Introduction to running Gaussian

Oliver Stueker

February 18th, 2020

Contents

Gaussian Input Files

General Input File Format:

```
%MEM=30GB                                                 {Link0 Commands}
%CHK=mycoolmolecule.chk
# B3LYP/6-31G(d) Opt Freq MaxDisk=400GB                   {Keyword lines}
                                                          {EMPTY LINE}
```

```
This is a very cool molecule                              {Title section}
                                                          {EMPTY LINE}
0 1                                                       {Charge, Multiplicity}
H    0.0000000000    0.0000000000     0.0000000000        {Molecule spec}
O    1.0480000000    0.0000000000     0.0000000000
C    1.5064000000    1.3312900000     0.0000000000
H    2.5540780000    1.3053060000    -0.0000000000
H    1.1371360000    1.7931990000     0.8652100000
H    1.1371360000    1.7931990000    -0.8652100000
                                                          {EMPTY LINE}
                                                          {optional sections}
```

Important: Don't forget to include the following empty lines:

- before *and* after the title section
- after the molecule specifications

Molecule Specifications

Molecules (elements and coordinates) can be specified in a number of different ways, using:

- Cartesian (XYZ) coordinates
- Z-Matrix coordinates
- a combination of XYZ- and Z-Matrix

Further, it is possible to add more information such as:

- Atom-type and partial charge (for Molecular Mechanics calculations)
- non-default Isotopes (spectra/thermo-chemistry from frequency calculations)

MolSpec reference at: http://gaussian.com/molspec/

Z-Matrix

Z-Matrices are an alternative way to specify molecule geometries, which can be more intuitive to use for a chemist. They specify a number of distances (e.g. bond lengths), angles and dihedral angles between the atoms.

1. Atom 1 is placed at the origin.
2. Atom 2 is placed at a distance *R1* of atom 1
   along the Z-axis (hence the name Z-Matrix).
3. Atom 3 is placed at a distance *R2* from atom 1,
   so such atoms 3, 2 & 1 are at an angle *A1*.
4. Atom 4 is placed at
   - a distance *R3* from atom 3
   - spanning an angle *A2* between atoms 4, 3, 2
   - and a dihedral angle *D1* between atoms 4, 3, 2, 1
5. All further atoms are defined by a distance, angle and dihedral with respect to any previously defined atoms.

Indeed atom 3 can also be defined distance 3-1 and angle 3-1-2 and atom 4 via any permutation of the already defined atoms 1, 2 & 3.

Z-Matrix reference http://gaussian.com/zmat/

Z-Matrix Examples

Z-Matrix of H2O2 with values within the coordinates (compact):

```
H
O  1  0.9
O  2  1.4   1  105.0
H  3  0.9   2  105.0   1  120.0

```

Z-Matrix of H2O2 using symbolic values:

```
H
O 1 R1
O 2 R2 1 A
H 3 R1 2 A 1 D
  Variables:
R1 0.9
R2 1.4
A 105.0
D 120.0

```

Important Keywords (Route section)

Output Level

The keyword line starts with a #-symbol either on it's own or followed by a letter T, N or P. This controls the amount of information that printed to the Gaussian output file:

- #N: "Normal" amount of output (same as #)
- #T: "Terse" (reduced) output
- #P: Print additional output

Model Chemistry

In most cases the Model chemistry is defined by a combination of "method" and "basis set". Examples of commonly used methods are:

- HF (Hartree Fock),
- B3LYP (a common DFT method) and
- MP2 (second order Møller-Plesset expansion).

Most methods can be prefixed with R (closed-shell restricted wavefunctions), U (unrestricted open-shell wavefunctions) or RO (restricted open-shell wavefunctions).

Links:

- Model Chemistry Reference: http://gaussian.com/capabilities/
- Hartree Fock theory in "Computational Chemistry" course material

---

Basis Set

A basis set defines a set of basis functions, which are used to calculate the molecular orbitals. Common basis sets are:

- `STO-3G` (minimal basis using “Slater Type Orbitals”),
- `6-31G` (a Pople-type split-valence basis set).
- `6-31G(d)` (same but with d-type polarization functions)
- `cc-pVDZ` (correlation-consisted polarized Double-zeta basis set)

For some methods however, no basis sets can be specified, e.g.: `PM7` (semi-empirical “Parametric Model number 7”) or `G3` (“Gaussian-3” composite method).

Links:

- Basis Set on Wikipedia
- Basis Sets in “Computational Chemistry” course material

Job Type

The job type defines what kind of calculation should be performed:

- `SP` – Single Point – just calculate the energy for the given coordinates
- `OPT` – Geometry Optimization – typically an energy minimization
- `Opt=TS` – Transition State search
- Scanning Potential Energy Surfaces

Notes about Keywords:

- Capitalization does not matter. `OPT` is the same as `Opt` or `opt`.
- keywords and options may be shortened to their shortest unique abbreviation. The keyword is actually `Optimization`, but as no other keyword begins with `Opt`, it can be abbreviated.
- Some keywords can take options (see documentation) e.g. `Opt=Z-Matrix` or or `Opt=TS`. These options can also be combined e.g. `Opt=(TS,Z-Matrix)`.

List of Gaussian Keywords: https://gaussian.com/keywords/

Coordinate Driving in Internal Coordinates:

```
#N  HF/6-31G(d)  Opt=Z-Matrix  NoSymm

H2O2 rotational potential 0.0 - 180.0, HF/6-31G(d) level internal coordinates.

0 1
H1
O2  1 r1
O3  2 r2  1 a1
H4  3 r1  2 a1  1 d1
  Variables:
r1=1.0
r2=1.3
a1=110.0
d1=0.0  S  18  +10.0

```

Coordinate Driving in Redundant Internal Coordinates:

```
#N  HF/6-31G(d)  Opt=ModRed

H2O2 rotational potential 0.0 - 180.0, HF/6-31G(d) level
redundant internal coordinates.

0 1
H1
O2  1 r1
O3  2 r2  1 a1
H4  3 r1  2 a1  1 d1
  Variables:
r1=1.0
r2=1.3
a1=110.0
d1=0.0

1 2 3 4  S  18  +10.0

```

Important Link0 commands

- `%MEM=30GB` – request 30GB of memory
- `%NProcShared=4` – (or `%NProc=4`) request a job for 4 CPUs
- `%CHK=/scratch/username/mycoolmolecule.chk` – set checkpoint filename

Gaussian Link0 commands: http://gaussian.com/link0/

Alternative Way to Specify Memory and CPUs

Order of precedence: Link0 > command line > environment variable

Link0 (%-lines)	command line	environment variable	Note
`%Mem=30GB`	`-m=30GB`	`export GAUSS_MDEF="30GB"`	*
`%NProc=4`		`export GAUSS_PDEF="4"`	†
`%CPU=0-31`	`-c="0-31"`	`export GAUSS_CDEF="0-31"`	‡

See *Equivalency*-tab on http://gaussian.com/options/ or http://gaussian.com/link0/.

*: Unit is indicated as `MB` or `GB` (not `M` or `G` as for SLURM).
†: The command line option `-p` does not work for Gaussian 16.c01.
‡: The Gaussian manual suggests using `%CPU` instead of `%NProc` to make sure that Gaussian processes are "pinned" to the processors, which can avoid cache misses. On Compute Canada clusters, processes are pinned automatically by the scheduler and at the time of job-submission it cannot be determined which specific CPU cores will be assigned to the job. If at all, the `%CPU` notation (or its counterparts) should only be used when requesting whole nodes.

Running Gaussian at Compute Canada

Gaussian License restrictions

Gaussian is only available at Graham and Cedar.

In order to use Gaussian, the PI (supervisor) and each user needs to accept Gaussian's terms of use by sending an email with the following content to support@computecanada.ca:

> I am not a member of a research group developing software competitive to Gaussian.
> I will not copy the Gaussian software, nor make it available to anyone else.
> I will properly acknowledge Gaussian Inc. and Compute Canada in publications.
> I will notify Compute Canada of any change in the above acknowledgement.

Small Gaussian Job

small_gaussian_job.sh:

```
#!/bin/bash
#SBATCH --mem=2G             # memory per node
#SBATCH --time=0-00:30       # expected runtime
#SBATCH --cpus-per-task=1    # cpus as defined by %NProcShared
module load gaussian/g16.c01

g16 < h2o2_b3lyp.com  > h2o2_b3lyp.log
```

h2o2_b3lyp.com:

```
#N  HF/6-31G(d)  Opt

H2O2  HF/6-31G(d) level

0 1
H  0.000000    0.992020    0.835282
O  0.000000    0.650000   -0.104410
O  0.000000   -0.650000   -0.104410
H  0.000000   -0.992020    0.835282

```

Big Gaussian Job

big_gaussian_job.sh:

```
#!/bin/bash
#SBATCH --mem-per-cpu=4000M   # memory-per-cpu
#SBATCH --cpus-per-task=8     # cpus as defined by %NProcShared
#SBATCH --time=03-00:00       # expect run time (DD-HH:MM)
module load gaussian/g16.c01
# use localscatch:
export GAUSS_SCRDIR=$SLURM_TMPDIR
g16 < big_molecule.com  > big_molecule.log
```

big_molecule.com:

```
%NProcShared=8
%MEM=30GB
%CHK=/scratch/<USERNAME>/big_molecule.chk
#N MP2/6-311++G(3df,2p)  OPT  FREQ   MaxDisk=400GB

My Big Molecule with a large basis set

...
```

Specifying Resources with environment vars

big_gaussian_job.sh:

```
#!/bin/bash
#SBATCH --mem=125G            # memory per node
#SBATCH --cpus-per-task=32    # cpus as defined by GAUSS_PDEF
#SBATCH --time=03-00:00       # expect run time (DD-HH:MM)
module load gaussian/g16.c01
# use localscatch:
export GAUSS_SCRDIR=$SLURM_TMPDIR
export GAUSS_MDEF="124GB"     # A bit lower than --mem=
export GAUSS_PDEF="32"

g16 < big_molecule.com  > big_molecule.log
```

big_molecule.com:

```
%CHK=/scratch/<USERNAME>/big_molecule.chk
#N MP2/6-311++G(3df,2p)  OPT  FREQ   MaxDisk=400GB

My Big Molecule with a large basis set

...
```

Note that the unit for `--mem` is `G`, while for `GAUSS_MDEF` it is `GB`!

Running Gaussian in parallel

- Gaussian recommends allocating 4GB or more per CPU-core for calculations with 50+ atoms and/or 500+ basis functions. http://gaussian.com/relnotes/?tabid=3
- Many, but not all stages of Gaussian calculations can be carried out utilizing multiple CPU cores and to varying degrees. Hartree-Fock and DFT energies, gradients and frequencies and MP2 energies and gradients seem to be well supported but comprehensive documentation is lacking.
- Number of CPUs that can be efficiently used depends on:
  - Method
  - number of Atoms
  - number of Basis functions
  - version of Gaussian

Remember:

> Your Mileage May Vary.

Comparison: Large Molecule ($C_20O_4H_28$) with 16, 32 CPUs

```
%MEM=120GB
%NProcShared=16
#N B3LYP/6-311++G(3df,2p) Opt FREQ
```

```
$ seff 26948781
Job ID: 26948781
Cluster: graham
User/Group: stuekero/stuekero
State: COMPLETED (exit code 0)
Cores per node: 16                                  <===
CPU Utilized: 13-06:23:40
CPU Efficiency: 96.40% of 13-18:17:52 core-walltime
Job Wall-clock time: 20:38:37                       <===
Memory Utilized: 37.02 GB
Memory Efficiency: 29.61% of 125.00 GB              <=== !!!
```

```
$ seff 26948862
Job ID: 26948862
Cluster: graham
User/Group: stuekero/stuekero
State: COMPLETED (exit code 0)
Cores per node: 32                                  <===
CPU Utilized: 13-07:42:35
CPU Efficiency: 99.85% of 13-08:11:12 core-walltime
Job Wall-clock time: 10:00:21                       <===
Memory Utilized: 73.00 GB
Memory Efficiency: 58.40% of 125.00 GB              <===
```

Estimating memory usage with freqmem

The `freqmem` utility helps to estimate the required memory for a frequency calculation.

It uses the following syntax:

```
freqmem  natoms  nbasis  r|u  function
```

Example ($C_20O_4H_28$ at B3LYP/6-311++G(3df,2p)):

```
$ grep --max 1 Stoichiometry *.log
 Stoichiometry    C20H28O4

$ grep --max 1 "basis functions," *.log
  1216 basis functions,  1728 primitive gaussians,  1360 cartesian basis functions

$ freqmem 52 1216  r sp
  NAtoms=52 NBasis=1216 closed-shell frequencies with up to SP functions:
  Minimum of 231.66 megawords per thread.
```

1 megaword = 64 bit = 8 bytes

Gaussian freqmem utility: http://gaussian.com/freqmem/

Links

- Gaussian Manual: http://gaussian.com/man/
- Gaussian on CC Docs Wiki: https://docs.computecanada.ca/wiki/Gaussian
- Gaussian error messages: https://docs.computecanada.ca/wiki/Gaussian_error_messages
- "Computational Chemistry" course material: www.schulz.chemie.uni-rostock.de/lehre/computerchemie/table-of-contents/
- Book "Exploring Chemistry with Electronic Structure Methods": http://gaussian.com/expchem3/
- Computational Chemistry List (CCL): http://ccl.net/

Software

- Molecule Editors:
  - Open Source:
    - Avogardro: https://avogadro.cc/
    - Gabedit: http://gabedit.sourceforge.net/
    - Molden: http://cheminf.cmbi.ru.nl/molden/
    - Jmol: http://www.jmol.org/
  - commercial:
    - GaussView: http://gaussian.com/gaussview6
- Tools:
  - OpenBabel: http://openbabel.org/

Online Databases/services

- NIH:
  - PubChem: https://pubchem.ncbi.nlm.nih.gov/
  - NIH CACTUS https://cactus.nci.nih.gov/
  - Online SMILES Translator and Structure File Generator https://cactus.nci.nih.gov/translate/
  - Chemical Structure Lookup Service (CSLS) https://cactus.nci.nih.gov/cgi-bin/lookup/search
- ChemSpider: http://www.chemspider.com/
- ChEMBL: https://www.ebi.ac.uk/chembl/
- Jmol Demo:
  - https://chemapps.stolaf.edu/jmol/jsmol/simple1.htm
  - http://ostueker.github.io/simple_jmol_demo/jmol_surfaces.html