

## LAMMPS 16Mar18 (stable release) for LX (gcc)

### ウェブページ

<http://lammps.sandia.gov/>

### バージョン

16Mar18

### ビルド環境

- GCC 6.3.1
- Intel MKL 2018 Update 2
- Intel MPI 2017.3.196
- libjpeg-turbo 1.2.90

### ビルドに必要なファイル

- lammps-stable.tar.gz (16Mar18)
- (一部ファイルは以下スクリプト中で取得)

### ビルド手順

(2018/7/4 修正)

```
#!/bin/sh

VERSION=16Mar18
INSTALL_PREFIX=/local/apl/lx/lammps${VERSION}-gnu

BASEDIR=/home/users/${USER}
LAMMPS_TARBALL=${BASEDIR}/lammps-stable.tar.gz

WORKDIR=/work/users/${USER}
WORKDIR_LAMMPS=${WORKDIR}/lammps-${VERSION}
WORKDIR_LAMMPS_ESC=`echo ${WORKDIR_LAMMPS} | sed -e 's/\//\\//g'`

PARALLEL=12

module purge
module load mkl/2018.0.2
module load mpi/intelmpi/2018.2.199
module load scl/devtoolset-6

#-- libs

MKLROOT_ESC=`echo /local/apl/lx/intel2018update2/compilers_and_libraries_2018.2.199/linux/mkl | sed -e 's/\//\\//g'`
LATTE_GIT=https://github.com/lanl/LATTE # latte
LATTEDIR=${WORKDIR_LAMMPS}/lib/latte/LATTE
LATTEDIR_ESC="${WORKDIR_LAMMPS_ESC}/lib/latte/LATTE"
METIS_VER=5.1.0
METIS=http://glaros.dtc.umn.edu/gkhome/fetch/sw/metis/metis-${METIS_VER}.tar.gz
BML=https://github.com/lanl/bml.git
PROGRESS=https://github.com/lanl/qmd-progress.git
VMD_PLUGIN_INC=`echo /local/apl/lx/vmd193/lib/plugins/include | sed -e 's/\//\\//g'` # molfile
VORO_VER=0.4.6 # voronoi
VORO=http://math.lbl.gov/voro++/download/dir/voro++-${VORO_VER}.tar.gz

#-----
umask 0022

cd ${WORKDIR}
if [ -d lammps-${VERSION} ]; then
  mv lammps-${VERSION} lammps_erase
```

```

rm -rf lammmps_erase &
fi

tar xzf ${LAMMPS_TARBALL}
cd lammmps-${VERSION}

# setup makefiles, libraries, and external resources
## main
sed -e "/g++_openmpi =/s/.*/# rccs = Intel MPI, MKL FFT + g++ /" \
-e "/OMPI_CXX/d" \
-e "s/^CCFLAGS.*/CCFLAGS = -g -O3 -fopenmp/" \
-e "s/^LINKFLAGS.*/LINKFLAGS = -g -O -fopenmp/" \
-e "s/DLAMMPS_GZIP/DLAMMPS_GZIP -DLAMMPS_JPEG/" \
-e "s/^FFT_INC.*/FFT_INC = -DFFT_FFTW3 -L${MKLROOT_ESC}V..V/compiler/lib/intel64 -m64 -I${MKLROOT_RSC}V/include/" \
-e "s/^FFT_PATH.*/FFT_PATH = -L${MKLROOT_ESC}V/lib/intel64 -Wl,--no-as-needed/" \
-e "s/^FFT_LIB.*/FFT_LIB = -lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -liomp5 -lpthread -lm -ldl/" \
-e "s/^JPG_LIB.*/JPG_LIB = -ljpeg/" \
src/MAKE/OPTIONS/Makefile.g++_openmpi > src/MAKE/MINE/Makefile.rccs
## atc
( cd lib/atc && \
sed -e s/linalg/linalg.empty/ Makefile.mpic++ > Makefile.rccs && \
make -f Makefile.rccs -j ${PARALLEL} && \
cd ../.. )
## awpmd
( cd lib/awpmd && \
sed -e s/linalg/empty/ Makefile.mpi > Makefile.rccs && \
make -f Makefile.rccs -j ${PARALLEL} && \
cd ../.. )
## colvars
( cd lib/colvars && \
make -f Makefile.mpi -j ${PARALLEL} && \
cd ../.. )
## h5md
( cd lib/h5md && \
make -f Makefile.mpi -j ${PARALLEL} && \
cd ../.. )
## latte (make jobserver not available?)
## METIS, BML, and (qmd-)PROGRESS are also built
( cd lib/latte && \
git clone ${LATTE_GIT} && \
cd LATTE && \
wget ${METIS} && \
tar xzf metis-${METIS_VER}.tar.gz && \
cd metis-${METIS_VER} && \
make config prefix=../install && \
make install && \
cd ../ && \
git clone ${BML} && \
cd bml && \
EXTRA_CFLAGS="-fPIC" EXTRA_FFLAGS="-fPIC" sh example_build.sh && \
cd build/ && \
make && \
make install && \
cd ../.. && \
git clone ${PROGRESS} && \
cd qmd-progress && \
EXTRA_FCFLAGS="-fPIC" EXTRA_LINK_FLAGS="-fPIC" CMAKE_PREFIX_PATH=${LATTEDIR}/metis-5.1.0/install PROGRESS_GRAPHLIB=yes
PKG_CONFIG_PATH=${LATTEDIR}/bml/install/lib64/pkgconfig ./build.sh configure && \
cd build/ && \
make && \
cd src/ && \
gcc -DGLOBAL_DEBUG=PROGRESS_LOG_DEBUG \
-DMETIS_INDEX_KIND=4 \
-DMETIS_REAL_KIND=0d0 \
-O3 -DNDEBUG -fPIC \

```

```

-I${LATTEDIR}/bml/install/include \
-fopenmp \
-o CMakeFiles/progress.dir/prg_memory_consumption.c.o \
-c ${LATTEDIR}/qmd-progress/src/prg_memory_consumption.c && \
cd ../ && \
make install && \
cd ../.. && \
sed -i -e "s/fopenmp/fopenmp -fPIC/" \
-e "s/^LIB =.*/LIB = -L${MKLRROOT_ESC}\libVintel64 -Wl,--no-as-needed -lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -liomp5 -lpthread -lm -ldl/" \
-e "s/^METIS =.*/METIS = ON/" \
-e "s/^PROGRESS =.*/PROGRESS = ON/" \
-e "s/.(HOME)/${LATTEDIR_ESC}/" \
-e "s/metisV//" \
-e "s/installVlib/installVlib64/" \
makefile.CHOICES && \
make && \
cd ../ && \
ln -s LATTE/src includelink && \
ln -s LATTE/liblink && \
ln -s LATTE/src/latte_c_bind.o filelink.o && \
echo "latte_SYSINC = -I../lib/latte/LATTE/src -I../lib/latte/LATTE/bml/install/include -I../lib/latte/LATTE/qmd-progress/install/include" > Makefile.lammps && \
echo "latte_SYSLIB = -fopenmp ../lib/latte/filelink.o -llatte -L../lib/latte/LATTE/qmd-progress/install/lib64 -lprogress -L../lib/latte/LATTE/bml/install/lib64 -lbml_fortran -lbml -L../lib/latte/LATTE/metis-5.1.0/install/lib -lmetis" >> Makefile.lammps && \
echo -n "latte_SYSPTH = -fopenmp" >> Makefile.lammps && \
cd ../.. )
## meam
( cd lib/meam && \
sed -e s/mpifort/mpif90/ Makefile.mpi > Makefile.rccs && \
make -f Makefile.rccs -j ${PARALLEL} && \
cd ../.. )
## molfile
##### plugin location: /local/apl/lx/vmd193/lib/plugins/LINUXAMD64/molfile
( cd lib/molfile && \
sed -i -e "s/molfile_SYSINC.*/molfile_SYSINC =-I$VMD_PLUGIN_INC/" Makefile.lammps && \
cd ../.. )
## poems
( cd lib/poems && \
make -f Makefile.g++ -j ${PARALLEL} && \
cd ../.. )
## reax
( cd lib/reax && \
make -f Makefile.gfortran -j ${PARALLEL} && \
cd ../.. )
## voronoi
( cd lib/voronoi && \
wget ${VORO} && \
tar xzf voro+-${VORO_VER}.tar.gz && \
cd voro+-${VORO_VER} && \
sed -i -e "s/^CFLAGS=.*CFLAGS=-Wall -ansi -pedantic -fPIC/" config.mk && \
make -j ${PARALLEL} && \
cd ../ && \
ln -s voro+-${VORO_VER}/src includelink && \
ln -s voro+-${VORO_VER}/src liblink && \
cd ../.. )
#----
# now make lammps
cd src
make yes-all no-ext
make no-KOKKOS \
no-GPU \
no-LATTE \

```

```

yes-VORONOI \
yes-USER-H5MD \
yes-USER-MOLFILE \
yes-USER-NETCDF
make -j ${PARALLEL} rccs
make -j ${PARALLEL} rccs mode=shlib
cd ../

# mkdir and install files
mkdir -p ${INSTALL_PREFIX}/src
cp src/Imp_rccs src/liblammps_rccs.so src/*.h ${INSTALL_PREFIX}/src
ln -s ${INSTALL_PREFIX}/src/liblammps_rccs.so ${INSTALL_PREFIX}/src/liblammps.so
cp -r LICENSE \
  README \
  bench/ \
  doc/ \
  examples/ \
  potentials/ \
  python/ \
  tools/ \
  ${INSTALL_PREFIX}
cp lib/h5md/LICENSE ${INSTALL_PREFIX}/LICENSE-h5md
cp lib/voronoi/voro++-${VORO_VER}/LICENSE ${INSTALL_PREFIX}/LICENSE-voro++

## latte extra
mkdir -p ${INSTALL_PREFIX}/lib/latte/LATTE
cp -r lib/latte/LATTE/{LICENSE_GPL-2.0,latte.in,MATRIX,Manual,Restarts,TBparam,animate,examples,tests,tools} ${INSTALL_PREFIX}/lib/latte/LATTE

mkdir -p ${INSTALL_PREFIX}/lib/latte/LATTE/bml
cp -r lib/latte/LATTE/bml/{LICENSE.md,install,examples} ${INSTALL_PREFIX}/lib/latte/LATTE/bml

mkdir -p ${INSTALL_PREFIX}/lib/latte/LATTE/metis-${METIS_VER}
cp -r lib/latte/LATTE/metis-${METIS_VER}/{LICENSE.txt,install} ${INSTALL_PREFIX}/lib/latte/LATTE/metis-${METIS_VER}

mkdir -p ${INSTALL_PREFIX}/lib/latte/LATTE/qmd-progress
cp -r lib/latte/LATTE/qmd-progress/{LICENSE.md,install,examples} ${INSTALL_PREFIX}/lib/latte/LATTE/qmd-progress

```

## パッケージリスト

ASPHERE, BODY, CLASS2, COLLOID, COMPRESS, CORESHELL, DIPOLE, GRANULAR  
 KSPACE, LATTE, MANYBODY, MC, MEAM, MISC, MOLECULE, MPIIO, OPT, PERI, POEMS  
 PYTHON, QEQ, REAX, REPLICA, RIGID, SHOCK, SNAP, SRD, VORONOI

USER-ATC, USER-AWPMD, USER-CGDNA, USER-CGSDK, USER-COLVARS,  
 USER-DIFFRACTION, USER-DPD, USER-DRUDE, USER-EFF, USER-FEP,  
 USER-H5MD, USER-INTEL, USER-LB, USER-MANIFOLD, USER-MEAMC,  
 USER-MESO, USER-MGPT, USER-MISC, USER-MOLFILE, USER-NETCDF,  
 USER-OMP, USER-PHONON, USER-QTB, USER-REAXC, USER-SMTBQ,  
 USER-SPH, USER-TALLY, USER-UEF

## テスト

- シリアルテスト(via run\_tests.py) については通過(legacyテストは除く)
- 並列テストについては、[intel版](#)と同じ方法で実行

## 注意

- ファイルは /local/apl/lx/lammps16Mar18-gnu/ 以下にあります
- 実行バイナリ(imp\_rccs)とライブラリは src/ 以下にあります。(bin/ というシンボリックリンクからもアクセスできます)
- サンプルは samples/ ディレクトリに置いてあります。
- pythonのファイルについてもpython/以下にコピーしました
- lammpsの src/ 内にあったヘッダファイルはまとめて src/ 以下にコピーしてあります。

並列テストに関するコメント(gcc版と同様のエラーが出るものについては太字で表記)

- balance/balance.var.dynamics: 軽微な(?)数値エラー(TotEng(3.99e-5), Press(0.00019)他)**

- **balance/balance.kSPACE:** 数値エラー(TotEng(0.17), Press(0.62)他)
- **balance/balance.clock.dynamic:** 軽微な(?)数値エラー(TotEng(1.23e-5), Press(9.36e-5))
- **VICOSITY/nemd.2d:** 数値エラー(TotEng(0.12), Press(0.35), v\_visc(5.16)他)
- **USER/eff/ECP/Si2H6/Si2H6.ang:** 途中でフリーズ
- **USER/eff/CH4/ch4\_ionized.dynamics:** 途中でフリーズ

その他テスト関連(未実行テスト、エラーが発生するテスト、個別テストに関して)

- mscg: パッケージ未導入
- USER/quip: パッケージ未導入
- kim: パッケージ未導入
- gcmc: パッケージ未導入
- tad/tad: 動作方法別。実際に実行しても動かず。(4バイト変数と8バイト変数のやりとりに関するエラー)
- neb, dpd/dpdx-shardlow: 動作方法別。スキップ
- imd, USER/lb: 時間がかかりすぎるのでスキップ
- COUPLE, USER/misc/i-pi, USER/misc/grem: スキップ (COUPLE/multiple だけは正常動作確認。他は未検証)
- ASPHERE/tri: ERROR: Illegal neigh\_modify command
- ASPHERE/line: ERROR: Illegal neigh\_modify command
- ASPHERE/poly: ERROR on proc 0: Neighbor list overflow, boost neigh\_modify one
- USER/eff/Li-dendritic: ファイル足りず? 動作させられず
- USER/atc: ファイル足りず
- USER/misc/pimd: 要ディレクトリ作成+検証方法が不明(?)

その他

- vmd molfile plugin の実体は /local/apl/lx/vmd193/lib/plugins/LINUXAMD64/molfile ディレクトリにあります

## 更新履歴

- 2018/7/4: LAMMPSとLATTEのライセンスは GPLv2以降 だったので問題ありませんでした。
- 2018/7/3: LATTEの内部で利用するMETISのライセンス(Apache License 2.0)とLAMMPSのGPLv2が共存不能であるため、ひとまずLATTEを公開パッケージから外しました(上の手順で該当箇所のコメント 指示子を外せばコンパイルは可能です)。METIS無しでのLATTE導入はおそらく可能ですが、今のところ保留しています。