ABIN: AB INITIO Molecular Dynamics *(and much more)*

version 1.0 coming soon*(ish)*

Daniel Hollas

FIRST PHOTOX seminar, ICT Prague

17.10.2014
(Really) Brief History of ABIN

- 2007(ish) – P. Slavíček&M. Ončák: "Let’s make a code for *ab initio* PIMD!" (not really sure about this)
- 2011 – D. Hollas: "I’ll take it from here."
- 2014 – First stable version v1.0 (hopefully)
What is ABIN?

Basic philosophy of the program

Let me take care of the time evolution of the system and take the energies and forces from external (ab initio) program (e.g. Gaussian).
What can it do?

- Classical MD in the gas phase. Temperature is regulated using Nosé-Hoover chain thermostat.
- Path Integral MD.
- Surface hopping algorithm. This code is currently interfaced with Molpro and GAMESS.
- Simulations with quantum thermostat and PI+GLE methods.
- Two-layer QM/MM ONIOM interface.
- Constraints using SHAKE algorithm (currently only for classical MD).
- Simple minimization using the steepest descent method.
Input files

1. Initial structure. This should be an XYZ file, typically named `mini.dat`.
2. Input file with all the parameters, typically named `input.in`.
3. Bash interface for the external program e.g. G09/r.g09.
Output files

- Unlike other programs, ABIN creates several different output files.
- They can be all directly visualized by Xmgrace.
- First line is a header(#) which tells you, which column is which.
- In the beginning of the simulations, all parameters are printed into standard output.
- Standard output files:
  movie.xyz, temper.dat, energies.dat
Understanding the interface

What it does:

1. Takes the geometry from AB\text{\textbackslash N} (\texttt{geom.dat.001}) and creates the input for the ab initio program.
2. Launches the program.
3. Collects the energies and gradients from the output and hands them to AB\text{\textbackslash N} via file \texttt{engrad.dat.001}.
   You have to be careful, whether you get the right numbers! For example in G09, this will work for B3LYP, but not for MP2!

\texttt{grep "SCF done" input.log} > \texttt{../engrad.dat}
Outline

1. ABIN
2. Classical MD
3. Surface Hopping
4. Path Integral MD
5. Quantum thermostat
6. utilities
Gurus in the field: Isaac Newton, Jiří Kolafa

We are solving Newton’s equation and use thermostat to keep it cool.

\[ F = ma \]  

Basic input is coming next (all sample input files are in directory SAMPLE/)  
(btw: atom masses are now automatically recognized for all elements and deuterium)
Classical MD

&general
  pot='g09'
  natom=27,
  ipimd=0,
  imini=2000,
  nstep=50000,
  dt=20.,
  irandom=1651563,
  irest=0,
  nwrite=1,
  n writex=5,
  nrest=5,
  nwritev=0,
/

&nhcopt
  temp=298.15,
  inose=1,
  tau0=0.001,
/

! this is the beginning of a namelist
! comments are after exclamation marks
! number of atoms
! classical simulation 0, quantum simulation 1, surface-hopping 2, steepest descent 3
! equilibration period (should be at least 2000)
! number of steps
! timestep [au]
! random seed
! should we restart from restart.xyz? (ignoring mini.dat)
! how often some output should be printed (estimators etc.)
! how often should we print coordinates?
! how often we print restart files?
! how often we print velocities?
! temperature [K] for Maxwell-Boltzmann sampling and thermostat
! Thermostating: Nose-Hoover 1, microcanonical 0, GLE 2
! relaxation time of NHC thermostat
Outline

1. ABIN
2. Classical MD
3. Surface Hopping
4. Path Integral MD
5. Quantum thermostat
6. utilities
Surface Hopping

What about the photodynamics?

Gurus in the field: John C. Tully, Ondřej Svoboda

```
&nhcopt
inoose=0, ! NO THERMOSTAT!
temp=0.00, ! Usually, you would take initial velocities from WIGNER.
/
&sh
istate_init=2, ! initial electronic state
nstate=3, ! number of electronic states
deltaE=2.0, ! maximum energy difference [eV], for which we calculate NA coupling
PopThr=0.001, ! minimum population of either state, for which we compute NA coupling
EnergyDifThr=1.00, ! maximum energy difference between two consecutive steps
EnergyDriftThr=1.00, ! maximum energy drift from initial total energy
substep=100, ! number of substeps for solving ESCH
/```

Daniel Hollas (ICT Prague)
Outline

1. ABIN
2. Classical MD
3. Surface Hopping
4. Path Integral MD
5. Quantum thermostat
6. utilities
Gurus in the field: Richard Feynman, Mark E. Tuckerman, Petr Slavíček
Path Integral MD

Gurus in the field: Richard Feynman, Mark E. Tuckerman, Petr Slavíček

- Within PIMD, quantum effects are captured via classical simulations of an extended system.

![Diagram showing quantum force and isomorphism](image)
Path Integral MD

Gurus in the field: Richard Feynman, Mark E. Tuckerman, Petr Slavíček

- Within PIMD, quantum effects are captured via classical simulations of an extended system.

- Computational cost scales linearly with number of beads. Tens of beads are typically needed (prohibitive for ab initio MD!).
PIMD setup

&general
.
.
.
ipimd=1,       ! PIMD = 1
nwalk=20,      ! number of beads
istage=1,      ! istage must be 1 for PIMD
nproc=10       ! Yes, there is a parallel version!
! nwalk must be divisible by nproc!
.
.
\
1. ABIN
2. Classical MD
3. Surface Hopping
4. Path Integral MD
5. Quantum thermostat
6. Utilities
We need a cheaper method!
Gurus in the field: Michele Ceriotti, Michele Parrinello, Daniel Hollas
Quantum thermostat – first trick

- Look at the harmonic oscillator in thermal equilibrium.

\[ \rho_c(x) \approx e^{\frac{m\omega^2(x-x_0)^2}{k_B T}} \]

\[ \sigma_c^2 = \frac{k_B T}{m\omega^2} \]

\[ \rho_q(x) \approx e^{\frac{m\omega(x-x_0)^2}{\hbar \coth \frac{\hbar \omega}{2k_B T}}} \]

\[ \sigma_q^2 = \frac{\hbar}{2m\omega \coth \frac{\hbar \omega}{2k_B T}} \]

\[ \sigma_q^2 = \frac{k_B T^*}{m\omega^2} \]

\[ T^* = \frac{\hbar \omega}{2k_B} \coth \frac{\hbar \omega}{2k_B T} \]
Quantum Thermostat – second trick

- We need a frequency dependent temperature.
- Use Generalized Langevin Equation as a thermostat which maintains normal modes of different frequencies on different temperature.
Quantum Thermostat – second trick

- We need a frequency dependent temperature.
- Use Generalized Langevin Equation as a thermostat which maintains normal modes of different frequencies on different temperature.

\[
\dot{q} = \frac{p}{m},
\]

\[
\begin{pmatrix}
\dot{p} \\
\dot{s}
\end{pmatrix} = \begin{pmatrix}
-U'(q) \\
0
\end{pmatrix} - \begin{pmatrix}
a_{pp}a_p^T \\
\bar{a}_pA
\end{pmatrix} \begin{pmatrix}
p \\
s
\end{pmatrix} + \begin{pmatrix}
b_{pp}b_p^T \\
b_pB
\end{pmatrix} \begin{pmatrix}
\xi
\end{pmatrix},
\]
Quantum Thermostat – Pros and Cons

Advantages

- Adds only negligible overhead to classical MD.
- Accurate for harmonic systems.
- Gives reasonable answers for slightly anharmonic systems.
Quantum Thermostat – Pros and Cons

Advantages

- Adds only negligible overhead to classical MD.
- Accurate for harmonic systems.
- Gives reasonable answers for slightly anharmonic systems.

Disadvantages

- Parametrization of GLE matrices is not trivial.
- **Danger of ZPE leakage for anharmonic systems!** Energy may flow from high to low frequency normal modes.
It is possible to combine PIMD with GLE thermostat → PI+GLE method.

GLE dramatically improves the convergence to the exact quantum limit. Only 4 beads are typically needed for nuclear distributions!

This is what we typically use at the moment.
GLE set-up

In your input:

ipimd=0 !( for quantum thermostat)
ipimd=1 !( for PI+GLE)
inose=2

- Parameters can be downloaded from: https://epfl-cosmo.github.io/gle4md/index.html?page=matrix
  Matrix A (a.u.) and matrix C (eV) should go to the files GLE-A and GLE-C. Put the parameter $N_s$ in the first line (this is the number of additional thermostat variables, i.e. dimension of the matrices - 1).

- What do you need to know in advance?
  Temperature and highest frequency in your system.
Outline

1. ABIN
2. Classical MD
3. Surface Hopping
4. Path Integral MD
5. Quantum thermostat
6. utilities
In the following slides, I will briefly present some handy scripts for an efficient use of ABIN and for analysis of the simulations. Some of these scripts must be in your $PATH (e.g. /home/$USER/bin). Particularly:

fetchabin.sh, ExitAbin.sh, prum.awk, checkenergy.awk, Mylrandom
Launching ABIN on clusters

r.abin, fetchabin.sh, ExitAbin.sh

**r.abin**

The launching script for the job queue. Copies all the data from the current directory to the scratch disc on the node, launches ABIN simulation, and then copies data back.

Usage: `qsub -cwd -pe shm nproc r.abin`

**fetchabin.sh**

Copies current data from the scratch disc to the server, so that you can look at the data during the simulation.

Usage: `cd DIRECTORY_WITH_YOUR_SIMULATION ; fetchabin.sh`

**ExitAbin.sh**

Ends the ABIN job nicely (without using qdel). ABIN checks at the beginning of each step, whether file EXIT exist in the working directory. If it does, ABIN prints out most recent restart file and exits.

Usage: `cd DIRECTORY_WITH_YOUR_SIMULATION ; ExitAbin.sh`
Analyzing trajectories
analyze_movie, analyzeSH.sh

**analyze_movie**

Fortran code for analyzing the bond distances, angles and dihedral angles from the xyz trajectory. It prints out both the time evolution and histograms. Type the following to get help:

```
./analyze_movie -h
```
There is a similar program (analyze_wigner) to analyze the geometries from Wigner transformation.

**analyzeSH.sh**

Script for analysis of Surface Hopping trajectories that prints out electronic populations and checks the energy conservation in each trajectory.
Generating initial conditions

wigner_sampling.sh

You can generate initial conditions for the multiple trajectories using Wigner transformation. Check HARMONIC-SAMPLING/README for details how to do that. (btw: These scripts require a working installation of MOLPRO12) It basically involves the following:

1. Optimize the geometry and calculate the frequencies. (Currently must be done in MOLPRO)

2. Convert the frequency output via script mp2freq into file Frequencies.dat. Then you need to dispose the low frequencies below 500$cm^{-1}$ by manually editing the file Frequencies.dat

3. Edit and launch script wigner_sampling.sh
Launch multiple trajectories at once!
Where do we get initial conditions?

1. From a set of geometries (one xyz trajectory file)
2. From a Wigner distribution via restart file.
Misc
cut_sphere, PickGeoms.sh, RecalcGeometries.sh

cut_sphere
Cut out a cluster of solvent molecules around a solute from a larger cluster. You can either take the given number of closest solvent molecules or all solvent molecules within a given radius.

PickGeoms.sh
Pick a given number of geometries out of the xyz trajectory. Do it randomly or by using fixes offset (e.g. take each 10th geometry). The output is again a xyz trajectory.

RecalcGeoms.sh
Make some calculations on all the geometries from a xyz trajectory. Current version works with Gaussian, but can be easily modified.
Any ideas/criticism?

Ask not what ABIN can do for you — ask what you can do for ABIN.
If you have any ideas for improvement/criticism, please tell me now or rema...
(or really any time you want, I won’t take it personally)

Possible projects (any volunteers?):

- We should really have our own code for Wigner transformation. The current solution is clumsy and requires MOLPRO.
- In the meantime, somebody could write the script for the conversion of normal modes and frequencies from Gaussian to the MOLPRO format.
- More tools for analyzing trajectories.
What about your know-how?

- We often seem to invent the same thing over and over again.
- We should have some platform to share out knowledge. How about...?
- Any web guru here to make it look nicer?
- Please, email me anything you might want to add, even if it seems too technical and unimportant. Some day, some other person may stumble upon the same problem!