NWChem 101: a tutorial for experienced quantum chemists *

Jeff R. Hammond
Leadership Computing Facility
Argonne National Laboratory
Argonne, IL 60439
jhammond@mcs.anl.gov

Last edited: December 22, 2009

1 Introduction

The document (101) introduces the molecular quantum chemistry capability of NWChem in pedagogical manner. It should not replace the *NWChem User Manual* (UM), but rather supplement it. The UM is organized by functionality and documents all options while 101 is organized from the most basic features to the most advanced.

1.1 Input File Structure

Figure 1 is a basis NWChem input file which introduces a number of key elements. The important input elements are described briefly in Table 1.1.

The `echo` directive should be considered essential — it is the best way to prevent data rot. It is common to modify input files throughout a project, thus the only way to preserve the exact settings used for a given calculation is to retain the input file with the corresponding output. It is tedious to extract the original input settings from the output file, and in some circumstances, these may not be available at all (e.g. when `print low` or `print none` are used).

The `start` directive is also essentially since it controls the names of files that NWChem will create. It is critical that the RTDB file be unique, as running two jobs simultaneously with the same prefix will frequently result in an error or incorrect behavior. Using `title` is optional and less useful than `echo` with a commented input file.

Comments can be introduced to input files using the `#` symbol. These comments will be echoed along with the rest of the input file. One can comment after a regular input line as demonstrated in the example above. Suggested comments include: (1) a description of where the input geometry came from (e.g. "this is the B3LYP/6-31G optimized geometry"), (2) the citation output from the EMSL Basis Set Exchange (`bse.pnl.gov`), when this is the source of the basis set input, and (3) annotation of undocumented options users, system-specific paramters and `set` directives (described in Section 3.7). Other important characters include `;` and \, which terminate and extend input file lines, respectively.

The `memory`, `permanent_dir` and `scratch_dir` directives are the most important system-dependent settings one must determine. If left blank, NWChem will be limited to 400 MB of memory and use the current working directory (cwd) for all permanent and scratch files. The memory directive requires great care for some methods, and will be discussed in detail in Section 3.1 and Section 4. The simple rule is that the total memory usage should be no more than 80-90% of the physical memory, due to the memory used by the kernel, MPI and other system libraries. The directories used for permanent and scratch files are intended to be on shared and local filesystems, respectively. The RTDB and other critical files which are not too large in size must be seen by all nodes, whereas scratch files, such as two-electron integrals, are designed to be accessed locally. As will often be the case, the TCE

Figure 1: A very simple input file.

```
 1   echo
 2   start water
 3   title "a simple calculation of water"
 4   # COMMENT: these options are system specific
 5   memory total 2000 mb
 6   permanent_dir /home/jeff/scratch/nwchem
 7   scratch_dir /home/jeff/scratch/nwchem
 8   geometry angstroms # angstroms are the default units
 9     O   0.00000000  0.00000000   0.11726921
10     H   0.75698224  0.00000000  -0.46907685
11     H  -0.75698224  0.00000000  -0.46907685
12   end
13   basis
14   * library 6-31G*
15   end
16   task scf energy
```

is an exception to this rule, as some four-index transformation algorithms make intensive use of the shared filesystem, which should not be `/home` on most supercomputers, which have high-performance shared filesystems for scratch, often instead of local disk.

Figure 2: The geometry input for an atom with a forced reduction in symmetry.

```
1   geometry
2     symmetry d2h
3     Ne 0.0 0.0 0.0
4   end
```

1.2 Geometry Input

Geometry input may be specified in both cartesian and zmatrix format. Point-group symmetry will be detected automatically but in some cases it must be set manually. First, not all modules implement non-degenerate point-group symmetry, in which case the symmetry must be set to D_{2h} or one its subgroups (see Figure 2). Some modules detect non-degenerate point-groups and disable symmetry altogether so it recommended that one set the appropriate symmetry manually for TCE jobs. In other cases, it is desirable to use symmetry to generate a molecule from a fragment. An example is given in Figure 3. Finally, an example of a non-trivial zmatrix geometry input is given in Figure 4.

The UM is an excellent resource for the geometry input block so we will not elaborate further on this topic.

Table 1: Important input directives to be used in any type of calculation. Line numbers refer to Figure 1. See Section 1.1 more detail descriptions.

directive	line	purpose
`echo`	1	prints the input file to output stream
`start`	2	initializes the RTDB and other files using given prefix
`title`	3	this title will be printed with each module's output
`#`	4,8	stops parsing at the # symbol
`memory`	5	configures memory usage
`permanent_dir`	6	the directory where persistent files will be stored
`scratch_dir`	7	the directory where scratch files will be stored
`geometry`	8	input block for the molecular geometry
`basis`	13	input block for the basis set
`task`	16	NWChem will execute the directive given here

Figure 3: Geometry input for C_{60} using symmetry completion.

```
 1   geometry print xyz units bohr
 2    symmetry d2h
 3    C  0.0000000   6.5776597   1.3147448
 4    C  2.2251790   5.7277170   2.6899811
 5    C  1.3752363   4.3524807   4.9151601
 6    C  4.3524807   4.9151601   1.3752363
 7    C  2.6899811   2.2251790   5.7277170
 8    C  4.9151601   1.3752363   4.3524807
 9    C  5.7277170   2.6899811   2.2251790
10    C  6.5776597   1.3147448   0.0000000
11    C  1.3147448   0.0000000   6.5776597
12   end
```

Figure 4: Geometry intput for hexafluorobenzene using the zmatrix format.

```
 1   geometry units bohr
 2      zmatrix
 3      X
 4      C 1 RXC
 5      C 1 RXC 2 A60
 6      C 1 RXC 3 A60 2 D180
 7      C 1 RXC 4 A60 3 D180
 8      C 1 RXC 5 A60 4 D180
 9      C 1 RXC 6 A60 5 D180
10      F 1 RXF 2 A60 7 D180
11      F 1 RXF 3 A60 2 D180
12      F 1 RXF 4 A60 3 D180
13      F 1 RXF 5 A60 4 D180
14      F 1 RXF 6 A60 5 D180
15      F 1 RXF 7 A60 6 D180
16      constants
17        RXC    2.63428
18        RXF    5.14195
19        A60   60.00
20        D180 180.00
21      end
22   end
```

1.3 Basis Set Input

The two best ways to specify the basis set in an input are to (1) use the NWChem basis set library (BASLIB) and (2) copy the input generated by the EMSL Basis Set Exchange (BSE). While the NWChem basis set library is extensive, very recent or highly specialized basis sets must be obtained from the BSE. Common basis sets in the BASLIB include those from the Pople series (e.g. 6-31G*), the Dunning series (e.g. cc-pVDZ to d-aug-cc-pV6Z for many elements) and specialized basis sets such as Roos ANO DZ, Sadlej pVTZ (POL), Ahlrichs, DGauss and DeMon fitting basis sets, and many basis sets for heavy elements (e.g. LANL2DZ and CRENBL ECP).

Figure 5 demonstrates the various ways to use the BASLIB. One can specify basis sets for each atom or for all the atoms, provided the basis set is defined for all atoms. One can also specify more than one basis set for each atom, which is needed for adding diffuse functions in some cases. The use of quotation marks and/or underscore characters is necessary for some basis sets.

One must carefully consider the type of angular functions used in conjunction with different basis sets. For example, the Pople basis sets were parameterized for use with cartesian d-functions, and the use of spherical functions will lead to incorrect energies compared to other codes. For the Dunning basis sets, one should use spherical angular functions, both to avoid linear dependency but also because the cost of the extra functions for higher angular momentum is prohibitive in correlated calculations. The default is cartesian angular functions, so one must explicitly specify `spherical` when required.

We will see later that one can name basis sets arbitrarily and swap them as needed as part of advanced techniques for convergence or as part of a multi-task input file.

Figure 5: Specification of the three types of basis sets used in NWChem. One must specify the atomic orbital basis set (`"ao basis"` is the default if not specified) for all calculations, while `"ri-mp2 basis"` and `"cd basis"` are used for RI-MP2 and DFT with density-fitting, respectively.

```
 1   basis "ao basis" cartesian
 2     C library 6-311G*
 3     H library 6-31G
 4   end
 5   basis "ri-mp2 basis" spherical print
 6     * library "cc-pVTZ-RI"
 7     * library "aug-cc-pVTZ-RI_diffuse"
 8   end
 9   basis "cd basis"  spherical
10     * library ahlrichs_coulomb_fitting
11   end
```

1.4 Task Input

The full scope of options for the `task` directive are documented in UM 5.10.1, but we review the basic options here. The format of the `task` directive is "`task <theory> <operation>`" such as "`task scf frequencies`" or "`task mp2 optimize`." Analytic gradients and frequencies are not available for all methods, so one should be careful not to request these operations for such methods if numerical derivatives are undesirable.

2 Basic Input Files

2.1 Hartree-Fock (SCF)

2.2 Density-Function-Theory (DFT)

2.3 Perturbation Theory (MP2)

2.4 Coupled-Cluster Theory (CCSD)

2.5 Multi-Configuration SCF (MCSCF)

3 Advanced Input Files

3.1 Parallel Submission and Performance Optimization

3.2 Multiple Tasks

3.3 Converging SCF/DFT

3.4 Effective Core Potential (ECP) Basis Sets

3.5 Custom Density Functionals

3.6 Undocumented Options

3.7 Set Directives

4 TCE Input Files

Table 2: A list of operations which are efficient (i.e. do not use numerical derivatives) for popular methods. Notation: E=energy, G=gradients (geometry optimization), H=hessian (vibrational frequencies). The availability of open-shell variants and excited-states are also listed.

method	efficient operations	references	excited-states
Hartree-Fock	E, G, H	RHF, ROHF, UHF	Yes
DFT (GGA/hybrid)	E, G, H	RHF, UHF	Yes
DFT (meta-GGA)	E, G	RHF, UHF	No
DFT (DHDF)	E	RHF, UHF	No
MP2 (semidirect)	E, G	RHF, ROHF, UHF	No
MP2 (direct)	E, G	RHF	No
RIMP2	E, G	RHF, UHF	No
CCSD/CCSD(T)	E, G	RHF	No
TCE	E, G[†]	RHF, ROHF, UHF	Yes

[†] The TCE gradients calculations are less efficient than those for the energy alone due to the use of spin-orbital integrals.