Running Gaussian16 and NBO7 effectively on Nibi and Fir: Performance Issues

Jemmy Hu
SHARCNET
Digital Research Alliance of Canada

Feb. 11, 2026

Gaussian on the new Alliance Clusters

- A licensed software available on Nibi and Fir
- Software are installed as Module packages
- Basic module commands

```
module avail                    # to list all available module software
module spider gaussian          # to see what gaussian versions are available
```

```
[jemmyhu@l1.nibi ~]$ module spider gaussian

----------------------------------------------------------------------------------------
----------------------------------
 gaussian:
----------------------------------------------------------------------------------------
----------------------------------
    Versions:
      gaussian/g03.d01
      gaussian/g09.e01
      gaussian/g16.b01
      gaussian/g16.c01
```

```
[jemmyhu@l1.nibi ~]$ module load gaussian/g16.c01
```

Gaussian Basics

Latest version: g16.c01

Gaussian files
- Input sample
- Restart

Run script .sh
- Resource request (mem, cpu, time)

File location, .chk, .rwf
- Disk: scratch vs local/tmp

Utilities: formchk, freqchk, etc..

https://docs.alliancecan.ca/wiki/Gaussian

Gaussian Files

input file: .com, .gjf

output: .log

runtime files (unnamed): .rwf, .inp, .d2e, .int, .skr

run script: .sh

executable

g16: /localscratch/username.jobid.0/

G16: /scratch/user/jobid/

Checkpoint files: .chk

User can name your own .chk and .rwf files

G16 simplest input: CO-b3lyp.com

```
//Simplest Input, serial, default memory

#p b3lyp/6-311++g(2df,p) opt freq
(blank line)
<< CO-6311++g(2df,p)—B3LYP-Opt-Freq >>
(blank line)
0 1
c
o 1 co

co=1.1281
(blank line)
```

```
//Simplest Input, 16 cpus parallel

%mem=10G
%nproc=16
#p b3lyp/6-311++g(2df,p) opt freq

<< CO-6311++g(2df,p)—B3LYP-Opt-Freq >>

0 1
c
o 1 co

co=1.1281
```

https://gaussian.com/input/

Input file with .chk and named .rwf: CO-b3lyp-rwf.com

```
//specify a .chk file

%chk=CO-b3lyp
%mem=10G
%nproc=16
#p b3lyp/6-311++g(2df,p) opt freq

<< CO-6311++g(2df,p)—B3LYP-Opt-Freq >>

0 1
c
o 1 co

co=1.1281
```

```
//specify a .rwf file

%rwf=CO-b3lyp (or /home/userid/scratch/CO-b3lyp)
%chk=CO-b3lyp
%nosave
%mem=10G
%nproc=16
#p b3lyp/6-311++g(2df,p) opt freq

<< CO-6311++g(2df,p)—B3LYP-Opt-Freq >>

0 1
c
o 1 co

co=1.1281
```

The named .chk and/or .rwf files will be created in the current job directory

Checkpointing

1. .chk can be used to
   restart an optimization job
   used for property (freq, etc) calculation afterwards
   used for GaussView

2. .rwf can be used to restart a job,
   particularly to restart a job failed in the one-step Energy
   and Freq calculation

   (remember the location of the .rwf file)

Restart

Restart opt job from .chk

```
%chk=Cu_test.chk
%mem=10g
%nproc=16
#B3LYP/DGDZVP Opt Geom=AllCheck Guess=Read
(one blank line)
```

Restart freq from .rwf

```
%rwf=/scratch/jemmyhu/Cu_test.rwf
%NoSave
%chk=Cu_test.chk
%mem=10g
%nproc=16
#p restart
(one blank line)
```

Slurm submit/Run script (.sh file):

- account type (default or RAC account)
- compute resource (cpus, memory, runtime)
- job type (mpi, openmp, serial)
- software (module load, run command)

Several ways to run Gaussian on Nibi and Fir

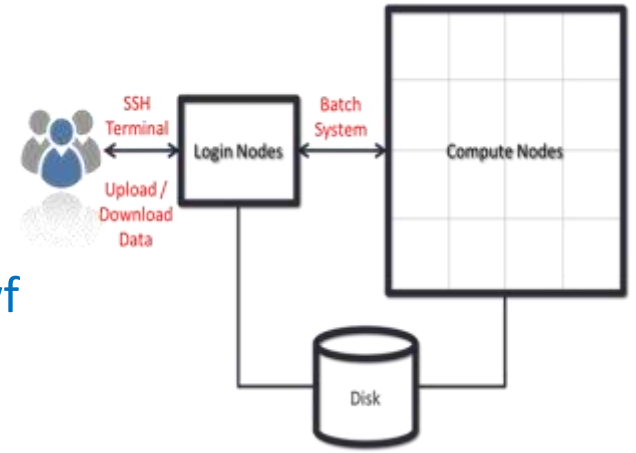
- by default, Gaussian runs as a serial job on one cpu, good for small molecules
- shared-memory parallel (use multi-cpus/cores on one node)

```
#!/bin/bash
#SBATCH --account=def-jemmyhu      # PI's group acount
#SBATCH --mem=20G                  # memory amount, roughly 2 times %mem
#SBATCH --time=00-03:00            # time (DD-HH:MM)
#SBATCH --cpus-per-task=16         # No. of cpus, the same as defined by %nprocs

module load gaussian/g16.c01
g16 < g16_test.com                 # g16 command
or
G16 g16_test.com                   # G16 command
```

[~$] sbatch g16_test.sh

File structure: scratch (G16) vs localscratch (g16)

	module load gaussian/g16.c01 G16 g16_test.com	**module load gaussian/g16.c01** g16 < g16_test.com >& g16_test.log
Runtime files (.rwf, .inp, .d2e, .int, .skr)	$SCRATCH /scratch/userid/jobid/	$SLURM_TMPDIR (/localscratch/username.jobid.0/)
Disk size	20 TB	~3TB
Speed		faster
File life time	Files stay when job failed. Clear up jobid folder yourself	removed right after the job is done
For restart	.rwf left for restart	Name .rwf for possible restart



Job location/folder: .com, .log, .sh; named .chk, .rwf

Gaussian Utilities

Utilities

- formchk
- cubegen
- freqchk
- ……

use the same version

direct run from command line

```
module load gaussian/g16.c01
formchk name.chk
```

Performance Issues for common Gaussian methods

Opt, Freq: B3LYP (DFT)/6-311G(d), MP2 /6-31G(d)

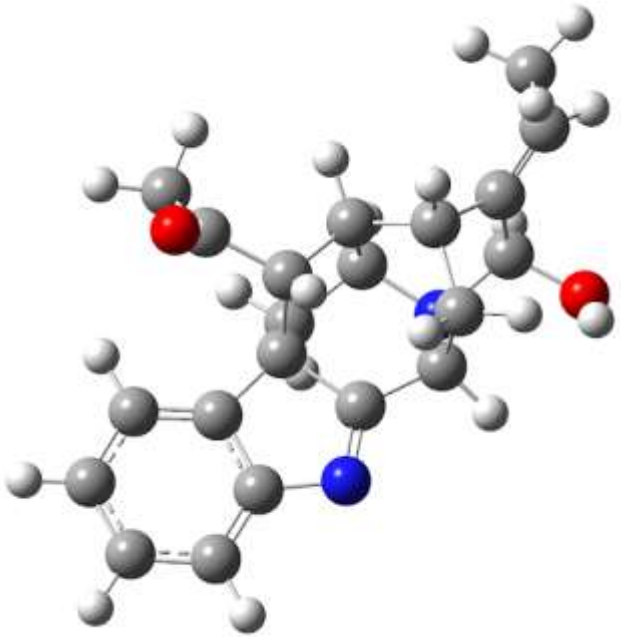
Energy (single point): CISD, CCSD(T) / 6-31G(d)

CISD (configuration interaction with all single and double substitutions)

CCSD(T) (coupled cluster using single, double and triple substitutions)

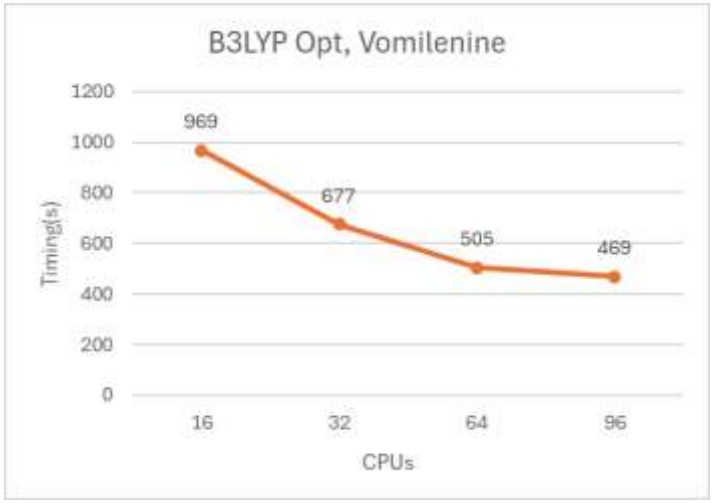
Molecule: Vomilenine

$C_{21}H_{22}N_2O_2$

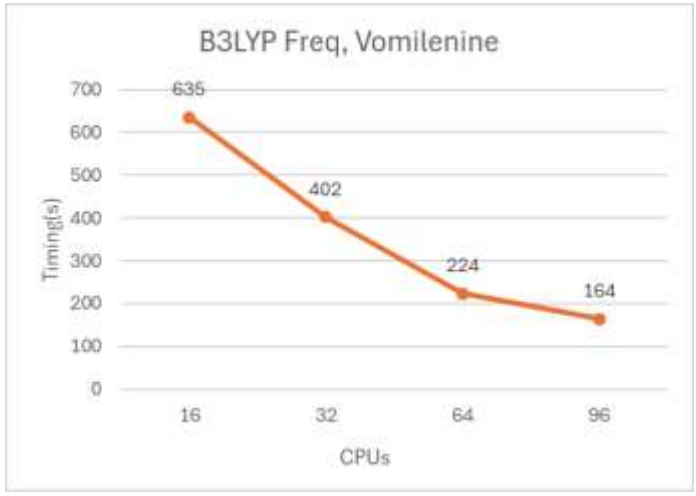


Vomilenine, B3LYP Opt, Freq 6-311G(d)

CPUs	Time(s)	Ratio	Memory(GB)
16	969	1	1.42
32	677	1.43	2.27
64	505	1.92	3.61
96	469	2.07	5.31

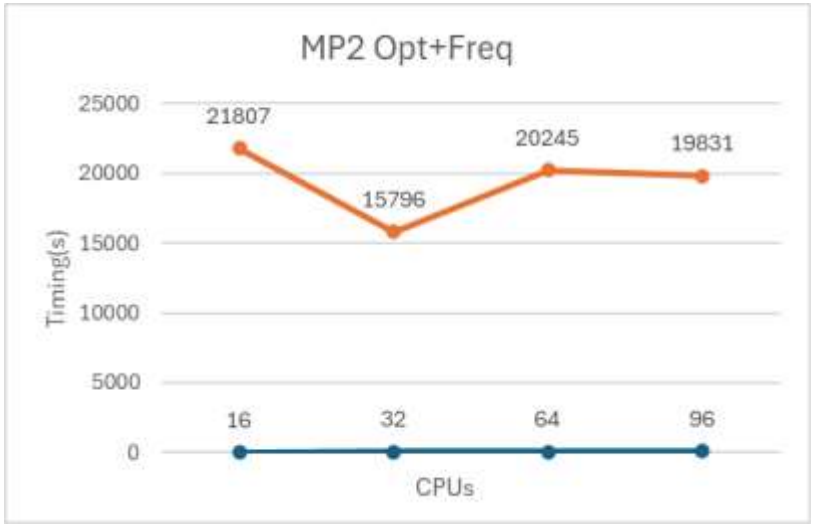


CPUs	Time(s)	Ratio	Memory(GB)
16	635	1	7.55
32	402	1.58	13.54
64	224	2.83	25.29
96	164	3.87	36.75



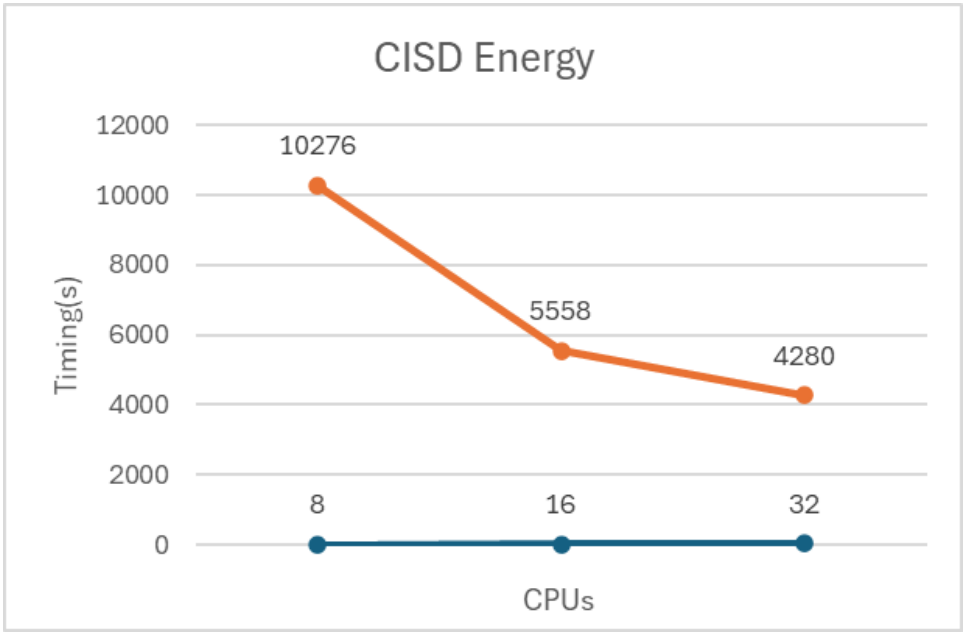
Vomilenine, MP2 Opt+Freq 6-31G(d)

CPUs	Time(s)	Ratio		Memory(GB)
16	21807	1		79.98
32	15796	1.38		79.98
64	20245	1.08		79.98
96	19831	1.1		79.99



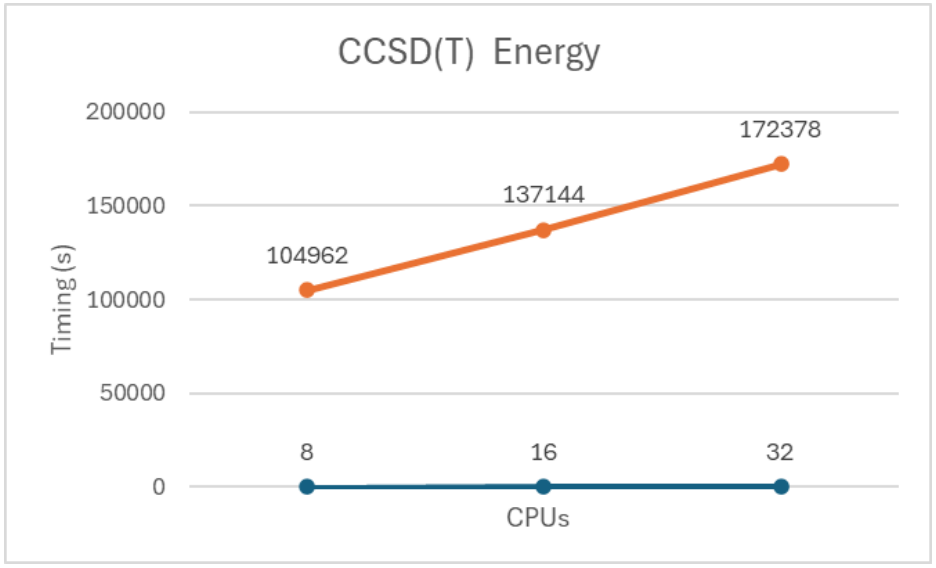
Vomilenine, CISD/6-31G(d) Energy

CPUs	Time(s)	Ratio		Memory(GB)
8	10276	1		138.35
16	5558	1.85		138.36
32	4280	2.4		140.43



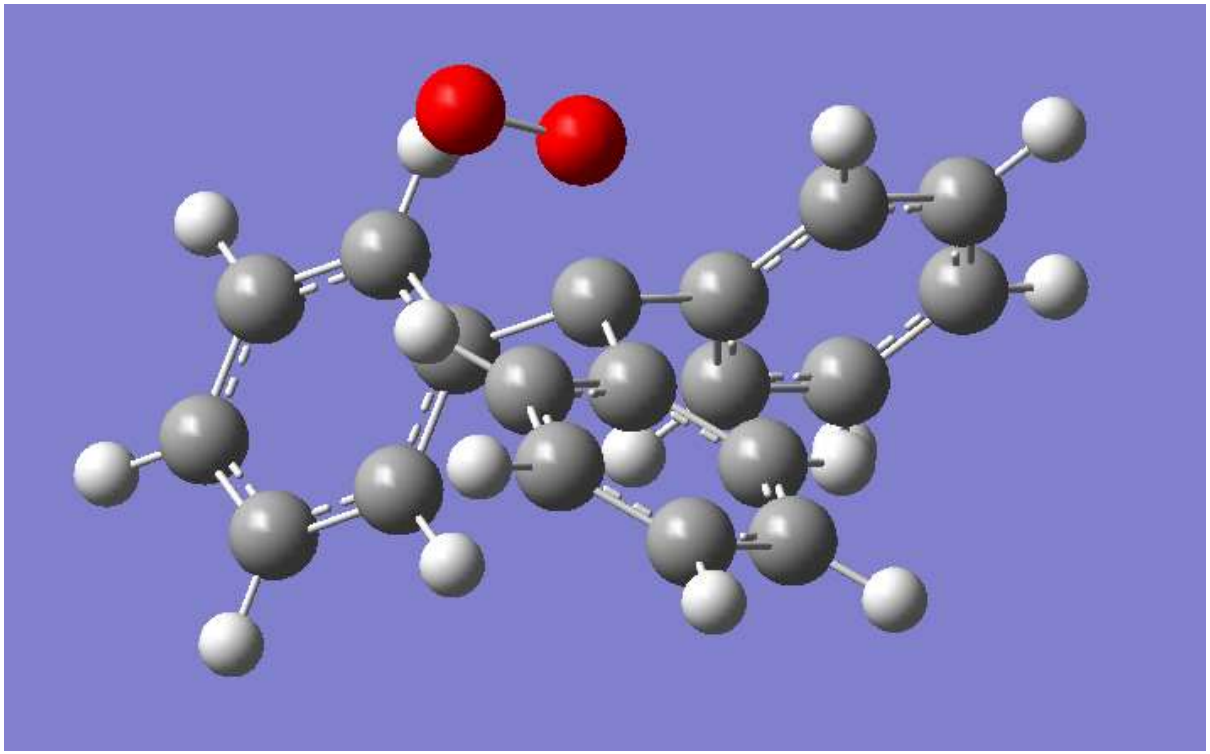
Vomilenine, CCSD(T)/6-31G(d) Energy

CPUs	Time(s)	Ratio	Memory(GB)
8	104962	1	199.98
16	137144	0.765	149.98
32	172378	0.609	283.66



Opt+Freq: UB3LYP (DFT)/6-31G(d)

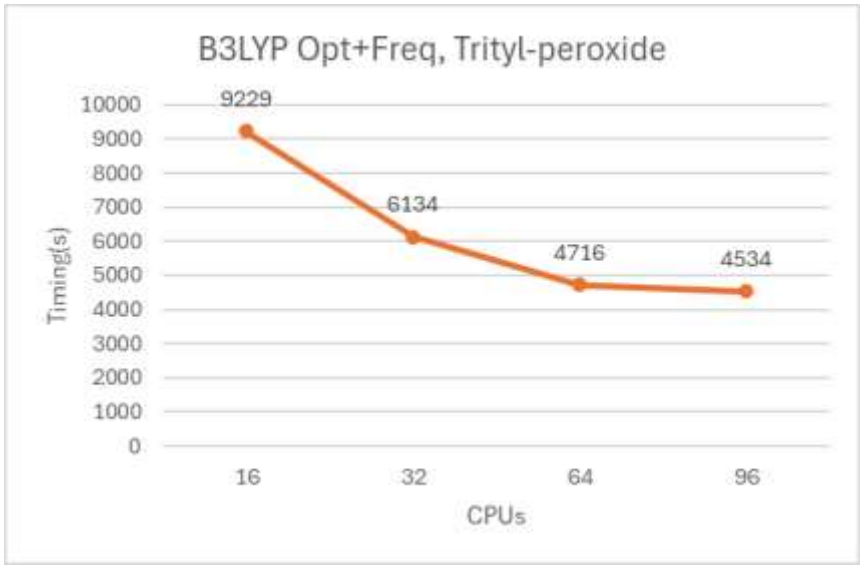
Molecules: trityl radical ($C_{19}H_{15}$)+ triplet oxygen (O_2)



trityl peroxide scan - redundant coordinate scan studying the formation of a C-O bond between trityl radical and triplet oxygen

Trityl-peroxide, UB3LYP Opt+Freq 6-31G(d)

CPUs	Time(s)	Ratio		Memory(GB)
16	9229	1		31.94
32	6134	1.5		33.04
64	4716	1.96		17.55
96	4534	2.04		24.85



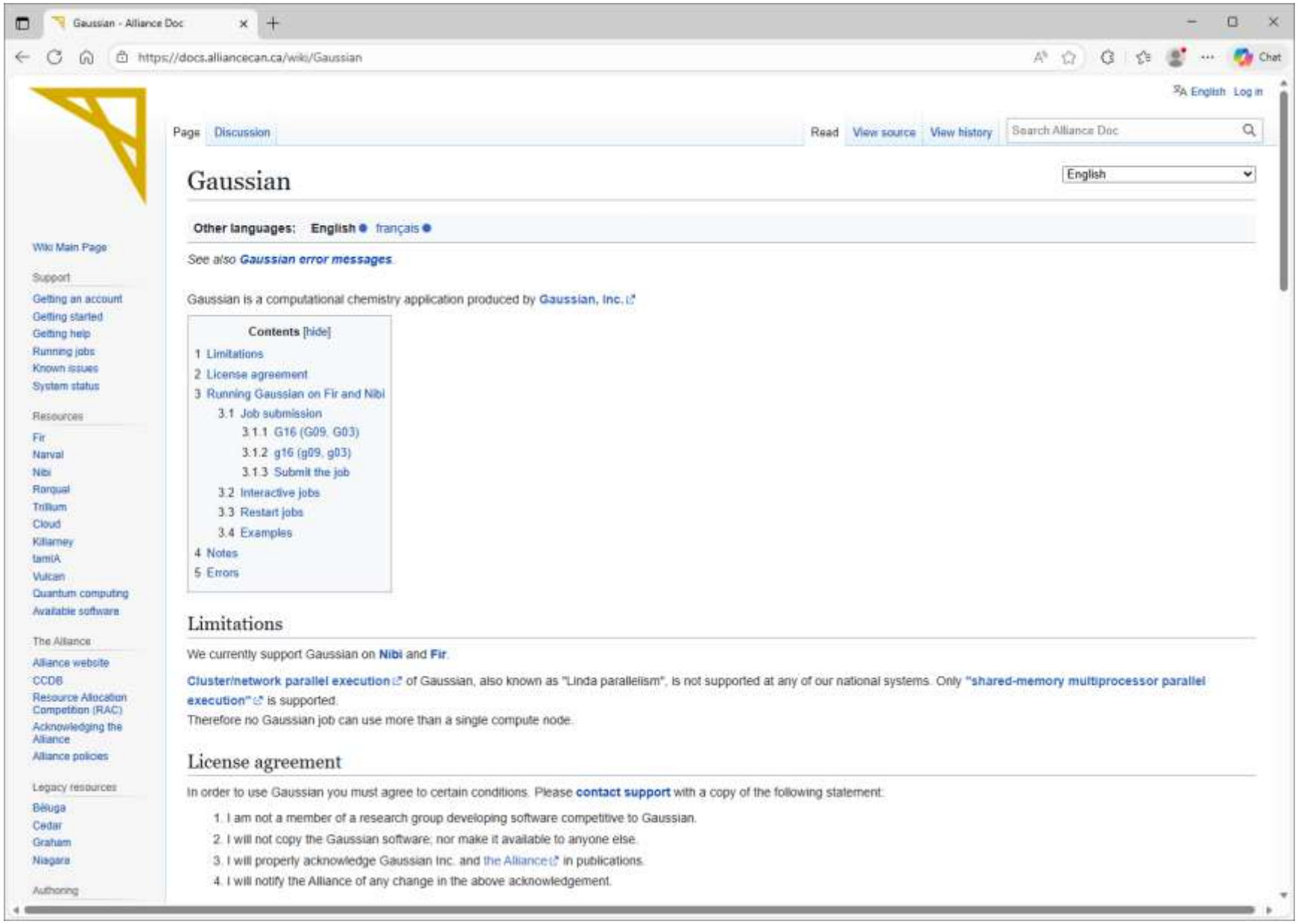
Summary: Optimal CPUs vs. Methods

Method	Optical CPUs	Memory
DFT (B3LYP)	32 (64 for large molecules)	
MP2	32	
CID, CISD	16 or 32	Large memory
CCSD, CCSD(T)	8	Very large memory

32 cpus for DFT, MP2 and/or CI-based methods
8 cpus for CCSD
64 cus for DFT for very large molecules

No more than 64 cpus, particularly do not run full node Gaussian jobs

#SBATCH --mem=0 # all available memory on a node X
#SBATCH --cpus-per-task=96, or 192 X



https://docs.computecanada.ca/wiki/Gaussian

NBO: Natural Bond 0rbital

The NBO program performs the analysis of a many-electron molecular wavefunction in terms of localized electron-pair bonding units.

The program carries out the determination of

- NAOs: natural atomic orbitals
- NHOs: natural hybrid orbitals
- NBOs: natural bond orbitals
- NLMOs: natural localized molecular orbitals

and uses these to perform

- NPA: natural population analysis (NPA),
- NBO energetic (deletions) analysis,
- NRT: natural resonance theory
- NCS: natural chemical shielding

and other tasks pertaining to localized analysis of wavefunction properties

What NBO version is in Gaussian package?

Gaussian has built-in NBO-3.1, the default NBO in Gaussian when using keyword pop=nbo

What NBO versions are on Nibi and Fir?

NBO is not installed as an independent package, it's built-in with Gaussian package on the clusters

Versions:

- gaussian/g03.d01
- gaussian/g09.e01 - NBO6
- gaussian/g16.b01 - NBO6
- gaussian/g16.c01 - NBO7

Recommend to use the latest versions

input

https://gaussian.com/relnotes/

•**[REV C]** NBO version 7 is supported. There are new options to the **Population** keyword:
• **Pop=NPA7**, Natural Population Analysis
• **Pop=NBO7**, full Natural Bond Orbital Analysis
• **Pop=NBO7Read,** full NBO with NBO input read from the input stream
• **Pop=NBO7Delete,** NBO analysis of the effects of deletion of some interactions
• **Pop=NEDA**, Natural Energy Decomposition Analysis

input to the NBO portion consists simply of one or more keywords enclosed in NBO keylists in the ESS input file.
The main NBO keylist (the $NBO keylist) is of the form:

```
$NBO ...(keywords)... $END
```

Simple examples of such $NBO keylists are

```
$NBO dipole nrt $END
$NBO file=myjob archive naonbo ncs=0.05 $END
```

Other keylist identifiers include

$CORE, $CHOOSE, $NRTSTR, $DEL, and $NPEPA

Keylists cannot be nested, and each new keylist must begin on a newline.

The NBO program reads the keywords of each keylist to set various job options, then interrogates the ESS program for information concerning the wavefunction to perform the requested tasks

NBO input is terminated by a blank line.

Thus, keylist input may appear as follows:

```
$nbo  steric  $end
$del
  delete 1 element 3 11
$end
<blank line>
```

Input Example

NBO analysis with geometry optimization

```
# rhf/3-21g opt pop=nbo7

Methylamine...rhf/3-21g optimization

0 1
C
N   1   1.47
H   1   1.09   2   109.4712
H   1   1.09   2   109.4712   3   120. 0
H   1   1.09   2   109.4712   3   240. 0
H   2   1.01   1   109.4712   3    60. 0
H   2   1.01   1   109.4712   3   300. 0
```

numerical optimization with deletions

```
# rhf/3-21g opt=z-matrix pop=nbo7del

Methylamine...reoptimization with LEWIS deletions

0 1
C
N   1   CN
H   1   CHa   2   NCHa
…
H   2   NH    1   CNH    3  -beta 0

CN        1.4712
CHa       1.0901
…
alfa     121.4224
beta      64.2403

$nbo  print=0  nbosum  $end      ! Turn off all printing but NBO Summary
$choose                          ! Fix Lewis structure for all geometries
   lone 2 1 end
   bond s 1 2 s 1 3 s 1 4 s 1 5 s 2 6 s 2 7 end
$end
$del  lewis  $end                ! Perform deletions analysis
```

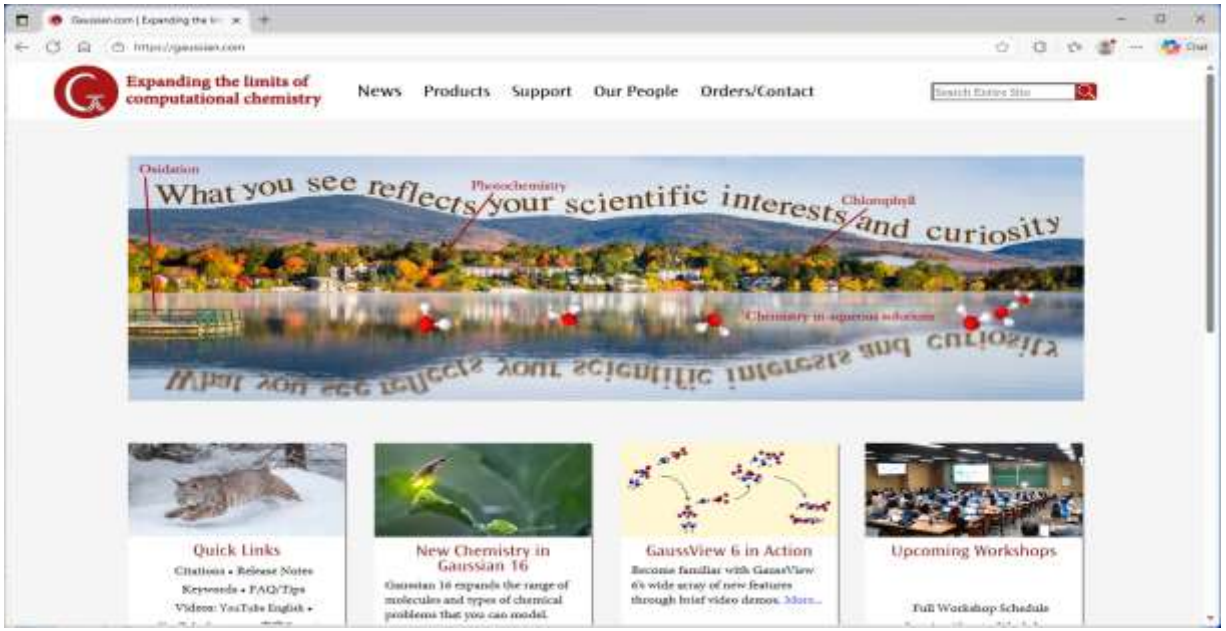
Output

```
********************************** NBO 7.0 **********************************
      N A T U R A L  A T O M I C  O R B I T A L  A N D
     N A T U R A L  B O N D  O R B I T A L  A N A L Y S I S
***************************** Jemmy Hu (S102210) ******************************
 (c) Copyright 1996-2021 Board of Regents of the University of Wisconsin System
    on behalf of the Theoretical Chemistry Institute.  All rights reserved.

       Cite this program [NBO 7.0.10 (8-Feb-2021)] as:

       NBO 7.0.  E. D. Glendening, J. K. Badenhoop, A. E. Reed,
       J. E. Carpenter, J. A. Bohmann, C. M. Morales, P. Karafiloglou,
       C. R. Landis, and F. Weinhold, Theoretical Chemistry Institute,
       University of Wisconsin, Madison, WI (2018)

Filename set to /scratch/jemmyhu/57734053/Gau-2187

Job title: Methylamine...rhf/3-21g optimization


NATURAL POPULATIONS:  Natural atomic orbital occupancies

 NAO Atom No lang   Type(AO)    Occupancy      Energy
-----------------------------------------------------------
  1    C  1  s     Cor( 1s)     1.99978     -11.17840
  2    C  1  s     Val( 2s)     1.08927      -0.19713
  3    C  1  s     Ryd( 3s)     0.00071       1.95478
  4    C  1  px    Val( 2p)     1.21779      -0.06924
```

NATURAL BOND ORBITALS (Summary):

NBO	Occupancy	Energy	Principal Delocalizations (geminal,vicinal,remote)
Molecular unit 1 (CH5N)			
------ Lewis ----------			
1. CR (1) C 1	1.99978	-11.17840	18(g)
2. CR (1) N 2	1.99984	-15.42894	22(g)
3. LP (1) N 2	1.97257	-0.40269	11(v),12(v),13(v)
4. BD (1) C 1- N 2	1.99876	-0.80619	
5. BD (1) C 1- H 3	1.99852	-0.65507	20(v)
6. BD (1) C 1- H 4	1.99370	-0.65295	14(v)
7. BD (1) C 1- H 5	1.99370	-0.65295	15(v)
8. BD (1) N 2- H 6	1.99539	-0.77276	12(v)
9. BD (1) N 2- H 7	1.99539	-0.77276	13(v)
------ non-Lewis ----------			
10. BD*(1) C 1- N 2	0.00009	0.57358	

Examples on Nibi in
/home/jemmyhu/tests/test_Gaussian/g16/g16.c01/NBO7



https://nbo6.chem.wisc.edu/