

Winmostar V11
MACE 利用向けマニュアル

令和 7 年 5 月 2 日
株式会社クロスアビリティ

本書では Winmostar V11.12.1 を用いて、リモートサーバ上で MACE 対応 LAMMPS を利用する手順を示します。本書で利用する機能はプロフェッショナル版エリートまたは無料トライアルで利用可能です。**無償版、学生版、プロフェッショナル版エコノミー、プレミアムではご利用いただけません。** Winmostar V11.12.0 以前を利用している場合は V11.12.1 以降にアップデートしてください。本書では MACE 対応 LAMMPS を RockyLinux 9.5 のリモートマシンにインストールします。なお、本書で利用している MACE 関連ソフトウェア・データの提供方法は今後変更される可能性が高いため、本書記載の手順は本書執筆時点での手順であることをご了承ください。また、[使用規約](#)記載の通り、MACE および MACE 対応 LAMMPS を含む Winmostar 本体以外のソフトウェアの動作については当社のサポート対象外です。

参考 URL:

<https://mace-docs.readthedocs.io/en/latest/guide/installation.html>

<https://mace-docs.readthedocs.io/en/latest/guide/lammps.html>

1. リモートサーバの準備

- ① 各種のターミナルソフトウェアでリモートサーバにログインします。
- ② Anaconda あるいは Miniconda 環境がない場合は、Miniconda をインストールします。

```
$ cd ~
```

```
$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86\_64.sh
```

```
$ bash Miniconda3-latest-Linux-x86_64.sh
```

(基本的にデフォルト設定で進め、最後に ~/.bashrc に初期化設定を追加するか聞かれるところだけデフォルト設定(no)ではなく"yes"と回答)

- ③ conda で MACE 専用の仮想環境を作成し、その仮想環境を立ち上げます。

```
$ conda create -n mace
```

```
$ conda activate mace
```

- ④ pip がない場合は pip をインストールします。

```
$ conda install pip
```

- ⑤ MACE をインストールします。

```
$ pip install --upgrade pip
```

```
$ pip install mace-torch
```

- ⑥ libtorch を入手します。

```
$ wget https://download.pytorch.org/libtorch/cpu/libtorch-shared-with-deps-1.13.0%2Bcpu.zip
```

```
$ unzip libtorch-shared-with-deps-1.13.0+cpu.zip
```

```
$ rm libtorch-shared-with-deps-1.13.0+cpu.zip
```

- ⑦ MKL をインストールします。

```
$ conda install mkl mkl-devel
```

⑧ MACE 対応 LAMMPS をインストールします。

```
$ git clone --branch mace --depth=1
```

```
https://github.com/ACEsuit/lammps
```

```
$ cd lammps; mkdir build; cd build
```

```
$ cmake -DCMAKE_INSTALL_PREFIX=$(pwd) ¥
```

```
-D CMAKE_CXX_STANDARD=17 ¥
```

```
-D CMAKE_CXX_STANDARD_REQUIRED=ON ¥
```

```
-D BUILD_MPI=ON ¥
```

```
-D BUILD_OMP=ON ¥
```

```
-D PKG_OPENMP=ON ¥
```

```
-D PKG_MOLECULE=ON ¥
```

```
-D PKG_MISC=ON ¥
```

```
-D PKG_RIGID=ON ¥
```

```
-D PKG_REAXFF=ON ¥
```

```
-D PKG_EXTRA-DUMP=ON ¥
```

```
-D PKG_MC=ON ¥
```

```
-D PKG_KSPACE=ON ¥
```

```
-D PKG_PHONON=ON ¥
```

```
-D PKG_QEQ=ON ¥
```

```
-D PKG_REACTION=ON ¥
```

```
-D PKG_REPLICA=ON ¥
```

```
-D PKG_DPD-BASIC=ON ¥
```

```
-D PKG_EXTRA-MOLECULE=ON ¥
```

```
-D PKG_MANYBODY=ON ¥
```

```
-D PKG_MEAM=ON ¥
```

```
-D PKG_EXTRA-FIX=ON ¥
```

```
-D PKG_FEP=ON ¥
```

```
-D PKG_TALLY=ON ¥
```

```
-D PKG_COLVARS=ON ¥
```

```
-D PKG_ML-MACE=ON ¥
```

```
-D CMAKE_PREFIX_PATH=$(pwd)/../../libtorch ¥
```

```
-DMKL_INCLUDE_DIR=$CONDA_PREFIX/include ¥
```

```
-DMKL_LIBRARIES="$CONDA_PREFIX/lib/libmkl_rt.so" ¥
```

```
../cmake
```

```
$ make -j 4
```

```
$ make install
```

- ⑨ <https://github.com/ACEsuit/mace?tab=readme-ov-file#pretrained-foundation-models> にある MACE の Latest Recommended Foundation Models の表の中から使いたいモデルの URL をコピーし、リモートサーバ上にダウンロードします。本書では例として MACE-MPA-0 を使います。

```
$ cd ~
```

```
$ wget https://github.com/ACEsuit/mace-mp/releases/download/mace\_mpa\_0/mace-mpa-0-medium.model
```

- ⑩ LAMMPS 用モデルファイルを作成します。create_lammps_model.py が見つからない場合は `$ find ~ -name "create_lammps_model.py"` で探します。

```
$ python ${HOME}/miniconda3/envs/mace/lib/python3.13/site-packages/mace/cli/create_lammps_model.py mace-mpa-0-medium.model
```

- ⑪ LAMMPS 用モデルファイル(本書の場合は mace-mpa-0-medium.model-lammps.pt)が生成されるので、これを Winmostar をインストールした(する)Windows PC に scp 等でコピーします。そして、ファイル名に拡張子「.mace」を追加します(本書の場合は mace-mpa-0-medium.model-lammps.pt.mace)。

2. Winmostar の操作方法

- ① Windows PC に Winmostar V11.12.1 以降をインストールします。インストール手順は <https://winmostar.com/jp/installation/> に従います。手順 7 以降の実施は不要です。
- ② Winmostar を起動します。
- ③ ツール | 環境設定をクリックし、計算タブの LAMMPS ポテンシャルフォルダの Open potential directory をクリックし、開いたフォルダに 1. ⑩でコピーしたファイル(本書の場合は mace-mpa-0-medium.model-lammps.pt.mace)を移動します。その後 LAMMPS の pair_style, Potential file の入力を許可にチェックを入れ OK をクリックします。
- ④ ファイル | 新規プロジェクトをクリックし、プロジェクト名を入力し OK をクリックします。
- ⑤ 計算したい初期構造を作成します。原子構造の作成方法の例は LAMMPS 各種チュートリアル <https://winmostar.com/jp/tutorials/index.html#LAMMPS> 詳細はユーザマニュアル https://winmostar.com/jp/manual_jp/html/operation/createsystem.html で確認できます。ひとまずサンプルデータで計算したい場合はファイル | インポート | Samples ファイル | Si.cif をクリックします。
- ⑥ MD | LAMMPS | ワークフロー設定をクリックします。
- ⑦ 「電荷を設定しますか？」と表示されたらいいえをクリックします。

- ⑧ **パラメータファイルを使用**にチェックを入れ **OK** をクリックします。
- ⑨ **無機物系を計算**にチェックを入れ(有機物を計算する場合もこちらを選択)、**Pair style** に「mace_no_domain_decomposition」(「_」はスペース)、**Potential file** に「mace-mpa-0-medium.model-lammps.pt.mace」(1. ⑪でコピーしたファイル)を選択し **OK** をクリックします。「力場が設定されました」と表示されたら **OK** をクリックします。
- ⑩ **LAMMPS Workflow Setup** ウィンドウで適宜計算条件を設定します。計算条件の設定例は **LAMMPS 各種チュートリアル**
<https://winmostar.com/jp/tutorials/index.html#LAMMPS>
 設定の詳細はユーザマニュアル
https://winmostar.com/jp/manual_jp/html/winmos/md/winmos_lammps.html#md-lammps-workflow
 で確認できます。
- ⑪ **LAMMPS Workflow Setup** の **1st job** の **Details** をクリックします。
- ⑫ **Advanced** タブの **Force create atom map** にチェックを入れ、**Enable dynamic load balancing** のチェックを外します。
- ⑬ **LAMMPS Keyword Setup** ウィンドウの **OK** をクリックします。
- ⑭ ⑪～⑬の手順を **LAMMPS Workflow Setup** の **2nd job** 以降の全てのジョブに対して同様に実行します。
- ⑮ **LAMMPS Workflow Setup** ウィンドウでの設定が終わったら **OK** をクリックします。
- ⑯ **ジョブの設定**ウィンドウでリモートジョブの設定を行います。リモートジョブの設定方法は **リモートジョブプロジェクトモード向けチュートリアル**
https://winmostar.com/jp/tutorials/RemoteJob_tutorial_1%28ProjectMode%29.pdf
 設定の詳細はユーザマニュアル
https://winmostar.com/jp/manual_jp/html/operation/calc_import.html#calc-import-proj
 で確認できます。

テンプレートスクリプトの「# Insert commands here」と「# Do not modify the followings」の間には以下のように記載します。

```
# Insert commands here
conda activate mace
MPI_COMMAND="mpirun -n 1 "
BIN_LAMMPS=${HOME}/lammps/build_pkg/lmp
export OMP_NUM_THREADS=%WM_NUM_PROC%
export MKL_NUM_THREADS=%WM_NUM_PROC%
# Do not modify the followings
```

ジョブの設定ウィンドウの**# of MPI Procs** に並列数を入力しますが、このテンプレートスクリ

プトにおいては MPI 並列数ではなくスレッド並列数として解釈されます。

- ⑰ **ジョブの設定**ウィンドウで設定が終わったら**実行**をクリックします。
- ⑱ 通常の LAMMPS を用いた計算と同様に結果解析を進めます。解析手順の例は LAMMPS 各種チュートリアルにて確認できます。

<https://winmostar.com/jp/tutorials/index.html#LAMMPS>

以上