

LAMMPS 22Jul2025 - Intel

ウェブページ

<https://www.lammps.org>

バージョン

22Jul2025

ビルド環境

- GCC 10.3.1 (gcc-toolset-10)
- oneAPI Compiler 2025.2.0
- Intel MPI 2021.16
- GSL 2.8
- MKL 2025.2.0

ビルドに必要なファイル

- lammps-stable_22Jul2025.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

```
#!/bin/sh

VERSION=2025-Jul22
NAME=lammps-stable_22Jul2025
INSTALL_PREFIX=/apl/lammps/${VERSION}-intel

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

PARALLEL=12

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

module -s purge

. ~/intel/oneapi/compiler/latest/env/vars.sh # 2025.2.0

module -s load gcc-toolset/10
module -s load compiler-rt/2025.2.0
module -s load intelmpi/2021.16
module -s load gsl/2.8
module -s load mkl/2025.2.0

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

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

fi

```
tar xzf ${LAMMPS_TARBALL}
```

```
cd ${NAME}
```

```
sed -i -e "s/xHost/march=core-avx2/" \
```

```
-e "s/-fp-model precise//" cmake/CMakeLists.txt
```

```
sed -i -e '26i#include "force.h"' unittest/force-styles/test_fix_timestep.cpp
```

```
sed -i -e '29i#include "kspace.h"' unittest/force-styles/test_pair_style.cpp
```

```
sed -i -e "59i \ -DPython_EXECUTABLE=${PYTHONEXE}" cmake/Modules/Packages/MDI.cmake
```

```
mkdir build && cd build
```

Disabled PKGs:

ADIOS, VTK: noavail

QUIP: The IntelLLVM Fortran compiler is not (yet) supported for building QUIP

KIM: to remove dependence on libkim-api.so.2

KOKKOS: to avoid performance degradation of intel package

SCAFACOS: compilation problem regarding gfortran?

```
cmake ../cmake \
```

```
-DLAMMPS_MACHINE=rccs \
```

```
-DENABLE_TESTING=on \
```

```
-DCMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
```

```
-DCMAKE_C_COMPILER=mpiicx \
```

```
-DCMAKE_CXX_COMPILER=mpiicpx \
```

```
-DCMAKE_Fortran_COMPILER=mpiifx \
```

```
-DCMAKE_C_FLAGS_RELEASE="-O3 -DNDEBUG -march=core-avx2" \
```

```
-DCMAKE_CXX_FLAGS_RELEASE="-O3 -DNDEBUG -march=core-avx2" \
```

```
-DCMAKE_Fortran_FLAGS_RELEASE="-O3 -DNDEBUG -march=core-avx2" \
```

```
-DPython_EXECUTABLE=${PYTHONEXE} \
```

```
-DPython_INCLUDE_DIR=${PYTHONINC} \
```

```
-D GSL_ROOT_DIR=${GSL_ROOT} \
```

```
-DBUILD_SHARED_LIBS=on \
```

```
-DBUILD_TOOLS=on \
```

```
-DBUILD_MPI=on \
```

```
-DBUILD_OMP=on \
```

```
-DBUILD_LAMMPS_GUI=on \
```

```
-DBUILD_WHAM=on \
```

```
-DFFT=MKL \
```

```
-DFFT_SINGLE=on \
```

```
-DFFT_MKL_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_APIP=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=on \
-DPKG_INTERLAYER=on \
-DPKG_KIM=off \
-DDOWNLOAD_KIM=off \
-DPKG_KOKKOS=off \
-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=on \
-DPKG_MEAM=on \
-DPKG_MESONT=on \
-DPKG_MGPT=on \
-DPKG_MISC=on \
-DPKG_ML-HDNNP=off \
-DDOWNLOAD_N2P2=off \
-DPKG_ML-IAP=off \
-DPKG_ML-PACE=on \
-DPKG_ML-POD=on \
-DPKG_ML-QUIP=off \
-DDOWNLOAD_QUIP=off \
-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=off \
-DDOWNLOAD_SCAFACOS=off \
-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 \
-DBLAS_LIBRARIES="-qmkl" \
-DCMAKE_BUILD_TYPE=Release

#make -j ${PARALLEL}
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 API 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 INTEL
 INTERLAYER KSPACE LATBOLTZ LEPTON MACHDYN MANIFOLD MANYBODY MC MDI MEAM MESONT
 MGPT MISC ML-PACE ML-POD 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 SHOCK SMTBQ SPH SPIN SRD TALLY UEF VORONOI
 YAFF

テスト結果

以下のテストに失敗しています。ログのコピーは [/apl/lammps/2025-Jul22-intel/Testing](#) 以下にあります。
[GCC 版の情報](#) も参考にしてください。

- 14:AtomStyles => body_nparticle 周りで挙動の違い？
- 38:SimpleCommands => Quit テストが失敗
- 40:Groups => VariableFunctions で数値エラー

- 54:LibraryProperties => GCC 版と同様
- 72:FortranProperties => 54:LibraryProperties と同じエラー。GCC 版と同様

以下のテストでは free(): invalid pointer が発生してエラー。それ以上の原因わからず。

- 55:LibraryObjects
- 84:PythonCommands
- 270:AtomicPairStyle:eam_alloy
- 271:AtomicPairStyle:eam_alloy_real
- 275:AtomicPairStyle:eam_fs
- 276:AtomicPairStyle:eam_fs_real

以下のインプットに lattice が含まれるテストは大きなエラー。

- 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-qtpie, reaxff, reaxff_lgvdw, reaxff_noqeq, reaxff_tabulate, reaxff_tabulate_flag

以下は大きな数値エラー。何か問題があると思われる。

- 337:ManybodyPairStyle:lcboop
- 383:BondStyle:harmonic_restrain

以下は軽微な数値エラー

- 130:MolPairStyle:coul_shield
- 219:MolPairStyle:lj_pirani
- 244:MolPairStyle:python_hybrid
- 291:AtomicPairStyle:meam_2nn
- 294:AtomicPairStyle:meam_spline
- 332:ManybodyPairStyle:ilp-graphene-hbn
- 333:ManybodyPairStyle:ilp-graphene-hbn_notaper
- 336:ManybodyPairStyle:kolmogorov_crespi_z
- 350:ManybodyPairStyle:polymorphic_sw
- 353:ManybodyPairStyle:rann
- 380:BondStyle:gaussian
- 384:BondStyle:harmonic_shift
- 397:AngleStyle:charmm
- 434:KSpaceStyle:pppm_ad
- 435:KSpaceStyle:pppm_cg
- 436:KSpaceStyle:pppm_cg_ad
- 437:KSpaceStyle:pppm_cg_tiled
- 454:KSpaceStyle:pppm_tip4p
- 455:KSpaceStyle:pppm_tip4p_ad
- 476:FixTimestep:deform_tri
- 502:FixTimestep:npt_iso
- 513:FixTimestep:nvt-psllod
- 514:FixTimestep:nvt-sllod
- 526:FixTimestep:recenter-init
- 535:FixTimestep:rigid_npt_small
- 565:FixTimestep:wall_lj1043_const
- 569:FixTimestep:wall_reflect
- 570:FixTimestep:wall_region_harmonic
- 571:FixTimestep:wall_region_lj1043
- 572:FixTimestep:wall_region_lj126
- 573:FixTimestep:wall_region_lj93
- 574:FixTimestep:wall_region_morse
- 575:FixTimestep:wall_region_sphere
- 591:DihedralStyle:quadratic
- 593:DihedralStyle:table_cut_linear

メモ

- [GCC 版の情報もご確認ください。](#)
- INTEL パッケージを使わないのであれば上記 GCC 版の方がテストの失敗も少なく、安定して動作すると思われます。

- INTEL パッケージ使用時の速度低下を防ぐため、インテル llvm コンパイラ時に追加される -fp-model precise を無効にしています。
 - -fp-model precise は clang のデフォルト設定ですが、intel llvm コンパイラではデフォルト設定ではありません。
 - -fp-model precise を外したことで従来の icc, icpc 系と同じデフォルト設定(fast)にしています。
 - -fp-model precise を指定した場合、INTEL パッケージによる SIMD 化による高速化の効果が小さくなります。
 - 同様に Kokkos の設定で Kokkos_ARCH_ZEN3 を有効にすると INTEL パッケージに影響があるため、Kokkos も無効化しています。
 - intel classic コンパイラ向けの設定では -fp-model fast=2 が使われていました。