767245     -56.342681      0.029520714    323.82973
411.19797      3375
       110  -56.750495      0.39748884    -56.353007      0.029520714    341.67861     -
183.63458      3375
       120  -56.657131      0.29271695    -56.364414      0.029520714    251.61743     -
1280.3085      3375
       130  -56.790176      0.41299214    -56.377184      0.029520714    355.00514     -
361.77334      3375
       140  -56.775318      0.37863664    -56.396682      0.029520714    325.47339
1571.3121      3375
       150  -56.710939      0.29871683    -56.412222      0.029520714    256.77489
905.73524      3375
       160  -56.662404      0.23272389    -56.42968       0.029520714    200.04782     -
740.82792      3375
       170  -56.722266      0.27982409    -56.442441      0.029520714    240.53482     -
489.95488      3375
       180  -56.833734      0.36788797    -56.465846      0.029520714    316.23391
737.11491      3375
       190  -56.736581      0.25335356    -56.483227      0.029520714    217.78094
1316.6973      3375
       200  -56.814998      0.31052813    -56.50447       0.029520714    266.92779
324.67438      3375
       210  -56.803047      0.27780233    -56.525245      0.029520714    238.79693     -
1169.1291      3375
       220  -56.81731       0.27292999    -56.54438       0.029520714    234.6087      -
605.02472      3375
       230  -56.707388      0.1508829     -56.556505      0.029520714    129.69788
518.71494      3375
       240  -56.841525      0.27165197    -56.569873      0.029520714    233.51012     -
512.59366      3375
       250  -56.771003      0.19054132    -56.580462      0.029520714    163.78798     -
836.26215      3375
```

260	-56.840124	0.25180484	-56.588319	0.029520714	216.44967	-41.520173	3375
270	-56.78089	0.18457252	-56.596317	0.029520714	158.65724	163.93396	3375
280	-56.887671	0.28576355	-56.601907	0.029520714	245.64034	52.956353	3375
290	-56.833084	0.22710208	-56.605982	0.029520714	195.21535	-179.97569	3375
300	-56.806612	0.19837103	-56.608241	0.029520714	170.51834	39.696392	3375
310	-56.796633	0.18738707	-56.609246	0.029520714	161.07661	1421.5992	3375
320	-56.843682	0.23537407	-56.608308	0.029520714	202.32589	895.14252	3375
330	-56.775	0.17095329	-56.604047	0.029520714	146.95025	-1107.7654	3375
340	-56.810292	0.21007497	-56.600217	0.029520714	180.57896	-846.69608	3375
350	-56.868631	0.27590806	-56.592723	0.029520714	237.16862	541.0767	3375
360	-56.798929	0.21657495	-56.582354	0.029520714	186.1663	669.20776	3375
370	-56.868016	0.3000958	-56.56792	0.029520714	257.96024	193.61338	3375
380	-56.754521	0.19843993	-56.556082	0.029520714	170.57757	-504.63438	3375
390	-56.760238	0.21872535	-56.541513	0.029520714	188.01477	-613.87232	3375
400	-56.75567	0.23286021	-56.522809	0.029520714	200.165	349.84204	3375
410	-56.854881	0.35662503	-56.498256	0.029520714	306.55237	-277.36086	3375
420	-56.636257	0.16022277	-56.476035	0.029520714	137.72637	-1227.9715	3375
430	-56.869792	0.41804011	-56.451752	0.029520714	359.34434	-105.58182	3375
440	-56.810046	0.39079146	-56.419255	0.029520714	335.92159	850.70545	3375
450	-56.708861	0.30876063	-56.4001	0.029520714	265.40846	-219.11164	3375
460	-56.61728	0.23251739	-56.384763	0.029520714	199.87031	-316.26439	3375
470	-56.765146	0.39114983	-56.373997	0.029520714	336.22964	667.1294	3375
480	-56.640011	0.27788144	-56.362129	0.029520714	238.86493	1602.4513	3375
490	-56.710132	0.35477414	-56.355358	0.029520714	304.96135	1413.0217	3375
500	-56.765815	0.41797102	-56.347844	0.029520714	359.28495	-876.84319	3375
510	-56.723333	0.38045565	-56.342877	0.029520714	327.037	-1922.4822	3375
520	-56.747844	0.40797971	-56.339865	0.029520714	350.69649	392.51873	3375
530	-56.63709	0.29874172	-56.338348	0.029520714	256.79628	955.0741	3375
540	-56.757056	0.42066271	-56.336393	0.029520714	361.59871	-845.15412	3375
550	-56.63656	0.30180857	-56.334752	0.029520714	259.43252	-794.44437	3375
560	-56.690203	0.35585344	-56.33435	0.029520714	305.88911	-551.17008	3375
570	-56.588605	0.25693672	-56.331668	0.029520714	220.861	70.217325	3375
580	-56.893428	0.56384238	-56.329585	0.029520714	484.67494	688.93189	3375

590	-56.700306	0.37020466	-56.330101	0.029520714	318.22532	-415.5865	3375
600	-56.754044	0.42171001	-56.332334	0.029520714	362.49896	-243.4885	3375
610	-56.594503	0.26029582	-56.334207	0.029520714	223.74845	1460.8462	3375
620	-56.735251	0.39827715	-56.336974	0.029520714	342.35624	-49.740075	3375
630	-56.645664	0.30712821	-56.338536	0.029520714	264.00525	-1164.4551	3375
640	-56.671394	0.33150091	-56.339893	0.029520714	284.95585	395.21247	3375
650	-56.714949	0.37275759	-56.342192	0.029520714	320.4198	1381.9229	3375
660	-56.80407	0.45988464	-56.344185	0.029520714	395.3136	823.37038	3375
670	-56.721085	0.37210615	-56.348979	0.029520714	319.85983	-631.51927	3375
680	-56.661484	0.30838098	-56.353103	0.029520714	265.08212	-1465.4376	3375
690	-56.780779	0.42179042	-56.358989	0.029520714	362.56807	169.83529	3375
700	-56.61026	0.24734706	-56.362912	0.029520714	212.61779	1207.067	3375
710	-56.735071	0.36646856	-56.368603	0.029520714	315.0138	-715.36615	3375
720	-56.663065	0.29094245	-56.372123	0.029520714	250.09208	-1635.3566	3375
730	-56.819343	0.44173674	-56.377607	0.029520714	379.7138	-501.81812	3375
740	-56.672659	0.28875402	-56.383905	0.029520714	248.21093	650.94075	3375
750	-56.819857	0.42874156	-56.391116	0.029520714	368.54323	507.86052	3375
760	-56.623966	0.22696272	-56.397004	0.029520714	195.09556	204.28569	3375
770	-56.856832	0.45222495	-56.404607	0.029520714	388.72938	222.59448	3375
780	-56.664073	0.25092312	-56.41315	0.029520714	215.69174	791.21654	3375
790	-56.679531	0.26145313	-56.418078	0.029520714	224.74327	166.85276	3375
800	-56.70343	0.27996518	-56.423464	0.029520714	240.6561	-1037.4053	3375
810	-56.834137	0.40469165	-56.429445	0.029520714	347.8701	-227.64203	3375
820	-56.749788	0.31226435	-56.437524	0.029520714	268.42024	1659.5484	3375
830	-56.764284	0.32107305	-56.443211	0.029520714	275.99213	679.80836	3375
840	-56.784566	0.33285153	-56.451715	0.029520714	286.11683	-1266.4224	3375
850	-56.702782	0.24571243	-56.457069	0.029520714	211.21267	-1203.569	3375
860	-56.747973	0.2845047	-56.463468	0.029520714	244.55824	-140.32229	3375
870	-56.676254	0.20941915	-56.466835	0.029520714	180.01522	888.55258	3375
880	-56.861563	0.38992909	-56.471634	0.029520714	335.1803	181.20898	3375
890	-56.706858	0.23169909	-56.475159	0.029520714	199.1669	-1274.6198	3375
900	-56.818265	0.33907008	-56.479195	0.029520714	291.46225	-520.11124	3375
910	-56.734345	0.25298141	-56.481363	0.029520714	217.46104	903.29398	3375

```
       920  -56.892408      0.40771626    -56.484692      0.029520714    350.47003
143.24506      3375
       930  -56.635392      0.14867123    -56.48672       0.029520714    127.79674     -
121.87574      3375
       940  -56.794124      0.30642417    -56.4877        0.029520714    263.40006
582.40766      3375
       950  -56.712521      0.22689483    -56.485626      0.029520714    195.0372
897.619        3375
       960  -56.806617      0.32278877    -56.483828      0.029520714    277.46695
94.583282      3375
       970  -56.749233      0.26986026    -56.479372      0.029520714    231.96998     -
864.66172      3375
       980  -56.825422      0.35142865    -56.473994      0.029520714    302.08559     -
425.64032      3375
       990  -56.782429      0.31754268    -56.464886      0.029520714    272.95745
1250.6998      3375
      1000  -56.776663      0.3183366     -56.458327      0.029520714    273.6399
689.77339      3375
Loop time of 2.03617 on 1 procs for 1000 steps with 10 atoms

Performance: 21.216 ns/day, 1.131 hours/ns, 491.119 timesteps/s, 4.911 katom-step/s
122.4% CPU use with 1 MPI tasks x 1 OpenMP threads

MPI task timing breakdown:
Section |  min time  |  avg time  |  max time  |%varavg| %total
---------------------------------------------------------------
Pair    | 2.0293     | 2.0293     | 2.0293     |   0.0 | 99.66
Neigh   | 3.5959e-05 | 3.5959e-05 | 3.5959e-05 |   0.0 |  0.00
Comm    | 0.00080595 | 0.00080595 | 0.00080595 |   0.0 |  0.04
Output  | 0.0028623  | 0.0028623  | 0.0028623  |   0.0 |  0.14
Modify  | 0.0026087  | 0.0026087  | 0.0026087  |   0.0 |  0.13
Other   |            | 0.0005675  |            |       |  0.03

Nlocal:             10 ave          10 max          10 min
Histogram: 1 0 0 0 0 0 0 0 0 0
Nghost:            138 ave         138 max         138 min
Histogram: 1 0 0 0 0 0 0 0 0 0
Neighs:              0 ave           0 max           0 min
Histogram: 1 0 0 0 0 0 0 0 0 0
FullNghs:           90 ave          90 max          90 min
Histogram: 1 0 0 0 0 0 0 0 0 0

Total # of neighbors = 90
Ave neighs/atom = 9
Neighbor list builds = 6
Dangerous builds = 0
Total wall time: 0:00:03
```

In [29]:

```python
def plot_lammps(x, y, xlabel, ylabel, label):

    plt.figure(figsize=(8, 5))

    mean = np.mean(y)
    plt.plot(x, y, label=f'{label}', color='blue', lw=1, alpha=0.7)
    plt.axhline(y=mean, color='red', linestyle='--', label=f'Mean {label} = {mean:.2f}')
#
    # Add labels and legend
    plt.xlabel(f'{xlabel}')
    plt.ylabel(f'{ylabel}')

    plt.xlim(min(x), max(x))

#    plt.title(f'{label}')
    plt.grid(ls='-.')
    plt.legend()
```

```
    plt.tight_layout()
    plt.show()
#    plt.plot(log.get("Step"), log.get("Temp"))
```

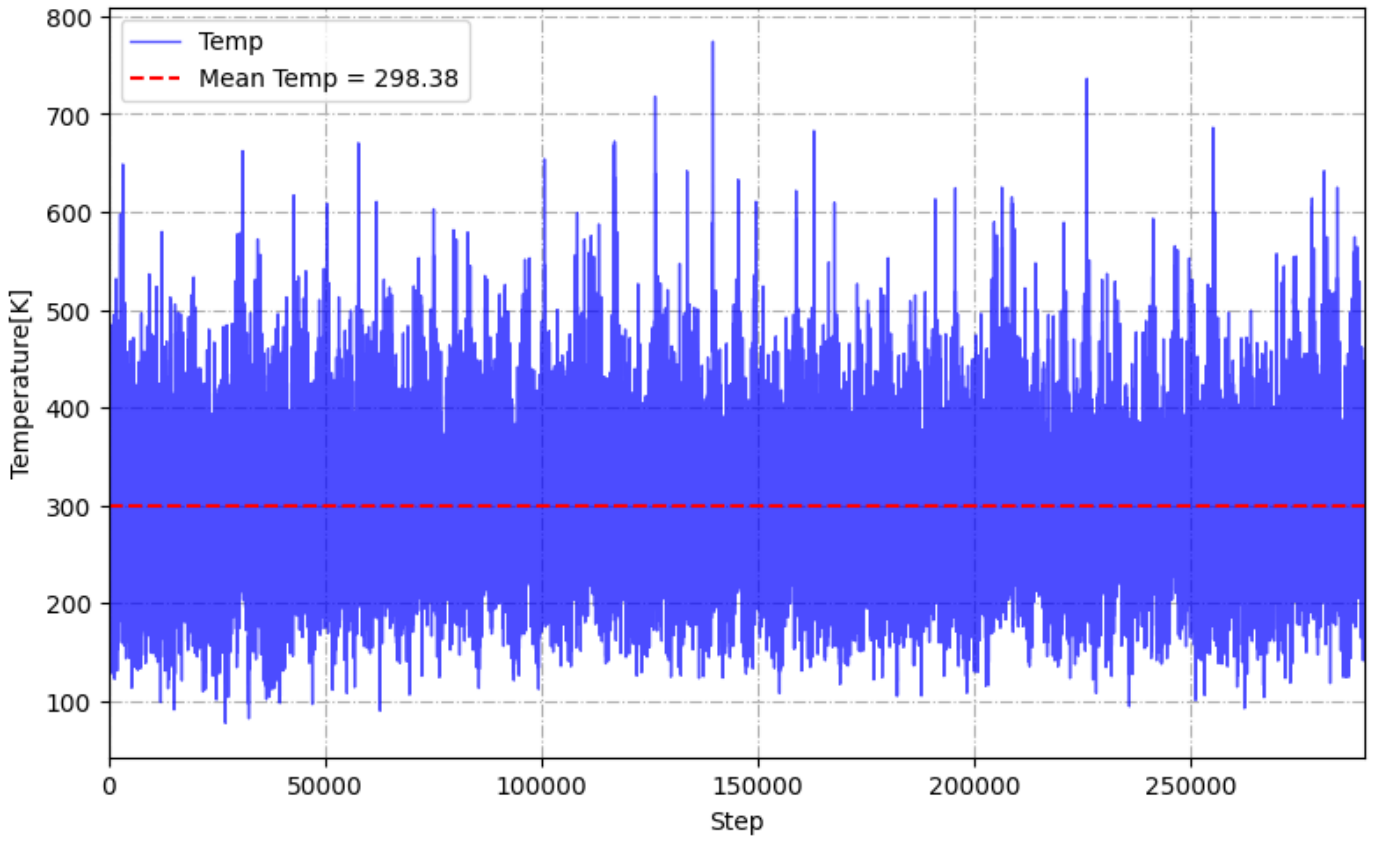
[Lammps log file parser](https://github.com/henriasv/lammps-logfile):

```
!pip install lammps-logfile
```

In [26]:
```
import lammps_logfile as lmp_log

log = lmp_log.File('log.lammps')

print("Log keywords: ", log.get_keywords())
```

```
Log keywords:  ['KinEng', 'PotEng', 'Press', 'Step', 'Temp', 'TotEng', 'Volume', 'v_sysd
ensity']
```

In [16]:
```
# Plot Temperature over the course of simulation
step = log.get("Step")
temp = log.get("Temp")

plot_lammps(x=step, y=temp, xlabel='Step', ylabel='Temperature[K]', label='Temp')
```



3. Enhance sampling MD simulations using [PLUMED] (https://www.plumed.org//)

We will perform Umbrella sampling simulation.

In [38]:
```
os.chdir(
    os.path.join(prefix_path, '03.plumed')
)
!pwd
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeeP
```

MD/Butadiene/from_traj/Tutorial/03.plumed

> plumed.dat

```
# PLUMED FILE
# ------------------------------------
#       UNITS
# ------------------------------------
UNITS LENGTH=A ENERGY=kcal/mol
#
d1: DISTANCE ATOMS=1,4
d2: DISTANCE ATOMS=2,1
d3: DISTANCE ATOMS=3,2
d4: DISTANCE ATOMS=4,3
#
# Activate Umbrella Sampling in distance (butadiene)
# with kapps equal to 0.4au/Bohr (224.1 kcal/mol/Ang^2), distance equal
Angstrom
#

rest: RESTRAINT ARG=d1 KAPPA=896.4 AT=1.3

FLUSH STRIDE=100

PRINT ARG=d1,d2,d3,d4,rest.bias STRIDE=100 FILE=COLVAR
```

We need to add the following line in the LAMMPS input file.

```
fix    2 all plumed plumedfile plumed.dat outfile PLUMED.log
```

In [53]:
```
from scipy.interpolate import interp1d
```

In [43]:
```
os.chdir(
    os.path.join(prefix_path, '03.plumed', 'analysis')
)
!pwd
```

```
/Users/rverma7/Library/CloudStorage/OneDrive-NorthCarolinaStateUniversity/NCSU_Work/DeeP
MD/Butadiene/from_traj/Tutorial/03.plumed/analysis
```

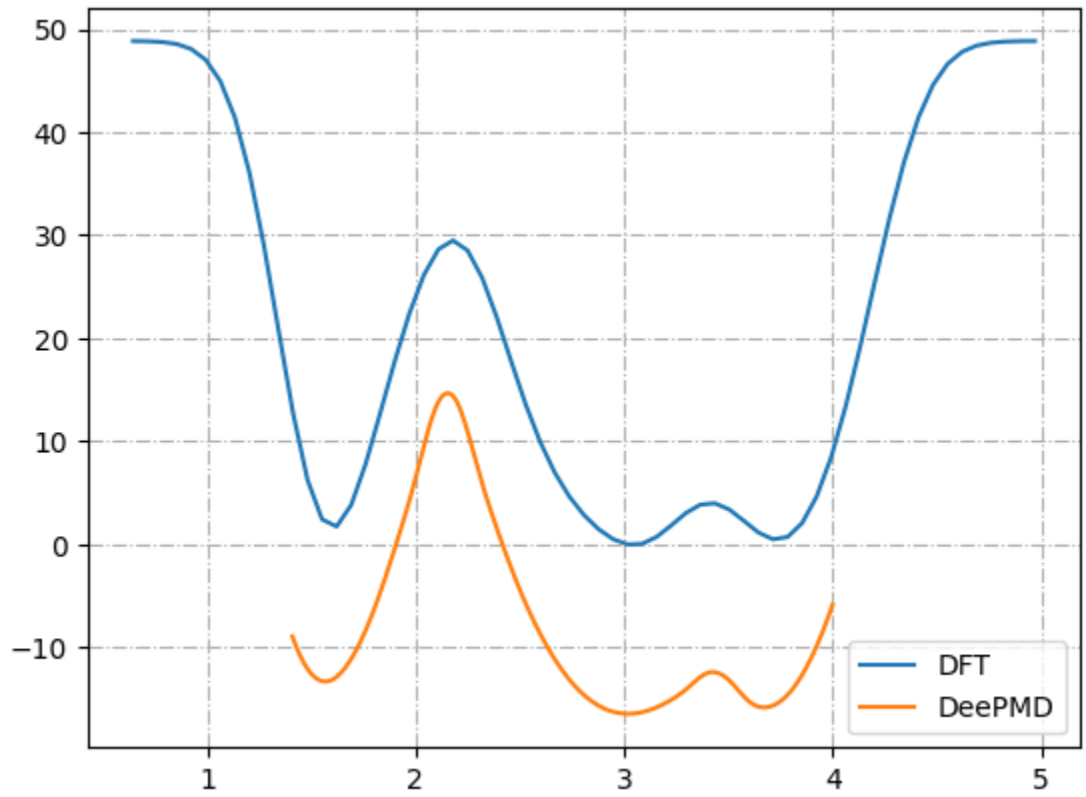
we will be using Reweighting-Package to reconstruct the free energy profile from PLUMED run

In [72]:
```
# Load the data
dft_fes_data = np.loadtxt('aimd_fes.dat', comments='!', skiprows=5, usecols=(0, 1))
dpmd_fes_data = np.loadtxt('interp_free_energy.dat', usecols=(0, 1))

#
print(f"DFT: \n {dft_fes_data[1:5]}, \n DeePMD: \n {dpmd_fes_data[1:5]}")
```

```
DFT:
 [[ 0.7148395  48.8225759 ]
 [ 0.78459    48.75265659]
 [ 0.8543405  48.557655  ]
 [ 0.924091   48.07544632]],
 DeePMD:
 [[  1.41857143  -9.44430566]
 [  1.42785714  -9.92960753]
 [  1.43714286 -10.38017597]
 [  1.44642857 -10.79645904]]
```

In [73]:
```
# Separate positions and PMFs
dft_fes_pos, dft_fes_pmf = dft_fes_data[:, 0], dft_fes_data[:, 1]
dpmd_fes_pos, dpmd_fes_pmf = dpmd_fes_data[:, 0], dpmd_fes_data[:, 1]
```

In [75]:
```
plt.plot(dft_fes_pos, dft_fes_pmf, label='DFT')
plt.plot(dpmd_fes_pos, dpmd_fes_pmf, label='DeePMD')

plt.legend()
plt.grid(ls='-.')
plt.show()

plt.show()
```



Info: Both plots are shifted along the Y-axis. Let's align these with respect to a certain value of X.
Also, since the `DeePMD` has no data after a certain X-range, we will create a mask to filter out data outside this range: `[xmin: 1.4, xmax: 4.0]`

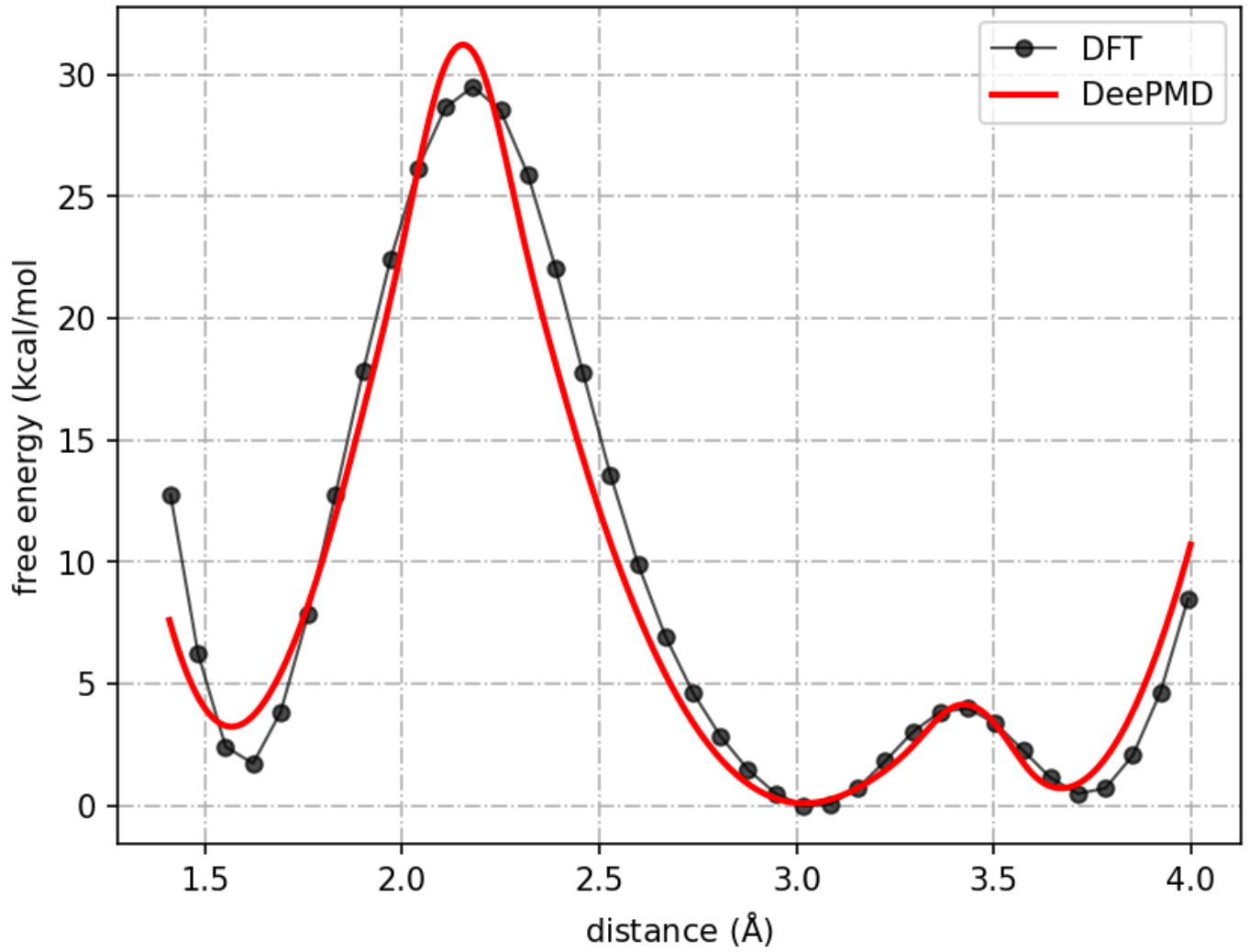
In [48]:
```
def align_fes(dft_pos, dft_pmf, ml_pos, ml_pmf, pos_x=3.0):

    # Min and Max position at which PMF is considerd
    min_pos, max_pos = 1.4, 4.0
    # min_pos = min_x
    # max_pos = max_x

    # create a mask array to filter the positions outside of range (type[boolean: True o
    dft_mask = (dft_pos >= min_pos) & (dft_pos <= max_pos)
    ml_mask = (ml_pos >= min_pos) & (ml_pos <= max_pos)

    # Filter the positions outside of the range
    dft_fes_pos = dft_pos[dft_mask]
    dft_fes_pmf = dft_pmf[dft_mask]
    ml_fes_pos = ml_pos[ml_mask]
    ml_fes_pmf = ml_pmf[ml_mask]
```

```
    # Find the PMF values at position 3.0 for both datasets
    pos_x = pos_x
    dft_pmf_at_position = np.interp(pos_x, dft_fes_pos, dft_fes_pmf)
    ml_pmf_at_position = np.interp(pos_x, ml_fes_pos, ml_fes_pmf)

    # Calculate the difference at the specific position
    alignment_shift = dft_pmf_at_position - ml_pmf_at_position

    # Align dpmd_fes_pmf with dft_fes_pmf at position 3.0
    ml_fes_pmf += alignment_shift

    return dft_fes_pos, dft_fes_pmf, ml_fes_pos, ml_fes_pmf
```

In [49]:
```
dft_pos, dft_pmf, dpmd_pos, dpmd_pmf = align_fes(dft_fes_pos, dft_fes_pmf, dpmd_fes_pos,
```

In [96]:
```
def plt_fes(dft_pos, dft_pmf, ml_pos, ml_pmf, c1='red', c2='blue', model_1='DFT', ml_mod

    plt.plot(dft_pos, dft_pmf, lw=1, c=f'{c1}', label=f'{model_1}', marker='o', ms=5, al
    plt.plot(ml_pos, ml_pmf, lw=2, c=f'{c2}', label=f'{ml_model}')

    plt.xlabel(r'distance ($\rm{\AA}$)')
    plt.ylabel('Free Energy (kcal/mol')

    plt.legend()
    plt.grid(ls='-.')
    plt.show()

    plt.show()
```

In [94]:
```
plt_fes(dft_pos, dft_pmf, dpmd_pos, dpmd_pmf, c1='black', c2='red', model_1='DFT', ml_mo
```



error between the free energies obtained from DFT and DeePMD simulation.

Root Mean Square Deviation (RMSD) metrics:

$$RMSD(F,G) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[(F(S_i) - \langle F(S)\rangle) - (G(S_i) - \langle G(S)\rangle)]^2}$$

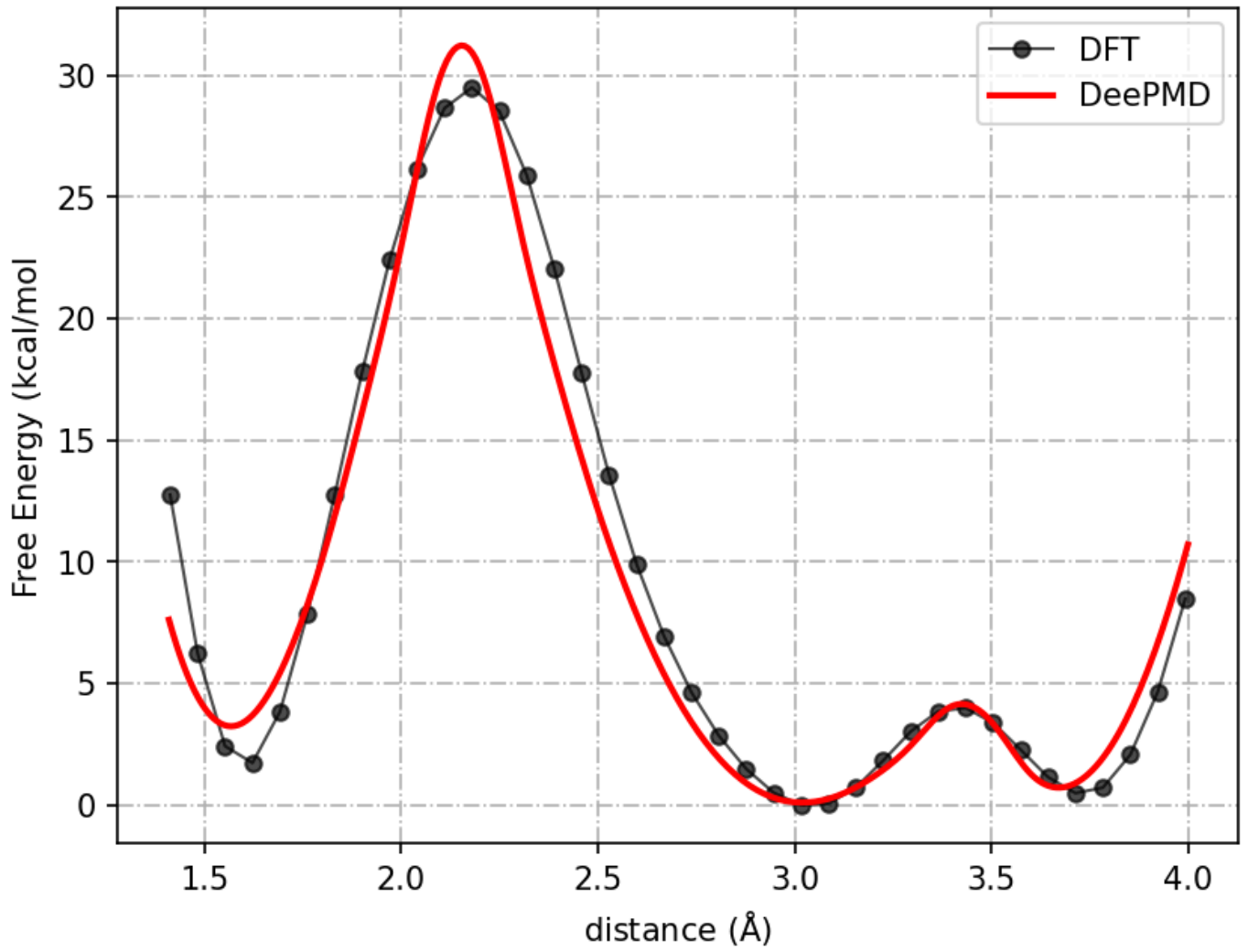
where, S is a collective variable, $\langle F(S))\rangle$ and $\langle G(S))\rangle$ are the DFT and DeePMD free energies averaged over N.

In [87]:
```
# Min and Max position at which PMF is considerd
min_pos, max_pos = 1.4, 4.0
# dpmd_fes_pmf_interp = np.interp(X_pos, dpmd_fes_pos, dpmd_fes_pmf)

# Scipy interpolation
X_pos = np.linspace(min_pos, max_pos, num=500)  # num --> number of interpolation points
dft_fes_pmf_interp = interp1d(dft_pos, dft_pmf, kind='linear', fill_value='extrapolate')
dpmd_fes_pmf_interp = interp1d(dpmd_pos, dpmd_pmf, kind='linear', fill_value='extrapolat

# Calculate Root Mean Square Deviation (RMSD) with respect to the minimum PMF at postion
rmsd = np.sqrt(np.mean((dft_fes_pmf_interp - dpmd_fes_pmf_interp) ** 2))
print(f"Error between free energies: {rmsd:6f} kcal/mol")

# Save interpolated FES and positions to a file using numpy.vstack -> Stack arrays
output_data = np.vstack((X_pos, dft_fes_pmf_interp, dpmd_fes_pmf_interp)).T
```

```
np.savetxt('interpolated_fes.dat', output_data, header='#Position, DFT_FES_PMF, DPMD_FES

# for i in range(len(common_positions)):
#     print(common_positions[i], dft_fes_pmf_interpolated[i], dpmd_fes_pmf_interpolated[

print("Interpolated FES and positions saved to 'interpolated_fes.dat'.")
```

```
Error between free energies: 1.701638 kcal/mol
Interpolated FES and positions saved to 'interpolated_fes.dat'.
```

In [97]:
```
plt_fes(dft_pos, dft_pmf, dpmd_pos, dpmd_pmf, c1='black', c2='red', model_1='DFT', ml_mo
```



In [ ]: