www.newtonx.org

A Tutorial for NEWTON-X

Tutorial based on NEWTON-X versions 1.2 and 1.4

Matthias Ruckenbauer,[1] Daniele Fazzi,[2] and Mario Barbatti[2]

[1]Institute for Theoretical Chemistry – University of Vienna, Vienna
[2]Max-Planck-Institut für Kohlenforschung, Mülheim an der Ruhr

2009-2014

The views or opinions expressed in this tutorial are solely those of the authors. They do not necessarily represent those of the institutions to which they are or were affiliated.

Please cite this tutorial as:

- Matthias Ruckenbauer, Daniele Fazzi, and Mario Barbatti, "A Tutorial for Newton-X", www.newtonx.org, 2009-2014.

Table of contents

1. Before starting

1.1 Programs necessary to follow this tutorial

- Newton-X.[1-3] The variable $NX should be defined pointing to the program. Most of this tutorial is based on Newton-X version 1.2. Changes between version 1.2 and 1.4 are indicated in proper places. Chapters dealing with Gaussian 09[4] interface are based on version 1.4.
- Columbus[5, 6] for spectrum and dynamics using MRCI or MCSCF methods. The variable $COLUMBUS should also be defined pointing to the program.
- Turbomole[7] for spectrum and dynamics using TD-DFT or RI-CC2/ADC(2) methods. The Turbomole addresses should be in $path.
- Gaussian 09 for spectrum and dynamics using TD-DFT. $g09root should be defined and g09.login should have been sourced.
- Tinker[8] for QM/MM dynamics. Tinker executable 'testgrad.x' should be in the standard path or available.
- Perl 5.8 or higher.
- Molden.
- Gnuplot.

1.2 Notation

1. The Tutorial is divided in several sections with different jobs. Advanced jobs may make reference to basics jobs.
2. The basic instructions are sequentially numbered.

```
This kind of font indicates what is seen in the screen
and the command lines that you should type <ENTER>     ! Comments come here
```

Important information related to Newton-X *but not necessarily connected to the current job comes in boxes like this. It is advisable to read the boxes, but you can follow the tutorial without them.*

1.3 Preparing the jobs

3. Create a directory TUTORIAL. Inside TUTORIAL, new subdirectories will be created for each job.

You should organize your jobs using the suggested structure of directories. This will make easier to copy files between different jobs.

2. Initial conditions and spectrum simulation: TD-DFT – Turbomole

In this section, we will simulate absorption spectrum with TURBOMOLE. The system is pyrazine and two singlet excited states will be included. The initial geometry distribution will be generated by a Wigner distribution in the ground vibrational state of the ground electronic state. The transition energies and moments will be also computed at TD-DFT(B3-LYP)/SV(P). Theoretical details about the spectrum simulation are discussed in Ref.[9].

2.1 Geometry file creation

1. In the TUTORIAL directory (see section 1.3) create a subdirectory called PYRAZINE_TDDFT_SPEC:

```
> mkdir PYRAZINE_TDDFT_SPEC
```

2. Move to this directory and create a file called pyrazine.xyz containing the geometry in XYZ format:

```
> cd PYRAZINE_TDDFT_SPEC
> vi pyrazine.xyz
```

```
10

      N        -0.000000         2.421234         1.397837
      N        -0.000000         0.003637         0.002163
      C         0.000000         1.241014         2.044972
      C        -0.000000         0.033417         1.347829
      C        -0.000000         2.391454         0.052171
      C         0.000000         1.183858        -0.644972
      H         0.000000         1.271951         3.113613
      H        -0.000000        -0.907483         1.855431
      H        -0.000000         3.332354        -0.455431
      H         0.000000         1.152921        -1.713613
```

This initial geometry can be created by hand with a text editor (like VI*) or with any molecular editor like* MOLDEN*. It is the simple xyz Cartesian format in Angstrom. The first line contains the number of atoms and it is followed by a blank line.*

3. Convert geometry into NEWTON-X format.

```
> $NX/xyz2nx < pyrazine.xyz <ENTER>
```

The NEWTON-X *geometry file is called 'geom' and looks like*

```
N     7.0    0.00000000    4.57546916    2.64152911   14.00307401
N     7.0    0.00000000    0.00687293    0.00408748   14.00307401
C     6.0    0.00000000    2.34517659    3.86443703   12.00000000
C     6.0    0.00000000    0.06314898    2.54702768   12.00000000
C     6.0    0.00000000    4.51919312    0.09858890   12.00000000
C     6.0    0.00000000    2.23716740   -1.21882044   12.00000000
H     1.0    0.00000000    2.40363904    5.88387585    1.00782504
H     1.0    0.00000000   -1.71489434    3.50625645    1.00782504
H     1.0    0.00000000    6.29723644   -0.86063986    1.00782504
H     1.0    0.00000000    2.17870494   -3.23825927    1.00782504
```

The second column contains the atomic number, and the following three columns contain the x, y and z coordinates in atomic units (Bohr). The last column contains the atomic masses, which can be eventually changed to simulate isotopic effects. The file can be freely formatted.

The conversion back from NEWTON-X *format to xyz can be done with.*

```
> $NX/nx2xyz
```

After running this program, the xyz-format geometry is written to file 'geom.xyz'.

2.2 Normal-modes step

4. Create a new directory to run normal-mode calculation:

```
> mkdir FREQ-DFT <ENTER>
```

5. Move to this directory and copy the geom file there.

```
> cd FREQ-DFT
> cp ../geom .
```

6. Transform the geometry into TURBOMOLE format.

```
> $NX/nx2tm
```

nx2tm program converts NEWTON-X *"geom" file into* TURBOMOLE *"coord" file (atomic units). The back conversion can be done with*

```
> $NX/tm2nx
```

7. Prepare the input and run a normal mode calculation at DFT(B3-LYP)/SV(P) level with TURBOMOLE. In order to do that, follow specific TURBOMOLE documentation (www.turbomole.com). Let the geometry be optimized in order to have only real frequencies. The TURBOMOLE output containing the normal modes and frequencies can have any name. Here we will suppose it is called "force.out".

NEWTON-X *can read normal modes and frequencies from the outputs of the flowing programs:* GAMESS, TURBOMOLE, COLUMBUS, GAUSSIAN, MOLDEN, DFTB, *and* ACES2.

Note that the normal modes and harmonic frequencies are used only to set the width of the random number distributions. For this reason they do not need be calculated at the same level as the one that will be used in the dynamics. It is, however, absolutely necessary that the order of atoms and geometry orientation in the "geom" file and in the normal modes are exactly the same. Using NEWTON-X *conversion tools (like nx2tm) assures that.*

2.3 Energy and transition moment input

8. Move to the parent directory (TUTURIAL/PYRAZINE_TDDFT_SPEC) and create a new subdirectory called JOB_AD.

```
> cd ..
> mkdir JOB_AD
```

The directory must be named JOB_AD since NEWTON-X *will search for it.*

9. Move into JOB_AD and copy the TURBOMOLE geometry file there.

```
> cd JOB_AD
> cp ../FREQ-DFT/coord .
```

10. In this directory, prepare a TURBOMOLE input for single point calculation at TD-DFT(B3-LYP)/SV(P) without symmetry, with two singlet excited states, and transition moments (oscillator strengths) calculations. Follow the TURBOMOLE documentation (www.turbomole.com).

This is the basic procedure for any program. If, for example, the initial conditions should be generated using RI-CC2 energies, JOB_AD should contain a TURBOMOLE *input for single point calculations at RI-CC2 level.*

Before submitting running any NEWTON-X *job is highly advisable to run the job in JOB_AD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_AD into a test directory and run the* TURBOMOLE *job. Check whether the three roots and the oscillator strengths are computed and show reasonable values.*

2.4 NEWTON-X input

11. Move to directory TUTORIAL/PYRAZINE_TDDFT_SPEC and create a subdirectory called IC-TDDFT.

```
> cd ..
> mkdir IC-TDDFT
```

12. Move to this directory and copy the optimized DFT geometry file there.

```
> cd IC-TDDFT
> cp ../FREQ-DFT/coord .
```

13. Convert the new TURBOMOLE geometry file into NEWTON-X format.

```
> $NX/tm2nx
> rm -f coord    ! We do not need this file anymore.
```

14. Copy the JOB_AD directory and the normal modes file into IC-TDDFT.

```
> cp -rf ../JOB_AD .
> cp ../FREQ-DFT/force.out .
> ls
force.out  geom  JOB_AD
```

15. Run the nxinp program and select option 1: Generate initial conditions.

nxinp is the input tool for Newton-X. *All input procedures, from the initial condition generation, through the dynamics inputs, to the statistical analysis options are performed with it.*

```
> $NX/nxinp


                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT

Select one option (1-7): 1 <ENTER>
```

16. Select the adequate options for the initial condition properties.

nxinp will guide you through a set of questions. Each of them contains a short explanation and the available options. Most of times, the default value is the best option. To accept the default you can just press <ENTER>.

```
...
nact:  Distribution:
        1 - Classic harmonic oscillator
        2 - Q/P-uncorrelated quantum harmonic oscillator (Wigner)
        3 - Q/P-correlated quantum harmonic oscillator
        4 - Pick points from previous dynamics
        5 - Random-velocities generation
        6 - Single point on a previous calculation
 The current value of nact  is: 2
 Enter the new value of nact  : <ENTER>
 Setting nact = 2

 numat:  Number of atoms.
 The current value of numat  is: 10
 Enter the new value of numat  : <ENTER>
 Setting numat = 10

 npoints:  Number of initial conditions to be generated
 The current value of npoints  is: 1
 Enter the new value of npoints  : 50 <ENTER>
 Setting npoints = 50
```

npoints sets the number of initial conditions that should be generated. This number depends on how many points you want to compute to generate the spectrum and how many trajectories you want to simulate.

In real applications, you will need much more than 50 points. 500 may be a good number.

You can also run several short jobs with few points in each one and merge them afterwards using the tool "merge_initcond.pl". Thus, with 10 computers running 50 points each, you can quickly have 500 points. See in section 2.5 a note about how to split and merge the jobs.

```
 file_geom:  File containing the equilibrium geometry.
 The current value of file_geom  is: 'geom'
 Enter the new value of file_geom  : <ENTER>
 Setting file_geom = 'geom'

 iprog:  Read vibrational modes from:
        1 - GAMESS output
        2 - TURBOLMOLE output
        3 - COLUMBUS output
        4 - GAUSSIAN output
        5 - MOLDEN file
        6 - DFTB output
        7 - ACES2 output
 The current value of iprog  is: 2
 Enter the new value of iprog  : <ENTER>
 Setting iprog = 2

 file_nmodes:  File containing the normal modes.
 The current value of file_nmodes  is: 'force.out'
 Enter the new value of file_nmodes  : <ENTER>
 Setting file_nmodes = 'force.out'

 anh_f:  Multiply harmonic frequencies by this factor.
 The current value of anh_f  is: 1
 Enter the new value of anh_f  : <ENTER>
 Setting anh_f = 1

 chk_e:  Restrict (or do not) the statistical distribution:
        0 - Do not check the energies.
        1 - Check the energies between states NIS and NFS.
 The current value of chk_e  is: 0
 Enter the new value of chk_e  : 1 <ENTER>
 Setting chk_e = 1


The next keywords define the restriction on the statistical distribution.


 nis:  Intial state (State in which the previous dynamics run. Ground state = 1.).
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Final state.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 3 <ENTER>
 Setting nfs = 3
```

The Wigner distribution will be generated in the ground electronic state (nis = 1) and transition energies between the ground and the second excited (nfs = 3) states will be computed. Indeed, with nfs = N, transition information for all states below N will also be computed and collected.

```
kvert:  Center of the restriction:
       0 - Provide the energy for center of restriction
       1 - Use the vertical excitation energy of the equilibrium geometry
The current value of kvert  is: 1
Enter the new value of kvert  : <ENTER>
Setting kvert = 1

de:  Width of restriction (center +/- de/2).
The current value of de (eV) is: 0.5
Enter the new value of de (eV) : 100 <ENTER>
Setting de = 100
```

This large "de" implies that this restriction will not be used. It can be imposed later on.

```
prog:  Program and method to compute vertical excitation energies:
       1   - COLUMBUS
       2.0 - TURBOMOLE RI-CC2
       2.1 - TURBOMOLE TD-DFT
       5   - DFTB
The current value of prog  is: 1
Enter the new value of prog  : 2.1 <ENTER>    ! Turbomole TD-DFT
Setting prog = 1

iseed:  Random number seed.
        0 - default seed value.
       -1 - set random seed.
       >0 - (integer) set specific value for the seed.
The current value of iseed  is: 0
Enter the new value of iseed  : <ENTER>
Setting iseed = 0

lvprt:  Print level.
       1 - standard level
       2 - debug level
The current value of lvprt  is: 1
Enter the new value of lvprt  : <ENTER>
Setting lvprt = 1
```

17. Exit nxinp.

2.5 Running NEWTON-X: initial conditions

18. Run the initial condition generation program.

```
> $NX/initcond.pl > initcond.log &
```

In general, with chk_e = 1, you might want to submit this job to a batch system. Files "pini" and "pini-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

Splitting and merging jobs:

To split the job among several computers, just prepare the input as it was done until here. Then run

```
> $NX/split_initcond.pl
```

This program will create a directory called INITIAL_CONDITIONS and inside it a series a sequence of subdirectories called I1, I2, ..., each one containing a complete set of input files.

Copy each directory to a different computer and run the job normally.

To merge the jobs, after the calculations, copy the directories I1, I2, ... back to INITIAL_CONDITIONS directory and from inside this directory run

```
> $NX/merge_initcond.pl
```

This program will ask the number of jobs to be merged and it will create a new directory called I_merged with merged results. Move to this directory and proceed with the spectrum simulation as explained below.

2.6 Checking the results

19. Normal job termination is indicated in the end of initcond.log file by:

```
End of initial conditions generation.
          ====================== NEWTON-X ends here ==================
```

20. The main information is written to the final_output files.

21. Both, final_output.1.2 and final_output.1.3 contains the same 50 geometries and velocities generated according to the Wigner distribution. final_output.1.2 contains transition energies and oscillator strengths to excitations into state 2 (first excited state), while final_output.1.3 contains the same information to excitations into state 3 (second excited state).

2.7 Generating the spectrum

22. Go to directory TUTORIAL/PYRAZINE_TDDFT_SPEC and create the directory SPECTRUM.

```
> mkdir SPECTRUM  !This name is not compulsory.
```

23. Move to this directory and copy final_output files there.

```
> cd SPECTRUM
> cp ../IC_TDDFT/final_output* .
```

24. Run nxinp and select option 5: “Generate Trajectories and Spectrum”.

```
> $NX/nxinp
                                1. GENERATE INITIAL CONDITIONS

                                2. SET BASIC INPUT

                                3. SET GENERAL OPTIONS

                                4. SET NONADIABATIC DYNAMICS

                                5. GENERATE TRAJECTORIES AND SPECTRUM

                                6. SET STATISTICAL ANALYSIS

                                7. EXIT
```

```
Select one option (1-7): 5
```

25. Go through the next options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : 1
 Setting type = 1
```

If for this first option you are asked for "kindjob" instead "type", this means you are using an old version of Newton-X*. Contact the distributors to get an updated version of the program.*

```
nis:  Initial state.
The current value of nis  is: 1
Enter the new value of nis  : <ENTER>
Setting nis = 1

nfs:  Array of final states (coma or dash separated, e.g., 2-4 or 2,3,4).
For trajectories generation only one final state is allowed.
The current value of nfs  is: 2
Enter the new value of nfs  : 2,3 <ENTER>  !Spectrum including both states
Setting nfs = 2,3
```

If nfs should be given as a 'space separated' list, this menas that you are using an old version of Newton-X. Contact the distributors to get an updated version of the program.

```
screen:  Energy restriction
       0 - don't apply any restriction
       1 - use the original energy restriction written in the final_output files
       2 - apply new energy restriction
The current value of screen  is: 0
Enter the new value of screen  : <ENTER>
Setting screen = 0

os_condon:  Oscillator strength:
 -1 - try to read from final_output file.
  x - oscillator strength is always x (Condon approximation).
The current value of os_condon  is: -1
Enter the new value of os_condon  : <ENTER>
Setting os_condon = -1
```

If the oscillator strength is not available but the transition energy is, it is still possible to simulate the spectrum by giving a value to the oscillator strength in os_condon option. In this case, all transitions will be computed with the same oscillator strength. For only one final state this option is not bad, but it can be completely inadequate when more than one final state is selected.

```
prob_kind:  Formula to calculate the transition probabilities:
       A - Einstein-coefficient A (spontaneous emission)
       B - Einstein-coefficient B (induced absorption or induced emission)
       F - Oscillator strength (photoabsorption cross section)
The current value of prob_kind  is: B
Enter the new value of prob_kind  : <ENTER>
Setting prob_kind = F

norm:  Normalization of transition intensities:
       local  - Use energy-restricted data set
       global - Use complete data set
The current value of norm  is: local
Enter the new value of norm  : <ENTER>  ! with no energy restriction, doesn't
Setting norm = local                    ! matter.

seed:  Seed for the random number generation
       0 - a default random number seed is used
       1 - a randomized seed is used
       Any other positive integer is used as the random number seed
The current value of seed  is: 0
Enter the new value of seed  : <ENTER>
Setting seed = 0

l_shape:  Line shape:
       gauss   - Normalized Gaussian function.
       lorentz - Normalized Lorentzian function.
The current value of l_shape  is: lorentz
Enter the new value of l_shape  : <ENTER>
Setting l_shape = lorentz

delta:  Phenomenological broadening of the spectrum.
The current value of delta (eV) is: 0.05
Enter the new value of delta (eV) : <ENTER>
Setting delta = 0.05
```

"delta" is possibly the only option that you need to play with. It controls the width of the gaussian curve attributed to each one of the 100 transitions (50 initial conditions x 2 final states).

```
temp:  Temperature.
The current value of temp (K) is: 0
Enter the new value of temp (K) : <ENTER>

nref:  Refraction index.
The current value of nref  is: 1
Enter the new value of nref  : <ENTER>

eps:  Distance between consecutive points in the spectrum.
The current value of eps (eV) is: 0.005
Enter the new value of eps (eV) : <ENTER>
Setting eps = 0.005

kappa:  The range of the spectrum is defined between
Emin-kappa*delta and Emax+kappa*delta, where kappa is an integer.
The current value of kappa  is: 3
Enter the new value of kappa  : <ENTER>
Setting kappa = 3
```

26. Exit nxinp.

27. The simulated cross section using a Lorentzian line shape with phenomenological broadening δ= 0.05 eV is written to cross-section.dat. Looking in this file, we see:

```
    ! DE(eV)  Wavelength(nm)  cross-section(Å^2.molecule^-1)
      3.1914   3.8849e+02       0.0003
      3.1964   3.8789e+02       0.0003
      3.2014   3.8728e+02       0.0003
...
      3.8014   3.2615e+02       0.0304
      3.8064   3.2573e+02       0.0306
...
```

28. Run Gnuplot to visualize the spectrum.

```
> gnuplot
gnuplot> plot 'cross-section.dat' using 1:3 with lines
```

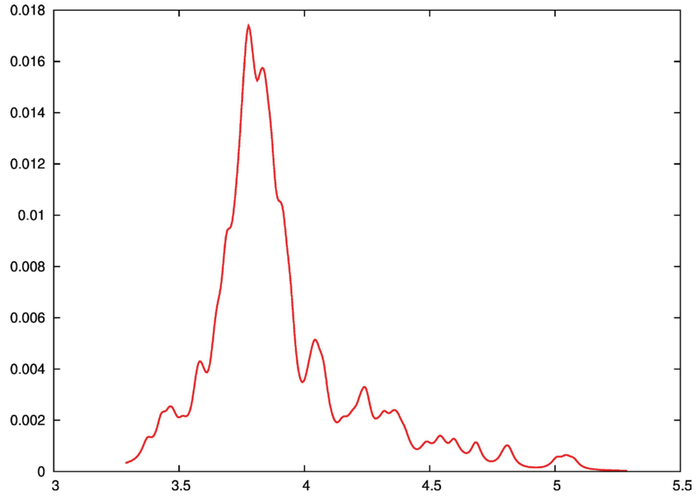


Fig. 1. Photoabsorption cross section in Å^2.molecule^{-1} as a function of the excitation energy in electron-Volts.

3. Surface hopping dynamics: TDDFT – Turbomole

In this section, we will prepare the inputs and run dynamics simulations. The system is pyrazine and the dynamics will be initiated in the second excited state. The initial conditions were generated in Chapter 2 of this tutorial. The surface hopping[10] dynamics will be performed at TDDFT level using TURBOMOLE. The trajectory time will be 100 fs with 0.5 fs time step.

The TDDFT will use the B3-LYP functional and the SV(P) basis set. The dynamics will start in S_2, with initial conditions selected in the interval 3.5 to 4 eV.

3.1 Copying the initial conditions

1. Go to TUTORIAL directory (see section 1.3) create a subdirectory called PYRAZINE_TDDFT_DYN:

```
> mkdir PYRAZINE_TDDFT_DYN
```

2. Move to this directory and copy the initial conditions file for the second excited state there.

```
> cd PYRAZINE_TDDFT_DYN
> cp ../PYRAZINE_TDDFT_SPEC/IC_TDDFT/final_output.1.3 final_output ! note the new
                                                                     name
```

3. Create a subdirectory called JOB_AD.

> *The directory must be named JOB_AD since* NEWTON-X *will search for it.*

3.1 Preparing electronic structure input

4. Move to JOB_AD and create an input for single point TDDFT/B3LYP/SV(P) calculation. Use C_1 point group (no symmetry) and request the calculation of two excited states belonging to representation *a*.

> *Before submitting running any Newton-X job is highly advisable to run the job in JOB_AD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_AD into a test directory and run the* TURBOMOLE *job. Check whether the three roots are computed and show reasonable values.*

3.2 NEWTON-X input

5. Run nxinp.

```
> $NX/nxinp
```

```
                ============================================================
                                          NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT

Select one option (1-7): 2 <ENTER>
```

6. Select option 2: “set basic input” and answer the questions to set the dynamics options.

```
 nat:  Number of atoms.
 The current value of nat  is: 10
 Enter the new value of nat  : <ENTER>
 Setting nat = 10

 nstat:  Number of states.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>     ! 3 states will be computed
 Setting nstat = 3

 nstatdyn:  Initial state (1 - ground state).
 The current value of nstatdyn  is: 3
 Enter the new value of nstatdyn  : <ENTER>    ! dynamics will start in state 3
 Setting nstatdyn = 3

 dt:  Time step for the classical equations.
 The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 tmax:  Total duration of the trajectory.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : <ENTER>
 Setting tmax = 100
```

```
prog:  Quantum chemistry program and method
       0    - ANALYTICAL MODEL
       1    - COLUMBUS
       2.0 - TURBOMOLE RI-CC2
       2.1 - TURBOMOLE TD-DFT
       2.2 - TURBOMOLE ADC(2)             ! NX > 1.4
       3    - ACES2 EOM-CCSD (in development)
       4    - MOPAC (in development)
       5    - DFTB
       6    - GAUSSIAN                    ! NX < 1.2
       6.0 - GAUSSIAN(03) CASSCF          !
       6.5 - GAUSSIAN(09) TDDFT           !
       7    - TINKER
       8    - DFTB+
       10.0    - GAMESS
       10.1    - GAMESS ARBITRARY ADIABATIC METHOD
       20   - HYBRID GRADIENT
The current value of prog  is: 1
Enter the new value of prog  : 2.1 <ENTER>
Setting prog = 2.1

thres: Threshold to initiate nonadiabatic coupling calculation.
         0 - Only adiabatic dynamics.
       100 - Only nonadiabatic dynamics
The current value of thres (eV) is: 0
Enter the new value of thres (eV) : 100 <ENTER>
Setting thres = 100

Would you like to generate internal coordinate file?  (default=y) n <ENTER>
```

7. In the main nxinp menu you have the option to set more general options if you select either option 3 or option 4. Normally, this is not necessary and the default values are good enough.

8. Select option 5: “generate trajectories and spectrum”.

```
============================================================
                          NEWTON-X
         Newtonian dynamics close to the crossing seam
                       www.newtonx.org
============================================================

                MAIN MENU


                1. GENERATE INITIAL CONDITIONS


                2. SET BASIC INPUT


                3. SET GENERAL OPTIONS


                4. SET NONADIABATIC DYNAMICS


                5. GENERATE TRAJECTORIES AND SPECTRUM


                6. SET STATISTICAL ANALYSIS


                7. EXIT
```

```
Select one option (1-7): 5 <ENTER>
```

9. Select the job options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : <ENTER>
 Setting type = 3
```

If for this first option you are asked for "kindjob" instead "type", this means you are using an old version of NEWTON-X. Contact the distributors to get an updated version of the program.

```
 Would you like to apply energy and probability restrictions (y/n)?  (default=n) y
<ENTER>

 nis:  Initial state.
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Array of final states (coma or dash separated, e.g., 2-4 or 2,3,4).
 For trajectories generation only one final state is allowed.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 3 <ENTER>
 Setting nfs = 3
```

The next keywords will allow setting the initial condition restrictions between 3.5 and 4 eV.

If nfs should be given as a 'space separated' list, this means that you are using an old version of NEWTON-X. Contact the distributors to get an updated version of the program.

```
 screen:  Energy restriction
        0 - don't apply any restriction
        1 - use the original energy restriction written in the final_output files
        2 - apply new energy restriction
 The current value of screen  is: 0
 Enter the new value of screen  : 2 <ENTER>
 Setting screen = 2

 e_center:  Center of the energy restriction.
        x     - value of the center of restriction
        ref n - use the vertical excitation of final_output.nis.n file
 The current value of e_center (eV) is: 0.0
 Enter the new value of e_center (eV) : 3.75 <ENTER>
 Setting e_center = 3.75

 e_var:  Width of the energy restriction.
 The current value of e_var (eV) is: 0.5
 Enter the new value of e_var (eV) : 0.25 <ENTER>
 Setting e_var = 0.25

 os_condon:  Oscillator strength:
 -1 - try to read from final_output file.
   x - oscillator strength is always x (Condon approximation).
 The current value of os_condon  is: -1
 Enter the new value of os_condon  : <ENTER>
 Setting os_condon = -1
```

```
prob_kind:  Formula to calculate the transition probabilities:
        A - Einstein-coefficient A (spontaneous emission)
        B - Einstein-coefficient B (induced absorption or induced emission)
        F - Oscillator strength (photoabsorption cross section) ! NX < 1.2
        E - Fluorescence (radiative decay rate)        ! Options for NX > 1.4
        F - Absorption (photoabsorption cross section) !
 The current value of prob_kind  is: F
 Enter the new value of prob_kind  : <ENTER>
 Setting prob_kind = F

 norm:  Normalization of transition intensities:
        local  - Use energy-restricted data set
        global - Use complete data set
 The current value of norm  is: local
 Enter the new value of norm  : <ENTER>
 Setting norm = local

 seed:  Seed for the random number generation
        0 - a default random number seed is used
        1 - a randomized seed is used
        Any other positive integer is used as the random number seed
 The current value of seed  is: 0
 Enter the new value of seed  : <ENTER>
 Setting seed = 0

 Will the trajectories run in a batch system? n <ENTER>

        Processing data: This may take some minutes. Please, wait...
```

In general, you might want to submit this job to a batch system. In this case, answer "y" to "Will the trajectories run in a batch system?". You will be asked the name of the submission script and nxinp will look for it existence in the input directory. Files "pmold" and "pmold-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

10. Exit nxinp.

11. Now the input directory should contain a new directory called TRAJECTORIES, which was created by nxinp.

```
> ls
control.dyn  final_output  intcfl  JOB_NAD  makedir.log  mkd.inp  TRAJECTORIES
```

12. Move to TRAJECTORIES and list the content. You will see that 2 directories named TRAJ1 to TRAJ2 were created. Each one of these directories contains a complete set of input files to run one independent trajectory with different initial conditions (geometry and velocities).

The number of TRAJn directories is controlled by the number of initial conditions in the final_output file and by the energy restrictions imposed. In this case only two trajectories were created because our restriction 3.75 ± 0.25 eV is a too low energy witndow to sample the S_2 state. You can see a summary of the sampling process in makedir.log.

Each accepted initial condition card is split into a geom and veloc files that appear inside TRAJn. The TRAJn structure of directories (see Fig. 7) is very important in NEWTON-X*. Several tools to control the jobs and to perform statistical analysis use it.*

Each TRAJn directory can be copied to a different computer or submitted to a different node of a batch system. They should, however, be grouped again after having finished the dynamics calculations.

3.3 Running NEWTON-X: dynamics simulations

13. Move to TRAJ1 and run this trajectory.

```
> cd TRAJ1
> $NX/moldyn.pl > moldyn.log &
```

14. Move to TRAJ2 and run that trajectory.

If the jobs should be submitted to a batch system you can either go to each TRAJn directory and run

```
> qsub pmold      ! pmold is the submission script
```

or in the TRAJECTORIES directory you can run

```
> $NX/submit.pl
```

In this case, you will be asked the initial and final trajectory number that should be submitted, and the submission of all these trajectories will be done automatically.

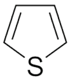
For a discussion about how to visualize the results and how to perform a statistical analysis, see sections 7.5 and 7.6.

4. Initial conditions and spectrum simulation: TD-DFT – Gaussian 09

In this section, we will simulate absorption spectrum with GAUSSIAN09. The system is the five member heterocycle ring **thiophene** (e.g. *1***T**) and ten singlet excited states will be included. The initial geometry distribution will be generated by a Wigner distribution in the ground vibrational state of the ground electronic state. The transition energies and moments will be also computed at TD-DFT(ωB97XD)/6-31G*. Theoretical details about the excited states structure of *1***T** can be found in Refs.[11, 12].

4.1 Molecular structure optimization and geometry file creation

1. Chemical structure of the thiophene molecule:



2. In the TUTORIAL directory (see section 1.3) create a subdirectory called ‘OPT-DFT’

```
> mkdir OPT-DFT <ENTER>
```

3. In the directory ‘OPT-DFT’ prepare the input and run a geometry optimization calculation at DFT(ωB97XD)/6-31G* level with GAUSSIAN 09. To do that, follow specific GAUSSIAN09 documentation (http://www.gaussian.com/g_prod/g09.htm).

4. In the TUTORIAL directory create another subdirectory called THIOPHENE_TDDFT_SPEC:

```
> mkdir THIOPHENE_TDDFT_SPEC <ENTER>
```

5. Once optimized the molecular structure move in the subdirectory called THIOPHENE_TDDFT_SPEC and create a file called ‘thiophene.xyz’ containing the optimized DFT geometry in XYZ format:

```
> cd THIOPHENE_TDDFT_SPEC <ENTER>
> vi thiophene.xyz <ENTER>
```

```
9

    S        0.000000     1.192685     0.000000
    C        1.237296    -0.008625     0.000000
    C        0.713548    -1.267517     0.000000
    C       -0.713772    -1.267466     0.000000
    C       -1.236952    -0.008600     0.000000
```

```
H          2.279067     0.280785     0.000000
H          1.319826    -2.165766     0.000000
H         -1.320853    -2.165283     0.000000
H         -2.278765     0.280547     0.000000
```

6. Convert the optimized geometry (file name ‘thiophene.xyz’) into Newton-X format.

```
> $NX/xyz2nx < thiophene.xyz <ENTER>
```

The Newton-X *geometry file is called ‘geom’ and looks like*

```
 s    16.0    0.00000000    2.25384801    0.00000000   31.97207180
 c     6.0    2.33815059   -0.01629889    0.00000000   12.00000000
 c     6.0    1.34841030   -2.39526000    0.00000000   12.00000000
 c     6.0   -1.34883360   -2.39516362    0.00000000   12.00000000
 c     6.0   -2.33750052   -0.01625164    0.00000000   12.00000000
 h     1.0    4.30681247    0.53060675    0.00000000    1.00782504
 h     1.0    2.49410968   -4.09270461    0.00000000    1.00782504
 h     1.0   -2.49605043   -4.09179187    0.00000000    1.00782504
 h     1.0   -4.30624177    0.53015700    0.00000000    1.00782504
```

The second column contains the atomic number, and the following three columns contain the x, y and z coordinates in atomic units (Bohr). The last column contains the atomic masses, which can be eventually changed to simulate isotopic effects. The file can be freely formatted. The conversion back from Newton-X *format to xyz can be done with.*

```
> $NX/nx2xyz
```

After running this program, the xyz-format geometry is written to file ‘geom.xyz’.

4.2 Normal modes step

7. In the TUTORIAL directory create another subdirectory called ‘FREQ-DFT’ to run normal mode calculation:

```
> mkdir FREQ-DFT <ENTER>
```

8. Move to this directory and copy the ‘thiophene.xyz’ file there.

```
> cd FREQ-DFT <ENTER>
> cp ../THIOPHENE_TDDFT_SPEC/thiophene.xyz . <ENTER>
```

9. Prepare the input and run a normal mode calculation at DFT(ωB97XD)/6-31G* level with Gaussian 09. To do that, follow specific Gaussian 09 documentation (http://www.gaussian.com/g_prod/g09.htm). The Gaussian 09 output containing the normal modes and frequencies can have any name. Here we will suppose it is called “freq.out”.

4.3 Energy and transition moment input

10. The TUTORIAL directory now looks like:

```
> ls <ENTER>
FREQ-DFT OPT-DFT THIOPHENE_TDDFT_SPEC
```

11. Move to the directory 'THIOPHENE_TDDFT_SPEC' and create a new subdirectory called JOB_AD.

```
> cd THIOPHENE_TDDFT_SPEC <ENTER>
> mkdir JOB_AD <ENTER>
```

The directory must be named JOB_AD since NEWTON-X *will search for it.*

12. Move into JOB_AD and prepare two files named, 'basis' and 'gaussian.com' respectively. 'basis' contains the basis set information while 'gaussian.com' contains the TD-DFT Gaussian09 input, like:

```
> cd JOB_AD <ENTER>
> vi basis <ENTER>
  6-31G*
> vi gaussian.com <ENTER>
  %chk=gaussian
  %rwf=gaussian
  %nproc=2
  %mem=400mw
  #TD(NStates=10) 6-31G* wB97XD NoSymm

  thiophene wb97xd/6-31gs

  0       1
    S        0.000000    1.192685    0.000000
    C        1.237296   -0.008625    0.000000
    C        0.713548   -1.267517    0.000000
    C       -0.713772   -1.267466    0.000000
    C       -1.236952   -0.008600    0.000000
    H        2.279067    0.280785    0.000000
    H        1.319826   -2.165766    0.000000
    H       -1.320853   -2.165283    0.000000
    H       -2.278765    0.280547    0.000000
```

For Gaussian 09 calculations it is mandatory that the 'JOB_AD' directory contains the above two files, 'basis' and 'gaussian.com'.

To prepare a GAUSSIAN09 input for single point calculation at TD-DFT(ωB97XD)/6-31G* without symmetry, with ten singlet excited states, and transition moments (oscillator strengths) calculations follow the GAUSSIAN09 documentation (http://www.gaussian.com/g_prod/g09.htm).

Before submitting running any NEWTON-X *job is highly advisable to run the job in JOB_AD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_AD into a test directory and run the* GAUSSIAN09 *job. Check whether the roots and the oscillator strengths are computed and show reasonable values.*

From a general point of view, at this stage we have:
1) optimized the molecular structure;
2) computed the normal modes;
3) converted the optimized structure file 'name.xyz' into NEWTON-X *format 'geom';*
4) created the directory (e.g. THIOPHENE_TDDFT_SPEC) in which the spectrum will be computed using the NEWTON-X *program interfaced with* GAUSSIAN09*.*

4.4 NEWTON-X input

12. Move to directory TUTORIAL/THIOPHENE_TDDFT_SPEC and create a subdirectory called IC-TDDFT.

```
> cd TUTORIAL/THIOPHENE_TDDFT_SPEC <ENTER>
> mkdir IC-TDDFT <ENTER>
```

13. Move to this directory and copy the optimized DFT geometry file in NEWTON-X format (geom) there.

```
> cd IC-TDDFT <ENTER>
> cp ../geom . <ENTER>
```

14. Copy the JOB_AD directory and the normal modes file into IC-TDDFT.

```
> cp -rf ../JOB_AD .
> cp ../../FREQ-DFT/freq.out .
> ls <ENTER>
freq.out geom JOB_AD
! for a further check
> ls JOB_AD/ <ENTER>
basis gaussian.com
```

15. Run the nxinp program and select option 1: Generate initial conditions.

nxinp is the input tool for NEWTON-X*. All input procedures, from the initial condition generation, through the dynamics inputs, to the statistical analysis options are performed with it.*

```
> $NX/nxinp
```

```
                    ============================================================
                                              NEWTON-X
                             Newtonian dynamics close to the crossing seam
                                         www.newtonx.org
                    ============================================================

                                    MAIN MENU


                                    1. GENERATE INITIAL CONDITIONS


                                    2. SET BASIC INPUT


                                    3. SET GENERAL OPTIONS


                                    4. SET NONADIABATIC DYNAMICS


                                    5. GENERATE TRAJECTORIES AND SPECTRUM


                                    6. SET STATISTICAL ANALYSIS


                                    7. EXIT

Select one option (1-7): 1 <ENTER>
```

16. Select the adequate options for the initial condition properties.

nxinp will guide you through a set of questions. Each of them contains a short explanation and the available options. Most of times, the default value is the best option. To accept the default you can just press <ENTER>.

```
...
nact:  Distribution:
        1 - Classic harmonic oscillator
        2 - Q/P-uncorrelated quantum harmonic oscillator (Wigner)
        3 - Q/P-correlated quantum harmonic oscillator
        4 - Pick points from previous dynamics
        5 - Random-velocities generation
        6 - Single point on a previous calculation
 The current value of nact  is: 2
 Enter the new value of nact  : <ENTER>
```

```
Setting nact = 2

numat:  Number of atoms.
The current value of numat  is: 9
Enter the new value of numat  : <ENTER>
Setting numat = 9

npoints:  Number of initial conditions to be generated
The current value of npoints  is: 1
Enter the new value of npoints  : 50 <ENTER>
Setting npoints = 50
```

npoints sets the number of initial conditions that should be generated. This number depends on how many points you want to compute to generate the spectrum and how many trajectories you want to simulate.

In real applications, you will need much more than 50 points. 500 may be a good number.

You can also run several short jobs with few points in each one and merge them afterwards using the tool "merge_initcond.pl". Thus, with 10 computers running 50 points each, you can quickly have 500 points. See in section 4.5 a note about how to split and merge the jobs.

```
file_geom:  File containing the equilibrium geometry.
The current value of file_geom  is: 'geom'
Enter the new value of file_geom  : <ENTER>
Setting file_geom = 'geom'

iprog:  Read vibrational modes from:
       1 - GAMESS output
       2 - TURBOLMOLE output
       3 - COLUMBUS output
       4 - GAUSSIAN output
       5 - MOLDEN file
       6 - DFTB output
       7 - ACES2 output
The current value of iprog is: 2
Enter the new value of iprog  : 4 <ENTER>
Setting iprog = 4

file_nmodes:  File containing the normal modes.
The current value of file_nmodes  is: gaussian.log
Enter the new value of file_nmodes  : freq.out <ENTER>

anh_f:  Multiply harmonic frequencies by this factor.
The current value of anh_f  is: 1
```

```
 Enter the new value of anh_f  : <ENTER>
 Setting anh_f = 1

 chk_e:  Restrict (or do not) the statistical distribution:
        0 - Do not check the energies.
        1 - Check the energies between states NIS and NFS.
 The current value of chk_e  is: 0
 Enter the new value of chk_e  : 1 <ENTER>
 Setting chk_e = 1

The next keywords define the restriction on the statistical distribution.

 nis:  Intial state (State in which the previous dynamics run. Ground state = 1.).
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Final state.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 11 <ENTER>
 Setting nfs = 11
```

The Wigner distribution will be generated in the ground electronic state (nis = 1) and transition energies between the ground and the tenth excited (nfs = 11) states will be computed. Indeed, with nfs = N, transition information for all states below N will also be computed and collected.

```
 kvert:  Center of the restriction:
        0 - Provide the energy for center of restriction
        1 - Use the vertical excitation energy of the equilibrium geometry
 The current value of kvert  is: 1
 Enter the new value of kvert  : <ENTER>
 Setting kvert = 1

 de:  Width of restriction (center +/- de/2).
 The current value of de (eV) is: 0.5
 Enter the new value of de (eV) : 100 <ENTER>
 Setting de = 100
```

This large “de” implies that this restriction will not be used. It can be imposed later on.

```
prog:  Program and method to compute vertical excitation energies:
        1   - COLUMBUS
        2.0 - TURBOMOLE CC2
```

```
       2.1 - TURBOMOLE TDDFT
       2.2 - TURBOMOLE ADC2
       5   - DFTB
       6.5 - GAUSSIAN 09
       9   - DFT-MRCI
       10  - GAMESS MCSCF
       20  - HYBRID ENERGY
The current value of prog  is: 1
Enter the new value of prog  : 6.5 <ENTER> ! Gaussian09 TDDFT
Setting prog = 6.5

iseed:  Random number seed.
        0 - default seed value.
       -1 - set random seed.
       >0 - (integer) set specific value for the seed.
The current value of iseed  is: 0
Enter the new value of iseed  : <ENTER>
Setting iseed = 0

lvprt:  Print level.
       1 - standard level
       2 - debug level
The current value of lvprt  is: 1
Enter the new value of lvprt  : <ENTER>
Setting lvprt = 1
```

17. Exit nxinp.

4.5 Running NEWTON-X: initial conditions

18. Run the initial condition generation program.

```
> $NX/initcond.pl > initcond.log &
```

In general, with chk_e = 1, you might want to submit this job to a batch system. Files "pini" and "pini-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

Splitting and merging jobs:

To split the job among several computers, just prepare the input as it was done until here. Then run

```
> $NX/split_initcond.pl
```

This program will create a directory called INITIAL_CONDITIONS and inside it a series a sequence of subdirectories called I1, I2, ..., each one containing a complete set of input files.

Copy each directory to a different computer and run the job normally.

To merge the jobs, after the calculations, copy the directories I1, I2, ... back to INITIAL_CONDITIONS directory and from inside this directory run

```
> $NX/merge_initcond.pl
```

This program will ask the number of jobs to be merged and it will create a new directory called I_merged with merged results. Move to this directory and proceed with the spectrum simulation as explained below.

4.6 Checking the results

19. Normal job termination is indicated in the end of initcond.log file by:

```
End of initial conditions generation.
          ====================== NEWTON-X ends here ==================
```

20. The main information is written to the final_output file.

21. Each final_ouput.1.# file (# = 1,11 in this case) contains the same 50 geometries and velocities generated according to the Wigner distribution. For instance, final_output.1.2 contains transition energies and oscillator strengths to excitations into state 2 (first excited state S_1), while final_output.1.3 contains the same information to excitations into state 3 (second excited state S_2) and so on.

4.7 Generating the spectrum

22. Go to directory TUTORIAL/THIOPHENE_TDDFT_SPEC and create the directory SPECTRUM.

```
> mkdir SPECTRUM  !This name is not compulsory.
```

23. Move to this directory and copy each final_output files there.

```
> cd SPECTRUM <ENTER>
> cp ../IC_TDDFT/final_output* . <ENTER>
```

24. Run nxinp and select option 5: “Generate Trajectories and Spectrum”.

```
> $NX/nxinp
```

```
                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT
```

```
                                       3. SET GENERAL OPTIONS


                                       4. SET NONADIABATIC DYNAMICS


                                       5. GENERATE TRAJECTORIES AND SPECTRUM


                                       6. SET STATISTICAL ANALYSIS


                                       7. EXIT


Select one option (1-7): 5 <ENTER>
```

25. Go through the next options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : 1 <ENTER>
 Setting type = 1
```

If for this first option you are asked for "kindjob" instead "type", this means you are using an old version of NEWTON-X*. Contact the distributors to get an updated version of the program.*

```
 nis:  Initial state.
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Array of final states (coma or dash separated, e.g., 2-4 or 2,3,4).
 For trajectories generation only one final state is allowed.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 2-11 <ENTER>  !Spectrum including each state
 Setting nfs = 2-11
```

If nfs should be given as a 'space separated' list, this means that you are using an old version of NEWTON-X. Contact the distributors to get an updated version of the program.

```
screen:  Energy restriction
       0 - don't apply any restriction
       1 - use the original energy restriction written in the final_output files
       2 - apply new energy restriction
The current value of screen  is: 0
Enter the new value of screen  : <ENTER>
Setting screen = 0
os_condon:  Oscillator strength:
 -1 - try to read from final_output file.
  x - oscillator strength is always x (Condon approximation).
The current value of os_condon  is: -1
Enter the new value of os_condon  : <ENTER>
Setting os_condon = -1
```

If the oscillator strength is not available but the transition energy is, it is still possible to simulate the spectrum by giving a value to the oscillator strength in os_condon option. In this case, all transitions will be computed with the same oscillator strength. For only one final state this option is not bad, but it can be completely inadequate when more than one final state is selected.

```
prob_kind:  Formula to calculate the transition probabilities:
       A - Einstein-coefficient A (spontaneous emission)
       B - Einstein-coefficient B (induced absorption or induced emission)
       E - Fluorescence (radiative decay rate)
       F - Absorption (photoabsorption cross section)
The current value of prob_kind  is: F
Enter the new value of prob_kind  : <ENTER>
Setting prob_kind = F

norm:  Normalization of transition intensities:
       local  - Use energy-restricted data set
       global - Use complete data set
The current value of norm  is: local
Enter the new value of norm  : <ENTER>  ! with no energy restriction, doesn't
Setting norm = local                    ! matter.

seed:  Seed for the random number generation
       0 - a default random number seed is used
       1 - a randomized seed is used
       Any other positive integer is used as the random number seed
The current value of seed  is: 0
Enter the new value of seed  : <ENTER>
Setting seed = 0
```

```
l_shape:  Line shape:
       gauss   - Normalized Gaussian function.
       lorentz - Normalized Lorentzian function.
The current value of l_shape  is: lorentz
Enter the new value of l_shape  : <ENTER>
Setting l_shape = lorentz

delta:  Phenomenological broadening of the spectrum.
The current value of delta (eV) is: 0.05
Enter the new value of delta (eV) : <ENTER>
Setting delta = 0.05
```

"delta" is possibly the only option that you need to play with. It controls the width of the gaussian curve attributed to each one of the 100 transitions (50 initial conditions × 2 final states).

```
temp:  Temperature.
The current value of temp (K) is: 0
Enter the new value of temp (K) : <ENTER>

nref:  Refraction index.
The current value of nref  is: 1
Enter the new value of nref  : <ENTER>

eps:  Distance between consecutive points in the spectrum.
The current value of eps (eV) is: 0.005
Enter the new value of eps (eV) : <ENTER>
Setting eps = 0.005

kappa:  The range of the spectrum is defined between
Emin-kappa*delta and Emax+kappa*delta, where kappa is an integer.
The current value of kappa  is: 3
Enter the new value of kappa  : <ENTER>
Setting kappa = 3

Processing data: This may take some minutes. Please, wait...
```

26. Exit nxinp.

27. The simulated cross section using a Lorentzian line shape with phenomenological broadening $\delta = 0.05$ eV is written to cross-section.dat. Looking in this file, we see:

```
!      DE/eV     lambda/nm        sigma/A^2         +/-error/A^2
     5.1006    2.4308E+02       0.00477581        0.00049100
```

```
      5.1056   2.4284E+02       0.00491096       0.00053555
      5.1106   2.4260E+02       0.00505910       0.00058786
      5.1156   2.4236E+02       0.00522267       0.00064969
      5.1206   2.4213E+02       0.00540474       0.00072344
      5.1256   2.4189E+02       0.00560926       0.00081227
      5.1306   2.4166E+02       0.00584135       0.00092032
      5.1356   2.4142E+02       0.00610763       0.00105312
      5.1406   2.4119E+02       0.00641675       0.00121801
      5.1456   2.4095E+02       0.00677996       0.00142465
      5.1506   2.4072E+02       0.00721155       0.00168546
      5.1556   2.4048E+02       0.00772885       0.00201555
...
```

28. Run GNUPLOT to visualize the spectrum.

```
> gnuplot
gnuplot> plot 'cross-section.dat' using 1:3 with lines
```

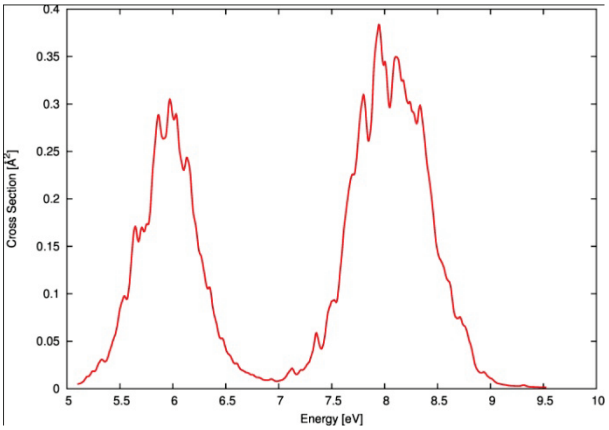


Fig. 2. Photoabsorption cross section in Å^2.molecule^{-1} as a function of the excitation energy in electron-Volts of 1T (ωB97XD/6-31G*).

Your spectrum should not look exactly the one in Fig. 2. This one was done with 300 points instead 50.

5. Surface hopping dynamics: TD-DFT – Gaussian 09

In this section, we will prepare the inputs and run dynamics simulations. The system is thiophene ring (*1***T**) and the dynamics will be initiated in the second excited state (state 3, S_2).

The initial conditions were generated in Chapter 4 of this tutorial. The surface hopping[10] dynamics will be performed at TDDFT level using GAUSSIAN 09. The trajectory time will be 300 fs with 0.5 fs time step.

The TDDFT will use the ωB97XD functional and the 6-31G* basis set. The dynamics will start in S_2, with initial conditions selected in the interval 5.5 to 6.5 eV.

(An energy width of ±0.5 eV allows to generate trajectories belonging to the whole first absorption band, see Fig. 2. This will generate a large number of trajectories and, for practical reasons, only few of them will be computed and analyzed in this tutorial).

5.1 Copying the initial conditions

1. Go to TUTORIAL directory create a subdirectory called THIOPHENE_TDDFT_DYN:

```
> mkdir THIOPHENE_TDDFT_DYN <ENTER>
```

2. Move to this directory and copy the initial conditions file for the first excited state there.

```
> cd THIOPHENE_TDDFT_DYN
> cp ../THIOPHENE_TDDFT_SPEC/IC_TDDFT/final_output.1.3 final_output ! note the new
                                                                      name
```

3. Create a subdirectory called JOB_AD.

4. check the content of the THIOPHENE_TDDFT_DYN directory:

```
> ls <ENTER>
final_output  geom  JOB_AD
```

The directory <u>must</u> be named JOB_AD since NEWTON-X *will search for it.*

5.2 Preparing electronic structure input

5. Move to JOB_AD and create the file 'basis' and the input file 'gaussian.com' for single point TDDFT/ωB97XD/6-31G* calculation. Use C_1 point group (no symmetry) and request the calculation of three excited states specifying we are interesting in the second root (i.e. S_2).

```
> ls <ENTER>
basis  gaussian.com

> cat gaussian.com

%chk=gaussian
%rwf=gaussian
%nproc=2
%mem=400mw
#TD(NStates=3,Root=2) 6-31G* wB97XD scf(MaxCycle=512) NoSymm

thiophene wb97xd/6-31gs

0       1
    S        0.000000    1.192685    0.000000
    C        1.237296   -0.008625    0.000000
    C        0.713548   -1.267517    0.000000
    C       -0.713772   -1.267466    0.000000
    C       -1.236952   -0.008600    0.000000
    H        2.279067    0.280785    0.000000
    H        1.319826   -2.165766    0.000000
    H       -1.320853   -2.165283    0.000000
    H       -2.278765    0.280547    0.000000
```

Before submitting running any Newton-X job is highly advisable to run the job in JOB_AD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_AD into a test directory and run the GAUSSIAN 09 job. Check whether the three roots are computed and show reasonable values.

5.3 NEWTON-X input

6. Run nxinp.

```
> $NX/nxinp
```

```
                ============================================================
                                          NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT
```

```
Select one option (1-7): 2 <ENTER>
```

7. Select option 2: “set basic input” and answer the questions to set the dynamics options.

```
 nat:  Number of atoms.
 The current value of nat  is: 9
 Enter the new value of nat  : <ENTER>
 Setting nat = 9

 nstat:  Number of states.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>  ! 3 states will be computed
 Setting nstat = 3


 nstatdyn:  Initial state (1 - ground state).
 The current value of nstatdyn  is: 3
 Enter the new value of nstatdyn  : <ENTER> ! dynamics will start in state 3 (e.g.
S2)
 Setting nstatdyn = 3

 dt:  Time step for the classical equations.
 The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 tmax:  Total duration of the trajectory.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) :300 <ENTER>
 Setting tmax = 300

prog:  Quantum chemistry program and method
        0    - ANALYTICAL MODEL
        1    - COLUMBUS
        2.0 - TURBOMOLE CC2
        2.1 - TURBOMOLE TDDFT
        2.2 - TURBOMOLE ADC2
        3    - ACES2 EOM-CCSD (in development)
        4    - MOPAC (in development)
        5    - DFTB
        6.0 - GAUSSIAN(03) CASSCF
        6.5 - GAUSSIAN(09) TDDFT
        7    - TINKER
        8    - DFTB+
        10.0    - GAMESS
        10.1    - GAMESS ARBITRARY ADIABATIC METHOD
        20   - HYBRID GRADIENT
 The current value of prog  is: 1
 Enter the new value of prog  : 6.5 <ENTER>
 Setting prog = 6.5

 thres: Threshold to initiate nonadiabatic coupling calculation.
          0 - Only adiabatic dynamics.
        100 - Only nonadiabatic dynamics
 The current value of thres (eV) is: 100
 Enter the new value of thres (eV) : <ENTER>
 Setting thres = 100

 Would you like to generate internal coordinate file?  (default=y) y <ENTER>
```

8. In the main nxinp menu you have the option to set more general options if you select either option 3 or option 4. Normally, this is not necessary and the default values are good enough.

9. Select option 5: “generate trajectories and spectrum”.

```
                ============================================================
                                          NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 5 <ENTER>
```

10. Select the job options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : <ENTER>
 Setting type = 3
```

If for this first option you are asked for “kindjob” instead “type”, this means you are using an old version of NEWTON-X. Contact the distributors to get an updated version of the program.

```
 Would you like to apply energy and probability restrictions (y/n)?  (default=n) y
<ENTER>

 nis:  Initial state.
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Array of final states (coma or dash separated, e.g., 2-4 or 2,3,4).
 For trajectories generation only one final state is allowed.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 3 <ENTER>
 Setting nfs = 3
```

The next keywords will allow setting the initial condition restrictions between 5.5 and 6.5 eV.

If nfs should be given as a 'space separated' list, this means that you are using an old version of Newton-X. Contact the distributors to get an updated version of the program.

```
 screen:  Energy restriction
        0 - don't apply any restriction
        1 - use the original energy restriction written in the final_output files
        2 - apply new energy restriction
 The current value of screen  is: 0
 Enter the new value of screen  : 2 <ENTER>
 Setting screen = 2

 e_center:  Center of the energy restriction.
        x      - value of the center of restriction
        ref n - use the vertical excitation of final_output.nis.n file
 The current value of e_center (eV) is: 0.0
 Enter the new value of e_center (eV) : 6.0 <ENTER>
 Setting e_center = 6.0

 e_var:  Width of the energy restriction.
 The current value of e_var (eV) is: 0.5
 Enter the new value of e_var (eV) : <ENTER>
 Setting e_var = 0.5

 os_condon:  Oscillator strength:
 -1 - try to read from final_output file.
   x - oscillator strength is always x (Condon approximation).
 The current value of os_condon  is: -1
 Enter the new value of os_condon  : <ENTER>
 Setting os_condon = -1

prob_kind:  Formula to calculate the transition probabilities:
        A - Einstein-coefficient A (spontaneous emission)
        B - Einstein-coefficient B (induced absorption or induced emission)
        E - Fluorescence (radiative decay rate)
        F - Absorption (photoabsorption cross section)
 The current value of prob_kind  is: F
 Enter the new value of prob_kind  : <ENTER>
 Setting prob_kind = F

 norm:  Normalization of transition intensities:
        local  - Use energy-restricted data set
        global - Use complete data set
 The current value of norm  is: local
 Enter the new value of norm  : <ENTER>
 Setting norm = local

 seed:  Seed for the random number generation
        0 - a default random number seed is used
        1 - a randomized seed is used
        Any other positive integer is used as the random number seed
 The current value of seed  is: 0
 Enter the new value of seed  : <ENTER>
 Setting seed = 0

 Will the trajectories run in a batch system? n <ENTER>

        Processing data: This may take some minutes. Please, wait...
```

In general, you might want to submit this job to a batch system. In this case, answer "y" to "Will the trajectories run in a batch system?". You will be asked the name of the submission script and nxinp will look for it existence in the input directory. Files "pmold" and "pmold-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

11. Exit nxinp.

12. Now the input directory should contain a new directory called TRAJECTORIES, which was created by nxinp.

```
> ls
control.dyn   final_output   geom   intcfl   JOB_AD   makedir.log   mkd.inp   TRAJECTORIES
```

13. Move to TRAJECTORIES and list the content. You will see that 176 directories named TRAJ1, TRAJ2, …, TRAJ176 were created. Each one of these directories contains a complete set of input files to run one independent trajectory with different initial conditions (geometry and velocities).

The number of TRAJn directories is controlled by the number of initial conditions in the final_output file and by the energy restrictions imposed.

Each accepted initial condition card is split into a geom and veloc files that appear inside TRAJn. The TRAJn structure of directories (see Fig. 7) is very important in NEWTON-X. *Several tools to control the jobs and to perform statistical analysis use it.*

Each TRAJn directory can be copied to a different computer or submitted to a different node of a batch system. They should, however, be grouped again after having finished the dynamics calculations.

5.4 Running NEWTON-X: dynamics simulations

14. Move to TRAJ1 and run this trajectory.

```
> cd TRAJ1
> $NX/moldyn.pl > moldyn.log &
```

15. Move to TRAJ2 and run that trajectory; redo the same until TRAJ10.

If the jobs should be submitted to a batch system you can either go to each TRAJn directory and run

```
> qsub pmold        ! pmold is the submission script
```

or in the TRAJECTORIES directory you can run

```
> $NX/submit.pl
```

In this case, you will be asked the initial and final trajectory number that should be submitted, and the submission of all these trajectories will be done automatically.

5.5 Checking the results

16. For each TRAJn, the dynamics output is in three directories:

a. RESULTS: the dynamics information;

b. INFO_RESTART: stored information to restart the trajectory;

c. DEBUG: error messages, convergence information and additional electronic structure files like molecular orbital files.

and the content of each TRAJn directory looks like:

```
> ls TRAJ1/
  control.dyn  DEBUG  geom  INFO_RESTART  intcfl JOB_AD  moldyn.log   RESULTS
veloc
```

17. Go to directory TRAJ1/RESULTS.

```
> cd TRAJ1/RESULTS
```

18. If it finished with error, nx.log should finish like:

```
moldyn.pl:   ::ERROR::   step =      123 time = 61.5 fs
Trying to save TEMP directory to ../DEBUG
```

In this example, the job finished with error after 123 steps (61.5 fs). The last GAUSSIAN09 calculation was written into TRAJ1/DEBUG/TEMP and you can check what was wrong. More error messages are also written to DEBUG/runnx.error file.

19. If the job ended normally, the file nx.log should finish with:

```
moldyn.pl: End of dynamics
          ====================== NEWTON-X ends here ==================
```

20. Plot the energies (Fig. 8):

```
> $NX/plot
Number of states: 3
```

The energy information used by plot program is in the file RESULTS/en.dat. It looks like:

```
 !Time(fs)  E0 (au)       E1 (au)      E2 (au)     E(current)(au)  Etot (au)
     0.00   -552.894216   -552.674187   -552.668582   -552.668582   -552.649141
     0.50   -552.900033   -552.680213   -552.674932   -552.674932   -552.649355
     1.00   -552.905180   -552.685933   -552.680921   -552.680921   -552.649487
     1.50   -552.907398   -552.689155   -552.684274   -552.684274   -552.649503
     2.00   -552.906227   -552.689465   -552.684475   -552.684475   -552.649437
     2.50   -552.902523   -552.687756   -552.682277   -552.682277   -552.649348
     3.00   -552.897612   -552.685315   -552.678932   -552.678932   -552.649260
     3.50   -552.892784   -552.683324   -552.675698   -552.675698   -552.649198
     4.00   -552.888995   -552.682629   -552.673522   -552.673522   -552.649162
     4.50   -552.886775   -552.683658   -552.672935   -552.672935   -552.649155
     5.00   -552.886234   -552.686446   -552.674047   -552.674047   -552.649172
     5.50   -552.887131   -552.690695   -552.676617   -552.676617   -552.649199
     6.00   -552.888979   -552.695879   -552.680169   -552.680169   -552.649241
     6.50   -552.891112   -552.701298   -552.684037   -552.684037   -552.649291
...
```

If you give one more state (4, in this example), the total energy is also printed.

You can run plot program for a sequence of trajectories either to see them on screen or only to generate pictures. To do so, go to TRAJECTORIES directory and run:

```
> $NX/plotall
```

In GNUPLOT the basic script (plot.gpi) can be:

```
set terminal postscript enhanced color solid
set xlabel 'time / fs'
set ylabel 'Energy / Hertree'
set key right bottom
```

```
plot\
'en.dat' u 1:2 t 'S_{0}'   w p pt 6 ps 2, \
'en.dat' u 1:3 t 'S_{1}'   w p pt 6 ps 2, \
'en.dat' u 1:4 t 'S_{2}'   w p pt 6 ps 2, \
'en.dat' u 1:5 t 'current' w p pt 7 lt 7 ps 2
```

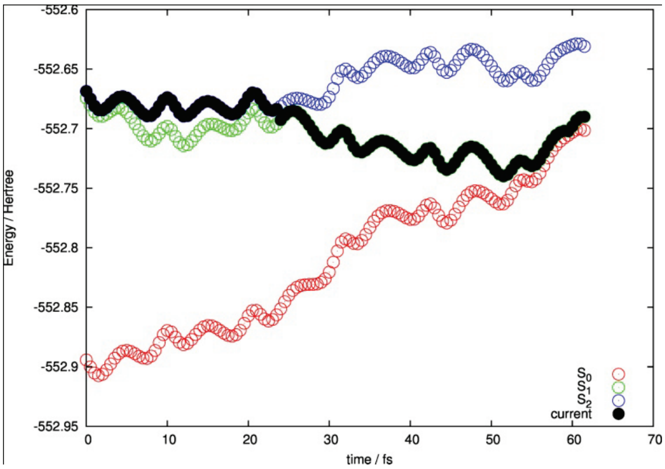


Fig. 3. Potential energies of the three states (S_0, S_1 and S_2) for TRAJ1, as a function of time plotted with GNUPLOT. The black circles indicate the current state of the system at each time step.

21. Visualize the dynamics:

```
> molden dyn.mld &
```

dyn.mld file is a simple multiple-xyz format file with all geometries along the trajectory. It can be opened by most of programs for dynamics visualization. One good option is VMD (www.ks.uiuc.edu/Research/vmd), which allow a quite complete control of the time line and excellent rendering quality.

To save disk space, dyn.mld is not automatically created starting from NEWTON-X version 1.2.5. In these new versions, you can get the xyz file by moving to the RESULTS directory and running

```
> $NX/dynout2xyz.pl
```

A file called dyn.xyz (multiple xyz format) is created and can be visualized with programs like Molden or VMD.

22. Geometry, velocities, energies and other quantities are written in a more verbose way in file dyn.out.

The first time-step looks like:

```
      *** Molecular Dynamics ***

Output printed at each kt=   1 steps ( 0.50 fs).

STEP         0    Molecular dynamics on state  3    TIME =        0.00 fs
Type of dyn. =    2     N. of hoppings =    0    N. of rejec. hoppings =    0

  Initial geometry:
```

```
 s    16.0   -0.01344068    2.19466790   -0.00363459   31.97207180
 c     6.0    2.40677418    0.05835506   -0.07775881   12.00000000
 c     6.0    1.36330143   -2.41575449    0.06668980   12.00000000
 c     6.0   -1.44655585   -2.34258424   -0.07326877   12.00000000
 c     6.0   -2.27564538    0.01234952    0.03977154   12.00000000
 h     1.0    4.17763770    0.69542380    0.53765428    1.00782504
 h     1.0    2.43927573   -3.87923298   -0.16835954    1.00782504
 h     1.0   -2.52844608   -4.11962585    0.27288499    1.00782504
 h     1.0   -4.22150801    0.40278622    0.04398443    1.00782504

  Initial velocity:
   -0.00008485    0.00001268   -0.00004137
   -0.00023103   -0.00003321    0.00016062
    0.00023174    0.00001566   -0.00023973
   -0.00011618   -0.00044362   -0.00007019
    0.00029570    0.00058595   -0.00009650
    0.00014906   -0.00028854    0.00109718
    0.00106823   -0.00159003    0.00048862
   -0.00002047    0.00060141    0.00140778
   -0.00063164   -0.00057783    0.00158049

     Time    Etot          Ekin       Epot E0,      E1, ...
%     0.0   -552.649141      0.019442   -552.894216   -552.674187   -552.668582
Etot variation =   0.000000 au    Worse conservation =   0.000000 au

 Wave function state  1:         0.00000000000000         0.00000000000000
 Wave function state  2:         0.00000000000000         0.00000000000000
 Wave function state  3:         1.00000000000000         0.00000000000000

 ------------
...
```

The last time-step looks like:

```
STEP      123    Molecular dynamics on state  2    TIME =      61.50 fs
Type of dyn. =    4    N. of hoppings =    1    N. of rejec. hoppings =    0

  New geometry:
 s    16.0    0.70340260    3.07317235   -0.05504104   31.97207180
 c     6.0    2.29325059    0.25270304   -0.16881052   12.00000000
 c     6.0    0.91275966   -2.38661106    0.23034902   12.00000000
 c     6.0   -1.57197537   -3.13649503   -0.00236842   12.00000000
 c     6.0   -3.42722163   -1.43306138    0.14777090   12.00000000
 h     1.0    4.12022179   -0.25329381   -0.44739620    1.00782504
 h     1.0    2.12132046   -4.26057143    0.63263093    1.00782504
 h     1.0   -1.83590406   -5.26397036   -0.16118117    1.00782504
 h     1.0   -5.30394868   -0.89750746    0.23519054    1.00782504

  New velocity:
    0.00036909    0.00043416    0.00004654
   -0.00024132    0.00006964   -0.00064342
    0.00032666   -0.00049094    0.00045817
   -0.00001929   -0.00003415   -0.00032210
   -0.00071953   -0.00073804    0.00044689
    0.00076679    0.00128969    0.00193094
   -0.00311104    0.00164735   -0.00003364
   -0.00068100   -0.00061212    0.00012677
   -0.00087992   -0.00184894   -0.00239036

     Time    Etot          Ekin       Epot E0,      E1, ...
%      61.50   -552.627499      0.062561   -552.701426   -552.690060   -552.630923
Etot variation =   0.003809 au    Worse conservation =   0.021642 au

 Wave function state  1:         0.00000000000000         0.00000000000000
 Wave function state  2:         0.57219705163435        -0.82011522798407
 Wave function state  3:         0.00115939315479        -0.00045026337002
```

```
------------
```

"Etot variation" gives $E_{tot}(t)$-$E_{tot}(t-\Delta t)$. When this quantity is larger than a threshold, the trajectory breaks with "Etot_jump" error message. This threshold can be controlled with the keyword "Etot_jump" in the advanced input.

"Worse conservation" gives MAX[$E_{tot}(t)$-$E_{tot}(0)$] until time t. When this quantity is larger than a threshold, the trajectory breaks with "Etot_drift" error message. This threshold can be controlled with the keyword "Etot_drift" in the advanced input.

In surface hopping approach, the time-dependent wavefunction is written as:[10]

$$\Psi(t) = \sum_{n=1}^{Nstat} \chi_n(t)\psi_i(\mathbf{R}(t)),$$

where ψ_i is the electronic wavefunction (in the adiabatic representation, in this example). The time-dependent coefficients χ_n are complex numbers: The real and imaginary parts of them are given in dyn.out. For example, the line in the last time-step means

```
Wave function state  2:         0.57219705163435        -0.82011522798407
```

$$\mathrm{Re}(\chi_0) = 0.572197 \ \ and \ \ \mathrm{Im}(\chi_0) = -0.82011$$

The normalization of these coefficients is an important check of the quality of the dynamics and it will be discussed below.

23. Energy information is also written in a different format in file typeofdyn.log. It looks like:

```
...
Time = 22.50 Threshold=****  PES = 3  DE_inf =  0.39  DE_sup = -----  Type = 2   Next type = 2
Time = 23.00 Threshold=****  PES = 3  DE_inf =  0.37  DE_sup = -----  Type = 2   Next type = 2
Time = 23.50 Threshold=****  PES = 3  DE_inf =  0.34  DE_sup = -----  Type = 2   Next type = 2
Time = 24.00 Threshold=****  PES = 2  DE_inf =  4.32  DE_sup =  0.31  Type = 2   Next type = 4
Time = 24.50 Threshold=****  PES = 2  DE_inf =  4.26  DE_sup =  0.28  Type = 4   Next type = 4
Time = 25.00 Threshold=****  PES = 2  DE_inf =  4.19  DE_sup =  0.26  Type = 4   Next type = 4
...
```

At time 23.50 fs the molecule was on surface 3 (S_2). The energy gap to S_1 DE_inf was 0.34 eV. At time 24.00 fs, the molecule was on surface 2 (S_1), therefore a surface hopping took place. The energy gap with S_0 was 4.32eV and with S_2 was 0.31eV.

24. Gradients, oscillator strengths and non-adiabatic coupling vectors and log information are written to nx.log (and also in the standard output).

25. The information about the integration of the time dependent Schrödinger equation is in sh.out file. It looks like:

```
...
  substep      step   surf     populations       Total pop
...
|v.h|=      0.000000     0.000000    -0.001959
      941        48      3     0.000000000     1.000000000
                               0.103439012
                               0.896560988
...
```

NEWTON-X *keeps two related time lines. The first is composed by the steps used to integrate the Newton's equations. For step n, the time is n.Δt (Δt = 0.5 fs in this example). The second* *time line is counted in substeps, which are used to integrate the time-dependent Schrödinger equation. The number of substeps per step is defined by keyword "ms", which can be changed in the advance input. The default is ms = 20. At the end of a trajectory with N steps, we will have N.ms substeps.*

At the substep 941, still during step 48, with the molecule still on surface 3 (S_2), the population of the second excited state (S_2) was:

$$|\chi_2|^2 = 0.896560988\,.$$

The population is defined as the diagonal terms of the matrix $\chi_i \chi_j^*$*. These terms are written in sh.out. Note that*

$$\sum_{n=1}^{Nstat} |\chi_n|^2 = 1.$$

The normalization factor is also printed there ("Total pop"). When it deviates more than "pop_dev" (a keyword defined in the advance input, whose default is 0.05), the job breaks with an error message.

The inner products of the nuclear velocity and the non-adiabatic coupling vectors, which are proportional to the hopping probability, are also printed:

$$\mathbf{v}\cdot\mathbf{h}_{10} = 0.000000,$$
$$\mathbf{v}\cdot\mathbf{h}_{20} = 0.000000,$$
$$\mathbf{v}\cdot\mathbf{h}_{21} = -0.001959.$$

26. The hopping probability is written in tprob file. It looks like:

```
  random number    substep     step   probabilities
...
     0.511167348        945       48     0.000000000     0.001554715     0.000000000
     0.001460153        946       48     0.000000000     0.001609113     0.000000000
     0.834699438       1068       54     0.000000000     0.000000000     0.000046495
     0.894276913       1069       54     0.000000000     0.000000000     0.000111712
...
```

In the substep 945 (during step 48), the hopping probabilities were:

$$P_{2\to 0} = 0.000000000,$$
$$P_{2\to 1} = 0.001554715,$$
$$P_{2\to 2} = 0.000000000.$$

These probabilities were compared with the random number 0.511167348 to decide whether a hopping occurs or not. Since the random number was larger than the probabilities, the molecule stays on S_2 surface in the next substep (946).

In the substep 946, the random number (0.001460153) was smaller than $P_{2\to 1}$ (0.001609113), therefore the molecules hops to surface S_1.

In the fewest-switches approach under adiabatic representation, the hopping probability from state j to state i is given by:

$$P_{j\to i}(t)=\begin{cases}-2\dfrac{\mathrm{Re}\left(\chi_i^{*}\chi_j e^{i\gamma_{ji}}\right)}{\left|\chi_{jj}\right|^{2}}\mathbf{v}(t)\cdot\mathbf{h}_{ji}(t)\delta t, & \text{if } P_{j\to i}>0.\\ 0, & \text{if } P_{j\to i}\le 0.\end{cases}$$

In this equation, $\delta t = \Delta t/ms$. Due to this definition, the probability becomes zero for some time intervals (not reported in the above example…have a look by yourself in the tprob file!).

27. Points close to conical intersections (energy gaps smaller than "e_ci", which is keyword defined in advanced input. The default is 0.2 eV) are written to "report.ci" file.

5.6 Statistical analysis of the trajectories

28. After running your trajectories, you can perform a series of statistical analysis on them.

29. Go to TRAJECTORIES directory, run nxinp, and select option 6: "Set statistical analysis".

```
                  ============================================================
                                        NEWTON-X
                           Newtonian dynamics close to the crossing seam
                                      www.newtonx.org
                  ============================================================

                                  MAIN MENU


                                  1. GENERATE INITIAL CONDITIONS


                                  2. SET BASIC INPUT


                                  3. SET GENERAL OPTIONS


                                  4. SET NONADIABATIC DYNAMICS


                                  5. GENERATE TRAJECTORIES AND SPECTRUM


                                  6. SET STATISTICAL ANALYSIS


                                  7. EXIT


Select one option (1-7): 6 <ENTER>
```

30. First, perform a general diagnostic of errors.

```
        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 1): 1 <ENTER>
```

```
 ad:  Path to the trajectories. The default is the current directory.
 The current value of ad  is: /home/TUTORIAL/THIOPHENE_TDDFT_DYN_S2/TRAJECTORIES

 Enter the new value of ad  : <ENTER>
 Setting ad = /home/TUTORIAL/THIOPHENE_TDDFT_DYN_S2/TRAJECTORIES

 itraj:  Initial trajectory to be checked.
 The current value of itraj  is: 1
 Enter the new value of itraj  : <ENTER>
 Setting itraj = 1

 ftraj:  Final trajectory to be checked. There is not sensible default value.
 The current value of ftraj  is:
 Enter the new value of ftraj  : 10 <ENTER>  ! We run 10 trajs out of 176 generated
 Setting ftraj = 10

 tmin:  Initial time. It must be common to all trajectories.
 The current value of tmin (fs) is: 0.0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0.0

 Etot_dev:  Maximal allowed total energy variation.
 The current value of Etot_dev (eV) is: 0.5
 Enter the new value of Etot_dev (eV) : <ENTER>
 Setting Etot_dev = 0.5

 pop_dev:  Maximal allowed variation in the norm of the adiabatic population.
 The current value of pop_dev  is: 0.1
 Enter the new value of pop_dev  : 0.05 <ENTER>
 Setting pop_dev = 0.05
```

31. Exit nxinp and run:

```
> $NX/diagnostic.pl
```

32. The result is the file diag.log. It contains a summary of what happened to the trajectories. It looks like:

```
DIAGNOSTIC OF NEWTON-X TRAJECTORIES

Diagnostic of trajectories 1 to 10

Path to the trajectories:  /home/TUTORIAL/THIOPHENE_TDDFT_DYN/TRAJECTORIES
Initial time (fs):         0.0
Max. Energy deviation(eV): 0.5
Max. population deviation: 0.1
Number of states:          3
Normal termination:        0
Error termination:         10


TRAJECTORY 1:
moldyn.pl:   ::ERROR::   step =      123 time = 61.5 fs
Energy conservation failed at 61.50 fs
Suggestion: make analysis until time (fs): 61.50

TRAJECTORY 2:
moldyn.pl:   ::ERROR::   step =      110 time = 55 fs
Energy conservation failed at 55.00 fs
Suggestion: make analysis until time (fs): 55.00
...

Maximum time reached (fs): 192.00

Distribution of trajectories:
Number of trajs finished between 0.0 and 48 fs: 1
Number of trajs finished between 48 and 96 fs: 8
Number of trajs finished between 96 and 144 fs: 0
Number of trajs finished between 144 and 192 fs: 1

...
```

In this case study all the 10 trajectories finished in error.

1 trajectory in the time interval between 0.0 and 48.0 fs, 8 trajectories in the 48-96 fs interval and 1 trajectory in the 144-192 fs range.

One of the most often reason for finishing in error is the connected with the adequacy of electronic structure method to compute the energies and gradients for some specific nuclear geometry.

For instance, dynamics trajectories performed with TD-DFT will crash as soon as the molecule enters in a region of strong multireference character (e.g. S_0/S_1 crossing seam). Trajectories performed with CASSCF may crash if the molecule dissociates and the necessary anti-bonding orbitals are not in the active space.

The dynamics information until these critical points, however, usually is correct and can be useful for the global statistical analysis.

For each trajectory, diagnostic.pl program prints until which time the calculation is reliable according to the criteria defined in its input. The log file, for example, says that for trajectory 1, data until 61.50 fs can be used.

It is not mandatory to run diagnostic.pl before the statistical analysis but it is quite advisable.

If the statistical analysis program finds the diag.log file, it will automatically use the "suggested time" to decide which data to use from each trajectory. You can change these suggested times in diag.log if you want.

33. Run nxinp and select option 6: "Set statistical analysis".

```
                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 6 <ENTER>
```

34. Make the input for statistical analysis. Start making the "wave function analysis".

```
        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 2): 2 <ENTER>

 itrj:  Initial trajectory to be analysed.
 The current value of itrj  is: 1
 Enter the new value of itrj  : <ENTER>
 Setting itrj = 1

 jtrj:  Final trajectory to be analysed.
 The current value of jtrj  is: 10
 Enter the new value of jtrj  : 10 <ENTER>
 Setting jtrj = 10

 tmin:  Initial time for the analysis.
 The current value of tmin (fs) is: 0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0

 tmax:  Final time for the analysis.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : 300 <ENTER>
 Setting tmax = 300

 dt:  Time step in which outputs are written.
        If TRAJ1/control.dyn is found, default is DT*KT,
        otherwise default is 0.5 fs.
 The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 proptype:  Kind of properties to be analysed.
        1. Energy.
        2. Wave function.
        3. Internal coordinates.
        4. Internal forces (only for COLUMBUS dynamics).
        5. Velocity autocorrelation.
 The current value of proptype  is: 1
 Enter the new value of proptype  : 2 <ENTER>

 nstat:  Number of states to be analysed.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>
 Setting nstat = 3

 complete_data:  Complete data for broken trajectories.
        0. Do not complete data. (default)
        Other positive value. If the last time in the trajectory
        is larger than complete_data, repeat last set of data until
        tmax. Neglect trajectories whose last time is
        smaller than complete_data.(Only for proptype = 2.)
 The current value of complete_data  is: 0
 Enter the new value of complete_data  : <ENTER>
 Setting complete_data = 0
```

35. Exit nxinp and run:

```
> $NX/analysis.pl > analysis.log &
```

analysis.pl program makes averages over trajectories for several different properties.

36. If everything runs normally, anaysis.log ends with:

```
analysis: End of analysis
```

```
                ===================== NEWTON-X ends here =================
```

37. Go to the directory ANALYSIS and look at file prop.2. It looks like:

```
!Traj Time/fs SA SB SA.SB  a0       a1       a2
   1    0.00  3  3  3.3   0.0000   0.0000   1.0000
   1    0.50  3  3  3.3   0.0000   0.0048   0.9952
   1    1.00  3  3  3.3   0.0000   0.0464   0.9536
...
   1   24.00  3  3  3.3   0.0000   0.2126   0.7874
   1   24.50  3  2  3.2   0.0000   0.3576   0.6424
   1   25.00  2  2  2.2   0.0000   0.5168   0.4832
...
```

Trajectory 1 starts at time 0 fs in surface 3 (given by SA) and ends this time stem still on surface 3 (given by SB). In this step all population is in surface 3 ($a2 = |\chi_{22}|^2 = 1.0000$ *).*

At time 24.50 fs, TRAJ1 starts on surface 3 and ends on surface 2, meaning that a hopping between S$_2$ *and S*$_1$ *took place. Now* $a2 = |\chi_{22}|^2 = 0.6424$.

This description continues for all trajectories until TRAJ10, for which only data until 62.0 fs was used (because of the instruction in diag.log).

38. Look at mean_value.2.

```
!Time/fs  N    N0      <a0>    da0     N1      <a1>    da1     N2      <a2>    da2
   0.00  10   0.000   0.000   0.000   0.000   0.000   0.000   1.000   1.000   0.000
   0.50  10   0.000   0.000   0.000   0.000   0.002   0.003   1.000   0.998   0.003
   1.00  10   0.000   0.000   0.000   0.000   0.021   0.031   1.000   0.979   0.031
   1.50  10   0.000   0.000   0.000   0.000   0.077   0.146   1.000   0.923   0.146
...
  24.00  10   0.000   0.000   0.000   0.700   0.746   0.394   0.300   0.254   0.394
  24.50  10   0.000   0.000   0.000   0.900   0.790   0.351   0.100   0.210   0.351
  25.00  10   0.000   0.000   0.000   0.900   0.835   0.327   0.100   0.165   0.327
  25.50  10   0.000   0.000   0.000   0.900   0.859   0.318   0.100   0.141   0.318
```

This file contains the average over all trajectories as a function of time.

A time 1 fs, 10 trajectories were used to compute the averages. The fraction of trajectories on state S$_2$ *was N2 = 1 and the average adiabatic population was* $\left\langle |\chi_{22}|^2 \right\rangle = 0.979$ *with standard deviation da2=0.031.*

At time 24.50fs, 10 trajectories were used to compute the averages. The fraction of trajectories on state S$_2$ *was N2 = 0.1 and the average adiabatic population was* $\left\langle |\chi_{22}|^2 \right\rangle = 0.210$ *with standard deviation da2=0.351, while the fraction of trajectories on state S*$_1$ *was N1 = 0.9 and the average adiabatic population was* $\left\langle |\chi_{11}|^2 \right\rangle = 0.790$ *with standard deviation da1=0.351.*

39. Go to TRAJECTORIES directory, run nxinp again, select Statistical analysis option and choose analysis of "Internal coordinates".

```
                    ============================================================
                                              NEWTON-X
                                Newtonian dynamics close to the crossing seam
                                          www.newtonx.org
```

```
                ============================================================

                                MAIN MENU

                                1. GENERATE INITIAL CONDITIONS

                                2. SET BASIC INPUT

                                3. SET GENERAL OPTIONS

                                4. SET NONADIABATIC DYNAMICS

                                5. GENERATE TRAJECTORIES AND SPECTRUM

                                6. SET STATISTICAL ANALYSIS

                                7. EXIT

Select one option (1-7): 6 <ENTER>

        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 2): <ENTER>

 Would you like to delete old input files? (prop.inp) y <ENTER>

 itrj:  Initial trajectory to be analysed.
 The current value of itrj  is: 1
 Enter the new value of itrj  : <ENTER>
 Setting itrj = 1

 jtrj:  Final trajectory to be analysed.
 The current value of jtrj  is: 10
 Enter the new value of jtrj  : 10 <ENTER>
 Setting jtrj = 10

 tmin:  Initial time for the analysis.
 The current value of tmin (fs) is: 0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0

 tmax:  Final time for the analysis.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : 300 <ENTER>
 Setting tmax = 300

 dt:  Time step in which outputs are written.
        If TRAJ1/control.dyn is found, default is DT*KT,
        otherwise default is 0.5 fs. The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 proptype:  Kind of properties to be analysed.
        1. Energy.
        2. Wave function.
        3. Internal coordinates.
        4. Internal forces (only for COLUMBUS dynamics).
        5. Velocity autocorrelation.
 The current value of proptype  is: 1
 Enter the new value of proptype  : 3 <ENTER>

 nstat:  Number of states to be analysed.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>
 Setting nstat = 3

 nic:  Number of internal coordinates to be analised (Maximum = 100).
 The current value of nic  is: 1
 Enter the new value of nic  : 1 <ENTER>
 Setting nic = 1
```

The internal coordinates are defined in the intcfl file in the JOB_NAD directory. The coordinate number one in this example corresponds to the CS stretching.

```
 iclist:  Array with the number of internal coordinates to be analysed.
        Example: if the stretch corresponds coordinate 1 and the torsion to 12,
        and one wants to analyse both, iclist = 1,12 and nic = 2.
 The current value of iclist  is: 1
 Enter the new value of iclist  : <ENTER>
 Setting iclist = 1

 bmat:  Internal coordinates options
        0. Get the internal coord. from output files. COLUMBUS only.
        1. Run cart2int.x program from COLUMBUS to get the internal
 coordinates. In this case, an intcfl file with definition
 of the internal coordinates is required.
 The current value of bmat  is: 0
 Enter the new value of bmat  : <ENTER>
 Setting bmat = 0
```

40. Exit nxinp and run:

```
> $NX/analysis.pl > analysis.log &
```

analysis.pl program makes averages over trajectories for several different properties.

41. If everything runs normally, anaysis.log ends with:

```
analysis: End of analysis
          ====================== NEWTON-X ends here ==================
```

42. Go to the directory ANALYSIS and look at file prop.3. It looks like:

```
! Traj  Time/fs R_CN (Angstrom)
   1     0.00    1.7087400
   1     0.50    1.7079124
   1     1.00    1.7071965
...
   4    25.50    2.1650025
   4    26.00    2.1732732
   4    26.50    2.1811207
...
  10    60.50    1.6632894
  10    61.00    1.6499948
  10    61.50    1.6368442
```

TRAJ1 starts with CS bond length 1.708740 Å. In the next step this distance is 1.7079124 Å. This information is printed for all trajectories until the last one, TRAJ10, for which only data until 61.50 fs is printed due to the restriction in diag.log.

If you had asked the analysis of more internal coordinates, they would also have appeared in this file as additional columns.

43. Look at mean_value.3.

```
    0.00  10       1.729985    0.053
    0.50  10       1.731448    0.053
    1.00  10       1.732855    0.053
...
   74.00   2       2.643658    0.402
   74.50   2       2.653141    0.420
   75.00   2       2.662796    0.438
...
```

At time 0 fs, the average CS distance was 1.729985 Å with standard deviation 0.053. 10 trajectories were used to compute the average. At 75.00 fs, this average bond length increased to 2.6627 Å and only two trajectories were used..

44. Plot the CS distance for all trajectories as a function of time.

```
> gnuplot
gnuplot> plot[x=1:100] 'prop.3' using 2:3
```

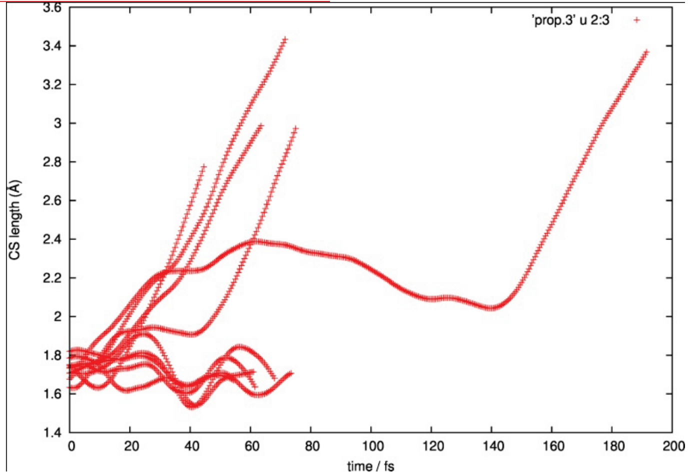


Fig. 4. CS distance (Å) as a function of time (fs) for all 10 trajectories.

Fig. 10 shows that the trajectories split in two. Five of them keeps the CS distance oscillating below 2 Å, while other part shows dissociation.

6. Initial conditions and spectrum simulation: MCSCF – Columbus

In this section, we will prepare initial conditions for dynamics using Columbus. The system is methaniminium cation $CH_2NH_2^+$ and the dynamics will be initiated in the second excited state. The initial condition will be generated by a Wigner distribution in the ground vibrational state of the ground electronic state. The transition energies and moments will be also computed with two aims: 1) to have a better control on the initial conditions and 2) to simulate the absorption spectrum.

The distribution will be created using normal modes obtained at DFT level with Turbomole.

The energies and transition moments will be calculated at MCSCF level using a complete active space composed by twelve electrons in eight orbitals [CASSCF(12,8)]. Three singlet states will be included in the state average procedure (SA-3) and the 6-31G* basis set will be used.

6.1 Geometry file creation

1. In the TUTORIAL directory (see section 1.3) create a subdirectory called CNH4_MCSCF_IC:

```
> mkdir CNH4_MCSCF_IC
```

2. Move to this directory and create a file called cnh4.xyz containing the geometry in XYZ format:

```
6

     N        0.051443        -0.125748         0.596619
     C        0.067113        -0.025698        -0.683445
     H        0.002169         0.695516         1.199263
     H        0.087711        -1.030762         1.065358
     H        0.027120         0.954772        -1.143351
     H        0.120118        -0.922908        -1.288953
```

This initial geometry can be created by hand using any text editor or with any molecular editor like Molden. It is the simple xyz Cartesian format in Angstrom. The first line contains the number of atoms and it is followed by a blank line.

These values are the ground state geometry optimized at the same CASSSCF level that will be used in the dynamics.

Because Columbus will be used, the hydrogen atoms must appear at the end of the file.

3. Convert geometry into Newton-X format.

```
> $NX/xyz2nx < cnh4.xyz <ENTER>
```

The Newton-X geometry file is called 'geom' and looks like

```
 N     7.0    0.09721318   -0.23762928    1.12744652   14.00307401
 C     6.0    0.12682519   -0.04856218   -1.29152388   12.00000000
 H     1.0    0.00409882    1.31433476    2.26627863    1.00782504
 H     1.0    0.16574977   -1.94785789    2.01323485    1.00782504
 H     1.0    0.05124937    1.80425760   -2.16062026    1.00782504
 H     1.0    0.22699012   -1.74404337   -2.43576817    1.00782504
```

The second column contains the atomic number, and the following three columns contain the x, y and z coordinates in atomic units. The last column contains the atomic mass, which can be eventually changed to simulate isotopic effects. The file can be freely formatted.

The conversion back from Newton-X *format to xyz can be done with.*

```
> $NX/nx2xyz
```

After running this program, the xyz-format geometry is written to file 'geom.xyz'.

6.2 Normal modes step

4. Create a new directory to run normal mode calculation:

```
> mkdir FREQ-DFT <ENTER>
```

5. Move to this directory and copy the geom file there.

```
> cd FREQ-DFT
> cp ../geom .
```

6. Transform the geometry into Turbomole format.

```
> $NX/nx2tm
```

nx2tm program converts Newton-X *"geom" file into* Turbomole *"coord" file (atomic units). The back conversion can be done with*

```
> $NX/tm2nx
```

7. Prepare the input and run a normal mode calculation at DFT(B3-LYP)/SV(P) level with Turbomole. In order to do that, follow specific Turbomole documentation (www.turbomole.com). Let the geometry be optimized in order to have only real frequencies. The Turbomole output containing the normal modes and frequencies can have any name. Here we will suppose it is called "force.out".

Newton-X *can read normal modes and frequencies from the outputs of the flowing programs:* Gamess, Turbomole, Columbus, Gaussian, Molden, Dftb, *and* Aces2.

Note that the normal modes and harmonic frequencies are used only to set the width of the random number distributions. For this reason they do not need be calculated at the same level as the one that will be used in the dynamics. It is, however, absolutely necessary that the *order of atoms and geometry orientation in the "geom" file and in the normal modes are exactly the same. Using* Newton-X *conversion tools (lie nx2tm) assures that.*

6.3 Energy and transition moment input

8. Move to the parent directory (TUTORIAL/CNH4_MCSCF_IC) and create a new subdirectory called JOB_AD.

```
> cd ..
> mkdir JOB_AD
```

The directory must be named JOB_AD since Newton-X *will search for it.*

9. Move into JOB_AD and copy the geom file there.

```
> cd JOB_AD
> cp ../geom .
```

10. In this directory, prepare a Columbus input for single point calculation at SA-3-CASSCF(12,8)/6-31G* without symmetry. The orbital occupation and DRT tables are given in the Appendix "Methaniminium at CAS(12,8)". The mocoef file containing the initial molecular orbitals coefficients should be provided. Important: even though the simulation will be performed at MCSCF level, the full MRCI input must be prepared. In the MRCI input section select "Excitation level = 0", "NROOT = 3", and transition moments at CI level between the ground and the two excited states. Follow the Columbus tutorial (www.univie.ac.at/columbus/tutorial).

This is the basic procedure for any program. If, for example, the initial conditions should be generated using DFT energies, JOB_AD should contain a Turbomole *input for single point calculations at DFT level.*

Before submitting running any Newton-X *job is highly advisable to run the job in JOB_AD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_AD into a test directory and run the* Columbus *job. Check whether the three roots and the oscillator strengths are computed and show reasonable values.*

6.4 Newton-X input

11. Move to directory TUTORIAL/CNH4_MCSCF_IC and create a new subdirectory called IC_MCSCF.

```
> mkdir IC_MCSCF
```

12. Move to this directory and copy the geom file, JOB_AD directory, and the Turbomole output (force.out) there.

```
> cd IC_MCSCF
> cp ../geom .
> cp -rf ../JOB_AD .
> cp ../FREQ-DFT/force.out .
> ls
force.out  geom  JOB_AD
```

13. Run the nxinp program and select option 1: Generate initial conditions.

nxinp is the input tool for NEWTON-X. *All input procedures, from the initial condition generation, through the dynamics inputs, to the statistical analysis options are performed with it.*

```
> $NX/nxinp

                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 1 <ENTER>
```

14. Select the adequate options for the initial condition properties.

nxinp will guide you through a set of questions. Each of them contains a short explanation and the available options. Most of times, the default value is the best option. To accept the default you can just press <ENTER>.

```
...
 nact:  Distribution:
        1 - Classic harmonic oscillator
        2 - Q/P-uncorrelated quantum harmonic oscillator (Wigner)
        3 - Q/P-correlated quantum harmonic oscillator
        4 - Pick points from previous dynamics
        5 - Random-velocities generation
        6 - Single point on a previous calculation
 The current value of nact  is: 2
 Enter the new value of nact  : <ENTER>
 Setting nact = 2

 numat:  Number of atoms.
 The current value of numat  is: 6
 Enter the new value of numat  : <ENTER>
 Setting numat = 6
```

```
npoints:  Number of initial conditions to be generated
The current value of npoints  is: 1
Enter the new value of npoints  : 100 <ENTER>
Setting npoints = 100
```

npoints sets the number of initial conditions that should be generated. This number depends on how many points you want to compute to generate the spectrum and how many trajectories you want to simulate.

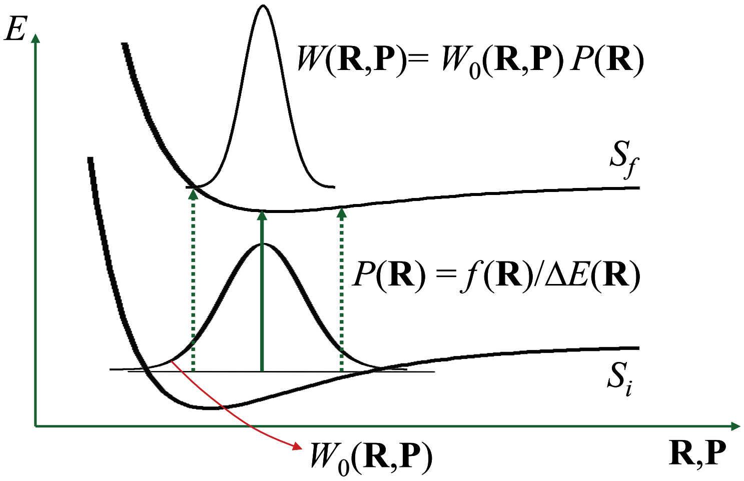

E
$W(\mathbf{R},\mathbf{P}) = W_0(\mathbf{R},\mathbf{P})\,P(\mathbf{R})$
S_f
$P(\mathbf{R}) = f(\mathbf{R})/\Delta E(\mathbf{R})$
S_i
$W_0(\mathbf{R},\mathbf{P})$
$\mathbf{R},\mathbf{P}$

Fig. 5. Initial conditions generation for excited state dynamics: first, a ground state distribution of nuclear geometries and velocities is generated (W_0, nact option). After that, the distribution is transformed according to the transition probabilities $f/\Delta E$ (chk_e = 1 option).

```
file_geom:  File containing the equilibrium geometry.
The current value of file_geom  is: 'geom'
Enter the new value of file_geom  : <ENTER>
Setting file_geom = 'geom'

iprog:  Read vibrational modes from:
       1 - GAMESS output
       2 - TURBOLMOLE output
       3 - COLUMBUS output
       4 - GAUSSIAN output
       5 - MOLDEN file
       6 - DFTB output
       7 - ACES2 output
The current value of iprog  is: 2
Enter the new value of iprog  : <ENTER>
Setting iprog = 2

file_nmodes:  File containing the normal modes.
The current value of file_nmodes  is: 'force.out'
Enter the new value of file_nmodes  : <ENTER>
Setting file_nmodes = 'force.out'

anh_f:  Multiply harmonic frequencies by this factor.
The current value of anh_f  is: 1
Enter the new value of anh_f  : <ENTER>
Setting anh_f = 1

chk_e:  Restrict (or do not) the statistical distribution:
       0 - Do not check the energies.
       1 - Check the energies between states NIS and NFS.
The current value of chk_e  is: 0
Enter the new value of chk_e  : 1 <ENTER>
Setting chk_e = 1
```

The default 0 for chk_e means that the initial conditions will be generated without checking the transition energies and moments. In this case the JOB_AD input (section 6.3) is not necessary and can be skipped. Since energies will not be computed, the generation of the initial condition is very fast, only few seconds. The result, however, will not allow generating the spectrum and to initiate the trajectories according to the transition energy (to simulate a monochromatic photo-excitation. for example).

When chk_e = 1, NEWTON-X *generates the "npoints" initial conditions uses the JOB_AD input to compute the energies and transition moments for each one (Fig. 5). Depending on the molecule and on the theoretical method this can be quite computationally expensive.*

```
The next keywords define the restriction on the statistical distribution.


 nis:  Intial state (State in which the previous dynamics run. Ground state = 1.).
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Final state.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 3 <ENTER>
 Setting nfs = 3
```

The Wigner distribution will be generated in the ground electronic state (nis = 1) and transition energies between the ground and the second excited (nfs = 3) states will be computed. Indeed, with nfs = N, transition information for all states below N will also be computed and collected.

```
kvert:  Center of the restriction:
       0 - Provide the energy for center of restriction
       1 - Use the vertical excitation energy of the equilibrium geometry
The current value of kvert  is: 1
Enter the new value of kvert  : <ENTER>
Setting kvert = 1

de:  Width of restriction (center +/- de/2).
The current value of de (eV) is: 0.5
Enter the new value of de (eV) : 100 <ENTER>
Setting de = 100
```

This large "de" implies that this restriction will not be used. It can be imposed later on.

```
prog:  Program and method to compute vertical excitation energies:
       1   - COLUMBUS
       2.0 - TURBOMOLE RI-CC2
       2.1 - TURBOMOLE TD-DFT
       5   - DFTB
The current value of prog  is: 1
Enter the new value of prog  : <ENTER>
Setting prog = 1

iseed:  Random number seed.
        0 - default seed value.
       -1 - set random seed.
       >0 - (integer) set specific value for the seed.
The current value of iseed  is: 0
```

```
 Enter the new value of iseed  : <ENTER>
 Setting iseed = 0

 lvprt:  Print level.
        1 - standard level
        2 - debug level
 The current value of lvprt  is: 1
 Enter the new value of lvprt  : <ENTER>
 Setting lvprt = 1

 mem:   COLUMBUS core memory (1 GB = 134 Mwords).

 The current value of mem  is: 200
 Enter the new value of mem  : <ENTER>
```

15. Exit nxinp.

6.5 Running NEWTON-X: initial conditions

16. Run the initial condition generation program.

```
> $NX/initcond.pl > initcond.log &
```

In general, with chk_e = 1, you might want to submit this job to a batch system. Files "pini" and "pini-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

You can split the job among several computers. To do that, see the note "Splitting and merging jobs" in section 2.5.

6.6 Checking the results

17. Normal job termination is indicated in the end of initcond.log file by:

```
End of initial conditions generation.
          ====================== NEWTON-X ends here ==================
```

18. The main information is written to the final_output files.

19. Both, final_output.1.2 and final_output.1.3 contains the same 100 initial conditions generated according to the Wigner distribution. final_output.1.2 contains transition energies and oscillator strengths to excitations into state 2 (first excited state), while final_output.1.3 contains the same information to excitations into state 3 (second excited state).

20. final_output.1.2, for example, has the following structure:

```
! The first card is the information for the initial geometry (geom)
Equilibrium geometry with null velocity:
 Geometry in COLUMBUS and NX input format:
 N     7.0    0.09721318   -0.23762928    1.12744652   14.00307401
 C     6.0    0.12682519   -0.04856218   -1.29152388   12.00000000
 H     1.0    0.00409882    1.31433476    2.26627863    1.00782504
...
Velocity in NX input format:
    0.000000000    0.000000000    0.000000000
 ...
 Reference energy (au):     -94.432419    ! CAS(12,8)/6-31G* ground state energy
 Vibrational energy (eV):       1.4587    ! Harmonic zero point energy
 Vertical excitation (eV):      9.2106    ! ΔE between S0 and S1
 Oscillator strength:           0.0000    ! S0->S1 oscillator strength
 Accept initial conditions between-90.7894 eV and109.2106 eV. ! kvert and de
                                                               choices are
                                                               reflected here

! npoints = 100 cards are given next. Each one contains random geometry, random
velocity, energy and oscillator strength information.

 Initial condition =    1                 ! Initial condition 1
 Geometry in COLUMBUS and NX input format:
 N     7.0    0.13997128   -0.23122935    1.15406634   14.00307401
...
 Velocity in NX input format:
   -0.000073416    0.000060077   -0.000254145
...
 Epot of initial state (eV):    2.8428  Epot of final state (eV):     12.0552
 Vertical excitation (eV):      9.2124  Is Ev in the required range?  YES
 Ekin of initial state (eV):    0.5082  Etot of initial state (eV):    3.3510
 Oscillator strength:           0.1295
```

“Epot of initial state” is the energy of state “nis” (in this case nis = 1) calculated in eV in relation to the reference energy given above.

“Epot of final state” is the energy of state 2 (this file is final_output.1.2). In final_output.1.3, the energy of state 3 is given.

“Vertical excitation”= “Epot of final state” – “Epot of initial state”

“Ekin” is the kinetic energy.

“Etot of initial state”= “Epot of initial state”+ “Ekin”

```
...
...
 Initial condition = 100                  ! Initial condition 100
 Geometry in COLUMBUS and NX input format:
 N     7.0    0.13840985   -0.25156066    1.10567304   14.00307401
...
 Velocity in NX input format:
    0.000174253    0.000281108   -0.000102347
...
 Epot of initial state (eV):    0.6489  Epot of final state (eV):      9.4031
 Vertical excitation (eV):      8.7542  Is Ev in the required range?  YES
 Ekin of initial state (eV):    0.7481  Etot of initial state (eV):    1.3970
 Oscillator strength:           0.0027
```

If you have a final_output file and for any reason you would like to update the energy and oscillator strength values on it (for example, because in the first run ckq_e was set to zero), you can use the option NACT = 6. It will read the geometries and velocities from your file (which should be renamed final_output.old) and perform single point calculations for each of the geometries contained there at the level set in JOB_AD directory.

6.7 Generating the spectrum

21. Go to directory TUTORIAL/CNH4_MCSCF_IC and create the subdirectory SPECTRUM.

```
> mkdir SPECTRUM
```

22. Move to this directory and copy final_output files there.

```
> cd SPECTRUM
> cp ../IC_MCSCF/final_output* .
```

23. Run nxinp and select option 5: “Generate Trajectories and Spectrum”.

```
> $NX/nxinp
                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 5
```

24. Go through the next options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : 1
 Setting type = 1
```

If for this first option you are asked for “kindjob” instead “type”, this means you are using an old version of Newton-X. Contact the distributors to get an updated version of the program.

```
 nis:  Initial state.
 The current value of nis  is: 1
 Enter the new value of nis  : <ENTER>
 Setting nis = 1

 nfs:  Array of final states (coma or dash separated, e.g., 2-4 or 2,3,4).
 For trajectories generation only one final state is allowed.
 The current value of nfs  is: 2
 Enter the new value of nfs  : 2,3 <ENTER>  ! Spectrum including both states
 Setting nfs = 2,3
```

If nfs should be given as a ‘space separated’ list, this menas that you are using an old version of Newton-X. Contact the distributors to get an updated version of the program.

```
 screen:  Energy restriction
        0 - don't apply any restriction
```

```
       1 - use the original energy restriction written in the final_output files
       2 - apply new energy restriction
The current value of screen  is: 0
Enter the new value of screen  : <ENTER>
Setting screen = 0

os_condon:  Oscillator strength:
-1 - try to read from final_output file.
  x - oscillator strength is always x (Condon approximation).
The current value of os_condon  is: -1
Enter the new value of os_condon  : <ENTER>
Setting os_condon = -1
```

If the oscillator strength is not available but the transition energy is, it is still possible to simulate the spectrum by giving a value to the oscillator strength in os_condon option. In this case, all transitions will be computed with the same oscillator strength. For only one final state this option is not bad, but it can be completely inadequate when more than one final state is selected.

```
prob_kind:  Formula to calculate the transition probabilities:
       A - Einstein-coefficient A (spontaneous emission)
       B - Einstein-coefficient B (induced absorption or induced emission)
       F - Oscillator strength (photoabsorption cross section)
The current value of prob_kind  is: B
Enter the new value of prob_kind  : B <ENTER>
Setting prob_kind = B

norm:  Normalization of transition intensities:
       local  - Use energy-restricted data set
       global - Use complete data set
The current value of norm  is: local
Enter the new value of norm  : <ENTER>
Setting norm = local

seed:  Seed for the random number generation
       0 - a default random number seed is used
       1 - a randomized seed is used
       Any other positive integer is used as the random number seed
The current value of seed  is: 0
Enter the new value of seed  : <ENTER>
Setting seed = 0

l_shape:  Line shape:
       gauss   - Normalized Gaussian function.
       lorentz - Normalized Lorentzian function.
The current value of l_shape  is: lorentz
Enter the new value of l_shape  : <ENTER>
Setting l_shape = lorentz

delta:  Phenomenological broadening of the spectrum.
The current value of delta (eV) is: 0.05
Enter the new value of delta (eV) : <ENTER>
Setting delta = 0.05
```

“delta” is possibly the only option that you need to play with. It controls the width of the gaussian curve attributed to each one of the 200 transitions (100 initial conditions x 2 final states).

```
temp:  Temperature.
The current value of temp (K) is: 0
Enter the new value of temp (K) : <ENTER>

nref:  Refraction index.
The current value of nref  is: 1
Enter the new value of nref  : <ENTER>
```

```
eps:  Distance between consecutive points in the spectrum.
The current value of eps (eV) is: 0.005
Enter the new value of eps (eV) : <ENTER>
Setting eps = 0.005

kappa:  The range of the spectrum is defined between
Emin-kappa*delta and Emax+kappa*delta, where kappa is an integer.
The current value of kappa  is: 3
Enter the new value of kappa  : <ENTER>
Setting kappa = 3
```

25. Exit nxinp.

26. The simulated spectrum using the “gaussian broadening” method as a function of excitation energy is written to spectrum.dat. Looking in this file, we see:

```
     ! DE(eV)  Wavelength(nm)  cross-section(Å^2.molecule^-1)
       7.3546   1.6858e+02        0.0004
       7.3596   1.6847e+02        0.0004
       7.3646   1.6835e+02        0.0004
...
```

27. Run Gnuplot to visualize the spectrum (Fig. 6).

```
> gnuplot
gnuplot> plot 'spectrum.dat' using 1:2 with lines
```

28. The files final_output* will be also used to initialize the trajectories.

As already observed, the final_output files are used by NEWTON-X *to simulate the spectrum and to initialize dynamics trajectories. In this section they have been generated by means of a Wigner distribution. Several other options – random velocities, classical distributions, pick-points from previous dynamics – are included in* NEWTON-X. *Indeed, for* NEWTON-X *does not matter how final_output was generated, only that the file with the correct structure exists.*

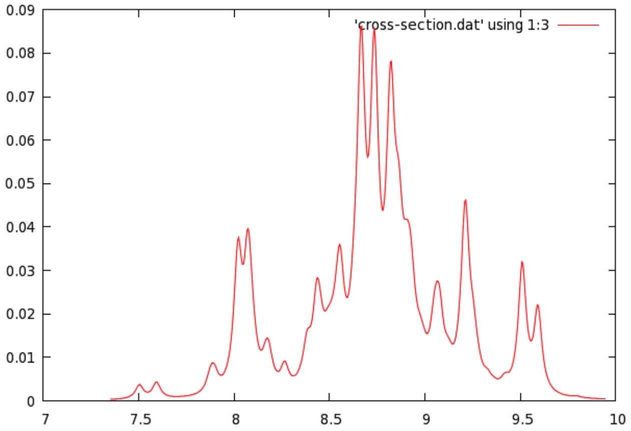


Fig. 6. Spectrum (Å^2.molecule^{-1}) as a function of the excitation energy (eV). Computed with Lorentzian line shape and phenomenological broadening δ = 0.05 eV.

7. Surface hopping dynamics: MCSCF – Columbus

In this section we will prepare the inputs and run dynamics simulations. The system is methaniminium cation $CH_2NH_2^+$ and the dynamics will be initiated in the second excited state. The initial conditions were generated in Chapter 0 of this tutorial. The surface hopping[10] dynamics will be performed at MCSCF level using COLUMBUS. The trajectory time will be 100 fs with 0.5 fs time step.

The MCSCF level is a complete active space composed by twelve electrons in eight orbitals [CASSCF(12,8)]. Three singlet states will be included in the state average procedure (SA-3) and the 6-31G* basis set will be used.

7.1 Copying the initial conditions

1. Go to TUTORIAL directory (see section 1.3) create a subdirectory called CNH4_MCSCF_DYN:

```
> mkdir CNH4_MCSCF_DYN
```

2. Move to this directory and copy the initial conditions file for the second excited state there.

```
> cd CNH4_MCSCF_DYN
> cp ../CNH4_MCSCF_IC/IC_MCSCF/final_output.1.3 final_output ! note the new
                                                                name
```

3. Create a subdirectory called JOB_NAD.

The directory must be named JOB_NAD since NEWTON-X *will search for it.*

7.2 Preparing electronic structure input

4. Move to JOB_NAD and create an input for single point non-adiabatic coupling calculation with COLUMBUS at SA-3-CASSCF(12,8)/6-31G*. The orbital occupation and DRT tables are given in the Appendix 9.1 “Methaniminium at CAS(12,8)”. The mocoef file containing the initial molecular orbital coefficients should be provided. Important: even though the simulation will be performed at MCSCF level, the input should be

prepared for CI analytical gradient and full MRCI input must be prepared. In the MRCI input section select “Excitation level = 0”, “NROOT = 3”, and non-adiabatic coupling vectors between the second excited and the ground states and between the second and the first states. Follow the Columbus tutorial (www.univie.ac.at/columbus/tutorial).

> *Before submitting running any Newton-X job is highly advisable to run the job in JOB_NAD directory just to test whether the input and outputs are OK. In the current case, just copy the content of the JOB_NAD into a test directory and run the COLUMBUS job. Check whether the three roots and the non-adiabatic coupling vectors are computed and show reasonable values.*

7.3 NEWTON-X input

5. Run nxinp.

```
> $NX/nxinp

                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 2 <ENTER>
```

6. Select option 2: “set basic input” and answer the questions to set the dynamics options.

```
 nat:  Number of atoms.
 The current value of nat  is: 6
 Enter the new value of nat  : <ENTER>
 Setting nat = 6

 nstat:  Number of states.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>     ! 3 states will be computed
 Setting nstat = 3
```

```
nstatdyn:  Initial state (1 - ground state).
The current value of nstatdyn  is: 3
Enter the new value of nstatdyn  : <ENTER>     ! dynamics will start in state 3
Setting nstatdyn = 3

dt:  Time step for the classical equations.
The current value of dt (fs) is: 0.5
Enter the new value of dt (fs) : <ENTER>
Setting dt = 0.5

tmax:  Total duration of the trajectory.
The current value of tmax (fs) is: 100
Enter the new value of tmax (fs) : <ENTER>
Setting tmax = 100

prog:  Quantum chemistry program and method
       0    - ANALYTICAL MODEL
       1    - COLUMBUS
       2.0 - TURBOMOLE RI-CC2
       2.1 - TURBOMOLE TD-DFT
       3    - ACES2 EOM-CCSD (in development)
       4    - MOPAC (in development)
       5    - DFTB
       6    - GAUSSIAN              ! NX < 1.2
       6.0 - GAUSSIAN(03) CASSCF   ! NX > 1.4
       6.5 - GAUSSIAN(09) TDDFT
       7    - TINKER
       8    - DFTB+
       10.0    - GAMESS
       10.1    - GAMESS ARBITRARY ADIABATIC METHOD
       20   - HYBRID GRADIENT
The current value of prog  is: 1
Enter the new value of prog  : <ENTER>
Setting prog = 1

thres: Threshold to initiate nonadiabatic coupling calculation.
         0 - Only adiabatic dynamics.
       100 - Only nonadiabatic dynamics
The current value of thres (eV) is: 100
Enter the new value of thres (eV) : <ENTER>
Setting thres = 100

Would you like to generate internal coordinate file?  (default=y) n <ENTER>
```

7. In the main nxinp menu you have the option to set more general options if you select either option 3 or option 4. Normally, this is not necessary and the default values are good enough.

8. Select option 5: "generate trajectories and spectrum".

```
============================================================
                         NEWTON-X
         Newtonian dynamics close to the crossing seam
                       www.newtonx.org
============================================================

                 MAIN MENU


                 1. GENERATE INITIAL CONDITIONS


                 2. SET BASIC INPUT
```

```
                                3. SET GENERAL OPTIONS

                                4. SET NONADIABATIC DYNAMICS

                                5. GENERATE TRAJECTORIES AND SPECTRUM

                                6. SET STATISTICAL ANALYSIS

                                7. EXIT

Select one option (1-7): 5 <ENTER>
```

9. Select the job options.

```
type:  What do you want to do?
         1 - Generate spectrum
         2 - Select initial conditions for multiple initial states
         3 - Generate trajectories
         4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : <ENTER>
 Setting type = 3
```

If for this first option you are asked for "kindjob" instead "type", this means you are using an old version of Newton-X. Contact the distributors to get an updated version of the program.

```
 Would you like to apply energy and probability restrictions (y/n)?  (default=n)
<ENTER>

 Will the trajectories run in a batch system? n <ENTER>

        Processing data: This may take some minutes. Please, wait...
```

In general, you might want to submit this job to a batch system. In this case, answer "y" to "Will the trajectories run in a batch system?". You will be asked the name of the submission script and nxinp will look for it existence in the input directory. Files "pmold" and "pmold-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

10. Exit nxinp.

11. Now the input directory should contain a new directory called TRAJECTORIES, which was created by nxinp.

```
> ls
control.dyn  final_output  JOB_NAD  makedir.log  mkd.inp  TEMP  TRAJECTORIES
```

12. Move to TRAJECTORIES and list the content. You will see that 100 directories named TRAJ1 to TRAJ100 were created. Each one of these directories contains a complete set of input files to run one independent trajectory with different initial conditions (geometry and velocities).

The number of TRAJn directories is controlled by the number of initial conditions in the final_output file. Each initial condition card is split into a geom and veloc files that appear inside TRAJn.

The TRAJn structure of directories (see Fig. 7) is very important in NEWTON-X. *Several tools to control the jobs and to perform statistical analysis use it.*

Each TRAJn directory can be copied to a different computer or submitted to a different node of a batch system. They should, however, be grouped again after having finished the dynamics calculations.

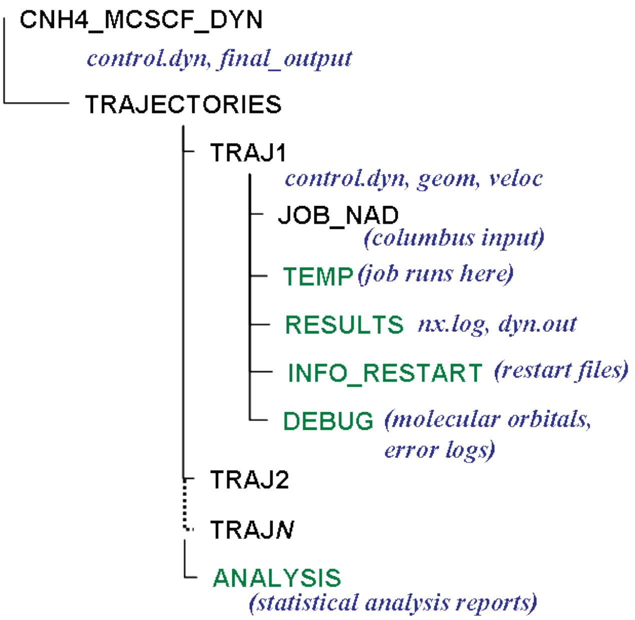


Fig. 7. Directory tree for a complete NEWTON-X surface hopping dynamics job.

7.4 Running NEWTON-X: dynamics simulations

13. Move to TRAJ1 and run this trajectory.

```
> cd TRAJ1
> $NX/moldyn.pl > moldyn.log &
```

14. Move to TRAJ2 and run that trajectory; the same with TRAJ3, TRAJ4, and so on. In the test calculations for this TUTORIAL we have run a total of 60 trajectories, from TRAJ1 to TRAJ60.

If the jobs should be submitted to a batch system you can either go to each TRAJn directory and run

```
> qsub pmold       ! pmold is the submission script
```

or in the TRAJECTORIES directory you can run

```
> $NX/submit.pl
```

In this case, you will be asked the initial and final trajectory number that should be submitted, and the submission of all these trajectories will be done automatically.

7.5 Checking the results

15. For each TRAJn, the dynamics output is in three directories:

- RESULTS: the dynamics information;
- INFO_RESTART: stored information to restart the trajectory;
- DEBUG: error messages, convergence information and additional electronic structure files like molecular orbital files.

16. Go to directory TRAJ1/RESULTS.

```
> cd TRAJ1/RESULTS
```

17. If it finished with error, nx.log should finish like:

```
moldyn.pl:   ::ERROR::   step =       60 time = 30 fs
Trying to save TEMP directory to ../DEBUG/COL.30
```

In this example, the job finished with error after 60 steps (30 fs). The last COLUMBUS calculation was written into TRAJ1/DEBUG/COL.30/TEMP and you can check what was wrong. More error messages are also written to DEBUG/runnx.error file.

18. Go to directory TRAJ2/RESULTS.

```
> cd ../../TRAJ2/RESULTS
```

19. If the job ended normally, the file nx.log should finish with:

```
moldyn.pl: End of dynamics
           ====================== NEWTON-X ends here ==================
```

20. Plot the energies (Fig. 8):

```
> $NX/plot
Number of states: 3
```

The energy information used by plot program is in the file RESULTS/en.dat. It looks like:

```
 !Time(fs)  E0 (au)       E1 (au)       E2 (au)      E(current) (au)  Etot (au)
    0.00    -94.412074    -94.081511    -94.016479    -94.016479    -93.971870
    0.50    -94.410343    -94.085650    -94.013214    -94.013214    -93.971792
    1.00    -94.402481    -94.083425    -94.008725    -94.008725    -93.971517
...
```

If you give one more state (4, in this example), the total energy is also printed.

You can run plot program for a sequence of trajectories either to see them on screen or only to generate pictures. To do so, go to TRAJECTORIES directory and run:

```
> $NX/plotall
```

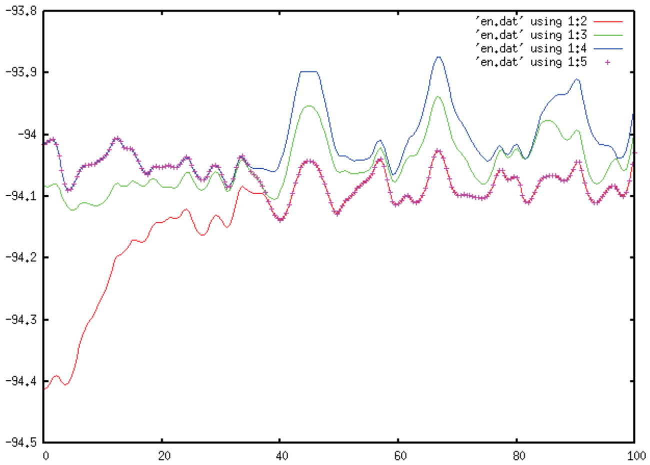


Fig. 8. Potential energies of the three states as a function of time ploted with `$NX/plot`. The crosses indicate the current state of the system at each time step.

21. Visualize the dynamics:

```
> molden dyn.mld &
```

dyn.mld file is a simple multiple-xyz format file with all geometries along the trajectory. It can be opened by most of programs for dynamics visualization. One good option is VMD (www.ks.uiuc.edu/Research/vmd), which allow a quite complete control of the time line and excellent rendering quality.

To save disk space, dyn.mld is not automatically created starting from NEWTON-X version 1.2.5. In these new versions, you can get the xyz file by moving to the RESULTS directory and running

```
> $NX/dynout2xyz.pl
```

A file called dyn.xyz (multiple xyz format) is created and can be visualized with programs like Molden or VMD.

22. Geometry, velocities, energies and other quantities are written in a more verbose way in file dyn.out. It looks like:

```
...
STEP       2    Molecular dynamics on state 3    TIME =       1.00 fs
Type of dyn. =    2    N. of hoppings =    0    N. of rejec. hoppings =    0

  New geometry:
 N      7.0     0.09015243   -0.20946057    1.09319790   14.00307401
 C      6.0     0.17542005   -0.09999328   -1.25102014   12.00000000
 H      1.0     0.08720540    1.23985176    2.05594456    1.00782504
 H      1.0     0.02427253   -1.94636141    2.14504766    1.00782504
 H      1.0    -0.26087804    1.97499271   -1.89595326    1.00782504
 H      1.0     0.11655483   -1.62034288   -2.62874082    1.00782504

  New velocity:
   -0.00001223    0.00029810    0.00052112
    0.00012438    0.00027577   -0.00025372
    0.00047115   -0.00279091   -0.00053004
   -0.00081528   -0.00210269    0.00076752
   -0.00100439    0.00038167   -0.00241802
    0.00003574   -0.00291432   -0.00204379
```

```
     Time    Etot         Ekin       Epot E0,      E1, ...
%     1.00    -93.971517      0.037208    -94.402481    -94.083425    -94.008725
Etot variation =  0.000274 au    Worse conservation =  0.000353 au

 Wave function state 1:        -0.00027092051898        -0.00017265970290
 Wave function state 2:         0.00514996477635         0.00752162117458
 Wave function state 3:         0.99995839829598        -0.00003931597926
...
```

"Etot variation" gives $E_{tot}(t)$-$E_{tot}(t-\Delta t)$. When this quantity is larger than a threshold, the trajectory breaks with "Etot_jump" error message. This threshold can be controlled with the keyword "Etot_jump" in the advanced input.

"Worse conservation" gives MAX[$E_{tot}(t)$-$E_{tot}(0)$] *until time t. When this quantity is larger than a threshold, the trajectory breaks with "Etot_drift" error message. This threshold can be controlled with the keyword "Etot_drift" in the advanced input.*

In surface hopping approach, the time-dependent wavefunction is written as:[10]

$$\Psi(t)=\sum_{n=1}^{Nstat}\chi_n(t)\psi_i(\mathbf{R}(t)),$$

where ψ_i is the electronic wavefunction (in the adiabatic representation, in this example). The time-dependent coefficients χ_n are complex numbers: The real and imaginary parts of them are given in dyn.out. For example, the line

```
Wave function state 1:        -0.00027092051898        -0.00017265970290
```

means

$$\mathrm{Re}(\chi_0)=-0.00027 \ \ \textit{and} \ \ \mathrm{Im}(\chi_0)=-0.00017.$$

The normalization of these coefficients is an important check of the quality of the dynamics and it will be discussed below.

23. Energy information is also written in a different format in file typeofdyn.log. It looks like:

```
...
Time = 37.00 Threshold=** PES= 2 DE_inf = 0.32 DE_sup = 0.78 Type = 4 Next type = 4
Time = 37.50 Threshold=** PES= 2 DE_inf = 0.19 DE_sup = 0.99 Type = 4 Next type = 4
Time = 38.00 Threshold=** PES= 1 DE_inf = ---- DE_sup = 0.23 Type = 4 Next type = 3
...
```

At time 37.5 fs the molecule was on surface 2 (S_1). The energy gap to S_0 was 0.19 eV and to S_2, 0.99 eV (absolute values). At time 38.0 fs, the molecule was on surface 1 (S_0), therefore a surface hopping took place.

24. Gradients, oscillator strengths and non-adiabatic coupling vectors and log information are written to nx.log (and also in the standard output).

25. The information about the integration of the time dependent Schrödinger equation is in sh.log file. It looks like:

```
...
  substep     step  surf     populations      Total pop
...
|v.h|=     0.000000     0.000145   -0.000015
        4        1     3     0.000000164    1.000000000
```

```
                                             0.000000001
                                             0.999999835
...
```

NEWTON-X *keeps two related time lines. The first is composed by the steps used to integrate the Newton's equations. For step n, the time is n.Δt (Δt = 0.5 fs in this example). The second time line is counted in substeps, which are used to integrate the time-dependent Schrödinger equation. The number of substeps per step is defined by keyword "ms", which can be changed in the advance input. The default is ms = 20. At the end of a trajectory with N steps, we will have N.ms substeps.*

At the substep 4, still during step 1, with the molecule still on surface 3 (S_2), the population of the second excited state (S_2) was:

$$|\chi_2|^2 = 0.999999835.$$

The population is defined as the diagonal terms of the matrix $\chi_i \chi_j^*$. *These terms are written in sh.log. Note that*

$$\sum_{n=1}^{Nstat} |\chi_n|^2 = 1.$$

The normalization factor is also printed there ("Total pop"). When it deviates more than "pop_dev" (a keyword defined in the advance input, whose default is 0.05), the job breaks with an error message.

The inner products of the nuclear velocity and the non-adiabatic coupling vectors, which are proportional to the hopping probability, are also printed:

$$\mathbf{v}\cdot\mathbf{h}_{10} = 0.000000,$$
$$\mathbf{v}\cdot\mathbf{h}_{20} = 0.000145,$$
$$\mathbf{v}\cdot\mathbf{h}_{21} = -0.000015.$$

26. The hopping probability is written in tprob file. It looks like:

```
  random number    substep     step    probabilities
...
     0.586478467      1386       70     0.000009603     0.013179556     0.000000000
     0.186681612      1387       70     0.000009707     0.013189909     0.000000000
     0.008996025      1388       70     0.000009800     0.013172354     0.000000000
    20.000000000      1389       70     0.000000000     0.000000000     0.000000000
     0.344794082      1390       70     0.000000000     0.000000000     0.000000000
     0.161303038      1391       70     0.000000000     0.000000000     0.000000000
...
```

In the substep 1386 (during step 70), the hopping probabilities were:

$$P_{2\to 0} = 0.000009603,$$
$$P_{2\to 1} = 0.013179556,$$
$$P_{2\to 2} = 0.000000000.$$

These probabilities were compared with the random number 0.586478467 to decide whether a hopping occurs or not. Since the random number was larger than the probabilities, the molecule stays on S_2 surface in the next substep (1387). In the substep 1388, the random number (0.008996) was smaller than $P_{2\to 1}$, therefore the molecules hops to surface S_1. The *surface hopping along the trajectories are marked by a "20." in the place of the random number.*

In the fewest-switches approach under adiabatic representation, the hopping probability from state j to state i is given by:

$$P_{j\to i}(t)=\begin{cases}-2\dfrac{\mathrm{Re}\left(\chi_i^{*}\chi_j e^{i\gamma_{ji}}\right)}{\left|\chi_{jj}\right|^{2}}\mathbf{v}(t)\cdot\mathbf{h}_{ji}(t)\delta t, & \text{if } P_{j\to i}>0.\\ 0, & \text{if } P_{j\to i}\leq 0.\end{cases}$$

In this equation, $\delta t = \Delta t/ms$. Due to this definition, the probability becomes zero for some time intervals (like in substep 1391).

27. Points close to conical intersections (energy gaps smaller than "e_ci", which is keyword defined in advanced input. The default is 0.2 eV) are written to "report.ci" file.

28. Oscillator strength and transition dipole moments are written to properties.

29. You can generate MOLDEN-format files for any time step including non-adiabatic coupling vectors (as vibrations) and molecular orbitals (if available for that time step). To do so, run:

```
> $NX/arrow
```

COLUMBUS *molecular orbital files are stored every "prt_mo" time steps, which is a keyword that can be changed in the advanced input. The MO files are compressed in the DEBUG/COL.t directories with name mocoef_mc.sp.gz.*

30. You can check the convergence of the MCSCF and CI calculations during the dynamics in file DEBUG/log.conv.

7.6 Statistical analysis of the trajectories

31. After running your trajectories, you can perform a series of statistical analysis on them.

32. Go to TRAJECTORIES directory, run nxinp, and select option 6: "Set statistical analysis".

```
============================================================
                           NEWTON-X
         Newtonian dynamics close to the crossing seam
                       www.newtonx.org
============================================================

                MAIN MENU


                1. GENERATE INITIAL CONDITIONS


                2. SET BASIC INPUT
```

```
                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT

Select one option (1-7): 6 <ENTER>
```

33. First, perform a general diagnostic of errors.

```
        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 1): 1 <ENTER>

 ad:  Path to the trajectories. The default is the current directory.
 The current value of ad  is: /home/TUTORIAL/CNH4_MCSCF_DYN_S2/TRAJECTORIES

 Enter the new value of ad  : <ENTER>
 Setting ad = /home/TUTORIAL/CNH4_MCSCF_DYN_S2/TRAJECTORIES

 itraj:  Initial trajectory to be checked.
 The current value of itraj  is: 1
 Enter the new value of itraj  : <ENTER>
 Setting itraj = 1

 ftraj:  Final trajectory to be checked. There is not sensible default value.
 The current value of ftraj  is:
 Enter the new value of ftraj  : 60      ! In the test calculations for
 Setting ftraj = 60                        this tutorial, we have run 60
                                               trajectories.

 tmin:  Initial time. It must be common to all trajectories.
 The current value of tmin (fs) is: 0.0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0.0

 Etot_dev:  Maximal allowed total energy variation.
 The current value of Etot_dev (eV) is: 0.5
 Enter the new value of Etot_dev (eV) : <ENTER>
 Setting Etot_dev = 0.5

 pop_dev:  Maximal allowed variation in the norm of the adiabatic population.
 The current value of pop_dev  is: 0.1
 Enter the new value of pop_dev  : 0.05 <ENTER>
 Setting pop_dev = 0.05
```

34. Exit nxinp and run:

```
> $NX/diagnostic.pl
```

35. The result is the file diag.log. It contains a summary of what happened to the trajectories. It looks like:

```
Diagnostic of trajectories 1 to 60

Path to the trajectories:  /home/TUTORIAL/CNH4_MCSCF_DYN_S2/TRAJECTORIES
Initial time (fs):         0.0
Max. Energy deviation(eV): 0.5
Max. population deviation: 0.05
Number of states:          3
Normal termination:        36        ! Normal termination
Error termination:         24        ! Error termination

TRAJECTORY 1:
moldyn.pl:   ::ERROR::   step =       60 time = 30 fs
Suggestion: make analysis until time (fs): 30.00

TRAJECTORY 2:
Normal termination
Suggestion: make analysis until time (fs): 100.00
...
Distribution of trajectories:
Number of trajs finished between 0.0 and 25 fs: 4
Number of trajs finished between 25 and 50 fs: 8
Number of trajs finished between 50 and 75 fs: 5
Number of trajs finished between 75 and 100 fs: 43
...
```

36 trajectories had normal termination and 43 performed at least 75 fs before finishing.

A trajectory finished in error can still contain relevant information. One of the most often reason for finishing in error is the connected with the adequacy of electronic structure method to compute the energies and gradients for some specific nuclear geometry.

For instance, dynamics trajectories performed with TD-DFT will crash as soon as the molecule enters in a region of strong multireference character. Trajectories performed with CASSCF may crash if the molecule dissociates and the necessary anti-bonding orbitals are not in the active space.

The dynamics information until these critical points, however, usually is correct and can be useful for the global statistical analysis.

For each trajectory, diagnostic.pl program prints until which time the calculation is reliable according to the criteria defined in its input. The log file, for example, says that for TRAJ1, data until 30 fs can be used.

It is not mandatory to run diagnostic.pl before the statistical analysis but it is quite advisable.

If the statistical analysis program finds the diag.log file, it will automatically use the "suggested time" to decide which data to use from each trajectory. You can change these suggested times in diag.log if you want.

36. Run nxinp and select option 6: "Set statistical analysis".

```
MAIN MENU


1. GENERATE INITIAL CONDITIONS


2. SET BASIC INPUT


3. SET GENERAL OPTIONS


4. SET NONADIABATIC DYNAMICS
```

```
                              5. GENERATE TRAJECTORIES AND SPECTRUM


                              6. SET STATISTICAL ANALYSIS


                              7. EXIT


Select one option (1-7): 6 <ENTER>
```

37. Make the input for statistical analysis. Start making the “wave function analysis”.

```
        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 2): 2 <ENTER>

 itrj:  Initial trajectory to be analysed.
 The current value of itrj  is: 1
 Enter the new value of itrj  : <ENTER>
 Setting itrj = 1

 jtrj:  Final trajectory to be analysed.
 The current value of jtrj  is: 10
 Enter the new value of jtrj  : 60 <ENTER>
 Setting jtrj = 60

 tmin:  Initial time for the analysis.
 The current value of tmin (fs) is: 0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0

 tmax:  Final time for the analysis.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : <ENTER>
 Setting tmax = 100

 dt:  Time step in which outputs are written.
        If TRAJ1/control.dyn is found, default is DT*KT,
        otherwise default is 0.5 fs.
 The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 proptype:  Kind of properties to be analysed.
        1. Energy.
        2. Wave function.
        3. Internal coordinates.
        4. Internal forces (only for COLUMBUS dynamics).
        5. Velocity autocorrelation.
 The current value of proptype  is: 1
 Enter the new value of proptype  : 2 <ENTER>

 nstat:  Number of states to be analysed.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>
 Setting nstat = 3

 complete_data:  Complete data for broken trajectories.
        0. Do not complete data. (default)
        Other positive value. If the last time in the trajectory
        is larger than complete_data, repeat last set of data until
        tmax. Neglect trajectories whose last time is
        smaller than complete_data.(Only for proptype = 2.)
 The current value of complete_data  is: 0
 Enter the new value of complete_data  : <ENTER>
 Setting complete_data = 0
```

38. Exit nxinp and run:

```
> $NX/analysis.pl > analysis.log &
```

analysis.pl program makes averages over trajectories for several different properties.

39. If everything runs normally, anaysis.log ends with:

```
analysis: End of analysis
          ====================== NEWTON-X ends here ==================
```

40. Go to the directory ANALYSIS and look at file prop.2. It looks like:

```
!Traj Time/fs  SA  SB  SA.SB   a0        a1        a2
   1     0.00   3   3   3.3   0.0000    0.0000    1.0000
   1     0.50   3   3   3.3   0.0000    0.0041    0.9959
   1     1.00   3   3   3.3   0.0000    0.1007    0.8993
...
   1     6.00   3   3   3.3   0.0000    0.4545    0.5455
   1     6.50   3   2   3.2   0.0000    0.6710    0.3290
   1     7.00   2   2   2.2   0.0000    0.8052    0.1948
...
  60    15.00   3   3   3.3   0.0000    0.0388    0.9612
  60    15.50   3   3   3.3   0.0000    0.0628    0.9372
  60    16.00   3   3   3.3   0.0000    0.0927    0.9073
```

Trajectory 1 starts at time 0 fs in surface 3 (given by SA) and ends this time stem still on surface 3 (given by SB). In this step all population is in surface 3 ($a2 = |\chi_{22}|^2 = 1$ *).*

At time 6.5 fs, TRAJ1 starts on surface 3 and ends on surface 2, meaning that a hopping took place. Now $|\chi_{22}|^2 = 0.3290$.

This description continues for all trajectories until TRAJ60, for which only data until 16.0 fs was used (because of the instruction in diag.log).

41. Look at mean_value.2.

```
!Time/fs  N    N0      <a0>    da0     N1      <a1>    da1     N2      <a2>    da2
    0.00  60   0.000   0.000   0.000   0.000   0.000   0.000   1.000   1.000   0.000
    0.50  60   0.000   0.000   0.000   0.000   0.001   0.001   1.000   0.999   0.001
    1.00  60   0.000   0.000   0.000   0.000   0.019   0.080   1.000   0.981   0.080
...
   49.50  48   0.542   0.530   0.483   0.417   0.431   0.475   0.042   0.039   0.183
   50.00  48   0.542   0.528   0.481   0.417   0.435   0.472   0.042   0.036   0.173
   50.50  48   0.521   0.532   0.480   0.438   0.437   0.468   0.042   0.031   0.147
...
   99.00  36   0.722   0.733   0.420   0.222   0.211   0.382   0.056   0.056   0.231
   99.50  36   0.722   0.738   0.419   0.222   0.207   0.381   0.056   0.056   0.232
  100.00  36   0.722   0.743   0.415   0.222   0.201   0.376   0.056   0.056   0.232
```

This file contains the average over all trajectories as a function of time.

A time 1 fs, 60 trajectories were used to compute the averages. The fraction of trajectories on state S_2 *was N2 = 1 and the average adiabatic population was* $\left\langle |\chi_{22}|^2 \right\rangle = 0.981$ *with standard deviation 0.080.*

At time 100, only 36 trajectories were used to compute the average, most of the trajectories was in the ground state (0.722) and the adiabatic population of the ground state was $\left\langle |\chi_{22}|^2 \right\rangle = 0.743$.

42. Plot the fraction of trajectories (occupation) and the average adiabatic population for the second excited state as a function of time (Fig. 9).

```
> gnuplot
gnuplot>plot[x=1:100] 'mean_value.2' using 1:9 with lines,'mean_value.2' using 1:10
```

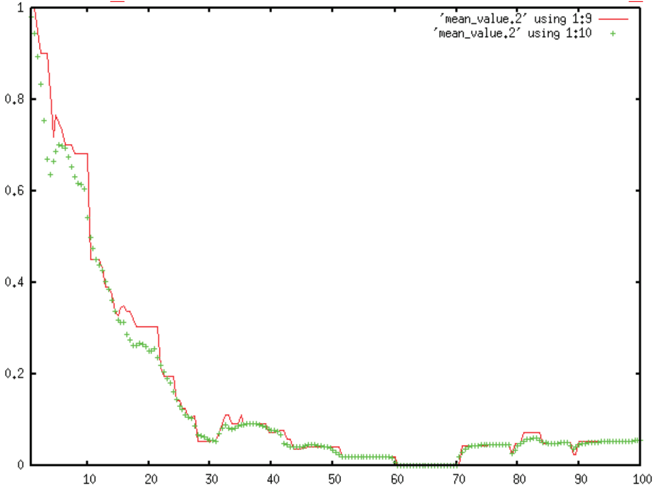


Fig. 9. Fraction of trajectories (solid line) and average adiabatic population (dots) in the second excited state as a function of time.

Fig. 9 shows an exponential decay of the population with time constant (lifetime) of about 20 fs.

One important test for the quality of the surface hopping dynamics is the comparison between the occupation (fraction of trajectories) and the average adiabatic population.[13] *They should be more or less the same.*

43. Go to TRAJECTORIES directory, run nxinp again, select Statistical analysis option and choose analysis of “Internal coordinates”.

```
============================================================
                           NEWTON-X
          Newtonian dynamics close to the crossing seam
                        www.newtonx.org
============================================================

                MAIN MENU


                1. GENERATE INITIAL CONDITIONS


                2. SET BASIC INPUT


                3. SET GENERAL OPTIONS
```

```
                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 6 <ENTER>

        What do you want to do?
        1 - Generate input for diagnostic (only statistics on errors)
        2 - Generate input for statistical analysis (statistics on wavefunction,
energies, etc.)
        3 - Return to main menu

        Enter the option (default = 2): <ENTER>

 Would you like to delete old input files? (prop.inp) y <ENTER>

 itrj:  Initial trajectory to be analysed.
 The current value of itrj  is: 1
 Enter the new value of itrj  : <ENTER>
 Setting itrj = 1

 jtrj:  Final trajectory to be analysed.
 The current value of jtrj  is: 10
 Enter the new value of jtrj  : 60 <ENTER>
 Setting jtrj = 60

 tmin:  Initial time for the analysis.
 The current value of tmin (fs) is: 0
 Enter the new value of tmin (fs) : <ENTER>
 Setting tmin = 0

 tmax:  Final time for the analysis.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : <ENTER>
 Setting tmax = 100

 dt:  Time step in which outputs are written.
        If TRAJ1/control.dyn is found, default is DT*KT,
        otherwise default is 0.5 fs. The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 proptype:  Kind of properties to be analysed.
        1. Energy.
        2. Wave function.
        3. Internal coordinates.
        4. Internal forces (only for COLUMBUS dynamics).
        5. Velocity autocorrelation.
 The current value of proptype  is: 1
 Enter the new value of proptype  : 3 <ENTER>

 nstat:  Number of states to be analysed.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>
 Setting nstat = 3

 nic:  Number of internal coordinates to be analised (Maximum = 100).
 The current value of nic  is: 1
 Enter the new value of nic  : 1 <ENTER>
 Setting nic = 1
```

The internal coordinates are defined in the intcfl file in the JOB_NAD directory. The coordinate number one in this example corresponds to the CN stretching.

```
iclist:  Array with the number of internal coordinates to be analysed.
         Example: if the stretch corresponds coordinate 1 and the torsion to 12,
         and one wants to analyse both, iclist = 1,12 and nic = 2.
The current value of iclist  is: 1
Enter the new value of iclist  : <ENTER>
Setting iclist = 1

bmat:  Internal coordinates options
       0. Get the internal coord. from output files. COLUMBUS only.
       1. Run cart2int.x program from COLUMBUS to get the internal
coordinates. In this case, an intcfl file with definition
of the internal coordinates is required.
The current value of bmat  is: 0
Enter the new value of bmat  : <ENTER>
Setting bmat = 0
```

44. Exit nxinp and run:

```
> $NX/analysis.pl > analysis.log &
```

analysis.pl program makes averages over trajectories for several different properties.

45. If everything runs normally, anaysis.log ends with:

```
analysis: End of analysis
          ====================== NEWTON-X ends here ==================
```

46. Go to the directory ANALYSIS and look at file prop.3. It looks like:

```
! Traj  Time/fs R_CN (Angstrom)
   1     0.00    1.3237134
   1     0.50    1.3169850
...
  60    16.00    1.6106203
  60    16.50    1.6013954
```

TRAJ1 starts with CN bond length 1.324 Å. In the next step this distance is 1.317 Å. This information is printed for all trajectories until the last one, TRAJ60, for which only data until 16.5 fs is printed due to the restriction in diag.log.

If you had asked the analysis of more internal coordinates, they would also have appeared in this file as additional columns.

47. Look at mean_value.2.

```
    0.00  60       1.278966   0.037
    0.50  60       1.279945   0.035
    1.00  60       1.286777   0.033
...
   99.50  36       3.586967   2.436
  100.00  36       3.597304   2.446
```

At time 0 fs, the average CN distance was 1.279 Å with standard deviation 0.037. 60 trajectories were used to compute the average. At 100 fs, this average bond length increased to 3.597 Å. The standard deviation, 2.446 Å, is too large to be due only to the smaller number of trajectories included in the computation of the average (36). It may be indicating that more that one reaction channel is present, with rather different CN distances.

48. Plot the CN distance for all trajectories as a function of time.

```
> gnuplot
gnuplot> plot[x=1:100] 'prop.3' using 2:3
```

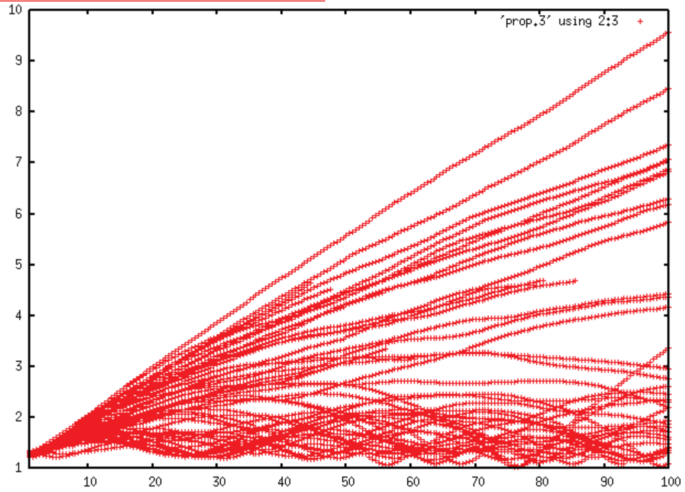


Fig. 10. CN distance (Å) as a function of time (fs) for all 60 trajectories.

Fig. 10 shows that the trajectories split in two. Part of them keeps the CN distance oscillating below 2 Å, while other part shows dissociation.

8. Hybrid dynamics: QM/MM – Columbus/Tinker

8.1 General explanations of hybrid calculations

The most popular hybrid method is the QM/MM approach, in which a reactive primary system is treated quantum mechanically (QM), while the remaining (secondary) system is treated by molecular mechanics (MM). There is a vast literature reviewing the QM/MM approach and we can specially refer to Refs.[14-16].

The NEWTON-X implementation the energies and gradients are for subsets of atoms are treated with different programs and the partial results are joint into a resulting total (hence 'hybrid') energy and gradient.

For that purpose the complete set of atoms of the whole system is split in disjoint regions. These regions need not follow physical reasoning (they often will, but the can also pick e.g. single atoms out of molecules), but are logical entities for the definition of the single partial calculations.

The whole calculation is split into jobs. Each of these jobs can treat one or more regions of atoms and the result enters the total multiplied by a user-defined factor. One region can be treated by multiple jobs and to care about 'double counting' of atoms is left to the user completely.

A hybrid setup (which is done in the JOB_AD or JOB_NAD directory) consists of some general information about the hybrid setup ('control' parameters), the definition of the atoms, some of their properties and membership to the regions and the definition of the partial jobs, which regions they are concerned with and how they shall be put together to the overall result.

The exact file-structure of a hybrid setup is explained in the NEWTON-X documentation.

8.2 Formamide with water using hybrid gradients

In this Chapter we will set up a calculation with hybrid gradients for a formamide molecule with one water molecule as solvent (which is physically not very meaningful, but keeps the input files short). We will start from standard XYZ files and set up QM/MM non-adiabatic surface-hopping dynamics trajectories. Ground and two excited states will be computed and the dynamics will start in the second excited state.

Formamide (QM part) will be treated with COLUMBUS at the SA-3-CASSCF(6,4)/6-31G* level, while the water (MM part) will be introduced by means of electrostatic embedding using OPLSAA/TIP3P-parameters[17] and TINKER for the intramolecular water-forces and the van der Waals interaction.

The initial conditions will be created with random velocities.

This kind of initial conditions is not without problems. The quantum mechanical core should in reality have a much higher kinetic energy in order to overcome the zero point energy while the environment should only get the thermal energy. There are tools and extended explanations available for the creation of hybrid initial conditions with NX.

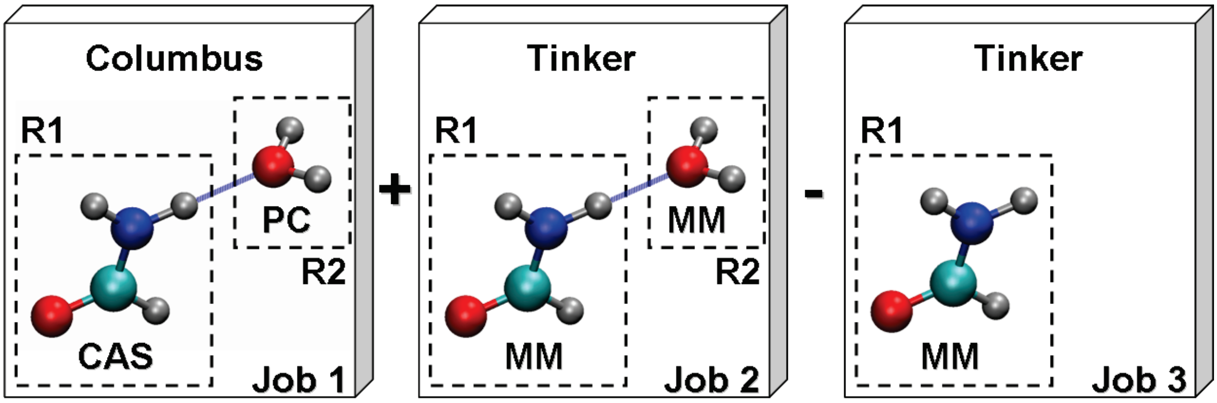


Fig. 11.QM/MM scheme for formamide (QM) and one water molecule (MM) within the electrostatic embedding.

8.3 Before starting

For NEWTON-X everything has to be setup as usual, especially the $NX- and $COLUMBUS-environment variables have to be set and the TINKER-executable 'testgrad.x' has to be in the standard path or available somewhere. See the NEWTON-X documentation on the usage of TINKER on the last topic.

Before beginning to setup a hybrid calculation one needs some additional information at hand. In the documentation for NEWTON-X ($NX/../docs/) a file 'hybrid_setup_sheet.pdf' is provided. It is an aid for having the essential information necessary in the setup of a hybrid dynamics together. For the example in this tutorial a pre-filled sheet is in Appendix 9.2. Print this sheet or display it on your screen. It contains the information to be entered in the setup.

The whole system will consist of one formamide- and one water-molecule (see Fig. 11). The formamide will be the QM-region, treated with Columbus at the SA-3-CASSCF(6,4)/6-31G* level and the water will be the second region, included via electrostatic embedding using OPLSAA/TIP3P-parameters. This means, that we will need 3 jobs

1. QM-calculation using COLUMBUS with the formamide in the geom file and the water integrated as point-charges (thus region 1+2). The value for the point-charges will be taken from the OPLSAA/TIP3P force field. The result of this job will be added to the total result. This job will give the non-adiabatic couplings and oscillator strengths.
2. MM-calculation using TINKER applying the OPLSAA/TIP3P parameters for all atoms in the system (region 1+2). It is important to make sure, that the electrostatic QM-MM interactions are **not** computed by TINKER (this will be done with special parameters for the formamide in the .key file for the tinker job). The result of this job will be added to the total result.
3. MM-calculation using TINKER with OPLSAA/TIP3P for only the formamide atoms. Again the electrostatic QM-MM interactions must not be computed. The result of this job will be subtracted from the total result.

In principle, if we used only zero force constants for the formamide intramolecular parameters the third job would not be necessary. On the one side it is easier to use the present parameters (with exception of the electrostatics) and on the other side for bigger molecules the intramolecular 1,5 and above vdW interactions cannot be excluded from a Tinker calculation (in job 2) and have to be subtracted.

8.4 Hybrid input set up

1. In the tutorial directory (see section 1.3) create a subdirectory called MD-OCNH3_H2O-NAD-QMMM:

```
> mkdir MD-OCNH3_H2O-NAD-QMMM
```

2. Move to this directory and create a directory JOB_NAD

```
> cd MD-OCNH3_H2O-NAD-QMMM
> mkdir JOB_NAD
```

3. Move to JOB_NAD and create two standard XYZ format files, one conataining the coordinates of the formamide (formamide.xyz) and another containing the coordinates of water (water.xyz):

```
> cd JOB_NAD
> vi formamide.xyz
```

```
6

   C     -0.3106169684   -0.0533771829    0.0674384499
   O      0.8916220270    0.0010327193    0.2950091787
   N     -0.9169204320   -1.2047342003   -0.1861665224
   H     -0.9363365876    0.8391191694    0.0627462758
   H     -1.9262814755   -1.2273336136   -0.3743612135
   H     -0.3679476519   -2.0465508730   -0.1892965762
```

If Columbus is one of the programs that will be used in the calculation, the hydrogen atoms should appear at the end of the file. This is the case of formamide in our example. If you have multiple regions entering the geom. file they will be appended in ascending order of the number of the regions, so your hydrogen atoms should all be at the end of the last region.

The regions, which enter the Columbus-calculation only as point charges (and thus will not be in the geom file, but in the potential.xyz file) need not be reordered. This is the case of water molecule in our example.

```
> vi water.xyz

3

   OW    -3.6514305598   -1.2731920247   -0.6968800390
   HW    -4.1044098748   -1.1659898886   -1.5379231182
   HW    -4.3744584769   -1.3471341056   -0.0677564351
```

The setup-script will use the names given in these files to distinguish between different kinds of atoms, e.g. atoms of the same atom type, but with different point charges. It is, before all, necessary to give different names to atoms that will have different point charge atoms. In this example the oxygen atom in formamide is labeled O, while the oxygen atom in water is labeled OW.

The values used in these files are the equilibrium structure of the two molecules. They will be used only to generate the input values. The initial conditions for the dynamics are prepared separately (see section 8.6).

4. Run hybrid_input_fromxyz.pl. It will ask you a series of questions. The data to be entered are on the setup sheet (Appendix 9.2).

```
> $NX/hybrid_input_fromxyz.pl

       Of how may regions consist you system?
       ? 2<Enter>          ! QM and MM

       .xyz for region 1?
       ? formamide.xyz<Enter>

       .xyz for region 2?
       ? water.xyz<Enter>

       How many jobs will be in the calculation?
       ? 3<Enter>          ! 1- Columbus: [formamide (CAS) + water (point charge)]
                           ! 2- Tinker:   +[formamide + water (MM)]
                           ! 3- Tinker:   -[formamide (MM)]

! The following sections always appear on a cleared screen
       Regions chosen for job 1:
       Choose regions for job 1.
              region    1 (  formamide.xyz) .. [ 1]
              region    2 (      water.xyz) .. [ 2]
              clear choice .................. [D]
              finished choosing ..............[X]
       ? 1<Enter>

! clear screen
       Regions chosen for job 1: 1
       Choose regions for job 1.
```

```
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? 2<Enter>

! clear screen
      Regions chosen for job 1: 1,2
      Choose regions for job 1.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? x<Enter>

! clear screen
      Which method should be used in job #1
             COLUMBUS            [1]
             TINKER              [2]
             TURBOMOLE           [3]
             ANALYTICAL MODEL    [4]
      ? 1<Enter>

      Should some region(s) be treated as pointcharges?(y/n)
      ? y<Enter>

! clear screen
      Chosen for pointcharges in job #1:
      Choose pointcharge-regions for job #1.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? 2<Enter>

! clear screen
      Chosen for pointcharges in job #1:2
      Choose pointcharge-regions for job #1.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? x<Enter>

      With which factor shall the $method-job #1 enter the result?
      ? 1<Enter>
```

The most common factors will be +1 and -1 for addition or subtraction of energies and gradients of a job to or from the total result. Other values are absolutely valid and allowed, though, even if they may not be physically sensible.

```
! clear screen
      Regions chosen for job 2:
      Choose regions for job 2.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? 1<Enter>

! clear screen
      Regions chosen for job 2: 1
      Choose regions for job 2.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? 2<Enter>
```

```
! clear screen
      Regions chosen for job 1: 1,2
      Choose regions for job 1.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? x<Enter>

! clear screen
      Which method should be used in job #2
             COLUMBUS             [1]
             TINKER               [2]
             TURBOMOLE            [3]
             ANALYTICAL MODEL     [4]
      ? 2<Enter>

      With which factor shall the $method-job #2 enter the result?
      ? 1<Enter>

! clear screen
      Regions chosen for job 3:
      Choose regions for job 3.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? 1<Enter>

! clear screen
      Regions chosen for job 3: 1
      Choose regions for job 3.
             region    1 (  formamide.xyz) .. [ 1]
             region    2 (      water.xyz) .. [ 2]
             clear choice ................... [D]
             finished choosing ...............[X]
      ? x<Enter>

! clear screen
      Which method should be used in job #2
             COLUMBUS             [1]
             TINKER               [2]
             TURBOMOLE            [3]
             ANALYTICAL MODEL     [4]
      ? 2<Enter>

      With which factor shall the $method-job #1 enter the result?
      ? -1<Enter>

! clear screen
      Which job will give the properties
      (nonadiabatic couplings, oscillator strengths, ...)?
      Choose 0 for none.
      ? 1<Enter>

      Restrict nonadibatic couplings to some regions?(y/n)
      ? y<Enter>
```

It is possible (and probably wise) to restrict the computation of the hopping-related effects to the nonadiabatic couplings of the core region. The electrostatic embedding in Columbus in principle also computes nonadiabatic coupling vectors for the point charges. In many cases (e.g. when a thermostat is acting on the MM-environment) this could give artificial results.

```
! clear screen
       Chosen for nadregions:

       Choose nadregions.
```

```
             region   1 (  formamide.xyz) .. [ 1]
             region   2 (      water.xyz) .. [ 2]
             clear choice .................. [D]
             finished choosing ............. [X]
       ? 1<Enter>

        Chosen for nadregions: 1

        Choose nadregions.
             region   1 (  formamide.xyz) .. [ 1]
             region   2 (      water.xyz) .. [ 2]
             clear choice .................. [D]
             finished choosing ............. [X]
       ? x<Enter>
```

Properties that cannot be linearly combined are all given by only one of the jobs for the whole system. This affects mainly non-adiabatic coupling vectors and oscillator strengths. The user has to take care that he selects a job which can provide them.

Now the script will ask you for each kind of atom it found in the XYZ-files for the point charge and, if the name of the atom is not an atomic symbol, for the atom-type. It knows only atom types in the periodic system of elements up to Krypton, which is the same limitation as in COLUMBUS.

The default value for charges is 0.0000 and for all atoms that are not treated as point charges by Columbus or Turbomole you can accept this value by just hitting <Enter>.

```
! clear screen
      Point charge for symbol C?[0.0000]
      ? <Enter>

! clear screen
      Point charge for symbol O?[0.0000]
      ? <Enter>

! clear screen
      Point charge for symbol N?[0.0000]
      ? <Enter>

! clear screen
      Point charge for symbol H?[0.0000]
      ? <Enter>

! clear screen
      Atom type for symbol OW?
      ? O<Enter>

      Point charge for symbol HC?[0.0000]
      ? -0.834<Enter>
! this value is taken from the OPLSAA-parameters for Oxygen in Water (TIP3P)

! clear screen
      Atom type for symbol HW?
      ? H<Enter>

      Point charge for symbol HW?[0.0000]
      ? 0.417<Enter>
```

5. At this point, several input files and directories will have been generated.

```
> ls -t

JOB_3.tinker    job_2.tinker.data  hybrid.coords        hybrid.nuclear_charges  job_1.columbus.data
JOB_2.tinker    job_3.tinker.data  hybrid.geom          hybrid.pointcharges     water.xyz
JOB_1.columbus  hybrid.add_data    hybrid_gradients.inp hybrid.xyz              formamide.xyz
```

```
hybrid.control  hybrid.atoms       hybrid.masses        job_1.columbus.control
```

From the user point of view, the most important file is hybrid_gradients.inp. It contains the complete set of information about the job.

```
> vi hybrid_gradient.inp
```

```
$job ID = 1  regions = 1,2  program = columbus  pointcharges = 2  factor = 1 $end
$job ID = 2  regions = 1,2  program = tinker  factor = 1 $end
$job ID = 3  regions = 1  program = tinker  factor = -1 $end
$control  properties = 1  natoms = 9 $end
$atoms
         C   6.0    -0.58698100    -0.10086826     0.12744020    12.00000000   0.0000   1
         O   8.0     1.68492145     0.00195156     0.55748655    15.99491464   0.0000   1
         N   7.0    -1.73272850    -2.27661770    -0.35180374    14.00307401   0.0000   1
         H   1.0    -1.76941972     1.58570542     0.11857328     1.00782504   0.0000   1
         H   1.0    -3.64014444    -2.31932440    -0.70744017     1.00782504   0.0000   1
         H   1.0    -0.69532029    -3.86742067    -0.35771869     1.00782504   0.0000   1
        OW   8.0    -6.90020375    -2.40598424    -1.31691242    15.99491464  -0.8340   2
        HW   1.0    -7.75621060    -2.20340156    -2.90625351     1.00782504   0.4170   2
        HW   1.0    -8.26652850    -2.54571452    -0.12804111     1.00782504   0.4170   2
$end
```

Refer to the Newton-X documentation for an explanation of the groups, keywords and formats in this file. It is, however, trivial to see that, for example, the line:

```
$job ID = 3  regions = 1  program = tinker  factor = -1 $end
```

is telling that JOB 3 should apply for region 1, using Tinker and that the result should be subtracted (factor = -1). The geometry is written with an extended format:

```
!    Symbol At.Number    x/bohr          y/bohr          z/bohr        Mass/amu    charge/au region
         C    6.0     -0.58698100     -0.10086826     0.12744020     12.00000000    0.0000    1
```

The usage of the program hybrid_input_fromxyz.pl to prepare the input is optional. Indeed, the user needs only to create the hybrid_gradient.inp file by any means. The remaining files can then be generated by simply executing:

```
> $NX/hybrid_read_onefile.pl
```

The same is true if the user wants to modify the job inputs. This may done either by running hybrid_input_fromxyz.pl or by directly changing hybrid_gradient.inp file and updating the remaining files with the hybrid_read_onefile.pl.

8.5 Preparation of the third-party jobs

The subdirectories for the single jobs (JOB_1.columbus, JOB_2.tinker and JOB_3.tinker) have now been created. The next step in the input procedure is to create the third-party inputs for each one of the three jobs.

8.5.a) Columbus

6. Change to the directory of the Columbus job.

```
> cd JOB_1.columbus
```

Use the geom file there to create an input for single point non-adiabatic coupling calculation with Columbus at SA-3-CASSCF(6,4)/6-31G*. The same remarks as in section 7.2 apply. The electronic structure setup-sheet can be found in appendix 9.4. Don't forget to add a line to the 'control.run' file (any position) containing the keyword 'potmat'. Refer to the COLUMBUS documentation, section 'Properties' → Solvent Effects on this topic.

You need to add the keyword potmat to the control.run only for the generation of the initial mocoef file. During runtime NEWTON-X will add the keyword automatically if you have point charges in your COLUMBUS-job.

8.5.b) TINKER

7. Here we will begin creating the tinkin.key file. This file will contain not only the usual keywords (parameters, archive, digits 8) but also the forged electrostatic parameters for formaldehyde.

   Change now to the directory of the second job (JOB_2.tinker). Copy the oplsaa.prm–file from the 'params' directory of your TINKER-installation into this directory.

```
> cd ../JOB_2.tinker
> cp <tinker_install_dir>/params/oplsaa.prm ./
```

Look up in oplsaa.prm the parameter numbers that would suite formaldehyde. In the file provided with Tinker Ver. 4 these are

```
atom     82     21     C        "NC=O Amide"               6     12.000     3
atom     83     22     O        "NC=O Amide"               8     15.999     1
atom     84     23     N        "H2NC=O Amide"             7     14.007     3
atom     87     24     H        "H2NC=O Amide"             1      1.008     1
atom    119     29     HC       "HC=O Formyl Aldehyde"     1      1.008     1
```

Copy these lines to the tinkin.key file and edit the parameter number (second column) giving numbers that do not occur yet in the parameter file (>3000 should be a safe choice). Keep the group numbers (which are used by Tinker for the identification when calculating vdW-, and intramolecular parameters). Add for each of the new atom types a “charge ### 0.000” line. Your finished file should look like this:

```
parameters oplsaa
archive
digits 8

atom   3001    21    C    "NC=O Amide"               6    12.000    3
atom   3002    22    O    "NC=O Amide"               8    15.999    1
atom   3003    23    N    "H2NC=O Amide"             7    14.007    3
atom   3004    24    H    "H2NC=O Amide"             1     1.008    1
atom   3005    29    HC   "HC=O Formyl Aldehyde"     1     1.008    1

charge 3001   0.0000
charge 3002   0.0000
charge 3003   0.0000
charge 3004   0.0000
charge 3005   0.0000
```

This will effect the formamide atoms to behave exactly as if they had gotten the original parameters with exception of the electrostatic interaction which is switched off.

Use either the geom file or the file geometry.xyz (standard XYZ in Angstrøm) to create a TINKER input for a NEWTON-X dynamics with OPLSAA-parameters. Force field numbers are (for the file provided with Tinker Ver. 4):

formamide	**C**	3001
	O	3002
	N	3003
	H-C	3004
	H-N	3005
water	**O** (OW)	186
	H (HW)	187

```
> vi tinkin.xyz
```

```
9 Tinker input
   1    C    -0.31061700    -0.05337700     0.06743800   3001     2    3    4
   2    O     0.89162200     0.00103300     0.29500900   3002     1
   3    N    -0.91692000    -1.20473400    -0.18616700   3003     1    5    6
   4    H    -0.93633700     0.83911900     0.06274600   3005     1
   5    H    -1.92628100    -1.22733400    -0.37436100   3004     3
   6    H    -0.36794800    -2.04655100    -0.18929700   3004     3
   7    O    -3.65143100    -1.27319200    -0.69688000    186     8    9
   8    H    -4.10441000    -1.16599000    -1.53792300    187     7
   9    H    -4.37445800    -1.34713400    -0.06775600    187     7
```

You can use MOLDEN for the conversion of the XYZ file to a TINKER input. This will write MM3 parameters, though. The script $NX/hybrid_tinker_paramconv.pl can help you to convert this file more easy. Edit a file 'paramconv.dat' and write on each line the parameter numbers (comma separated) which should be substituted (it is assumed here that the TINKER input with MM3 paramters has been saved to a file called geometry_tinker_mm3.xyz).

```
> vi paramconv.dat

3,3001

7,3002

5,3003

28,3004

9,3005

6,186

21,187
```

Execute the script – it will substitute every occurrence of the first parameter stated in each line with the second one and write a file with the ending .paramconv

```
> $NX/hybrid_tinker_paramconv.pl geometry_tinker_mm3.xyz

> cp geometry_tinker_mm3.xyz.paramconv tinkin.xyz
```

If you do not have the Tinker executables in your $PATH you can copy the binary testgrad.x to your job-directory. It will then be executed from there. Refer to the Tinker section of the Newton-X documentation on this topic.

You can test, whether your job is set up correct by manually executing the command line Newton-X is calling.

```
> testgrad.x tinkin.xyz -k tinkin.key y n n
```

The last output lines should be

```
Total Gradient Norm and RMS Gradient per Atom :
 Anlyt      Total Gradient Norm Value              28.92022947
 Anlyt      RMS Gradient over All Atoms             9.64007649
```

8. Change now to the directory of the third job (JOB_3.tinker). Copy the oplsaa.prm–file and the tinkin.key from the JOB_2.tinker directory.

```
> cd ../JOB_3.tinker
> cp ../JOB_2.tinker/oplsaa.prm ./
> cp ../JOB_2.tinker/tinkin.key ./
```

Use either the geom. file or the file geometry.xyz (standard XYZ in Angstrøm) to create a Tinker input for a Newton-X dynamics with the forged OPLSAA-parameters of the .key-file.

```
> vi tinkin.key
```

```
6 Tinker input
   1    C    -0.31061700    -0.05337700     0.06743800   3001    2    3    4
   2    O     0.89162200     0.00103300     0.29500900   3002    1
   3    N    -0.91692000    -1.20473400    -0.18616700   3003    1    5    6
   4    H    -0.93633700     0.83911900     0.06274600   3005    1
   5    H    -1.92628100    -1.22733400    -0.37436100   3004    3
   6    H    -0.36794800    -2.04655100    -0.18929700   3004    3
```

8.6 Initial conditions generation

There are several possible ways to set up the initial conditions for a QM/MM job. In this example this will be done by setting the initial geometry constant at the equilibrium structure and generating random velocities. Other possible ways to generate initial conditions is to perfume dynamics in the ground state and then picking random points along the trajectory to initiate the excited state dynamics. In this latter case, special care should be taken to avoid correlated points and to have a representative sample of the phase space.

The preferable and soundest way to create hybrid initial conditions is to use a Wigner distribution of the quantum harmonic oscillator for the QM-core, pick q/p points from a thermalized trajectory for the MM-environment and merge these two sets of initial conditions. This procedure is subject of a separate HOWTO.

For this example simply create initial conditions with random velocities giving a kinetic energy of 1eV.

One tool to improve the hybrid initial conditions is already included in Newton-X, $NX/hybrid_mixed initcond. It will let you set the kinetic energy of the two molecules to half of their respective zero point energy, as expected from the Virial theorem. For formamide the zero point energy is ZPE = 1.232 eV, therefore the initial kinetic energy will be assumed to be E_{kin} *= 0.616 eV. For water, ZPE = 0.559 eV and* E_{kin} *= 0.280 eV. In a more realistic example when hundreds of water molecules are included, the kinetic energy of the water could be set to 3/2*k_BT*.*

In the appendix there is a short description how to use this script for our example.

Change to the main directory (one directory above JOB_NAD) and copy the hybrid.geom file there as “geom”. Run nxinp to create the initial conditions.

```
> cd ../..
> ls
JOB_NAD
> cp JOB_NAD/hybrid.geom ./geom
> $NX/nxinp
                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7):1

                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================
                                   GENERATE INITIAL CONDITIONS


 nact:  Distribution:
        1 - Classic harmonic oscillator
        2 - Q/P-uncorrelated quantum harmonic oscillator (Wigner)
        3 - Q/P-correlated quantum harnonic oscillator
        4 - Pick points from previous dynamics
        5 - Random-velocities generation
        6 - Single point on a previous calculation
 The current value of nact  is: 2
 Enter the new value of nact  : 5

 numat:  Number of atoms.
 The current value of numat  is: 9
 Enter the new value of numat  :
 Setting numat = 9

 npoints:  Number of initial conditions to be generated
 The current value of npoints  is: 1
 Enter the new value of npoints  : 10
 Setting npoints = 10

 file_geom:  File containing the equilibrium geometry.
 The current value of file_geom  is: geom
 Enter the new value of file_geom  :
 Setting file_geom = geom

 ekin:  Kinetic energy (eV).
 The current value of ekin (eV) is: 0
 Enter the new value of ekin (eV) : 1
 Setting ekin = 1

 temp:  Temperature
 The current value of temp (K) is: 0
 Enter the new value of temp (K) :
 Setting temp = 0

 iseed:  Random number seed.
         0 - default seed value.
        -1 - set random seed.
        >0 - (integer) set specific value for the seed.
 The current value of iseed  is: 0
 Enter the new value of iseed  :
Setting iseed = 0

 lvprt:  Print level.
        1 - standard level
        2 - debug level
 The current value of lvprt  is: 1
 Enter the new value of lvprt  :
                ============================================================
                                          NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                      www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS
                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 7


                  NEWTON-X INPUT DONE

>  ls
geom  initqp_input  JOB_NAD
> $NX/initcond.pl

          ============================================================
                                     NEWTON-X
                   Newtonian dynamics close to the crossing seam
          ============================================================

                                  www.newtonx.org

    Newton-X version 1.2.4 (01/Dec/2010)
    www.newtonx.org
    Contact: barbatti@kofo.mpg.de
-----------------------------------------------------------------------
The Newton-X program is distributed free of charge for non-commercial
and non-profit uses. You may use the program freely and
adapt the code to your needs. Please, if you have any enhancements.
share them with us. You are, however, not allowed to re-distribute code
or binaries in parts or total. Anyone intending to use Newton-X must
contact us. This program is shipped to the user "as is". There is no
guarantee that it will work as described. Usage is at your own risk,
there is no liability taken for any damage or loss of data and/or money.
If you experience problems please tell the developers supplying the
version-number - they are always grateful for any hint, but bear in
mind that there is no kind of official support for Newton-X.

Cite Newton-X as:
M. Barbatti, G. Granucci, M. Ruckenbauer, J. Pittner, M. Persico, H. Lischka,
NEWTON-X: a package for Newtonian dynamics close to the crossing seam,
version 1.2, www.newtonx.org (2010).

M. Barbatti, G. Granucci, M. Persico, M. Ruckenbauer, M. Vazdar, M.
Eckert-Maksic, and H. Lischka, J. Photochem. Photobio. A 190, 228 (2007).
-----------------------------------------------------------------------


Starting Initial Condition generation

initcond.pl: paths:
Base:    /cellar/matruc/NX-devel/nx-tutorial/hybrid
Temp:    /cellar/matruc/NX-devel/nx-tutorial/hybrid/TEMP
Debug:   /cellar/matruc/NX-devel/nx-tutorial/hybrid/DEBUG


    NACT = 5: initial conditions will be generated by
    generating random velocities


initcond.pl:
       NACT      = 5
       FILE_GEOM = geom
       NUMAT     = 9
       NPOINTS   = 10
       EKIN      = 1
       TEMP      = 0
       ISEED     = 1234
       LVPRT     = 1

Cheking geometry lines


== Calling RANINCON program by B. Sellner, 2007 ==

---------------------------------------------------------------------------
Starting raninicon at Wed Mar 16 10:13:49 CET 2011
 Job done
Finished raninicon successfuly at Wed Mar 16 10:13:50 CET 2011
---------------------------------------------------------------------------

 End of initial conditions generation.
          ====================== NEWTON-X ends here ==================
> ls
DEBUG  final_output  geom  initqp_input  JOB_NAD  TEMP
```

8.7 NEWTON-X input

9. Run again nxinp and create an input for non-adiabatic simulations with 3 states, starting in state 3 with 0.5 fs timestep and 10 fs total duration. As method choose 'HYBRID GRADIENT' with COLUMBUS and TINKER as programs.

```
> $NX/nxinp

                ============================================================
                                           NEWTON-X
                         Newtonian dynamics close to the crossing seam
                                       www.newtonx.org
                ============================================================

                                MAIN MENU


                                1. GENERATE INITIAL CONDITIONS


                                2. SET BASIC INPUT


                                3. SET GENERAL OPTIONS


                                4. SET NONADIABATIC DYNAMICS


                                5. GENERATE TRAJECTORIES AND SPECTRUM


                                6. SET STATISTICAL ANALYSIS


                                7. EXIT


Select one option (1-7): 2 <ENTER>

 nat:  Number of atoms.
 The current value of nat  is: 9
 Enter the new value of nat  : <ENTER>
 Setting nat = 9
 nstat:  Number of states.
 The current value of nstat  is: 2
 Enter the new value of nstat  : 3 <ENTER>     ! 3 states will be computed
 Setting nstat = 3

 nstatdyn:  Initial state (1 - ground state).
 The current value of nstatdyn  is: 3
 Enter the new value of nstatdyn  : <ENTER>    ! dynamics will start in state 3
 Setting nstatdyn = 3

 dt:  Time step for the classical equations.
 The current value of dt (fs) is: 0.5
 Enter the new value of dt (fs) : <ENTER>
 Setting dt = 0.5

 tmax:  Total duration of the trajectory.
 The current value of tmax (fs) is: 100
 Enter the new value of tmax (fs) : <ENTER>
 Setting tmax = 100

 prog:  Quantum chemistry program and method
        0   - ANALYTICAL MODEL
        1   - COLUMBUS
        2.0 - TURBOMOLE RI-CC2
        2.1 - TURBOMOLE TD-DFT
        2.2 - TURBOMOLE ADC2    ! NX > 1.4
        3   - ACES2 EOM-CCSD (in development)
        4   - MOPAC (in development)
        5   - DFTB
        6   - GAUSSIAN
        6.0 - GAUSSIAN(03) CASSCF
        6.5 - GAUSSIAN(09) TDDFT
        7   - TINKER
        8   - DFTB+
        10.0   - GAMESS
        10.1   - GAMESS ARBITRARY ADIABATIC METHOD
        20  - HYBRID GRADIENT
 The current value of prog  is: 1
 Enter the new value of prog  : 20<ENTER>
 Setting prog = 20

Programs used for hybrid gradient
      1      - COLUMBUS
      2      - TINKER
      3      - TURBOMOLE
      99     - finished
 (default=99) 1<Enter>

Programs used for hybrid gradient
      1      - COLUMBUS
      2      - TINKER
      3      - TURBOMOLE
      99     - finished
 (default=99) 2<Enter>

Programs used for hybrid gradient
      1      - COLUMBUS
      2      - TINKER
      3      - TURBOMOLE
      99     - finished
 (default=99) <Enter>

thres: Threshold to initiate nonadiabatic coupling calculation.
          0 - Only adiabatic dynamics.
        100 - Only nonadiabatic dynamics
 The current value of thres (eV) is: 100
 Enter the new value of thres (eV) : <ENTER>
 Setting thres = 100

 Would you like to generate internal coordinate file?  (default=y) n <ENTER>
```

10. At this point the main NEWTON-X input file control.dyn has been created. You may proceed through the other options in the main menu to advanced settings. In our example the default options will be used. We proceed directly to the trajectory directories generation (option 5).

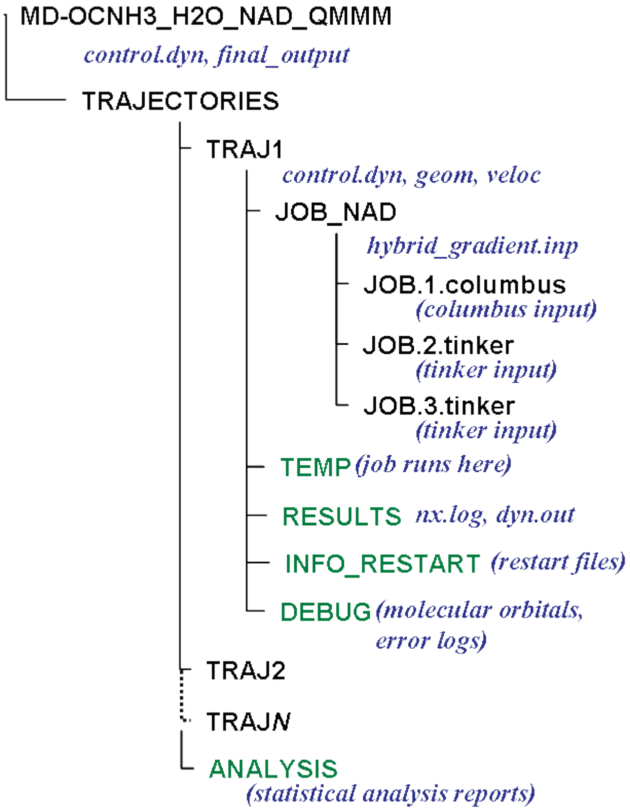


Fig. 12. Directory tree for a complete QM/MM surface hopping dynamics job with NEWTON-X. Compare to the conventional case in Fig. 7.

```
============================================================
                         NEWTON-X
         Newtonian dynamics close to the crossing seam
                      www.newtonx.org
============================================================

                MAIN MENU


                1. GENERATE INITIAL CONDITIONS


                2. SET BASIC INPUT


                3. SET GENERAL OPTIONS


                4. SET NONADIABATIC DYNAMICS
```

```
                                   5. GENERATE TRAJECTORIES AND SPECTRUM

                                   6. SET STATISTICAL ANALYSIS

                                   7. EXIT

Select one option (1-7): 5 <ENTER>
```

11. Select the job options.

```
type:  What do you want to do?
        1 - Generate spectrum
        2 - Select initial conditions for multiple initial states
        3 - Generate trajectories
        4 - Return to main menu
 The current value of type  is: 3
 Enter the new value of type  : <ENTER>
 Setting type = 3

Would you like to apply energy and probability restrictions (y/n)?  (default=n)
<ENTER>

 Will the trajectories run in a batch system? n <ENTER>

        Processing data: This may take some minutes. Please, wait...
```

In general, you might want to submit this job to a batch system. In this case, answer "y" to "Will the trajectories run in a batch system?". You will be asked the name of the submission script and nxinp will look for it existence in the input directory. Files "pmold" and "pmold-ew" in $NX/../batch/ directory are templates that can be adapted to your own batch system.

12. Select option 7 to exit nxinp.

13. The trajectories have been created in the TRAJECTORIES directory from where you can run them as usual. The final directory structure is shown in Fig. 12, including the results directory.

```
> ls

control.dyn  final_output  JOB_NAD  makedir.log  mkd.inp  TRAJECTORIES

> ls TRAJECTORIES

TRAJ1  TRAJ10  TRAJ2  TRAJ3  TRAJ4  TRAJ5  TRAJ6  TRAJ7  TRAJ8  TRAJ9

> ls TRAJECTORIES/TRAJ1

control.dyn  geom  JOB_NAD  veloc
```

14. Move to the first trajectory to execute it.

```
> cd TRAJECTORIES/TRAJ1
> $NX/moldyn.pl > moldyn.log &
```

45. After finishing the calculation, move to the RESULTS. If the job ended normally, the file nx.log should finish with:

```
moldyn.pl: End of dynamics
          ====================== NEWTON-X ends here ==================
```

15. Plot the energies (Fig. 13).

```
> cd RESULTS
> $NX/plot
Number of states:3<ENTER>
```

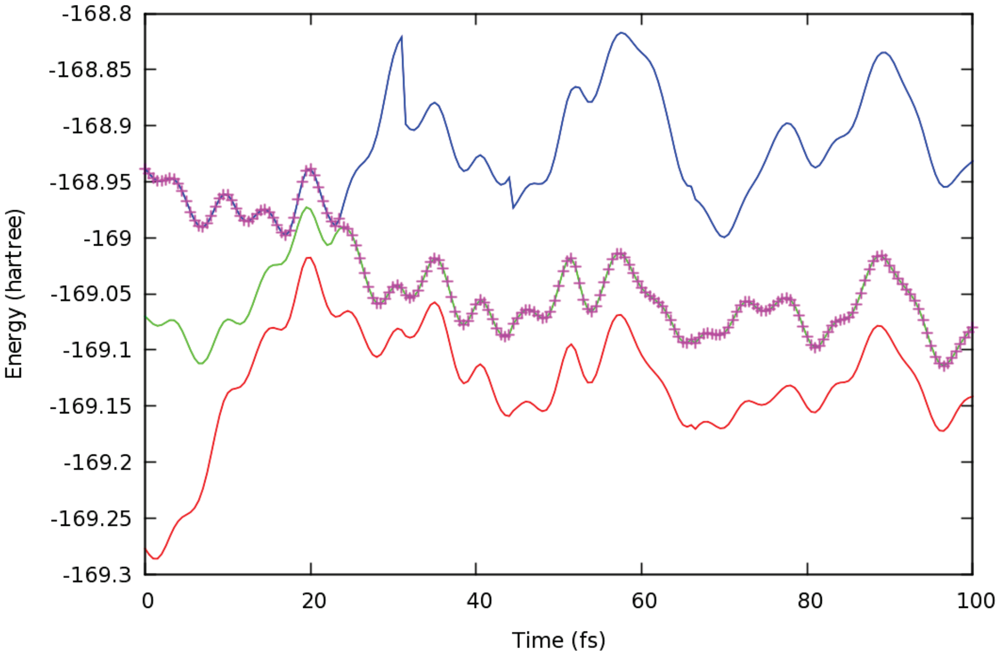


Fig. 13. Potential energy as a function of time for the first trajectory. Crosses indicate the current state. A surface hopping from S_2 to S_1 occurs at 23.5 fs.

To know more about how to make the analysis of individual trajectories and statistical analysis of several trajectories you should study Chapter 7.

9. Appendices

9.1 Methaniminium at CASSCF(12,8)

System: $CH_2NH_2^+$ Point Group: C_1

N. Electrons: 16

Level: SA-3-CASSCF(12,8)

		IRREP							
		a							
SCF	DOCC	8							
	OPSH	0							
MCSCF	DOCC	2							
	RAS	0							
	CAS	8							
	AUX	0							
MRCI	FC	2							
	FV	0							
	DOCC	0							
	ACT	8							
	AUX	0							
	INT	8							

State	Multiplicity	N. electrons	Symmetry
1	1	16	A
1	1	16	A
1	1	16	A

Number of distinct rows (DRTs): 1

	1	2	3	4	5	6	7	8
D_{2h}	a_g	b_{3u}	b_{2u}	b_{1g}	b_{1u}	b_{2g}	b_{3g}	a_u
D_2	a	b_2	b_1	b_3				
C_{2h}	a_g	b_u	a_u	b_g				
C_{2v}	a_1	b_1	b_2	a_2				
C_i	a_g	a_u						
C_s	a′	a′′						
C_2	a	b						
C_1	a							

9.2 Generation of mixed random velocities

Change back to the parent directory and call the script $NX/hybrid_mixed_initcond.pl

> *This script spares you the work to split the geom file manually, generate the initial conditions for the different regions separately, and then merge the final_output files.*

> *Note, that in this example we won't equilibrate the structure of the system. The original coordinates are optimized with OPLSAA/TIP3P-parameters, but usually one will have to equilibrate the structure to the method used in the dynamics. A damped dynamics, beginning with zero velocities is a good choice of method for doing that.*

```
> cd ../
> $NX/hybrid_mixed_initcond.pl
```

16. Answer the questions so as to generate 5 initial conditions with 2 methods. Region one (the formamide) gets halve the zero point energy, the water gets kinetic energy for 298 K.

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X                        MIXED INITIAL CONDITIONS GENERATOR                 X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

       This script allows generating initial conditions for NEWTON-X simulations
       using different methods for different parts of the system. The regions
       for the hybrid gradients must have been defined (hybrid.atoms) beforehand.

How many simulation starting points do you want to generate?
10<Enter>           ! 10 trajectories

How many different initial condition generation methods will be used?
2<Enter>            ! Separated init. cond. for formamide and water

Which regions should be treated by method 1 (separated by spaces)?
Available regions: 1 2
1                   ! First, formamide

*** Initial velocities generation for method 1 ***

For random velocities, type in two positive real numbers, the kinetic energy
energy (in eV) and the temperature (in K) in that order, with blanks in between.

If you want to generate the velocities with a different method, type
anything else (enter is enough) to open the Newton-X input program.

0.566  0            ! ZPE/2 (± 0 K standard deviation)
calling nxinp...
Generating initial conditions with method 1...

Which regions should be treated by method 2 (separated by spaces)?
Available regions: 2
2                   ! Now, water

*** Initial velocities generation for method 1 ***

For random velocities, type in two positive real numbers, the kinetic energy
energy (in eV) and the temperature (in K) in that order, with blanks in between.

If you want to generate the velocities with a different method, type
anything else (enter is enough) to open the Newton-X input program.

0.028  0            ! ZPE/2 (± 0 K standard deviation)
```

```
calling nxinp...
Generating initial conditions with method 2...
Merging into fianl_output
Done! Initial conditions succesfully generated in final_output
```

The initial conditions are written to the final_output file. It contains ten cards, each one with the same geometry but with different random velocities for the whole system formamide + water.

9.3 Datasheet for hybrid setup

System: HCONH-H2O
Date:

Number of regions: 2
XYZ-files for regions

region	1	2	3	4	5
filename	formamide.xyz	water.xyz			

Number of jobs: 3
Job-definitions:

ID	1	2	3	4	5	6	7	8
regions	1,2	1,2	1					
program	Col	Tinker	Tinker					
(pointcharges)	2	---	---					
factor	1	1	-1					

Job to give properties: 1

Atom type definitions

name	atomic symbol	point charge
C	C	---
O	O	---
H	H	---
N	N	---
Ow	O	-0.834
Hw	H	0.417

9.4 Formamide at CASSCF(6,4)

System: $HCONH_2$ Point Group: C_1

N. Electrons: 24

Level: SA-3-CASSCF(6,4)

		IRREP							
		a							
SCF	DOCC	12							
	OPSH	0							
MCSCF	DOCC	9							
	RAS	0							
	CAS	4							
	AUX	0							
MRCI	FC	9							
	FV	0							
	DOCC	0							
	ACT	4							
	AUX	0							
	INT	4							

State	Multiplicity	N. electrons	Symmetry
1	1	24	A
1	1	24	A
1	1	24	A

Number of distinct rows (DRTs):	1

10. References

[1] M. Barbatti, M. Ruckenbauer, F. Plasser, J. Pittner, G. Granucci, M. Persico, H. Lischka *WIREs: Comp. Mol. Sci.* **2014**, 4, 26-33.

[2] M. Barbatti, G. Granucci, M. Ruckenbauer, F. Plasser, R. Crespo-Otero, J. Pittner, M. Persico, H. Lischka *NEWTON-X: a package for Newtonian dynamics close to the crossing seam*. **2013**, www.newtonx.org.

[3] M. Barbatti, G. Granucci, M. Persico, M. Ruckenbauer, M. Vazdar, M. Eckert-Maksić, H. Lischka *J. Photochem. Photobiol., A*. **2007**, 190, 228-240.

[4] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. Montgomery, J. A., J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, N. J. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, D. J. Fox *Gaussian 09, Revision D.01. Gaussian, Inc., Wallingford CT*. **2013**.

[5] H. Lischka, R. Shepard, F. B. Brown, I. Shavitt *Int. J. Quantum Chem.* **1981**, S15, 91-100.

[6] H. Lischka, R. Shepard, I. Shavitt, R. M. Pitzer, M. Dallos, T. Müller, P. G. Szalay, F. B. Brown, R. Ahlrichs, H. J. Boehm, A. Chang, D. C. Comeau, R. Gdanitz, H. Dachsel, C. Ehrhardt, M. Ernzerhof, P. Höchtl, S. Irle, G. Kedziora, T. Kovar, V. Parasuk, M. J. M. Pepper, P. Scharf, H. Schiffer, M. Schindler, M. Schüler, M. Seth, E. A. Stahlberg, J.-G. Zhao, S. Yabushita, Z. Zhang, M. Barbatti, S. Matsika, M. Schuurmann, D. R. Yarkony, S. R. Brozell, E. V. Beck, J.-P. Blaudeau, M. Ruckenbauer, B. Sellner, F. Plasser, J. J. Szymczak *COLUMBUS, an ab initio electronic structure program, release 5.9.2*. **2008**, www.univie.ac.at/columbus.

[7] R. Ahlrichs, M. Bär, M. Häser, H. Horn, C. Kölmel *Chem. Phys. Lett.* **1989**, 162, 165-169.

[8] J. W. Ponder, F. M. Richards *J. Comput. Chem.* **1987**, 8, 1016-1024.

[9] R. Crespo-Otero, M. Barbatti *Theor. Chem. Acc.* **2012**, 131, 1237.

[10] J. C. Tully *J. Chem. Phys.* **1990**, 93, 1061-1071.

[11] S. Salzmann, M. Kleinschmidt, J. Tatchen, R. Weinkauf, C. M. Marian *Phys. Chem. Chem. Phys.* **2008**, 10, 380-392.

[12] R. Weinkauf, L. Lehr, E. W. Schlag, S. Salzmann, C. M. Marian *Phys. Chem. Chem. Phys.* **2008**, 10, 393-404.

[13] G. Granucci, M. Persico *J. Chem. Phys.* **2007**, 126, 134114-134111.

[14] D. Bakowies, W. Thiel *J. Phys. Chem.* **1996**, 100, 10580-10594.

[15] P. Sherwood in *Hybrid Quantum Mechanics/Molecular Mechanics, Vol. 3* (Ed. J. Grotendorst), John von Neumann Institute for Computing, Jülich, **2000**, pp.285-305.

[16] H. Lin, D. G. Truhlar *Theor. Chem. Acc.* **2007**, 117, 185-199.

[17] W. L. Jorgensen, N. A. McDonald *J. Mol. Struct. THEOCHEM*. **1998**, 424, 145-155.