

PSI4-1.9.1

Webpage

<https://psicode.org/>

Version

1.9.1

Build Environment

- Intel oneAPI Compiler Classic 2023.2.0
- cmake 3.28.3
- (MKL and many packages are installed in conda environment; see below)

Files Required

- psi4-1.9.1.tar.gz
- (many packages are obtained during the installation)

Build Procedure

conda environment (miniforge)

(short summary of setup)

```
$ sh Miniforge3-Linux-x86_64.sh
...
[...] >>> /apl/psi4/1.9.1/miniforge3
...
$ /apl/psi4/1.9.1/miniforge3/bin/conda shell.bash hook > /apl/psi4/1.9.1/conda_init.sh
$ /apl/psi4/1.9.1/miniforge3/bin/conda shell.tcsh hook > /apl/psi4/1.9.1/conda_init.csh
$ . /apl/psi4/1.9.1/conda_init.sh
$ conda install pybind11 msgpack-python numpy networkx pint pydantic scipy py-cpuinfo psutil
$ conda install matplotlib pytest
$ conda install eigen boost-cpp
$ conda install ambit libecpint libefp gau2grid pcmsolver pylibefp optking pyddx adcc pymdi
$ conda install jsonschema
$ conda install regex
$ conda install mkl-devel
```

psi4

```
#!/bin/sh

# assume miniforge for psi4 was already installed

VERSION=1.9.1
INSTALL_PREFIX=/apl/psi4/${VERSION}

BASEDIR=/home/users/${USER}/Software/PSI4/${VERSION}
TARBALL=${BASEDIR}/psi4-${VERSION}.tar.gz

WORKDIR=/gwork/users/${USER}
PARALLEL=12

#-----

umask 0022
export LANG=C
ulimit -s unlimited

module -s purge
```

```

~/intel/oneapi/compiler/latest/env/vars.sh # 2023.2.0

module -s load cmake/3.28.3
module -s load pbs/22.05.11

cd ${WORKDIR}
if [ -d psi4-${VERSION} ]; then
  mv psi4-${VERSION} psi4-erase
  rm -rf psi4-erase &
fi

# load miniforge3 env
. ${INSTALL_PREFIX}/conda_init.sh

tar xzf ${TARBALL}
cd psi4-${VERSION}

export MATH_ROOT=/apl/psi4/1.9.1/miniforge3
sed -i -e "s/xHost/march=core-avx2/" cmake/xhost.cmake # for 3rd gen AMD EPYC

mkdir build
cd build
cmake .. \
  -DCMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
  -DBUILD_SHARED_LIBS=ON \
  -DENABLE_ambit=ON \
  -DENABLE_bse=ON \
  -DENABLE_CheMPS2=ON \
  -DENABLE_cppe=ON \
  -DENABLE_adcc=ON \
  -DENABLE_ddx=ON \
  -DENABLE_dkh=ON \
  -DENABLE_ecpint=ON \
  -DENABLE_libefp=ON \
  -DENABLE_gdma=ON \
  -DENABLE_PCMSolver=ON \
  -DENABLE_v2rdm_casscf=ON \
  -DENABLE_psi4fockci=ON \
  -DENABLE_mdi=ON \
  -DENABLE_Libint1t=ON \
  -DCMAKE_CXX_COMPILER=icpc \
  -DCMAKE_C_COMPILER=icc \
  -DCMAKE_Fortran_COMPILER=ifort
make -j ${PARALLEL}
make install

ctest -j${PARALLEL}
mkdir -p ${INSTALL_PREFIX}/test_results
cp Testing/Temporary/* ${INSTALL_PREFIX}/test_results

```

Notes

- All the tests have passed. The copy of the test output files are available in /apl/psi4/1.9.1/test_results.
 - However, in another try with the same environmental setting, "dft-pruning" test failed due to a minor numerical error.
- The performance of Intel compiler version is slightly better than GCC one.
- Einsums, simint, ccsort, snsmp2, transqt2 were skipped; to avoid compilation errors.