

## LAMMPS 29Aug2024

### ウェブページ

<https://www.lammps.org>

### バージョン

29Aug2024

### ビルド環境

- GCC 13.1.1 (gcc-toolset-13)
- Intel MPI 2021.13
- GSL 2.8

### ビルドに必要なファイル

- lammps-stable\_29Aug2024.tar.gz
- MDI\_Library-1.4.29.tar.gz
- n2p2-2.2.0.tar.gz
- (一部ファイルは以下スクリプト中で取得)

### ビルド手順

#### N2P2 2.2.0 (抜粋)

```
$ module -s purge
$ module -s load gcc-toolset/13
$ module -s load intelmpi/2021.13
$ module -s load gsl/2.8
$ cd /apl/lammps/2024-Aug29
$ tar zxvf n2p2-2.2.0.tar.gz
$ cd n2p2-2.2.0/src
$ make INTERFACES=LAMMPS COMP=gnu PROJECT_CC=g++ PROJECT_MPICC=mpicxx PROJECT_CFLAGS="-O3 -march=native -std=c++11 -fPIC"
APP_CORE=nnp-convert APP_TRAIN=nnp-train APP=nnp-convert -j8
```

- LAMMPS の方で自動ビルドがうまくいかなかったため手動で実行
- -fPIC オプションはおそらく必須
- include と lib だけをコピーした場合はうまくいかなかったため、lammps のインストールディレクトリで展開

#### MDI 1.4.29 (抜粋)

```
$ tar xf MDI_Library-1.4.29.tar.gz
$ cd MDI_Library-1.4.29
$ mkdir build && cd build
$ cmake .. -DCMAKE_INSTALL_PREFIX=/apl/lammps/2024-Aug29/mdi-1.4.29 -DPython_EXECUTABLE=/usr/bin/python3.6m -
DPython_INCLUDE_DIR=/usr/include/python3.6m
$ make -j8
$ make install
```

- LAMMPS の自動ビルドでは Python の指定がうまく伝播しなかったため別途対応

#### LAMMPS

```
#!/bin/sh

VERSION=2024-Aug29
NAME=lammps-stable_29Aug2024
INSTALL_PREFIX=/apl/lammps/${VERSION}

BASEDIR=/home/users/${USER}/Software/LAMMPS/${VERSION}
LAMMPS_TARBALL=${BASEDIR}/${NAME}.tar.gz

WORKDIR=/gwork/users/${USER}
```

```

LAMMPS_WORKDIR=${WORKDIR}/${NAME}

FFMPEG_BIN=/apl/ffmpeg/6.1/bin/ffmpeg
VMD_MOLFILE_INC=/home/users/${USER}/Software/VMD/1.9.4/vmd-1.9.4a57/plugins/include
GSL_ROOT=/apl/gsl/2.8
MDI_ROOT=/apl/lammps/2024-Aug29/mdi-1.4.29
N2P2_ROOT=/apl/lammps/2024-Aug29/n2p2-2.2.0

PARALLEL=12

#-----
umask 0022
export LANG=C
ulimit -s unlimited

module -s purge
module -s load gcc-toolset/13
module -s load intelmpi/2021.13
module -s load gsl/2.8

PYTHONEXE=/usr/bin/python3.6m
PYTHONINC=/usr/include/python3.6m

export CPATH="${MDI_ROOT}/include/mdi:${CPATH}"
export LIBRARY_PATH="${MDI_ROOT}/lib64/mdi:${LIBRARY_PATH}"
export LD_LIBRARY_PATH="${MDI_ROOT}/lib64/mdi:${LD_LIBRARY_PATH}"

cd ${WORKDIR}
if [ -d ${NAME} ]; then
  mv ${NAME} lammps_erase
  rm -rf lammps_erase &
fi

tar xzf ${LAMMPS_TARBALL}

cd ${NAME}
mkdir build && cd build

# Disabled PKGs:
# ADIOS, VTK: noavail
# GUI: to avoid complicated dependencies
# KIM: CDDL is incompatible with GPL
# INTEL: not necessary for gcc build
# ML-IAP: compilation error

cmake ../cmake \
-DLAMMPS_MACHINE=rccs \
-DENABLE_TESTING=on \
-DCMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
-DCMAKE_C_COMPILER=gcc \
-DCMAKE_CXX_COMPILER=g++ \
-DCMAKE_Fortran_COMPILER=gfortran \
-DCMAKE_MPI_C_COMPILER=mpicc \
-DCMAKE_MPI_CXX_COMPILER=mpicxx \
-DCMAKE_MPI_Fortran_COMPILER=mpif90 \
-DCMAKE_C_FLAGS_RELEASE="-O3 -DNDEBUG" \
-DCMAKE_CXX_FLAGS_RELEASE="-O3 -DNDEBUG" \
-DCMAKE_Fortran_FLAGS_RELEASE="-O3 -DNDEBUG" \
-DPython_EXECUTABLE=${PYTHONEXE} \
-DPython_INCLUDE_DIR=${PYTHONINC} \
-DGSL_ROOT_DIR=${GSL_ROOT} \
-DBUILD_SHARED_LIBS=on \
-DBUILD_TOOLS=on \
-DBUILD_MPI=on \
-DBUILD_OMP=on \
-DBUILD_LAMMPS_GUI=off \

```

-DFFT=FFTW3 \  
-DFFT\_SINGLE=on \  
-DFFT\_FFTW\_THREADS=on \  
-DWITH\_JPEG=on \  
-DWITH\_PNG=on \  
-DWITH\_GZIP=on \  
-DWITH\_FFmpeg=on \  
-DFFMPEG\_EXECUTABLE=\${FFMPEG\_BIN} \  
-DPKG\_ADIOS=off \  
-DPKG\_AMOEBA=on \  
-DPKG\_ASPHERE=on \  
-DPKG\_ATC=on \  
-DPKG\_AWPMO=on \  
-DPKG\_BOCS=on \  
-DPKG\_BODY=on \  
-DPKG\_BPM=on \  
-DPKG\_BROWNIAN=on \  
-DPKG\_CG-DNA=on \  
-DPKG\_CG-SPICA=on \  
-DPKG\_CLASS2=on \  
-DPKG\_COLLOID=on \  
-DPKG\_COLVARS=on \  
-DPKG\_COMPRESS=on \  
-DPKG\_CORESHELL=on \  
-DPKG\_DIELECTRIC=on \  
-DPKG\_DIFFRACTION=on \  
-DPKG\_DIPOLE=on \  
-DPKG\_DPD-BASIC=on \  
-DPKG\_DPD-MESO=on \  
-DPKG\_DPD-REACT=on \  
-DPKG\_DPD-SMOOTH=on \  
-DPKG\_DRUDE=on \  
-DPKG\_EFF=on \  
-DPKG\_ELECTRODE=on \  
-DPKG\_EXTRA-COMMAND=on \  
-DPKG\_EXTRA-COMPUTE=on \  
-DPKG\_EXTRA-DUMP=on \  
-DPKG\_EXTRA-FIX=on \  
-DPKG\_EXTRA-MOLECULE=on \  
-DPKG\_EXTRA-PAIR=on \  
-DPKG\_FEP=on \  
-DPKG\_GPU=off \  
-DPKG\_GRANULAR=on \  
-DPKG\_H5MD=on \  
-DPKG\_INTEL=off \  
-DPKG\_INTERLAYER=on \  
-DPKG\_KIM=off \  
-DDOWNLOAD\_KIM=off \  
-DPKG\_KOKKOS=on \  
-DKokkos\_ARCH\_ZEN3=on \  
-DKokkos\_ENABLE\_OPENMP=on \  
-DPKG\_KSPACE=on \  
-DPKG\_LATBOLTZ=on \  
-DPKG\_LEPTON=on \  
-DPKG\_MACHDYN=on \  
-DDOWNLOAD\_EIGEN3=on \  
-DPKG\_MANIFOLD=on \  
-DPKG\_MANYBODY=on \  
-DPKG\_MC=on \  
-DPKG\_MDI=on \  
-DDOWNLOAD\_MDI=off \  
-DPKG\_MEAM=on \  
-DPKG\_MESONT=on \  
-DPKG\_MGPT=on \  
-DPKG\_MISC=on \

```
-DPKG_ML-HDNNP=on \  
-DDOWNLOAD_N2P2=off \  
-DN2P2_DIR=${N2P2_ROOT} \  
-DPKG_ML-IAP=off \  
-DMLIAP_ENABLE_PYTHON=off \  
-DPKG_ML-PACE=on \  
-DPKG_ML-POD=on \  
-DPKG_ML-QUIP=on \  
-DDOWNLOAD_QUIP=on \  
-DPKG_ML-RANN=on \  
-DPKG_ML-SNAP=on \  
-DPKG_ML-UF3=on \  
-DPKG_MOFFF=on \  
-DPKG_MOLECULE=on \  
-DPKG_MOLFILE=on \  
-DMOLFILE_INCLUDE_DIR=${VMD_MOLFILE_INC} \  
-DPKG_NETCDF=on \  
-DPKG_OPENMP=on \  
-DPKG_OPT=on \  
-DPKG_ORIENT=on \  
-DPKG_PERI=on \  
-DPKG_PHONON=on \  
-DPKG_PLUGIN=on \  
-DPKG_PLUMED=on \  
-DDOWNLOAD_PLUMED=on \  
-DPKG_POEMS=on \  
-DPKG_PTM=on \  
-DPKG_PYTHON=on \  
-DPKG_QEQ=on \  
-DPKG_QMMM=on \  
-DPKG_QTB=on \  
-DPKG_REACTION=on \  
-DPKG_REAXFF=on \  
-DPKG_REPLICA=on \  
-DPKG_RHEO=on \  
-DPKG_RIGID=on \  
-DPKG_SCAFACOS=on \  
-DDOWNLOAD_SCAFACOS=on \  
-DPKG_SHOCK=on \  
-DPKG_SMTBQ=on \  
-DPKG_SPH=on \  
-DPKG_SPIN=on \  
-DPKG_SRD=on \  
-DPKG_TALLY=on \  
-DPKG_UEF=on \  
-DPKG_VORONOI=on \  
-DDOWNLOAD_VORO=on \  
-DPKG_VTK=off \  
-DPKG_YAFF=on \  
-DBLA_VENDOR=OpenBLAS \  
-DCMAKE_BUILD_TYPE=Release
```

```
make VERBOSE=1 -j ${PARALLEL}
```

```
export OMP_NUM_THREADS=2
```

```
make test
```

```
make install
```

```
cp -a ../examples ${INSTALL_PREFIX}
```

```
cd ${INSTALL_PREFIX}
```

```
for f in etc/profile.d/*; do
```

```
if [ -f $f ]; then
```

```
ln -s $f .
```

```
fi
```

```
done
```

```
cd lib64
```

```
if [ -f liblammps_rccs.so ]; then
```

```
ln -s liblammps_rccs.so liblammps.so
```

```
fi
```

```
if [ -f liblammps_rccs.so.0 ]; then
```

```
ln -s liblammps_rccs.so.0 liblammps.so.0
```

```
fi
```

## 有効なパッケージ

```
AMOeba ASPHERE ATC AWPMD BOCS BODY BPM BROWNIAN CG-DNA CG-SPICA CLASS2 COLLOID  
COLVARS COMPRESS CORESHELL DIELECTRIC DIFFRACTION DIPOLE DPD-BASIC DPD-MESO  
DPD-REACT DPD-SMOOTH DRUDE EFF ELECTRODE EXTRA-COMMAND EXTRA-COMPUTE EXTRA-DUMP  
EXTRA-FIX EXTRA-MOLECULE EXTRA-PAIR FEP GRANULAR H5MD INTERLAYER KOKKOS KSPACE  
LATBOLTZ LEPTON MACHDYN MANIFOLD MANYBODY MC MDI MEAM MESONT MGPT MISC ML-HDNNP  
ML-PACE ML-POD ML-QUIP ML-RANN ML-SNAP ML-UF3 MOFFF MOLECULE MOLFILE NETCDF  
OPENMP OPT ORIENT PERI PHONON PLUGIN PLUMED POEMS PTM PYTHON QEQ QMMM QTB  
REACTION REAXFF REPLICa RHEO RIGID SCAFACOS SHOCK SMTBQ SPH SPIN SRD TALLY UEF  
VORONOI YAFF
```

## テスト

テストログのコピーが </apl/lammps/2024-Aug29/Testing> 以下にあります。

The following tests FAILED:

```
37 - SimpleCommands (SEGFAULT)  
85 - PythonPyLammps (Failed)  
152 - MolPairStyle:lepton (Failed)  
253 - AtomicPairStyle:buck_coul_cut_qeq_point (Failed)  
254 - AtomicPairStyle:buck_coul_cut_qeq_shielded (Failed)  
271 - AtomicPairStyle:edip (Failed)  
277 - AtomicPairStyle:lepton_sphere (Failed)  
278 - AtomicPairStyle:lj_cut_sphere (Failed)  
279 - AtomicPairStyle:lj_expand_sphere (Failed)  
283 - AtomicPairStyle:meam_ms (Failed)  
285 - AtomicPairStyle:meam_spline (Failed)  
286 - AtomicPairStyle:meam_sw_spline (Failed)  
288 - AtomicPairStyle:pedone (Failed)  
290 - AtomicPairStyle:reaxff-acks2 (Failed)  
291 - AtomicPairStyle:reaxff-acks2_efield (Failed)  
292 - AtomicPairStyle:reaxff (Failed)  
293 - AtomicPairStyle:reaxff_lgvdw (Failed)  
294 - AtomicPairStyle:reaxff_noeq (Failed)  
295 - AtomicPairStyle:reaxff_tabulate (Failed)  
296 - AtomicPairStyle:reaxff_tabulate_flag (Failed)  
321 - ManybodyPairStyle:ilp-graphene-hbn (Failed)  
322 - ManybodyPairStyle:ilp-graphene-hbn_notaper (Failed)  
372 - BondStyle:harmonic_restrain (Failed)  
422 - KSpaceStyle:pppm_ad (Failed)  
423 - KSpaceStyle:pppm_cg (Failed)  
424 - KSpaceStyle:pppm_cg_ad (Failed)  
425 - KSpaceStyle:pppm_cg_tiled (Failed)  
442 - KSpaceStyle:pppm_tip4p (Failed)  
447 - KSpaceStyle:scafacos_direct (Failed)  
448 - KSpaceStyle:scafacos_ewald (Failed)  
449 - KSpaceStyle:scafacos_fmm (Failed)  
450 - KSpaceStyle:scafacos_fmm_tuned (Failed)  
451 - KSpaceStyle:scafacos_p2nfft (Failed)  
457 - FixTimestep:addtorque_const (Failed)  
532 - FixTimestep:spring_rg (Failed)  
539 - FixTimestep:wall_harmonic_const (Failed)  
540 - FixTimestep:wall_lepton_const (Failed)  
544 - FixTimestep:wall_morse_const (Failed)
```

```
546 - FixTimestep:wall_table_linear (Failed)
547 - FixTimestep:wall_table_spline (Failed)
552 - DihedralStyle:cosine_squared_restricted (Failed)
561 - DihedralStyle:quadratic (Failed)
563 - DihedralStyle:table_cut_linear (Failed)
565 - DihedralStyle:table_linear (Failed)
566 - DihedralStyle:table_spline (Failed)
```

以下の lattice の入っているインプットについては大きなエラーがでてるように見える。

- AtomicPairStyle:
  - buck\_coul\_cut\_qeq\_point, buck\_coul\_cut\_qeq\_shielded, edip, lepton\_sphere, lj\_cut\_sphere, lj\_expand\_sphere, meam\_sw\_spline, pedone, reaxff-acks2, reaxff-acks2\_efield, reaxff, reaxff\_lgvdw, reaxff\_noqeq, reaxff\_tabulate, reaxff\_tabulate\_flag
- KSpaceStyle:
  - scafacos\_direct, scafacos\_ewald, scafacos\_fmm, scafacos\_fmm\_tuned, scafacos\_p2nfft

以下は軽微な数値誤差

- MolPairStyle:lepton, AtomicPairStyle:meam\_ms, AtomicPairStyle:meam\_spline, ManybodyPairStyle:ilp-graphene-hbn\*, BondStyle:harmonic\_restrain, KSpaceStyle:pppm\_\*, FixTimestep:\*, DihedralStyle:\*

そのほか

- SimpleCommands: Quit で segfault (timeout になる場合もあり)
- PythonPyLammps: test\_info\_queries で FAIL
  - (export OMP\_NUM\_THREADS=2 の宣言が原因だと思われる...)

## メモ

- ADIOS, VTK, KIM は未検証
- INTEL はこの設定では不要であるため外す。
  - [インテルコンパイラでビルドした INTEL パッケージ有効版は別途用意。](#)
    - INTEL パッケージ有効な場合は 2 割程度の速度向上を確認(rhodo 系)
    - INTEL パッケージを使わない状態ではわずかに GCC 版の方が速度が出ているように見える
- GUI は依存関係に必要なパッケージが多すぎるため断念
- ML-IAP はコンパイルエラーが発生したため回避
- GCC12 で試した場合もテスト結果や速度に特に違いは見られない。
- N2P2 は LAMMPS の cmake で自動ビルドを指定するとエラーが出るため手動で。
- MDI は LAMMPS の cmake の場合は Python\_EXECUTABLE などが渡らないために python2 が使われてしまう。そのため手動で回避。