Linux サーバ版 OpenMX インストールマニュアル

2024年10月8日

本マニュアルの目的

本マニュアルでは、単一ユーザが独占的にLinux サーバ (Rocky 9)を使用して OpenMX のジョブを並列実行するための 環境構築方法と Winmostar のリモートジョブ投入機能からLinux サーバへジョブを投入する方法を示しています。

<u>計算環境は全てユーザのホームディレクトリ配下で行うことを想定しています。</u>複数ユーザが使用する共用サーバの環境 を構築する方法、複数ノードを利用する環境を構築する方法などは本マニュアルに含まれませんので、別途お問い合わせ ください。

なお、本マニュアルではLinux サーバ上でジョブスケジューラ TORQUE が使用可能であると仮定しています¹。また以下の命名としています。

サーバホスト名:remote_server

ユーザ名:winmostar user

また、ここでインストールする OpenMX のバージョンは 3.9 ですが、OpenMX3.8 のインストール手順をVに詳述します。

Rocky8 と違って Rocky9 の場合 lapack をソースからビルドする必要があります。

- I. インストール準備
 - ① 以下のサイトにアクセスし、openmx3.9.tar.gz と patch3.9.9.tar.gz をダウンロードする。 http://www.openmx-square.org/download.html

Download of OpenMX

Available packages in terms of GNU-GPL Version 3 (GPLv3)

openmx3.9 (release date: 03/Dec./2019, 158 MB)	+patch (02/Jan./2019)	README.txt
openmx3.8 (release date: 03/April/2016, 136 MB)	+ patch (12/June/2018)	README.txt
openmx3.7 (release date: 23/May/2013, 112 MB)	+ patch (21/Feb./2015)	README.txt

 ② 以下のサイトにアクセスし、fermisurferの1.7.1.tar.gzをダウンロードする。 https://github.com/mitsuaki1987/fermisurfer/releases/

 <u>http://www.adaptivecomputing.com/products/open-source/torque/</u>を参照。管理者権限で yum などを用いて導入する。 あるいは、X-Ability のサイト内 <u>http://winmostar.com/jp/gmx4wm_jp_linux.html</u> (5. Torque) 参照。

FermiSurfer 1.7.1 is released	
Bug fix	
• Mac OSXでコンパイルが出来ない問題を修正しました (naotoumezawa氏、Louis Bouchard氏の厚意	による)。
▼Assets 2	
Source code (zip)	Nov 27, 2016
Source code (tar.gz)	Nov 27, 2016

- ③ openmx3.9.tar.gz, patch3.9.9.tar.gz と 1.7.1.zip を FTP などで Linux サーバに転送する。ここで はホームディレクトリ直下の
 - ~/openmx3.9.tar.gz
 - ~/patch3.9.9.tar.gz
 - ~/1.7.1.zip
 - に置いたと仮定する。
- ④ Linux サーバにログインし、次のコマンドを実行することで必要なパッケージをインストールする。
 - ・OpenMX のビルドに必要
 - \$ sudo dnf config-manager --set-enabled crb
 - \$ sudo dnf install epel-release
 - \$ sudo dnf install fftw3-devel lapack-devel blas-devel openmpi-devel gcc-c++ gcc-

gfortran make scalapack-openmpi-devel wget

- ・Fermisurfer のビルドに必要
- \$ sudo dnf install freeglut-devel
- ・Winmostarから OpenMX を実行する際に必要
- \$ sudo dnf install dos2unix

⑤ 環境変数を設定する。

- \$ export PATH=/usr/lib64/openmpi/bin/:\$PATH
- \$ export LIBRARY_PATH=/usr/lib64/openmpi/lib/:\$LIBRARY_PATH
- \$ export PATH=\${HOME}/openmx3.9/work/:\$PATH
- \$ export OPENMX DATA PATH=~/openmx3.9/DFT DATA19

- II. LAPACK のビルド
 - \$ wget https://github.com/Reference-LAPACK/lapack/archive/v3.9.0.tar.gz
 - \$ tar xvfz v3.9.0.tar.gz
 - \$ cd lapack-3.9.0
 - \$ cp make.inc.example make.inc
 - \$ make blaslib
 - \$ make lapacklib

III. OpenMX のビルド

```
① 圧縮ファイルを解凍する。
```

- \$ cd
- \$ tar xvf openmx3.9.tar.gz
- パッチをあてる。
 - \$ cd openmx3.9/source
 - \$ tar xvf .../patch3.9.9.tar.gz
- ③ makefile を編集する。

```
$ vi makefile
```

8-11 行を以下のようにコメントアウトする。

#MKLROOT = /opt/intel/mk1

```
#CC = mpicc -03 -xHOST -ip -no-prec-div -qopenmp -I/opt/intel/mkl/include/fftw
```

```
#FC = mpif90 -03 -xHOST -ip -no-prec-div -qopenmp
```

```
#LIB= -L${MKLROOT}/lib/intel64 -lmkl_scalapack_lp64 -lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core
-lmkl_blacs_openmpi_lp64 -lmpi_usempif08 -lmpi_usempi_ignore_tkr -lmpi_mpifh -liomp5 -lpthread -lm
-ldl
```

```
コメントアウトした直下に以下を書き加える。
CC = mpicc -fopenmp -O3 -z muldefs -Dkcomp
FC = mpif90 -fopenmp -O3 -fallow-argument-mismatch -Dkcomp
LIB = -lfftw3 ~/lapack-3.9.0/liblapack.a ~/lapack-3.9.0/librefblas.a -lmpi -
```

lgfortran -lmpi_mpifh -lscalapack

```
    ④ コンパイルとインストールを行う。
    $ make all
    $ make install
```

動作確認を行う。

\$ cd ../work
 \$ openmx Cdia.dat
 処理が終わって以下のメッセージが表示されたら正常に動作している。

The number of threads in each node for OpenMP parallelization is 1.

(途中省略)

RestartFileDFT	=	0	0.002	0	0.002
Mulliken_Charge	=	0	0.002	0	0.002
FFT(2D)_Density	=	0	0.011	0	0.011
Others	=	0	3.638	0	3.638

The calculation was normally finished.

- ⑥ Linux サーバにログインし直し、以下のコマンドを実行してインストールした OpenMX が表示されることを確認する。
 - \$ export PATH=/usr/lib64/openmpi/bin/:\$PATH
 - \$ export PATH=\${HOME}/openmx3.9/work/:\$PATH
 - \$ export OPENMX_DATA_PATH=~/openmx3.9/DFT_DATA19

\$ which openmx

IV. Fermisurfer のビルド

```
    圧縮ファイルを解凍する。
    $ cd
    $ unzip 1.7.1.zip
```

- ② コンパイルとインストールを行う。
 \$ cd fermisurfer-1.7.1/src
 \$ make all
- ⑦ Linux サーバにログインし直し、以下のコマンドを実行してインストールした bxsf2frmsf が表示されることを確認する。
 - \$ export PATH=\${HOME}/fermisurfer-1.7.1/src:\$PATH
 - \$ which bxsf2frmsf
- V. Winmostar から OpenMX をリモートジョブ実行する方法
 - ① Winmostar を起動しファイル | 新規ファイルをクリックする。
 - ② ファイル | インポート | Samples ファイル | dia.mol2 をクリックする。
 - ③ 破棄して読み込みをクリックする。
 - ④ **固体 | OpenMX | キーワード設定**をクリックする。
 - ⑤ OpenMX Keyword Setup ウィンドウ右下の OK をクリックする。

その後、<u>ユーザマニュアルのリモートジョブの実行手順</u>に従って操作を行い、Get All Files ボタンを押してファ

イルを取得するところまで実行する。

テンプレートスクリプトを編集する際には、「# Insert commands here」 から「# Do not modify the followings」の間に必要な環境変数を設定する。

⑥ MD | OpenMX | ログを表示をクリックし、デフォルトで選択されたファイルを開き、正常終了したことを確認する。

- VI. (補足) OpenMX3.8 のインストール手順
 - ① 以下のサイトにアクセスし、openmx3.8.tar.gzとpatch3.8.5.tar.gzをダウンロードする。 http://www.openmx-square.org/download.html

Download of OpenMX

Available packages in terms of GNU-GPL Version 3 (GPLv3)

 openmx3.9 (release date: 03/Dec./2019, 158 MB) 	+ patch (02/Jan./2019)	README.txt
openmx3.8 (release date: 03/April/2016, 136 MB)	+ patch (12/June/2018)	README.txt
openmx3.7 (release date: 23/May/2013, 112 MB)	+ patch (21/Feb./2015)	README.txt

② openmx3.8.tar.gz, patch3.8.5.tar.gzをFTPなどでLinuxサーバに転送する。ここではホームディレクトリ直下の

/ | / 直 | •/

~/openmx3.8.tar.gz

~/patch3.8.5.tar.gz

に置いたと仮定する。

③ Linux サーバにログインし、次のコマンドを実行することで必要なパッケージをインストールする。

- ・OpenMX のビルドに必要
- \$ sudo dnf config-manager --set-enabled crb

\$ sudo dnf install fftw3-devel lapack-devel blas-devel openmpi-devel gcc-c++ gccgfortran make

- **)** - - -
- ④ ~/.bashrcに次を追加する。

export PATH=/usr/lib64/openmpi/bin/:\$PATH export LIBRARY_PATH=/usr/lib64/openmpi/lib/:\$LIBRARY_PATH export PATH=\${HOME}/openmx3.8/work/:\$PATH export OPENMX_DATA_PATH=~/openmx3.8/DFT_DATA13

⑤ ~/.bashrc の設定を有効化する。

```
$ source ~/.bashrc
```

⑥ 圧縮ファイルを解凍する。

```
$ tar xzvf openmx3.8.tar.gz
```

- ⑦ パッチをあてる。
 - \$ cd openmx3.8/source
 - \$ tar xzvf ../../patch3.8.5.tar.gz
- makefile を編集する。

\$ vi makefile

139-141 行を以下のようにコメントアウトする。

#CC = mpicc -03 -xHOST -ip -no-prec-div -openmp -I/opt/intel/mkl/include/fftw
#FC = mpif90 -03 -xHOST -ip -no-prec-div -openmp
#LIB= -L/opt/intel/mkl/lib -mkl=parallel -lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -lpthread
-lifcore -lmpi -lmpi_f90 -lmpi_f77

コメントアウトした直下に以下を書き加える。

CC = mpicc -fopenmp -03 -z muldefs FC = mpif90 -fopenmp -03 -fallow-argument-mismatch LIB = -lfftw3 -llapack -lblas -lmpi -lgfortran -lmpi_mpifh

⑨ コンパイルとインストールを行う。

\$ make all

\$ make install

以上