A *practical* Seminar on MNDO

Zhenggang Lan
Oliver Weingart

April 2011

A *practical* Seminar on MNDO

Compiling the newest version

Basic operations: Optimisation, Frequencies

Molecular dynamics in the ground state

Excited state properties

Photodynamics

A *practical* Seminar on MNDO

MNDO the program

Collection of semi-empirical methods

MNDO the method

Modified neglect of differential overlap

Compiling the newest version

Get a CVS Account (ask Axel)

CVS → Version Control System

Enables multiple developers to work on common software projects

Allows to keep track on changes, restore old versions, merge new versions etc.

In your .cshrc :

```
setenv CVSEDITOR '/usr/bin/vi'
setenv CVSROOT ':pserver:yourusername@op22th1:/ns80th/nas/users/cvs/cvsroot'
setenv CVS_RSH ssh
```

Compiling the newest version

Login to CVS account

```
cvs login
```

Download newest version

```
cvs checkout -d ./myfirstmndo mndo99
```

Further reading on CVS

Open Source Development with CVS, 3rd Edition

by Karl Fogel and Moshe Bar

http://cvsbook.red-bean.com

Compiling the newest version

Make sure the Intel compiler (ifort) is in your path

```
source /opt/intel/Compiler/11.1/075/bin/ifortvars.csh intel64
```

MKL libraries should already be setup correctly

```
source /opt/intel/mkl/10.2.7.041/tools/environment/mklvarsem64t.csh
```

Configure MNDO and compile

```
cd myfirstmndo

aclocal;automake;autoconf
```

← prepare necessary make files
(don't forget, otherwise no MNDO!)

```
./configure

make -j6

./configure --help
```

← for enhanced options

Compiling the newest version

Test mndo

```
make test
```

```
Final results of test jobs:


Suite                      | Tests   Passed  Failed
-------------------------------------------------
Standard - formatted       |   50      50       0
Standard - keywords        |   50      50       0
MOPAC6 Compatibility       |    8       8       0
Analytic Derivatives       |   36      36       0
GUGA-CI Module             |  238     238       0
Surface Crossing/Hopping   |   72      69       3
HDLC Optimizer             |   20      12       4
Full-matrix DMS            |   60      52       2
Direct sparse DMS          |   52      29       3
Smooth COSMO solv. model   |    1       0       1
Semiemp. DFTB module       |    6       5       1
Valence bond module        |    5       5       0
-------------------------------------------------
```

Compiling the newest version

Documentation / Help

`mndo99.txt`	Explanation of Keywords
`mndo99.doc`	Strategies and Literature
`./mdtools/mdtools.doc`	Tutorial for Excited States MD with example files

Basic Operations

Structure of the input

Command section, continued in next line with "+"

Two comment lines

Geometry input cartesian or internal coordinates Mopac format, Atom numbers, not names!

Addit. options (orbitals, Configurations, param.)

```
iop=-6 jop=0 igeom=1 iform=1 +
nsav13=2 nsav7=4 +
icuts=-1 icutg=-1 iscf=11 +
iplscf=11 dstep=0.00001 +
kitscf=200 +
iprint=1 mprint=1 iprec=100 +
imult=1 ioutci=1 +
kci=5  imomap=1 maxrtl=200 +
movo=1 ici1=6 ici2=3 nciref=3 +
mciref=0 levexc=2 iroot=8 iuvcd=2

OM2
  6    -0.0038158628 1     0.0245291050 1     0.0002045116 1
  6     1.4444455762 1    -0.0028495847 1     0.0025579728 1
  6     2.1003818970 1    -1.2785586395 1     0.0015963221 1
  7     1.5244423372 1    -2.5151010722 1    -0.0013462915 1
  6     0.1907596134 1    -2.4583656889 1    -0.0034641595 1
  7    -0.5573002077 1    -1.2909838270 1    -0.0028239832 1
  7     2.3827054852 1     0.9939920225 1     0.0057199715 1
  6     3.5661060171 1     0.3955068888 1     0.0067161496 1
  7     3.4418138951 1    -1.0020073821 1     0.0042534315 1
  7    -0.5067042104 1    -3.6378509876 1    -0.0065229881 1
  8    -0.7914367066 1     0.9772658979 1     0.0004289937 1
  1     4.5315690746 1     0.8994845534 1     0.0091391637 1
  1     4.1858120431 1    -1.6820551368 1     0.0043893679 1
  1    -1.5706248963 1    -1.3493085940 1    -0.0045904916 1
  1    -1.5000816208 1    -3.6759518635 1    -0.0082346686 1
  1    -0.0118748942 1    -4.5009375650 1    -0.0071392252 1
  0     0.0000000000 0     0.0000000000 0     0.0000000000 0
 24 25  26 27 28 29 30 31 32
```

NO PLUS in last line !

← Closing with zero line

Basic Operations

Two *very* simple MNDO inputs:

```
iform=1
MNDO

1   0.000000 0   0.000000 0   0.000000 0    0  0  0
1   1.008000 1   0.000000 0   0.000000 0    1  0  0
0   0.000000 0   0.000000 0   0.000000 0    0  0  0
```

Optimize (1) or fix ccordinate (0)

```
iform=1 igeom=1
MNDO

1   0.000000 1   0.000000 1   0.000000 1
1   1.008000 1   0.000000 1   0.000000 1
0   0.000000 0   0.000000 0   0.000000 0
```

Basic Operations

Important keywords

geometry input **iform** → **=0 formatted**
=1 unformatted
(recommended)

igeom → **=0 internal**
=1 cartesian

Method **iop** →

=-10 MNDO/d
= -8 OM3
= -7 PM3
= -6 OM2
=- 5 OM1
=- 2 AM1
=- 1 MNDOC
= 0 MNDO
= 1 MINDO/3
= 2 CNDO/2
= 5 SCC-DFTB
= 6 SCC-DFTB with Jorgensens param.
=- 3 MNDO/H hydrogen bonds.
=- 4 MNDO with non-standard param.

Basic Operations

Important keywords

type of job **jop** →

```
= -2   gradient
= -1   single point
= 0/3  opt minimum/freq
= 1/4  transition state/freq
```

Output options

```
nsav7  → = 4  write restart file
nsav13 → = 2  write MOLDEN file
```

Basic Operations

A simple optimization

```
iform=1 iop=-6 jop=3 nsav7=4 nsav13=2

OM2
6     0.000000 0    0.000000 0    0.000000 0       0    0    0
6     1.335000 1    0.000000 0    0.000000 0       1    0    0
1     1.089000 1  120.000000 1    0.000000 0       2    1    0
1     1.089000 1  120.000000 1  180.000000 1       1    2    3
1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6     1.450000 1  120.000000 1  180.000000 1       2    1    5
6     1.335000 1  120.000000 1  180.000000 1       6    2    1
1     1.089000 1  120.000000 1    0.000000 1       7    6    2
1     1.089000 1  120.000000 1  180.000000 1       7    6    2
1     1.089000 1  120.000000 1    0.000000 1       6    7    9
0     0.000000 0    0.000000 0    0.000000 0       0    0    0
```

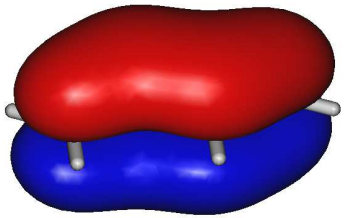
Butadien optimization+frequencies

`mndo99 < buta.inp > buta.log` → `molden.dat, fort.7`

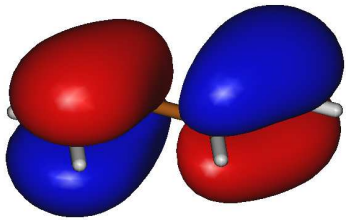
`qmndo99 buta.inp` → `buta.inp.molden, buta.inp.restart`

Basic Operations

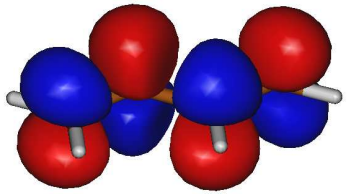
A simple optimization - Butadien



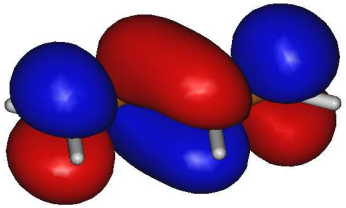
9



11



12



14

HOMO=Orbital 11, but:

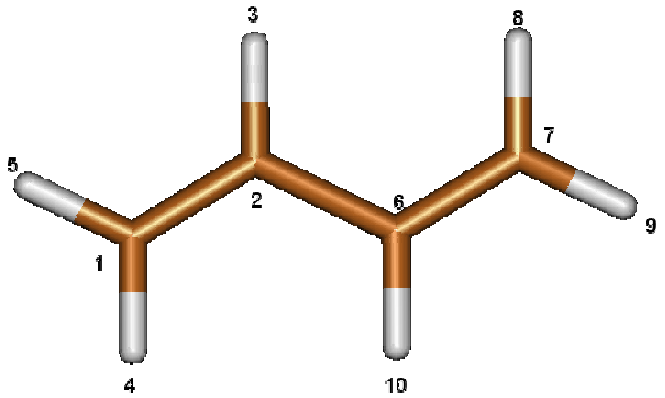
4*6 + 6*1 = 30 → HOMO should be 15!

MNDO does not consider the core orbitals:

C: 2s 2px 2py 2pz

H: 1s

→ *careful, especially with active space selection!!*



Basic Operations

A simple optimization - Butadien

PES Scan along central bond:

```
iform=1 iop=-6 jop=0 nsav7=4 nsav13=2 kgeom=1

OM2
6     0.000000 0    0.000000 0    0.000000 0       0    0    0
6     1.335000 1    0.000000 0    0.000000 0       1    0    0
1     1.089000 1  120.000000 1    0.000000 0       2    1    0
1     1.089000 1  120.000000 1  180.000000 1       1    2    3
1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6     1.450000 1  120.000000 1  180.000000 1       2    1    5
6     1.335000 1  120.000000 1  180.000000 1       6    2    1
1     1.089000 1  120.000000 1    0.000000 1       7    6    2
1     1.089000 1  120.000000 1  180.000000 1       7    6    2
1     1.089000 1  120.000000 1    0.000000 1       6    7    9
0     0.000000 0    0.000000 0    0.000000 0       0    0    0
7    3   24   -7.5
```

Basic Operations

A simple optimization - Butadien

PES Scan along central bond:

```
iform=1 iop=-6 jop=0 nsav7=4 nsav13=2 kgeom=1

OM2
6     0.000000 0    0.000000 0    0.000000 0       0    0    0
6     1.335000 1    0.000000 0    0.000000 0       1    0    0
1     1.089000 1  120.000000 1    0.000000 0       2    1    0
1     1.089000 1  120.000000 1  180.000000 1       1    2    3
1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6     1.450000 1  120.000000 1  180.000000 1       2    1    5
6     1.335000 1  120.000000 1  180.000000 1       6    2    1
1     1.089000 1  120.000000 1    0.000000 1       7    6    2
1     1.089000 1  120.000000 1  180.000000 1       7    6    2
1     1.089000 1  120.000000 1    0.000000 1       6    7    9
0     0.000000 0    0.000000 0    0.000000 0       0    0    0
7    3   24   -7.5
```

Coordinate holding the constraint

Basic Operations

A simple optimization - Butadien

PES Scan along central bond:

```
     iform=1 iop=-6 jop=0 nsav7=4 nsav13=2 kgeom=1

     OM2
1.   6     0.000000 0    0.000000 0    0.000000 0       0    0    0
2.   6     1.335000 1    0.000000 0    0.000000 0       1    0    0
3.   1     1.089000 1  120.000000 1    0.000000 0       2    1    0
4.   1     1.089000 1  120.000000 1  180.000000 1       1    2    3
5.   1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6.   6     1.450000 1  120.000000 1  180.000000 1       2    1    5
7.   6     1.335000 1  120.000000 1  180.000000 0       6    2    1
     1     1.089000 1  120.000000 1    0.000000 1       7    6    2
     1     1.089000 1  120.000000 1  180.000000 1       7    6    2
     1     1.089000 1  120.000000 1    0.000000 1       6    7    9
     0     0.000000 0    0.000000 0    0.000000 0       0    0    0
     7    3   24   -7.5
```

Type (1=distance,2=angle,3=dihedral)

Coordinate holding the constraint

Basic Operations

A simple optimization - Butadien

PES Scan along central bond:

```
iform=1 iop=-6 jop=0 nsav7=4 nsav13=2 kgeom=1

OM2
6     0.000000 0    0.000000 0    0.000000 0       0    0    0
6     1.335000 1    0.000000 0    0.000000 0       1    0    0
1     1.089000 1  120.000000 1    0.000000 0       2    1    0
1     1.089000 1  120.000000 1  180.000000 1       1    2    3
1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6     1.450000 1  120.000000 1  180.000000 1       2    1    5
6     1.335000 1  120.000000 1  180.000000 0       6    2    1
1     1.089000 1  120.000000 1    0.000000 1       7    6    2
1     1.089000 1  120.000000 1  180.000000 1       7    6    2
1     1.089000 1  120.000000 1    0.000000 1       6    7    9
0     0.000000 0    0.000000 0    0.000000 0       0    0    0
7    3   24   -7.5
```

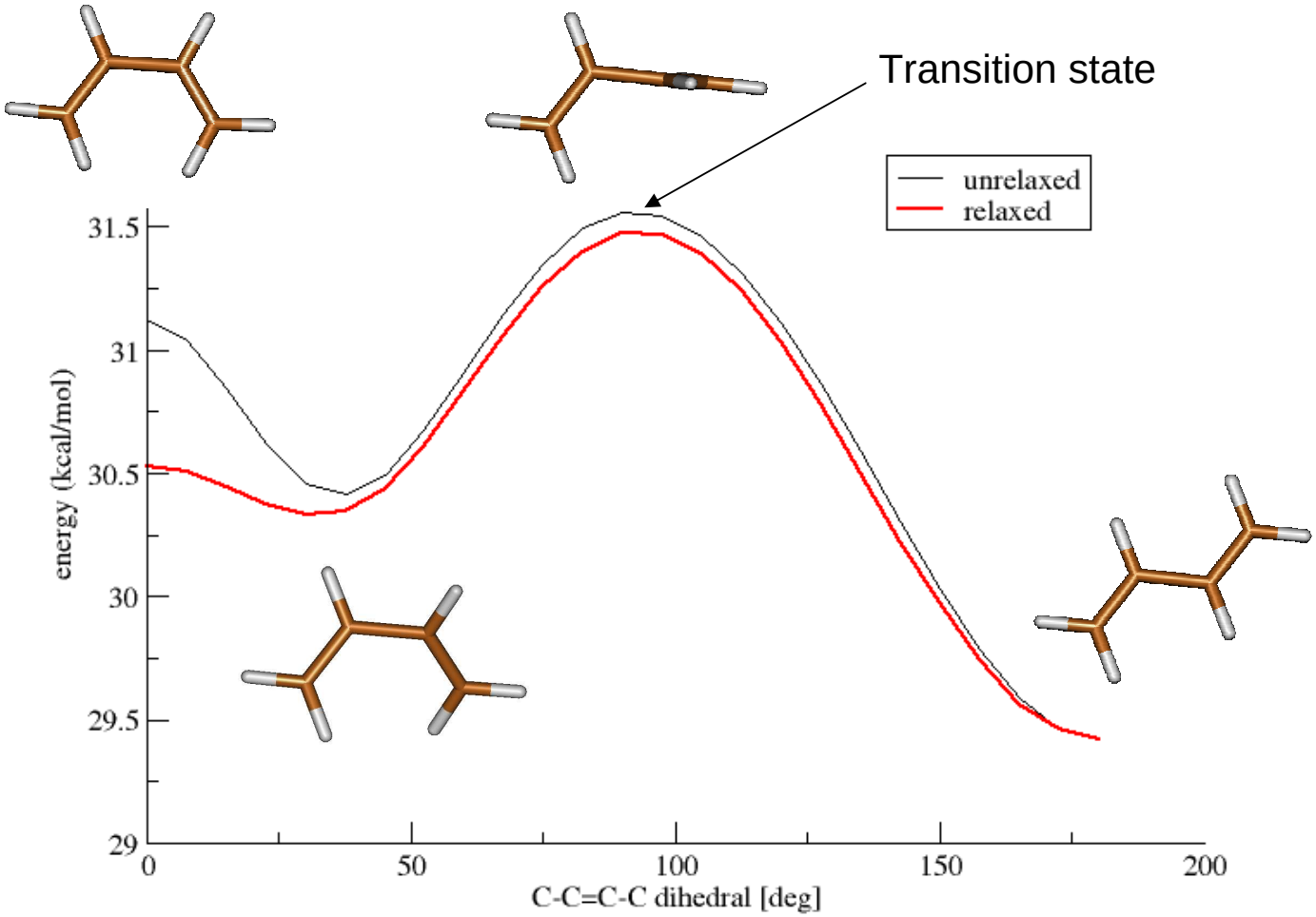
Step size

Number of steps

Basic Operations

	COORDINATE	FORMATION (KCAL/MOL)	MOMENT (DEBYE)	GRADIENT NORM	GRADIENT NORM
1	180.006	29.42046	0.001	0.256	0.441
2	172.506	29.46127	0.001	0.862	1.136
3	165.006	29.56477	0.006	0.413	2.430
4	157.506	29.74056	0.009	1.040	2.973
5	150.006	29.97242	0.026	2.076	4.185
6	142.506	30.21751	0.001	0.378	4.063
7	135.006	30.49772	0.019	0.671	4.235
8	127.506	30.77690	0.045	1.195	4.162
9	120.006	31.02912	0.081	0.164	3.544
10	112.506	31.23987	0.119	0.044	2.872
11	105.006	31.39033	0.156	0.075	1.908
12	97.506	31.46920	0.189	0.083	0.694
13	90.006	31.47539	0.215	0.062	0.595
14	82.506	31.39956	0.233	0.076	1.940
15	75.006	31.25342	0.244	0.126	3.124
16	67.506	31.05351	0.248	0.168	3.918
17	60.006	30.82594	0.248	0.204	4.134
18	52.506	30.60815	0.245	0.213	3.812
19	45.006	30.43881	0.239	0.211	2.613
20	37.506	30.34520	0.234	0.076	1.050
21	30.006	30.32967	0.229	0.127	0.496
22	22.506	30.37512	0.225	0.156	1.383
23	15.006	30.44550	0.223	0.160	1.529
24	7.506	30.50516	0.222	0.149	0.987
25	0.006	30.52819	0.222	0.155	0.267

Basic Operations

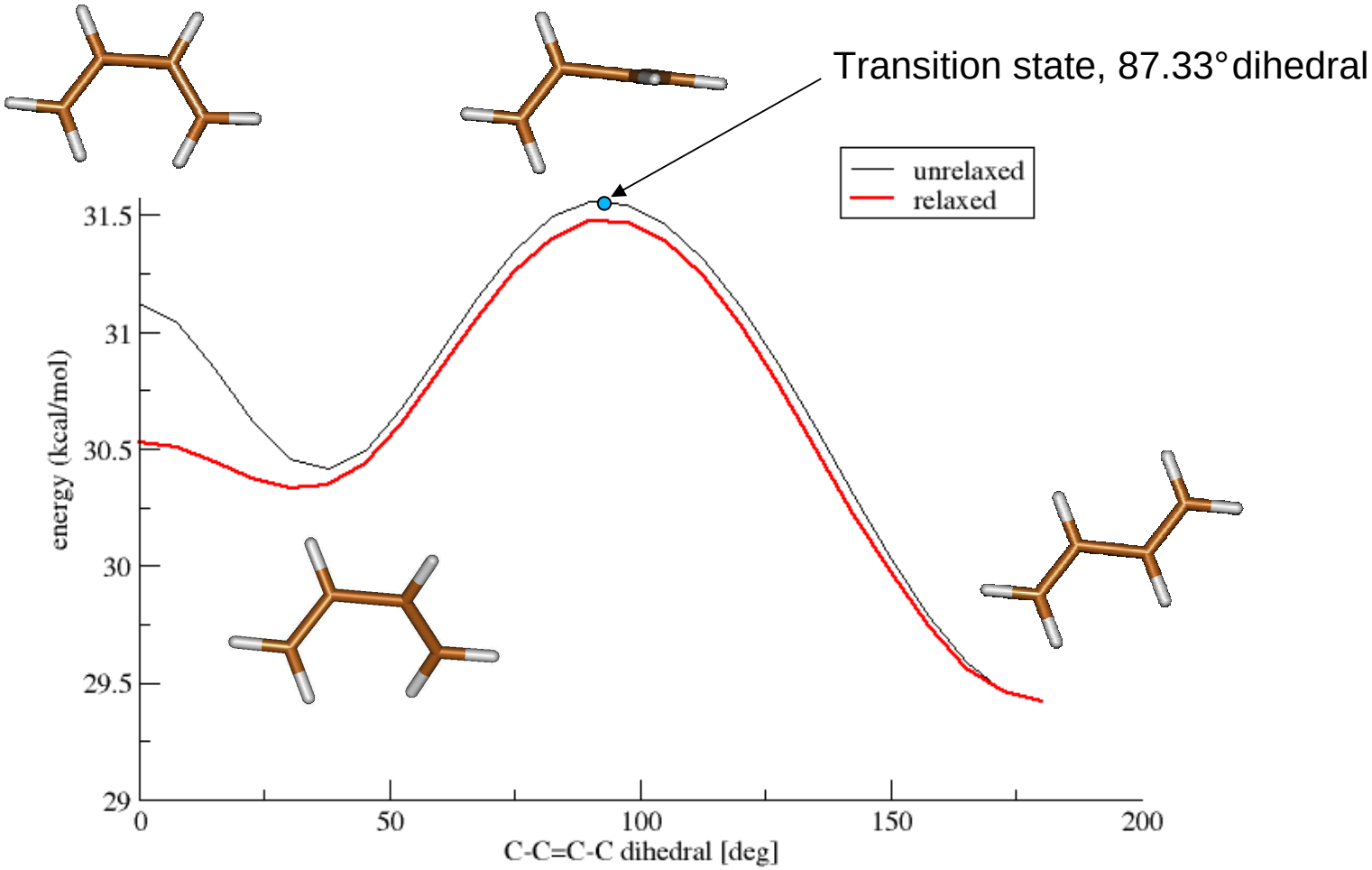


Basic Operations

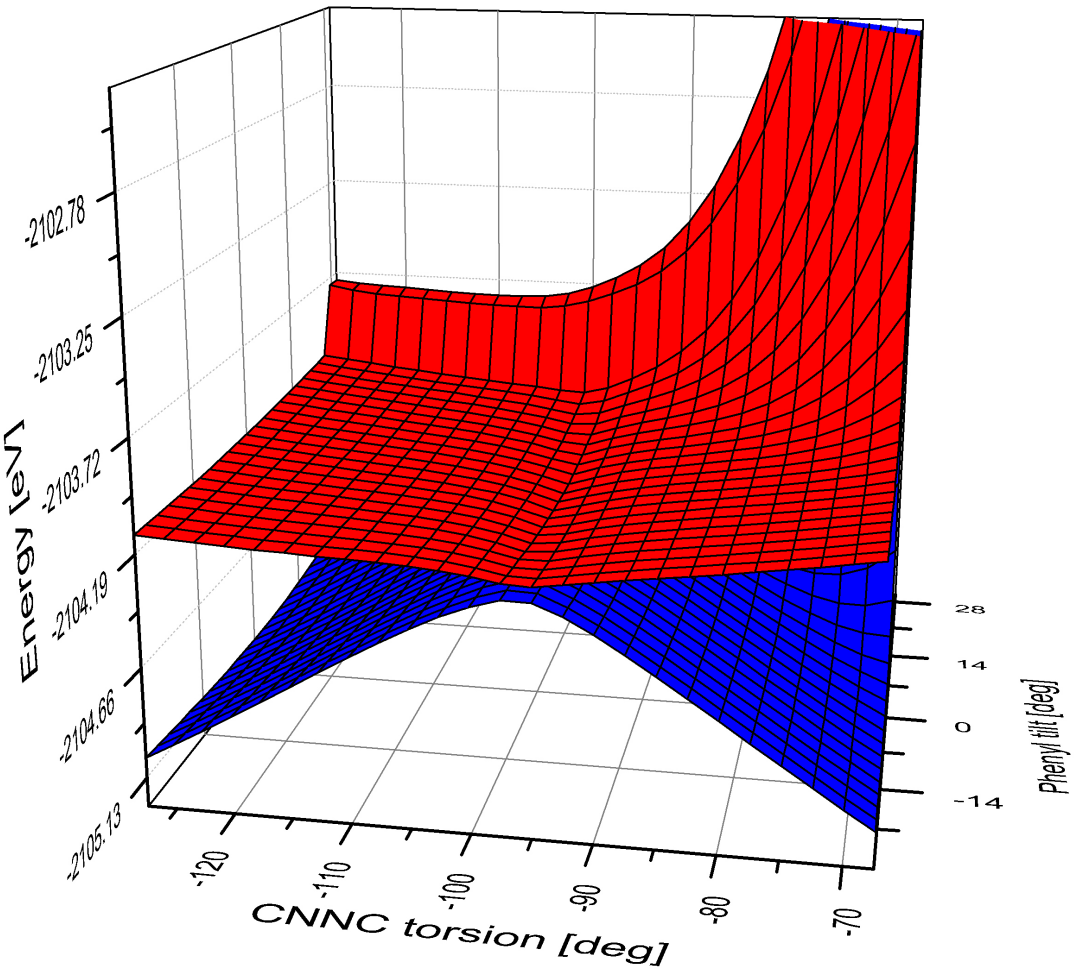
```
iform=1 iop=-6 jop=4 nsav7=4 nsav13=2

OM2
6     0.000000 0    0.000000 0    0.000000 0       0    0    0
6     1.335000 1    0.000000 0    0.000000 0       1    0    0
1     1.089000 1  120.000000 1    0.000000 0       2    1    0
1     1.089000 1  120.000000 1  180.000000 1       1    2    3
1     1.089000 1  120.000000 1    0.000000 1       1    2    3
6     1.450000 1  120.000000 1  180.000000 1       2    1    5
6     1.335000 1  120.000000 1   90.000000 1       6    2    1
1     1.089000 1  120.000000 1    0.000000 1       7    6    2
1     1.089000 1  120.000000 1  180.000000 1       7    6    2
1     1.089000 1  120.000000 1    0.000000 1       6    7    9
0     0.000000 0    0.000000 0    0.000000 0       0    0    0
```

TS, freq= -137 cm^{-1}

Basic Operations



Basic Operations



Azobenzene
3D Surface topology near Intersection

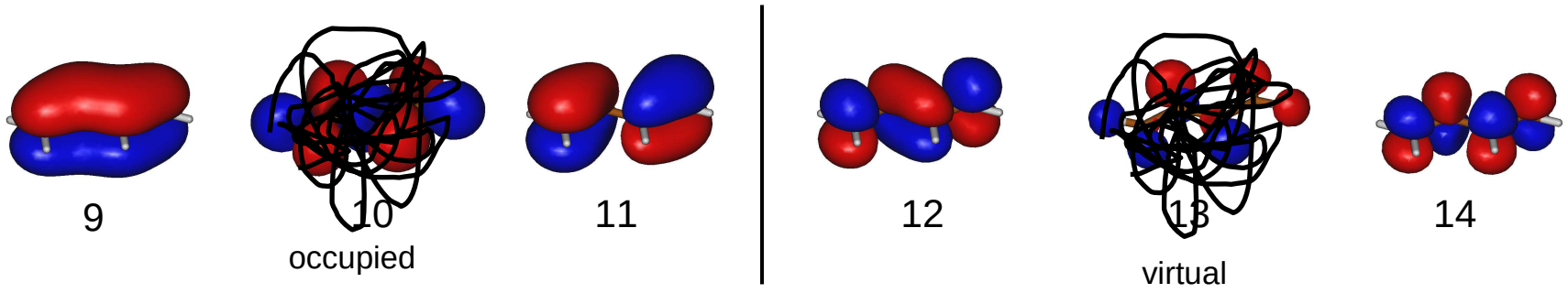
1200 points
Ca. 8 minutes

A little more advanced: correlation

… using the GUGA-CI graphical unitary group approach configuration interaction

Keyword : ***kci=5***

1. Select orbitals, procedure analog to CASSCF / CASPT2



Keyword : ***ici1***=*2* ***ici2***=*2* no of occup./virtual Orbs

Keyword : ***movo*** orbital selection mode, 0=automatic

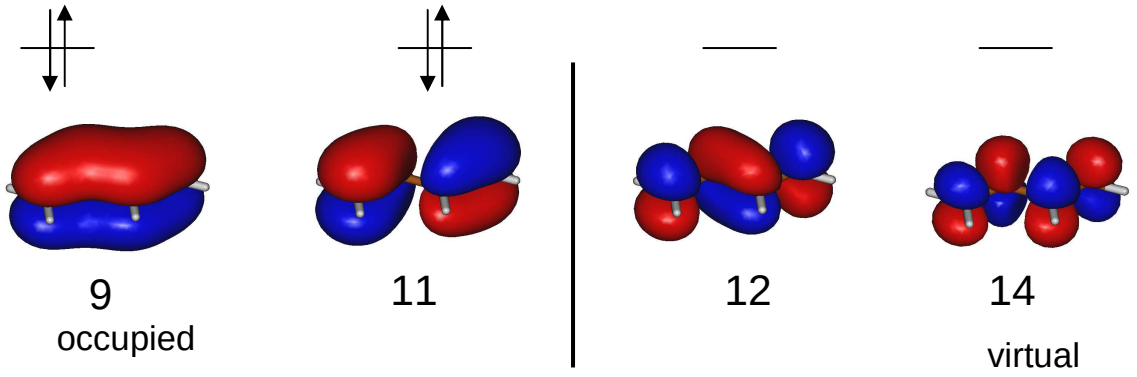
-1 to -4 : auto*magic* pi / d-orbital selection

A little more advanced: correlation

More confusion: closed shell or open shell??

Keyword : ***imult=0***

Closed shell RHF



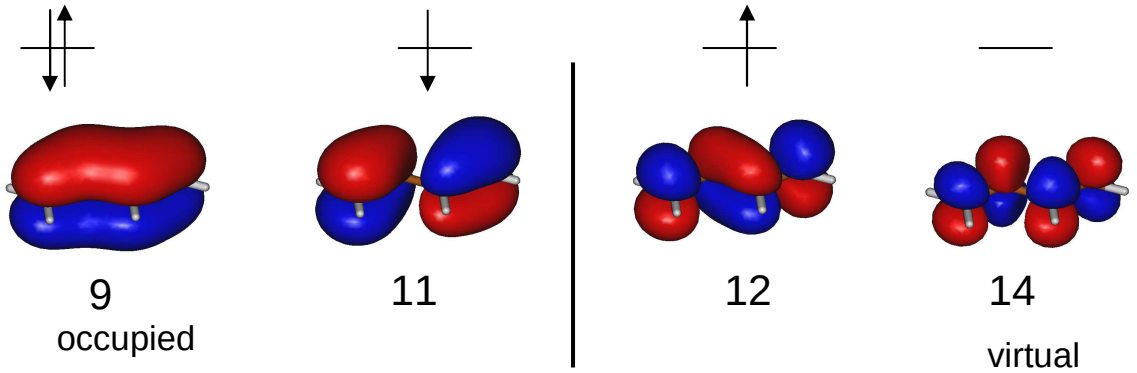
Keyword : ***ici1=2*** ***ici2=2*** no of occup./virtual Orbs

A little more advanced: correlation

More confusion: closed shell or open shell??

Keyword : ***imult=1***

Restricted open shell



Keyword : ***ici1=3*** ***ici2=1*** no of occup./virtual Orbs

sensible for excited state calculations

A little more advanced: correlation

levexc : Maximum excitation level (2=includes singles+doubles etc.)

iroot : Number of CI states to compute

nciref : Number of reference occupations

e.g. 3: 2 2 0 0
2 1 1 0
2 0 2 0

imomap : track orbital character by calculating the orbital overlap from step n-1 to step n (ensure orbital does not change)

mapthre : threshold for overlap

A little more advanced: correlation

Find the ground state minimum for butadiene:

```
jop=0 iop=-6  iform=1 nsav13=2 nsav7=4 +
icuts=-1 icutg=-1 iprint=1 mprint=1 ioutci=1 +
imult=1 kci=5 imomap=1 movo=1 ici1=3 ici2=1 nciref=3 +
mciref=0 levexc=2 iroot=8 iuvcd=2

OM2
6   0.0000000000 0     0.0000000000 0     0.0000000000 0    0    0    0
6   1.3369928941 1     0.0000000000 0     0.0000000000 0    1    0    0
1   1.0989330286 1   121.6512241461 1     0.0000000000 0    2    1    0
1   1.0841877889 1   121.8727580072 1   180.0048697504 1    1    2    3
1   1.0831647043 1   121.9934921422 1    -0.0086348342 1    1    2    3
6   1.4629956449 1   121.3381091819 1   180.0006023840 1    2    1    5
6   1.3369840108 1   121.3389136355 1   180.0063103715 1    6    2    1
1   1.0841915398 1   121.8738150692 1     0.0020269391 1    7    6    2
1   1.0831713785 1   121.9944742319 1   180.0023206952 1    7    6    2
1   1.0989393731 1   121.6516377824 1     0.0030140836 1    6    7    9
0   0.0000000000 0     0.0000000000 0     0.0000000000 0    0    0    0
9 11 12 14
```

A little more advanced: correlation

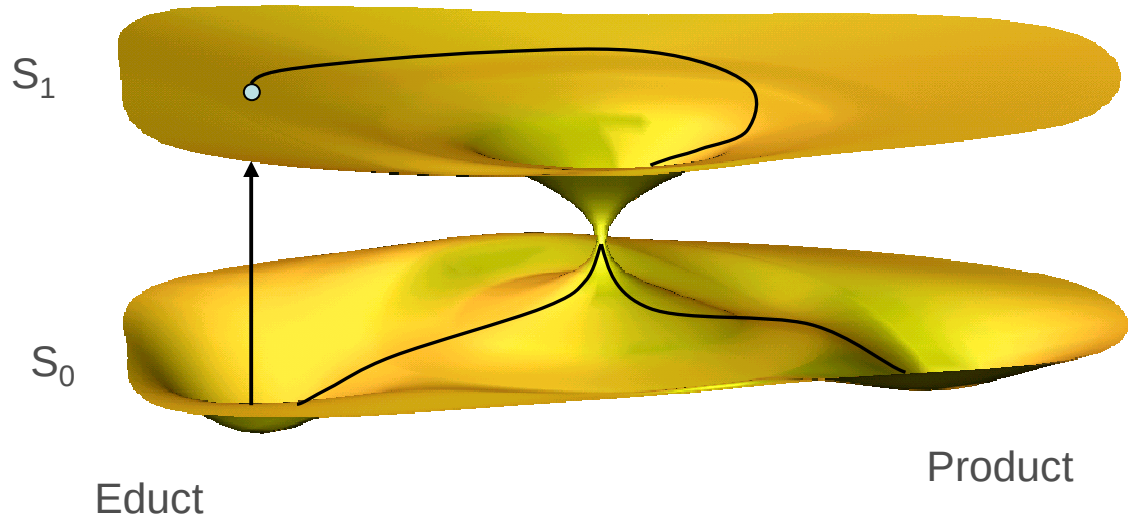
Find the ground state minimum for butadiene:

```
 State  1, E-E(1)=  0.000000 eV,  E=   -599.703824 eV

                          9  11   12  14
 main configuration:     ab  ab   -   -    (92%)
-----------------------------------------------------------
 State  2, E-E(1)=  5.892156 eV,  E=   -593.811668 eV

                          9  11  12  14
 main configurations:     ab   -  ab   -   (33%)
                          ab   a   -   b   (30%)
                          a   ab  b   -   ( 3%)
-----------------------------------------------------------
 State  3, E-E(1)=  6.194887 eV,  E=   -593.508936 eV

                          9  11  12  14
 main configuration:     ab   a   b   -    (92%)       HOMO-LUMO
```

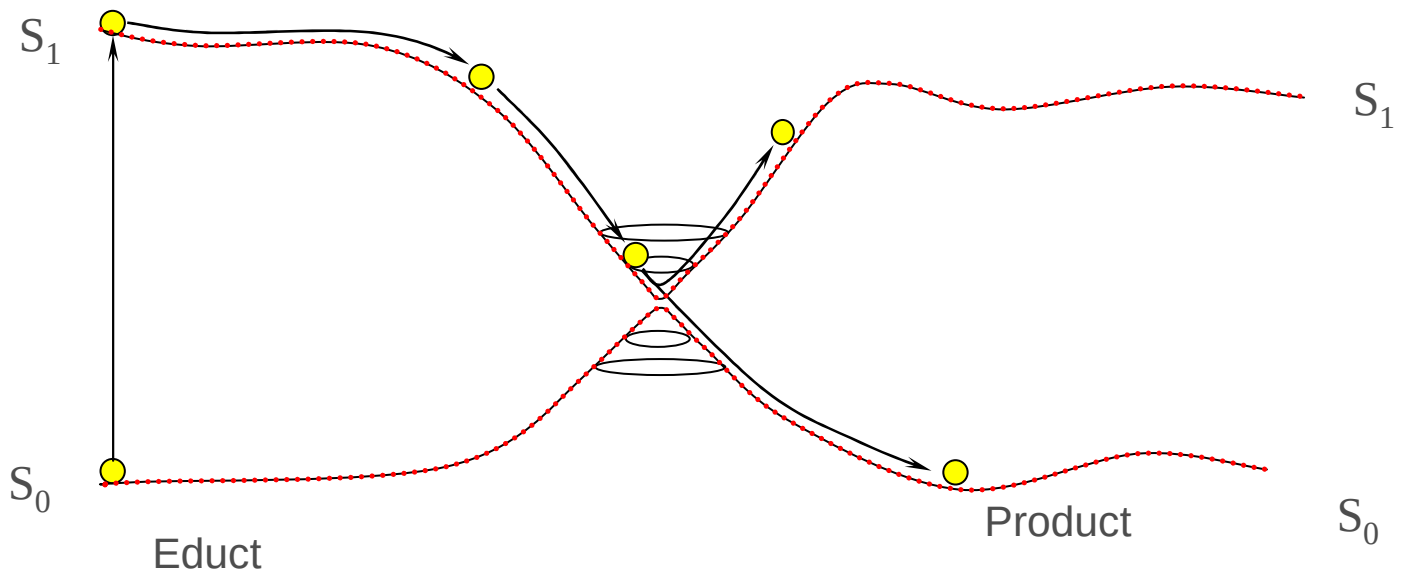
Transition properties:

1 -> 2 210.41 nm, f=0.000003
1 -> 3 200.13 nm, f=0.781175 ***EXP Value : 203.43 nm***

Methods for excited state molecular dynamics



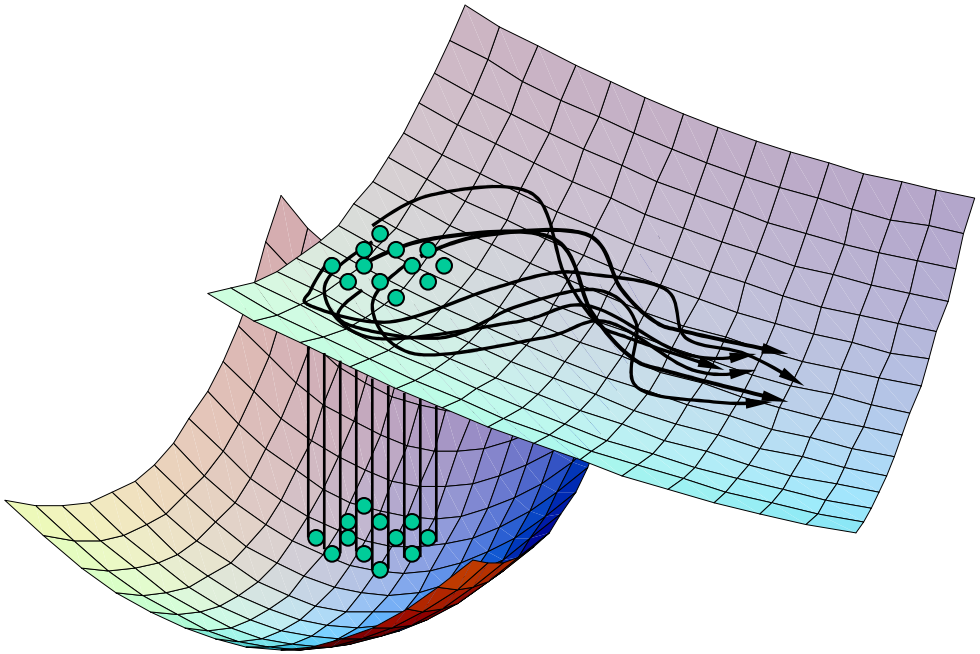
Methods for excited state molecular dynamics



- calculate all relevant potentials at the same time
- define starting conditions
- solve Newton‘s equations „on the fly“
- calculate transition properties

Methods for excited state molecular dynamics

Starting conditions for multiple trajectories by structure Sampling



Results from simulated photodynamics

- Life times of excited states
- Mechanisms
- Reaction channels
- Quantum yields

- ***Prerequisites***

- A reliable surface description (at least qualitatively)
- Statistically relevant number of trajectories

A strategy for excited state dynamics computations

- optimize the ground state structure
- Identify which states play a role
- spectroscopy, state properties
- Get an “idea” on the surface topology
- Identify relevant points on the surface
- i.e. conical intersections, turning points
- by: linear interpolations, minium energy path constrained optmization,

Finding conical intersections

- excited state PES Scan

```
iop=-6 iform=2 nsav13=2 nsav7=4 +
icuts=-1 icutg=-1 iscf=9 iplscf=9 iconv=4 kitscf=200 +
iprint=1 mprint=1 iprec=100 maxrtl=200 +
imult=1 ioutci=1 ici1=6 ici2=4 movo=1 +
kci=5  imomap=1 iroot=2 lroot=2 nciref=3 +
mciref=0 levexc=2 iroot=3 iuvcd=2

OM2
6    0.000000 0    0.000000 0    0.000000 0       0    0    0
6    1.392055 1    0.000000 0    0.000000 0       1    0    0
6    1.395900 1  120.191704 1    0.000000 0       2    1    0
6    1.396014 1  120.352814 1   -0.018576 1       3    2    1
6    1.392582 1  120.337189 1   -0.544254 1       4    3    2
6    1.402627 1  119.012833 1    1.001331 1       1    2    3
7    1.448153 1  117.420692 1 -176.708374 1       6    1    2
7    1.189038 1  123.056160 1 -129.040588 1       7    6    1
6    1.448180 1  123.078979 1   90.00000 0       8    7    6
6    1.401881 1  121.203911 1   55.357742 1       9    8    7
6    1.392601 1  118.860985 1  175.991989 1      10    9    8
6    1.395997 1  120.337524 1    0.077099 1      11   10    9
6    1.395905 1  120.355080 1   -0.534394 1      12   11   10
6    1.402635 1  121.227913 1    0.933470 1       9   10   11
1    1.092223 1  119.723793 1    3.494255 1       1    6    7
1    1.095119 1  119.827950 1 -179.643066 1       2    1    6
1    1.095504 1  119.840645 1  179.423401 1       3    2    1
1    1.094997 1  119.981316 1  179.313782 1       4    3    2
1    1.091694 1  120.488701 1   -3.383413 1       5    6    7
1    1.092223 1  119.723053 1  178.738480 1      14    9   10
1    1.095110 1  119.828735 1 -179.648422 1      13   14    9
1    1.095506 1  119.839020 1  179.425903 1      12   13   14
1    1.095001 1  119.681320 1 -179.773575 1      11   10    9
1    1.089944 1  120.431053 1 -178.723389 1      10    9   14
0    0.000000 0    0.000000 0   0.0000000 0       0    0    0
30  31  32  33  34  35  36  37  38  39
```

Some new commands

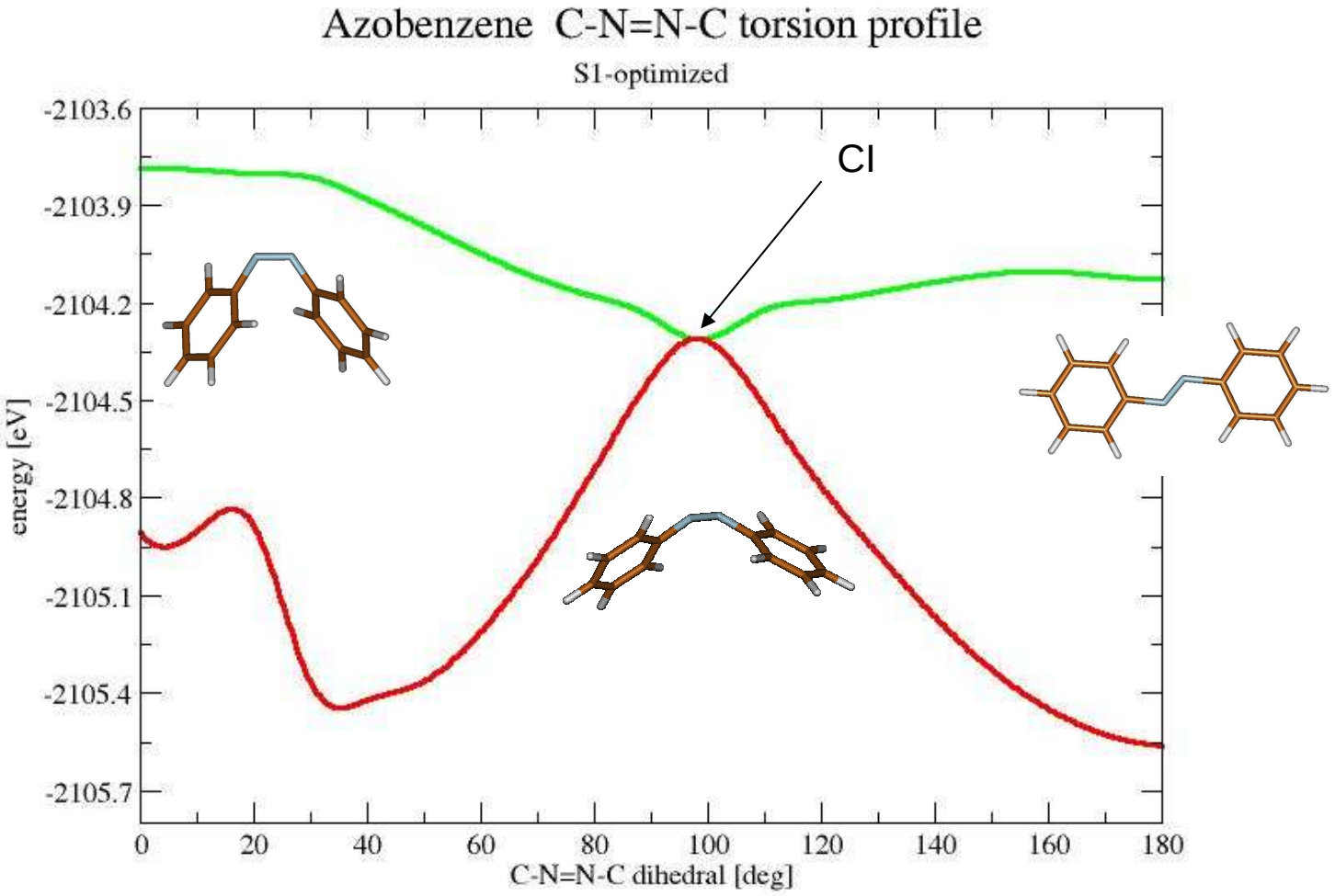
`icuts/icutg:`	integral cutoff (-1=none)
`iscf/iplscf:`	scf accuracy
`iconv:`	conv. criteria here:g98 values
`kitscf:`	max. scf iter.
`iprec:`	precision in geometry opt.
`Maxrtl:`	max. iter. for geom-opt

Problems:

- Automatic scan does not work reliably
- Watch out for “unsuccessful optimization”

Finding conical intersections

- excited state PES Scan



Finding conical intersections

```
iop=-6 iform=1 nsav13=2 nsav7=4 +
icuts=-1 icutg=-1 iscf=9 iplscf=9 +
iprint=1 mprint=1 iprec=100 +
ncigrd=2 icross=5 ief=1 lrscal=1 dmax=0.1 +
imult=1 ioutci=1 iuvcd=2 kci=5 imomap=1 maxrtl=200 +
movo=1 ici1=6 ici2=4 nciref=3 mciref=0 levexc=2 iroot=3 iuvcd=2 +
kitscf=200
CI-optimization Azobenzene
OM2
6     0.0000000000 0     0.0000000000 0     0.0000000000 0    0    0    0
6     1.4484112875 1     0.0000000000 0     0.0000000000 0    1    0    0
6     1.4459779594 1   117.6966453088 1     0.0000000000 0    2    1    0
6     1.3752082758 1   119.5396345456 1   -12.1667434977 1    3    2    1
6     1.4083959313 1   121.5088924960 1     4.7820727669 1    4    3    2
6     1.4059968285 1   119.0244711439 1     2.9981635534 1    5    4    3
7     1.4917626950 1   112.9830637031 1  -144.3739859433 1    2    3    4
7     1.2019910005 1   137.6801194230 1    51.2908304208 1    7    2    3
6     1.3816611614 1   133.0678659489 1   101.9616011121 1    8    7    2
6     1.4171809399 1   120.1552431166 1    -4.2182201339 1    9    8    7
6     1.3863601464 1   118.8457514349 1   177.7579318079 1   10    9    8
6     1.3974541858 1   120.8747049595 1    -0.4470403012 1   11   10    9
6     1.3997035197 1   119.9752523468 1     0.6584264619 1   12   11   10
6     1.3849114337 1   120.6970793145 1    -0.4238740804 1   13   12   11
1     1.0918237297 1   119.4881098949 1   181.5796950935 1   14    9   10
1     1.1047038909 1   119.9741012299 1  -179.6580055634 1   13   14    9
1     1.0968980204 1   119.6305947954 1  -180.1872329490 1   12   13   14
1     1.1016713777 1   119.6531868170 1   179.1030288912 1   11   10    9
1     1.0912114543 1   119.7865920238 1  -181.2948047173 1   10    9   14
1     1.0950064611 1   118.9735693279 1   -40.0007101523 1    1    2    7
1     1.1058627456 1   118.6393484376 1  -183.1874505080 1    6    5    4
1     1.0922074259 1   120.3750257539 1  -180.2254389980 1    5    4    3
1     1.1058366092 1   119.8491774878 1   184.3102850389 1    4    3    2
1     1.0930766251 1   118.7891250626 1    39.6873588510 1    3    2    7
0     0.0000000000 0     0.0000000000 0     0.0000000000 0    0    0    0
30  31  32  33  34  35  36  37  38  39
1   2
0.00010   1.00000
0
```

which gradients

Options for CI algorithm
(orth. Thresholds, standard values)

no more constraints

new commands

`ncigrd:` Number of gradients to compute

`icross:` multi-surface options, here: CI-optimizer
5=NR (Yarkony)
4=Bearpark
3=Ciminelli

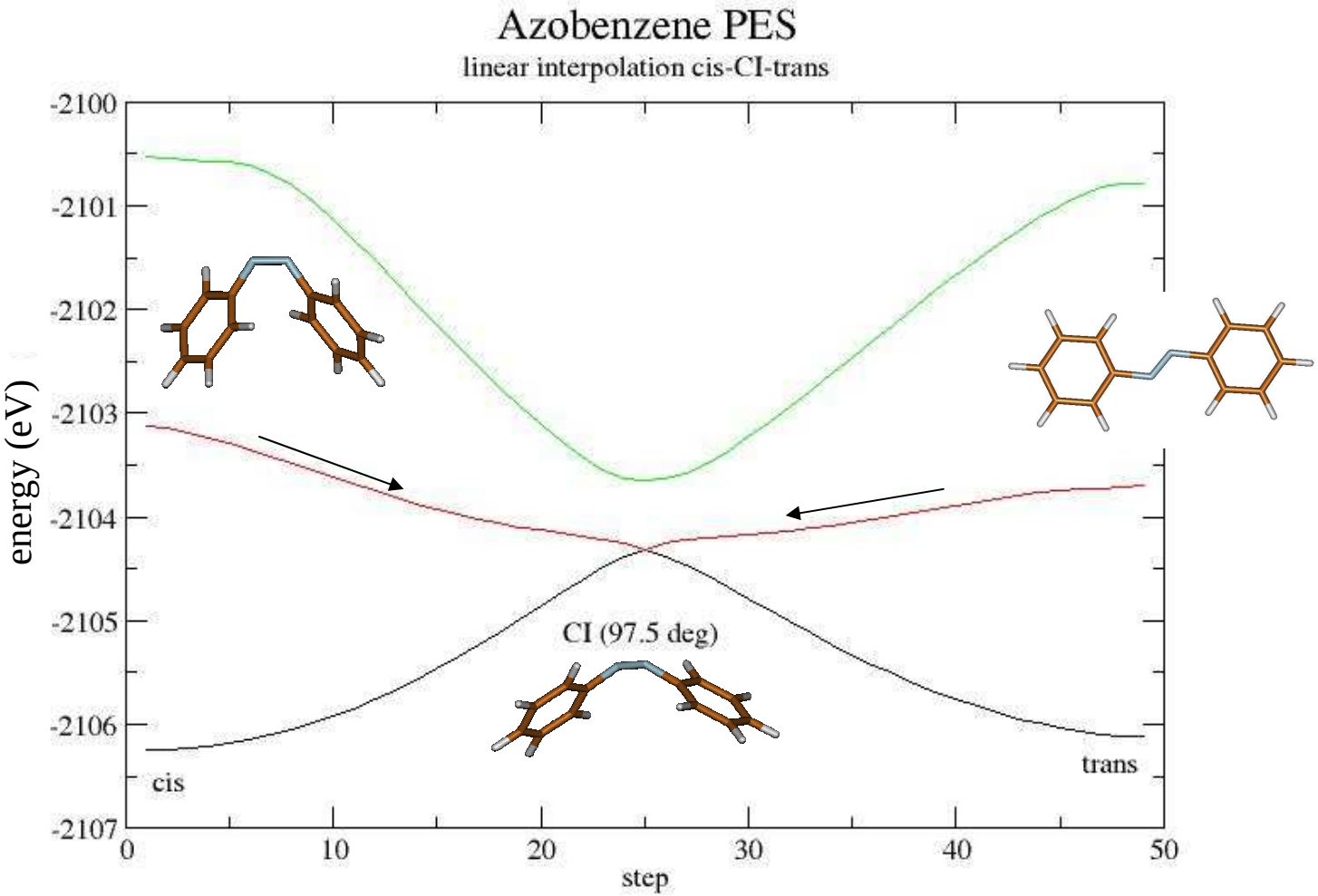
`ief:` 1=eigenvector following

`lrscal:` scale step

`dmax:` trust radius

Finding conical intersections

- Linear scaling between FC and CI points



Mario's ALMOST TEN COMMANDMENTS

THOU SHALT USE DYNAMICS ONLY FOR WHAT IS REALLY NECESSARY

TEST THY INPUTS AND OUTPUTS BEFORE SUBMITTING THY REAL PRODUCTION JOBS

THOU SHALT NOT LET COMPUTERS RUNNING ALONE WITHOUT CHECKING WHAT IS GOING ON

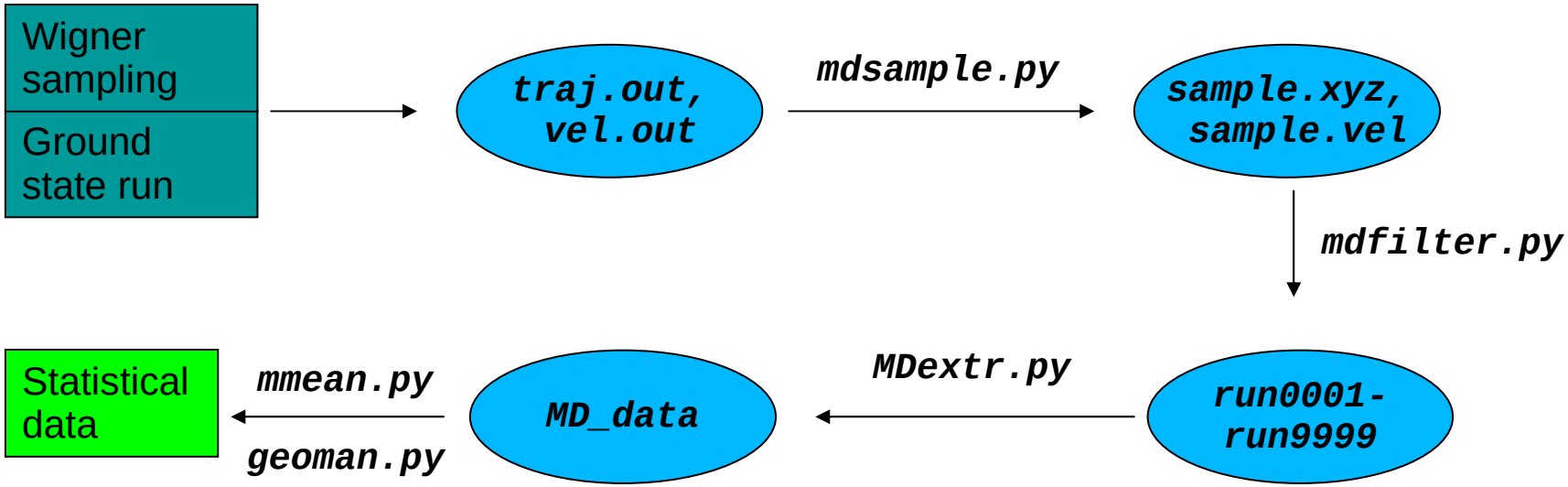
THOU SHALT NOT START DYNAMICS WITHOUT A VERY CAREFUL INVESTIGATION OF THE THEORETICAL LEVEL

THOU ART A LIMITED AND BIASED BEING, THEREFORE, THOU SHALT NOT RELY ON THY EYES TO PERFORM ANALYSIS

Preparing excited state trajectories

- Choose initial conditions
- 2 Options:
  - Start points from equillibrated long S_0 run
    - Trajectory points NOT independent
    - Equipartitioning of kinetic energy in vibrational modes
      too much energy in low frequency modes
  - create initial structures by Wigner / Boltzmann sampling
    - too much kinetic energy
    - transfer of energy into low frequency modes in long runs

Preparing excited state trajectories



Preparing excited state trajectories

- Starting from a long run trajectory
  - The ***mdsample.py*** script
    - Select geometries and velocities from traj.out and vel.out

Usage:

mdsample.py –vel –vel-input vel.out –sample-size=100

Selects randomly 100 points from traj.out + corresponding velocities

mdsample.py –vel –vel-input vel.out –--step=100

Collects geometries and velocities every 100 steps (very short interval)

⇒ ***sample.xyz, sample.vel***

Preparing excited state trajectories

- Selecting valid starting points
  - The ***`mdfilter.py`*** script
    - Select geometries and velocities from sample.xyz according to
      - a certain energy window
      - their transition probabilities
      - orbital overlap with the initial structure

AND

- prepare the necessary input files for the trajectory runs

`mdfilter.py` needs:

- a *template input file* holding the optimized(!) ground state structure of your system reference
- a ***`dynvar.in`*** file defining the properties of the dynamics
- a patient user(!)

Preparing excited state trajectories

template.inp :

```
iop=-6 igeom=1 iform=1 icuts=-1 icutg=-1 +
iscf=11 iplscf=11 dstep=0.00001 +
kharge=1 imult=0 kci=5 +
movo=1 ici1=3 ici2=1 nciref=3 mciref=0 levexc=2 iroot=4 +
mprint=-1 nprint=-5
Example methaniminium template file
Geometry from ground state optimisation
6      -0.00210  1    -0.00063  1    -0.00001  1
7       1.28587  1    -0.00077  1     0.00001  1
1      -0.58316  1     0.93103  1     0.00000  1
1       1.83665  1     0.85914  1     0.00003  1
1      -0.58341  1    -0.93214  1    -0.00003  1
1       1.83641  1    -0.86084  1     0.00000  1
0       0.0      0     0.0      0     0.0      0
3  5  6  7
```

No state information / orbital mapping options (added automagically)

Preparing excited state trajectories

dynvar.in : important options:

```
NSTEP        = 4000,
DT           = 5.0000000000000000E-005,    in ps
RESTART      = F,                          T with start vel.!
EHRENFEST    = F,
TULLY_HOP    = T,                          surface hopping
EINTEG       = UP3 ,                       integrator
NE           = 200,
AN_CC        = T,                          analytic couplings
RND_GEN      = PM_BD,                      check seed!
WRITE_HOP  = T,                            hop info
FHOP         = 1,                          follow hop
VS           = F,                          no velocity scaling
```

No state information / random seed (added automagically by mdfilter)

Preparing excited state trajectories

mdfilter.py does:

- comparison of ground state minimum with actual geometry
- generate input files for job submission (run0001-run9999)

submit_many_jobs.sh

- script for automated job submission on cluster nodes

important files

- stat.out potential, kinetic and total energies
- traj.out structures
- hopping.out hopping info

After completion:

- extract all relevant data from trajectories with ***MDextr.py***

Preparing excited state trajectories

Output files

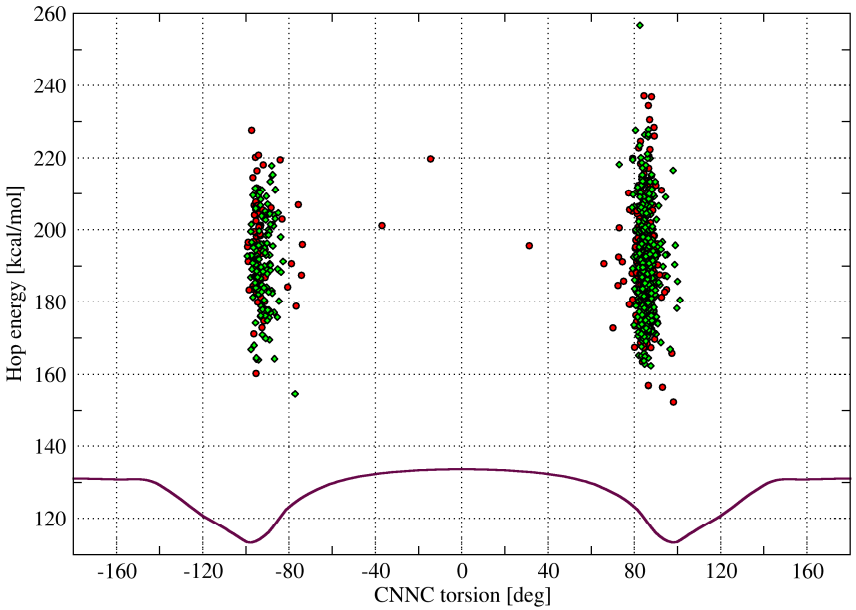
pop1.dat	electronic state populations
prob1.dat	state probabilities
state.dat	state occupation
E1.dat E12.dat	energies
CC12.dat	coupling vector
Rnd_num.dat	

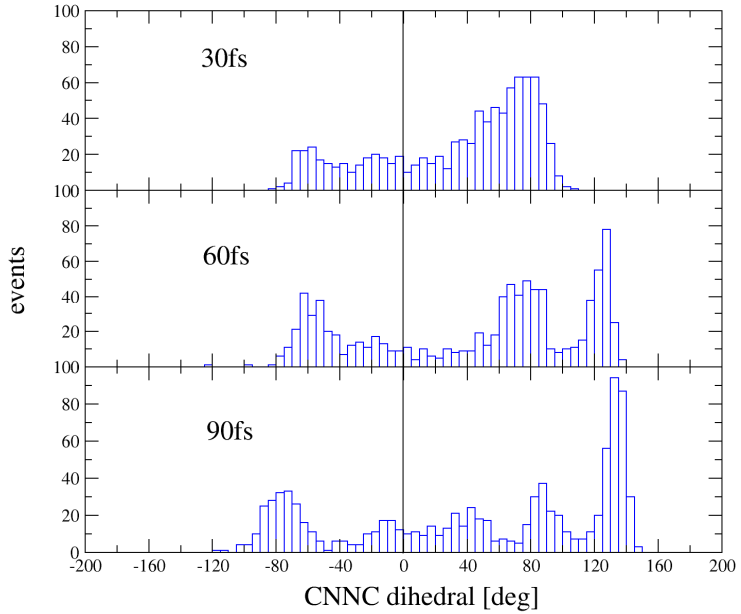
Data analysis

- ***mmean.py***
- ***geoman.py***

Data analysis

Look at individual trajs





Find correlation

Analyse distributions