LITESOPH CLUSTER INSTALL

The toolkit consists of several Python-based layers driven by popular and open-source TDDFT codes like OCTOPUS, GPAW and NWChem.

We are going to use only one set of compilers to install all the 3 software.We are mainly using intel software

1. Intel compilers
2. Intel mpi compilers
3. Intel mkl
4. Cmake

Litesoph dependencies

1. Octopus - 11.4 or greater
2. Nwchem - version 7.0.0 or greater
3. Gpaw - 23.6.1

Litesoph Github Page - https://github.com/aitgcodes/litesoph

After installing these software We will proceed to install litesoph software.

1 . OCTOPUS

Compilers used

ml compiler/intel/2020_4
ml cmake/3.24.2
ml gnu8/8.3.0

Dependency -- LIBXC ,FFTW,GSL

wget http://www.tddft.org/programs/libxc/down.php?file=6.2.2/libxc-6.2.2.tar.gz

wget https://www.fftw.org/fftw-3.3.10.tar.gz

untar the file

./configure --prefix="/home/apps/MSCC/lite/octopus" CC=mpiicc FC=mpiifort CCFLAGS=-O3 CFLAGS=-O3
make -j8
make install

GSL

Wget https://mirrors.hopbox.net/gnu/gsl/gsl-2.7.tar.gz

tar -xvf gsl-2.7.tar.gz

go to the gsl directory

./configure --prefix="/home/apps/MSCC/lite/octopus" CC=mpiicc --disable-shared --enable-static

make -j8
make install

OCTOPUS

wget https://octopus-code.org/download/13.0/octopus-13.0.tar.gz

tar -xvf octopus-13.0.tar.gz

go to the directory

./configure --prefix="/home/apps/MSCC/lite/octopus" CC=mpiicc CXX=mpiicpc FC=mpiifort CFLAGS="-O3" FCFLAGS="-O3" \
--enable-mpi --enable-openmp \
--with-blas="-L${MKLROOT}/lib/intel64_lin -lmkl_rt -lpthread -lm -ldl" \
--with-libxc-prefix="/home/apps/MSCC/lite/octopus" \
--with-fftw-prefix="/home/apps/MSCC/lite/octopus/lib" \
--with-blacs="-L${MKLROOT}/lib/intel64_lin -lmkl_rt -lpthread -lm -ldl" \
--with-scalapack="-L${MKLROOT}/lib/intel64_lin -lmkl_rt -lpthread -lm -ldl" \
--with-gsl-prefix="/home/apps/MSCC/lite/octopus" \
--disable-zdotc-test \
FCFLAGS_FFTW="-I/home/apps/MSCC/lite/octopus/include"

AFTER INSTALLING

ml intel/2020.4

export PATH=/home/apps/MSCC/lite/octopus/bin:$PATH

2 . NWCHEM

```
ml compiler/intel/2020_4
ml cmake/3.24.2
ml gnu8/8.3.0

 cd nwchem-7.2.1/

export PYTHONVERSION=3.11
export NWCHEM_TOP=$(pwd)
echo $NWCHEM_TOP
export NWCHEM_TARGET=LINUX64
export LARGE_FILES=TRUE
export USE_NOFSCHECK=TRUE
export USE_OPENMP=y

export USE_MPI=y
export USE_MPIF=y
export
MPI_LOC=/opt/ohpc/pub/syssoftware/intel2017/compilers_and_libraries_2017.4.196/
linux/mpi/intel64/bin
export MPI_LIB=$MPI_LOC/lib
export MPI_INCLUDE=$MPI_LOC/include

 export BLAS_SIZE=8
 export BLASOPT="-L${MKLROOT}/lib/intel64_lin -lmkl_rt -lpthread -lm -ldl"
 export LAPACK_SIZE=8
 export LAPACK_LIB="${BLASOPT}"
 export USE_SCALAPACK=y
 export SCALAPACK_SIZE=8
 export SCALAPACK="-L${MKLROOT}/lib/intel64_lin -lmkl_rt -lpthread -lm -ldl"
 export FC=ifort
 export CC=icc
 export CXX=icpc
 cd src
 make nwchem_config NWCHEM_MODULES=all
 make -j32
```

AFTER INSTALLING

```
module load intel/2017.0.4.196
export PATH=$PATH:/home/apps/MSCC/lite/nwchem/nwchem-7.0.0/bin/LINUX64
```

3 . GPAW – 23.6.1

ml cmake/3.14.3
ml intel/2020.4
ml python/3.9

Dependency - LIBXC ,FFTW

LIBXC and FFTW

./configure --prefix=/home/apps/MSCC/lite/gpaw CC=icc CFLAGS="-O2 -fPIC" --enable-shared --disable-fortran

make -j8
make install

GPAW

wget https://files.pythonhosted.org/packages/30/59/f01172c6ab5d4ed6bc8a9fb1b8547eba823a929c9e2c26da81237816ed2b/gpaw-23.6.1.tar.gz#sha256=ff56d323a499972c8991770a6ab0334a6dd18df36e9c94360e0aa1ddf8867dfd

tar -xvf gpaw-23.6.1.tar.gz

cd gpaw-23.6.1/

we need to create a siteconfig.py file

vi siteconfig.py

```
# openmp
extra_compile_args += ['-fopenmp']
extra_link_args += ['-fopenmp']
extra_link_args += ['-fPIC', '-O2']
extra_link_args += ['-mkl=sequential']

# compiler
```

```
compiler = 'icc'
mpicompiler = 'mpiicc'
mpilinker = 'mpiicc'
# platform_id = ''

# FFTW3:
fftw = True
if fftw:
    include_dirs += ['/opt/ohpc/pub/intel2020_u4/compilers_and_libraries_2020/linux/mkl']

# ScaLAPACK (version 2.0.1+ required):
scalapack = True
if scalapack:
    libraries += ['mkl_def', 'mkl_scalapack_lp64', 'mkl_blacs_intelmpi_lp64']

# LibXC:
# In order to link libxc installed in a non-standard location
# (e.g.: configure --prefix=/home/user/libxc-2.0.1-1), use:

# - static linking:
if True:
    xc = '/home/apps/MSCC/lite/gpaw/'
    include_dirs += [xc + 'include']
    extra_link_args += [xc + 'lib/libxc.a']
    if 'xc' in libraries:
        libraries.remove('xc')

# - dynamic linking (requires rpath or setting LD_LIBRARY_PATH at runtime):
if 0:
    xc = '/home/apps/MSCC/lite/gpaw/'
    include_dirs += [xc + 'include']
    library_dirs += [xc + 'lib']
    # You can use rpath to avoid changing LD_LIBRARY_PATH:
    extra_link_args += ['-Wl,-rpath={xc}/lib'.format(xc=xc)]
    if 'xc' not in libraries:
        libraries.append('xc')
```

```
python3.9 -m venv g
source ./g/bin/activate
```

it will activate the environment g

```
pip install -U pip
pip install numpy scipy ase cymem cython decorator mpi4py pytest pyyaml

python setup.py build
python setup.py install
```

after successful installation

gpaw –version

gpaw info

TO RUN TEST

export GPAW_SETUP_PATH=/home/apps/MSCC/lite/gpaw/gpaw-setups-0.9.20000

and run

gpaw test

---

4 - LITESOPH

It requires Python 3.7.6 or later

ml Anaconda/Anaconda3-2023.03

export PYTHONUSERBASE=/home/apps/MSCC/lite/litesoph

python3 -m pip install /home/apps/MSCC/lite/litesoph/litesoph-main/

export PATH=/home/apps/MSCC/lite/litesoph/bin:$PATH

EXPORT this

```
export PATH=$PATH:/home/apps/MSCC/lite/octopus/bin
export PATH=$PATH:/home/apps/MSCC/lite/nwchem/nwchem-7.0.0/bin/LINUX64
export PATH=$PATH:/home/apps/MSCC/lite/bin
source /home/apps/MSCC/lite/gpaw/g/bin/activate
export PATH=/home/apps/MSCC/lite/litesoph/bin:$PATH
```

after this run command

litesoph config -c

it will detect all of our dependency after recognising this

litesoph gui

it will open the litesoph gui

5 - MODULE FILE

vi litesoph

```
#%Module
proc ModulesHelp { } {
   puts stderr "This modulefile defines the system paths and"
   puts stderr "environment variables needed to use the"
   puts stderr "LITESOPH software package"
   puts stderr "MSCC-NSM software"
}

module load intel/2020.4

setenv      GPAW_SETUP_PATH      /home/apps/MSCC/lite/gpaw/gpaw-setups-0.9.20000
setenv      PYTHONUSERBASE       /home/apps/MSCC/lite/litesoph

prepend-path    PATH        /home/apps/MSCC/lite/octopus/bin
prepend-path    PATH        /home/apps/MSCC/lite/nwchem/nwchem-7.0.0/bin/LINUX64
prepend-path    PATH        /home/apps/MSCC/lite/bin/
prepend-path    PATH        /home/apps/MSCC/lite/gpaw/g/bin
prepend-path    PATH        /home/apps/MSCC/lite/gpaw/gpaw-setups-0.9.20000
prepend-path    PATH        /home/apps/MSCC/lite/litesoph/bin
```
