XMDYN manual

Zoltan Jurek
Center for Free-Electron Laser Science (CFEL), DESY, Hamburg, Germany

08.12.2016

1. What is XMDYN?

XMDYN [1] is a computer package for simulations of the dynamics of matter exposed to intense x-ray radiation. XMDYN is a microscopic approach that tracks the time evolution of all atoms, ions and electrons in the sample. A Monte Carlo algorithm is used to generate stochastic ionization events, while the Molecular Dynamics technique is applied to follow the real space dynamics of the atoms, ions and ionized electrons. The main ionization mechanisms and decay processes that occur to atoms in a molecular or cluster environment as a consequence of x-ray irradiation are included, such as photoionization, Auger and fluorescent decays and electron impact (collisional) ionization. Further, three-(or many-)body recombination is included that may occur during the classical dynamics.

Fundamental model features and approximations:

- The sample is described as a set of electronically independent atoms in ground state. The code follows how the occupation number of the atomic orbitals changes due to ionization events, atomic decays or recombination.

- Atoms, ions and ionized electrons are treated as classical particles interacting via Coulomb forces. The real space propagation is done by solving Newton's equation of motion numerically.

- Atomic parameters such as electron binding energy, cross section and rate data are calculated by the XATOM toolkit [2].

- Chemical details of the samples are neglected in the released version of XMDYN, therefore the code is applicable for the high x-ray intensity (highly ionized matter) case. Such conditions are present typically at X-ray Free-Electron Laser (XFEL) experiments in the beam focus.

More detailed description of the methodology can be found in refs. [1,3,4].

XMDYN is available as part of the XRAYPAC package at http://www.desy.de/~xraypac

2. Installation

XMDYN is distributed as a statically linked 64-bit linux binary and can be run from the command line. It needs no special installation procedure, however, in case of no pre-calculated atomic data tables (see later) the binary 'xatom' must be callable for XMDYN. To make the binary 'xatom' located e.g. in the folder /home/foo/bin available one should add the folder to the search path, e.g. in the bash shell:

```
    export PATH=/home/foo/bin:$PATH
```

This line could be added to a shell configuration file to make it executed each time a new shell is launched. E.g. in case of the bash shell such a file is ~/.bash_profile or ~/.bashrc.

3. Running XMDYN

The input files should be located in a folder, e.g. /home/foo/xmdyn/test1, and organized the following way:

```
    /home/foo/xmdyn/test1/input.txt
    /home/foo/xmdyn/test1/sample/r.txt
    /home/foo/xmdyn/test1/sample/Z.txt
    /home/foo/xmdyn/test1/sample/v.txt (optional)
```

Running a simulation from the command line:

```
    xmdyn /home/foo/xmdyn/test1
```

The output files are saved in the same folder (in this example in /home/foo/xmdyn/test1).

4. Input file structure

The input text file ‘input.txt’ contains settings such as the XFEL pulse parameters, propagation related parameters (e.g. timestep, number of steps, snapshots) and other control parameters.

```
Format:
        INPUTFIELD      VALUE
Comment character: #
```

If the same input field is specified more than one time within input.txt, the value of the last definition is used in the calculation.

```
############################################################################
##########  A minimalistic XMDYN input file for 1ps propagation  ##########

### Propagation, snapshots
DT              1e−18           # timestep [s]
STEPS           1000000         # number of timesteps
SNP             90000:1000:110000,150000:50000:      # snapshots of the system
SNP_E           0:1:20000       # snapshots of energy for monitoring
                                # energy conservation
PROGRESS        1000            # progress logging

### X−ray pulse parameters
EPH             485             # photon energy [eV]
NPH             1e+10           # fluence [1/(micrometer)^2]
T               10e−15          # pulse duration Full−Width−Half−Maximum [s]
T0              20e−15          # center of beam in time [s]
PROFILE         gaussian        # temporal pulse profile: step or gaussian

### Blocks, processes on/off (y/n)
#md              n              # MD−block on/off
#ph              n              # photoionization on/off
#fl              n              # fluorescent decay on/off
#au              n              # Auger decay on/off
#si              n              # secondary ionization on/off
#re              n              # recombination on/off

### Misc
R0              1e−10           # Coulomb softening parameter (see [1])
RANDSEED        2               # random seed for generating different
                                # realizations of the sample dynamics

############################################################################
```

```
# Note: SNP and SNP_E is a combination of numbers and ranges separated by
#       commas. E.g. in case of
#            SNP     10,15,20:2:40,50,100:100:
#       snapshot are saved at steps 10,15,20,22,24,...,40,50,100,200,300,...
#       until the end of the run. ‘A:B:C’ format can be anywhere within the
#       string , while format ‘A:B:’ can appear only as the very last item.
```

5. Files describing the sample

The subfolder ‘sample’ contains files describing the sample, e.g. for a system of N atoms with type and data size:

Filename	Description [unit]	datatype	data size
sample/Z.txt:	atomic numbers	float	Nx1 numbers
sample/r.txt:	position data [m]	float	Nx3 numbers
sample/v.txt	velocity data [m/s]	float	Nx3 numbers

If no sample/v.txt files exists, all velocities are set to zero initially.

E.g. for a diatomic system the files may contain:

```
   sample/r.txt:
                  0.0e-10   0.0e-10   0.0e-10
                  1.5e-10   0.0e-10   0.0e-10
   sample/Z.txt:
                  6 8
```

6. Output files

Subfolders created and files saved during a run:

dbase/	folder containing database of the on-the-fly calculated parameters
dbase/02-atm/<Z>/<config>	atomic parameters saved
dbase/02-pcs/<EPH>/<Z>/<config>	photoionization parameters saved
dbase/02-scd/<Z>/<config>	atomic form factors
snp/	folder containing subfolders of snapshot data of the system taken at different steps together with other propagation related data
snp/snp_times.dat	simulation times when snapshots are saved in increasing order
snp/snp_beam.dat	x-ray fluence delivered during one timestep at the times when snapshots are saved
snp/all_energy.dat	snapshots of different energy terms in the simulation saved according to SNP_E. It can be used for tracking energy conservation

A snapshot folder example:

snp/0000100/	folder of snapshot data taken at step 100
snp/0000100/r.dat	real space coordinates of classical particles
snp/0000100/v.dat	velocity data of classical particles
snp/0000100/q.dat	charge data of classical particles
snp/0000100/m.dat	mass data of classical particles
snp/0000100/Z.dat	atomic numbers of classical particles (0 for electrons)
snp/0000100/uid.dat	unique ids of particles (useful for tracking)
snp/0000100/T.dat	atomic types
snp/0000100/econf.dat	atomic number and electronic configuration of atom types
snp/0000100/f0.dat	x-ray elastic scattering form factors for the atom types present in the current snapshot. Saved only if ‘scd’ is specified in input.txt (see 7.1)

snp/0000100/Q.dat: scattering vector values in atomic units for which f0 is tabulated (see: XATOM manual). Saved only if ‘scd’ is specified in input.txt (see 7.1)

events.log: logfile containing all events (e.g. photoionization, Auger decay, secondary ionization, etc.) occurred during the simulation run with their respective data (e.g. timestep of occurrence, atom id, electron ids, kinetic energies of particles involved, etc.)

progress.log: log of runtime related data: memory usage and time

xparams.dat: file listing the values of all input parameters one can set in input.txt. Fields not defined in input.txt are also printed with their default values.

zzz-commandline.txt: command used to launch the simulation run

Atom type: atoms of the same atomic number and electronic configuration are classified as to have the same atom type. Integers 0,1,2,... are used to identify these classes. Atom types 0,1,2,... are defined by rows 1,2,3... of econf.dat, respectively. Similarly, corresponding atomic form factors are listed in rows of file f0.dat. In a snapshot only the present atom types are listed, e.g. a class of a given electronic configuration of a given atomic number may have different atom type ids in different snapshots.

7. Miscellaneous

7.1 Saving x-ray elastic scattering form factor data

By default XMDYN saves the dynamics related data only in the snapshot folders. In order to save form factors as well, the following is to be specified in input.txt:

```
DBASE_TYPE        atm,pcs,scd
```

7.2 Reusing calculated XATOM data

Atomic data files from the folder ‘dbase’ of different XMDYN runs may be collected in one folder (e.g. /home/foo/xmdyb/dbase/) keeping the relative directory structure. In that case the data can be reused in later calculations, e.g. XMDYN loads the data from the files rather than calling XATOM to perform calculations. To make XMDYN find the database the following is to be added to input.txt:

```
DBASEDIR          /home/foo/xmdyn/dbase
```

7.3 Specifying non-default XATOM options

Although the default settings of XATOM are optimized for performance and accuracy, one may want to specify other XATOM command line options (see XATOM manual) when it is called from XMDYN. This can be done within the file input.txt, e.g.:

```
XATOM_OPTS        -xmdyn -rmax 100
```

Note that the XATOM option ‘-xmdyn’ (which is the default value when XATOM_OPTS is not specified) must be always present.

7.4 Builtin sample definition

One may run a simulation without defining a system via files in subfolder ‘sample’. The following settings in input.txt define a system of 23 carbon atoms (Z=6) on a simple cubic lattice of lattice constant 2.5 Angstrom:

```
N             23
Z             6
DIST          2.5e−10
```

Note that the system must be specified either via the folder ‘sample’ or in the file ‘input.txt’.

7.5 Changing the initial electron−ion distance at ionization

By default the ionized electrons are launched at 0.5 Angstrom distance from the mother ion. This value can be changed via input.txt:

```
RSTARTE       1.0e−10
```

7.6 Energy conservation and numerical error in propagation

Energy data is saved in the file snp/all_energy.dat at steps defined by SNP_E in input.txt. A python script ‘xmdyn_energy.py’ is provided that loads the data and visualizes different energy terms in the system. Running the script from the linux command line, e.g.:

```
python ./xmdyn_energy.py  path_to_the_data/snp/all_energy.dat
```

figures are saved in pdf format, while running it within a(n) (i)python console, e.g.:

```
run ./xmdyn_energy.py
xmdyn_energy_conservation( ’path_to_the_data/snp/all_energy.dat’ )
```

it shows the figures on screen without saving them in files.

The total energy of the system ‘E_total’ is partly stored in the classical subsystem (‘E_class’, the sum of the classical kinetic and potential energies) and partly in the atoms and ions by their bound electrons (‘E_bound’). The total energy of the system can change only due to photoabsorption or by emitting fluorescent photons. Ionization and recombination processes can transfer energy from the bound electrons to the classical system and vica versa while E_total remains constant. In the first figure (saved as xmdyn−DeltaE.pdf) the overlapping curves ‘E_total,theory’ and ‘E_total,ref’ show the time dependence of E_total. The curve of E_total,calc should agree well with the previous two, but due to the numerical error that arises unavoidably in the propagation some discrepancy is always observable. The same error appears between E_class,ref and E_class,calc reflecting that the source of the numerical error is from the classical molecular dynamics module. The figure also shows the energy stored by the bound electrons.

The two other figures saved as xmdyn−MD−E_error.pdf and xmdyn−MD−E_rel_error.pdf show the absolute and relative numerical error of E_class as a function of timesteps, respectively.

If the energy conservation is strongly violated, smaller timestep should be specified.

8. References

[1] Z. Jurek, S.−K. Son, B. Ziaja and R. Santra, J. Appl. Cryst. 49, 1048 (2016)
[2] S.−K. Son, L. Young, and R. Santra, Phys. Rev. A 83, 033402 (2011)
[3] Z. Jurek, G. Faigel, M. Tegze, Eur. Phys. J. D 29, 217 (2004)
[4] B. F. Murphy et al, Nat. Commun. 5, 4281 (2014)

==