Introduction to Gaussian

June 22, 2006

Benjamin Lynch

help@msi.umn.edu • 612-626-0802

www.msi.umn.edu

- Description of Code
- How to Create Input Files
- How to Submit Calculations
- How to View Output
- A few notes on solving errors

Gaussian 03:

an electronic structure package capable of predicting many properties of atoms, molecules, reactive systems, *e.g.;*

- molecular energies
- structures
- vibrational frequencies
- electron densities

utilizing *ab initio*, density functional theory, semi-empirical, molecular mechanics, and various hybrid methods.

GaussView:

Graphical interface for Gaussian 03

- sketch molecules
- setup Gaussian 03 input files
- graphically examine results

Molden:

A graphical interface for Gaussian and other programs

- setup Gaussian 03 input files
- graphically examine results

Gaussian, Inc

http://www.gaussian.com

Supercomputing Institute

http://www.msi.umn.edu/tutorial

How to Create Input Files

Description • Input • Submit • Visualize

www.msi.umn.edu

Overview:

% Resource management
# Route card

blank line

Title section

blank line

Molecular coordinates

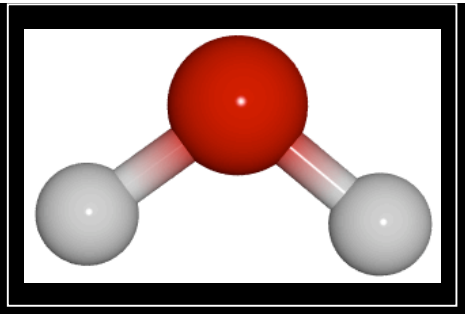
blank line

Geometric variables

blank line

Other input options (isotope masses, include files, …)

blank line



Example: Water Optimization

```
%mem=32mb
#p hf/6-31g opt

hf/6-31g optimization of water

0  1
o
h  1  oh
h  1  oh  2  aoh

oh=0.9
aoh=104.0
```

system resources

computational model
type of calculation

title

charge & multiplicity

structure definition
(z-matrix)

variable values

Commands for Resource Management (link 0)

%mem=*n*	sets the amount of dynamic memory (*n*), default is 32Mb. Units allowed, kb, mb, gb, kw, mw, or gw.
%nproc=*n*	sets the number of processors, *n*, to use
%chk=*file*	location and name of checkpoint *file*
%rwf=*file*	location and name of rwf *file*
%subst l*n dir*	substitute link *n* with alternate executable found in *dir*

Route Card Description:

- specifies keywords and options
- always begins with a # character
- keywords can be specified in any order
- options are grouped in parentheses, ()
- keywords should not be repeated
- route section can be up to 5 lines long
- ends with a blank line

Syntax:

#[p] method/basis [keyword=(opt1,opt2)] [keyword2=(opt1[,opt2])]

Example Routes:

hf/6-31g*

#p B3LYP/cc-pvdz opt

#MP2/cc-pvtz maxdisk=12gb

HF/6-311g(d,p) freq scf=direct nosymm opt

#CCSD=(T,maxcyc=200)/6-31G* opt=ts

Calculation Types:

- single point energy and properties
- geometry optimization
- frequency
- reaction path following/searching

Levels of Theory:

- molecular mechanics
  - mm2, uff
- semi-empirical
  - AM1, PM3, MNDO, …
- density functional theory
  - B3LYP, mPWPW91, custom …
- *ab initio*
  - HF, MP2, CCSD, QCISD, …
- hybrid
  - G2, G3, oniom, …

Basis Set Types:

– all electron; Pople, Dunning, etc.

– effective core potential

User Supplied:

– use GEN keyword, #hf/gen

via input file:

```
C H 0
6-31G(d,p)
****
F 0
6-31G(d',p')
****
```

via external file:

```
@/home/blynch/basis/custom.gbs
```

http://www.emsl.pnl.gov/forms/basisform.html

Spin multiplicity:

multiplicity = *n* + 1

where *n* = # of unpaired electrons

Molecular Coordinate Types:

- Cartesian coordinates
- Z-matrix

Initial guess:

- where do you get initial geometric values?
  - experimental data
  - chemical intuition
  - previous calculation
- how do I generate the geometry?
  - by hand
  - graphical builder

Resource Considerations: %mem

%mem=200mb

200mb should be enough for all interactive jobs and many jobs that you submit to the queue.

Some jobs can take advantage of much more memory.

For large mp2 frequencies %mem=2gb can significantly increase the Speed of the calculation

Resource Considerations: %rwf

%rwf is obeyed by all HF, all DFT, MP2 energies, MP2 gradients

%rwf is ignored by QCISD, CCSD, CCSD(T), MP4, and

MP2 frequency calculations.

Resource Considerations: %chk

The checkpoint file contains a lot of information that does not Appear in the output file.

You can use this command to specify the location of the checkpoint file.

%chk=mychkfile

Resource Considerations: %nproc

This will set the number of processors used for the calculation.

Choosing the Number of Processors:

Parallelized

- HF
- MCSCF
- DFT
- MP2
- CIS

Suggestion: Maximum of 4 processors should be used

Choosing the Number of Processors:

NOT- Parallelized much

- CCSD
- CCSD(T)
- AM1
- Almost everything not listed on the previous slide

Suggestion: Max of 1 processor

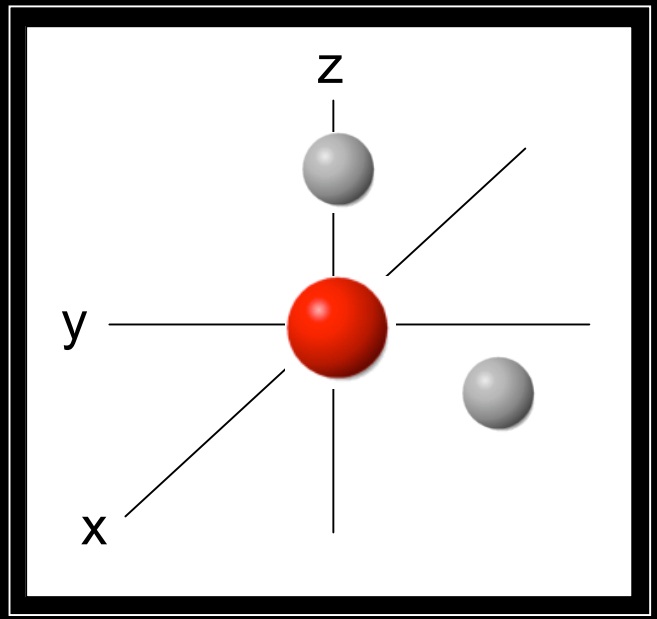
Recommended Number of Processors

Method	Energy	Gradient / Opt	Freq / Hessian
HF	**4**	**4**	**4**
HDFT	**4**	**4**	**4**
Pure DFT	**4**	**4**	**4**
MP2	**4**	**3**	**1-2**
MP3	1	1	
MP4	**2-4**		
MP5	1		
CCD	1	1	
CCSD	1	1	
CCSD(T)	1-2		
CIS	**4**	**3**	
CISD	1	1	
AM1	1	1	

Jobs that take less than 1 hour on the Altix on 1 processor

Method	Energy	Gradient	Freq / Hessian
HF/6-31+G(d,p)	**$C_{18}H_{38}$**		**C_7H_{16}**
B3LYP/6-31+G(d,p)	**$C_{16}H_{34}$**		**C_5H_{12}**
BLYP/6-31+G(d,p)	**$C_{20}H_{42}$**		**C_8H_{18}**
MP2/6-31+G(d,p)	**$C_{14}H_{30}$**	**C_8H_{20}**	**C_5H_{12}**
MP4/6-31+G(d,p)	**C_5H_{12}**		
MP5	**CH_4**		
CCSD/**6-31+G(d,p)**	**C_5H_{12}**	**C_3H_8**	
CCSD(T) /**6-31+G(d,p)**	**C_4H_{10}**		
AM1	**$C_{400}H_{40}$**	**$C_{400}H_{40}$**	

Building by Hand:



Convention:

– first atom placed at origin

– second atom placed along *z*-axis

– third atom placed in *yz*-plane

– subsequent atoms placed in relation to first three

Building with Molden:

- Login remotely

  ssh -X machine_name.msi.umn.edu

- Load proper module

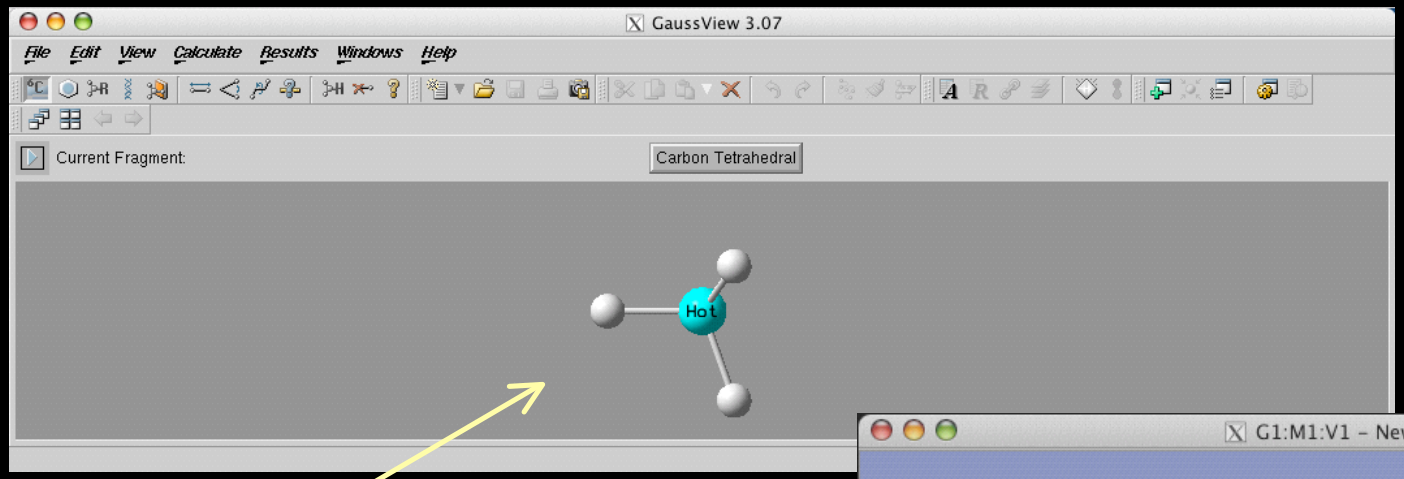
  module add molden

- Launch Molden

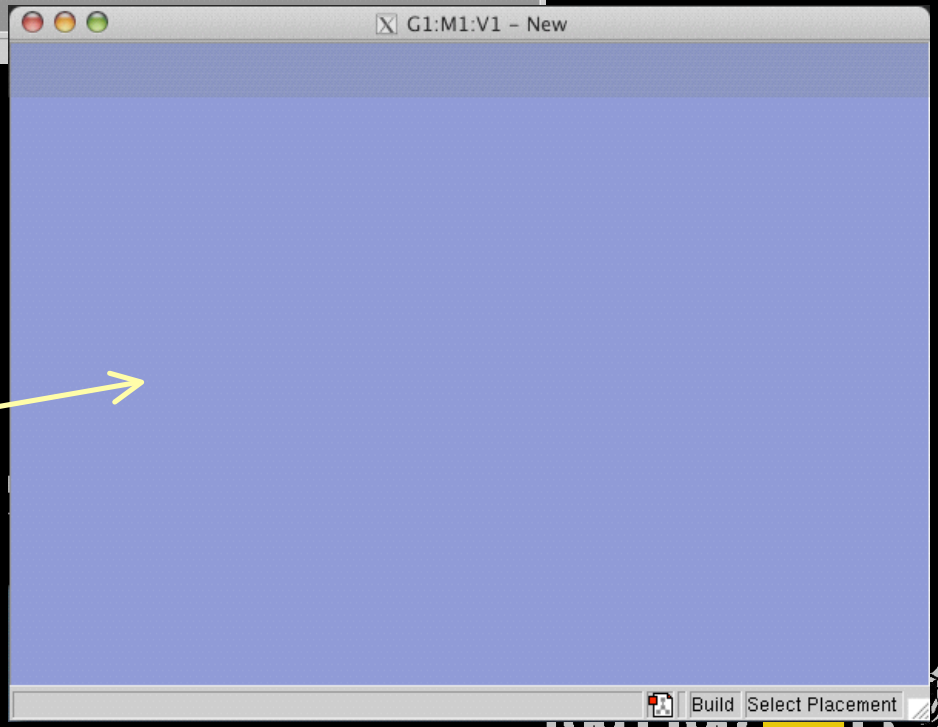
  molden

Building with GaussView:

- Login remotely
  - ssh -X regatta.msi.umn.edu
  - ssh -X sp.msi.umn.edu
  - ssh -X balt.msi.umn.edu
- Load proper module
  - module add g03
  - module add g03/d01
  - etc.
- Launch GaussView
  - gv



Builder

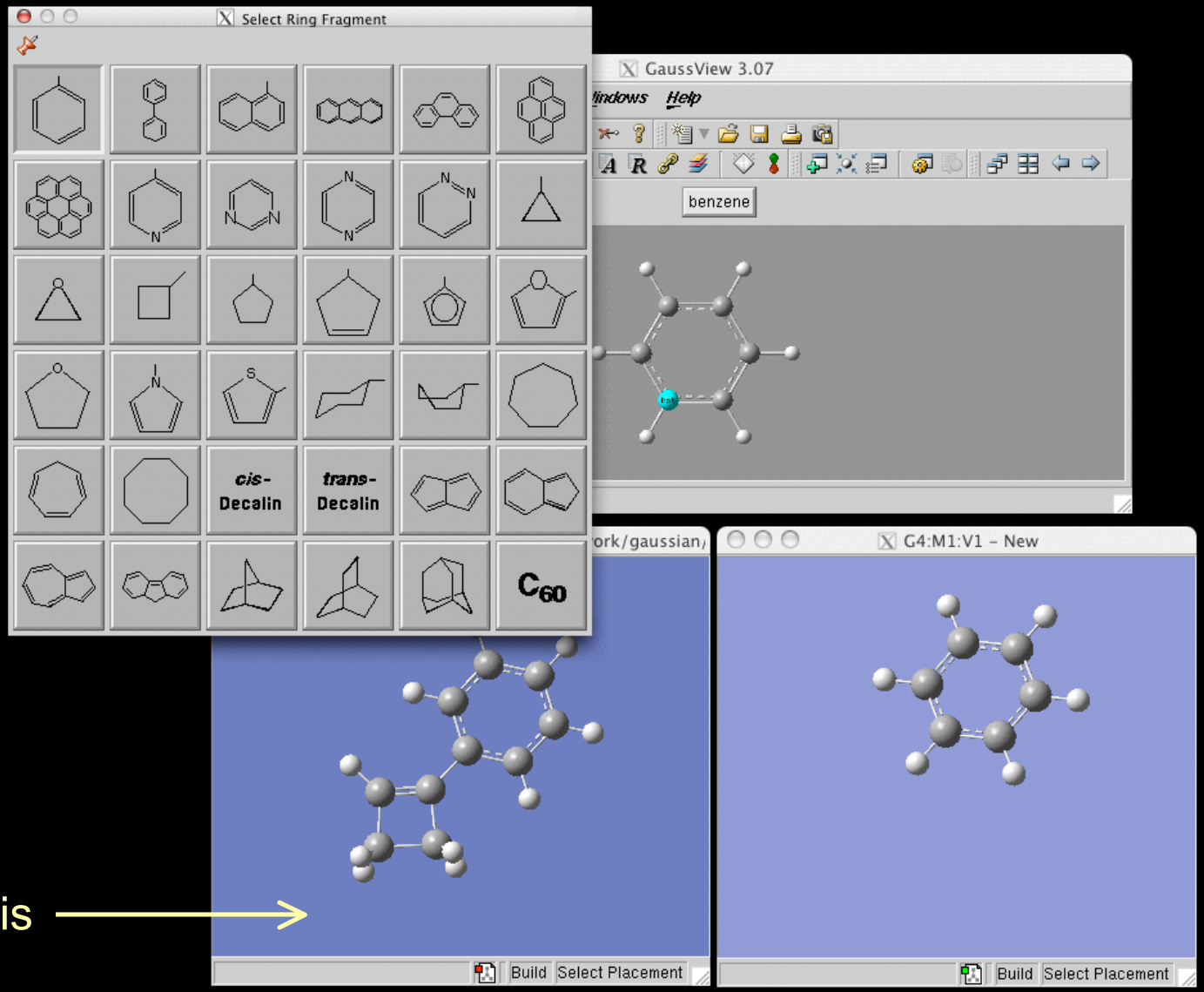


Viewer

Description • **Input** • Submit • Visualize

www.msi.umn.edu

Molecule Building

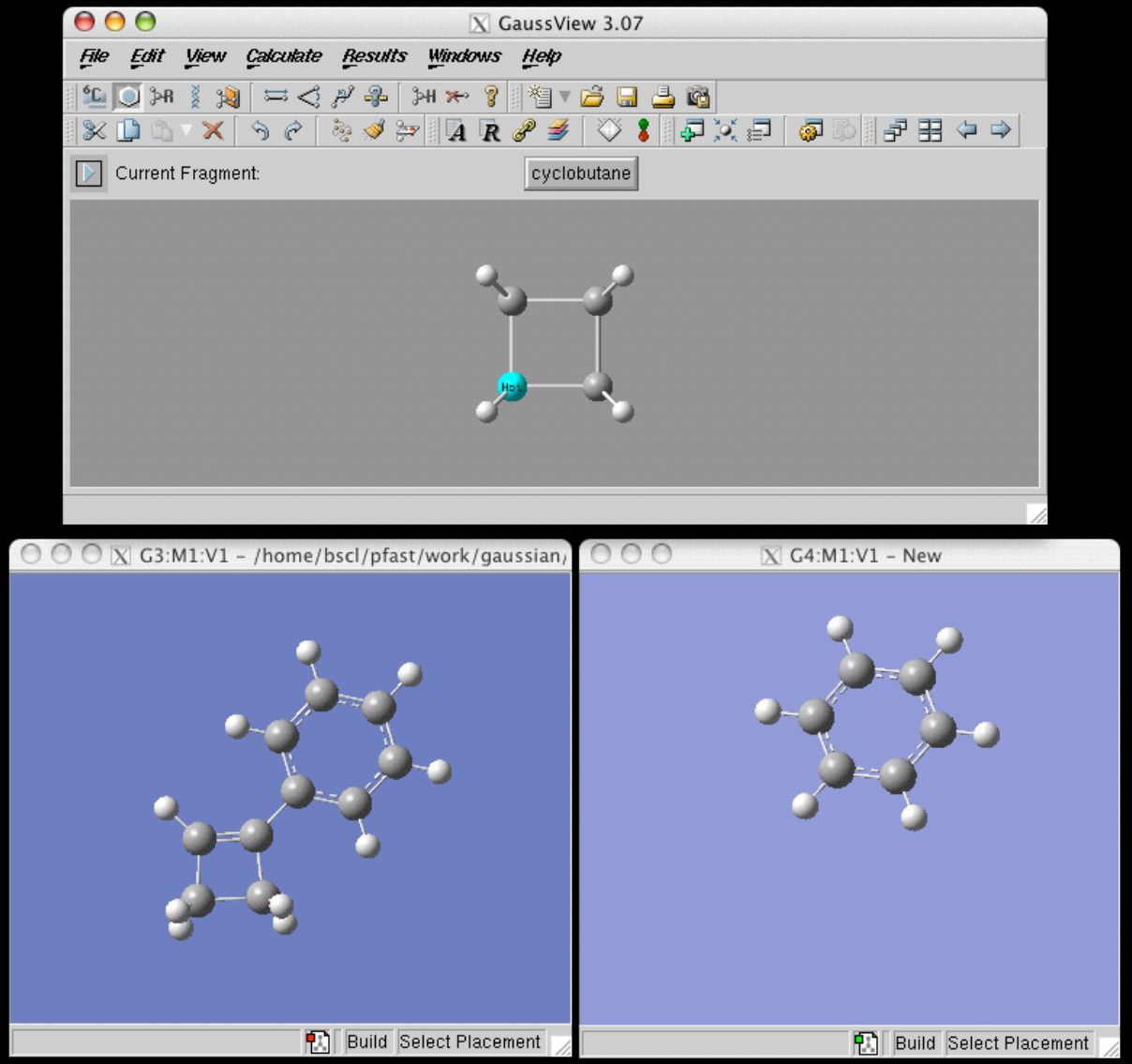


making this →

Description • **Input** • Submit • Visualize

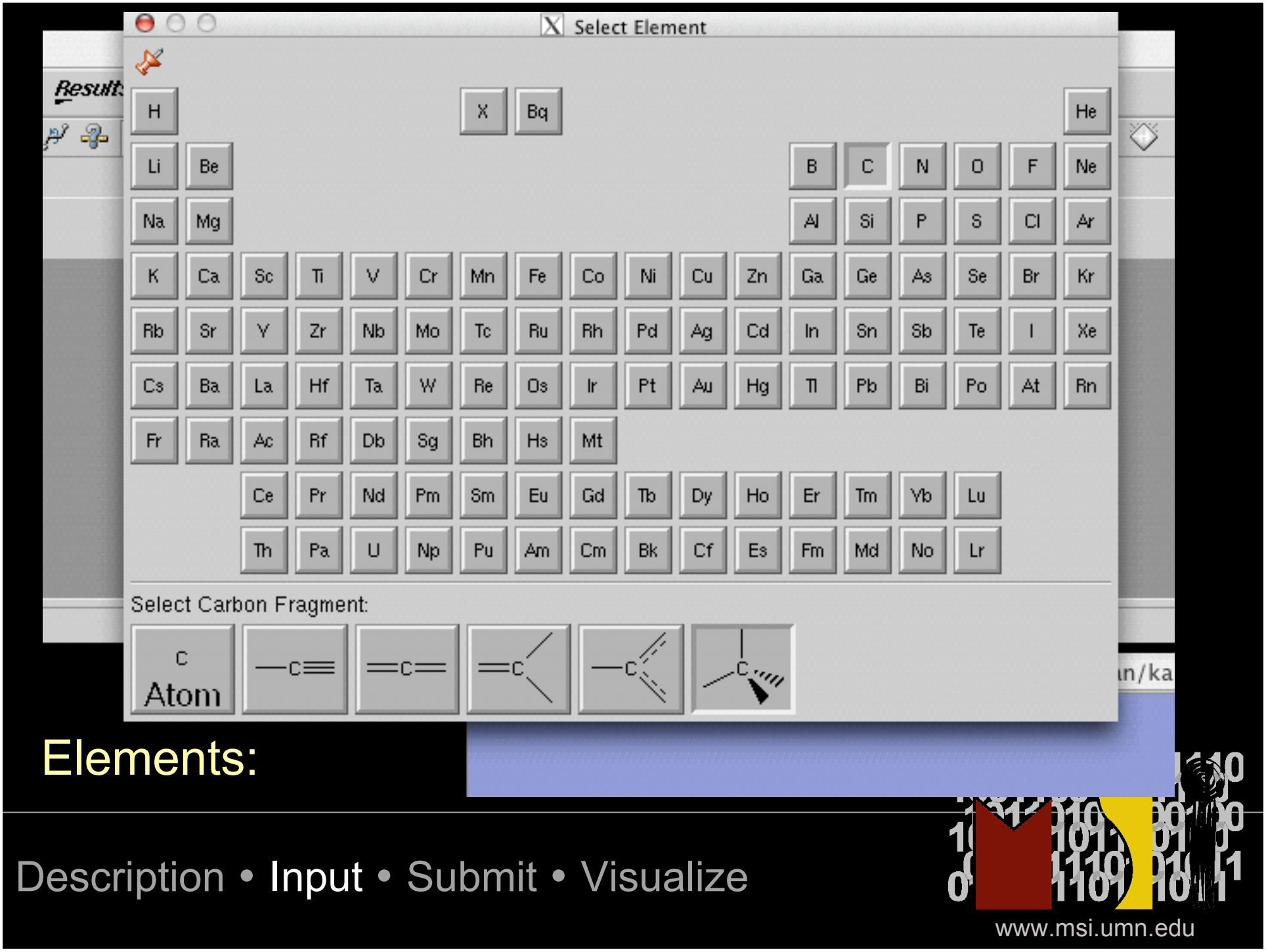
www.msi.umn.edu

Molecule Building



Description • **Input** • Submit • Visualize

www.msi.umn.edu



Elements:

Description • **Input** • Submit • Visualize

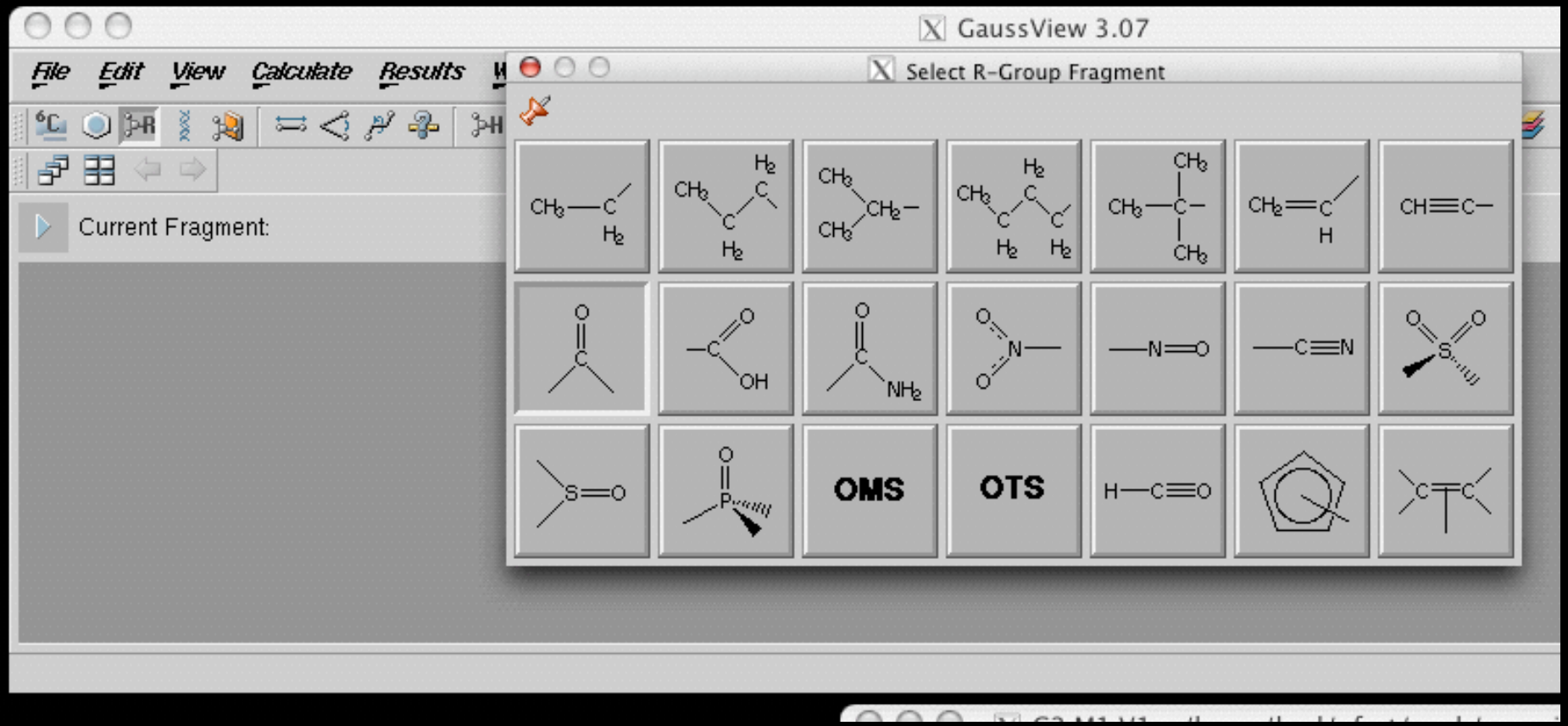
www.msi.umn.edu

Ring Fragments:

Description • Input • Submit • Visualize

www.msi.umn.edu

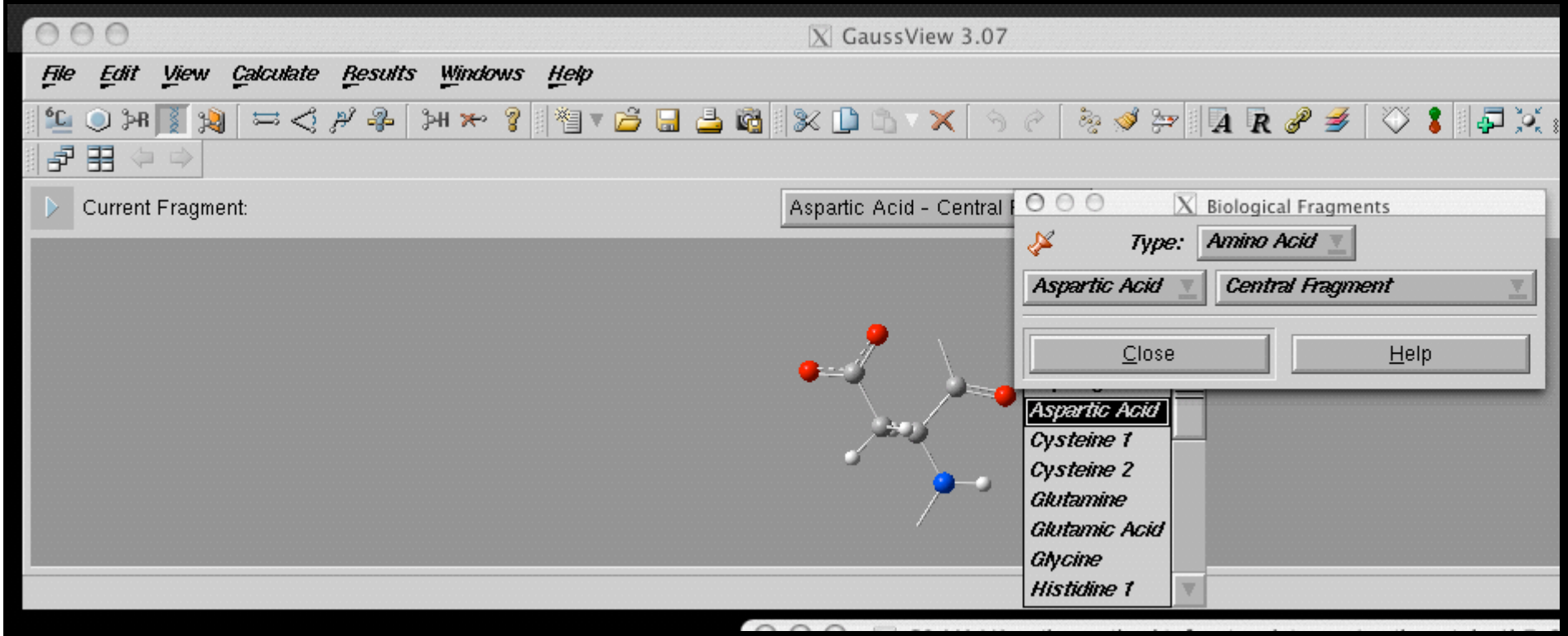
R-Group Fragments:



Description • Input • Submit • Visualize

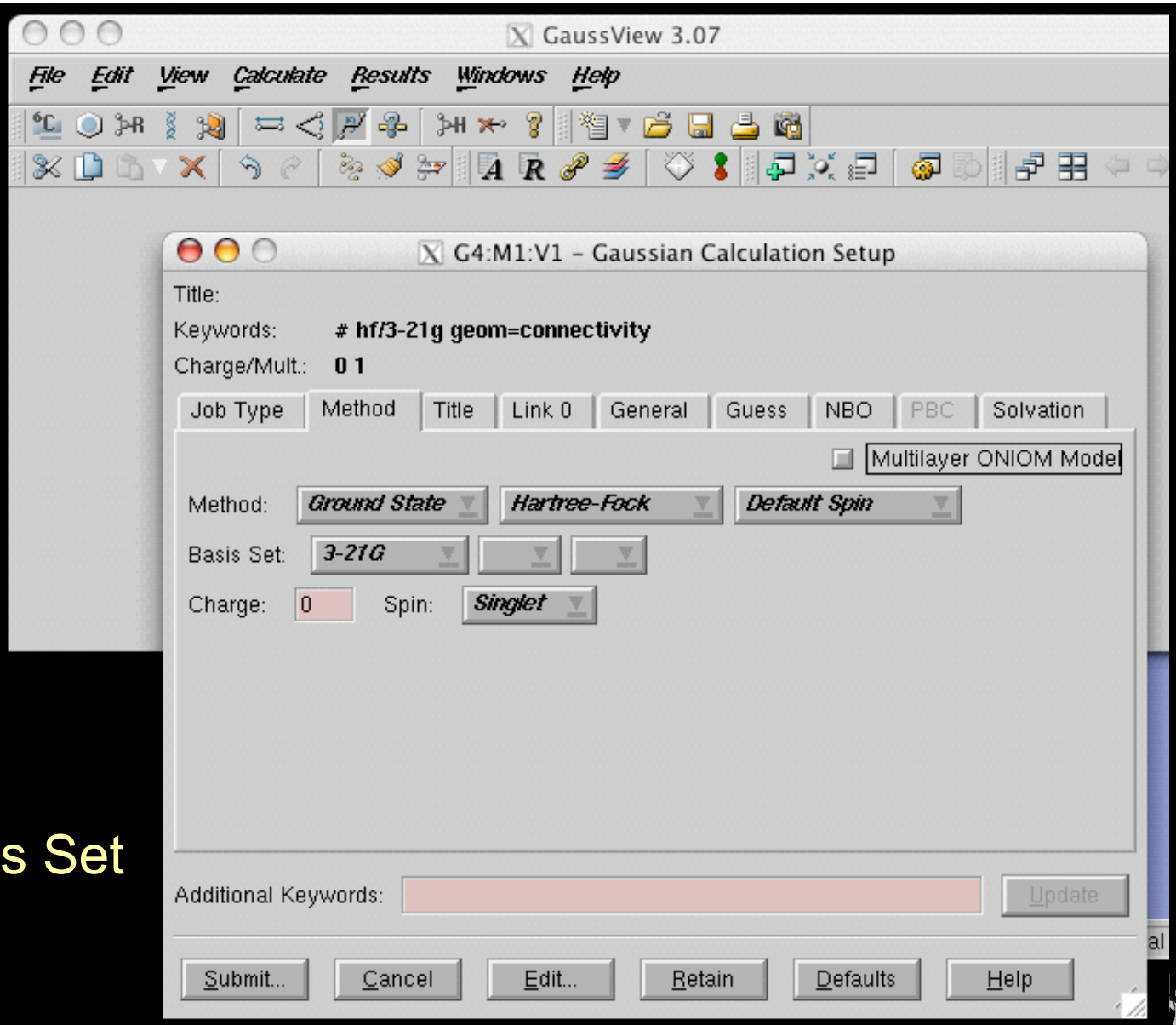
www.msi.umn.edu

Biological Fragments:

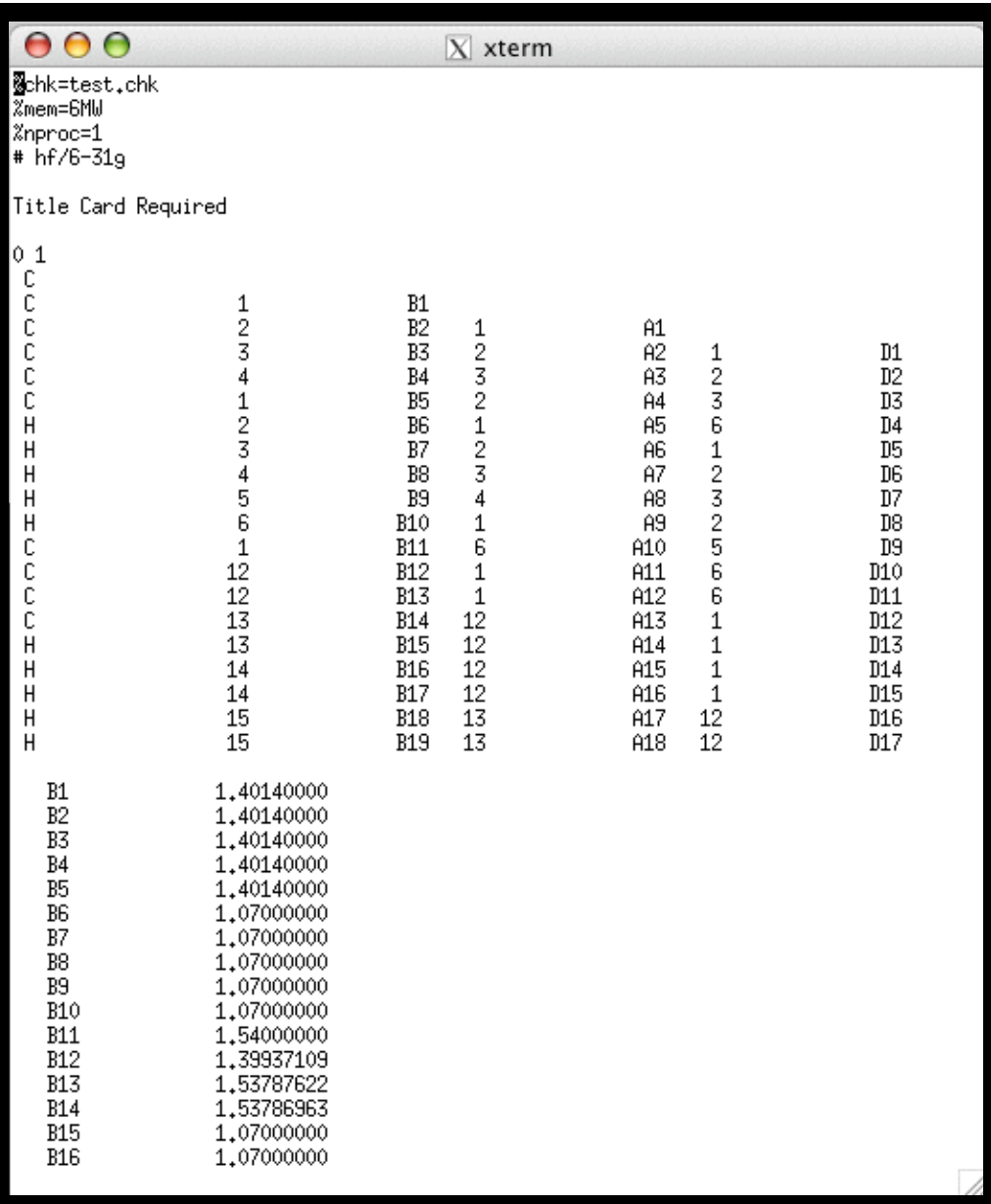


Description • Input • Submit • Visualize

www.msi.umn.edu



Method & Basis Set



Sample
Input File

Submitting Calculations

Submitting your calculation:

Command:

module load <*version*>

Available versions:

g03	latest version available
g03/b05	Gaussian03 Revision B.05
g03/c01	Gaussian03 Revision C.01
g03/d01	Gaussian03 Revision D.01

Systems:

Regatta, Altix, Netfinity, BSCL, SDVL, VWL

Interactive:

Command:

```
g03 < input_file > output_file
```

Notes:

1) input and output files can use absolute or relative pathnames
2) Gaussian temporary files for can be redirected by setting the GAUSS_SCRDIR environment variable.

```
setenv GAUSS_SCRDIR /scratch2/blynch
```

3) BE CAREFUL how long you run interactively

Batch:

Command:

qg03 [-options] input_file

Options:

-h	help, display a usage list and the defaults
-e	email notification
-m	total amount of memory
-p	number of processors
-r	run?
-s	scratch space
-t	amount of wallclock time

Specific queue options

Queue Commands:

Loadleveler:

llsubmit *job.ll*	submit *job.ll*
llq [options]	check the queue
llcancel *jobid*	cancel job in the queue

PBS:

qsub *job.pbs*	submit *job.pbs*
qstat [options]	check the queue
qdel *jobid*	cancel job in the queue

How to View Output

Description • Input • Submit • Visualize

www.msi.umn.edu

less

vi

pico

emacs

Gaussview

Molden

tail

How to View Output

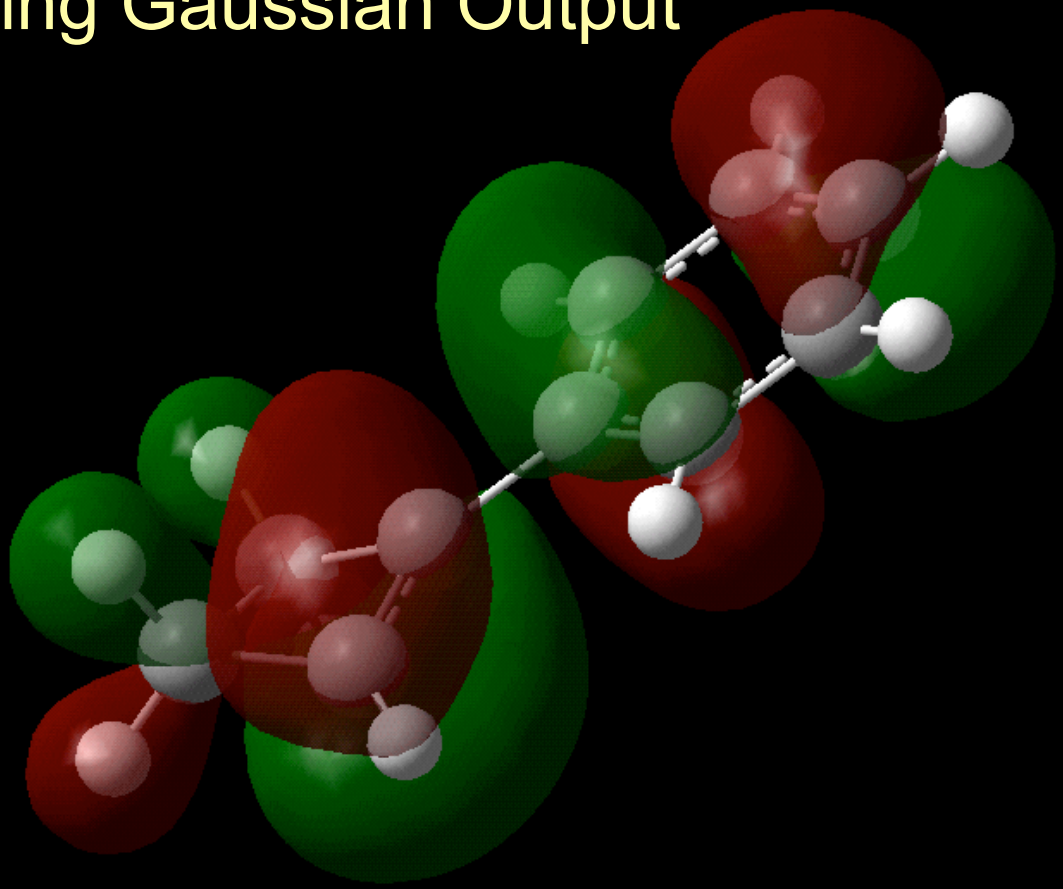
Open the file created called XXXX.out

(where XXXX was the input file name)

It's all text

It gives an outline of the calculation performed

Visualizing Gaussian Output



Description • Input • Submit • Visualize

www.msi.umn.edu

How to View Output Visually

module load g03

gv myoutput.out

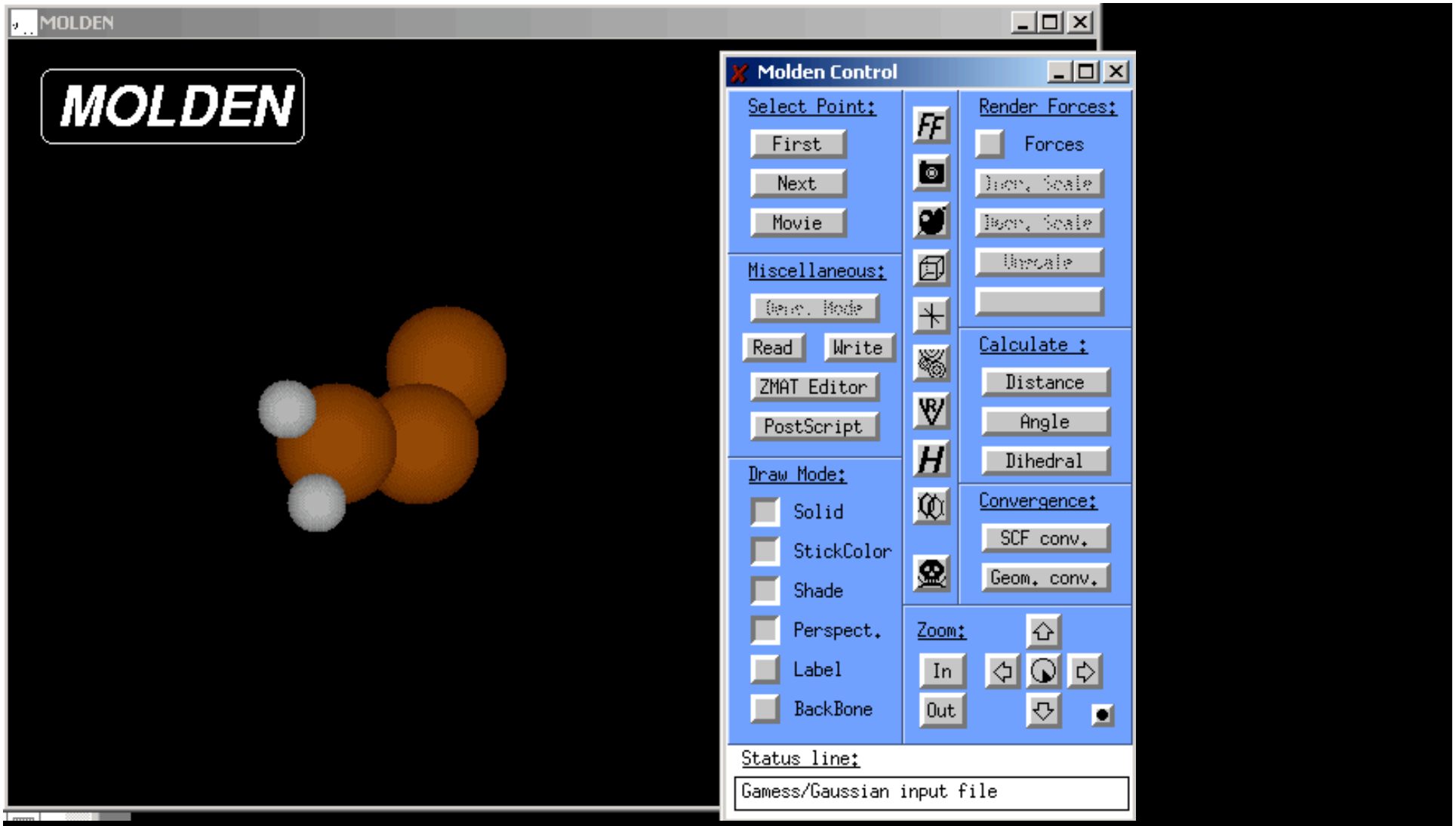
module load molden

molden myoutput.out

Visualization using Molden

Description • Input • Submit • Visualize

www.msi.umn.edu



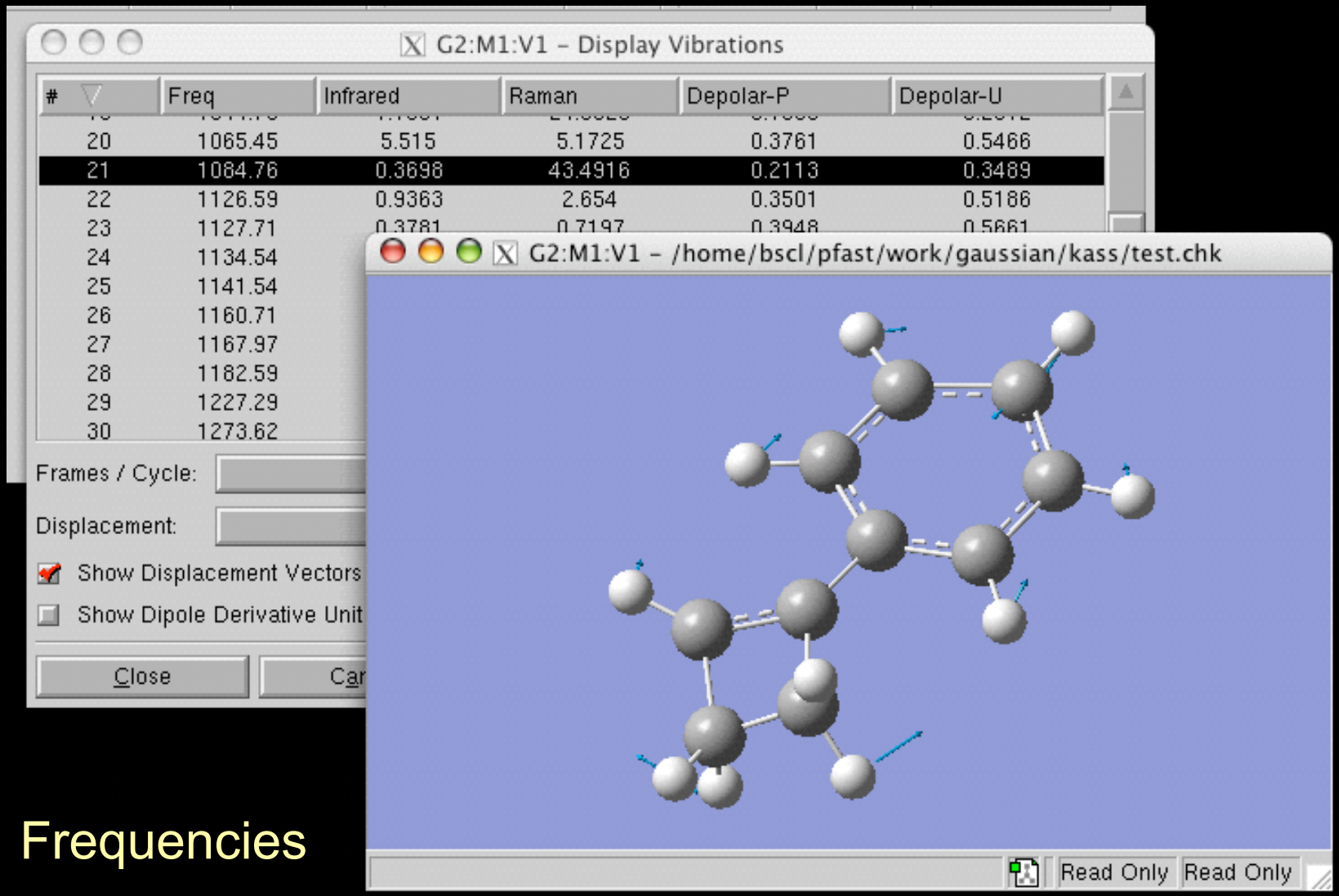
Description • Input • Submit • Visualize

www.msi.umn.edu

Visualization using Gaussview

Description • Input • Submit • Visualize

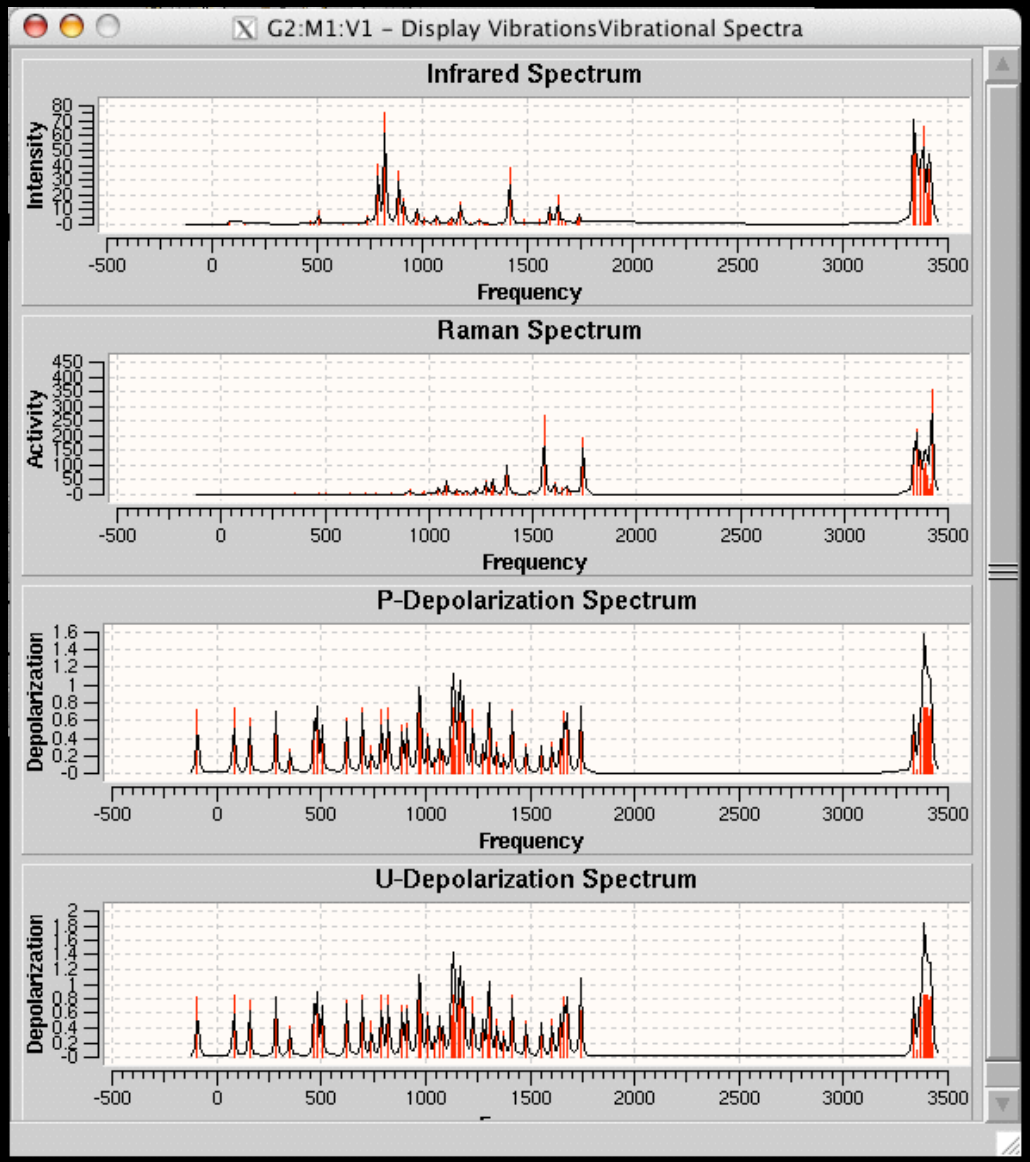
www.msi.umn.edu



Frequencies

Description • Input • Submit • Visualize

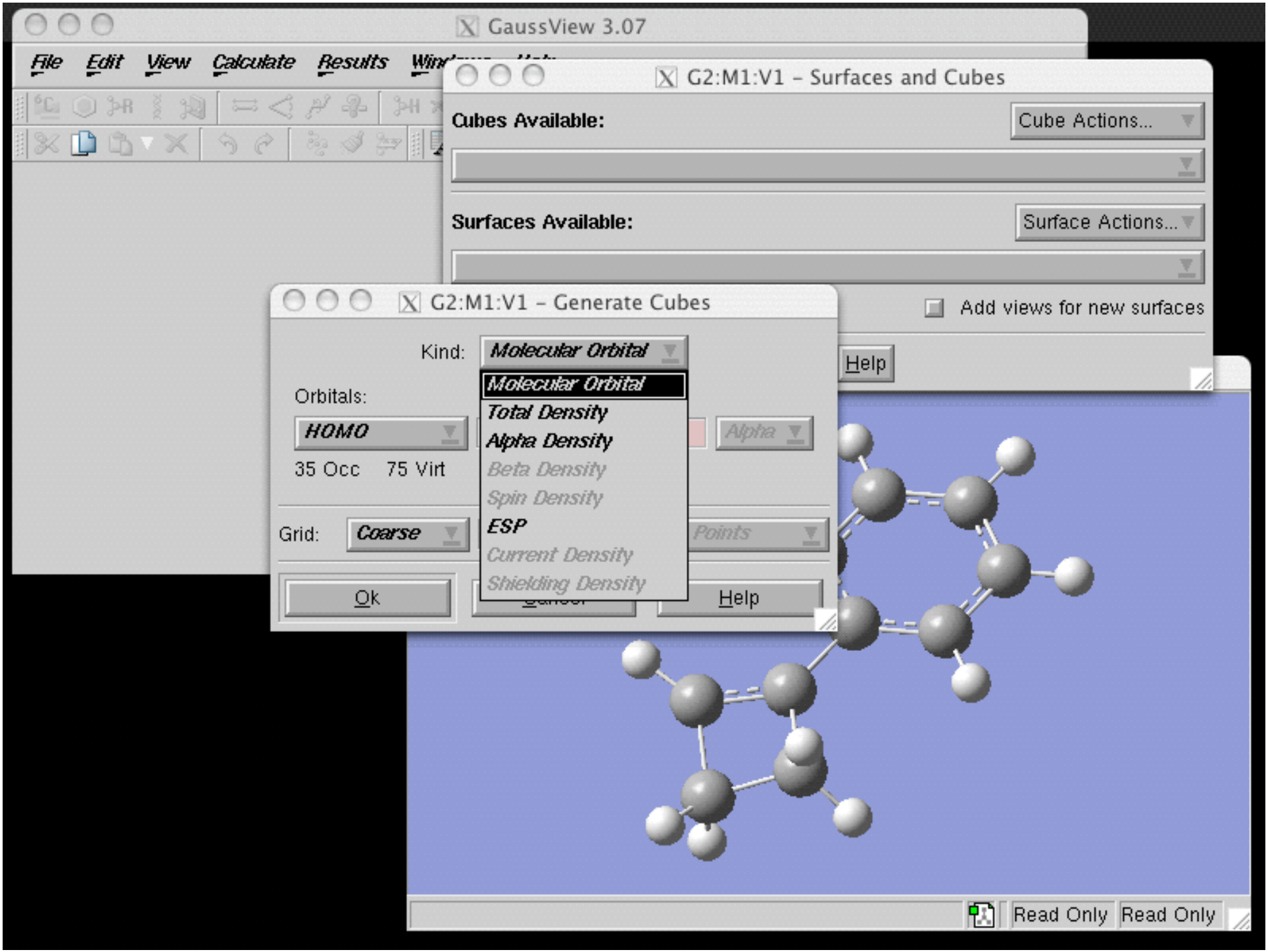
www.msi.umn.edu

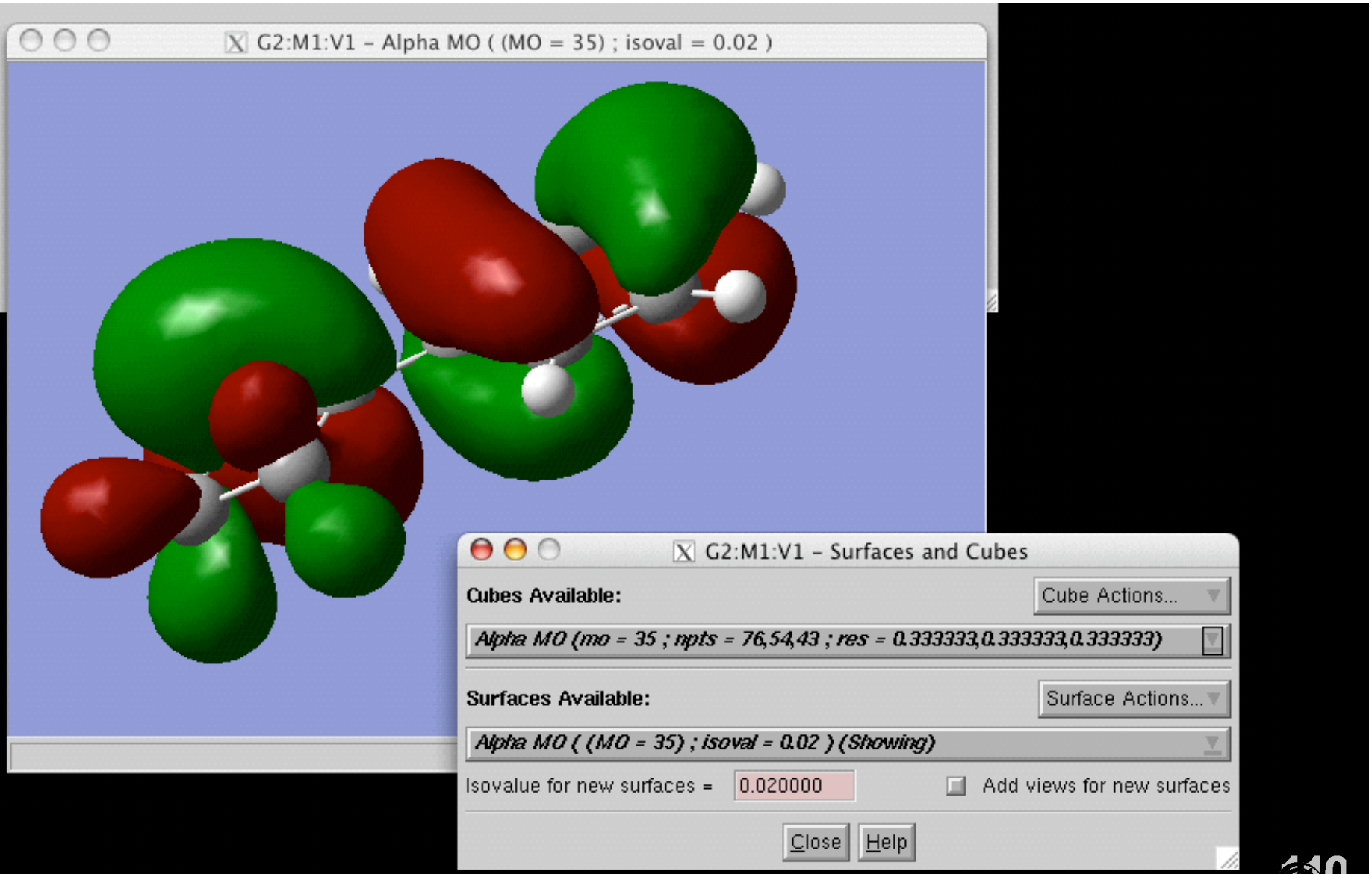


Spectra

Description • Input • Submit • Visualize

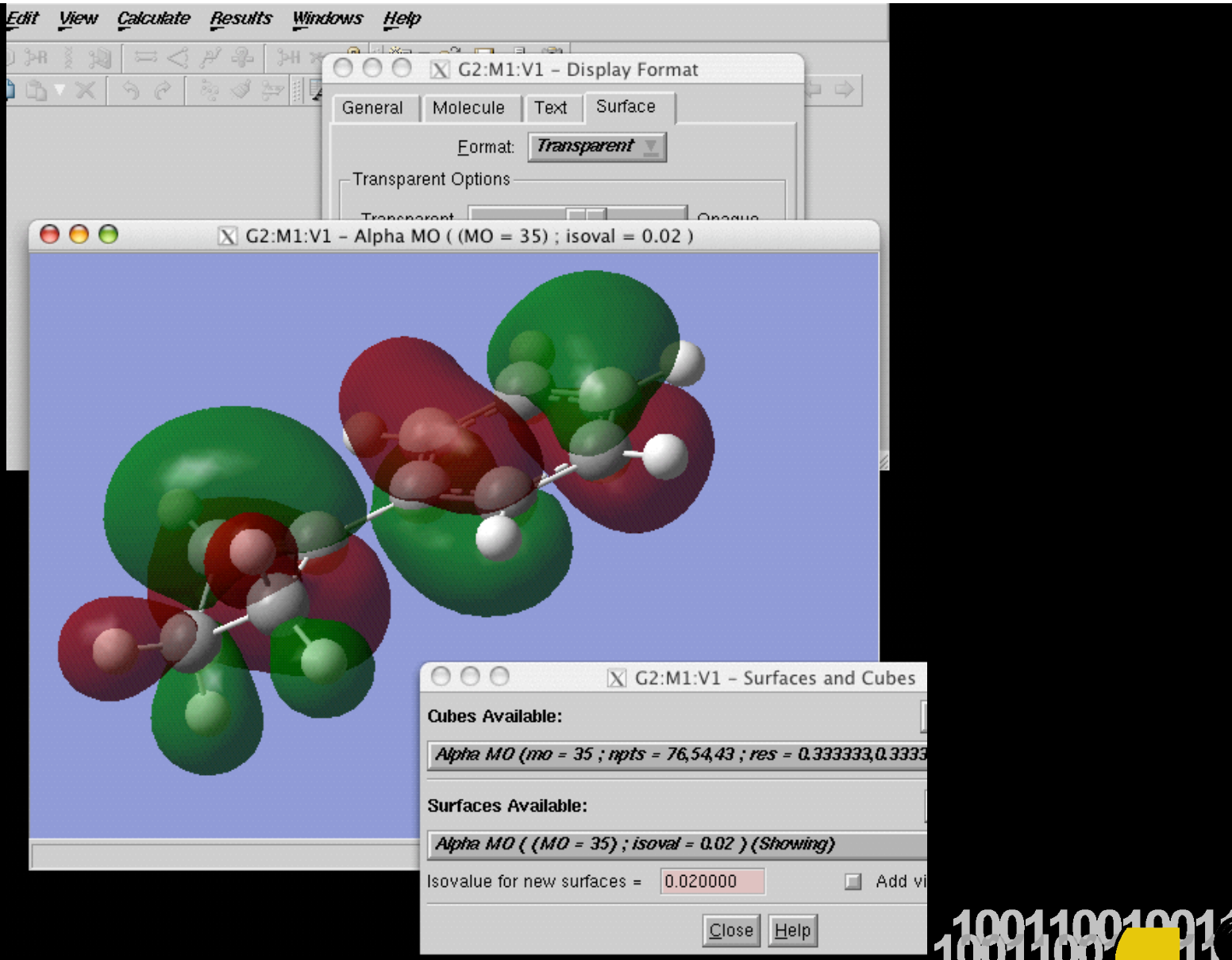
www.msi.umn.edu


GaussView 3.07
File
Edit
View
Calculate
Results
G2:M1:V1 – Surfaces and Cubes
Cubes Available:
Cube Actions...
Surfaces Available:
Surface Actions...
Add views for new surfaces
Help
G2:M1:V1 – Generate Cubes
Kind:
Molecular Orbital
Total Density
Alpha Density
Beta Density
Spin Density
ESP
Current Density
Shielding Density
Orbitals:
HOMO
35 Occ
75 Virt
Alpha
Grid:
Coarse
Points
Ok
Help
Read Only
Read Only



Description • Input • Submit • Visualize

www.msi.umn.edu



Description • Input • Submit • Visualize

www.msi.umn.edu

Common Errors

Description • Input • Submit • Visualize

www.msi.umn.edu

Memory Errors

Not enough memory to run at all: LenERI 3000000

Increase memory by 87346382 words

Error termination via Lnk1e in /usr/local/g03/g03.b01/g03/l906.exe

Solution: Increase %mem

Disk Errors

No file to extend for IUnit=1 –

Out of disk space

Error termination in NtrErr

Solutions: make more space available

check disk quotas

maxdisk=1gb

SCF Failure

What to do?

Possible keywords that may help:

scf=xqc

scf=(maxcyc=500)

scf=…

guess=read (read in a converged guess from a different theory)

Coupled cluster iteration failure

CCSD(maxcyc=200)

Not many options currently available

Re-consider your choice of basis set

Check the orbitals

Re-consider the applicability of CCSD to The system you are studying.

Optimization failures

What to do?

- Restart the calculation from the last geometry and continue
- Look at the structure, adjust it manually, and continue.
- For difficult saddle-point searches: opt=(TS,CalcAll)

Think about how the program works

First, Gaussian needs to read your input file (does file exist?)

Make sense of geometry (do atoms overlap?)

Make sense of keywords (do keywords define an ambiguous calculation?)

Make sense of electronic state (do charge and multiplicity make sense?)

Solve HF equations, converge CC iterations, converge geometry, calculate frequencies, …

Questions?

email: blynch@msi.umn.edu
help@msi.umn.edu

phone: 612-624-4122
612-626-0802 (helpline)